



Installation der SCANLAB XL SCAN-Komponenten und Erstinbetriebnahme des XL SCAN-Systems

syncAXIS control **V1.8.0**

SCANLAB GmbH
Siemensstr. 2a
82178 Puchheim
Deutschland

Tel.+49 (89) 800 746-0
Fax+49 (89) 800 746-199

info@scanlab.de
www.scanlab.de

© SCANLAB GmbH
(Doc. Rev. 2.0.4 de-DE - 2022-08-25)

SCANLAB GmbH behält sich vor, dieses Dokument jederzeit und ohne Ankündigung inhaltlich zu aktualisieren.
Kein Teil dieses Dokuments darf in irgendeiner Form (Fotokopie, Druck, Mikrofilm oder in einem anderen Verfahren) ohne ausdrückliche schriftliche Genehmigung der SCANLAB GmbH reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Alle erwähnte Marken unterliegen dem Markenschutz der jeweiligen Markeninhaber.

Inhalt

1	Über dieses Handbuch	5
1.1	Weiterführende Dokumente	5
1.2	Hersteller	5
1.3	Glossar	6
2	Systemübersicht	8
2.1	Typischer 1-Kopf-Aufbau	8
2.2	Mehr-Kopf-Aufbau (4 Scan-Köpfe)	9
2.3	Typischer Lieferumfang (SCANLAB-XL SCAN-Komponenten)	10
2.4	Nicht-SCANLAB-Komponenten (Voraussetzungen)	13
3	Die syncAXISConfig.xml prüfen und anpassen	15
4	Installieren der SCANLAB-XL SCAN-Komponenten	17
4.1	Nicht-SCANLAB-Komponenten vorbereiten	17
4.2	excelliSCAN(s) installieren	17
4.3	Gilt nur mit RTC6 PCI-Express-Karten – Windows-PC mit RTC6-Komponenten erweitern	18
4.4	Gilt nur mit RTC6 Ethernet-Karten – RTC6 Ethernet-Karte installieren	18
4.5	RTC6-Karte(n) mit anderen Komponenten verbinden	19
4.6	syncAXIS-Software auf dem Windows-PC entpacken und SCANLAB-USB-Dongle einstecken	19
5	Inbetriebnehmen des XL SCAN-Systems – Software	20
5.1	Installation_Project.exe vorbereiten	20
5.1.1	Alternative 1 – “Installation_Project” für MS Visual Studio erstellen und Installation_Project.exe kompilieren	20
5.1.2	Alternative 2 – Mitgelieferte Installation_Project.exe verwenden	21
5.2	Installation_Project.exe starten und syncAXIS control-Instanz im Simulationsmodus initialisieren	22
5.3	“TEST_MARKING” im Betriebsmodus “ScannerAndStage” simulieren	24
6	Inbetriebnehmen des XL SCAN-Systems – Hardware	26
6.1	Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren	26
6.2	Prüfen, ob Laserstrahlung ausgegeben wird	28
6.3	Überprüfen der korrekten Kommunikation mit Scan-Kopf und Verfahrtsch	29
6.4	Den (vorgängig in 5.3) simulierten “TEST_MARKING” markieren	30
7	Optimierung und Verifikation	32
7.1	Optimierung – Laser-Delays finden, die zu Ihrer Anwendung passen	33
7.2	Optimierung – Das Scan-Kopf-Arbeitsfeld kalibrieren	36
7.3	Verifikation – Genauigkeit der statischen Kalibrierungen von Scan-Kopf und ACS-Achsen prüfen	39
7.4	Verifikation – Synchronisation aller System-Komponenten prüfen	41
7.5	Verifikation – Genauigkeit der kombinierten Bewegung bestimmen	43
8	Anhang A: RTC6 PCI-Express-Karte – Steckverbinder	46
8.1	SCANHEAD	46
8.2	LASER	46
9	Anhang B: SSHC-Slotblech (#115132)	47
9.1	Ansicht (nicht eingebaut)	47
9.2	Steckverbinder	47

10 Anhang C: syncAXIS control-Softwarepaket – Entpackt	48
11 Anhang D: Über das "Installation_Project"	52
12 Anhang E: Kabel für Datenübertragung gemäß SL2-100-Protokoll – Anforderungen	56
13 Anhang F: Scan-Kopf-Arbeitsfeld kalibrieren – Alternative mit der CalSheet-Software	57
13.1 Definieren zweier unterschiedlicher Kalibriermuster	57
13.2 Markieren der zwei Kalibriermuster	58
13.3 Generieren der optimierten ct5-Datei mit der CalSheet-Software	58
14 Anhang G: Über das Error Handling mit syncAXIS control	59
14.1 Rückgabewerte der syncAXIS control-Funktionen	60
14.2 slsc_ctrl_get_error	61
14.3 Logging in syncAXIS control	62
14.4 Tipps zum Error Handling mit syncAXIS control	62
15 Änderungsindex	63

1 Über dieses Handbuch

XL SCAN besteht aus Systemkomponenten verschiedener Hersteller (SCANLAB, ACS, Verfahrtschhersteller).

Dieses Handbuch beschreibt:

- Diejenigen Komponenten, die von SCANLAB für ein XL SCAN-System geliefert werden: Scan-Köpfe mit Objektiv, RTC6-Karten, **SCANLAB-USB-Dongle** und Software zur synchronen Ansteuerung von Scan-Kopf, Verfahrtsch und Laser (siehe **Kapitel 2.3 "Typischer Lieferumfang (SCANLAB-XL SCAN-Komponenten)"**, Seite 10),
- Sonstige Komponenten eines Mehr-Achsen-Scan-Systems, die typischerweise zusammen mit den SCANLAB XL SCAN-Komponenten zu einem Gesamtsystem integriert werden (siehe Übersicht in **Abbildung 1, Seite 8** und **Kapitel 2.4 "Nicht-SCANLAB-Komponenten (Voraussetzungen)"**, Seite 13),
- Die Installation und Erstinbetriebnahme des Scan-Systems unter Verwendung des mitgelieferten "Installation_Project". Weiterer Zweck des "Installation_Project" ist, Systembetreibern Quellcode zur Verfügung zu stellen, der die Implementierung der syncAXIS control-Software zur Steuerung eines XL SCAN-Systems demonstriert (z. B. wie erfolgt das Aufbauen einer **syncAXIS control-Instanz**, wie können Markiermuster codiert und ausgeführt werden).



Vorsicht!

Lesen und befolgen Sie alle Sicherheitshinweise in diesem Handbuch!

SCANLAB übernimmt keine Haftung für Schäden oder Folgeschäden aufgrund Nichtbeachtung dieses Handbuchs, insbesondere der hierin genannten Sicherheitshinweise.

1.1 Weiterführende Dokumente

- excelliSCAN-Handbuch
- RTC6-Handbuch (im syncAXIS control-Software-Paket enthalten, siehe **Seite 49**)
- Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"
- Handbuch "syncAXIS Viewer"
- Handbuch "syncAXIS Configurator"
- "AN ACS Components in XL SCAN System" (Publiziert durch ACS und ist dort anzufragen)
- "SLEC EtherCAT Node Installation Guide" (Publiziert durch ACS und ist dort anzufragen)

1.2 Hersteller

SCANLAB GmbH
Siemensstr. 2a
82178 Puchheim
Deutschland
Tel. +49 (89) 800 746-0
Fax +49 (89) 800 746-199
info@scanlab.de
www.scanlab.de

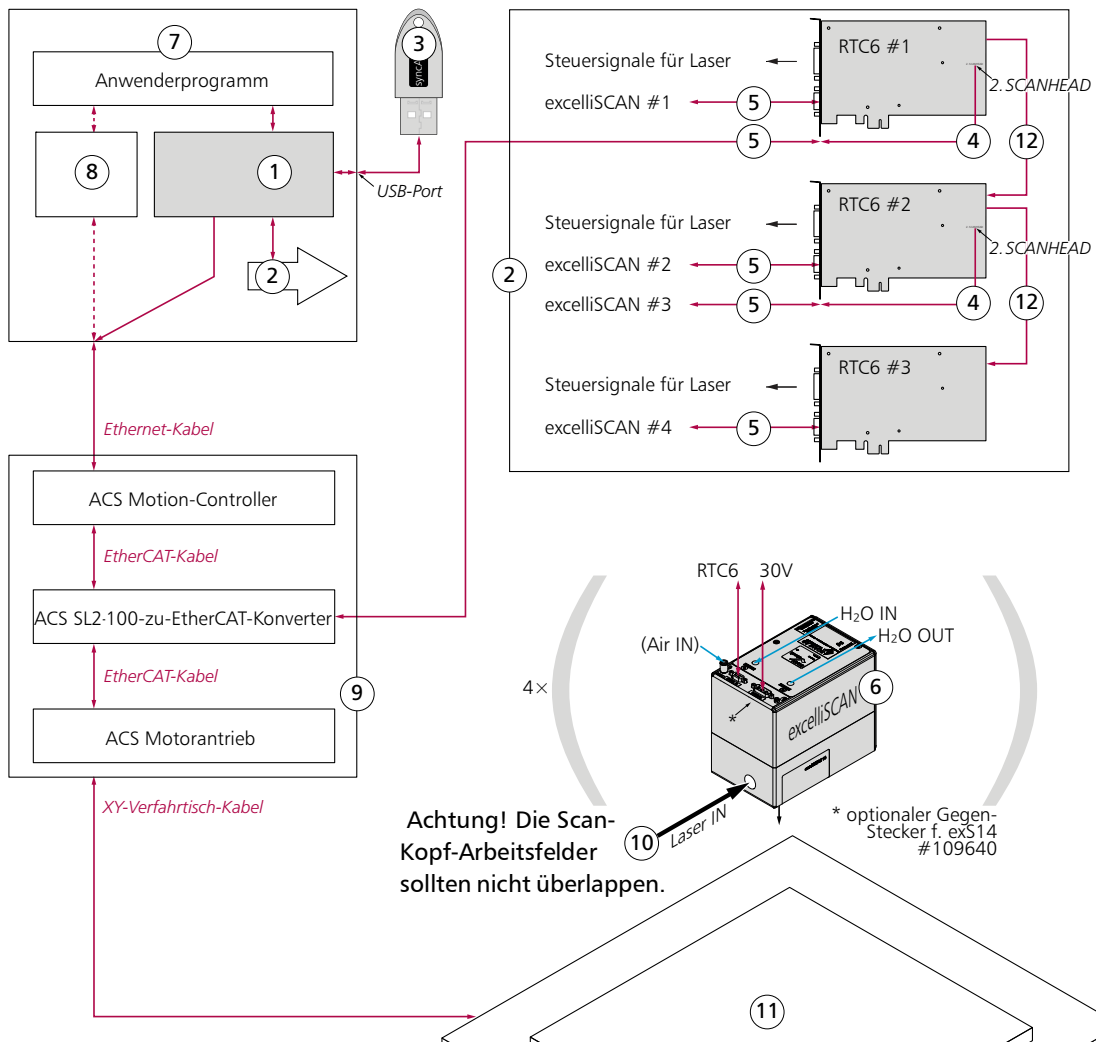
1.3 Glossar

ACS	Bezeichnet den Hersteller von Maschinensteuerungssystemen, dessen Komponenten aktuell für XL SCAN verwendet werden müssen.
API	Abkürzung für Application Programming Interface ("Schnittstelle zur Anwendungsprogrammierung"). Programmteil (z.B. der syncAXIS-DLL), der anderen Programmen zur Anbindung an das System zur Verfügung gestellt wird (z.B. Funktionen der syncAXIS-DLL). Siehe Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung" .
BIOS	Basic Input/Output System. Ist im Flash-Speicher der RTC6-Karte dauerhaft gespeichert.
Dongle	Siehe SCANLAB-USB-Dongle, Seite 11 .
Eingangspuffer	Der syncAXIS-DLL-interne Zwischenspeicher am Beginn der Verarbeitungskette. Siehe Puffer und Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung" .
Error-Mapping	"Stage error mapping". Die statische Kalibrierung des Verfahrtischs. (Hinweis: Die Scan-Kopf-Arbeitsfeld-Kalibrierung ist die statische Kalibrierung des Scan-Kopfs.)
Flash-Speicher	Nichtflüchtiger Speicher auf der RTC6-Karte, der das EEPROM der RTC5-Karte ersetzt.
Handle	Begriff aus der Programmierung: Abstrakte Referenz auf eine Ressource. Bezieht sich in diesem Handbuch auf eine bestimmte syncAXIS control-Instanz . Deren Handle-Wert wird durch <code>slsc_cfg_initialize_from_file</code> zugeteilt, siehe Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung" .
Job	Bezeichnet eine zwingend einzuhaltende Abfolge von Job-Funktionen (= <code>slsc_list_*</code>).
Multi-Head	Bezeichnet ein XL SCAN-System, bei dem 1 syncAXIS control-Instanz mehr als einen excelliSCAN-Scan-Kopf und 1 Verfahrtisch ansteuert.
Multi-Instanz	Bezeichnet die (optionale) Fähigkeit, mehr als eine syncAXIS control-Instanz auf einem PC gleichzeitig ausführen zu können. Erfordert einen Dongle , der diese Option explizit unterstützt ^(a) . Multi-Instanz und Multi-Head sind grundsätzlich ^(b) kompatibel.
Multi-Stage	Bezeichnet die (optionale) Funktionalität der syncAXIS control-Instanz , den Verfahrtisch wechseln zu können. Erfordert einen Dongle , der diese Option explizit unterstützt.
Puffer	Bezeichnet die syncAXIS-DLL-internen Zwischenspeicher der Verarbeitungskette. Siehe Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung" .
Pufferunterlauf	"Buffer underrun". Nach dem Ausführungs-Start hat die RTC6-Karte alle RTC6-Mikrovektorbefehle in ihrem Listenspeicher abgearbeitet, weil die syncAXIS-DLL in der Berechnung und Übertragung von weiteren RTC6-Mikrovektorbefehlen zu langsam war ("Berechnung langsamer als Ausführung"). Mögliche Maßnahmen siehe Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung", Kapitel 2.7.1 "Über die Puffer der syncAXIS control-Instanzen", Seite 42 .

syncAXIS control-Instanz	Softwareobjekt, das beim Aufruf einer gültigen <code>syncAXISConfig.xml</code> durch ein syncAXIS control-basiertes Anwenderprogramms im PC-RAM angelegt wird. Jede syncAXIS control-Instanz kann durch ein eindeutiges Handle angesprochen werden. Siehe Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung" .
<code>syncAXISConfig.xml</code>	"XML-Konfigurationsdatei". Obwohl der Dateiname frei gewählt werden kann, wird sie in diesem Dokument durchgehend als " <code>syncAXISConfig.xml</code> " bezeichnet. Die vollständige Beschreibung aller Tags finden Sie in Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung" .
Trajektorie	Kurve mit Zeitparameterisierung.

- (a) Die Anzahl der zulässigen **syncAXIS control-Instanzen** ist auf dem **Dongle** codiert.
- (b) Nicht kompatibel mit bestimmten Sonder-Systemen, bei denen 2 **syncAXIS control-Instanzen** auf einem PC 2 Master/Slave-verbundene RTC6-Karten ansteuern, die in das gleiche EtherCAT-Netzwerk einspeisen.

2.2 Mehr-Kopf-Aufbau (4 Scan-Köpfe)



Legende


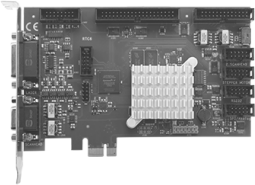
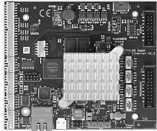

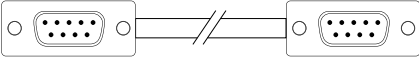
a. SCANLAB-XL SCAN-Komponenten
(typischer Lieferumfang von SCANLAB):

- 1 Software (syncAXIS control, RTC-Treiber, *.ct5-Datei)
- 2 RTC6 PCI-Express-Karten
- 3 SCANLAB-USB-Dongle
- 4 SSHC-Slotblech, siehe Seite 47
- 5 SL2-100-Datenkabel
(z.B. #115428 = 5 m, nicht schleppkettentauglich)
- 6 excelliSCAN-Scan-Kopf (4x), je mit Objektiv
- 12 Master/Slave-Kabel

b. Nicht-SCANLAB-Komponenten
(Voraussetzungen):

- 7 Windows-PC und Anwenderprogramm
- 8 ACS-Software
- 9 ACS-Hardware inkl. Kabel
- 10 Laser inkl. Kabel
- 11 Verfahrtsch





2.3 Typischer Lieferumfang (SCANLAB-XL SCAN-Komponenten)

SCANLAB-XL SCAN-Komponente	Für 1-Kopf- Aufbau	Für 2-Kopf- Aufbau	Für 3-Kopf- Aufbau	Für 4-Kopf- Aufbau
<p>excelliSCAN-Scan-Kopf</p>  <p>– mit Objektiv</p>	1	2	3	4
<p>RTC6 PCI-Express-Karte</p>   <p>RTC6 Ethernet-Karte</p> <p>– mit Option "syncA" und Option "SCANa"</p>	1	2	2	3
<p>SSHC-Slotblech</p>  <p>Siehe auch Abbildung 16, Seite 47.</p> <p>– #115132</p>	1	1	2	2
<p>SL2-100-Datenkabel^{(a)(b)}</p>  <ul style="list-style-type: none"> – Davon immer 1× zur Verbindung RTC6 / SL2-100-zu-Ether-CAT-Konverter^(c) – Restliche zur Verbindung RTC6 / Scan-Kopf – Jeweils mit zwei Steckverbindern 9-polig male Sub-D – z. B. #115428 = 5 m 	2	3	4	5

(a) Benutzer, die diese Kabel nicht bei SCANLAB erwerben, müssen **Seite 56** beachten.

(b) Aktuell sind SCANLAB-Kabel nicht schleppkettentauglich.

(c) Kurzform: "SLEC".

SCANLAB-XL SCAN-Komponente	Für 1-Kopf- Aufbau	Für 2-Kopf- Aufbau	Für 3-Kopf- Aufbau	Für 4-Kopf- Aufbau
Master/Slave-Kabel  – #117241 l = 50 mm oder  – #141553 l = 100 mm	0	1	1	2
Kabel – #116048 l = 200 mm – Wird nur mit RTC6 Ethernet-Karte(n) benötigt, siehe Abbildung 1, Seite 8	1	–	–	–
Y-Kabel – #116050 l = 200 mm – Wird nur mit RTC6 Ethernet-Karte(n) benötigt, siehe Abbildung 1, Seite 8	1	–	–	–
SCANLAB-USB-Dongle  – #127456 für 1-Kopf-Aufbau . – #139325 für 2-Kopf-Aufbau . – #139783 für 3-Kopf-Aufbau . – #138746 für 4-Kopf-Aufbau .	1	1	1	1
CD  – Mit syncAXIS control-Software-Paket syncAXIS-<Version>.zip. Inhalt siehe Kapitel 10 "Anhang C: syncAXIS control-Softwarepaket – Entpackt", Seite 48 – Beachten Sie den Sicherheitshinweis auf Seite 12 – Mit Handbüchern – Mit ct5-Korrekturdatei passend zum Objektiv	1	1	1	1

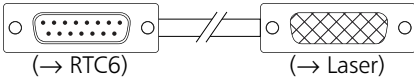
SCANLAB-XL SCAN-Komponente	Für 1-Kopf- Aufbau	Für 2-Kopf- Aufbau	Für 3-Kopf- Aufbau	Für 4-Kopf- Aufbau
<p><code>syncAXISConfig.xml</code></p> <ul style="list-style-type: none"> – Separat, i.d.R. per E-mail geliefert – Diese Datei ist spezifisch für jeden Kunden vorbereitet (d.h. entsprechend dessen, was der Kunde dem SCANLAB-Vertriebsmitarbeiter kommuniziert hat, sind darin Werte für <code>MotionDecompositionConfig</code>, Markiergeschwindigkeit und Sprunggeschwindigkeit eingetragen). Befolgen Sie die Sicherheitshinweise in Kapitel 3 "Die syncAXISConfig.xml prüfen und anpassen", Seite 15. 	1	1	1	1

Achtung!

Auch wenn die RTC6-Karte bereits mit bestimmten (bereits früher gelieferten) RTC6-Dateien unabhängig von syncAXIS control in Verwendung ist, dann dürfen RTC6-Karten und syncAXIS control zusammen nur mit den RTC6-Dateien aus dem jeweiligen syncAXIS control-Software-Paket `syncAXIS-<Version>.zip` verwendet werden.

2.4 Nicht-SCANLAB-Komponenten (Voraussetzungen)

Die folgenden Komponenten sind typischerweise *nicht* im SCANLAB-Lieferumfang für XL SCAN, siehe auch [Abbildung 1, Seite 8](#). Für ein vollständiges XL SCAN-System müssen sie verfügbar sein und mindestens die in folgender Tabelle genannten Voraussetzungen erfüllen.

Komponente	Voraussetzungen
Verfahrtisch und ACS-Komponenten (Hardware, Software, Verkabelung)	<ul style="list-style-type: none"> Sind installiert, konfiguriert, betriebsbereit. Das Stage-Error-Mapping (= "Feldkorrektur" für den Verfahrtisch) muss bereits durchgeführt worden sein (Wichtig: <i>vor</i> der Scan-Kopf-Feldkorrektur zum Ausgleich einer Rotation von Scan-Kopf- vs. Verfahrtisch-Arbeitsfeld^(a)). Es sind geeignete Maßnahmen zu treffen, damit die Hardware keine Ansteuerwerte ausführt, die zu Schäden führen können.
Lasers	<ul style="list-style-type: none"> Ist installiert, konfiguriert, betriebsbereit. Die Lasersicherheit ist gegeben.
RTC6-zu-Laser-Kabel	<ul style="list-style-type: none"> Ist vorhanden. Erfüllt die Vorgaben seitens des Laserherstellers. Steckverbinder 1: 15-pol male Sub-D. Zum Einstecken in die RTC6. Steckverbinder 2: gemäß Laserhersteller. Zum Einstecken in den Laser. Pin-out von Steckverbinder 1 passt zu LASER-Anschluss der RTC6, siehe Abbildung 15, Seite 46. Vorgesehene Verwendung: Siehe Kapitel 4.5 "RTC6-Karte(n) mit anderen Komponenten verbinden", Seite 19. Skizze:  Hinweis für Multi-Head-Setups Aktuell unterstützt XL SCAN (syncAXIS control V1.6.0) höchstens 3 RTC6-Karten. Soll ein 4-Kopf-Aufbau mit 4 Lasern betrieben werden, sind die Lasersteuersignale einer Karte Hardware-seitig zu duplizieren. Die Lasersteuersignale aller Karten sind identisch. Allgemein kann zur Lasersteuerung auch nur eine Karte verwendet werden, vorausgesetzt die Signale werden entsprechend aufgeteilt.

(a) Beschrieben in [Kapitel 7.2 "Optimierung – Das Scan-Kopf-Arbeitsfeld kalibrieren", Seite 36](#).

Sub-System (Forts.)	Voraussetzungen (Forts.)
Spannungsversorgung für den excelliSCAN-Scan-Kopf	<ul style="list-style-type: none"> Ist installiert und betriebsbereit (Die Installation darf nur durch geschulte Personen erfolgen, die über ausreichende Kenntnisse im sicheren Umgang mit elektrischen Geräten verfügen). Hinweis: optional lieferbar ist der Gegenstecker für den excelliSCAN 14-Power-Anschluss (#109640^(a)).
Windows-PC	<ul style="list-style-type: none"> Ist mit einem Intel i5-Prozessor oder besser ausgestattet. Ist auch mit einer SSD ausgestattet. Ist installiert, konfiguriert, betriebsbereit. Ein USB-Port für den SCANLAB-USB-Dongle ist frei. Entsprechend der Anzahl (siehe Seite 10) der RTC6 PCI-Express-Karten freie PCIe-Slots sowie 1 oder 2 (siehe Seite 10) freie Slots, um das SSHC-Slotblech montieren zu können. Softwareentwicklungsumgebung mit C++-Compiler (z.B. MS Visual Studio 2015 Professional oder später). Voraussetzung, wenn das "Installation_Project" (siehe Seite 48) selber kompiliert werden soll: CMake^(b).

(a) Nicht für excelliSCAN 20.

(b) CMake-Dateien (cross-platform make) dienen zur einfachen Erstellung von MS Visual Studio Solution-Dateien. CMake (<https://cmake.org>) ist ein freies und open-source plattformunabhängiges Programmierwerkzeug für die Entwicklung und Erstellung von Software. Mit CMake werden aus Skriptdateien (CMakeLists.txt) Makefiles und Projekte für viele integrierte Entwicklungsumgebungen und Compiler erzeugt.

3 Die syncAXISConfig.xml prüfen und anpassen

SCANLAB erstellt zwar – basierend auf Angaben des Systembetreibers bzgl. der geplanten Systemkomponenten – eine passende `syncAXISConfig.xml` und stellt diese (i.d.R. per E-Mail, zusätzlich und separat zum syncAXIS control-Software-Paket) bereit.

Allerdings darf die Datei niemals ohne vorherige Prüfung und Anpassung durch den Systembetreiber zum Betrieb des XL SCAN-System verwendet werden.

Das gilt insbesondere dann, wenn der Systembetreiber nicht sicher ist, dass er SCANLAB alle aktuellen Systemkomponenten (bzw. deren technischen Spezifikationen) korrekt und vollständig kommuniziert hat und/oder ob nachträglich Systemkomponenten verändert wurden.



Vorsicht!

Trotz der in der `syncAXISConfig.xml` eingetragenen Grenzwerten berechnet syncAXIS control u.U. Ansteuerwerte und Prozessgeschwindigkeiten *außerhalb* der Systemgrenzen. Es können Ansteuerwerte generiert werden, die zu Sach- und/oder Personenschäden führen könnten. Der Systembetreiber ist dafür verantwortlich, dass Mechanismen wirksam sind^(a), die die tatsächliche Ausführung von schädlichen Ansteuerwerten verhindern. Ein syncAXIS control-basiertes Anwenderprogramm muss außerdem immer erst im Simulationsmodus ausgeführt werden (um eine Prüfung auf Überschreitung der Systemgrenzen durchzuführen), bevor mit ihm erstmalig eine reale Markierung mit Laser, Scan-Kopf und Verfahrtschicht ausgeführt wird.

SCANLAB übernimmt keine Haftung für Schäden oder Folgeschäden aufgrund Nichtbeachtung dieses Handbuchs.

(a) Siehe entsprechende Sicherheitskapitel im **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**.



Vorsicht!

Der Systembetreiber ist dafür verantwortlich, dass alle in der syncAXIS control `syncAXISConfig.xml` eingetragenen Werte sinnvoll und zum verwendeten XL SCAN-System passend sind.

Konsultieren Sie dazu die Teil-Systemhersteller und die zugehörigen Dokumentationen.

Hinweise

- Ab syncAXIS control \geq V1.2 entfällt die XML-System-Konfigurationsdatei `syncAXISSysConfig.xml`!
- Im Installationspaket befinden sich auch Vorlagen (`syncAXISConfig.Template.xml`, `syncAXISConfig_MultiHead.Template.xml`) für `syncAXISConfig.xml`-Dateien. Diese werden als Referenz bereitgestellt und dürfen unverändert keinesfalls zum Betrieb von XL SCAN-Systemen verwendet werden.
- Zusätzliche Unterstützung (z. B. zulässige Wertebereiche und Argumente, Inline-Dokumentation etc.) finden Benutzer im **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"** (in **Anhang F: Referenz der syncAXISConfig.xml-Tags**) sowie im mitgelieferten XML-Schema `syncAXIS_<major vers'n>_<minor vers'n>.xsd` (z. B. `syncAXIS_1_8.xsd`).

Ziele

- Die `syncAXISConfig.xml` ist für den nächsten Schritt so vorbereitet, dass eine sichere Ausführung von syncAXIS control-Anwenderprogrammen gewährleistet ist.

Materialien

- `syncAXISConfig.xml`⁽¹⁾ (und zur Referenz das XML-Schema, siehe **Seite 15** sowie das **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**)
- xml/Text-Editor oder syncAXIS Configurator.

(1) Siehe **Seite 15**.

Vorgehen

- (1) Öffnen Sie die `syncAXISConfig.xml` (schreibend) mit Ihrem bevorzugten xml/Text-Editor.
- (2) Gehen Sie den Inhalt durch, prüfen Sie die eingetragenen Werte und nehmen die erforderlichen Änderungen oder Angaben vor, u.a.:
 - syncAXIS control Sicherheitsfeatures⁽¹⁾
(`DynamicViolationReaction` sowie `MonitoringLevel`)
 - RTC6-Kartenkonfiguration (Seriennummer, für Scan-Kopf und Verfahrtsch benutzte Stecker)
 - nutzbarer Verfahrtsch-Arbeitsbereich
 - nutzbares Scan-Kopf-Arbeitsfeld
 - max. Verfahrtschgeschwindigkeit-, beschleunigung, -ruck
 - ct5-Datei(en)Befolgen Sie auch die zugehörigen Hinweise im Handbuch "`syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung`". Dort finden Sie auch die Dokumentation der xml-Tags.
- (3) Speichern Sie die Datei.

(1) Siehe Kapitel 2.2 "Über die SICHERE Verwendung von `syncAXIS control` – Allgemeines Vorgehen".

4 Installieren der SCANLAB-XL SCAN-Komponenten

In diesem Kapitel:

- Kapitel 4.1 "Nicht-SCANLAB-Komponenten vorbereiten", Seite 17
- Kapitel 4.2 "excelliSCAN(s) installieren", Seite 17
- Kapitel 4.3 "Gilt nur mit RTC6 PCI-Express-Karten – Windows-PC mit RTC6-Komponenten erweitern", Seite 18
- Kapitel 4.5 "RTC6-Karte(n) mit anderen Komponenten verbinden", Seite 19
- Kapitel 4.6 "syncAXIS-Software auf dem Windows-PC entpacken und SCANLAB-USB-Dongle einstecken", Seite 19

4.1 Nicht-SCANLAB-Komponenten vorbereiten

Bereiten Sie die Nicht-SCANLAB-Komponenten des Scan-Systems (**Verfahrtisch und ACS-Komponenten, Laser, RTC6-zu-Laser-Kabel und Spannungsversorgung**) so vor, wie in Kapitel 2.4 "Nicht-SCANLAB-Komponenten (Voraussetzungen)", Seite 13 dargestellt.

Beachten Sie auch "AN ACS Components in XL SCAN System".

4.2 excelliSCAN(s) installieren

Die Installation und Herstellung der Betriebsbereitschaft ist vollumfänglich im excelliSCAN-Handbuch beschrieben, siehe dort. Achten Sie in der Einbausituation insbesondere darauf, dass:

- der excelliSCAN den spezifizierten Arbeitsabstand hat
- der Laserfokus auf der Oberfläche des Markiersubstrats geführt wird⁽¹⁾

Hinweise

- Für Aufbauten, bei denen die Scan-Köpfe bewegt/verfahren werden, ist zusätzlich sicherzustellen, dass auch deren Wasser-, Luft- und elektrische Leitungen auch dieser Anforderung gerecht werden (z.B. Schleppkettentauglichkeit, entsprechende Materialflexibilitäten und Biege- radien, geeignete Zugentlastungen und Fittings).

(1) Siehe excelliSCAN-Handbuch, Kapitel 5.3 Justage.

4.3 Gilt nur mit **RTC6 PCI-Express-Karten** – Windows-PC mit RTC6-Komponenten erweitern

Ziele

- Der **Windows-PC** ist mit der entsprechenden Anzahl von **RTC6 PCI-Express-Karte(n)** und **SSHC-Slotblech(en)** ausgestattet. Bei Verwendung mehrerer Karten sind diese Master/Slave verbunden. Der RTC6-Treiber ist in Windows installiert.
 - Für **1-Kopf-Aufbau** siehe **Abbildung 1, Seite 8**
 - Für **4-Kopf-Aufbau** siehe **Abbildung 2, Seite 9**

Materialien

- **Windows-PC**
- $n \times$ **RTC6 PCI-Express-Karte**, n siehe **Seite 10**
- $n \times$ **SSHC-Slotblech**, n siehe **Seite 10**
- $n \times$ **Master/Slave-Kabel**, n siehe **Seite 11**
- RTC6-Treiber für Windows (in
 \---RTC6
 | \---Driver)
- **RTC6-Handbuch** (im syncAXIS control-Software-Paket enthalten, siehe **Seite 49**)

Vorgehen

- (1) Installieren Sie die **RTC6 PCI-Express-Karte(n)** und das (die) **SSHC-Slotblech(e)** in den **Windows-PC**. Bei mehreren Karten verbinden Sie diese als Master/Slave (**Master/Slave-Kabel**). Siehe auch **RTC6-Handbuch** (im syncAXIS control-Software-Paket enthalten, siehe **Seite 49**).
- (2) Stecken Sie den **SSHC-Slotblech-Stecker**, siehe **Abbildung 16, Seite 47**, in den 2. SCANHEAD-Stecker der betreffenden Karte:
 - Für **1-Kopf-Aufbau** siehe **Abbildung 1, Seite 8**
 - Für **4-Kopf-Aufbau** siehe **Abbildung 2, Seite 9**
- (3) Schließen Sie das PC-Gehäuse und starten Sie Windows.
- (4) Installieren Sie den RTC6-Treiber für Windows.
- (5) Installieren Sie an dieser Stelle *keine* weitere Software aus dem RTC6-Software-Paket. Die RTC6-Dateien *.dll, *.rbf, *.out, *.dat werden später installiert und sind immer aus dem syncAXIS control-Software-Paket (nie aus dem RTC6-Software-Paket⁽¹⁾) zu entnehmen, siehe **Kapitel 4.6, Seite 19**.

4.4 Gilt nur mit **RTC6 Ethernet-Karten** – **RTC6 Ethernet-Karte** installieren

Hinweise

- **RTC6 Ethernet-Karten** sind aktuell nur für einen **1-Kopf-Aufbau** möglich, siehe **Abbildung 1, Seite 8**
- **RTC6 Ethernet-Karten** benötigen den RTC6-Treiber für Windows *nicht*.

Ziele

- Die **RTC6 Ethernet-Karte** ist installiert (z.B. Schaltschrank).

Materialien

- $1 \times$ **RTC6 Ethernet-Karte**, siehe **Seite 10**
- $1 \times$ **SSHC-Slotblech**, siehe **Seite 10**
- $1 \times$ **Kabel**, siehe **Seite 11**
- $1 \times$ **Y-Kabel**, siehe **Seite 11**
- **RTC6-Handbuch** (im syncAXIS control-Software-Paket enthalten, siehe **Seite 49**)

Vorgehen

- (1) Installieren Sie die **RTC6 Ethernet-Karte** wie im **RTC6-Handbuch** (im syncAXIS control-Software-Paket enthalten, siehe **Seite 49**) beschrieben gemäß **Abbildung 1, Seite 8**.
- (2) Installieren Sie an dieser Stelle *keine* weitere Software aus dem RTC6-Software-Paket. Die RTC6-Dateien *.dll, *.rbf, *.out, *.dat werden später installiert und sind immer aus dem syncAXIS control-Software-Paket (nie aus dem RTC6-Software-Paket⁽¹⁾) zu entnehmen, siehe **Kapitel 4.6 "syncAXIS-Software auf dem Windows-PC unpacken und SCANLAB-USB-Dongle einstecken", Seite 19**.

- (1) Für syncAXIS control-Anwenderprogramme müssen Sie immer die RTC6-Dateien aus dem syncAXIS control-Software-Paket verwenden.

4.5 RTC6-Karte(n) mit anderen Komponenten verbinden

Ziele

- Die **RTC6 PCI-Express-Karte** ist mit SL2-100-zu-EtherCAT-Konverter, Laser und excelliSCAN elekt-risch verbunden.
 - Für 1-Kopf-Aufbau siehe **Abbildung 1, Seite 8**
 - Für 4-Kopf-Aufbau siehe **Abbildung 2, Seite 9**

Materialien

- Windows-PC** aus "Gilt nur mit RTC6 PCI-Express-Karten – Windows-PC mit RTC6-Komponenten erweitern", Seite 18
- 1 × **SL2-100-Datenkabel**
 - Zur Verbindung RTC6 / SL2-100-zu-EtherCAT-Konverter
- n × **SL2-100-Datenkabel**
 - Zur Verbindung RTC6 / Scan-Kopf, n siehe **Seite 10**
- n × **RTC6-zu-Laser-Kabel**
 - Zur Verbindung RTC6 / Laser
 - n hängt von Ihren Anforderungen ab

Vorgehen

- RTC6 und SL2-100-zu-EtherCAT-Konverter verbinden.
Verwenden Sie das **SL2-100-Datenkabel**.
Stecken Sie den Steckverbinder 1 in den **SSHC-Slotblech**-Steckverbinder der Master-RTC6 (siehe **Abbildung 17, Seite 47**) und den Steckverbinder 2 in den entsprechenden Steckverbinder des SL2-100-zu-EtherCAT-Konverter.
- RTC6-Karte(n) und excelliSCAN(s) verbinden.
Verwenden Sie das (die) **SL2-100-Datenkabel**.
 - Für 1-Kopf-Aufbau siehe **Abbildung 1, Seite 8**
 - Für 4-Kopf-Aufbau siehe **Abbildung 2, Seite 9**
(Anders als in dieser Zeichnung könnte excelliSCAN#4 alternativ mit dem 2. SCANHEAD-Steckverbinder der RTC6-Karte #3 verbunden werden)
 Hinweis: Den SCANHEAD-Steckverbinder der **RTC6 PCI-Express-Karte** zeigt **Abbildung 14, Seite 46**.
- RTC6-Karte(n) und Laser verbinden.
Verwenden Sie das (die) **RTC6-zu-Laser-Kabel**.
Hinweis: Den LASER-Steckverbinder der **RTC6 PCI-Express-Karte** zeigt **Abbildung 15, Seite 46**.

4.6 syncAXIS-Software auf dem Windows-PC entpacken und SCANLAB-USB-Dongle einstecken

Ziele

- Der **Windows-PC** ist mit dem syncAXIS control-Software-Paket und dem **SCANLAB-USB-Dongle** ausgestattet (d.h. Benutzer können in den Folgeschritten auf die relevanten Dateien, wie RTC6-Dateien, "Installation_Project") etc., zugreifen).

Materialien

- Windows-PC** aus "Gilt nur mit RTC6 PCI-Express-Karten – Windows-PC mit RTC6-Komponenten erweitern", Seite 18
- SCANLAB-USB-Dongle**
- syncAXIS control-Software-Paket

Vorgehen

- Stecken Sie den **SCANLAB-USB-Dongle** in einen freien USB-Port des PCs.
- Entpacken Sie das syncAXIS control-Software-Paket (*.zip-Datei) auf dem PC (z.B. Windows Desktop) in einen Zielordner Ihrer Wahl.
- Verschaffen Sie sich einen Überblick über die enthaltenen Dateien. Siehe auch "**Anhang C: syncAXIS control-Softwarepaket – Entpackt**", Seite 48.

5 Inbetriebnehmen des XL SCAN-Systems – Software

In diesem Kapitel:

- Kapitel 5.1 "Installation_Project.exe vorbereiten", Seite 20
 - Kapitel 5.1.1 "Alternative 1 – "Installation_Project" für MS Visual Studio erstellen und Installation_Project.exe kompilieren", Seite 20
 - Kapitel 5.1.2 "Alternative 2 – Mitgelieferte Installation_Project.exe verwenden", Seite 21
- Kapitel 5.2 "Installation_Project.exe starten und syncAXIS control-Instanz im Simulationsmodus initialisieren", Seite 22
- Kapitel 5.3 "TEST_MARKING" im Betriebsmodus "ScannerAndStage" simulieren", Seite 24

Hinweise

- Die diesem Kapitel sind relevant für alle Anwender, die den XL SCAN in Betrieb nehmen möchten.
- Für Anwender die einen "Office-Dongle" besitzen und durch Simulationen mit dem System vertraut werden möchten, sind dies die einzigen auszuführenden Schritte.

5.1 Installation_Project.exe vorbereiten

Führen Sie wahlweise aus:

- Kapitel 5.1.1 "Alternative 1 – "Installation_Project" für MS Visual Studio erstellen und Installation_Project.exe kompilieren", Seite 20
- Kapitel 5.1.2 "Alternative 2 – Mitgelieferte Installation_Project.exe verwenden", Seite 21

5.1.1 Alternative 1 – "Installation_Project" für MS Visual Studio erstellen und Installation_Project.exe kompilieren

In diesem Schritt wird das "Installation_Project"⁽¹⁾ (mittels CMake⁽²⁾ und den mitgelieferten cmake-Dateien) in der Softwareentwicklungsumgebung eingerichtet und kompiliert. Außerdem wird die (muss vorher vom Systembetreiber angepasst worden sein, siehe Kapitel 3 "Die syncAXISConfig.xml prüfen und anpassen", Seite 15) syncAXISConfig.xml im erwarteten Verzeichnis verfügbar gemacht. Das "Installation_Project" wird für alle Folgeschritte in diesem Handbuch benutzt.

Ziele

- Das "Installation_Project" ist in der Softwareentwicklungsumgebung eingerichtet (hier: als MS Visual Studio Solution). Die Installation_Project.exe wird fehlerlos kompiliert.

Materialien

- Softwareentwicklungsumgebung (MS Visual Studio 2015 Professional oder später)
- CMake (angenommener Installationsort: C:\Program Files\CMake\bin\cmake.exe)
- Mitgelieferte Dateien für CMake im Verzeichnis \---Demo.
- syncAXISConfig.xml aus Kapitel 3 "Die syncAXISConfig.xml prüfen und anpassen", Seite 15

(1) Siehe auch Seite 5.

(2) Siehe Fußnote auf Seite 14.

Vorgehen

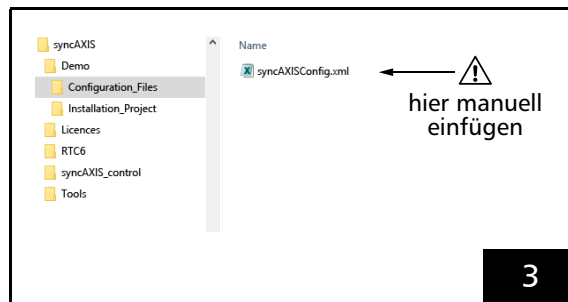
- (1) Kopieren Sie die `syncAXISConfig.xml` aus **Kapitel 3** "Die `syncAXISConfig.xml` prüfen und anpassen", **Seite 15** in das Verzeichnis

```

\---Demo
|   \---Configuration_Files

```

 (des entpackten Pakets, siehe Schritt 2, **Seite 19**), siehe **Abbildung 3, Seite 21**.



Zu Schritt 1, **Seite 21**: vor dem Ausführen der Batch-Datei muss sich die `syncAXISConfig.xml` am bezeichneten Ort befinden (Dieser Pfad wird automatisch in die `IncludePath.h` eingetragen, siehe auch **INIT**).

- (2) Öffnen Sie mit einem Editor die Datei

```

\---Demo
|   |   GenerateCMakeTestProject.bat.

```

Wenn nötig, passen Sie darin die MS Visual Studio-Version mit der von ihnen benutzten an, z. B. "14 2015" zu "15 2017".

- (3) Führen Sie die `GenerateCMakeTestProject.bat` aus.
Im Unterverzeichnis `Build_syncAXIS` wird eine "Solution" für MS Visual Studio erstellt.
- (4) Starten Sie MS Visual Studio.
- (5) Öffnen Sie im Unterverzeichnis `Build_syncAXIS` die Solution-Datei `*.slu`.
- (6) Erstellen ("Build") Sie die ausführbare Datei `Installation_Project.exe`.

5.1.2 Alternative 2 – Mitgelieferte

`Installation_Project.exe`
verwenden

Dieser Schritt betrifft Anwender, die keine C++ Entwicklungsumgebung auf ihrem **Windows-PC** installiert haben und z. B. eine Drittanbieter-GUI-Lösung benutzen möchten.

Die Konsolenanwendung

"`Installation_Project.exe`" (siehe **Abbildung 4, Seite 22**) aus dem `syncAXIS` control-Software-Paket wird vorbereitet, um damit die weiteren Schritte aus diesem Handbuch vornehmen zu können.

Ziele

- `Installation_Project.exe` ist bereit zum Ausführen

Materialien

- `syncAXISConfig.xml` aus den vorigen Schritten
- `Installation_Project.exe` aus

```

\---Tools
|   \---syncAXIS_Installation

```

Vorgehen

- (1) Kopieren Sie die `syncAXISConfig.xml` in das Verzeichnis, in dem auch die `Installation_Project.exe` liegt.
- (2) Tragen Sie in der `syncAXISConfig.xml` unter `BaseDirectoryPath` den Pfad aus Schritt 1 ein.
- (3) Tragen Sie in der `syncAXISConfig.xml` unter `ProgramFileDirectory` den absoluten Pfad von

```

\---RTC6
|   \---ProgramFiles

```

 ein.

5.2 Installation_Project.exe starten und syncAXIS control-Instanz im Simulationsmodus initialisieren

Aus dem vorausgehenden Schritt **Kapitel 5.1** "Installation_Project.exe vorbereiten", Seite 20 ist Installation_Project.exe vorhanden.

In diesem Schritt soll mit dieser eine **syncAXIS control-Instanz** im Simulationsmodus⁽¹⁾⁽²⁾ und Betriebsmodus "ScannerAndStage" initialisiert werden.

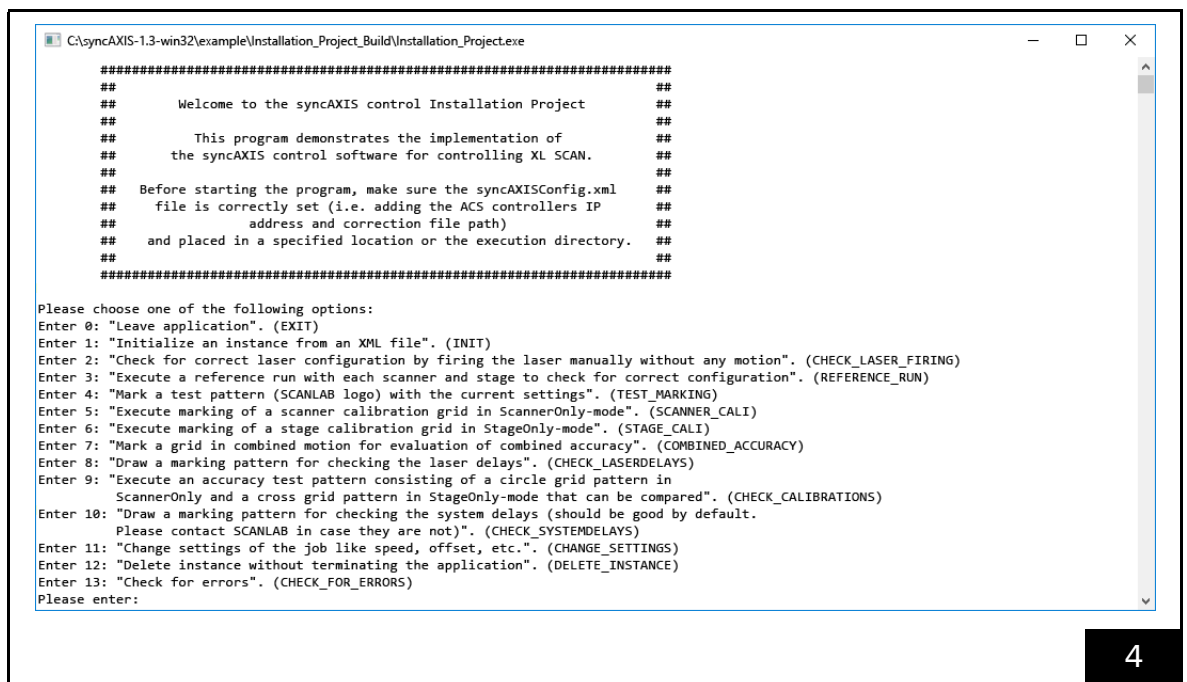
(1) In **syncAXISConfig.xml**:

```
<cfg:SimulationMode>true
</cfg:SimulationMode>
```

Ist die Initialisierung erfolgreich, wird in Installation_Project.exe "finished with return value 0" angezeigt. Das bedeutet u.a.:

- Der **SCANLAB-USB-Dongle** ist eingesteckt und wird erkannt
- Die **syncAXISConfig.xml** ist rein Software-technisch korrekt
 - Sie ist valide gegen das entsprechende xml-Schema (z.B. es sind keine negativen Geschwindigkeitswerte eingetragen)

(2) Im Simulationsmodus steuert die ausführbare Datei den Laser/Verfahrtisch/**RTC6 PCI-Express-Karte** nicht tatsächlich an. Es wird auch noch nicht die Verfahrtisch-Kommunikation überwacht.



Installation_Project.exe öffnet sich als Konsolenanwendung.

Zum Aufbauen der **syncAXIS control-Instanz**: **1** eingeben

Zur Fehlerabfrage: **13** eingeben.

Zum Beenden der **syncAXIS control-Instanz** (ohne diese Konsolenanwendung zu beenden): **12** eingeben.

Zum Beenden dieser Konsolenanwendung: **0** eingeben.

Ziele

- Eine **syncAXIS control-Instanz** wird Simulationsmodus aufgebaut

Materialien

- **syncAXISConfig.xml** aus dem vorausgehenden Schritt
- **Installation_Project.exe** aus **Kapitel 5.1** "Installation_Project.exe vorbereiten", Seite 20

Vorgehen

- (1) Stellen Sie sicher, dass in der **syncAXISConfig.xml** eingetragen ist:
 - Simulationsmodus⁽¹⁾
 - Eine IP-Adresse für den **ACS** Motion Controller (allerdings wird nur das Format geprüft)
 - Pfad zur **ct5**-Datei
- (2) **Installation_Project.exe** starten.
Ein neues Fenster wird geöffnet, siehe **Abbildung 4, Seite 22**.
- (3) Zum Initialisieren der **syncAXIS control-Instanz** (**INIT**) geben Sie ein:
 - **'1'** – Eingabetaste – [**'1'** oder **'2'**⁽²⁾] – Eingabetaste.

Wenn erfolgreich, wird im Fenster angezeigt:

"Finished with return value 0".

Wenn nicht erfolgreich, wird im Fenster angezeigt: "Finished with return value <nn>".

- Zur Fehlerabfrage (**CHECK_FOR_ERRORS**) geben Sie ein:
 - **'13'** – Eingabetaste
- Zum Löschen der **syncAXIS control-Instanz** (**DELETE_INSTANCE**) geben Sie ein:
 - **'12'** – Eingabetaste
- Beheben Sie den Fehler und wiederholen Sie diesen Schritt **3**.

- (4) Zum Beenden (**EXIT**) von **Installation_Project.exe** geben Sie ein:
 - **'0'** – Eingabetaste

(1) Siehe Fußnote (1), Seite 22.

(2) **'1'** wenn Sie die **Installation_Project.exe** selber kompiliert haben (Kap. 5.1.1) und **'2'**, wenn nicht (Kap. 5.1.2).

5.3 “TEST_MARKING” im Betriebsmodus “ScannerAndStage” simulieren

Im vorausgehenden Schritt

“Installation_Project.exe starten und syncAXIS control-Instanz im Simulationsmodus initialisieren”, Seite 22 wurde sichergestellt, dass mit der Installation_Project.exe und der angegebenen syncAXISConfig.xml erfolgreich eine syncAXIS control-Instanz aufgebaut werden kann.

Vor einer allerersten Ausführung eines jeden Markiermusters mit syncAXIS control muss dieses zuerst (= in diesem Schritt) simuliert werden.



Vorsicht!

Trotz der in den syncAXISConfig.xml eingetragenen Grenzwerten berechnet syncAXIS control u.U. Ansteuerwerte und Prozessgeschwindigkeiten *außerhalb* der Systemgrenzen.

Führen Sie deshalb syncAXIS control-basierte Anwenderprogramme immer erst im Simulationsmodus aus (um eine Prüfung auf Überschreitung der Systemgrenzen durchzuführen), bevor Sie mit ihm erstmalig ein reale Markierung mit Laser, Scan-Kopf und Verfahrtsch ausführen.

So können Anwender sicherstellen, dass keine Grenzwertüberschreitungen (= Positions- und Dynamikverletzungen) in den Ansteuerwerten enthalten sind (dazu ist die generierte Simulationsdatei z. B. mit dem SCANLAB-Tool “syncAXIS Viewer”, siehe **Abbildung 5, Seite 25**, sichtbar zu prüfen und auszuwerten). Betroffene Stellen können im syncAXIS Viewer hervorgehoben werden (siehe syncAXIS Viewer-Handbuch).⁽¹⁾

(1) Die von SCANLAB gelieferte syncAXISConfig.xml basiert auf den Kunden-Angaben für Grenzwerte von Scan-Device und Verfahrtsch. Diese “initiale” Datei wird so erstellt, dass es bei keinem der Installation_Project.exe-Markiermuster zu Grenzwertüberschreitungen kommen sollte (lassen Sie die Parameterwerte unverändert!). Sollten Sie trotzdem eine Grenzwertüberschreitung beobachten, wenden Sie sich unverzüglich an den SCANLAB-Support und führen Sie keine Markierung mit dem XL SCAN-Hardwareaufbau durch.

Ziele

- Eine Simulationsdatei von “TEST_MARKING” kann erstellt und anschließend mit dem syncAXIS Viewer geöffnet werden
- Es ist sichergestellt, dass in den Ansteuerwerten weder für Scan-Device noch Verfahrtsch Grenzwertüberschreitungen enthalten sind

Materialien

- Installation_Project.exe aus dem vorausgehenden Schritt
- syncAXISConfig.xml (Simulationsmodus ist eingetragen⁽²⁾, Betriebsmodus “ScannerAndStage” ist eingetragen⁽³⁾) aus dem vorausgehenden Schritt
- SCANLAB-Tool “syncAXIS Viewer” zum Darstellen und Auswerten der Simulationsdatei

Vorgehen

- (1) Installation_Project.exe starten.
- (2) Zum Initialisieren der syncAXIS control-Instanz (INIT) geben Sie ein:
 - ‘1’ – Eingabetaste – [‘1’ oder ‘2’⁽⁴⁾] – Eingabetaste
- (3) Zum Generieren der TEST_MARKING-Simulationsdatei geben Sie ein:
 - ‘4’ – Eingabetaste – dann die tatsächliche Größe des Objekts (in mm, sollte etwas kleiner als das Scan-Kopf-Arbeitsfeld sein) – y
- (4) Zum Löschen der syncAXIS control-Instanz (DELETE_INSTANCE) geben Sie ein:
 - ‘12’ – Eingabetaste
- (5) Zum Beenden (EXIT) von Installation_Project.exe geben Sie ein:
 - ‘0’ – Eingabetaste

- (2) In syncAXISConfig.xml:

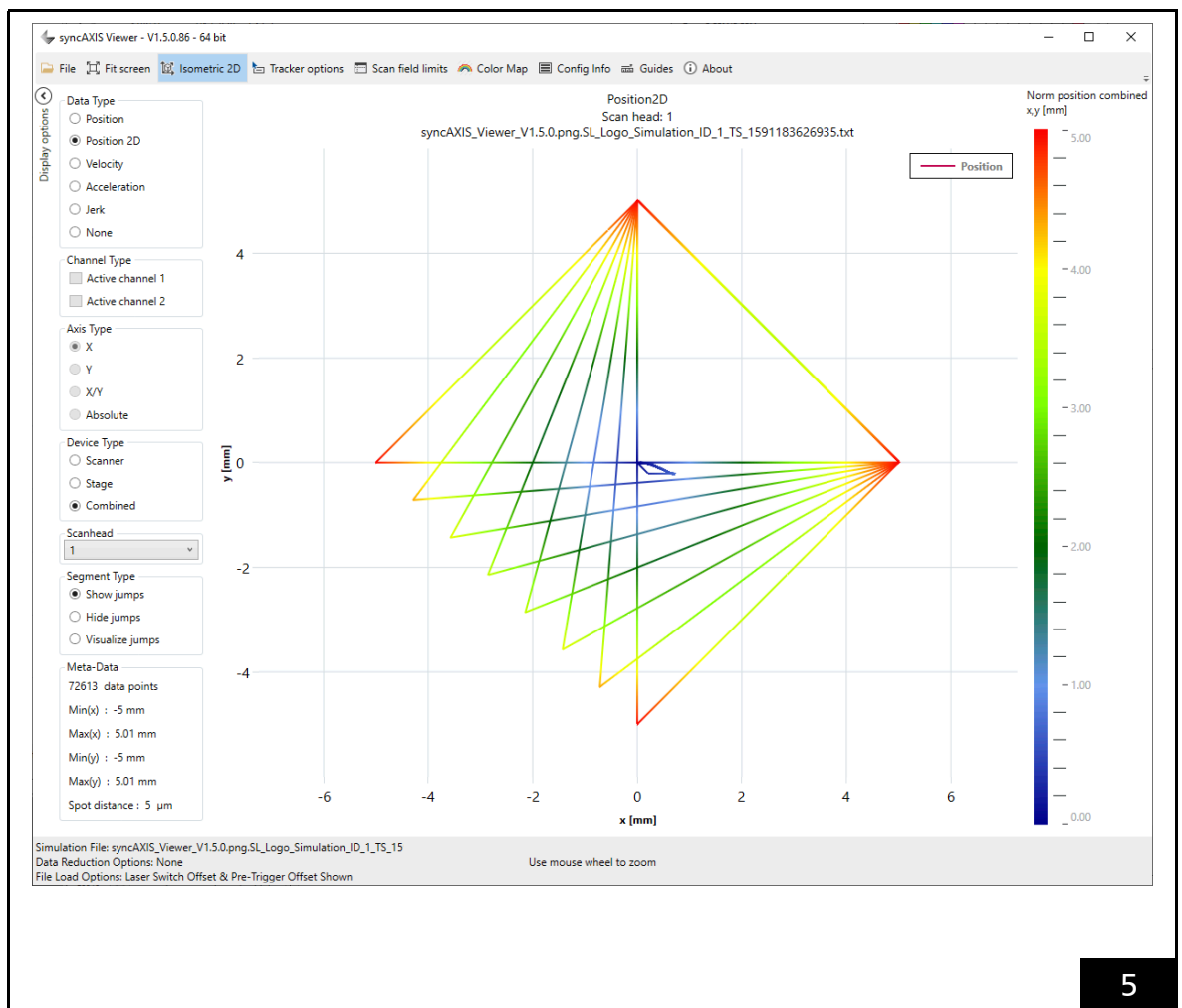

```
<cfg:SimulationMode>true
</cfg:SimulationMode>
```
- (3) In syncAXISConfig.xml:


```
<cfg:InitialOperationMode>ScannerOnly
</cfg:InitialOperationMode>
```
- (4) ‘1’ wenn Sie die Installation_Project.exe selber kompiliert haben (Kap. 5.1.1) und ‘2’, wenn nicht (Kap. 5.1.2).

(6) Prüfen Sie die Simulationsdatei (z. B. mit "syncAXIS Viewer") daraufhin, ob Grenzwertüberschreitungen (= Positions- und Dynamikverletzungen) in den Ansteuerwerten enthalten sind. Passen Sie, wenn notwendig, die `syncAXISConfig.xml` an und wiederholen diese Prozedur ab Schritt 1.

Siehe auch **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**, Kapitel 2.6 "Über das Optimieren von syncAXIS control-basierten Anwenderprogrammen", Seite 36.

Fahren Sie erst dann mit dem nächsten Schritt "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren", Seite 26, fort, wenn Sie keine Grenzwertüberschreitungen mehr feststellen.



SCANLAB-Tool "syncAXIS Viewer" V1.5 mit geöffneter Simulationsdatei. Als Plot ist 'Position 2D' eingestellt.

6 Inbetriebnehmen des XL SCAN-Systems – Hardware



Vorsicht!

Es besteht Verletzungsgefahr durch Hardwarekomponenten zur Laserbearbeitung, die in den nachfolgenden Schritten involviert sind.

Stellen Sie sicher dass alle diese Komponenten betriebsbereit sind und alle Sicherheitsvorkehrungen getroffen sind.

6.1 Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren

Im vorausgehenden Schritt **“TEST_MARKING” im Betriebsmodus “ScannerAndStage” simulieren**, Seite 24 wurde das Markiermuster simuliert und etwaige Grenzwertüberschreitungen entfernt.

Deswegen darf im vorliegenden Schritt (mit Installation_Project.exe und syncAXISConfig.xml aus dem vorausgehenden Schritt) versucht werden, zu initialisieren:

- eine syncAXIS control-Instanz
 - im Hardwaremodus⁽¹⁾
 - im Betriebsmodus “ScannerAndStage”⁽²⁾

Ist die Initialisierung erfolgreich, wird im Installation_Project.exe-Fenster angezeigt: “Finished with return value 0”.

Dies bedeutet u.a.:

- Der **SCANLAB-USB-Dongle** ist eingesteckt und wird erkannt
- Die Hardware ist korrekt in syncAXISConfig.xml konfiguriert
- Die Hardware ist korrekt auf dem **ACS** Motion Controller konfiguriert
- Die Hardware ist korrekt verkabelt
- Die Hardware wird korrekt initialisiert und alle Komponenten sind betriebsbereit
- Die Einträge in der syncAXISConfig.xml sind zwar korrekt, jedoch nicht zwangsläufig schon “optimal”

Ziele

- Eine syncAXIS control-Instanz wird im Hardwaremodus aufgebaut.

Materialien

- Installation_Project.exe aus dem vorausgehenden Schritt
- syncAXISConfig.xml (noch ist darin der Simulationsmodus eingetragen) aus dem vorausgehenden Schritt

Vorgehen

(1) Stellen Sie sicher, dass die syncAXISConfig.xml die korrekten Werte enthält für:

- Hardwaremodus = <cfg:SimulationMode>>false⁽¹⁾
- **ACS** Motion Controller-IP-Adresse (ACSController)
- Pfad zu RTC6-Dateien (ProgramFileDirectory)
- Pfad zu(r) ct5-Datei(en) (CorrectionFilePath)
- RTC6-Seriennummer (SerialNumber)
- StageConfig:
 - StageAxisX
 - StageAxisY
- SlecEtherCATNodeID, siehe **Abbildung 6, Seite 27**

(1) In syncAXISConfig.xml:

```
<cfg:SimulationMode>>false
</cfg:SimulationMode>
```

(2) syncAXISConfig.xml:

```
<cfg:InitialOperationMode>ScannerAndStage
</cfg:InitialOperationMode>
```

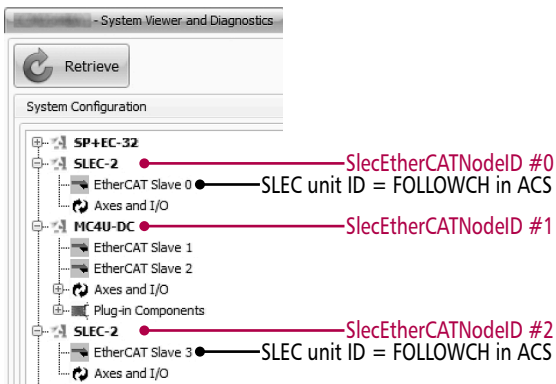
- (2) `Installation_Project.exe` starten.
Ein neues Fenster wird geöffnet, siehe [Abbildung 4, Seite 22](#).
- (3) Zum Initialisieren der **syncAXIS control**-Instanz (**INIT**) geben Sie ein:
 - '1' – Eingabetaste – ['1' oder '2'⁽¹⁾] – Eingabetaste
 Zum **ACS** Motion Controller wird eine IP-Verbindung aufgebaut, Verfahrtsch und Scan-Device werden akquiriert sowie die RTC6-Karte angesprochen.
 Wenn erfolgreich, wird im Fenster angezeigt: "Finished with return value 0".
 Wenn nicht erfolgreich, wird im Fenster angezeigt: "Finished with return value <nn>".
 Siehe nebenstehende [Hinweise](#).
 - Zur Fehlerabfrage (**CHECK_FOR_ERRORS**) geben Sie ein:
 - '13' – Eingabetaste
 - Zum Löschen der **syncAXIS control**-Instanz (**DELETE_INSTANCE**) geben Sie ein:
 - '12' – Eingabetaste
 - Beheben Sie den Fehler und wiederholen Sie diesen Schritt 3.
- (4) Zum Beenden (**EXIT**) von `Installation_Project.exe` geben Sie ein:
 - '0' – Eingabetaste

(1) Siehe Fußnote 4, [Seite 24](#).

Hinweise

- In diesem Schritt "**Installation_Project.exe** starten und **syncAXIS control**-Instanz im **Hardwaremodus initialisieren**", [Seite 26](#) kommt es häufig zu Fehlern. Wenn nicht "Finished with return value 0" angezeigt wird, liegt meistens ein Konfigurations- oder Verkabelungsfehler vor:
 - Ein häufiger Fehler ist die Verwechslung von `SlecEtherCATNodeID` und `Slec unit ID`. Zur Identifizierung siehe [Abbildung 6, Seite 27](#). Die `SlecEtherCATNodeID` wird in der `syncAXISConfig.xml` angegeben und kann einen anderen Wert haben als die `Slec unit ID`. Die `Slec unit ID` wird auf dem **ACS** Motion Controller in der Variable `FOLLOWCH` verwendet.
 - Prüfen Sie erneut die Verkabelung.
 - Stellen Sie sicher, dass alle Geräte bestromt sind.
 - Stellen Sie sicher, dass die EtherCAT-Kommunikation zwischen **Windows-PC** und dem **ACS** Motion Controller besteht.
 - Führen Sie einen RTC6-Karten-PowerCycle durch, wenn es einen entsprechenden Fehler gibt.
- Häufig auftretende Fehlermeldungen und deren Deutung zeigt [Kapitel 14 "Anhang G: Über das Error Handling mit syncAXIS control"](#), [Seite 59](#) und [Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"](#), [Kapitel 2.7.1 "Über die Puffer der syncAXIS control-Instanzen"](#), [Seite 42](#).

ACS SPiiPlus MMI Application Studio



6

ACS SPiiPlus MMI Application Studio > System Viewer and Diagnostics: `SlecEtherCATNodeIDs` und `Slec unit IDs` feststellen. Siehe auch [Hinweise, Seite 27](#). Weitere Informationen finden Sie in "AN ACS Components in XL SCAN System" und "SLEC EtherCAT Node Installation Guide".

6.2 Prüfen, ob Laserstrahlung ausgegeben wird

Im vorausgehenden Schritt **Kapitel 6.1** "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren", Seite 26 wurde mit dem Installation_Project.exe und der syncAXISConfig.xml erfolgreich eine syncAXIS control-Instanz im Hardwaremodus⁽¹⁾ initialisiert.

In diesem Schritt **6.2** soll nun mit dieser syncAXIS control-Instanz der Laser angesteuert werden. Dazu wird CHECK_LASER_FIRING benutzt.

Nachdem der Laser aktiviert wurde, emittiert er solange, bis er wieder deaktiviert wird. Stellen Sie also sicher, dass alle Sicherheitsvorkehrungen getroffen sind.

Wenn der Laser nach seiner Aktivierung tatsächlich emittiert, bedeutet dies:

- Laser und RTC6-Karte sind korrekt verbunden
- In der syncAXISConfig.xml sind wenigstens TTL Level, Leistung, Pinbelegung und Puls erzeugung "richtig" definiert (insofern, als dass Laserstrahlung tatsächlich ausgegeben wird)
- Die Kommunikation zur RTC6-Karte funktioniert

Nachdem festgestellt wurde, dass der Laser Strahlung ausgibt, kann (wahlweise) CHECK_LASER_FIRING bei dieser Gelegenheit gleich auch noch genutzt werden, um Leistungs-Werte und Pulsfrequenz-Werte zu optimieren (so dass später das Werkstückmaterial "sauber" bearbeitet werden wird).

Ziele

- Den Laser (über die RTC6-Karte) mit syncAXIS control in Modus "Manuelle Positionierung" aktivieren.
- Wahlweise: Zusätzlich Leistungs-Werte und Pulsfrequenz-Werte so anpassen, dass "gute" Markierungsergebnisse erzielt werden können.

Materialien

- Installation_Project.exe aus dem vorausgehenden Schritt
- syncAXISConfig.xml aus dem vorausgehenden Schritt
- Markiersubstrat (inkl. Reservevorrat)

(1) Siehe Fußnote 1 auf Seite 26.

Hinweise

- Mit CHECK_LASER_FIRING werden weder Scan-Kopf noch Verfahrtschicht verfahren.

Voraussetzungen

- Alle Sub-Systeme (z. B. RTC6-Karte, Laser) sind betriebsbereit.
- Ein Markiersubstrat ist eingelegt.

Vorgehen

- **⚠️ Warnung!** Verletzungsgefahr durch Laserstrahlung! Lasersicherheitsbestimmungen einhalten!
- (1) Stellen Sie sicher, dass syncAXIS control im Hardwaremodus⁽²⁾ arbeiten wird.
- (2) Starten Sie Installation_Project.exe und Initialisieren Sie eine syncAXIS control-Instanz (INIT) wie in Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren", Seite 26 beschrieben.
- (3) Starten Sie CHECK_LASER_FIRING wie folgt:
 - '2' – Eingabetaste (der Job wird gestartet und fragt nach Aktivierung des Lasers) – dann zur Aktivierung der Laserstrahlung y – Warnung! es wird im direkten Anschluss Laserstrahlung ausgegeben: Eingabetaste – dann zum Beenden der Strahlung eine Taste auf der Tastatur drücken – Schleife: es wird wieder nach der Aktivierung der Lasers gefragt⁽³⁾; zum Verlassen der Schleife geben Sie ein: n – Eingabetaste.
- (4) Zum Löschen der syncAXIS control-Instanz (DELETE_INSTANCE) geben Sie ein:
 - '12' – Eingabetaste
- (5) Zum Beenden (EXIT) von Installation_Project.exe geben Sie ein:
 - '0' – Eingabetaste

- (2) In syncAXISConfig.xml:


```
<cfg:SimulationMode>false
</cfg:SimulationMode>
```

- (3) Wahlweise an dieser Stelle: Leistungs-Wert und Pulsfrequenz-Wert in syncAXISConfig.xml ändern und die Schleife wiederholen.

6.3 Überprüfen der korrekten Kommunikation mit Scan-Kopf und Verfahrtsch

Im vorausgehenden Schritt **Kapitel 6.2 "Prüfen, ob Laserstrahlung ausgegeben wird"**, Seite 28 wird gezeigt, dass die RTC6-Karte erfolgreich mit der 'ersten' Hardwarekomponente, dem Laser, kommuniziert. Die Kommunikation von RTC6-Karte und Laser ist die Grundvoraussetzung zum weiteren Nachweis der Scan-Kopf-Bewegung.

In diesem Schritt **6.3** sollen nun mit einer **syncAXIS control-Instanz** im Hardwaremodus die weiteren Hardwarekomponenten Scan-Kopf und Verfahrtsch angesteuert werden.

Dazu wird **REFERENCE_RUN** ausgeführt.

In **REFERENCE_RUN** fährt zunächst der Verfahrtsch ein Rechteck (Größe: vom Anwender eingegeben) und dann der Scan-Kopf (Größe: 10 mm). In den Ecken des Scan-Kopf-Rechtecks wird der Laser kurz aktiviert, um eine Scan-Kopf-Bewegung nachzuweisen.

Diese Vorgänge erfolgen im Modus "Manuelle Positionierung". Deshalb sind weder **Trajektorien-Planung** (inkl. Bewegungsaufteilung) noch die Kommunikation zu den **ACS**-Komponenten (über SL2-100-zu-EtherCAT-Konverter) involviert.

Stattdessen werden lediglich die korrekte Kommunikation und Funktionalität der Komponenten geprüft.

Falls **REFERENCE_RUN** wie beschrieben ausgeführt wird, bedeutet das:

- Die Kommunikation zu **ACS**-Komponenten über Ethernet TCP/IP funktioniert
- Die **ACS**-Achsen sind bereit zur Bewegung
- In der **syncAXISConfig.xml** sind **StageAxisX** und **StageAxisY** korrekt definiert
- Der Betrieb des excellSCAN funktioniert korrekt

Ziele

- Die korrekte Kommunikation und Funktionalität der Hardwarekomponenten Scan-Kopf und Verfahrtsch ist bestätigt

Materialien

- **Installation_Project.exe** aus dem vorausgehenden Schritt
- **syncAXISConfig.xml** aus dem vorausgehenden Schritt
- Markiersubstrat (inkl. Reservevorrat)

Voraussetzungen

- Alle Sub-Systeme (z.B. Laser) sind betriebsbereit.
- Ein Markiersubstrat ist eingelegt.

Vorgehen

- **⚠ Warnung!** Verletzungsgefahr durch Laserstrahlung! Lasersicherheitsbestimmungen einhalten!
 - **⚠ Warnung!** Verletzungsgefahr durch Verfahrtschbewegung! Keine Personen im Gefahrenbereich!
 - **⚠ Vorsicht!** Gefahr der Sachbeschädigung durch Verfahrtschbewegung! Keine Fremdgegenstände im Gefahrenbereich!
- (1) Stellen Sie sicher, dass **syncAXIS control** im Hardwaremodus⁽¹⁾ arbeiten wird.
 - (2) Starten Sie **Installation_Project.exe** und initialisieren Sie eine **syncAXIS control-Instanz (INIT)** wie in **Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren"**, Seite 26 beschrieben.
 - (3) Starten Sie **REFERENCE_RUN** wie folgt:
 - **'3'** – Eingabetaste – dann die Geschwindigkeit, mit der der Verfahrtsch verfahren werden soll – Eingabetaste – dann die Rechteckgröße die vom Verfahrtsch gefahren werden soll – Eingabetaste – y – Eingabetaste.
 Der **Job** wird für den Modus "Manuelle Positionierung" (keine **Trajektorien-Planung**, keine Bewegungsaufteilung) berechnet. Der erfolgreiche Abschluss der Vorgänge wird durch "Finished with return value 0" angezeigt.
 - (4) Zum Löschen der **syncAXIS control-Instanz (DELETE_INSTANCE)** geben Sie ein:
 - **'12'** – Eingabetaste
 - (5) Zum Beenden (**EXIT**) von **Installation_Project.exe** geben Sie ein:
 - **'0'** – Eingabetaste

(1) Siehe Fußnote Seite 28.

6.4 Den (vorgängig in 5.3) simulierten "TEST_MARKING" markieren

In den vorausgehenden Schritten wurde gezeigt:

- Die Hardware ist betriebsbereit
- Die Systemkomponenten kommunizieren miteinander

Diejenige Operation, die es noch mit syncAXIS control auszuführen gilt, ist die kombinierte Laserbearbeitung durch Scan-Kopf und Verfahrtsch.

Mit den Verfahrtsch-Achsen wird in diesem Fall nicht über Ethernet TCP/IP kommuniziert, sondern über den SL2-100-zu-EtherCAT-Konverter. Zudem wird der Listenspeicher der RTC6-Karte mit RTC6-Befehlen gefüllt und dann in Echtzeit ausgeführt.

In Kapitel 5.3 "TEST_MARKING" im Betriebsmodus "ScannerAndStage" simulieren", Seite 24 haben Sie den Job TEST_MARKING simuliert, evaluiert und dann sichergestellt, dass es bei der Ausführung zu keinen Grenzwertüberschreitungen kommt. Deswegen dürfen Sie im vorliegenden Schritt 6.4 nun diesen Job auf dem XL SCAN-System ausführen.

Ein erfolgreicher Abschluss von 6.4 sagt aus:

- Die Hardware ist korrekt konfiguriert
- Die Hardware ist korrekt verkabelt
- Die Hardware wird korrekt initialisiert
- Die Einträge in der syncAXISConfig.xml sind Software-technisch korrekt, jedoch noch nicht zwangsläufig "optimal"

Ziele

- TEST_MARKING wird tatsächlich (= im Hardwaremodus) erfolgreich markiert:
 - im Betriebsmodus "ScannerAndStage"

Materialien

- Installation_Project.exe aus dem vorausgehenden Schritt
- syncAXISConfig.xml aus dem vorausgehenden Schritt
- Markiersubstrat (inkl. Reservevorrat)

Voraussetzungen

- Alle Sub-Systeme (z.B. Laser) sind betriebsbereit.
- Ein Markiersubstrat ist eingelegt.

Vorgehen

- **⚠ Warnung!** Verletzungsgefahr durch Laserstrahlung! Lasersicherheitsbestimmungen einhalten!
 - **⚠ Warnung!** Verletzungsgefahr durch Verfahrtschbewegung! Keine Personen im Gefahrenbereich!
 - **⚠ Vorsicht!** Gefahr der Sachbeschädigung durch Verfahrtschbewegung! Keine Fremdgegenstände im Gefahrenbereich!
- (1) Stellen Sie sicher, dass syncAXIS control im Hardwaremodus⁽¹⁾ arbeiten wird.
 - (2) Stellen Sie sicher, dass syncAXIS control im Betriebsmodus "ScannerAndStage"⁽²⁾ arbeiten wird.
 - (3) Starten Sie Installation_Project.exe und Initialisieren Sie eine syncAXIS control-Instanz (INIT) wie in Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren", Seite 26 beschrieben.

(1) Siehe Fußnote Seite 28.

(2) Siehe Fußnote Seite 26.

(4) Starten Sie **TEST_MARKING** wie folgt:

- '4' – Eingabetaste – dann die tatsächliche Größe des Markierobjekts (in mm, sollte kleiner als das kombinierte Gesamt-Arbeitsfeld sein) – y.
Der **Job** wird für den Betriebsmodus "ScannerAndStage" mitsamt **Trajektorien**-Planung und mit Bewegungsaufteilung berechnet.
Der erfolgreiche Abschluss der Vorgänge wird durch "Finished with return value 0" angezeigt.

(5) Zum Löschen der **syncAXIS control-Instanz** (**DELETE_INSTANCE**) geben Sie ein:

- '12' – Eingabetaste

(6) Zum Beenden (**EXIT**) von `Installation_Project.exe` geben Sie ein:

- '0' – Eingabetaste

(7) Fahren Sie mit nachfolgender Prozedur "**Evaluieren**" fort.

Evaluieren

- (1) Sollte die Markierung nicht vollständig durchgeführt worden sein, prüfen Sie auf Fehlermeldungen. Dafür können Sie die Log-Datei (siehe `<cfg:LogfilePath>`) sowie das ACS SPiiPlus MMI Application Studio verwenden. Beachten Sie auch die **Kapitel 14 "Anhang G: Über das Error Handling mit syncAXIS control"**, Seite 59 und **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**, Kapitel 2.7.1 "**Über die Puffer der syncAXIS control-Instanzen**", Seite 42.
- (2) Führen Sie eine Sichtprüfung des Markierergebnisses unter Vergrößerung (z. B. mit Auflichtmikroskop) durch:
 - Wenn es grob den Erwartungen aus der Simulation entspricht, fahren Sie fort mit **Kapitel 7 "Optimierung und Verifikation"**, Seite 32
 - Wenn es grob von der Simulation abweicht, prüfen Sie erneut die korrekte Achs-Zuweisung, Bewegung aller Komponenten etc. und führen Sie Schritt 6.4 erneut durch.

7 Optimierung und Verifikation

Die Installation ist mit **Kapitel 6.4 "Den (vorgängig in 5.3) simulierten "TEST_MARKING" markieren", Seite 30** abgeschlossen, da die Kommunikation aller Komponenten erfolgreich demonstriert wurde und erste Markierungen erfolgt sind.

Zur Erzielung von optimalen Markiierungsergebnissen und der erfolgreichen Arbeit mit XL SCAN, sind noch einige wenige Optimierungs- und Verifikations-schritte erforderlich.

In den nachfolgend beschriebenen Schritten wird davon ausgegangen, dass Sie mit dem Handling der `Installation_Project.exe` vertraut sind und vor jeder Markierungsausführung zur Sicherheit eine Simulation durchführen.

Für alle Schritte sind erforderlich:

- `syncAXISConfig.xml`
- `Installation_Project.exe`
- Markiersubstrate

In diesem Kapitel:

- **Kapitel 7.1 "Optimierung – Laser-Delays finden, die zu Ihrer Anwendung passen", Seite 33**
- **Kapitel 7.2 "Optimierung – Das Scan-Kopf-Arbeitsfeld kalibrieren", Seite 36**
- **Kapitel 7.3 "Verifikation – Genauigkeit der statischen Kalibrierungen von Scan-Kopf und ACS-Achsen prüfen", Seite 39**
- **Kapitel 7.4 "Verifikation – Synchronisation aller System-Komponenten prüfen", Seite 41**
- **Kapitel 7.5 "Verifikation – Genauigkeit der kombinierten Bewegung bestimmen", Seite 43**

7.1 Optimierung – Laser-Delays finden, die zu Ihrer Anwendung passen

In diesem Schritt **7.1** sind die Laser-Delay-Werte zu optimieren:

- `LaserSwitchOffsetTime` – Zeit, um die die Ausgabe der Lasersignale verschoben wird⁽¹⁾
- `LaserPreTriggerTime` – Zeit, in der das Lasersignal im Voraus getriggert wird, wenn ein Markierabschnitt ausgeführt wird⁽¹⁾

Dazu wird `CHECK_LASERDELAYS` ausgeführt.

`CHECK_LASERDELAYS` enthält ein 11×11 -Raster aus kleinen Markierobjekten, siehe **Abbildung 7, Seite 33**. Zwischen den Markierobjekten werden die Laser-Delays automatisch verändert:

- `LaserSwitchOffsetTime` wird erhöht in positiver x-Richtung
- `LaserPreTriggerTime` wird erhöht in positiver y-Richtung

Als Start-Werte (in Schritt **4, Seite 34**) werden empfohlen:

- `LaserSwitchOffsetTime`: $-40 \mu\text{s}$ bis $+10 \mu\text{s}$

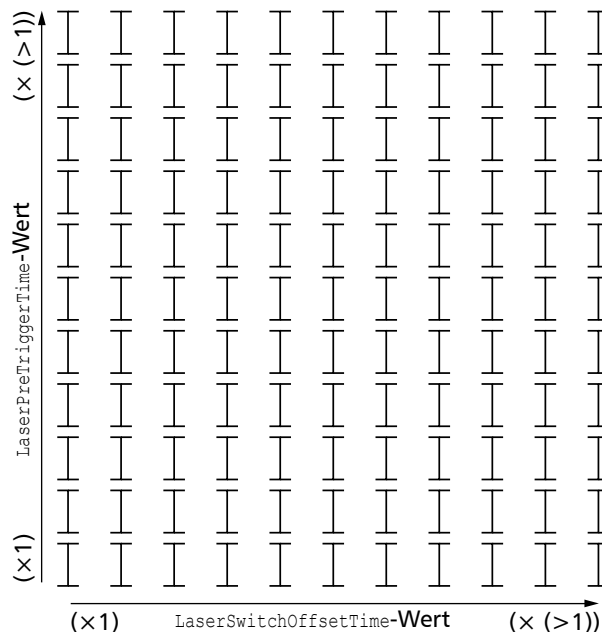
- `LaserPreTriggerTime`: $-10 \mu\text{s}$ bis $+10 \mu\text{s}$

(1) Siehe **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**, **Abbildung 41, Seite 296**.

Führen Sie diesen Schritt **7.1** gegebenenfalls mehrfach aus, um die Variation der Laser-Delays zwischen den Markierobjekten zu verringern. Nach der Markierung evaluieren Sie das Ergebnis und entscheiden sich für dasjenige markierte Objekt, das am besten Ihre Anwendung repräsentiert, siehe **Abbildung 8, Seite 35**. Anhand dieser Wahl bestimmen Sie Ihre Werte von `LaserSwitchOffsetTime` und `LaserPreTriggerTime` (die Sie als Start-Werte für einen weiteren optionalen Durchlauf dieses Schritts **7.1** benutzen können).

Achtung!

In diesem Schritt **7.1** werden die Laser-Delays angepasst, bevor im nachfolgenden Schritt **7.2** die Scan-Kopf-Arbeitsfeld-Kalibrierung durchgeführt wird. Das ist wichtig, weil eine gute Kalibrierung ein gutes Markierergebnis voraussetzt. Trotzdem sollte nach der Scan-Kopf-Arbeitsfeld-Kalibrierung **7.2** (das Positionieren der Spiegel ändert sich) dieser Schritt **7.1** erneut ausgeführt werden.



Definiertes Raster in `CHECK_LASERDELAYS`.

Ziele

- Die für Ihre Anwendung optimalen LaserSwitchOffsetTime-Wert und LaserPreTriggerTime-Wert finden und in der syncAXISConfig.xml eintragen.

Materialien

- Markiersubstrat (inkl. Reservevorrat)

Voraussetzungen

- Die Markiermuster müssen zunächst (wie immer vor einer allerersten Ausführung) simuliert worden sein, um sicherzustellen, dass keine Grenzwertüberschreitungen in den Ansteuerwerten enthalten sind.
- Alle Sub-Systeme (z.B. Laser) sind betriebsbereit.
- Ein Markiersubstrat ist eingelegt.

Vorgehen

- ⚠️ Warnung!** Verletzungsgefahr durch Laserstrahlung! Lasersicherheitsbestimmungen einhalten!
 - ⚠️ Warnung!** Verletzungsgefahr durch Verfahrtschbewegung! Keine Personen im Gefahrenbereich!
 - ⚠️ Vorsicht!** Gefahr der Sachbeschädigung durch Verfahrtschbewegung! Keine Fremdgegenstände im Gefahrenbereich!
- Starten Sie Installation_Project.exe und Initialisieren Sie eine syncAXIS control-Instanz (INIT) wie in Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren", Seite 26 beschrieben.
 - Starten Sie CHECK_LASERDELAYS wie folgt:
 - '8' – Eingabetaste – dann die höchste Geschwindigkeit, die Ihr Scan-Kopf erreichen kann (je höher die Geschwindigkeit, desto größer der Einfluss 'schlecht' gewählter Laser-Delays) – Eingabetaste – dann den LaserSwitchOffsetTime-Start-Wert – Eingabetaste – dann den Inkrement-Wert für LaserSwitchOffsetTime – Eingabetaste – dann den LaserPreTriggerTime-Start-Wert – Eingabetaste – dann den Inkrement-Wert für LaserPreTriggerTime – Eingabetaste – y – Eingabetaste.
 - Die Jobs werden für den Betriebsmodus "ScannerOnly" mitsamt Trajektorien-Planung,

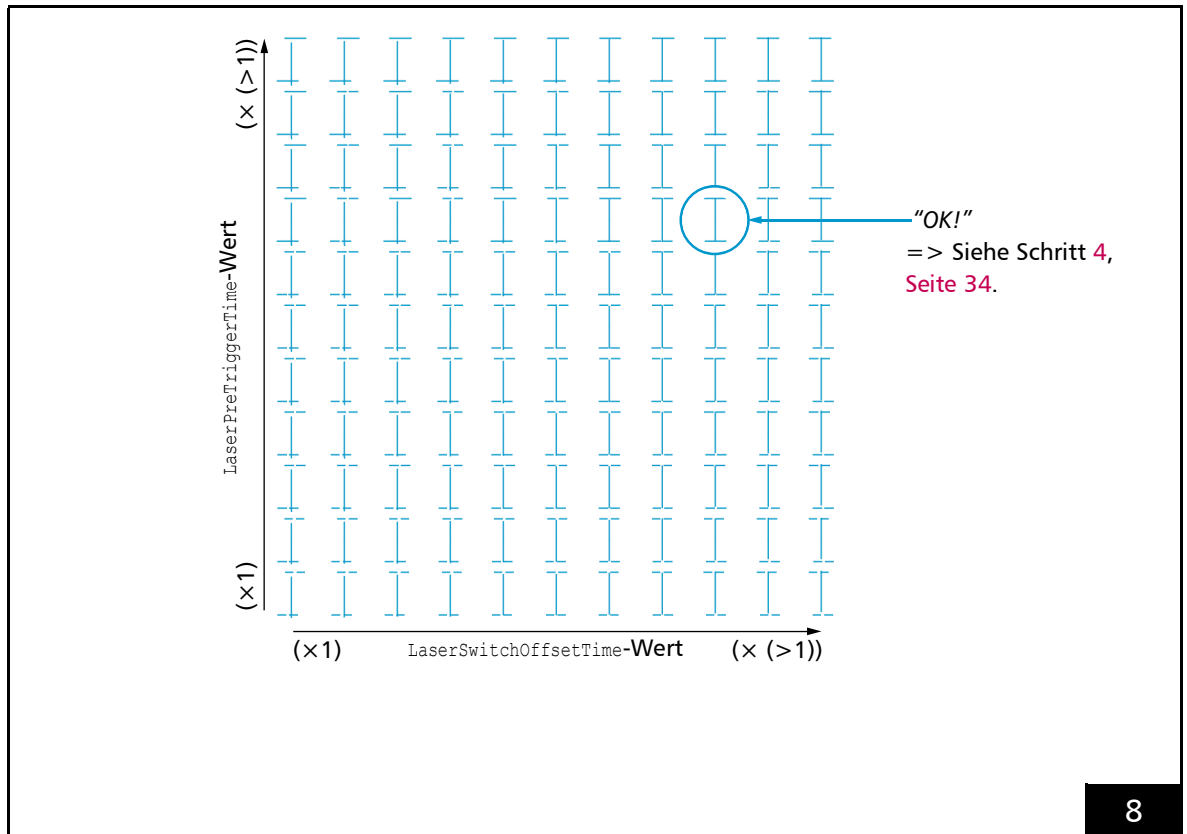
aber ohne Bewegungsaufteilung berechnet. Der erfolgreiche Abschluss der Vorgänge wird durch "Finished with return value 0" angezeigt.

- Beenden Sie Installation_Project.exe (DELETE_INSTANCE > EXIT) wie beschrieben in Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren", Seite 26.
- Fahren Sie mit nachfolgender Prozedur "Evaluieren" fort.

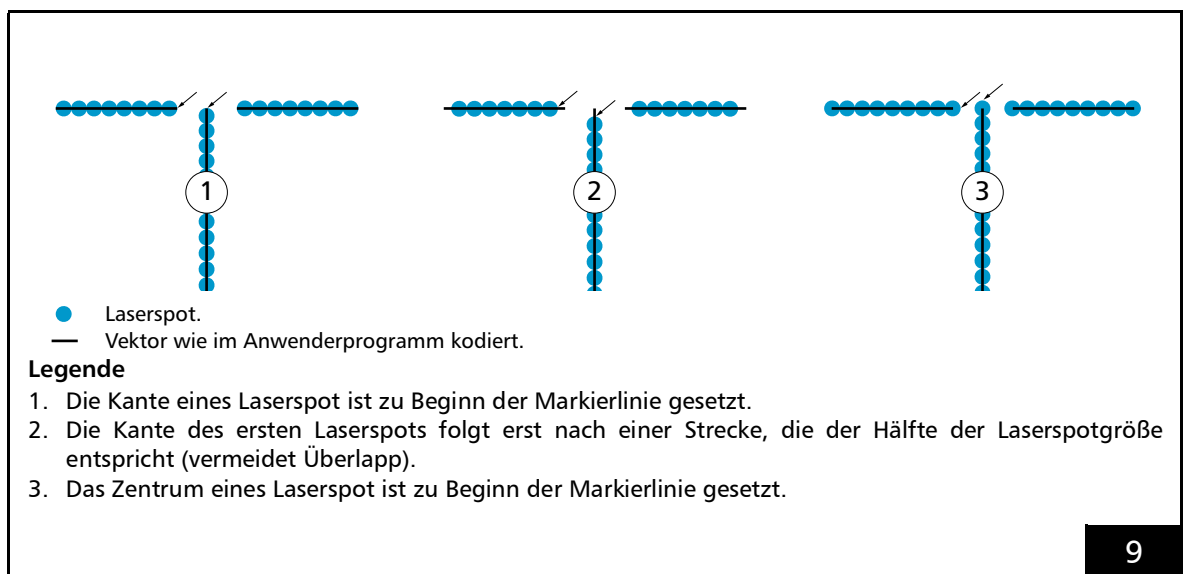
Evaluieren

- Sollten die Markierungen nicht vollständig durchgeführt worden sein, prüfen Sie auf Fehlermeldungen. Dafür können Sie die Log-Datei (siehe <cfg:LogfilePath>) sowie das ACS SPiPlus MMI Application Studio verwenden. Beachten Sie auch die Kapitel 14 "Anhang G: Über das Error Handling mit syncAXIS control", Seite 59 und Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung", Kapitel 2.7.1 "Über die Puffer der syncAXIS control-Instanzen", Seite 42.
- Führen Sie eine Sichtprüfung des Markierungsergebnisses unter Vergrößerung (z. B. mit Auflichtmikroskop) durch. Es sollte in etwa so aussehen wie in Abbildung 8, Seite 35.
- Identifizieren Sie im Markierungsergebnis (Abbildung 8, Seite 35) das für Ihre Applikation geeignetste Markierobjekt. Einige relevante Kriterien zeigt Abbildung 9, Seite 35.
- Stellen Sie den dafür verwendeten LaserSwitchOffsetTime-Wert und LaserPreTriggerTime-Wert fest. Angenommen, die ideale Markierung wäre in x-Richtung an Stelle 9 und in y-Richtung an Stelle 4, dann gilt:
 - $\text{LaserSwitchOffsetTime} = -40 \mu\text{s} + (9-1) \times 5 \mu\text{s} = 0 \mu\text{s}$
 - $\text{LaserPreTriggerTime} = -10 \mu\text{s} + (4-1) \times 2 \mu\text{s} = -4 \mu\text{s}$
- Tragen Sie diese Werte in die syncAXISConfig.xml ein.

(6) Wahlweise können Sie die 7.1 mehrmals mit jeweils leichten Variationen dieser Werte wiederholen.



Evaluieren des Markierergebnisses (Beispiel), siehe Seite 34 in Kapitel 7.1 "Optimierung – Laser-Delays finden, die zu Ihrer Anwendung passen", Seite 33.



Kriterien zur Beurteilung des Markierergebnisses in Schritt 3.

7.2 Optimierung – Das Scan-Kopf-Arbeitsfeld kalibrieren

In diesem Schritt ist das Arbeitsfeld eines jeden der Scan-Köpfe zu kalibrieren. Das Ziel dieser ("Arbeitsfeldkalibrierungs"-)Prozedur ist, eine *optimierte* ct5-Datei⁽¹⁾ für jeden Scan-Kopf⁽²⁾ zu erhalten.

Hier ist eine Vorgehensweise unter Verwendung von SCANLAB correXion Pro, sowie einer kundeneigenen Koordinatenmessmaschine⁽³⁾ beschrieben.

Eine alternative Vorgehensweise finden Sie in Kapitel 13 "Anhang F: Scan-Kopf-Arbeitsfeld kalibrieren – Alternative mit der CalSheet-Software", Seite 57.

Eine weitere, "gemeinsame" Kalibrierung ist nicht notwendig.

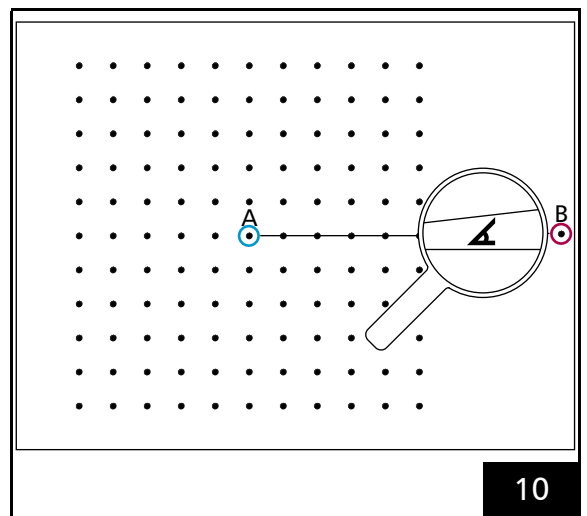
Prinzipieller Ablauf

- (1) **Voraussetzungen**, Seite 37 einhalten.
- (2) Ein Raster (z. B. regelmäßige Anordnung aus gefüllten Kreisen; gewöhnlich 11×11 , immer ungerade Zahlen) definieren, das sich über das gesamte Scan-Kopf-Arbeitsfeld erstreckt.
Dieses Markiermuster ist in **SCANNER_CALI** enthalten.
- (3) Ein einzelnes Rasterobjekt (gefüllter Kreis) definieren.
Dieses Markiermuster ist in **SCANNER_CALI** enthalten.
- (4) Verfahrtsch bei (0,0) mm (Arbeitsfeldmitte = "Nullposition") belassen und das Raster nur mit dem Scan-Kopf (ohne Verfahrtsch-Bewegung) markieren.
- (5) Verfahrtsch (entweder nur in X- oder nur in Y-Richtung) so bewegen, dass die Arbeitsfeldmitte nun außerhalb des soeben markierten Rasters ist.
- (6) Das einzelne Rasterobjekt in die Arbeitsfeldmitte markieren.

- (7) Markierergebnis, siehe **Abbildung 10, Seite 36**, mit Koordinatenmessmaschine vermessen⁽⁴⁾.
Anmerkung: Die Koordinatenmessmaschine vermisst die Lage der 11×11 Rasterobjekte relativ zum Koordinatensystem des Verfahrtsch (als dessen x-Achse wird dazu die Gerade von "A" zu "B" verwendet).

- (8) Eine dat-Datei basierend auf den Messergebnissen erstellen⁽⁵⁾.

- (9) *Optimierte* ct5-Datei mittels correXion Pro generieren.



Markierergebnis in Schritt 1, Seite 38: Typischerweise sind Scan-Kopf-Koordinatensystem und Verfahrtsch-Koordinatensystem gegeneinander um einen bestimmten Winkel leicht verdreht. Deshalb liegen bei Verwendung einer nicht-optimierten ct5-Datei Scan-Kopf-Koordinatensystem-Mittelpunkt (A blau) und Verfahrtsch-Koordinatensystem-Mittelpunkt (B rot) nicht auf gleicher Höhe. In der optimierten ct5-Datei soll dieser Winkel kompensiert sein. Dazu vermisst die Messmaschine die Lage der 11×11 Gitterpunkte (= Scan-Kopf-Koordinatensystem) relativ zum Verfahrtsch-Koordinatensystem (als x-Achse des Verfahrtsch-Koordinatensystems wird die Linie "A-zu-B" verwendet).

- (1) Wesentliche Eigenschaften der optimierten ct5-Datei sind:
 1. Bei der Ansteuerung im Scan-Kopf-Arbeitsfeld ist der Fehler noch kleiner als bei der ct5-Datei, die standardmäßig geliefert wurde (= bessere Scan-Kopf-Arbeitsfeld-Kalibrierung).
 2. Sie gleicht die Rotation von Verfahrtsch vs. Scan-Kopf aus.
- (2) Die Prozedur (Kap. 7.2) sollte wiederholt werden, wenn Sie Drifteffekte des Scan-Kopfs feststellen (um diese Drifteffekte auszugleichen).

- (3) Alle Anwender, die sich für ein in diesem Handbuch nicht beschriebenes Verfahren entscheiden, müssen den Sicherheitshinweis auf Seite 38 befolgen.
- (4) Die Position der Kreise kann z. B. im entsprechenden Bild-Tool über das "center of gravity" bestimmt werden.
- (5) Siehe Hinweise im correXion Pro-Handbuch und die Template-Datei im Programmverzeichnis von correXion Pro.

Materialien

- Standard-ct5-Datei zum Markieren der Markiermuster.
- Passende⁽¹⁾ Markiersubstrate (inkl. Reservevorrat).
- Eine kundeneigene Koordinatenmessmaschine (inkl. Bildverarbeitungssoftware). Das Gerät muss das Markierergebnis aufnehmen und auswerten können.
- correXion Pro inkl. Handbuch.

Voraussetzungen

Vor dem Ausführen dieser Prozedur müssen folgende Punkte erfüllt sein:

- Der Ausgleich der statischen Fehler des Verfahrtschiffs wurde ausgeführt (wichtige Voraussetzung für die Ausrichtungskorrektur)
- Das **Error-Mapping** von **ACS** für den Verfahrtschiff wurde angewandt und ist aktiv.
- Die Strahl-Justage wurde ausgeführt – siehe Handbuch "Installation und Inbetriebnahme" des Scan-Kopfs, Abschnitt "Justage" bezüglich der Werte von "Neigung"⁽²⁾ und "Versatz"⁽³⁾.
- Feineinstellung der Brennweite, der Laserleistungs-Parameter⁽⁴⁾ und Laser-Delays⁽⁵⁾ wurde ausgeführt. Die Markierergebnisqualität ist wichtig für die Bildauswertung. Achten Sie besonders darauf, dass die Geschwindigkeit in diesem Schritt vergleichsweise niedrig gewählt ist, sodass der Laserstrahlungseintrag pro Strecke eher hoch ist. Passen Sie die Laserleistungsparameter entsprechend an.

- (1) Das Markierergebnis darauf muss ausgewertet werden können.
- (2) engl.: "Tilt"
- (3) engl.: "Displacement"
- (4) Siehe Kapitel 6.2 "Prüfen, ob Laserstrahlung ausgegeben wird", Seite 28.
- (5) Siehe Kapitel 7.1 "Optimierung – Laser-Delays finden, die zu Ihrer Anwendung passen", Seite 33.

- Das Markieren muss mit einem syncAXIS control-basierten Anwenderprogramm erfolgen.
- Benutzen Sie zum Markieren und Springen relativ niedrige Geschwindigkeiten (z.B. 0,25 m/s), um so gut wie möglich dynamische Folgefehler zu vermeiden. Diese Prozedur verfolgt als Hauptziel die statische Scan-Kopf-Arbeitsfeld-Kalibrierung. Deswegen ist eine hohe Genauigkeit (und keine hohe Geschwindigkeit) nötig.
- Alle Sub-Systeme (z.B. Laser) sind betriebsbereit.
- Ein Markiersubstrat ist eingelegt.

Vorgehen

- **⚠️ Warnung!** Verletzungsgefahr durch Laserstrahlung! Lasersicherheitsbestimmungen einhalten!
 - **⚠️ Warnung!** Verletzungsgefahr durch Verfahrtschiffbewegung! Keine Personen im Gefahrenbereich!
 - **⚠️ Vorsicht!** Gefahr der Sachbeschädigung durch Verfahrtschiffbewegung! Keine Fremdgegenstände im Gefahrenbereich!
- (1) Starten Sie `Installation_Project.exe` und Initialisieren Sie eine **syncAXIS control-Instanz** (**INIT**) wie in Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren", Seite 26 beschrieben.
 - (2) Starten Sie `SCANNER_CALI` wie folgt:
 - '5' – Eingabetaste – dann die Größe des Rasters (kleiner als die Scan-Kopf-Arbeitsfeldsgröße) – Die Markierung erfolgt im Betriebsmodus `ScannerOnly` – Eingabetaste – dann die Anzahl der Rasterobjekte (Empfohlen 11×11) – Eingabetaste – dann die Geschwindigkeit zum Verfahren des Verfahrtschiffs – Eingabetaste – y – Eingabetaste.
 Der erfolgreiche Abschluss der Vorgänge wird durch "Finished with return value 0" angezeigt.
 - (3) Beenden Sie `Installation_Project.exe` (**DELETE_INSTANCE > EXIT**) wie beschrieben in Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren", Seite 26.
 - (4) Fahren Sie mit nachfolgender Prozedur "Evaluieren" fort.

Evaluieren

- (1) Markierergebnis, siehe **Abbildung 10, Seite 36**, mit Koordinatenmessmaschine vermessen.
 - Wählen Sie das Referenzsystem so, dass der Einzel-Punkt nach Verfahren des Verfahrtschis zusammen mit dem Punkt im Zentrum des Rasters die Referenz-x-Achse für die Messung ergibt. In diesem Fall wird das Scan-Kopf-Arbeitsfeld durch die Korrekturdatei automatisch auf das Verfahrtschis-Arbeitsfeld ausgerichtet. Falls die Messmethodik kein solches Feature zur Verfügung stellt, muss der Winkel manuell ermittelt werden. Dieser ist entweder bei der Auswertung der Messdaten einzubeziehen, oder später als Alignment-Matrix (siehe **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**) in der `syncAXISConfig.xml` zu verwenden.
- (2) Erstellen Sie basierend auf den Messergebnissen eine `dat`-Datei. Informationen zu Form und Inhalt finden Sie im `correXion Pro`-Handbuch.
- (3) Generieren Sie mit der `correXion Pro`-Software und der `dat`-Datei Ihre *optimierte* `ct5`-Datei.

Achtung!

Nach diesem Schritt **7.2** Scan-Kopf-Arbeitsfeld-Kalibrierung (das Positionieren der Spiegel ändert sich) sollte Schritt **7.1** erneut ausgeführt werden.

- (4) Stellen Sie sicher, dass ab sofort innerhalb `XL SCAN` nur noch die *optimierte* `ct5`-Datei aus Schritt **3** verwendet wird. Dazu ist die `syncAXISConfig.xml` entsprechend anzupassen, siehe `<cfg:CorrectionFileList>`.

Achtung!

- Wichtiger Hinweis für Anwender, die Scan-Kopf-Arbeitsfeld-Kalibrierungs-Prozeduren anders als in diesem Kapitel oder anders als in **Kapitel 13 "Anhang F: Scan-Kopf-Arbeitsfeld kalibrieren – Alternative mit der CalSheet-Software"**, Seite 57 beschrieben, ausführen:
 - Sie müssen den Wert für den "Scan Field Calibration K" selber bestimmen.
 - Sie müssen diesen Wert in die `syncAXISConfig.xml` eintragen^(a).
- (a) Bei `<cfg:CorrectionFilePath CalibrationFactor = -1.0>`. -1.0 ist der Default-Eintrag und ist entsprechend zu ändern. Bei -1.0 wird der Wert aus der `ct5`-Datei ausgelesen. Allerdings kann es sein, dass in Ihrer optimierten `ct5`-Datei kein/falscher Wert vorhanden ist (z.B. weil die optimierte `ct5`-Datei mit einem kundeneigenen Tool generiert wurde).

7.3 Verifikation – Genauigkeit der statischen Kalibrierungen von Scan-Kopf und ACS-Achsen prüfen

Vorabgehend wurden folgenden Kriterien bearbeitet:

- Die statische Kalibrierung des Scan-Kopfs (= Scan-Kopf-Arbeitsfeld-Kalibrierung)
- Die statische Kalibrierung des Verfahrtschis (= **Error-Mapping**)
- Die Anpassung der Laser-Delays

Es empfiehlt sich die Verifizierung dieser Kalibrierungen (im vorliegenden Kapitel beschrieben).

Dazu ist in **Abbildung 11, Seite 39** eine mögliche Herangehensweise illustriert. Als Markiermuster sind 2 Raster (hier: aus $19 \times 19^{(1)}$ Rasterobjekten (= Gitter"punkte") zu definieren. Sie sollten in das Scan-Kopf-Arbeitsfeld passen. Als Rasterobjekt wird für das eine Raster ein Kreuz verwendet, für das andere ein Kreis⁽²⁾.

- (1) Bei einer ungeradzahlgigen Anzahl kommt das zentrale Rasterobjekt bei (0|0) zu liegen.

Diese Markiermuster sind in **CHECK_CALIBRATIONS** enthalten.

Dann werden die beiden Markiermuster (Reihenfolge egal) nacheinander auf ein und dasselbe Markier-substrat markiert (*zwischenzeitlich nicht bewegen*):

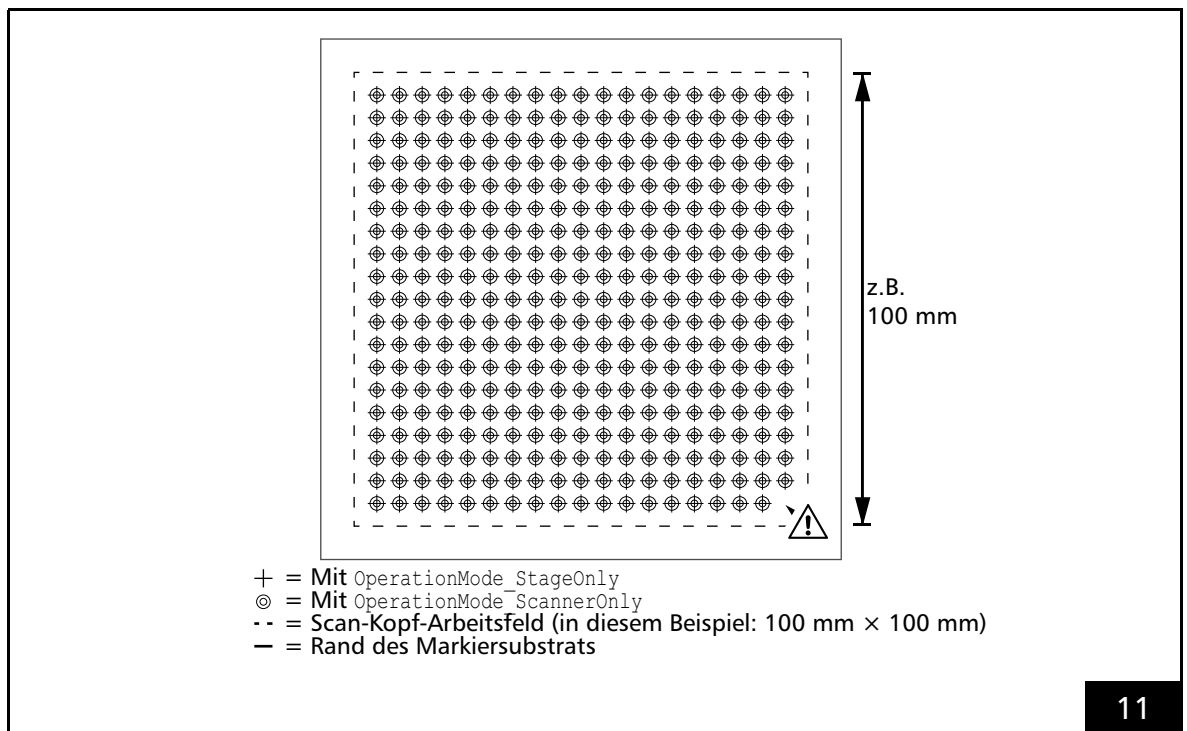
- Das Raster aus Kreuzen im Betriebsmodus "StageOnly"
- Das Raster aus Kreisen im Betriebsmodus "ScannerOnly"

Das Markierergebnis wird anschließend einer Sichtprüfung unterzogen. Kriterien:

- Keine Verdrehung der Gitter zueinander
- Die sich entsprechenden Einzel-Rasterobjekte kommen (möglichst) zur Deckung

Bei Bedarf kann auch eine quantitative Auswertung mittels Koordinatenmessmaschine vorgenommen werden.

- (2) Durch das Weglassen eines Rasterobjekts (hier analog zu QR-Codes: rechts unten) kann man im Markierergebnis sofort die Orientierung des Gitters feststellen. Weiterer Vorteil: siehe Bildunterschrift von **Abbildung 11, Seite 39**.



Beispielhaftes Markierergebnis in Kapitel 7.3 "Verifikation – Genauigkeit der statischen Kalibrierungen von Scan-Kopf und ACS-Achsen prüfen", Seite 39.

Beide Markiermuster wurden so angelegt, dass rechts unten (Warndreieck) ein Rasterobjekt fehlt. Ist im Markierergebnis an dieser Stelle dennoch eine Markierung zu sehen, sind die Koordinatensysteme nicht korrekt definiert.

Ziele

- Die Qualität der Kalibrierungen von Verfahrtsch und Scan-Kopf ist bestätigt.

Materialien

- Markiersubstrat (inkl. Reservevorrat)
- Bei Bedarf zur quantitativen Auswertung: kundeneigene Koordinatenmessmaschine (oder Kamera-basierte Lösung)

Voraussetzungen

- Die Markiermuster müssen zunächst (wie immer vor einer allerersten Ausführung) simuliert werden, um sicherzustellen, dass keine Grenzwertüberschreitungen in den Ansteuerwerten enthalten sind.
- Alle Sub-Systeme (z.B. Laser) sind betriebsbereit.
- Ein Markiersubstrat ist eingelegt.

Vorgehen

- ⚠️ Warnung!** Verletzungsgefahr durch Laserstrahlung! Lasersicherheitsbestimmungen einhalten!
 - ⚠️ Warnung!** Verletzungsgefahr durch Verfahrtschbewegung! Keine Personen im Gefahrenbereich!
 - ⚠️ Vorsicht!** Gefahr der Sachbeschädigung durch Verfahrtschbewegung! Keine Fremdgegenstände im Gefahrenbereich!
- Starten Sie `Installation_Project.exe` und Initialisieren Sie eine **syncAXIS control-Instanz** (`INIT`) wie in **Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren"**, Seite 26 beschrieben.
 - Starten Sie `CHECK_CALIBRATIONS` wie folgt:
 - '9' – Eingabetaste – dann die Größe des Rasters (kleiner als die Scan-Kopf-Arbeitsfeldgröße) – Eine der Markierungen erfolgt im Betriebsmodus `ScannerOnly` – Eingabetaste – dann die Anzahl der Rasterobjekte (Empfohlen: 15 × 15) – Eingabetaste – dann die Geschwindigkeit zum Verfahren des Verfahrtschs – Eingabetaste – y – Eingabetaste.
 Die **Jobs** werden jeweils für den Betriebsmodus `"ScannerOnly"` und `"StageOnly"` mitsamt **Trajektorien-Planung**, aber ohne

Bewegungsaufteilung berechnet.

Der erfolgreiche Abschluss der Vorgänge wird durch "Finished with return value 0" angezeigt.

- Beenden Sie `Installation_Project.exe` (`DELETE_INSTANCE > EXIT`) wie beschrieben in **Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren"**, Seite 26.
- Fahren Sie mit nachfolgender Prozedur **"Evaluieren"** fort.

Evaluieren

- Sollten die Markierungen nicht vollständig durchgeführt worden sein, prüfen Sie auf Fehlermeldungen. Dafür können Sie die Log-Datei (siehe `<cfg:LogfilePath>`) sowie das ACS SPiPlus MMI Application Studio verwenden. Beachten Sie auch die **Kapitel 14 "Anhang G: Über das Error Handling mit syncAXIS control"**, Seite 59 und Handbuch **"syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**, Kapitel 2.7.1 **"Über die Puffer der syncAXIS control-Instanzen"**, Seite 42.
- Führen Sie eine Sichtprüfung des Markierungsergebnisses unter Vergrößerung (z. B. mit Auflichtmikroskop) durch. Ziel ist es, dass alle Kreuze und Kreise möglichst perfekt zur Deckung kommen:
 - Ist dies visuell der Fall, empfiehlt sich zusätzlich eine quantitative Auswertung mittels einer Koordinatenmessmaschine. Dann können quantitative Aussagen über die Scan-Kopf-Arbeitsfeld-Kalibrierung gemacht werden. Im Regelfall ist die statische Positioniergenauigkeit des Verfahrtschs höher als die des Scan-Kopfs.
 - Ist dies nicht der Fall, ist mindestens eine der beiden Kalibrierungen nicht ideal.
 - Die Schritte in **Kapitel 7.4 "Verifikation – Synchronisation aller System-Komponenten prüfen"**, Seite 41 und **Kapitel 7.5 "Verifikation – Genauigkeit der kombinierten Bewegung bestimmen"**, Seite 43 sollten in diesem Zustand nicht durchgeführt werden. Finden Sie die unpräzise Kalibrierung und führen Sie die Kalibrierungs-Schritte erneut durch.

7.4 Verifikation – Synchronisation aller System- Komponenten prüfen

Vorabgehend wurden folgenden Kriterien bearbeitet:

- Die statische Kalibrierung des Scan-Kopfs
(= Scan-Kopf-Arbeitsfeld-Kalibrierung)
- Die statische Kalibrierung des Verfahrtschis
(= **Error-Mapping**)

In diesem Schritt **7.4** soll verifiziert werden, ob die Bewegungen von Scan-Kopf und Verfahrtsch synchron sind.

SCANLAB empfiehlt, diese Verifikation wenigstens 1 × auszuführen. Dann sind Abweichungen aufgrund fehlerhafter Synchronisation der Bewegungen ausgeschlossen.

Scan-Kopf und Verfahrtsch können einfach über eine Verzögerung des Soll-Profiles synchronisiert werden, weil:

- Sie haben jeweils keinen Schleppverzug
- Bei beiden ist jeweils das Soll-Profil und Ist-Profil *konstant* zeitlich versetzt

Die Herangehensweise illustriert **Abbildung 12, Seite 41**: Als Markiermuster werden 42 parallele Linien verwendet, von denen je 2 auf einer Geraden liegen. Die Linien verlaufen orthogonal zur Anordnungsrichtung. Die ersten 21 Linien sind in einer Richtung (hier "positive" genannt) angeordnet, die zweiten 21 Linien in die entgegengesetzte (hier "negative" genannt). Dieses Markiermuster ist in **CHECK_SYSTEMDELAYS** enthalten.

Die Verfahrtsch-Bewegung ist in positiver Richtung für die erste Reihe und in negativer Richtung für die zweite⁽¹⁾.

Um eine hohe Geschwindigkeit zu erreichen, soll der Verfahrtsch auf die höchst mögliche Geschwindigkeit beschleunigt werden. Zu diesem Zweck baut syncAXIS control Sprünge und Wartepunkte ein, die zur Beschleunigung und zum Abbremsen des Verfahrtschis dienen.

Wenn Scan-Kopf und Verfahrtsch zeitlich nicht optimal synchronisiert sind, dann werden die Linien in den beiden Reihen nicht perfekt auf je einer Geraden liegend markiert. Stattdessen werden sie einen gewissen Versatz zueinander aufweisen. Dies ist dann der Fall, wenn die Verzögerung der Ausführung von einer der beiden Komponenten nicht der geplanten gleicht⁽²⁾. Da sich der Verfahrtsch orthogonal zu den markierten Linien mit konstanter Geschwindigkeit bewegt, ist lediglich ein Versatz zu erwarten. Die Linien bleiben jedoch weiterhin parallel.

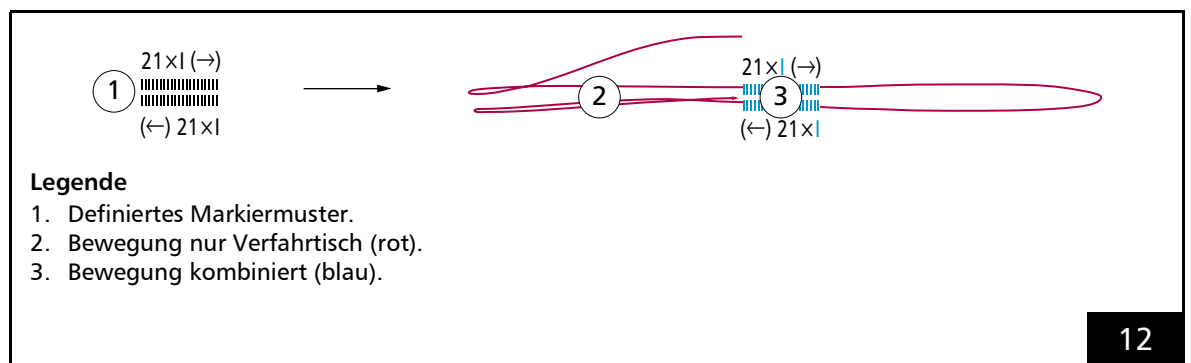
Um sämtliche Einflüsse aller Achsen auszuschließen, wird das oben beschriebene Muster zudem in allen 4 Raumrichtungen markiert.

Die Markierungsergebnisse werden anschließend einer Sichtprüfung unterzogen. Kriterien:

- Je Muster sollen die 21 Linienpaare jeweils auf einer Geraden liegen.

(1) Gemäß Tiefpassfilter, der für die Bewegungsaufteilung verantwortlich ist.

(2) Praktisch sollte dies nicht der Fall sein. Dennoch sollte eine Verifizierung dieser Annahme durchgeführt werden.



Erläuterungen siehe **Kapitel 7.4 "Verifikation – Synchronisation aller System-Komponenten prüfen", Seite 41**.

Ziele

- Die zeitliche Synchronisation von Scan-System und Verfahrtschicht ist verifiziert.

Materialien

- Markiersubstrate (inkl. Reservevorrat)

Voraussetzungen

- Das Markiermuster muss zunächst (wie immer vor einer allerersten Ausführung) simuliert werden, um sicherzustellen, dass keine Grenzwertüberschreitungen in den Ansteuerwerten enthalten sind.
- Alle Sub-Systeme (z.B. Laser) sind betriebsbereit.
- Ein Markiersubstrat ist eingelegt.

Vorgehen

- ⚠️ Warnung!** Verletzungsgefahr durch Laserstrahlung! Lasersicherheitsbestimmungen einhalten!
 - ⚠️ Warnung!** Verletzungsgefahr durch Verfahrtschichtbewegung! Keine Personen im Gefahrenbereich!
 - ⚠️ Vorsicht!** Gefahr der Sachbeschädigung durch Verfahrtschichtbewegung! Keine Fremdgegenstände im Gefahrenbereich!
- Starten Sie `Installation_Project.exe` und Initialisieren Sie eine **syncAXIS control-Instanz** (`INIT`) wie in **Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren"**, Seite 26 beschrieben.
 - Starten Sie `CHECK_SYSTEMDELAYS` wie folgt:
 - '10' – Eingabetaste – dann die höchste erreichbare Geschwindigkeit des Verfahrtschichts – Eingabetaste – dann die maximal mögliche Bewegungsdistanz des Verfahrtschichts – Eingabetaste – y – Eingabetaste.
 - Der **Job** wird für den Betriebsmodus "ScannerAndStage" mitsamt **Trajektorien**-Planung und Bewegungsaufteilung berechnet (Achten Sie auf einen realistischen `FilterBandwidth`-Wert; empfohlen werden 1...2 Hz; siehe in Ihrer initialen `syncAXISConfig.xml` von SCANLAB).
 - Der erfolgreiche Abschluss der Vorgänge wird durch "Finished with return value 0" angezeigt.

(3) Beenden Sie `Installation_Project.exe`

(`DELETE_INSTANCE > EXIT`) wie beschrieben in **Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren"**, Seite 26.

(4) Fahren Sie mit nachfolgender Prozedur "Evaluieren" fort.

Evaluieren

- Sollte die Markierung nicht vollständig durchgeführt worden sein, prüfen Sie auf Fehlermeldungen. Dafür können Sie die Log-Datei (siehe `<cfg:LogfilePath>`) sowie das ACS SPiPlus MMI Application Studio verwenden. Beachten Sie auch die **Kapitel 14 "Anhang G: Über das Error Handling mit syncAXIS control"**, Seite 59 und **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**, Kapitel 2.7.1 "Über die Puffer der syncAXIS control-Instanzen", Seite 42.
- Führen Sie eine Sichtprüfung des Markierungsergebnisses unter Vergrößerung (z. B. mit Auflichtmikroskop) durch. Je Muster sollen die 21 Linienpaare jeweils auf einer Geraden liegen:
 - Ist dies der Fall, können Sie mit dem nächsten Schritt fortfahren.
 - Ist dies nicht der Fall, weist Ihre Maschine ein unerwartetes Verhalten auf. Ihre Markierungsergebnisse werden in kombinierter Bewegung nicht ideal sein. Konsultieren Sie SCANLAB in diesem Fall.

7.5 Verifikation – Genauigkeit der kombinierten Bewegung bestimmen

Vorabgehend wurden folgenden Kriterien bearbeitet:

- Die statische Kalibrierung des Scan-Kopfs (= Scan-Kopf-Arbeitsfeld-Kalibrierung)
- Die statische Kalibrierung des Verfahrtisches (= **Error-Mapping**)
- Die Anpassung der Laser-Delays
- Verifizierung der zeitlichen Synchronisation von Scan-Kopf und Verfahrtisch

In diesem Schritt **7.5** kann nun die Genauigkeit der kombinierten Bewegung bestimmt werden.

Dafür eignet sich z.B. ein gleichförmiges Raster aus Markierobjekten (im Folgenden "Rasterobjekte" genannt). Um die zusätzlichen Messfehler gering zu halten, wählen Sie das Raster so, dass bei einer Messung die höchste Genauigkeit erzielt wird.

Dazu wird **COMBINED_ACCURACY** ausgeführt.

Als Rasterobjekte bietet **COMBINED_ACCURACY** an: Kreis, 2 konzentrische Kreise, Kreuz, Punkt (Spirale von Innen nach Außen, umrandet von einem Kreis).

Wahlweise kann in **COMBINED_ACCURACY** ausgewählt werden, dass eine verlängerte Umkehrbewegung für den Verfahrtisch nach jeder Raster-Linie ausgeführt werden soll (siehe Schritt **2** und links unten in **Abbildung 13, Seite 45**). Dann ist eine schnelle Bewegung des Verfahrtisches während der Markierung möglich und das Arbeitsfeld kann großflächig ausgenutzt werden.

Ohne diese verlängerte Umkehrbewegung (siehe Schritt **2** und rechts unten in **Abbildung 13, Seite 45**) bremst der Verfahrtisch (aufgrund des Tiefpassfilter-Designs der Bewegungsaufteilung) bereits vor Vollendung der Raster-Linie ab und der Scan-Kopf kann eine große Auslenkung erfahren (die gegebenenfalls sein Arbeitsfeld überschreiten kann).

SCANLAB empfiehlt, dass Sie beide Alternativen (= mit/ohne verlängerter Umkehrbewegung) ausprobieren.

Hinweise

- An diesem Punkt der Installation ist Ihr XL SCAN-Aufbau im Grunde genommen bereit zum Betrieb und optimal eingestellt. Anwender sollten nun einige Einstellungen ausprobieren und sich mit dem XL SCAN-Scan-System vertraut

machen. Führen Sie dabei jeweils eine Sichtprüfung durch und vermessen Sie die Markierungsergebnisse.

Ziele

- Die Genauigkeit des XL SCAN-Scan-Systems in kombinierter Bewegung ist ermittelt.

Materialien

- Markiersubstrate (inkl. Reservevorrat)

Voraussetzungen

- Das Markiermuster muss zunächst (wie immer vor einer allerersten Ausführung) simuliert werden, um sicherzustellen, dass keine Grenzwertüberschreitungen in den Ansteuerwerten enthalten sind.
- Alle Sub-Systeme (z.B. Laser) sind betriebsbereit.
- Ein Markiersubstrat ist eingelegt.

Vorgehen

- **⚠ Warnung!** Verletzungsgefahr durch Laserstrahlung! Lasersicherheitsbestimmungen einhalten!
 - **⚠ Warnung!** Verletzungsgefahr durch Verfahrtischbewegung! Keine Personen im Gefahrenbereich!
 - **⚠ Vorsicht!** Gefahr der Sachbeschädigung durch Verfahrtischbewegung! Keine Fremdgegenstände im Gefahrenbereich!
- (1) Starten Sie **Installation_Project.exe** und Initialisieren Sie eine **syncAXIS control-Instanz (INIT)** wie in **Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren"**, Seite 26 beschrieben.
 - (2) Starten Sie **COMBINED_ACCURACY** wie folgt:
 - **'7'** – Eingabetaste – dann die Größe des gesamten Rasters (kleiner als der Bewegungsbereich des Verfahrtisches) – Eingabetaste – dann die Anzahl an Rasterobjekten – Eingabetaste – dann die ungefähre Größe des Scan-Kopf-Arbeitsfelds, das genutzt werden soll – Eingabetaste – dann die Art der Rasterobjekte (1...4) – Eingabetaste – ob eine verlängerte Umkehrbewegung für die Verfahrtischbewegung eingebaut werden soll (y oder n).
Bei "y": Die ungefähr erreichte höchste Verfahrtischgeschwindigkeit entlang der Ras-

ter-Linie (Wert muss mittels Simulation gefunden werden, die aus Sicherheitsgründen ohnehin vor jeder Markierung stattfinden muss) – Eingabetaste – FilterBandwidth-Wert aus Ihrer `syncAXISConfig.xml` – Eingabetaste – Skalierungsfaktor der Abbremsbewegung / Beschleunigungsbewegung (Startwert z. B. 0.5) – Eingabetaste – y – Eingabetaste. Bei "n": y – Eingabetaste. Für "y" und "n" gilt: Der **Job** wird für den Betriebsmodus "ScannerAndStage" mitsamt **Trajektorien**-Planung und Bewegungsaufteilung berechnet (Achten Sie auf einen realistischen FilterBandwidth-Wert; empfohlen werden 1...2 Hz; siehe in Ihrer initialen `syncAXISConfig.xml` von SCANLAB). Der erfolgreiche Abschluss der Vorgänge wird durch "Finished with return value 0" angezeigt.

- (3) Beenden Sie `Installation_Project.exe` (`DELETE_INSTANCE > EXIT`) wie beschrieben in **Kapitel 6.1 "Installation_Project.exe starten und syncAXIS control-Instanz im Hardwaremodus initialisieren", Seite 26.**
- (4) Fahren Sie mit nachfolgender Prozedur **"Evaluieren"** fort.

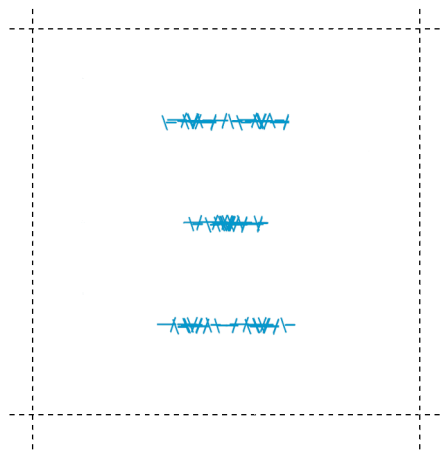
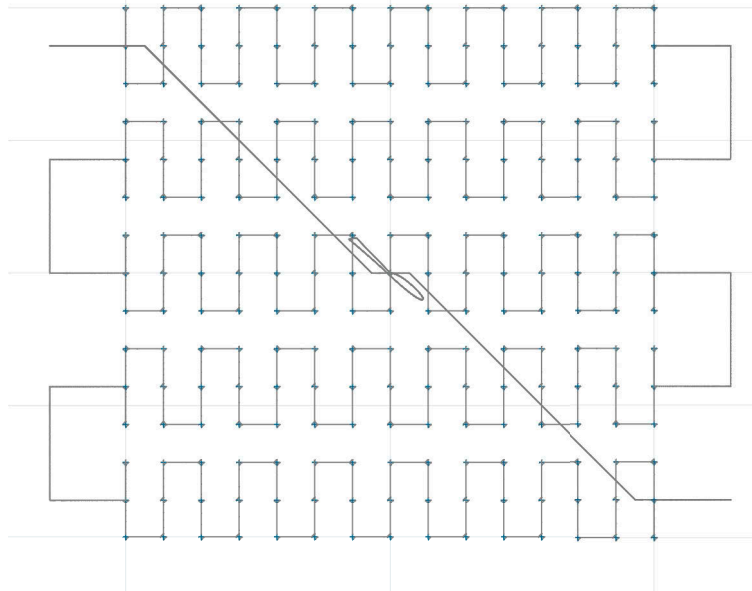
Evaluieren

- (1) Sollte die Markierung nicht vollständig durchgeführt worden sein, prüfen Sie auf Fehlermeldungen. Dafür können Sie die Log-Datei (siehe `<cfg:LogfilePath>`) sowie das ACS SPiiPlus MMI Application Studio verwenden. Beachten Sie auch die **Kapitel 14 "Anhang G: Über das Error Handling mit syncAXIS control", Seite 59** und **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung", Kapitel 2.7.1 "Über die Puffer der syncAXIS control-Instanzen", Seite 42.**
- (2) Führen Sie eine Sichtprüfung des Markierungsergebnisses unter Vergrößerung (z.B. mit Auflichtmikroskop) durch, sowie eine Vermessung der Rasterobjekte.

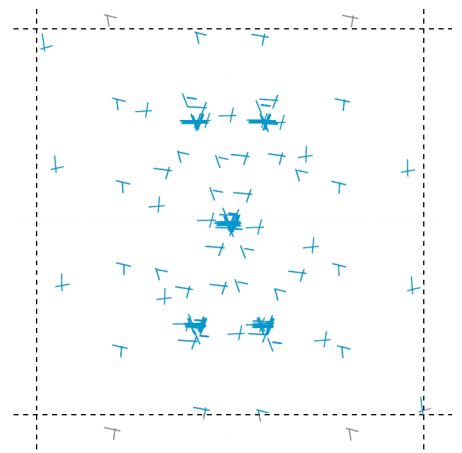
Interpretation

- Es ist zu erwarten, dass der kombinierte dynamische Gesamtfehler nicht deutlich über der Summe der jeweiligen statischen Fehler liegt, vorausgesetzt:
 - Alle vorausgehenden Tests und Verifikationen waren positiv
 - Die Simulation zeigte, dass sich alle Ansteuerwerte innerhalb der Systemgrenzen befinden
 - Die Position Error Toleranzen in der ACS-Software sind realistisch eingestellt

Gesamt-Markierung (Beispiel)



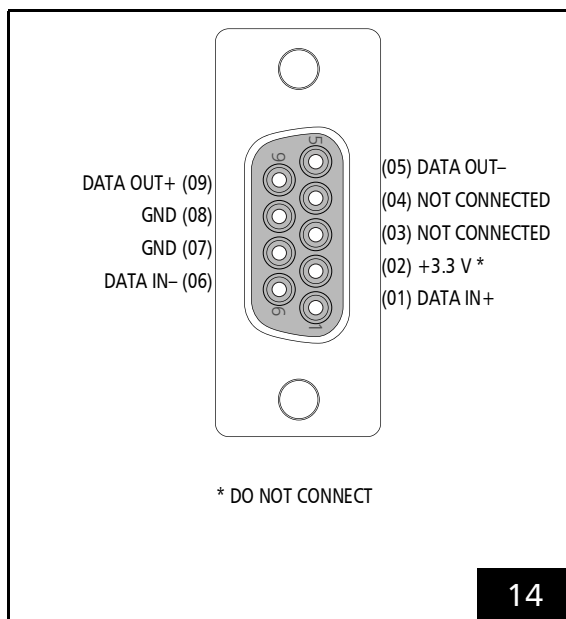
Nur der Scan-Kopf-Anteil, MIT verlängerter Umkehrbewegung des Verfahrtischs (siehe Schritt 2, Seite 43)



Nur der Scan-Kopf-Anteil, OHNE verlängerte Umkehrbewegung des Verfahrtischs (siehe Schritt 2, Seite 43)

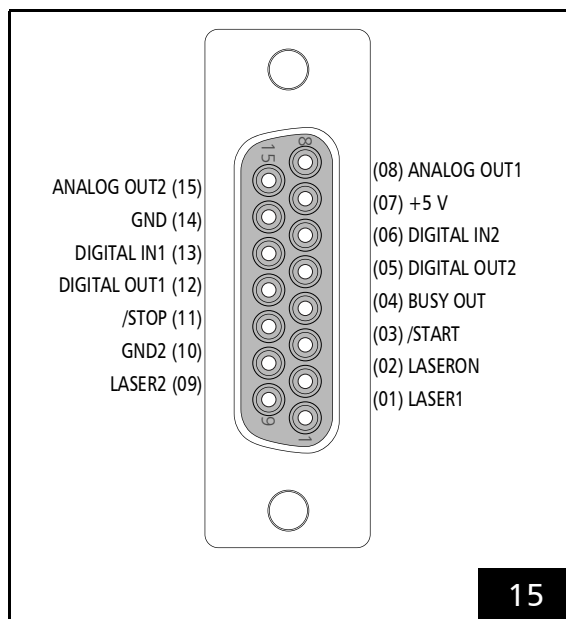
8 Anhang A: **RTC6 PCI-Express-Karte** – Steckverbinder

8.1 SCANHEAD



RTC6 PCI-Express-Karte: SCANHEAD Steckverbinder.
Sub-D 09-polig, female. Pinbelegung.

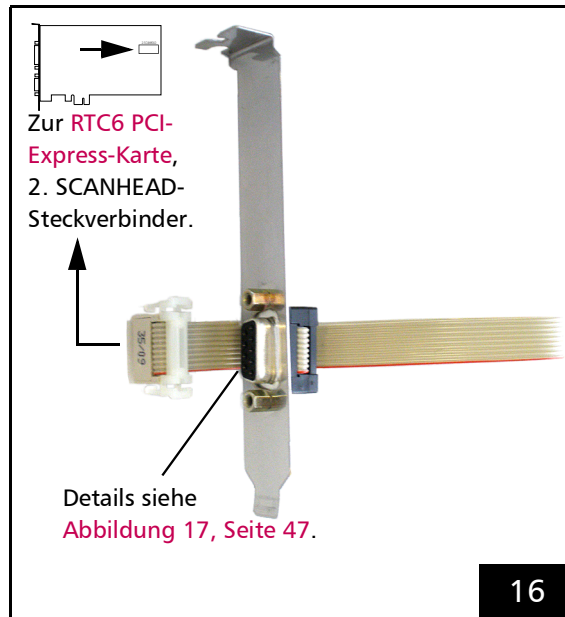
8.2 LASER



RTC6 PCI-Express-Karte: LASER Steckverbinder.
Sub-D 15-polig, female. Pinbelegung.

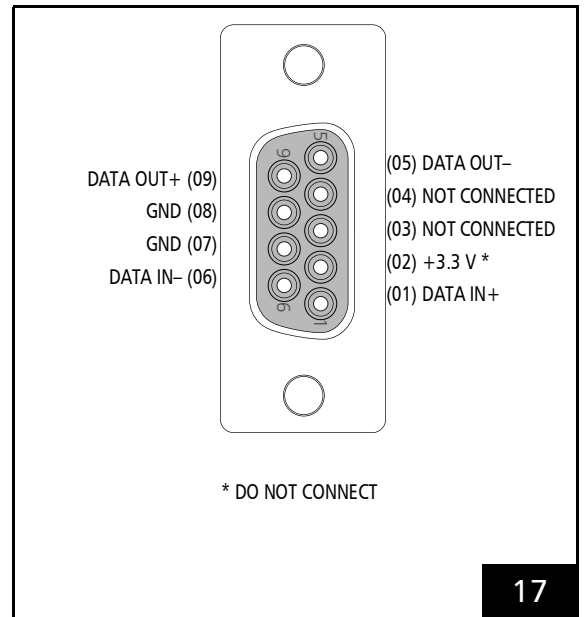
9 Anhang B: SSHC-Slotblech (#115132)

9.1 Ansicht (nicht eingebaut)



SSH-Slotblech: Ansicht (nicht eingebaut).

9.2 Steckverbinder



SSH-Slotblech: Steckverbinder (führt die Signale des 2. SCANHEAD-Anschlusses der RTC6 nach außen). Sub-D 09-polig, female. Pinbelegung.

10 Anhang C: syncAXIS control-Softwarepaket – Entpackt

Beispiel: syncAXIS-1.8.zip.

Readme.txt	Liesmich-Datei (nur englisch).
ReleaseNotes.txt	Release Notes-Datei (nur englisch).
syncAXIS_control_License_Agreement.pdf	Lizenzvertrag (englisch).
syncAXIS_control_Lizenzvertrag.pdf	Lizenzvertrag (deutsch).
syncAXIS_V1.8.0_API_de-DE.pdf	API und XML Referenz (deutsch).
syncAXIS_V1.8.0_API_en-US.pdf	API und XML Referenz (englisch).
syncAXIS_V1.8.0_Installation_de-DE.pdf	Installationshandbuch (deutsch). Dieses Dokument.
syncAXIS_V1.8.0_Installation_en-US.pdf	Installationshandbuch (englisch).
\---cmake	Verzeichnis mit den Dateien für CMake, siehe Seite 14 .
syncAXISConfig.cmake	syncAXIS cmake Target.
syncAXISConfigVersion.cmake	syncAXIS Version für cmake Target.
\---Demo	Verzeichnis mit den "Installation_Project"-Sourcen.
	Vorbereitet für CMake, siehe Seite 14 .
CMakeLists.txt	Hauptdatei für CMake.
GenerateCMakeTestProject.bat	Batch-Datei. Zum Erzeugen der MS Solution via CMake.
ResetBuild.bat	Batch-Datei. Löscht, sofern vorhanden, das Verzeichnis Build_syncAXIS.
syncAXIS.cmake	Datei für CMake.
\---Configuration_Files	Leeres Verzeichnis. Zweck: siehe Abbildung 3, Seite 21 .
\---Installation_Project	
CMakeLists.txt	Datei für CMake.
IncludePath.h.in	Datei für CMake.
\---include	
ConfigFileSearch.h	Datei für CMake.
DemoFunctions.h	Datei für CMake.
InputTypes.h	Datei für CMake.
Jobs.h	Datei für CMake.
MainLoop.h	Datei für CMake.
MarkingPatterns.h	Datei für CMake.
OperationTypesProcedures.h	Datei für CMake.
syncAXISControl.h	Datei für CMake.
\---source	
ConfigFileSearch.cpp	Datei für CMake.
DemoFunctions.cpp	Datei für CMake.
InputTypes.cpp	Datei für CMake.
Jobs.cpp	Datei für CMake.
Main.cpp	Datei für CMake.



		MainLoop.cpp	Datei für CMake.
		MarkingPatterns.cpp	Datei für CMake.
		OperationTypesProcedures.cpp	Datei für CMake.
		syncAXISControl.cpp	Datei für CMake.
		---Licences	Verzeichnis mit Lizenztexten.
		(mehrere Dateien)	
		---RTC6	Verzeichnis mit Dateien und Tools für die RTC6.
		RTC6_Doc.Rev.1.0.*de-DE.txt	RTC6-Handbuch (deutsch).
		RTC6_Doc.Rev.1.0.*en-US.txt	RTC6-Handbuch (englisch).
		---Driver	Verzeichnis mit dem RTC6-Kartentreiber für Windows.
		(mehrere Dateien in mehreren Verzeichnissen)	
		---ProgramFiles	Verzeichnis mit den RTC6-Dateien.
		RTC6BIOSETH_35.out	BIOS für die RTC6 Ethernet-Karte.
		RTC6BIOSOUT_23.out	BIOS für die RTC6 PCI-Express-Karte.
		RTC6Dat.dat	Binäre Hilfsdatei für die RTC6.
		RTC6Eth.out	Programmdatei für den RTC6 Ethernet-Karten-DSP.
		RTC6Out.out	Programmdatei für den RTC6 PCI-Express-Karten-DSP.
		RTC6RBF.rbf	Firmwaredatei für das RTC6 PCI-Express-Karten-FPGA.
		---Tools	
		---iSCANcfg	Verzeichnis mit dem Tool iSCANcfg.exe. Diagnose- und Konfigurationsprogramm für iDRIVE-Scan-Systeme/ RTC-Karten inkl. Handbuch.
		(mehrere Dateien)	
		---RTC6conf	Verzeichnis mit dem Tool RTC6conf.exe. Diagnose- und Konfigurationsprogramm für RTC6-Karten inkl. Beschreibung.
		(mehrere Dateien)	
		---SleepMode	Verzeichnis mit einem Script zum Deaktivieren aller Windows Sleep- und Hibernatemodi inkl. Beschreibung.
		(mehrere Dateien)	



\---syncAXIS_control	Verzeichnis mit Dateien und Tools für syncAXIS control.
\---bin	
\---dll	
RTC6DLL.dll	32-Bit-RTC6-DLL-Datei. Von syncAXIS-DLL benötigt.
syncAXIS.dll	Die 32-Bit-syncAXIS-DLL.
xerces-c_3_2.dll	DLL-Datei. Von syncAXIS-DLL benötigt.
\---lib	
syncAXIS.lib	Visual C++ Import-Bibliothek.
\---Wrapper	
\---C#	Verzeichnis mit Hilfsdateien für die Softwareentwicklung unter C#.
(mehrere Dateien)	
\---bin64	
\---dll	
RTC6DLLx64.dll	64-Bit-RTC6-DLL-Datei. Von syncAXIS-DLL benötigt.
syncAXIS.dll	Die 64-Bit-syncAXIS-DLL.
xerces-c_3_2.dll	DLL-Datei. Von syncAXIS-DLL benötigt.
\---lib	
syncAXIS.lib	Visual C++ Import-Bibliothek.
\---Wrapper	
\---C#	Verzeichnis mit Hilfsdateien für die Softwareentwicklung unter C#.
(mehrere Dateien)	
\---Configuration	
syncAXIS_1_8.xsd	XML-Schemadatei für die syncAXISConfig.xml .
syncAXISConfig.Template.xml	Vorlage für syncAXISConfig.xml -Dateien. Nicht im Real-Betrieb verwenden!
syncAXISConfig_MultiHead.Template.xml	Vorlage für Multi-Head- syncAXISConfig.xml -Dateien. Nicht im Real-Betrieb verwenden!
\---include	
syncAXIS.h	Header-Datei (u.a. zum impliziten Linken zur syncAXIS.dll).
syncAXISDefinitions.h	Header-Datei.



\---Tools	
\---syncAXIS_Configurator	Verzeichnis mit dem Tool syncAXIS Configurator
	inkl. Handbuch.
(mehrere Dateien)	
\---syncAXIS_Installation	Verzeichnis mit dem Tool Installation_Project.exe
	Das Handbuch ist im Root-Verzeichnis.
(mehrere Dateien)	
\---syncAXIS_MasterSlaveSynchronizer	Verzeichnis mit dem Tool nur für Sondersysteme
	inkl. Handbuch.
(mehrere Dateien)	
\---syncAXIS_Viewer	Verzeichnis mit dem Tool syncAXIS Viewer
	inkl. Handbuch.
(mehrere Dateien)	

11 Anhang D: Über das "Installation_Project"

Im syncAXIS control-Software-Paket wird auch ein Beispiel-Code für C++ bereitgestellt.

Er ist zu finden im Ordner:

```

\---Demo
|  \---Installation_Project

```

Das Beispielprojekt "Installation_Project" soll dazu dienen, die einfache Handhabung der syncAXIS-DLL zu demonstrieren und einfache Programmierkonzepte zu veranschaulichen, z. B. das asynchrone Listen laden und ausführen (siehe Kapitel List-Handling im **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**).

Im Folgenden werden die verschiedenen Aktionen 0...13 des "Installation_Project" aufgelistet und beschrieben.

Die **Jobs** werden außerdem in mehreren Kapiteln dieses Handbuchs verwendet, um eine einfache Installation und Konfiguration zu ermöglichen, sowie die Korrektheit aller Einstellungen zu verifizieren.

Eingabe	Kurzbezeichnung	Beschreibung	Demonstrations-Zweck
0	EXIT	Verlassen dieses Anwenderprogramms. Dabei wird <code>slsc_cfg_delete</code> aufgerufen. Benutzt in Kapitel 5.2 "Installation_Project.exe starten und syncAXIS control-Instanz im Simulationsmodus initialisieren", Seite 22 und anderen.	<code>slsc_cfg_delete</code> muss beim Beenden des Anwenderprogramms (egal ob absichtlich oder unabsichtlich) immer aufgerufen werden, um die korrekte Abbaureihenfolge des Programms zu gewährleisten.
1	INIT	Initialisierung der syncAXIS control-Instanz basierend auf den Einstellungen der angegebenen(*) <code>syncAXISConfig.xml</code> . (*) INIT bietet Pfade zur Auswahl an: das Arbeitsverzeichnis, in dem die <code>Installation_Project.exe</code> gestartet wurde und <pre> \---Demo \---Configuration_Files. </pre> Benutzt in Kapitel 5.2 "Installation_Project.exe starten und syncAXIS control-Instanz im Simulationsmodus initialisieren", Seite 22 und anderen.	Um eine syncAXIS control-Instanz zu initialisieren ist nur eine syncAXIS control-Funktion notwendig. Alle Einstellungen sind in <code>syncAXISConfig.xml</code> "sicher" gespeichert und werden von dieser Funktion eingelesen. Im Hardwaremodus bedeutet dies: Aufbau der syncAXIS control-Instanz , Initialisierung der RTC6-Karte, Synchronisierung mit den ACS -Komponenten, Setzen aller definierten Parameter auf der RTC6-Karte und im Anwenderprogramm, Ausführung der Laser-Initialisierungsroutine (falls definiert), etc.
2	CHECK_LASER_FIRING	Konfiguration des Lasers. Überprüfen der korrekten Laser-Kommunikation. Nützliches Tool zur Optimierung des Fokus, der Laserparameter, etc. Benutzt in Kapitel 6.2 "Prüfen, ob Laserstrahlung ausgegeben wird", Seite 28.	Demonstration des Modus "Manuelle Positionierung". Siehe Kapitel 2.12 im Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung" .

Eingabe (Forts.)	Kurzbezeichnung (Forts.)	Beschreibung (Forts.)	Demonstrations-Zweck (Forts.)
3	REFERENCE_RUN	<p>Überprüfung der Kommunikation der Systemkomponenten, sprich Verfahrtsch, Scan-Device und Laser.</p> <p>Benutzt in Kapitel 6.3 "Überprüfen der korrekten Kommunikation mit Scan-Kopf und Verfahrtsch", Seite 29.</p>	Demonstration des Modus "Manuelle Positionierung". Siehe Kapitel 2.12 im Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung".
4	TEST_MARKING	<p>Erste Testmarkierung zur Überprüfung der Funktionalität aller Komponenten inklusive der syncAXIS control-Trajektorien-Planung und der ACS-Kommunikation über den ACS SL2-100-zu-EtherCAT-Konverter. Erster Eindruck von Eckenabrundung dank "Blending" (Übergänge) oder Laser-Schaltzeitpunkte in den Ecken des Markierungsergebnisses.</p> <p>Benutzt in Kapitel 5.3 "TEST_MARKING" im Betriebsmodus "ScannerAndStage" simulieren", Seite 24 und Kapitel 6.4 "Den (vorgängig in 5.3) simulierten "TEST_MARKING" markieren", Seite 30.</p>	Ist das einfachste Test-Programm im gesamten "Installation_Project". Zeigt die Job-Struktur in der Funktion writeJob() auf, sowie paralleles Listen laden im asynchronen Thread. Die Funktion startJob() zeigt außerdem das Konzept der Abfrage des Ausführungsstatus.
5	SCANNER_CALI	<p>Markieren eines Gitters im Betriebsmodus ScannerOnly zur Kalibrierung des Scan-Device. Die Punkte können vermessen und an correXion pro übergeben werden, um eine optimierte Korrekturdatei zu erstellen. Idealerweise wird zur Vermessung die mechanische x-Achse als Referenz verwendet. Dazu verfährt dieser Job die x-Achse um einige Millimeter und markiert dann einen Referenzpunkt im Arbeitsfeld-Zentrum.</p> <p>Benutzt in Kapitel 7.2 "Optimierung – Das Scan-Kopf-Arbeitsfeld kalibrieren", Seite 36.</p>	Verwendung verschiedener Betriebsmodi (ScannerOnly). Einfacher Test-Programm mit einfacher Struktur bestehend aus mehreren Jobs.
6	STAGE_CALI	<p>Nicht in diesem Handbuch für die Installation verwendet.</p> <p>Markieren eines großen Gitters im Betriebsmodus StageOnly. Dieses Markiermuster kann mit einem Calibration-Plate (Glasmaster) abgeglichen werden, um das Error-Mapping der mechanischen Achsen zu erstellen.</p>	Verwendung verschiedener Betriebsmodi (StageOnly). Einfaches Test-Programm mit einfacher Struktur bestehend aus mehreren Jobs.

Eingabe (Forts.)	Kurzbezeichnung (Forts.)	Beschreibung (Forts.)	Demonstrations-Zweck (Forts.)
7	COMBINED_ACCURACY	<p>Markiert ein Gitter aus wählbaren Gitterelementen im Betriebsmodus <code>ScannerAndStage</code> (kombinierte Bewegung). Kann zum Messen und Auswerten verwendet werden. Um höhere Verfahrtisch-Geschwindigkeiten zu erreichen, kann ein Bewegungssegment zur Optimierung der Verfahrtisch-Wege integriert werden.</p> <p>Benutzt in Kapitel 7.4 "Verifikation – Synchronisation aller System-Komponenten prüfen", Seite 41 and Kapitel 7.5 "Verifikation – Genauigkeit der kombinierten Bewegung bestimmen", Seite 43.</p>	Sehr komplexer Job . Viele Gitterelemente verfügbar. Verwendung der Funktion <code>slsc_list_wait_with_laser_off</code> zur Optimierung der Verfahrtisch-Bewegung. Wenn Sie an weiteren Details interessiert sind, fragen Sie SCANLAB um Unterstützung zum Thema "Pfadoptimierung".
8	CHECK_LASERDELAYS	<p>Markiert je ein Gitter an Objekten (ähnlich "H") im Betriebsmodus <code>ScannerOnly</code> bei maximaler Geschwindigkeit zur Optimierung der Laserschaltzeitpunkte. Entlang des Gitters werden <code>LaserSwitchOffsetTime</code> und <code>LaserPreTriggerTime</code> angepasst, sodass aus dem Array an Objekten das Objekt mit den optimalen Laserschaltzeitpunkten ausgewählt werden kann. Diese Werte können anschließend verwendet werden, um diesen Test mit feineren Iterationsschritten erneut durchzuführen oder die neu ermittelten Laser-Delay-Werte direkt zu verwenden.</p> <p>Benutzt in Kapitel 7.1 "Optimierung – Laser-Delays finden, die zu Ihrer Anwendung passen", Seite 33.</p>	Funktional (eher weniger Demonstrationseffekt).
9	CHECK_CALIBRATIONS	<p>Markiert je ein Gitter im Betriebsmodus <code>ScannerOnly</code> (Kreise) und im Betriebsmodus <code>StageOnly</code> (Kreuze). Die Gitter können verglichen werden, um die Qualität der Kalibrierung zu prüfen.</p> <p>Benutzt in Kapitel 7.3 "Verifikation – Genauigkeit der statischen Kalibrierungen von ScanKopf und ACS-Achsen prüfen", Seite 39.</p>	Funktional (eher weniger Demonstrationseffekt).

Eingabe (Forts.)	Kurzbezeichnung (Forts.)	Beschreibung (Forts.)	Demonstrations-Zweck (Forts.)
10	CHECK_SYSTEMDELAYS	<p>Markiert Reihen an Linien orthogonal zur mechanischen Bewegung. Die Linien werden in positiver und negativer Richtung ausgeführt, und dann für alle 4 Raumrichtungen wiederholt. Ziel ist zu überprüfen, ob die Linien beider mechanischer Bewegungsrichtungen kollinear sind, oder ein Offset in Richtung der mechanischen Bewegung zu sehen ist. Für den Fall, dass die Linien nicht kollinear sind (Offset in Richtung der mechanischen Bewegung), dann wird die Verfahrtsch-Bewegung nicht perfekt synchron zur Scan-Device-Bewegung ausgeführt. Wenden Sie sich in einem solchen Fall an SCANLAB. Ein Pfeil gibt die mechanische Bewegungsrichtung an.</p> <p>Benutzt in Kapitel 7.4 "Verifikation – Synchronisation aller System-Komponenten prüfen", Seite 41.</p>	Funktional (eher weniger Demonstrations-effekt).
11	CHANGE_SETTINGS	<p>Nicht in diesem Handbuch für die Installation verwendet.</p> <p>Einige Einstellungen wie Geschwindigkeiten und Betriebsarten können angepasst werden. Beachten Sie, dass die meisten Jobs sowieso eigene Einstellungen verwenden. Nützlich für den Job TEST_MARKING und zum Anwenden eines Offsets, wenn der zu markierende Bereich sich nicht in der Verfahrtsch-Mitte befindet.</p>	Verwendung von Konfigurations-Funktionen (slsc_cfg_*). Reaktion auf Anwender-Input oder Peripherie-Input.
12	DELETE_INSTANCE	<p>Abbau der syncAXIS control-Instanz. Anschließend muss INIT durchgeführt werden, um die syncAXISConfig.xml neu zu laden und eine neue syncAXIS control-Instanz aufzubauen.</p> <p>Benutzt z.B. in Kapitel 5.2 "Installation_Project.exe starten und syncAXIS control-Instanz im Simulationsmodus initialisieren", Seite 22.</p>	Verwendung der Funktion slsc_cfg_delete zum Abbauen der syncAXIS control-Instanz , z.B. zum Freigeben aller Komponenten oder zum Neuladen der syncAXISConfig.xml .
13	CHECK_FOR_ERRORS	<p>Abfrage aller Systemfehler und Ausgabe an die Konsole.</p> <p>Benutzt z.B. in Kapitel 5.2 "Installation_Project.exe starten und syncAXIS control-Instanz im Simulationsmodus initialisieren", Seite 22.</p>	Demonstriert das Abfragen aller Fehlercodes. Anders als bei der RTC6-DLL akkumuliert syncAXIS control Fehlercodes nicht!

12 Anhang E: Kabel für Datenübertragung gemäß SL2-100-Protokoll – Anforderungen

Im typischen Lieferumfang der SCANLAB-Komponenten für XL SCAN sind (mindestens zwei) Kabel für Datenübertragung gemäß SL2-100-Protokoll⁽¹⁾ enthalten:

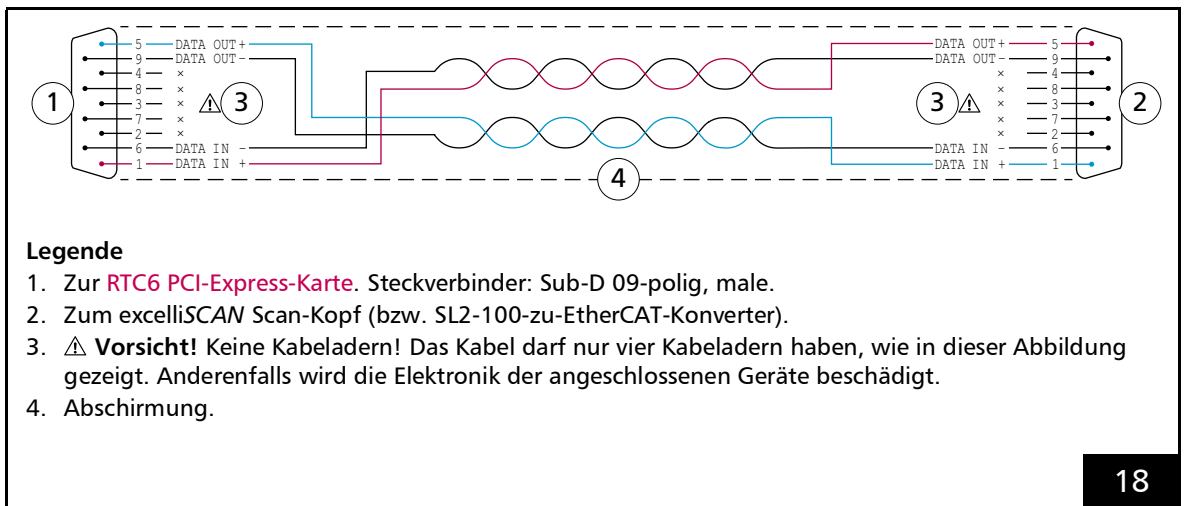
- 1 × SL2-100-Datenkabel zur Verbindung RTC6 / SL2-100-zu-EtherCAT-Konverter
- 1 × SL2-100-Datenkabel zur Verbindung RTC6 / Scan-Kopf



Vorsicht!

Im Falle von Reparatur, Austausch und auch bei Eigenbauten ist die interne Verdrahtung dieser Kabel zu beachten, siehe **Abbildung 18**, **Seite 56**. Anderenfalls wird die Elektronik der angeschlossenen Geräte beschädigt.

- (1) Bei SCANLAB sind erhältlich (nicht schleppkettentauglich):
 #115428 (Länge 5 m), #115430 (Länge 10 m),
 #115431 (Länge 15 m).

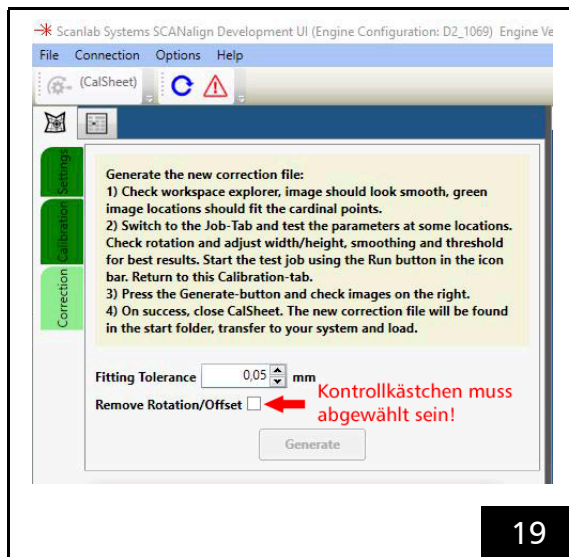


18

Kabel für Datenübertragung gemäß SL2-100-Protokoll: Anforderungen und Pinzuordnungen.

13 Anhang F: Scan-Kopf-Arbeitsfeld kalibrieren – Alternative mit der CalSheet-Software

- Dieses Kapitel beschreibt eine Alternative zum Ablauf, der in **Kapitel 7.2 "Optimierung – Das Scan-Kopf-Arbeitsfeld kalibrieren"**, Seite 36 (dort mit correXion Pro) beschrieben ist. Es gelten die gleichen **Voraussetzungen**, siehe Seite 37.
- Diese Prozedur setzt die CalSheet-Software V2.2.5 voraus⁽¹⁾. Erst ab dieser Version steht auf der Registerkarte 'Correction' das Kontrollkästchen **Remove Rotation/Offset** zur Verfügung, siehe **Abbildung 19, Seite 57**. Bevor Sie die neue (=optimierte) Korrekturdatei generieren, wählen Sie dieses Kontrollkästchen ab.



CalSheet-Software V2.2.5: Stellen Sie sicher, dass das gekennzeichnete Kontrollkästchen ausgewählt ist.

- In dieser Prozedur wird die Calibration-Plate aus dem CalSheet-Paket⁽²⁾ *nicht benutzt*. Funktional ersetzt wird die Calibration-Plate durch eine Liniengitter-Markierung (s.u.).

13.1 Definieren zweier unterschiedlicher Kalibriermuster

Die Feldkalibrierungs-Prozedur mit der CalSheet-Software V2.2.5 basiert auf zwei unterschiedlichen Kalibriermustern. Sie müssen sich über das gesamte Scan-System-Arbeitsfeld erstrecken und werden auf ein und dasselbe Markiersubstrat markiert, siehe **Abbildung 20, Seite 58**:

- 1 Raster aus Kreuzen (regelmäßige Anordnung von "×"-en; immer ungerade Zahlen, gewöhnlich 11 × 11)

Dieses Kalibriermuster wird nur mittels Scan-Kopf-Bewegungen markiert (d.h. ohne Verfahrtsch-Bewegungen).

- 1 Liniengitter (=Gitter aus durchgezogenen Linien)

Dieses Kalibriermuster wird nur mittels Verfahrtsch-Bewegungen markiert (d.h. ohne Scan-Kopf-Bewegungen).

Diese Markierung ersetzt funktional die Calibration-Plate.

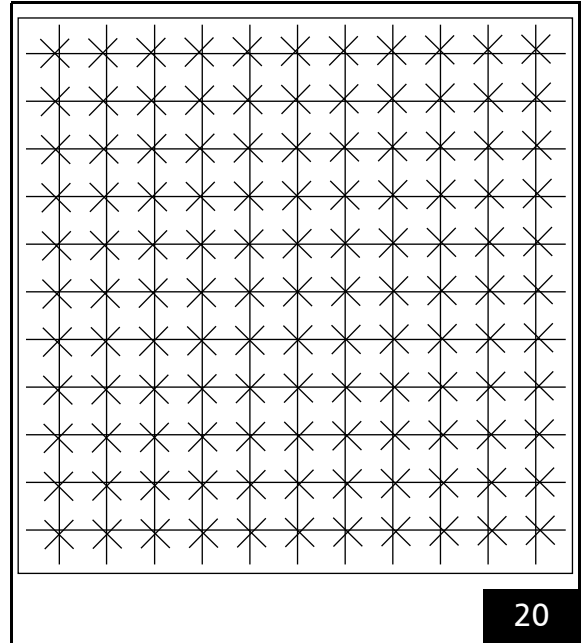
(1) Um die CalSheet-Software nutzen zu können, ist hier außerdem ein Bildscanner nötig.

(2) Im typischen Lieferumfang des CalSheet-Pakets ist eine Calibration-Plate enthalten.

13.2 Markieren der zwei Kalibriermuster

Hinweise

- Für die nachfolgende Bildauswertung ist es wesentlich, dass das Markierergebnis scharfe Kanten aufweist. Bei einem eingestellten und ausgerichteten Strahl, siehe [Voraussetzungen, Seite 37](#), sollte sich das normalerweise schon automatisch ergeben. Weiterhin können Sie das Laser-Delay oder die Geschwindigkeitseinstellungen anpassen, um bessere Ergebnisse zu erzielen.
- Benutzen Sie zum Markieren und Springen relativ niedrige Geschwindigkeiten (z. B. 0,25 m/s) um so gut wie möglich dynamische Folgefehler zu vermeiden. Hier ist das Hauptziel die statische Feldkalibrierung, deswegen ist eine hohe Genauigkeit (und keine hohe Geschwindigkeit) nötig.



Markiersubstrat mit Markierergebnis nach Schritt 3 (= mit beiden Kalibriermustern) für die Auswertung durch die CalSheet-Software. Typischerweise kommen einige x-Mittelpunkte und die Gitterlinienkreuzungen nicht zur Deckung.

Markieren der beiden Kalibriermuster – Beispielhafter Ablauf

- (1) Markieren Sie das Kreuzmuster (regelmäßige Anordnung von "x"-en). Verwenden Sie nur den Scan-Kopf zum Markieren, der Verfahrtschicht bleibt an Position (0,0).
- (2) Bewegen Sie das Markiersubstrat nicht.
- (3) Markieren Sie (auf das gleiche Markiersubstrat) das Gitter aus durchgezogenen Linien. Bewegen Sie zum Markieren nur den Verfahrtschicht. Die Galvanometerscanner des Scan-Kopfs bleiben fest bei (0,0).
Das Markierergebnis liegt nun vor wie in [Abbildung 20, Seite 58](#) gezeigt.

13.3 Generieren der optimierten ct5-Datei mit der CalSheet-Software

- (1) Scannen Sie mit einem Büros Scanner das Markiersubstrat aus Schritt 3 (siehe [Abbildung 20, Seite 58](#)) ein.
- (2) Laden Sie das eingescannte Bild in die CalSheet-Software V2.2.5 und erzeugen die neue (=optimierte) ct5-Datei, wie im CalSheet-Handbuch beschrieben.
Beachten Sie dabei [Abbildung 19, Seite 57](#)!

14 Anhang G: Über das Error Handling mit syncAXIS control



Warnung!

Die Code-Abschnitte dieses Dokuments dürfen niemals ohne vorhergehende Anpassung und Simulation auf echten XL SCAN-Systemen ausgeführt werden. Anderenfalls drohen Personenschäden und Sachschäden.

Haftungsausschluss: SCANLAB übernimmt keine Haftung für Schäden oder Folgeschäden, die sich aus der Nichtbeachtung dieser Warnung ergeben. SCANLAB übernimmt keine Verantwortung für die Richtigkeit oder Funktionalität dieser Code-Abschnitte.

Bei einem komplexen System mit einer Vielzahl von Funktionalitäten müssen viele Komponenten richtig konfiguriert und einige Fallstricke bei der Programmierung des Systems vermieden werden.

Um die Fehlerbehandlung für Integratoren und Benutzer so einfach wie möglich zu gestalten, bietet syncAXIS control viele Möglichkeiten, Probleme zu identifizieren, angefangen von Rückgabewerten als "erste Ebene" bis hin zu einem Logging-Mechanismus. Im Vergleich zur RTC6-Programmierung ist der Umfang der Fehlerbehandlung bei syncAXIS control bedeutend größer.

Während bei der Verwendung der RTC6-API nur wenige Befehle einen Status-Rückgabewert wie `init_rtc6_dll` oder `load_program_file` liefern, liefert jede syncAXIS control-Funktion einen Status-Rückgabewert, wobei 0 immer eine erfolgreiche Ausführung anzeigt, siehe [Kapitel 14.1 "Rückgabewerte der syncAXIS control-Funktionen"](#), Seite 60.

Ähnlich zur RTC6-API enthält die syncAXIS control-API eine Funktion `slsc_ctrl_get_error`, siehe [Kapitel 14.2 "slsc_ctrl_get_error"](#), Seite 61.

Im Gegensatz zum RTC6-Befehl `get_error` gibt die syncAXIS control-Funktion keinen kumulierten Fehlercode zurück, sondern einen fehlerspezifischen Fehlerwert, gepaart mit einer Fehlermeldung für jeden angegebenen Fehler ("zweite Ebene"). Die Anzahl der aufgetretenen Fehler kann mit `slsc_ctrl_get_error_count` abgefragt werden.

Völlig neu im Vergleich zur RTC6-API ist der syncAXIS control Logging-Mechanismus ("dritte Ebene"), siehe [Kapitel 14.3 "Logging in syncAXIS control"](#), Seite 62. In der `syncAXISConfig.xml` konfigurierbar, können Anwender wahlweise die Ausgabe an die Anwendungskonsole und/oder in eine zusätzliche Textdatei zu aktivieren. Es gibt 3 verschiedene Loglevel.

Normalerweise treten die meisten Probleme bei der Initialisierung des XL SCAN-Systems auf. Die Funktion `slsc_cfg_initialize_from_file` liest die `syncAXISConfig.xml` und startet die Kommunikation mit allen Hardwarekomponenten wie ACS Motion Controller, ACS SL2-100-zu-EtherCAT-Konverter und RTC6-Karte(n) und auch mit Software von Drittanbietern, die im syncAXIS control-Software-Paket enthalten sind.

14.1 Rückgabewerte der syncAXIS control-Funktionen

Anders als RTC6-Befehle bietet jede syncAXIS control-Funktion einen Rückgabewert, siehe auch [Abbildung 21, Seite 60](#).

Der Rückgabewert 0 zeigt immer an, dass die syncAXIS control-Funktion erfolgreich ausgeführt wurde.

Ist der Rückgabewert ... 0, deutet der Wert

- entweder auf eine unbeabsichtigte Verwendung einer bestimmten Funktion hin, die nicht unbedingt weitere Probleme in der Zukunft verursacht (z. B. Aufruf einer syncAXIS control-Funktion in einem Modus, in dem sie nicht verwendet werden darf)
- oder auf ein schwerwiegendes Problem hin (z. B. Automatischer Stop der Master-RTC6-Karte)

Im letzteren Fall wird die **syncAXIS control-Instanz** in einen Fehlerzustand versetzt.

Die meisten der nachfolgenden Funktionen geben den Wert 1 zurück, bis die **syncAXIS control-Instanz** neu initialisiert wird (d. h. die Kommunikation zur Hardware wiederherstellen und die Planungsobjekte neu aufbauen).

Um nach spezifischeren Informationen über das aufgetretene Problem zu suchen, kann die Funktion **slsc_ctrl_get_error** verwendet werden, siehe [Kapitel 14.2 "slsc_ctrl_get_error", Seite 61](#).

OK =	0x0000,
InErrorState =	0x0001,
ErrorOccured =	0x0002,
NotAllowedWithoutInitialization =	0x0004,
NotAllowedInExecuting =	0x0008,
BufferFull =	0x0010,
NotReadyForExecution =	0x0020,
UnplausibleOrUnknownParameter =	0x0040,
JobStructureNotValid =	0x0080,
Undefined0x0100 =	0x0100,
NotAllowedInCurrentConfiguration =	0x0200,
Undefined0x0400 =	0x0400,
NotAllowedInCurrentMode =	0x0800,
InvalidPosition =	0x1000,
Timeout =	0x2000,
XmlLoadError =	0x4000,
Initialization failed =	0x8000,
Undefined0x10000 =	0x10000,
Undefined0x20000 =	0x20000,
HandshakeFailed =	0x40000,
Undefined0x80000 =	0x80000,
Undefined0x100000 =	0x100000,
Undefined0x200000 =	0x200000,
Undefined0x400000 =	0x400000,
Undefined0x800000 =	0x800000,
Undefined0x1000000 =	0x1000000,
Undefined0x2000000 =	0x2000000,
Undefined0x4000000 =	0x4000000,
Undefined0x8000000 =	0x8000000,
UnknownDevice =	0x10000000,
Undefined0x20000000 =	0x20000000,
MaxInstancesReached =	0x40000000,
InvalidOrMissingDongle =	0x80000000

21

Beispiel: Rückgabewerte in syncAXIS control. Siehe Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung", Kapitel 4 "Standard-Rückgabewerte der syncAXIS-DLL Funktionen", Seite 279.

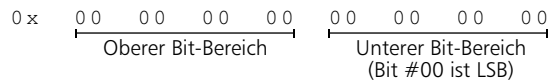
14.2 slsc_ctrl_get_error

Anders als bei RTC6-Befehlen gibt die syncAXIS control-Funktion **slsc_ctrl_get_error** keinen kumulierten Fehlercode zurück, sondern einen fehlerspezifischen Wert mit einer Fehlermeldung ("zweite Ebene").

Um alle aufgetretenen Fehler zu durchlaufen, kann **slsc_ctrl_get_error_count** verwendet werden, um die Anzahl der Fehler abzufragen.

Mit den beiden Funktionen **slsc_ctrl_get_error** und **slsc_ctrl_get_error_count** kann beispielsweise ganz einfach eine Funktion geschrieben werden, die die Fehlermeldungen aller aufgetretenen Fehler ausgibt, siehe **Abbildung 22, Seite 61**.

Der zurückgegebene Fehlercode ist ein 64-Bit-Wert mit getrennter Bedeutung für die oberen und unteren 32 Bits:



Die oberen 32 Bits definieren die Gruppe der Fehlercodes für die unteren 32 Bits.

So ist z.B.

0x00 00 00 00 nn nn nn nn

ein RTC6-Fehlercode.

Die Bedeutung aller Fehlercodes ist im **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"** beschrieben. Insbesondere finden Sie dort im **Kapitel 5 "Fehlercodes (Error Codes) bei slsc_ctrl_get_error, Log-Datei und Konsole"**, **Seite 282** weitere Hinweise zu Fehlercodes, deren unterer Bit-Bereich nicht komplett 0 ist.

```
// C++-Code-Abschnitt nur zu Schulungszwecken. Führen Sie diesen Code niemals ohne vorhergehende
// Anpassung und Simulation auf echten XL SCAN-Systemen aus!
// Beachten Sie die Sicherheitshinweise und den Haftungsausschluss auf Seite 59.

void checkForErrors(size_t SLHandle, uint32_t& ReturnValue)
{
    size_t ErrorCount = 0;
    ReturnValue |= slsc_ctrl_get_error_count(SLHandle, &ErrorCount);
    std::cout << "Error count: " << ErrorCount << std::endl;
    for (size_t Counter = 0; Counter < ErrorCount; ++Counter)
    {
        uint64_t ErrorCode = 0;
        const static size_t ErrorTextSize = 1000;
        char ErrorText[ErrorTextSize];
        ReturnValue |= slsc_ctrl_get_error(SLHandle, Counter, &ErrorCode, ErrorText, ErrorTextSize);
        std::cout << "Error with error code: " << std::hex << ErrorCode << std::dec << std::endl;
        std::cout << "Error text: " << ErrorText << std::endl;
    }
    return;
}
```

22

Einfache Funktion mit der sämtliche in einer **syncAXIS control-Instanz** vorhandenen Fehler abgefragt und in die Konsole ausgegeben werden.

14.3 Logging in syncAXIS control

Zur erweiterten Problemanalyse kann als "dritte Ebene" der syncAXIS control Logging-Mechanismus verwendet werden, siehe **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"**, Kapitel 2.8 "Über das Logging in syncAXIS control", Seite 47.

14.4 Tipps zum Error Handling mit syncAXIS control

- Prüfen Sie in Ihrem Anwenderprogramm die Rückgabewerte aller syncAXIS control-Funktionen. Falls diese $\neq 0$ sind, beenden Sie die **Job**-Ausführung.
Im Fehlerfall:
 - Fragen Sie die aufgetretenen Fehler mit **slsc_ctrl_get_error_count** und **slsc_ctrl_get_error** ab, siehe auch **Abbildung 22, Seite 61**.
 - Konsultieren Sie gegebenenfalls das **Handbuch "syncAXIS-DLL – Schnittstelle zur Anwendungsprogrammierung"** zu den aufgetretenen Fehlercodes, falls deren unterer Bit-Bereich ungleich 0 ist.
 - Lesen Sie die geloggten Meldungen in der Log-Datei sorgfältig. Diese enthalten manchmal zusätzliche Informationen zur Fehlerursache.
- Bei Problemen mit dem Verfahrtsch:
 - Prüfen Sie im ACS SPiiPlus MMI Application Studio, ob Fehler angezeigt werden.
- Gewöhnlich geben die aufgetretenen Fehlermeldungen guten Aufschluss über die Fehlerursache, so dass Sie Ihr Problem selber beheben können.
 - Falls nicht: Kontaktieren Sie den SCANLAB-Support. In diesem Fall ist es hilfreich, wenn Sie Ihr Vorgehen möglichst detailliert beschreiben und die erzeugte Log-Datei mit-senden.

15 Änderungsindex

Nachfolgend genannt sind Änderungen an diesem Handbuch aufgrund technischer Weiterentwicklung des Produkts sowie wesentliche redaktionelle Änderungen.

Änderungen nach Dokument-Revision **2.0.4 de-DE** von Dokument-Revision 2.0.3 de-DE

Name des Kapitels / Funktionstabelle	Hinweise / Änderungen
Global	Dokument Revision <ul style="list-style-type: none"> • 2.0.4 de-DE gilt für syncAXIS control-Software-Paket <ul style="list-style-type: none"> • V1.8.0
RTC6 Ethernet-Karte, Seite 10	Software-Änderung. Gelöscht: "Z.Zt. nur für 1-Kopf-Aufbau ".
Kapitel 10 "Anhang C: syncAXIS control-Softwarepaket – Entpackt", Seite 48	Software-Änderung. Datei-Namen.



Notizen