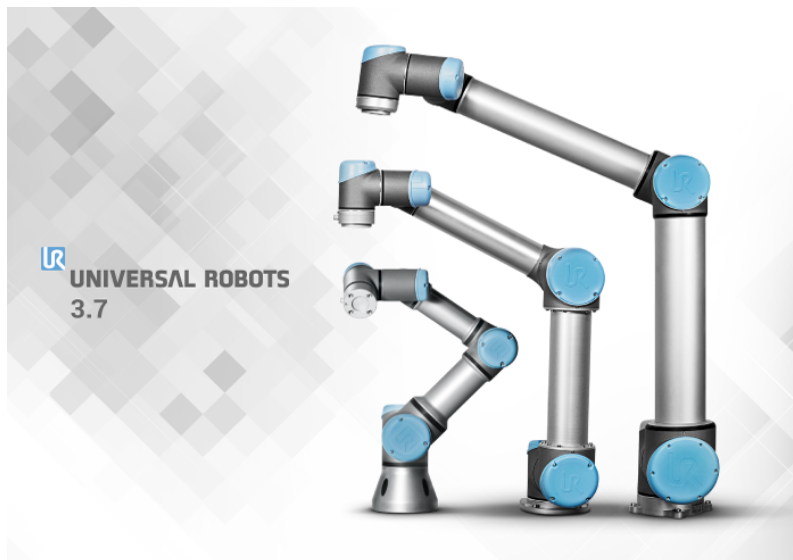




# UNIVERSAL ROBOTS

## PolyScope-Handbuch



**Version 3.7.0**

Übersetzung der originalen Anleitungen (de)

Die hier enthaltenen Informationen sind Eigentum von Universal Robots A/S und dürfen nur im Ganzen oder teilweise vervielfältigt werden, wenn eine vorherige schriftliche Genehmigung von Universal Robots A/S vorliegt. Diese Informationen können jederzeit und ohne vorherige Ankündigung geändert werden und sind nicht als Verbindlichkeit von Universal Robots A/S auszulegen. Dieses Handbuch wird regelmäßig geprüft und überarbeitet.

Universal Robots A/S übernimmt keinerlei Verantwortung für jedwede Fehler oder Auslassungen in diesem Dokument.

Copyright © 2009–2018 by Universal Robots A/S

Das Logo von Universal Robots ist eine eingetragene Handelsmarke von Universal Robots A/S.

# Inhaltsverzeichnis

<b>II PolyScope-Handbuch</b>	<b>II-1</b>
<b>10 Sicherheitskonfiguration</b>	<b>II-3</b>
10.1 Einleitung . . . . .	II-3
10.2 Änderung der Sicherheitskonfiguration . . . . .	II-5
10.3 Sicherheitssynchronisation und Fehler . . . . .	II-5
10.4 Toleranzen . . . . .	II-6
10.5 Sicherheitsprüfsumme . . . . .	II-7
10.6 Sicherheitsmodi . . . . .	II-7
10.7 Freedrive-Modus . . . . .	II-8
10.7.1 Zurückfahren . . . . .	II-8
10.8 Passwortsperrung . . . . .	II-8
10.9 Übernehmen . . . . .	II-9
10.10 Allgemeine Grenzwerte . . . . .	II-9
10.11 Gelenkgrenzen . . . . .	II-12
10.12 Grenzen . . . . .	II-13
10.12.1 Auswählen einer zu konfigurierenden Grenze . . . . .	II-14
10.12.2 3D-Visualisierung . . . . .	II-14
10.12.3 Sicherheitsebenenkonfiguration . . . . .	II-15
10.12.4 Werkzeuggrenzkonfiguration . . . . .	II-18
10.13 Sicherheits-E/A . . . . .	II-20
10.13.1 Eingangssignale . . . . .	II-20
10.13.2 Ausgangssignale . . . . .	II-22
<b>11 Programmierung starten</b>	<b>II-25</b>
11.1 Einleitung . . . . .	II-25
11.2 Erste Schritte . . . . .	II-26
11.2.1 Installation des Roboterarms und des Controllers . . . . .	II-26
11.2.2 Ein- und Ausschalten des Controllers . . . . .	II-26
11.2.3 Ein- und Ausschalten des Roboterarms . . . . .	II-26
11.2.4 Schnellstart . . . . .	II-27
11.2.5 Das erste Programm . . . . .	II-27
11.3 PolyScope-Programmierschnittstelle . . . . .	II-29
11.4 Startbildschirm . . . . .	II-30
11.5 Initialisierungsbildschirm . . . . .	II-31
<b>12 Bildschirm-Editoren</b>	<b>II-33</b>
12.1 Ausdruckeditor auf dem Bildschirm . . . . .	II-33
12.2 Bearbeitungsanzeige „Pose“ . . . . .	II-33

<b>13 Roboter-Steuerung</b>	<b>II-37</b>
13.1 Move-Tab	II-37
13.1.1 Roboter	II-37
13.1.2 Funktion und Werkzeugposition	II-38
13.1.3 Bewegung des Werkzeuges	II-38
13.1.4 Bewegung der Gelenke	II-38
13.1.5 Freedrive	II-38
13.2 E/A-Tab	II-39
13.3 MODBUS-Client-E/A	II-40
13.4 AutoMove-Tab	II-41
13.5 Installation → Laden/Speichern	II-42
13.6 Installation → TCP-Konfiguration	II-43
13.6.1 Hinzufügen, Ändern und Entfernen von TCPs	II-43
13.6.2 Standard-TCP und aktiver TCP	II-44
13.6.3 Anlernen (Teaching) der TCP-Position	II-44
13.6.4 Anlernen (Teaching) der TCP-Ausrichtung	II-45
13.6.5 Nutzlast	II-45
13.6.6 Schwerpunkt	II-45
13.7 Installation/ → Montage	II-46
13.8 Installation → E/A-Einstellung	II-47
13.8.1 E/A Signaltyp	II-48
13.8.2 Zuordnen von benutzerdefinierten Namen	II-48
13.8.3 E/A-Aktionen und E/A-Tabsteuerung	II-48
13.9 Installations→sicherheit	II-49
13.10 Installations→variablen	II-49
13.11 Installation → MODBUS-Client-E/A-Einstellung	II-50
13.12 Installation → Funktionen	II-54
13.12.1 Verwenden einer Funktion	II-55
13.12.2 Neuer Punkt	II-56
13.12.3 Neue Linie	II-57
13.12.4 Funktion Ebene	II-58
13.12.5 Beispiel: Manuelle Anpassung einer Funktion zur Anpassung eines Programms	II-59
13.12.6 Beispiel: Dynamisches Aktualisieren einer Funktion	II-60
13.13 Einrichtung der Fließbandverfolgung	II-61
13.14 Sanfter Übergang zwischen Sicherheitsmodi	II-62
13.15 Installation → Standardprogramm	II-62
13.16 Der Tab „Protokoll“	II-64
13.16.1 Fehlerberichte speichern	II-64
13.17 „Laden“ - Anzeige	II-65
13.18 Der „Ausführen“-Tab	II-67
<b>14 Programmierung</b>	<b>II-69</b>
14.1 Neues Programm	II-69
14.2 Programm - Tab	II-70
14.2.1 Programmstruktur	II-70
14.2.2 Programmausführungsanzeige	II-71
14.2.3 Schaltfläche „Suchen“	II-71

14.2.4	Rückgängig/Erneut ausführen - Taste . . . . .	II-72
14.2.5	Programm-Dashboard . . . . .	II-72
14.3	Variablen . . . . .	II-73
14.4	Befehl: Leer . . . . .	II-74
14.5	Befehl: Move . . . . .	II-75
14.6	Befehl: Fester Wegpunkt . . . . .	II-78
14.7	Befehl: Relativer Wegpunkt . . . . .	II-84
14.8	Befehl: Variabler Wegpunkt . . . . .	II-85
14.9	Befehl: Warten . . . . .	II-86
14.10	Befehl: Einstellen . . . . .	II-86
14.11	Befehl: Meldung . . . . .	II-87
14.12	Befehl: Halt . . . . .	II-88
14.13	Befehl: Kommentar . . . . .	II-88
14.14	Befehl: Ordner . . . . .	II-89
14.15	Befehl: Schleife . . . . .	II-90
14.16	Befehl: Unterprogramm . . . . .	II-91
14.17	Befehl: Zuordnung . . . . .	II-92
14.18	Befehl: If . . . . .	II-93
14.19	Befehl: Script . . . . .	II-94
14.20	Befehl: Ereignis . . . . .	II-95
14.21	Befehl: Thread . . . . .	II-96
14.22	Befehl: Switch . . . . .	II-96
14.22.1	Timer . . . . .	II-97
14.23	Befehl: Muster . . . . .	II-98
14.24	Befehl: Kraft . . . . .	II-99
14.25	Befehl: Palettieren . . . . .	II-103
14.26	Befehl: Suchen . . . . .	II-104
14.27	Befehl: Fließband-Tracking . . . . .	II-108
14.28	Befehl: Unterdrücken . . . . .	II-108
14.29	Grafik-Tab . . . . .	II-108
14.30	Struktur-Tab . . . . .	II-110
14.31	Der „Variablen“ -Tab . . . . .	II-111
14.32	Befehl: Variablen-Initialisierung . . . . .	II-112
<b>15</b>	<b>Set-up-Bildschirm</b>	<b>II-113</b>
15.1	Sprachen und Einheiten . . . . .	II-114
15.2	Roboter aktualisieren . . . . .	II-115
15.3	Passwort festlegen . . . . .	II-116
15.4	Bildschirm kalibrieren . . . . .	II-117
15.5	Netzwerk einstellen . . . . .	II-118
15.6	Uhrzeit einstellen . . . . .	II-118
15.7	URCaps-Einstellung . . . . .	II-119



**Teil II**

**PolyScope-Handbuch**



# 10 Sicherheitskonfiguration

## 10.1 Einleitung

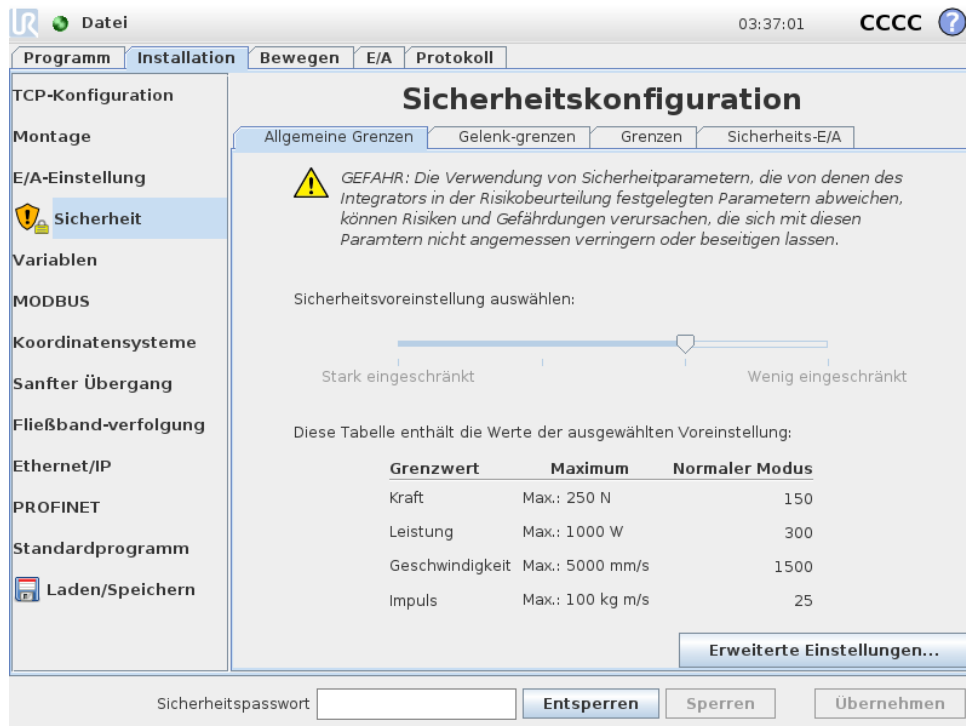
Der Roboter ist mit einem fortschrittlichen Sicherheitssystem ausgestattet. Abhängig von den bestimmten Charakteristiken seines Wirkungsbereichs sind die Einstellungen für das Sicherheitssystem so zu konfigurieren, dass die Sicherheit des Personals und der Geräte im Umfeld des Roboters garantiert werden kann. Das Anwenden von Einstellungen, die durch die Risikobewertung definiert wurden, gehört zu den ersten Handlungen des Integrators. Einzelheiten zum Sicherheitssystem finden Sie hier Hardware-Installationshandbuch.



### GEFAHR:

1. Die Verwendung und Konfiguration von sicherheitsrelevanten Funktionen und Schnittstellen muss gemäß der Risikobewertung erfolgen, die der Integrator für eine bestimmte Roboteranwendung durchführt (siehe Hardware-Installationshandbuch).
2. Die Sicherheitskonfigurationseinstellungen für Set-up und Teaching müssen gemäß der Risikobewertung des Integrators vorgenommen werden, bevor der Roboterarm zum ersten Mal eingeschaltet wird.
3. Alle Sicherheitskonfigurationseinstellungen, auf die über diesen Bildschirm zugegriffen werden kann, sowie deren Unter-Tabs müssen entsprechend der Risikobewertung des Integrators vorgenommen werden.
4. Der Integrator muss sicherstellen, dass alle Änderungen an den Sicherheitseinstellungen entsprechend seiner Risikobewertung durchgeführt werden.
5. Der Integrator hat dafür zu sorgen (z.B. durch einen Passwortschutz), dass es Unbefugten nicht möglich ist, Änderungen an der Sicherheitskonfiguration vorzunehmen.

Auf den Sicherheitskonfigurations-Bildschirm können Sie vom Startbildschirm aus zugreifen (siehe 11.4), indem Sie die Taste **Roboter programmieren** drücken, den Tab **Installation** auswählen und **Sicherheit** antippen. Die Sicherheitskonfiguration ist passwortgeschützt; siehe 10.8.



Die Sicherheitseinstellungen bestehen aus einer Anzahl von Grenzwerten, die verwendet werden, um die Bewegungen des Roboterarms zu beschränken, und den Sicherheitsfunktionseinstellungen für die konfigurierbaren Ein- und Ausgänge. Sie werden in den folgenden Unter-Tabs auf dem Sicherheitsbildschirm definiert:

- Der Unter-Tab *Allgemeine Grenzen* definiert die maximale *Kraft*, *Leistung*, *Geschwindigkeit* und das maximale *Drehmoment* des Roboterarms. Wenn das Risiko besonders hoch ist, dass der Roboterarm mit einer Person oder mit Teilen seiner Umgebung kollidieren könnte, müssen diese Einstellungen auf niedrige Werte festgelegt werden. Wenn das Risiko niedrig ist, ermöglichen es höhere allgemeine Grenzen dem Roboter, sich schneller zu bewegen und mehr Kraft auf seine Umgebung auszuüben. Für weitere Details, siehe 10.10.
- Der Unter-Tab *Gelenkgrenzen* besteht aus den Grenzen für die *Gelenkgeschwindigkeit* und *Gelenkposition*. Die Grenzen für die *Gelenkgeschwindigkeit* definieren die maximale Winkelgeschwindigkeit individueller Gelenke und dienen der weiteren Beschränkung der Geschwindigkeit des Roboterarms. Die Grenzen für die *Gelenkposition* definieren den zulässigen Positionsbereich der individuellen Gelenke (im Gelenkraum). Für weitere Details, siehe 10.11.
- Der Unter-Tab *Grenzen* definiert die Sicherheitsebenen (im kartesischen Raum) und eine Werkzeugausrichtungsgrenze für den Roboter-TCP. Die Sicherheitsebenen können entweder als harte Grenzen für die Position des Roboter-TCP oder als Auslöser für die Sicherheitsgrenzen des *Reduzierten* Modus konfiguriert werden (siehe 10.6)). Die Werkzeugausrichtungsgrenze setzt eine harte Grenze für die Ausrichtung des Roboter-TCPs. Für weitere Details, siehe 10.12.
- Der Unter-Tab *Sicherheits-E/A* definiert Sicherheitsfunktionen für konfigurierbare Ein- und Ausgänge (siehe 13.2). Zum Beispiel kann *Notabschaltung* als ein Eingang konfiguriert werden. Für weitere Details, siehe 10.13.

Copyright ©2009–2018 by Universal Robots A/S. Alle Rechte vorbehalten.

## 10.2 Änderung der Sicherheitskonfiguration

Änderungen bei Sicherheitskonfigurationseinstellungen sind nur gemäß der Risikobewertung des Integrators vorzunehmen.



Die empfohlene Prozedur zum Ändern der Sicherheitskonfiguration ist wie folgt:

1. Stellen Sie sicher, dass die Änderungen im Einklang mit der Risikobewertung des Integrators durchgeführt werden.
2. Passen Sie die Sicherheitseinstellungen an die Risikobewertung des Integrators an.
3. Stellen Sie sicher, dass die Sicherheitseinstellungen aktiv sind.
4. Fügen Sie den folgenden Text in das Bedienerhandbuch ein: „Stellen Sie vor jeglichen Arbeiten in der Nähe des Roboters sicher, dass die Sicherheitskonfiguration wie erwartet agiert. Dies kann beispielsweise getestet werden, indem Sie die Prüfsumme in der oberen rechten Ecke des PolyScope überprüfen (siehe 10.5 in PolyScope-Handbuch).“

---

## 10.3 Sicherheitssynchronisation und Fehler

Der Status der aktiven Sicherheitskonfiguration im Vergleich zu der aktuell in der GUI mit der Installationsdatei geladenen Konfiguration, wird durch das Schild-Symbol neben dem Text `Sicherheit` auf der linken Seite des Bildschirms angezeigt. Diese Symbole bieten eine unkomplizierte Anzeige des aktuellen Zustands. Sie sind wie folgt definiert:

-  *Konfiguration synchronisiert*: Zeigt an, dass die GUI-Installation mit der derzeit aktiven Sicherheitskonfiguration übereinstimmt. Es wurden keine Änderungen vorgenommen.
-  *Konfiguration geändert*: Zeigt an, dass die GUI-Installation mit der derzeit aktiven Sicherheitskonfiguration NICHT übereinstimmt.

Bei der Bearbeitung der Sicherheitskonfiguration zeigt das Schild-Symbol an, ob die aktuellen Einstellungen übernommen wurden.

Wenn eines der Textfelder im Tab `Sicherheit` eine ungültige Eingabe enthält, befindet sich die Sicherheitskonfiguration im Fehlerzustand. Dies wird auf mehrere Arten angezeigt:

1. Ein rotes Fehlersymbol erscheint neben dem Text `Sicherheit` auf der linken Seite des Bildschirms.
2. Fehler enthaltende Unter-Tabs sind oben mit einem roten Fehlersymbol markiert.
3. Textfelder, die Fehler enthalten, werden mit einem roten Hintergrund markiert.

Wenn Fehler vorhanden sind und Sie versuchen, den Tab `Installation` zu verlassen, erscheint ein Dialog mit den folgenden Optionen:

1. Lösen Sie das Problem, um alle Fehler zu beseitigen. Dies wird sichtbar wenn das rote Fehlersymbol neben dem Text `Sicherheit` auf der linken Seite des Bildschirms verschwunden ist.
2. Zuvor aktive Sicherheitskonfiguration wieder übernehmen. Verwirft jegliche Änderungen und lässt Sie nach Belieben fortfahren.

Wenn keine Fehler vorhanden sind und Sie versuchen, den Tab zu verlassen, erscheint ein Dialog mit diesen Optionen:

1. Änderungen übernehmen und das System neustarten. Übernimmt die Sicherheitskonfigurationsänderungen und startet das System neu. Hinweis: Dies bedeutet nicht, dass alle Änderungen gespeichert wurden; Herunterfahren des Roboters zu diesem Zeitpunkt macht alle Änderungen an der Roboterinstallation, einschließlich der Sicherheitskonfiguration, rückgängig.
2. Zuvor aktive Sicherheitskonfiguration wieder übernehmen. Verwirft jegliche Änderungen und lässt Sie nach Belieben fortfahren.

## 10.4 Toleranzen

Der *Roboterarm* verwendet integrierte Toleranzen, die Sicherheitsübertretungen verhindern. Eine Sicherheitstoleranz ist die Differenz zwischen einer Sicherheitsgrenze und dem maximalen, operativen Wert. So beträgt zum Beispiel die allgemeine Geschwindigkeitstoleranz  $-150\text{mm/s}$ . Konfiguriert der Benutzer demnach eine Geschwindigkeitsbegrenzung von  $250\text{mm/Sek.}$ , so beläuft sich die maximale Betriebsgeschwindigkeit auf  $250 - 150 = 100\text{mm/s}$ . Sicherheitstoleranzen verhindern Schutzverletzungen aber ermöglichen Fluktuationen im Programmverhalten. So kann es zum Beispiel eine Situation erfordern, dass der *Roboterarm* für die Handhabung einer schweren Nutzlast kurzzeitig über die normale, maximale Betriebsgeschwindigkeit hinausgehen muss, um einer programmierten Bahn zu folgen. Ein Beispiel für einen solchen Fall finden Sie in der Abbildung 10.1.



**WARNUNG:**

Eine Risikobewertung mit den Grenzwerten und ohne Toleranzen ist stets erforderlich.



**WARNUNG:**

Toleranzen sind von der Softwareversion abhängig. Beim Aktualisieren der Software werden die Toleranzen u. U. geändert. Toleranzänderungen sind stets in den Änderungsbeschreibungen neuer Versionen enthalten.

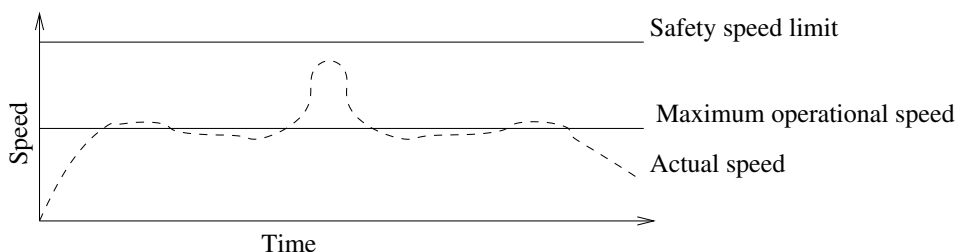


Abbildung 10.1: Beispiel für Sicherheitstoleranz

## 10.5 Sicherheitsprüfsumme

Der Text in der Ecke rechts oben auf dem Bildschirm bietet eine Kurzfassung der Sicherheitskonfiguration, die der Roboter derzeit nutzt. Wenn sich der Text ändert, zeigt dies an, dass sich auch die Sicherheitskonfiguration geändert hat. Durch Klicken auf die Prüfsumme werden die Details zur derzeit aktiven Sicherheitskonfiguration angezeigt.

## 10.6 Sicherheitsmodi

Unter normalen Bedingungen (d. h. wenn kein Sicherheitsstopp aktiv ist), befindet sich das Sicherheitssystem in einem der folgenden *Sicherheitsmodi*, von denen jeder über einen eigenen Satz von Sicherheitsgrenzen verfügt:

*Normaler Modus:* Der Sicherheitsmodus, der standardmäßig aktiv ist

*Reduzierter Modus:* Ist aktiv, wenn sich der TCP des Roboters in einer *Reduzierten Modus auslösen-* Ebene befindet (siehe 10.12) oder bei Auslösung durch einen konfigurierbaren Eingang (siehe 10.13).

*Wiederherstellungsmodus:* Kam es durch den Roboterarm zu einer Grenzwertüberschreitung eines der anderen Modi (d. h. des *Normalen* oder des *Reduzierten Modus*)<sup>1</sup> und wurde ein Stopp der Kategorie 0 durchgeführt, so wird der Roboterarm im **Wiederherstellungsmodus** gestartet. In diesem Modus kann der Roboter mittels **MoveTab** oder **Freedrive** langsam wieder in den zulässigen Bereich zurückgefahren werden. In diesem Modus ist es nicht möglich, Programme für den Roboter auszuführen.



### **WARNUNG:**

Beachten Sie, dass Grenzwerte für die *Gelenkposition*, *TCP-Position* und *TCP-Ausrichtung* im *Wiederherstellungsmodus* deaktiviert sind. Lassen Sie daher beim Bewegen des Roboterarms äußerste Vorsicht walten.

Die Unter-Tabs des *Sicherheitskonfigurations-Bildschirms* ermöglichen es dem Benutzer, separate Sätze von Sicherheitsgrenzen für den *Normalen* und den *Reduzierten Modus* festzulegen. Die Werkzeug- und Gelenkgrenzwerte des *Reduzierten Modus* bezüglich der Geschwindigkeit und des Schwungs müssen strenger sein als die des *Normalen Modus*.

Wenn eine Sicherheitsgrenze des aktiven Grenzwertsatzes überschritten wird, führt der Roboterarm einen Stopp der Kategorie 0 aus. Wenn eine aktive Sicherheitsgrenze, wie eine Gelenkpositionsgrenze oder eine Sicherheitsebene bereits beim Einschalten des Roboterarms überschritten ist, wird er im *Wiederherstellungsmodus* gestartet. So kann er leicht in den Bereich innerhalb der Sicherheitsgrenzen bewegt werden. Im *Wiederherstellungsmodus* ist die Bewegung des Roboterarms auf einen festen Bereich beschränkt, der vom Benutzer nicht angepasst werden kann. Details zu den Grenzwerten des *Wiederherstellungsmodus* befinden sich unter *Hardware-Installationshandbuch*.

<sup>1</sup>Gemäß IEC 60204-1, siehe Glossar für weitere Details.

## 10.7 Freedrive-Modus

Wenn sich im *Freedrive*-Modus (siehe 13.1.5) die Bewegung des Roboterarms bestimmten Grenzen annähert, fühlt der Benutzer einen Widerstand. Diese Kraft wird für die Grenzen auf Position, Ausrichtung und Geschwindigkeit des Roboter-TCPs sowie die Position und die Geschwindigkeit der Gelenke generiert.

Der Zweck dieses Widerstandes ist es, den Benutzer darüber zu informieren, dass sich die aktuelle Position oder Geschwindigkeit einem Grenzwert annähert und um zu vermeiden, dass der Roboter diese Grenze überschreitet. Wird jedoch ausreichend Kraft auf den Roboterarm durch den Benutzer ausgeübt, kann es zu einer Grenzwertüberschreitung kommen. Der Widerstand wird größer, je näher der Roboterarm sich der Grenze annähert.

### 10.7.1 Zurückfahren

Im *Freedrive*-Modus können die Robotergelenke mit relativ wenig Kraft bewegt werden, da die Bremsen gelöst sind. Während der Initialisierung des Roboterarms können kleinere Vibrationen auftreten, wenn die Roboterbremsen gelöst werden. In bestimmten Situationen, wenn z. B. der Roboter kurz vor einer Kollision steht, sind diese Erschütterungen nicht erwünscht. Mit der Funktion *Zurückfahren* kann dann erzwungen werden, dass bestimmte Gelenke in die gewünschte Position gebracht werden, ohne alle Bremsen im Roboterarm lösen zu müssen.

Um *Zurückfahren* zu aktivieren:

1. Drücken Sie auf ON, um die Gelenke mit Strom zu versorgen. Der Roboterstatus ist auf „Leerlauf“ gesetzt. Geben Sie die Bremsen **nicht** frei (d. h.: Drücken Sie nicht auf START).
2. Drücken und halten Sie die *Freedrive*-Taste. Der Roboterstatus wechselt zu „Zurückfahren“.
3. Die Bremsen werden nur in den Gelenken gelöst, auf die erheblicher Druck angewendet wird, solange die *Freedrive*-Taste betätigt wird/gedrückt ist. Wenn *Zurückfahren* verwendet wird, ist der Roboter schwer zu bewegen.

## 10.8 Passwortsperr

Alle Einstellungen auf diesem Bildschirm sind gesperrt, bis das korrekte Sicherheitspasswort (siehe 15.3) in das weiße Textfeld unten im Bildschirm eingegeben und die Taste *Entsperren* gedrückt wurde. Der Bildschirm kann durch Klick auf die Taste *Sperren* wieder gesperrt werden. Der Tab *Sicherheit* wird automatisch gesperrt, wenn der Sicherheitskonfigurations-Bildschirm verlassen wird. Wenn die Einstellungen gesperrt sind, wird neben dem Text *Sicherheit* auf der linken Seite des Bildschirms ein Schlosssymbol angezeigt. Ein entsprechendes Symbol wird angezeigt, wenn die Einstellungen freigegeben sind.



### HINWEIS:

Beachten Sie, dass der Roboterarm ausgeschaltet ist, solange der Sicherheitskonfigurations-Bildschirm freigegeben ist.

## 10.9 Übernehmen

Beim Freigeben der Sicherheitskonfiguration ist der Roboterarm ausgeschaltet, solange Änderungen vorgenommen werden. Der Roboterarm kann vor dem Übernehmen oder dem Abrechnen der Änderungen nicht eingeschaltet werden. Danach ist ein manuelles Einschalten des Initialisierungsbildschirms erforderlich.

Alle Änderungen an der Sicherheitskonfiguration müssen vor dem Verlassen des Installations-Tab übernommen oder rückgängig gemacht werden. Diese Änderungen treten *nicht* in Kraft, bevor die Taste Übernehmen gedrückt wurde und dies bestätigt wurde. Die Bestätigung erfordert eine Sichtprüfung der Änderungen am Roboterarm. Aus Sicherheitsgründen sind die Informationen in SI-Einheiten angegeben. Ein Beispiel des Bestätigungsdialogs finden Sie untenstehend.

Bestätigung der angewandten Sicherheitskonfiguration				
Allgemeine Grenzen	Gelenk-grenzen	Grenzen	Sicherheits-E/A	Sonstige
Grenzwert	Normaler Modus	Reduzierter Modus		
Kraft	150,00	120,00 N		
Leistung	300,00	200,00 W		
Geschwindigkeit	1,50	0,75 m/s		
Impuls	25,00	10,00 kg m/s		

Darüber hinaus werden die Änderungen bei der Bestätigung automatisch als Teil der aktuellen Roboterinstallation gespeichert. Siehe 13.5 für weitere Informationen zum Speichern der Roboterinstallation.

## 10.10 Allgemeine Grenzwerte

Die allgemeinen Sicherheitsgrenzen dienen der Begrenzung der linearen Geschwindigkeit des Roboter-TCPs und der Kraft, die dieser auf die Umgebung ausüben kann. Sie setzen sich aus den folgenden Werten zusammen:

*Kraft*: Eine Grenze für die maximale Kraft, die der Roboter-TCP auf die Umgebung ausübt.

*Leistung*: Eine Grenze für die maximale mechanische Arbeit, die vom Roboter in der Umgebung produziert wird, wobei berücksichtigt wird, dass die Nutzlast Teil des Roboters und nicht der Umgebung ist.

*Geschwindigkeit:* Eine Grenze für die maximale lineare Geschwindigkeit des Roboter-TCPs.

*Drehmoment:* Eine Grenze für das Maximale Drehmoment des Roboters.

Es gibt zwei Wege zur Konfiguration der allgemeinen Sicherheitsgrenzen in der Installation; *Grundlegende Einstellungen* und *Erweiterte Einstellungen*, die nachstehend ausführlicher beschrieben werden.

Die Definition der allgemeinen Sicherheitsgrenzen legt nur die Grenzen für das Werkzeug, jedoch nicht die allgemeinen Grenzen des Roboterarms fest. Das bedeutet, dass trotz spezifizierter Geschwindigkeitsgrenze *nicht* garantiert ist, dass andere Teile des Roboterarms dieselbe Grenze einhalten.

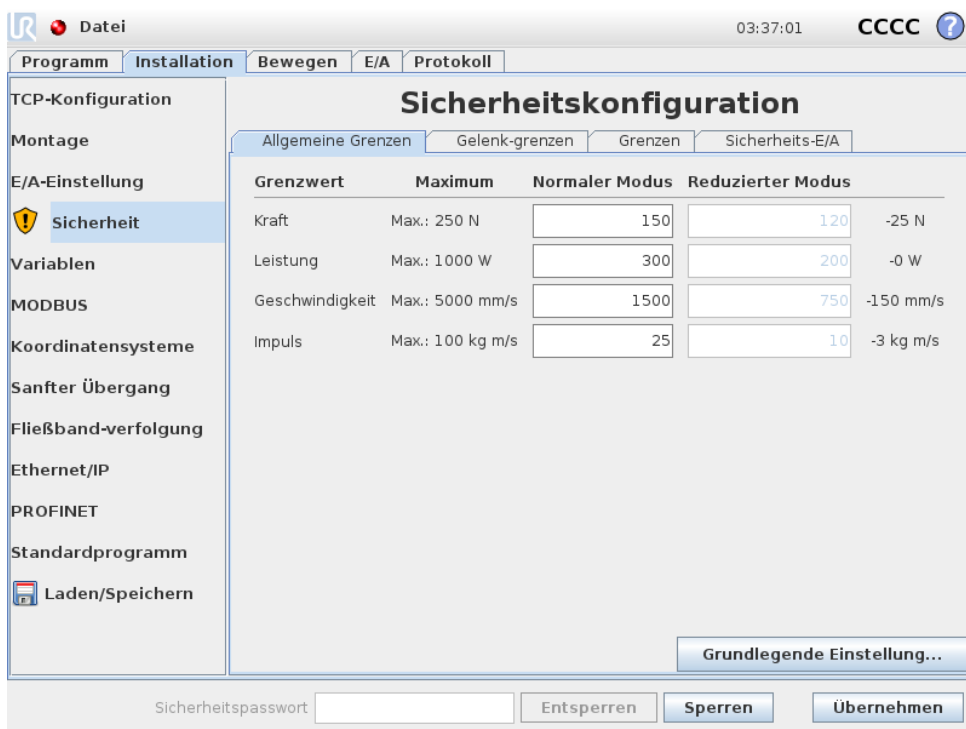
Nähert sich die aktuelle Geschwindigkeit des Roboter-TCP im *Freedrive*-Modus (siehe 13.1.5) zu sehr dem Grenzwert der *Geschwindigkeit*, fühlt der Benutzer einen Widerstand, der mit zunehmender Annäherung an die Geschwindigkeitsgrenze stärker wird. Die Kraft wird generiert, wenn die aktuelle Geschwindigkeit sich innerhalb von etwa 250 mm/s von der Grenze bewegt.

**Grundeinstellungen** Das Unterfeld allgemeine Grundeinstellungen, angezeigt als Standardbildschirm, verfügt über einen Schieber mit den folgenden vordefinierten Wertesätzen für die allgemeinen Grenzen in den *Normalen* und *Reduzierten* Modi:

Die bestimmten Wertesätze sind in der GUI dargestellt. Vordefinierte Wertesätze sind nur Vorschläge und kein Ersatz für eine ordnungsgemäße Risikobewertung.

**Zu erweiterten Einstellungen wechseln** Sollte *keiner* der vordefinierten Wertesätze befriedigend sein, kann die Taste *Erweiterte Einstellungen...* gedrückt werden, um zum Bildschirm „Erweiterte allgemeine Grenzwerte“ zu gelangen.

**Erweiterte Einstellungen**



Copyright ©2009–2018 by Universal Robots A/S. Alle Rechte vorbehalten.

## 10.10 Allgemeine Grenzwerte

Hier kann jede der allgemeinen Grenzen, die in 10.10 definiert sind, unabhängig von den anderen geändert werden. Dies erfolgt, indem das entsprechende Textfeld angetippt und der neue Wert eingegeben wird. Der höchste akzeptierte Wert für jede der Grenzen ist in der Spalte mit dem Namen *Maximum* aufgelistet. Die Kraftbegrenzung kann auf einen Wert zwischen 100 N bis 250 N und die Leistungsgrenze zwischen 80 W bis 1000 W festgesetzt werden.

Hinweis: Beachten Sie, dass die Felder für Begrenzungen im *Reduzierten Modus* deaktiviert sind, wenn weder eine Sicherheitsebene noch ein konfigurierbarer Eingang für die Auslösung eingestellt sind (siehe 10.12 und 10.13 für weitere Details). Weiterhin dürfen die Grenzen für *Geschwindigkeit* und *Schwung* im Modus *Reduziert* nicht höher als ihre Gegenstücke im Modus *Normal* sein.

Die Toleranz und Einheit der Grenzen sind jeweils am Ende der zugehörigen Zeile aufgelistet. Wenn ein Programm ausgeführt wird, wird die Geschwindigkeit des Roboterarms automatisch angepasst, damit keiner der eingegebenen Werte abzüglich der Toleranz überschritten wird (siehe 10.4). Beachten Sie, dass das Minuszeichen vor den Toleranzwerten lediglich angibt, dass die Toleranz vom eingegebenen Wert abgezogen wird. Das Sicherheitssystem führt einen Stopp der Kategorie 0 durch, falls der Roboterarm die Grenze (ohne Toleranz) überschreitet.

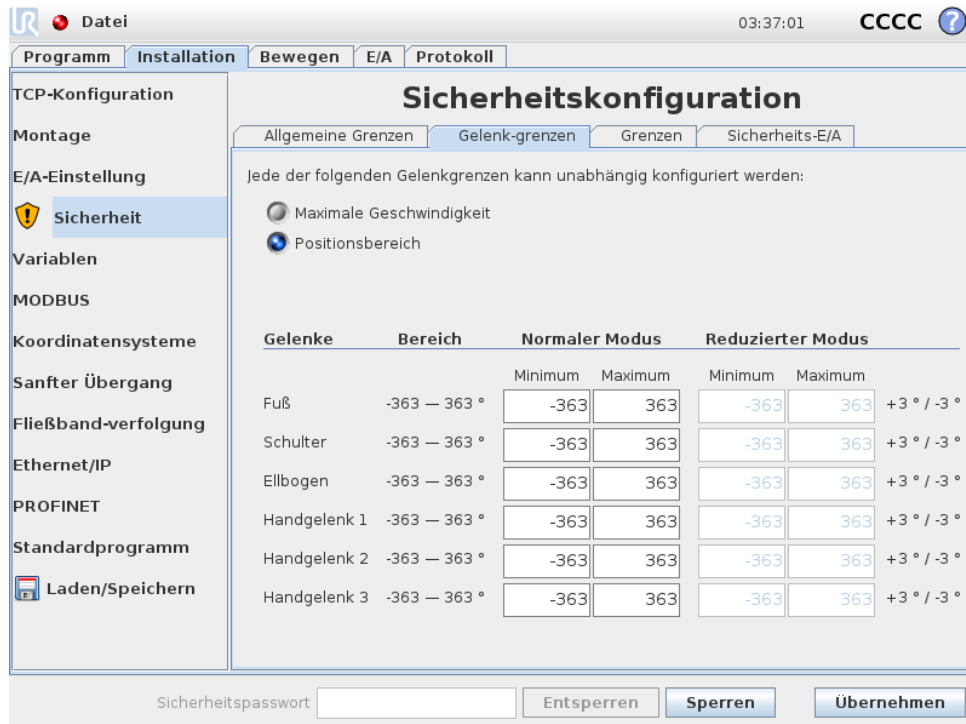


### **WARNUNG:**

Die Geschwindigkeitsgrenze trifft nur auf den Roboter-TCP zu. Daher können sich andere Teile des Roboterarms schneller bewegen, als durch den definierten Wert vorgegeben.

**Zu allgemeinen Grundeinstellungen wechseln** Durch Drücken der Taste *allgemeine Grundeinstellungen* wird wieder zurück zum Bildschirm *allgemeine Grenzen* gewechselt und alle allgemeinen Grenzen werden auf ihren vordefinierten *Standard* zurückgesetzt. Sollten dadurch angepasste Werte verloren gehen, wird ein Pop-up-Dialog zum Bestätigen der Aktion angezeigt.

## 10.11 Gelenkgrenzen



Gelenkgrenzen beschränken die Bewegung einzelner Gelenke im Gelenkraum, d.h. sie beziehen sich nicht auf den kartesischen Raum, sondern auf die interne (Drehungs-) Position der Gelenke und deren Drehgeschwindigkeit. Die Optionsschaltflächen im oberen Bereich des Unterfelds ermöglichen eine unabhängige Einstellung der Maximalen Geschwindigkeit und des Positionsbereichs für die Gelenke.

Wenn sich die aktuelle Position oder die Geschwindigkeit eines Gelenks im *Freedrive* Modus (siehe 13.1.5) den Grenzwerten zu sehr nähert, fühlt der Benutzer einen Widerstand, der immer stärker wird, je mehr sich das Gelenk der Grenze annähert. Die Kraft wird generiert, wenn die Gelenkgeschwindigkeit sich in etwa  $20^\circ/s$  von der Geschwindigkeitsgrenze oder die Gelenkposition sich in etwa  $8^\circ$  der Positionsgrenze befindet.

**Maximale Geschwindigkeit** Diese Position definiert die maximale Winkelgeschwindigkeit für jedes Gelenk. Dies erfolgt, indem das entsprechende Textfeld angetippt und der neue Wert eingegeben wird. Der höchste akzeptierte Wert ist in der Spalte mit dem Namen *Maximum* aufgelistet. Keiner der Werte kann unter den Toleranzwert gesetzt werden.

Beachten Sie, dass die Felder für Grenzen im Modus *Reduziert* deaktiviert sind, wenn weder eine Sicherheitsebene noch ein konfigurierbarer Eingang für die Auslösung eingestellt sind (siehe 10.12 und 10.13 für weitere Details). Weiterhin dürfen die Grenzen im Modus *Reduziert* nicht höher als ihre Gegenstücke im Modus *Normal* sein.

Die Toleranz und Einheit der Grenzen sind jeweils am Ende der zugehörigen Zeile aufgelistet. Wenn ein Programm ausgeführt wird, wird die Geschwindigkeit des Roboterarms automatisch angepasst, damit keiner der eingegebenen Werte abzüglich der Toleranz überschritten wird (siehe 10.4). Beachten Sie, dass das Minuszeichen vor den Toleranzwerten lediglich angibt, dass die Toleranz vom eingegebenen Wert abgezogen wird. Sollte die Winkelgeschwindigkeit eines

Gelenks den eingegebenen Wert (ohne Toleranz) dennoch überschreiten, führt das Sicherheitssystem einen Stopp der Kategorie 0 aus.

**Positionsbereich** Dieser Bildschirm definiert den Positionsbereich für jedes Gelenk. Dies erfolgt, indem das entsprechende Textfeld angetippt wird und die neuen Werte für die Ober- und Untergelenkpositionsgrenze eingegeben werden. Das eingegebene Intervall muss sich innerhalb der Werte, die in der Spalte namens *Bereich* aufgelistet sind, bewegen und die Untergrenze darf die Obergrenze nicht überschreiten.

Hinweis: Beachten Sie, dass die Felder für Begrenzungen im *Reduzierten Modus* deaktiviert sind, wenn weder eine Sicherheitsebene noch ein konfigurierbarer Eingang für die Auslösung eingestellt sind (siehe 10.12 und 10.13 für weitere Details).

Die Toleranzen und Einheit der Grenzen sind jeweils am Ende der zugehörigen Zeile aufgelistet. Der erste Toleranzwert gilt für den Mindestwert, der zweite für den Maximalwert. Die Programmausführung wird abgebrochen, sobald die Position eines Gelenkes den Bereich, der sich aus der Addition der ersten Toleranz zum eingegebenen Mindestwert und Subtraktion der zweiten Toleranz vom eingegebenen Maximalwert errechnet, verlässt, falls es sich weiterhin entlang der voraussichtlichen Bahn fortbewegt. Beachten Sie, dass das Minuszeichen vor den Toleranzwerten lediglich angibt, dass die Toleranz vom eingegebenen Wert abgezogen wird. Sollte die Position des Gelenkes den eingegebenen Bereich dennoch verlassen, führt das Sicherheitssystem einen Stopp der Kategorie 0 aus.

---

## 10.12 Grenzen

In diesem Tab können Sie Grenzwerte bestehend aus Sicherheitsebenen und ein Limit auf der maximal zulässigen Abweichung der Roboterwerkzeugausrichtung konfigurieren. Es ist auch möglich, Ebenen zu definieren, die einen Übergang in den *Reduzierten* Modus auslösen.

Sicherheitsebenen können verwendet werden, um den zulässigen Wirkungsbereich des Roboters zu beschränken, indem erzwungen wird, dass der Roboter-TCP auf der richtigen Seite der definierten Ebenen bleibt und diese nicht durchquert. Es können bis zu acht Sicherheitsebenen konfiguriert werden. Die Beschränkung der Ausrichtung des Werkzeugs kann verwendet werden, um sicherzustellen, dass die Ausrichtung des Roboterwerkzeugs nicht um mehr als den spezifizierten Wert von einer gewünschten Ausrichtung abweicht.



### **WARNUNG:**

Das Definieren von Sicherheitsebenen begrenzt nur den TCP, jedoch nicht die allgemeinen Grenzen des Roboterarms. Dies bedeutet, dass trotz spezifizierter Sicherheitsebene *nicht* garantiert ist, dass andere Teile des Roboterarms dieselbe Grenze einhalten.

Die Konfiguration jedes Grenzlimits basiert auf einer der Funktionen, die in der aktuellen Roboterinstallation definiert sind (siehe 13.12).



**HINWEIS:**

Es wird dringend empfohlen, dass Sie vor der Bearbeitung der Sicherheitskonfiguration alle erforderlichen Funktionen für die Konfiguration der gewünschten Grenzlimits erstellen und ihnen entsprechende Namen zuordnen. Beachten Sie, dass mit dem Entsperren des Tab *Sicherheit* auch der Roboterarm abgeschaltet wird. Damit ist die *Werkzeug*-Funktion (die die aktuelle Position und Ausrichtung des Roboter-TCP beinhaltet) sowie der *Freedrive*-Modus (siehe 13.1.5) nicht mehr verfügbar.

Wenn sich im Modus *Freedrive* (siehe 13.1.5) die aktuelle Position des Roboter-TCP einer Sicherheitsebene zu sehr nähert oder sich die Ausrichtung des Roboterwerkzeugs zu sehr an die spezifizierte, maximale Abweichung annähert, fühlt der Benutzer einen Widerstand, der zunimmt, je mehr sich der TCP der Grenze annähert. Die Widerstand wird generiert, wenn sich der TCP in etwa 5 cm Abstand zu einer Sicherheitsebene befindet oder die Ausrichtungsabweichung des Werkzeugs etwa 3° von der spezifizierten maximalen Abweichung beträgt.

Wenn eine Ebene als *Reduzierten Modus auslösen*-Ebene definiert ist und der TCP sich über den Grenzwert hinaus bewegt, wird der *Reduzierte* Modus aktiv, der wiederum die Sicherheitseinstellungen des *Reduzierten* Modus aktiviert. Auslöseebenen folgen denselben Regeln wie normale Sicherheitsebenen, abgesehen davon, dass sie zulassen, dass der Roboterarm sie durchquert.

**10.12.1 Auswählen einer zu konfigurierenden Grenze**


Das Feld *Sicherheitsgrenzwerte* auf der linken Seite des Tab wird verwendet, um ein zu konfigurierendes Grenzlimes auszuwählen.

Um eine Sicherheitsebene einzurichten, klicken Sie auf einen der acht oberen Einträge, die in dem Feld aufgelistet sind. Wenn die ausgewählte Sicherheitsebene bereits konfiguriert wurde, wird die zugehörige 3D-Darstellung der Ebene in der *3D-Ansicht* (siehe 10.12.2) rechts von diesem Feld hervorgehoben. Die Sicherheitsebene kann im Abschnitt *Eigenschaften der Sicherheitsebene* (siehe 10.12.3) im unteren Tab-Bereich eingestellt werden.

Klicken Sie auf den Eintrag *Werkzeuggrenze*, um die Ausrichtungsgrenze für das Roboterwerkzeug zu konfigurieren. Die Konfiguration des Limits kann im Abschnitt *Eigenschaften der Werkzeuggrenze* (siehe 10.12.4) im unteren Bereich des Tab spezifiziert werden.

Klicken Sie auf die Taste  / , um die 3D-Visualisierung des Grenzlimes ein-/auszuschalten. Falls ein Grenzlimes aktiv ist, wird der *Sicherheitsmodus* (siehe 10.12.3 und 10.12.4) durch eines der folgenden Symbole angezeigt  /  /  / .

**10.12.2 3D-Visualisierung**

Die *3D-Ansicht* zeigt die konfigurierten Sicherheitsebenen und das Limit der Ausrichtungsgrenze für das Roboterwerkzeug zusammen mit der aktuellen Position des Roboterarms an. Im Abschnitt *Sicherheitsgrenzen* werden alle konfigurierten Grenzeinträge, bei denen die Sichtbarkeitsschaltung ausgewählt ist (d. h.  wird angezeigt), zusammen mit dem aktuell ausgewählten Grenzlimes angezeigt.

Die (aktiven) Sicherheitsebenen werden in Gelb und Schwarz zusammen mit einem kleinen Pfeil angezeigt, der für die Normalebene steht, was angibt, auf welcher Seite der Ebene der

Copyright ©2009–2018 by Universal Robots A/S. Alle Rechte vorbehalten.

Roboter-TCP positioniert werden darf. Auslöseebenen werden in blau und grün dargestellt. Ein kleiner Pfeil zeigt die Seite der Ebene an, die *nicht* den Übergang in den *Reduzierten* Modus auslöst. Wenn eine Sicherheitsebene im Feld auf der linken Seite des Tab ausgewählt wurde, wird die zugehörige 3D-Darstellung hervorgehoben.

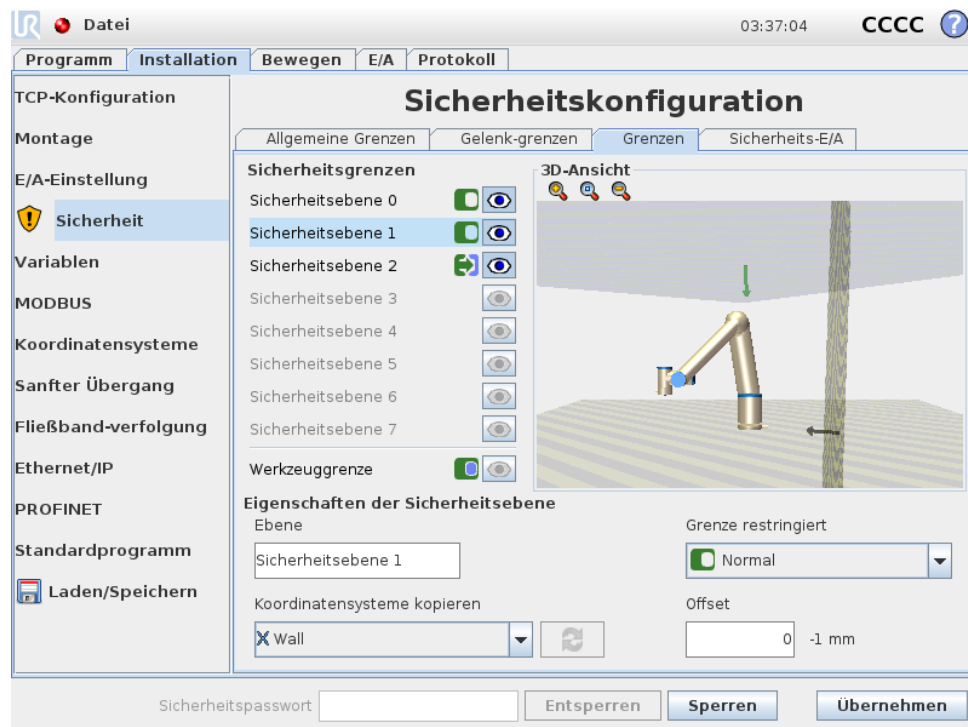
Das Limit der Werkzeugausrichtungsgrenze wird anhand eines sphärischen Kegels visualisiert, wobei ein Vektor die aktuelle Ausrichtung des Roboterwerkzeugs anzeigt. Das Innere des Kegels repräsentiert den zulässigen Bereich für die Werkzeugausrichtung (Vektor).

Wenn eine Ebene oder das Limit der Werkzeugausrichtungsgrenze konfiguriert, jedoch nicht aktiv ist, ist die Visualisierung grau.

Betätigen Sie die Lupensymbole, um hinein-/herauszuzoomen oder ziehen Sie einen Finger darüber, um die Ansicht zu ändern.

### 10.12.3 Sicherheitsebenenkonfiguration




Der Abschnitt *Eigenschaften* der Sicherheitsebene im unteren Teil des Tab definiert die Konfiguration der ausgewählten Sicherheitsebene im Feld *Sicherheitsgrenzen* im oberen linken Tab-Bereich.







**Name** Das Textfeld *Name* ermöglicht es dem Benutzer, der ausgewählten Sicherheitsebene einen Namen zuzuweisen. Dieser Name kann durch Tippen auf das Textfeld und Eingabe eines neuen Namens geändert werden.

**Kopierfunktion** Die Position und die Normale der Sicherheitsebene wird mithilfe einer Funktion (siehe 13.12) von der aktuellen Roboterinstallation spezifiziert. Nutzen Sie die Dropdown-Box auf der unteren linken Seite des Abschnitts *Eigenschaften der Sicherheitsebene*, um eine Funktion auszuwählen. Nur die Punkte und die Ebenentypenfunktionen sind verfügbar. Durch Auswahl des Elements *<Nicht definiert>* wird die Konfiguration der Ebene gelöscht.

Die z-Achse der ausgewählten Funktion zeigt auf den verweigerten Bereich und die Normale der Ebene in die entgegengesetzte Richtung, mit der Ausnahme, dass die Funktion `Basis` ausgewählt wird, in welchem Fall die Normale der Ebene in dieselbe Richtung zeigt. Falls die Ebene als *Reduzierten Modus auslösen*-Ebene konfiguriert ist (siehe 10.12.3), zeigt die Normale der Ebene die Seite an, die *nicht* den Übergang in den *Reduzierten Modus* auslöst.

Es ist zu beachten, dass die Positionsinformationen bei der Konfiguration einer Sicherheitsebene durch Auswahl einer Funktion nur in die Sicherheitsebene *kopiert* werden; die Ebene ist *nicht* mit dieser Funktion verknüpft. Dies bedeutet, dass die Sicherheitsebene bei Änderungen an der Position oder Ausrichtung einer Funktion, die zur Konfiguration einer Sicherheitsebene genutzt wurde, nicht automatisch aktualisiert wird. Wenn sich die Funktion verändert hat, wird dies durch ein  Symbol angezeigt, das sich über dem Funktionseinsteller befindet. Klicken Sie die Taste  neben der Auswahlfunktion, um die Sicherheitsebene mit der aktuellen Position und Ausrichtung der Funktion zu aktualisieren. Das  Symbol wird auch angezeigt, wenn die ausgewählte Funktion von der Installation gelöscht wurde.

**Sicherheitsmodus** Mit dem Dropdown-Menü auf der rechten Seite des Felds `Eigenschaften` der Sicherheitsebene wird der *Sicherheitsmodus* der Sicherheitsebene ausgewählt. Dabei stehen folgende Modi zur Auswahl:

Deaktiviert	Die Sicherheitsebene ist <i>zu keiner Zeit aktiv</i> .
 Normal	Wenn sich das Sicherheitssystem im <i>Normalen</i> Modus befindet, ist eine <i>Normaler-Modus-Ebene aktiv</i> und agiert als <i>strenge Begrenzung</i> der TCP-Position des Roboters.
 Reduziert	Wenn sich das Sicherheitssystem im <i>Reduzierten</i> Modus befindet, ist eine <i>Reduzierter-Modus-Ebene aktiv</i> und agiert als <i>strenge Begrenzung</i> der TCP-Position des Roboters.
 Normal & Reduziert	Wenn sich das Sicherheitssystem im <i>Normalen</i> oder <i>Reduzierten</i> Modus befindet, ist eine <i>Normale u. Reduzierte Modus-Ebene aktiv</i> und agiert als <i>strenge Begrenzung</i> der TCP-Position des Roboters.
 Reduzierten Modus auslösen	Wenn sich das Sicherheitssystem im <i>Normalen</i> oder <i>Reduzierten</i> Modus befindet, ist eine <i>Reduzierten Modus auslösen-Ebene aktiv</i> , die bewirkt, dass das Sicherheitssystem im <i>Reduzierten</i> Modus bleibt, solange sich der TCP des Roboters außerhalb dieser Ebene befindet.

Der ausgewählte *Sicherheitsmodus* wird durch ein Symbol im zugehörigen Eintrag im Feld `Sicherheitsgrenze` angezeigt. Ist der *Sicherheitsmodus* auf `Deaktiviert` eingestellt, wird kein Symbol angezeigt.

**Verdrängung** Wenn eine Funktion in der Dropdown-Box auf der linken Seite des Felds `Eigenschaften` der Sicherheitsebene ausgewählt wurde, kann die Sicherheitsebene seitlich bewegt werden, indem das Textfeld `Verdrängung` im unteren rechten Bereich dieses Felds angetippt und

ein Wert eingegeben wird. Durch die Eingabe eines positiven Werts wird der zulässige Wirkungsbereich des Roboters erhöht, indem die Ebene in die entgegengesetzte Richtung der Ebenennormalen bewegt wird. Durch die Eingabe eines negativen Werts wird der zulässige Bereich verringert, indem die Ebene in Richtung der Ebenennormalen bewegt wird.

Die Toleranz und Einheit für die Verdrängung der Grenzebene wird rechts neben dem Textfeld angezeigt.

**Wirkung *strenger Grenz-Ebenen*** Die Programmausführung wird abgebrochen, wenn die TCP-Position die Grenze einer aktiven, strengen Sicherheitsebene abzüglich der Toleranz überschreitet (siehe 10.4), wenn sie sich weiter entlang der voraussichtlichen Bahn fortbewegt. Beachten Sie, dass das Minuszeichen vor den Toleranzwerten lediglich angibt, dass die Toleranz vom eingegebenen Wert abgezogen wird. Das Sicherheitssystem führt einen Stopp der Kategorie 0 durch, falls die TCP-Position die festgelegte Grenze einer Sicherheitsebene (ohne Toleranz) überschreitet.

**Wirkung von *Reduzierten Modus auslösen- Ebenen*** Wenn kein Sicherheitsstopp aktiv ist und das Sicherheitssystem sich nicht in dem besonderen *Wiederherstellungsmodus* befindet (siehe 10.6), ist es entweder im *Normalen* oder im *Reduzierten* Modus und die Bewegungen des Roboterarms sind durch die jeweiligen Grenzwerte beschränkt.

Standardmäßig befindet sich das Sicherheitssystem im *Normalen* Modus. Es wechselt in den *Reduzierten* Modus, sobald eine der folgenden Situationen eintritt:

- a.) Der TCP des Roboters wird außerhalb einer *Reduzierten Modus auslösen-Ebene* positioniert, d. h. er befindet sich auf der Seite der Ebene, die der Richtung des kleinen Pfeils in der Ebenendarstellung *gegenüber* liegt.
- b.) Die Sicherheitseingangsfunktion *Reduzierter Modus* ist konfiguriert und die Eingangssignale sind niedrig (siehe 10.13 für weitere Details).

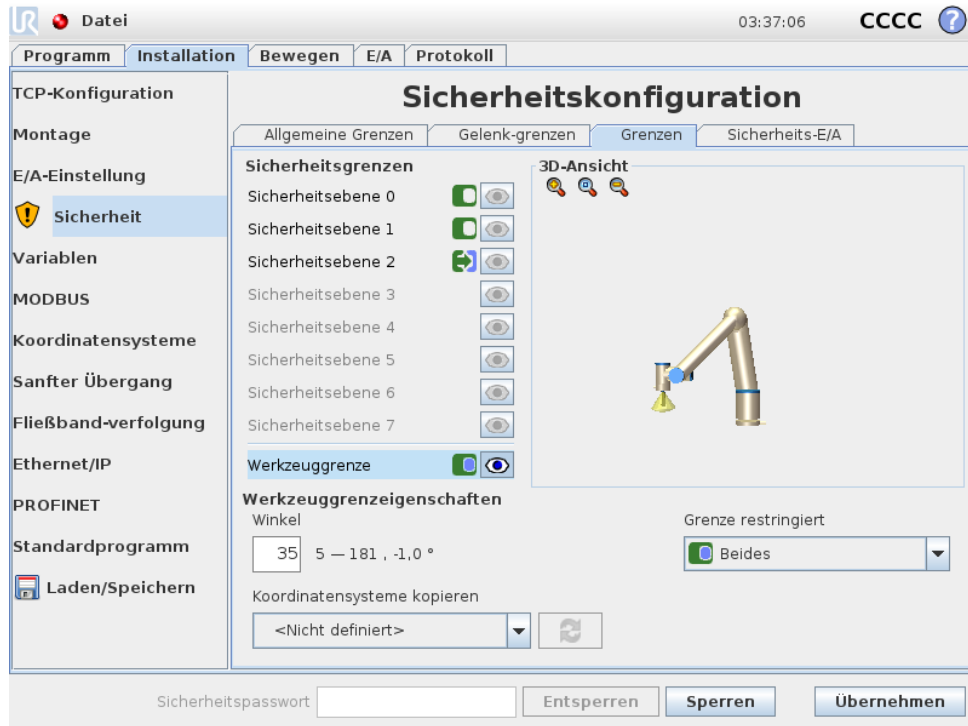
Wenn keiner der oben genannten Fälle mehr vorliegt, wechselt das Sicherheitssystem in den *Normalen* Modus zurück.

Wenn der Übergang vom *Normalen* in den *Reduzierten* Modus durch eine Bewegung durch eine *Reduzierten Modus auslösen-Ebene* ausgelöst wird, wechselt das System von den Grenzwerten des *Normalen* Modus zu denen des *Reduzierten* Modus. Sobald sich der TCP des Roboters 20 mm oder näher an der *Reduzierten Modus auslösen-Ebene* (jedoch noch auf der *Normalen* Modus-Seite) befindet, werden die toleranteren Grenzen des *Normalen* und *Reduzierten* Modus für die Grenzwerte angewendet. Sobald sich der TCP des Roboters auf die *Reduzierten Modus auslösen-Ebene* bewegt, werden die Grenzen des *Normalen* Modus deaktiviert und die des *Reduzierten* Modus aktiviert.

Wenn ein Übergang vom *Reduzierten* in den *Normalen* Modus durch eine Bewegung in eine *Reduzierten Modus auslösen-Ebene* ausgelöst wird, wechselt das System von den Grenzwerten des *Reduzierten* Modus zu denen des *Normalen* Modus. Sobald sich der TCP des Roboters in die *Reduzierten Modus auslösen-Ebene* bewegt, werden die toleranteren Grenzen des *Normalen* und *Reduzierten* Modus für die Grenzwerte angewendet. Sobald sich der TCP des Roboters 20 mm oder weiter von der *Reduzierten Modus auslösen-Ebene* entfernt (auf der *Normalen* Modus-Seite) befindet, sind die Grenzen des *Reduzierten* Modus nicht mehr aktiv und die des *Normalen* Modus werden aktiviert.

Wenn die voraussichtliche Bahn des TCP durch eine *Reduzierten Modus auslösen-* Ebene verläuft, wird der Roboterarm bereits vor dem Eindringen in die Ebene abgebremst, falls er sonst die Grenzwerte der Gelenkgeschwindigkeit, der Werkzeuggeschwindigkeit oder des Moments dieser Ebene überschreiten würde. Beachten Sie, dass dieser Abbremsvorgang aufgrund der geringeren Grenzwerte im *Reduzierten* Modus nur beim Übergang vom *Normalen* in den *Reduzierten* Modus möglich ist.

### 10.12.4 Werkzeuggrenzkonfiguration






Das Feld *Eigenschaften* der *Werkzeuggrenze* im unteren Bereich der Registerkarte definiert ein Limit für die Ausrichtung des Roboterwerkzeugs, das sich aus der gewünschten Werkzeugausrichtung und einem Wert für die maximal zulässige Abweichung von dieser Ausrichtung zusammensetzt.

**Abweichung** Das Textfeld *Abweichung* zeigt den Wert für die maximal zulässige Abweichung der Ausrichtung des Roboterwerkzeugs von der gewünschten Position. Ändern Sie diesen Wert, indem Sie das entsprechende Textfeld antippen und den neuen Wert eingeben.

Der zulässige Wertebereich zusammen mit der Toleranz und Einheit der Abweichung sind neben dem Feld aufgelistet.

**Kopierfunktion** Die gewünschte Ausrichtung des Roboterwerkzeugs wird mithilfe einer Funktion (siehe 13.12) von der aktuellen Roboterinstallation spezifiziert. Die z-Achse der ausgewählten Funktion wird als Vektor für die gewünschte Werkzeugausrichtung für dieses Limit verwendet. Nutzen Sie die Dropdown-Box auf der unteren linken Seite des Felds *Eigenschaften* der *Werkzeuggrenze*, um eine Funktion auszuwählen. Nur die Punkte und die Ebenentypenfunktionen sind verfügbar. Durch Auswahl des Elements *<Nicht definiert>* wird die Konfiguration der Ebene gelöscht.

Copyright ©2009–2018 by Universal Robots A/S. Alle Rechte vorbehalten.

Es ist zu beachten, dass bei der Konfiguration eines Limits durch Auswahl einer Funktion die Ausrichtungsinformationen nur in das Limit *kopiert* werden; das Limit ist *nicht* mit dieser Funktion verknüpft. Das bedeutet, dass wenn Änderungen an der Position und Ausrichtung einer Funktion, die zur Konfiguration des Limits genutzt wurde, gemacht wurden, das Limit nicht automatisch aktualisiert wird. Wenn sich die Funktion verändert hat, wird dies durch ein  Symbol angezeigt, das sich über dem Funktionseinsteller befindet. Klicken Sie die Taste  neben der Auswahlfunktion, um das Limit mit der aktuellen Ausrichtung der Funktion zu aktualisieren. Das  Symbol wird auch angezeigt, wenn die ausgewählte Funktion von der Installation gelöscht wurde.

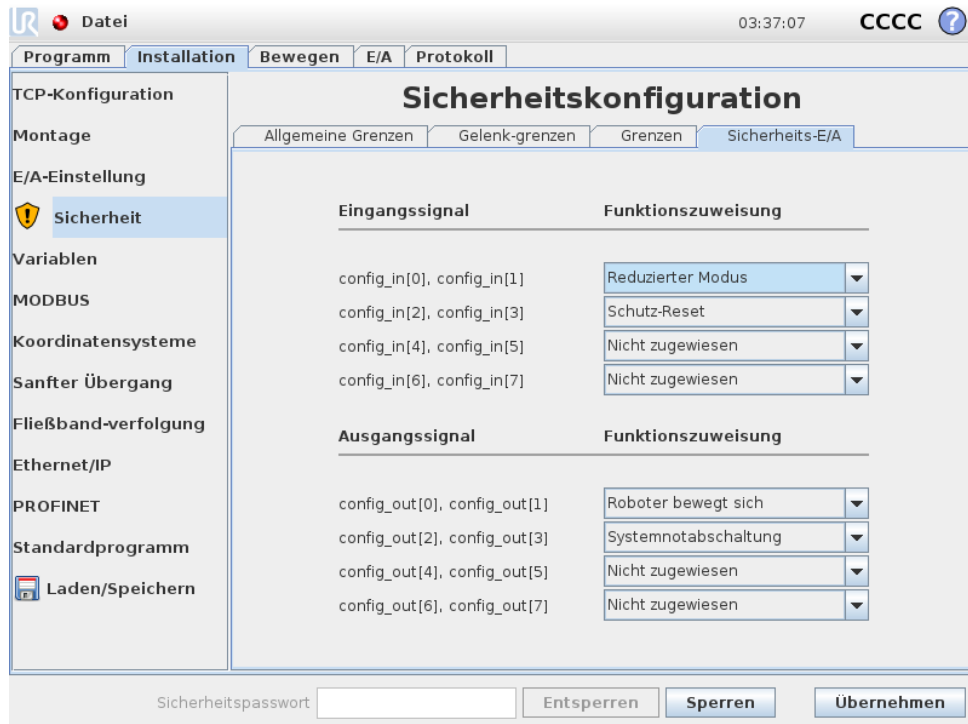
**Sicherheitsmodus** Mit dem Dropdown-Menü auf der rechten Seite des Felds *Werkzeuggrenzeigenschaften* wird der *Sicherheitsmodus* der Werkzeugausrichtungsgrenze ausgewählt. Die verfügbaren Optionen sind:

<input type="checkbox"/> Deaktiviert	Das Werkzeuggrenzlimit ist niemals aktiv.
<input checked="" type="checkbox"/> Normal	Das Werkzeuggrenzlimit ist aktiv, wenn sich das Sicherheitssystem im <i>Normalen</i> Modus befindet.
<input type="checkbox"/> Reduziert	Das Werkzeuggrenzlimit ist aktiv, wenn sich das Sicherheitssystem im <i>Reduzierten</i> Modus befindet.
<input checked="" type="checkbox"/> Normal & Reduziert	Das Werkzeuggrenzlimit ist aktiv, wenn sich das Sicherheitssystem im <i>Normalen</i> oder im <i>Reduzierten</i> Modus befindet.

Der ausgewählte *Sicherheitsmodus* wird durch ein Symbol im zugehörigen Eintrag im Feld *Sicherheitsgrenzen* angezeigt. Ist der *Sicherheitsmodus* auf *Deaktiviert* eingestellt, wird kein Symbol angezeigt.

**Wirkung** Die Programmausführung wird abgebrochen, wenn die Abweichung der Werkzeugausrichtung die eingegebene maximale Abweichung abzüglich der Toleranz überschreitet (siehe 10.4) und der Roboter sich weiter entlang der voraussichtlichen Bahn fortbewegt. Beachten Sie, dass das Minuszeichen vor den Toleranzwerten lediglich angibt, dass die Toleranz vom eingegebenen Wert abgezogen wird. Das Sicherheitssystem führt einen Stopp der Kategorie 0 durch, falls die Abweichung der Werkzeugausrichtung die Grenze (ohne Toleranz) überschreitet.

## 10.13 Sicherheits-E/A



Dieser Bildschirm definiert die *Sicherheitsfunktionen* für konfigurierbare Ein- und Ausgänge (E/A). Die E/As sind zwischen den Eingängen und Ausgängen aufgeteilt und werden paarweise so zusammengefasst, dass jede Funktion eine Kategorie<sup>2</sup> 3 und PLd E/A umfasst.

Jede *Sicherheitsfunktion* kann jeweils nur ein E/A-Paar steuern. Wenn Sie versuchen, dieselbe Sicherheitsfunktion ein zweites Mal auszuwählen, wird sie aus dem ersten Paar der zuvor definierten E/A entfernt. Es bestehen 5 *Sicherheitsfunktionen* für Eingangssignale und 5 für Ausgangssignale.

Hinweis: Die Anwendung einer Sicherheitsfunktion bei einer Pin-Kontaktreihe überschreibt alle E/A-Aktionen für Pins in der E/A-Konfiguration (siehe 13.8).

### 10.13.1 Eingangssignale

Für Eingangssignale können die folgenden *Sicherheitsfunktionen* ausgewählt werden: System-Notabschaltung, Reduzierter Modus, Schutz-Reset, 3-Stellungs-Zustimmschalter und Betriebsart.

**System-Notabschaltung** Wenn konfiguriert, ermöglicht dies einen weiteren Not-Aus-Schalter neben dem Not-Aus-Schalter auf dem Teach Pendant. Diese Funktion erfordert die Verwendung eines ISO 13850-kompatiblen Geräts.

**Reduzierter Modus** Alle Sicherheitsgrenzen haben zwei Modi, in denen sie angewandt werden können: *Normaler Modus* (gibt die standardmäßige Sicherheitskonfiguration an) und *Reduzierter Modus* (siehe 10.6 für weitere Details). Wenn diese Eingangs-Sicherheitsfunktion

<sup>2</sup>(gemäß ISO 13849-1, siehe Glossar für weitere Details).

gewählt ist, bewirkt ein niedriges Signal an die Eingänge, dass das Sicherheitssystem in den *Reduzierten* Modus übergeht. Wenn nötig, bremst der Roboterarm anschließend ab, um die Grenzen des *Reduzierten* Modus einzuhalten. Sollte der Roboterarm eine der Grenzen des *Reduzierten* Modus weiterhin überschreiten, führt er einen Stopp der Kategorie 0 aus. Der Wechsel zurück in den *normalen* Modus geschieht auf gleiche Weise. Beachten Sie, dass Sicherheitsebenen auch einen Wechsel in den *Reduzierten* Modus bewirken können (siehe 10.12.3 für weitere Details).



**Schutz-Reset** Wenn Schutzstopp in den Sicherheits-E/A verkabelt ist, kann dieser Eingang verwendet werden, um sicherzustellen, dass der Schutzstopp-Status beibehalten wird, bis ein Reset ausgelöst wird. Der Roboterarm bewegt sich solange nicht, wie er sich im Schutzstopp-Status befindet.



**WARNUNG:**

Standardmäßig ist die Schutz-Reset-Funktion für die Eingangspins 0 und 1 konfiguriert. Vollständiges Deaktivieren bedeutet, dass der Roboterarm per Schutzstopp nicht länger deaktiviert bleibt, sobald der Schutzstopp-Eingabewert ansteigt. Mit anderen Worten bedeutet das, dass die Schutz-Reset-Eingänge SI0 und SI1 ohne eine Schutzstopp-Eingabe ermitteln, ob der Schutzstopp-Status aktiv ist oder nicht (siehe Hardware-Installationshandbuch).

**3-Stellungs-Zustimmschalter und Betriebsart** Diese ermöglichen die Verwendung eines 3-Punkt-Schalters als zusätzliche Schutzmaßnahme während der Installation und Programmierung des Roboters. Nachdem der 3-Stellungs-Zustimmschalter-Eingang konfiguriert wurde, befindet sich der Roboter entweder im „Aktivbetrieb“ oder „Programmiermodus“. Ein Symbol in der rechten oberen Ecke zeigt die aktuelle Betriebsart an:

-  **Aktivbetrieb:** Der Roboter kann nur vordefinierte Aufgaben ausführen. Der „Move“-Tab und der Freedrive-Modus stehen nicht zur Verfügung.
-  **Programmiermodus:** Die Einschränkungen aus dem *Aktivbetrieb* sind außer Kraft. Wenn jedoch der Eingang am 3-Stellungs-Zustimmschalter ein LOW-Signal hat, löst der Schutzstopp des Roboters aus. Zudem wird der Geschwindigkeitsregler auf einen Anfangswert gesetzt, der 250 mm/s entspricht und der schrittweise erhöht werden kann, um eine höhere Geschwindigkeit zu erreichen. Der Geschwindigkeitsregler wird stets auf den niedrigen Wert zurückgesetzt, wenn der 3-Stellungs-Zustimmschalter-Eingang von LOW auf HIGH geht.

Es gibt zwei Verfahren zur Konfigurierung der Wahl der Betriebsart:

1. Um die Betriebsart mit Hilfe eines externen Moduswahlgeräts auszuwählen, konfigurieren Sie den Betriebsart-Eingang. Die Option, mit der er konfiguriert wird, erscheint im Dropdown-Menü, sobald der 3-Stellungs-Zustimmschalter-Eingang eingestellt ist. Der Roboter befindet sich im *Aktivbetrieb* wenn der Betriebsart-Eingang LOW ist und im *Programmiermodus*, wenn er HIGH ist.

- Um die Betriebsart von Polyscope auszuwählen, darf nur der 3-Stellungs-Zustimmschalter-Eingang konfiguriert sein und in der Sicherheitskonfiguration angewendet werden. In diesem Fall ist der Standardmodus *Aktivbetrieb*. Um zum *Programmiermodus* zu wechseln, wählen Sie die Taste „Roboter programmieren“ auf dem Begrüßungsbildschirm. Um zur Betriebsart *Aktivbetrieb* zurückzukehren, beenden Sie einfach die Anzeige „Roboter programmieren“.

**HINWEIS:**

- Nachdem die Sicherheitskonfiguration für E/A mit dem aktivierten 3-Stellungs-Zustimmschalter bestätigt wurde, wird automatisch der Begrüßungsbildschirm angezeigt. Der Startbildschirm wird ebenso automatisch angezeigt, wenn sich die Betriebsart von *Programmierung* zu *Aktivbetrieb* ändert.
- Der physische Modusauswahlschalter, wenn verwendet, muss sich für die Auswahl vollständig an den Standard ISO 10218-1: Artikel 5.7.1 halten.
- Der 3-Stufenschalter inkl. Verhalten, Schaltcharakteristiken und Betrieb muss in allen Punkten ISO 10218-1: Artikel 5.8.3 für Freigabeschalter erfüllen.

---

**10.13.2 Ausgangssignale**

Für Ausgangssignale können die folgenden *Sicherheitsfunktionen* angewandt werden: Alle Signale werden wieder LOW, wenn der Status, der das HIGH-Signal ausgelöst hat, beendet ist:

**System-Notabschaltung** Ein LOW-Signal wird nur dann ausgegeben, nachdem das Sicherheitssystem in einen Schutzstopp-Status durch Eingang Roboter-Notabschaltung oder Not-Aus-Schalter überführt wurde. Um Blockierungen zu vermeiden, wird kein LOW-Signal ausgegeben, wenn der per Schutz-Aus-Status durch den System-Notabschaltung-Eingang ausgelöst wird.

**HINWEIS:**

Externe Maschinen, die den per Schutz-Aus-Status vom Roboter über den Ausgang System-Notabschaltung erhalten, müssen die Vorgaben der ISO 13850 erfüllen. Dies ist insbesondere bei Setups erforderlich, in denen der Roboter-Notabschaltung-Eingang an ein externes Not-Aus-Gerät angeschlossen ist. In solchen Fällen ändert sich der Ausgang System-Notabschaltung zu HIGH, wenn das externe Not-Aus-Gerät freigegeben wird. Dies bedeutet, dass der Status der Notabschaltung bei der externen Maschine ohne manuelles Eingreifen durch den Roboterbediener zurückgesetzt wird. Um die Sicherheitsnormen zu erfüllen, muss die externe Maschine für einen weiteren Betrieb manuell bedient werden.

**Roboterbewegung aktiv** Ein LOW-Signal wird ausgegeben, wenn sich der Roboterarm in einem Bewegungsstatus befindet. Befindet sich der Roboterarm in einer statischen Position, wird ein HIGH-Signal ausgegeben.

**Roboter stoppt nicht** Wurde der Roboterarm zum Stopp aufgefordert, wird ab dem Zeitpunkt der Anforderung etwas Zeit vergehen, bis der Arm anhält. Während dieser Zeit ist das Signal HIGH. Wenn sich der Roboterarm bewegt und nicht zum Stopp aufgefordert wurde oder er sich in einer gestoppten Position befindet, ist der Ausgang auf LOW geschaltet.

**Reduzierter Modus** Sendet ein LOW-Signal, wenn sich der Roboterarm im *Reduzierten Modus* befindet oder wenn der Sicherheitseingang mit einem Eingang des *Reduzierten Modus* konfiguriert ist und das Signal derzeit LOW ist. Andernfalls ist das Signal HIGH.

**Nicht-reduzierter Modus** Dies ist das Gegenstück zum oben definierten *Reduzierten Modus*.



# 11 Programmierung starten

## 11.1 Einleitung

Der Universal Robot Arm besteht aus Rohren und Gelenken. Die Gelenke und ihre üblichen Bezeichnungen sind in Abbildung 11.1 dargestellt. An der **Basis** ist der Roboter montiert und am anderen Ende (**Handgelenk 3**) ist das Roboterwerkzeug befestigt. Indem die Bewegung jedes der Gelenke koordiniert wird, kann der Roboter sein Werkzeug, abgesehen von dem Bereich direkt über und direkt unterhalb der Basis, frei umherbewegen.

PolyScope ist eine grafische Benutzeroberfläche (GUI), mit der Sie den Roboterarm und das Steuergerät bedienen, Roboterprogramme ausführen oder einfach neue Programme erstellen können.

Im folgenden Abschnitt finden Sie die ersten Schritte für den Betrieb des Roboters. Danach werden die Bildschirme und Funktionen von PolyScope detaillierter beschrieben.



### GEFAHR:

1. Das Hardware-Installationshandbuch enthält wichtige Sicherheitsinformationen, die vom Integrator von UR-Robotern gelesen und verstanden werden müssen, bevor der Roboter zum ersten Mal eingeschaltet wird.
2. Der Integrator muss die aus der Risikobewertung definierten Sicherheitskonfigurationsparameter einstellen, bevor der Roboterarm zum ersten Mal eingeschaltet wird, siehe Kapitel 10.

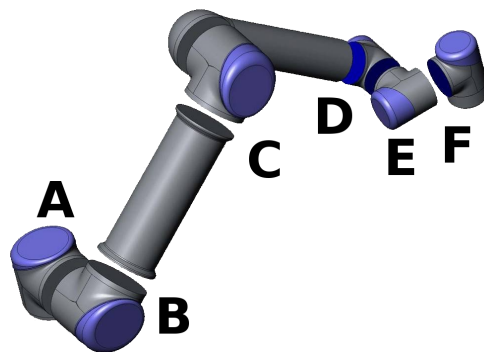


Abbildung 11.1: Gelenke des Roboters. A: *Basis*, B: *Schulter*, C: *Ellbogen* und D, E, F: *Handgelenk 1, 2, 3*

## 11.2 Erste Schritte

Vor der Verwendung von PolyScope müssen der Roboterarm und der Controller installiert und der Controller eingeschaltet werden.

### 11.2.1 Installation des Roboterarms und des Controllers

Um den Roboterarm und den Controller zu installieren, gehen Sie wie folgt vor:

1. Packen Sie den Roboter und den Controller aus.
2. Montieren Sie den Roboter auf einer stabilen, vibrationsfreien Oberfläche.
3. Positionieren Sie den Controller auf dessen Fuß.
4. Verbinden Sie den Roboter und den Controller durch Anschluss des Roboterkabels.
5. Stecken Sie den Netzstecker des Controllers ein.



**WARNUNG:**

Kippgefahr. Wird der Roboter nicht sicher auf einer stabilen Oberfläche platziert, kann er umfallen und Verletzungen verursachen.

Detaillierte Installationsanweisungen finden Sie hier: Hardware-Installationshandbuch. Beachten Sie, dass eine Risikobewertung erforderlich ist, bevor der Roboterarm für Arbeiten eingesetzt wird.

### 11.2.2 Ein- und Ausschalten des Controllers

Der Controller wird mit Hilfe der Power-Taste eingeschaltet. Diese befindet sich an der Vorderseite des Feldes mit dem Touch-Screen. Dieses Feld wird in der Regel *Teach Pendant* genannt. Wenn der Controller eingeschaltet ist, erscheint Text vom zugrundeliegenden Betriebssystem auf dem Touch-Screen. Nach etwa einer Minute erscheinen einige Schaltflächen auf dem Bildschirm und ein Pop-up-Fenster leitet den Benutzer zum Initialisierungsbildschirm (siehe 11.5).

Um den Controller auszuschalten, drücken Sie den grünen Ein/Aus-Schalter auf dem Bildschirm oder verwenden Sie die Schaltfläche `Roboter abschalten` auf dem Startbildschirm (siehe 11.4).



**WARNUNG:**

Eine Abschaltung durch Herausziehen des Netzkabels aus der Steckdose kann das Dateisystem des Roboters beschädigen und zu einer Fehlfunktion des Roboters führen.

### 11.2.3 Ein- und Ausschalten des Roboterarms

Der Roboterarm kann eingeschaltet werden, wenn der Controller eingeschaltet und kein Not-Aus-Schalter betätigt ist. Der Roboterarm wird über den Initialisierungsbildschirm (siehe 11.5) eingeschaltet, indem die Schaltfläche `Ein` auf dem Bildschirm und anschließend die Schaltfläche `Start` betätigt wird. Der Start eines Roboters ist mit einem Geräusch und leichter Bewegung während der Freigabe der Bremsen verbunden.

Die Stromversorgung zum Roboterarm kann über die Schaltfläche `AUS` auf dem Initialisierungsbildschirm unterbrochen werden. Der Roboter schaltet sich automatisch aus, wenn der Controller ausgeschaltet wird.

### 11.2.4 Schnellstart

Um den Roboter schnell zu starten, nachdem er installiert wurde, befolgen Sie die folgenden Schritte:

1. Betätigen Sie den Not-Aus-Schalter an der Vorderseite des Teach Pendant.
2. Drücken Sie den Ein/Aus-Schalter am Teach Pendant.
3. Warten Sie eine Minute, während das System hochfährt und Text auf dem Touch-Screen angezeigt wird.
4. Wenn das System bereit ist, erscheint ein Pop-up-Fenster auf dem Touch-Screen, das Ihnen mitteilt, dass der Roboter initialisiert werden muss.
5. Tippen Sie auf die Schaltfläche im Pop-up-Dialog. Sie werden nun zum Initialisierungsbildschirm geleitet.
6. Warten Sie, bis der Dialog `Bestätigung der angewandten Sicherheitskonfiguration` erscheint und drücken Sie die Taste `Sicherheitskonfiguration bestätigen`. Eine erste Reihe von sicherheitsrelevanten Parametern findet nun Anwendung, die auf der Grundlage einer Risikobewertung angepasst werden müssen.
7. Entriegeln Sie den Not-Aus-Schalter. Der Roboterzustand ändert sich nun von `Notabschaltung` zu `Roboterspannung Aus`.
8. Verlassen Sie die Reichweite (den Wirkungsbereich) des Roboters.
9. Tippen Sie auf die Schaltfläche `Ein` auf dem Touchscreen. Warten Sie einige Sekunden bis sich der Roboterzustand zu `Leerlauf` ändert.
10. Prüfen Sie, dass die Nutzlast und die gewählte Montage korrekt sind. Sie werden benachrichtigt, wenn die anhand von Sensordaten erkannte Montage nicht der ausgewählten Montage entspricht.
11. Tippen Sie auf die Schaltfläche `Start` auf dem Touchscreen. Der Roboter gibt nun ein Geräusch von sich und bewegt sich ein wenig, während er die Bremsen entriegelt.
12. Berühren Sie die Schaltfläche `OK`, wodurch der `Startbildschirm` erscheint.

### 11.2.5 Das erste Programm

Ein Programm ist eine Liste von Befehlen, die dem Roboter Anweisungen erteilt. PolyScope ermöglicht Personen mit nur wenig Programmiererfahrung, den Roboter zu programmieren. Für die meisten Aufgaben erfolgt die Programmierung ausschließlich mit dem Touch-Screen, ohne dabei kryptische Befehle eingeben zu müssen.

Die Werkzeugbewegung ist ein Teil des Roboterprogramms, das dem Roboterarm die Bewegungen beibringt. In PolyScope werden Werkzeugbewegungen mithilfe einer Reihe von **Wegpunkten** festgelegt. Die kombinierten Wegpunkte bilden einen Weg, dem der Roboterarm folgt. Ein Wegpunkt kann vorgegeben werden, über die Registerkarte `Move`, indem man den Roboter in eine bestimmte Position bewegt (Programmierung), oder indem man diesen durch die Software berechnen lässt. Klicken Sie auf der Registerkarte `Move` (siehe 13.1), um den Roboterarm in eine gewünschte Position zu bewegen, oder programmieren Sie die Position, indem

Sie den Arm in die richtige Position ziehen, während Sie die Taste `Freedrive` hinter Teach Pendant gedrückt halten.

Neben der Bewegung entlang verschiedener Wegpunkte kann das Programm an bestimmten Stellen entlang des Weges des Roboters E-/A-Signale an andere Maschinen senden und aufgrund von Variablen und E-/A-Signalen Befehle ausführen, beispielsweise **if... then** und **loop**.

Das Folgende ist ein einfaches Programm, mit dem ein Roboterarm, der gestartet wurde, zwischen zwei Wegpunkten bewegt werden kann.

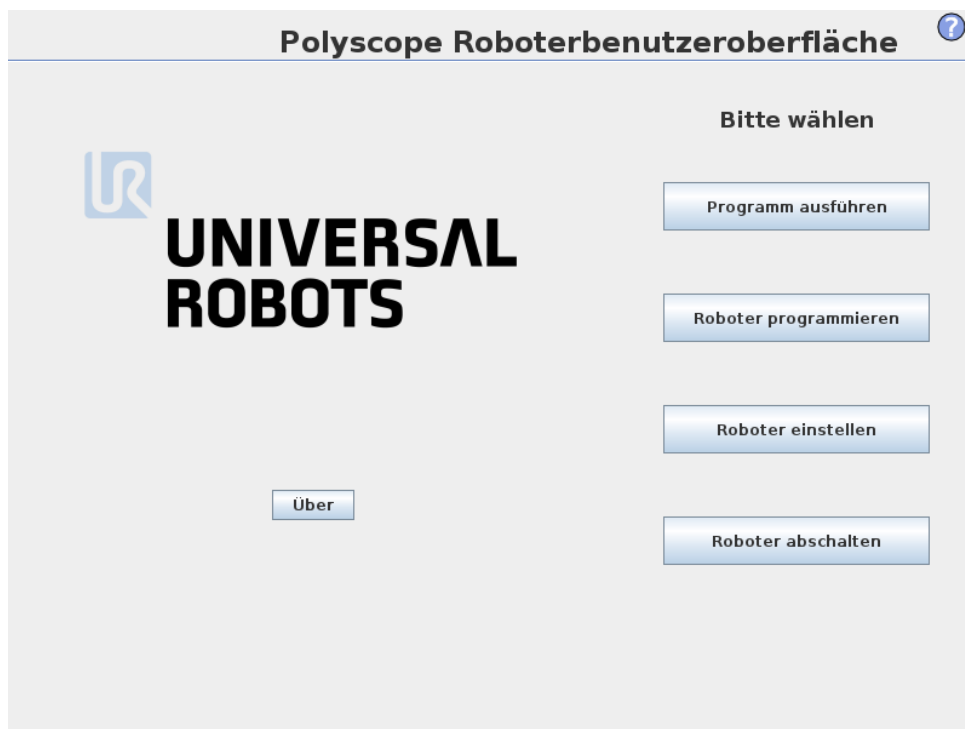
1. Tippen Sie auf die Schaltfläche `Roboter programmieren` und wählen Sie `Neues Programm`.
2. Berühren Sie die Schaltfläche `Weiter` (unten rechts), so dass die `<leere>` Ziele in der Baumstruktur auf der linken Bildschirmseite gewählt wird.
3. Öffnen Sie den Tab `Struktur`.
4. Tippen Sie auf die Schaltfläche `Move`.
5. Öffnen Sie den Tab `Befehl`.
6. Tippen Sie auf die Schaltfläche `Weiter`, um die Wegpunkt-Einstellungen zu öffnen.
7. Berühren Sie die Schaltfläche `Wegpunkt festlegen` neben dem Symbol „?“.
8. Bewegen Sie den Roboter im `Move`-Bildschirm, indem Sie die verschiedenen blauen Pfeile drücken oder indem Sie die Taste `Freedrive` auf der Rückseite des Teach Pendants gedrückt halten, während Sie den Arm des Roboters von Hand bewegen.
9. Drücken Sie `OK`.
10. Drücken Sie `Wegpunkt davor hinzufügen`.
11. Berühren Sie die Schaltfläche `Wegpunkt festlegen` neben dem Symbol „?“.
12. Bewegen Sie den Roboter im `Move`-Bildschirm, indem Sie die verschiedenen blauen Pfeile drücken oder indem Sie die `Freedrive`-Taste gedrückt halten, während Sie den Arm des Roboters von Hand bewegen.
13. Drücken Sie `OK`.
14. Ihr Programm ist fertig. Der Roboter wird sich zwischen den beiden Wegpunkten bewegen, wenn Sie das Symbol „Abspielen“ drücken. Treten Sie zurück und halten Sie eine Hand an der Notabschaltungstaste. Drücken Sie anschließend auf „Abspielen“.
15. Herzlichen Glückwunsch! Sie haben Ihr erstes Roboterprogramm erstellt, welches den Roboter zwischen zwei vorgegebenen Wegpunkten bewegt.

**WARNUNG:**

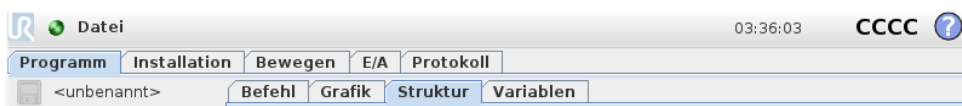
1. Bewegen Sie den Roboter nicht gegen sich selbst oder andere Dinge, da dies den Roboter beschädigen kann.
2. Halten Sie Ihren Kopf und Oberkörper vom Wirkungsbereich des Roboters fern. Halten Sie Finger fern von Bereichen, in denen sie sich verfangen können.
3. Dies ist nur eine Schnellstartanleitung, um zu demonstrieren, wie einfach es ist, einen UR Roboter zu verwenden. Dabei wurde von einer gefahrungsfreien Umgebung und einem sehr vorsichtigen Benutzer ausgegangen. Erhöhen Sie nicht die Geschwindigkeit oder die Beschleunigung über die Standardwerte hinaus. Führen Sie immer eine Risikobewertung durch, bevor Sie den Roboter in Betrieb nehmen.

## 11.3 PolyScope-Programmierschnittstelle

PolyScope läuft auf dem Touch-Screen der Control-Box.



Die oben stehende Abbildung zeigt den Startbildschirm. Die bläulichen Bereiche des Bildschirms sind Schaltflächen, die mit dem Finger oder der Rückseite eines Stiftes betätigt werden können. PolyScope verfügt über eine hierarchische Bildschirmstruktur. In der Programmierumgebung sind die Bildschirme für einen leichten Zugang in *Tabs* (Registerkarten) angeordnet.

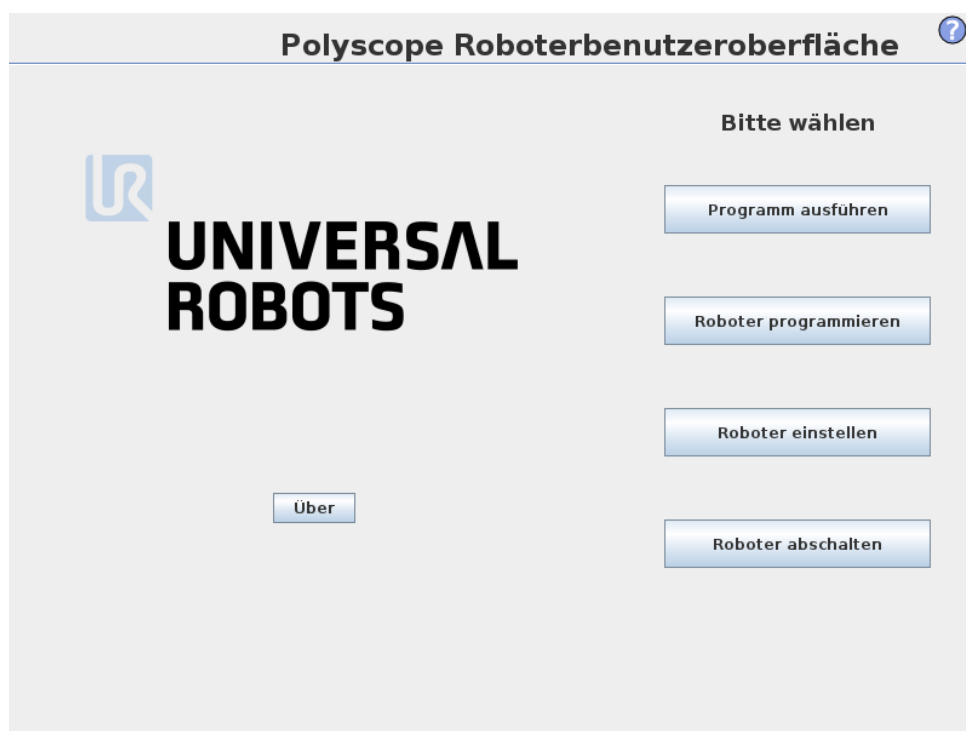


In diesem Beispiel ist der Tab **Programm** auf der obersten Ebene und darunter der Tab **Struktur** ausgewählt. Der Tab **Programm** enthält Informationen zum aktuell geladenen Programm. Wird der Tab **Move** ausgewählt, so wechselt der Bildschirm zum **Move**-Bildschirm, von wo aus der Roboter bewegt werden kann. Durch die Auswahl des Tab **E/A** wird gleichzeitig der aktuelle Zustand der elektrischen Ein- und Ausgänge überwacht und geändert.

Der Anschluss einer Maus oder einer Tastatur an das Steuergerät bzw. das Teach Pendant ist möglich, wird jedoch nicht benötigt. Fast alle Textfelder sind durch Berührung aktivierbar, so dass eine Berührung der Felder einen Nummernblock oder eine Tastatur auf dem Bildschirm anzeigt.

Die verschiedenen Bildschirme von PolyScope werden in den folgenden Abschnitten beschrieben.

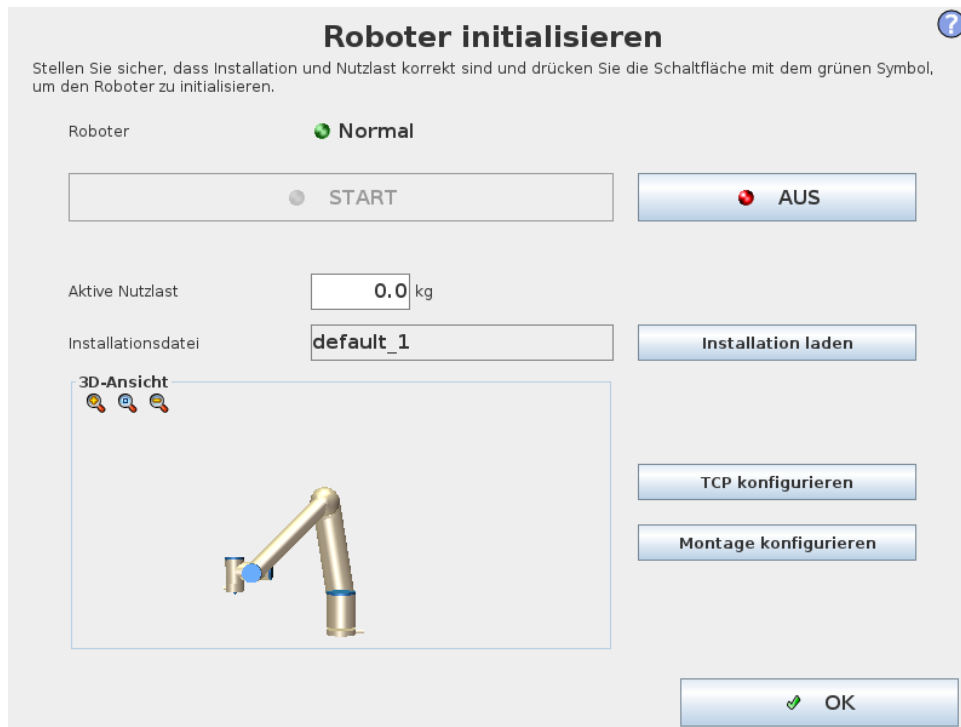
## 11.4 Startbildschirm



Nach dem Starten des Steuercomputers wird der Startbildschirm angezeigt. Der Bildschirm bietet die folgenden Optionen:

- **Programm ausführen:** Vorhandenes Programm auswählen und ausführen. Dies ist der einfachste Weg, den Roboterarm und das Steuergerät zu bedienen.
- **Roboter programmieren:** Ändern Sie ein Programm oder erstellen Sie ein neues Programm.
- **Roboter einrichten:** Ändern der Sprache, Passwörter, Software-Upgrade usw.
- **Roboter abschalten:** Schaltet den Roboterarm und das Steuergerät aus.
- **Über:** Enthält Details zu Software-Versionen, Hostname, IP-Adresse, Seriennummer und rechtliche Informationen.

## 11.5 Initialisierungsbildschirm



Mit diesem Bildschirm steuern Sie die Initialisierung des Roboterarms.

### Roboterarm-Statusanzeige

Diese Status-LED zeigt den aktuellen Status des Roboterarms an:

- Eine helle, rote LED zeigt an, dass sich der Roboterarm derzeit im Stoppzustand befindet, wofür es mehrere Gründe geben kann.
- Eine helle, gelbe LED zeigt an, dass der Roboterarm eingeschaltet ist, jedoch nicht für den normalen Betrieb bereit ist.
- Eine grüne LED zeigt an, dass der Roboterarm eingeschaltet und für den normalen Betrieb bereit ist.

Der neben der LED erscheinende Text beschreibt den aktuellen Status des Roboterarms näher.

### Aktive Nutzlast und Installation

Wenn der Roboterarm eingeschaltet ist, wird die Nutzlastmasse, die vom Steuergerät beim Bedienen des Roboterarms verwendet wird, in dem kleinen, weißen Textfeld angezeigt. Dieser Wert kann durch Tippen auf das Textfeld und Eingabe eines neuen Werts geändert werden.

Hinweis: Das Festlegen dieses Werts ändert nicht die Nutzlast in der Installation des Roboters (siehe 13.6). Nur die vom Steuergerät verwendete Nutzlast wird festgelegt.

Gleichermaßen wird der Name der aktuell geladenen Installationsdatei in dem grauen Textfeld angezeigt. Eine andere Installation kann durch Tippen auf das Textfeld oder mithilfe der danebenliegenden Taste **Laden** geladen werden. Alternativ kann die geladene Installation mithilfe der Tasten neben der 3D-Ansicht im unteren Bereich des Bildschirms angepasst werden.

Vor dem Starten des Roboterarms ist es sehr wichtig, zu verifizieren, dass sowohl die aktive Nutzlast als auch die aktive Installation zu der Situation gehören, in der sich der Roboterarm derzeit befindet.

## Initialisierung des Roboterarms



### GEFAHR:

Stellen Sie stets sicher, dass die tatsächliche Nutzlast und Installation korrekt ist, bevor Sie den Roboterarm starten. Wenn diese Einstellungen falsch sind, funktionieren der Roboter und das Steuergerät nicht korrekt und können eine Gefährdung für Menschen oder Geräte in ihrem Umfeld darstellen.



### VORSICHT:

Besondere Aufmerksamkeit sollte der Vermeidung jeglicher Berührung des Roboterarms mit einem Hindernis oder Tisch gelten, da ein Gelenkgetriebe beschädigt werden kann, wenn der Roboterarm in ein Hindernis gesteuert wird.

Die große Taste mit dem grünen Symbol dient zur Durchführung der eigentlichen Initialisierung des Roboterarms. Der Text darauf und die Aktion, die sie ausführt, verändern sich je nach Lage, in welcher sich der Roboterarm aktuell befindet.

- Nachdem der Controller-PC hochgefahren ist, muss die Taste einmal angetippt werden, um den Roboterarm einzuschalten. Der Roboterarm-Status wechselt dann zu **Power on** und anschließend in den **Leerlauf**. Bitte beachten Sie, dass der Roboterarm nicht eingeschaltet werden kann, wenn ein Not-Aus vorliegt - daher wird die Taste deaktiviert.
- Ist der Roboterarm im **Leerlauf**, muss die Taste noch einmal angetippt werden, um den Roboterarm zu starten. Nun werden die Sensordaten hinsichtlich der konfigurierten Aufstellung des Roboterarms geprüft. Wird eine fehlende Übereinstimmung entdeckt (mit einer Toleranz von 30°), wird die Taste deaktiviert und unter ihr eine Fehlermeldung angezeigt. Ist die Montageprüfung bestanden, werden durch Antippen der Taste alle Gelenkbremsen gelöst und der Roboterarm ist bereit für den normalen Betrieb. Bitte beachten Sie, dass der Roboter nun ein Geräusch von sich gibt und sich ein wenig bewegt, während er die Bremsen entriegelt.
- Überschreitet der Roboterarm eine der Sicherheitsgrenzen nachdem er gestartet wurde, arbeitet er in einem besonderen **Wiederherstellungsmodus**. In diesem Modus wird durch Tippen auf die Taste in einen Wiederherstellungsmodus gewechselt, in dem der Roboterarm in die Sicherheitsgrenzen zurückbewegt werden kann.
- Tritt eine Störung auf, kann der Controller mithilfe der Taste neu gestartet werden.
- Falls der Controller momentan nicht läuft, kann es durch Antippen der Taste gestartet werden.

Die kleinere Taste mit dem roten Symbol dient zum Ausschalten des Roboterarms.

# 12 Bildschirm-Editoren

## 12.1 Ausdruckeditor auf dem Bildschirm



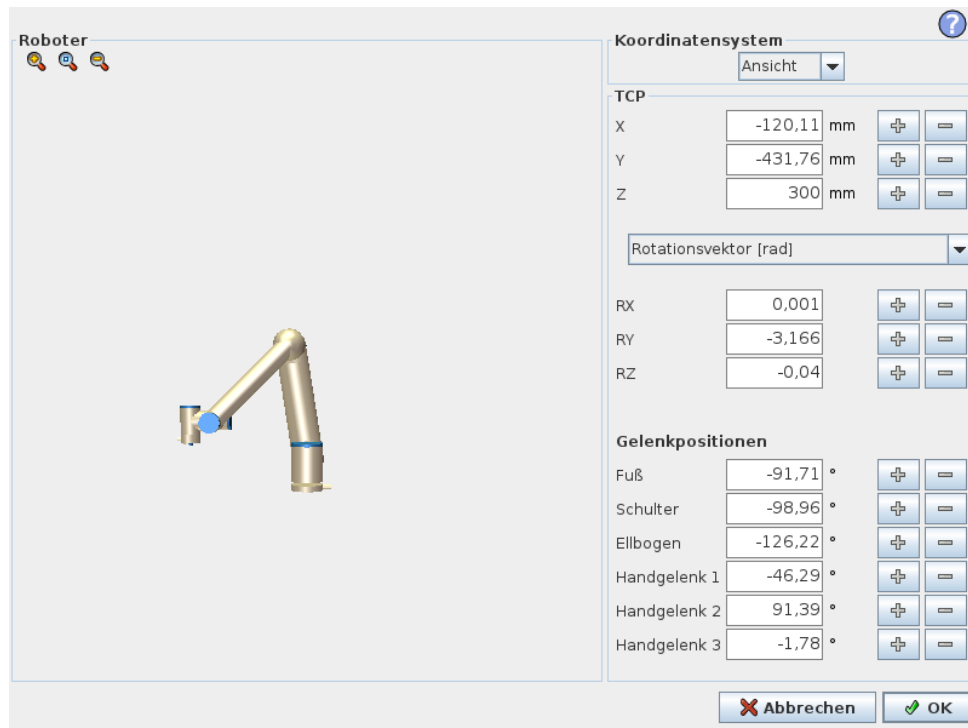
Während der Ausdruck selbst als Text bearbeitet wird, verfügt der Ausdruckeditor über eine Vielzahl von Schaltflächen und Funktionen zur Eingabe der speziellen Ausdruckssymbole, wie zum Beispiel \* zur Multiplikation und  $\leq$  für kleiner gleich. Die Tastatursymbol-Schaltfläche oben links im Bildschirm schaltet auf Textbearbeitung des Ausdrucks um. Alle definierten Variablen sind in der Variablen enthalten, während die Namen der Ein- und Ausgangsanschlüsse in den Auswahlfunktionen Eingang und Ausgang zu finden sind. Einige Sonderfunktionen finden Sie unter Funktion.

Der Ausdruck wird auf grammatische Fehler überprüft, wenn Sie die Schaltfläche OK betätigen. Mit der Schaltfläche Abbrechen verlassen Sie den Bildschirm und verwerfen alle Änderungen. Ein Ausdruck kann wie folgt aussehen:

```
digital.in[1]?=Wahr und analog.in[0]<0.5
```

## 12.2 Bearbeitungsanzeige „Pose“

Auf diesem Bildschirm können Sie die Zielpositionen der Gelenke oder eine Zielpose (Position und Ausrichtung) des Roboterwerkzeugs festlegen. Diese Anzeige ist „offline“ und steuert den Roboterarm nicht direkt.



## Roboter

Die aktuelle Position des Roboterarms und die festgelegte neue Zielposition werden in 3D-Grafiken angezeigt. Die 3D-Zeichnung des Roboterarms zeigt die aktuelle Position des Roboterarms an, während der „Schatten“ des Roboterarms die Zielposition des Roboterarms angibt, die durch die festgelegten Werte auf der rechten Bildschirmseite gesteuert wird. Betätigen Sie die Lupensymbole, um hinein-/herauszuzoomen oder ziehen Sie einen Finger darüber, um die Ansicht zu ändern.

Wenn die spezifizierte Zielposition des Roboter-TCP einer Sicherheits- oder Auslöseebene nahe ist oder die Ausrichtung des Roboterwerkzeugs sich nahe am Limit der Werkzeugausrichtungsgrenze (siehe 10.12) befindet, wird eine 3D-Darstellung des Näherungslimits der Grenze angezeigt.

Sicherheitsebenen werden in Gelb und Schwarz zusammen mit einem kleinen Pfeil angezeigt, der für die Normal-Ebene steht, was angibt, auf welcher Seite der Ebene der Roboter-TCP positioniert werden darf. Auslöseebenen werden in Blau und Grün zusammen mit einem kleinen Pfeil angezeigt, der auf die Seite der Ebene zeigt, auf der die Grenzen des Modus *Normal* (siehe 10.6) aktiv sind. Das Limit der Werkzeugausrichtungsgrenze wird anhand eines sphärischen Kegels visualisiert, wobei ein Vektor die aktuelle Ausrichtung des Roboterwerkzeugs anzeigt. Das Innere des Kegels repräsentiert den zulässigen Bereich für die Werkzeugausrichtung (Vektor).

Wenn der Zielroboter-TCP sich nicht mehr in Nähe zum Limit befindet, verschwindet die 3D-Darstellung. Wenn der Ziel-TCP ein Grenzlimit überschreitet oder dem sehr nahe ist, ändert sich die Limitanzeige zu rot.

## Funktion und Werkzeugposition

Oben rechts auf dem Bildschirm ist der Funktionseinsteller zu finden. Er legt fest, welche Funktion des Roboterarms angesteuert wird.

Der Name des aktuell aktiven Tool Center Point (TCP) wird unterhalb des Funktionseinstellers angezeigt. Weitere Informationen zur Konfigurationen mehrerer bezeichneter TCPs finden Sie hier 13.6. Die Textfelder zeigen die vollständigen Koordinatenwerte dieses TCPs relativ zur ausgewählten Funktion an.  $x$ ,  $y$  und  $z$  steuern die Position des Werkzeugs, während  $r_x$ ,  $r_y$  und  $r_z$  die Ausrichtung des Werkzeugs koordinieren.

Verwenden Sie das Auswahlmenü über den Feldern  $r_x$ ,  $r_y$  und  $r_z$ , um die Ausrichtungsdarstellung auszuwählen. Die folgenden Typen stehen zur Verfügung:

- **Rotationvektor [rad]** Die Ausrichtung wird als *Rotationsvektor* angegeben. Die Länge der Achse entspricht dem zu drehenden Winkel in Radianten, und der Vektor selbst gibt die Achse an, um die die Drehung erfolgt. Dies ist die Standardeinstellung.
- **Rotationsvektor [°]** Die Ausrichtung wird als *Rotationsvektor* angegeben, wobei die Länge des Vektors der Rotationswinkel in Grad ist.
- **RPY [rad]** *Roll*-, *Nick*- und *Gier*-Winkel (RPY), die als Radianten angegeben werden. Die RPY-Rotationsmatrix ( $x$ -,  $y$ '-,  $z$ ' - Rotation) wird angegeben durch:

$$R_{rpy}(\gamma, \beta, \alpha) = R_Z(\alpha) \cdot R_Y(\beta) \cdot R_X(\gamma)$$

- **RPY [°]** *Roll*-, *Nick*- und *Gier*-Winkel (RPY), die in Grad angegeben werden.

Die Werte können bearbeitet werden, indem Sie auf die Koordinate klicken. Durch Klicken auf die Schaltflächen + und – rechts neben einem Feld können Sie den aktuellen Wert um einen Betrag erhöhen oder verringern. Durch Gedrückthalten einer Schaltfläche wird der Wert direkt erhöht/verringert. Je länger Sie die Schaltfläche gedrückt halten, desto mehr wird der Wert erhöht oder verringert.

---

## Gelenkpositionen

Ermöglicht die direkte Festlegung der einzelnen Gelenkpositionen. Jede Gelenkposition kann einen Wert im Bereich von  $-360^\circ$  bis  $+360^\circ$  aufweisen, wobei es sich um die *Gelenkgrenzen* handelt. Die Werte können bearbeitet werden, indem Sie auf die Gelenkposition klicken. Durch Klicken auf die Schaltflächen + und – rechts neben einem Feld können Sie den aktuellen Wert um einen Betrag erhöhen oder verringern. Durch Gedrückthalten einer Schaltfläche wird der Wert direkt erhöht/verringert. Je länger Sie die Schaltfläche gedrückt halten, desto mehr wird der Wert erhöht oder verringert.

---

## Schaltfläche „OK“

Wenn dieser Bildschirm auf dem *Move*-Tab aktiviert wurde (siehe 13.1), gelangen Sie durch Klicken auf die Schaltfläche *OK* zurück zum *Move*-Tab, auf dem sich der Roboterarm in die festgelegte Zielposition bewegt. Wenn der zuletzt festgelegte Wert eine Werkzeugkoordinate war, bewegt sich der Roboterarm mithilfe der Bewegungsart *MoveL*-Tab in die Zielposition. Im Gegensatz dazu bewegt sich der Roboterarm mithilfe der Bewegungsart *MoveJ*-Tab in die Zielposition, wenn zuletzt eine Gelenkposition festgelegt wurde. Die unterschiedlichen Bewegungsarten werden im Abschnitt 14.5 erklärt.

---

## Schaltfläche „Abbrechen“

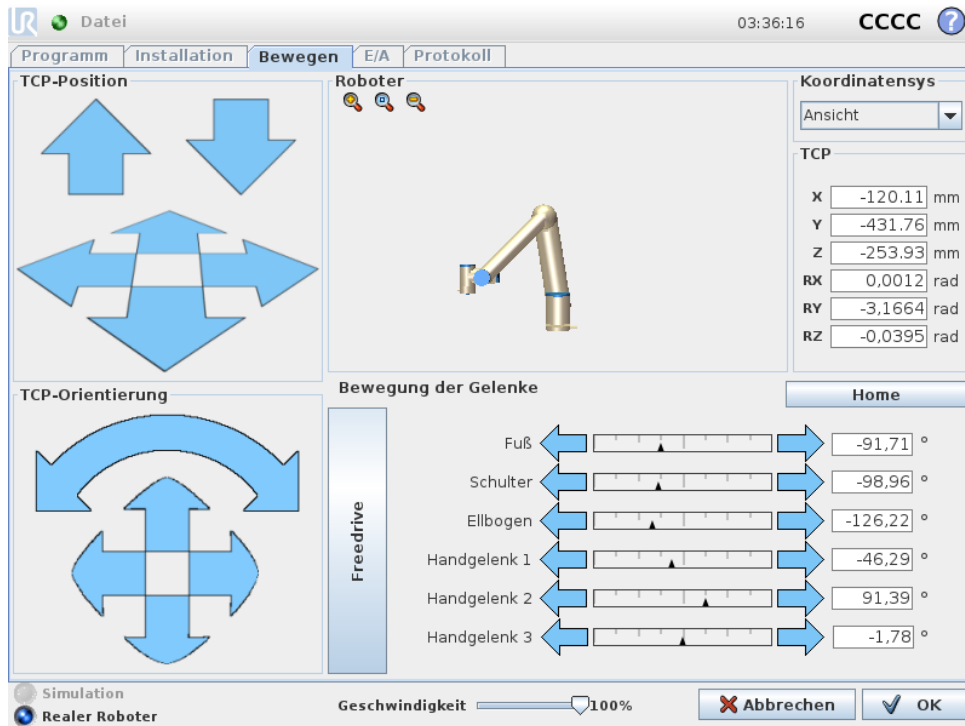
Mit der Schaltfläche „Abbrechen“ verlassen Sie den Bildschirm und verwerfen alle Änderungen.



# 13 Roboter-Steuerung

## 13.1 Move-Tab

Mit diesem Bildschirm können Sie den Roboterarm immer direkt bewegen (Joystick-Steuerung), entweder durch Versetzung/Drehung des Roboterwerkzeugs oder durch Bewegung der einzelnen Roboterarme.



### 13.1.1 Roboter

Die aktuelle Position des Roboterarms wird mit einer 3D-Grafik angezeigt. Betätigen Sie die Lupensymbole, um hinein-/herauszuzoomen oder ziehen Sie einen Finger darüber, um die Ansicht zu ändern. Um das beste Gefühl für die Steuerung des Roboterarms zu bekommen, wählen Sie die Funktion **Ansicht** und drehen Sie den Blickwinkel der 3D-Zeichnung, damit dieser Ihrer Ansicht des echten Roboterarms entspricht.

Wenn sich die aktuelle Position des Roboter-TCP einer Sicherheits- oder Auslöseebene nähert oder sich die Ausrichtung des Roboterwerkzeugs nah an einer Werkzeugausrichtungsgrenze befindet (siehe 10.12), wird eine 3D-Darstellung der angenäherten Bewegungsgrenze angezeigt. Beachten Sie, dass die Visualisierung der Grenzlimes deaktiviert wird, während der Roboter ein Programm ausführt.

Sicherheitsebenen werden in Gelb und Schwarz zusammen mit einem kleinen Pfeil angezeigt, der für die Normal-Ebene steht, was angibt, auf welcher Seite der Ebene der Roboter-TCP positioniert werden darf. Auslöseebenen werden in Blau und Grün zusammen mit einem kleinen Pfeil dargestellt, der auf die Seite der Ebene zeigt, bei der die Grenzen des Modus **Normal** aktiv sind (siehe 10.6). Das Limit der Werkzeugausrichtungsgrenze wird anhand eines

sphärischen Kegels visualisiert, wobei ein Vektor die aktuelle Ausrichtung des Roboterwerkzeugs anzeigt. Das Innere des Kegels repräsentiert den zulässigen Bereich für die Werkzeugausrichtung (Vektor).

Wenn der Roboter-TCP sich nicht mehr in Nähe zum Limit befindet, verschwindet die 3D-Darstellung. Wenn der TCP einen Grenzwert überschreitet oder dem sehr nahe ist, ändert sich die Limitanzeige zu rot.

---

### 13.1.2 Funktion und Werkzeugposition

Oben rechts auf dem Bildschirm ist der Funktionseinsteller zu finden. Er legt fest, welche Funktion des Roboterarms angesteuert wird.

Der Name des aktuell aktiven Tool Center Point (TCP) wird unterhalb des Funktionseinstellers angezeigt. Die Textfelder zeigen die vollständigen Koordinatenwerte dieses TCPs relativ zur ausgewählten Funktion an. Weitere Informationen zur Konfigurationen mehrerer benannter TCPs finden Sie hier (siehe 13.6).

Werte können manuell durch Anklicken der Koordinaten oder der Gelenkpositionen bearbeitet werden. Dies führt Sie auf den Posenbearbeitungsbildschirm (siehe 12.2), auf dem Sie eine Zielposition und die Ausrichtung des Werkzeugs oder der Zielgelenkpositionen angeben können.

---

### 13.1.3 Bewegung des Werkzeuges

- Das Gedrückthalten eines *Bewegungspfeils* (oben) bewegt die Werkzeugspitze des Roboters in die angegebene Richtung.
- Das Gedrückthalten eines *Drehungspfeils* (unten) dreht die Ausrichtung der Werkzeugspitze des Roboters in die angegebene Richtung. Der Drehpunkt ist der Werkzeugmittelpunkt (TCP), d. h. der Punkt am Ende des Roboterarms, der einen charakteristischen Punkt auf dem Roboterwerkzeug ergibt. Der TCP wird kleine blaue Kugel dargestellt.

Hinweis: Lassen Sie die Taste los, um die Bewegung jederzeit zu stoppen!

---

### 13.1.4 Bewegung der Gelenke

Ermöglicht die direkte Steuerung der einzelnen Gelenke. Jedes Gelenk kann sich von  $-360^\circ$  bis  $+360^\circ$  bewegen. Dies sind die standardmäßigen *Gelenkgrenzen*, die für jedes Gelenk durch eine horizontale Leiste dargestellt werden. Wenn ein Gelenk seine Grenze erreicht, kann es sich nicht weiter weg bewegen. Wurden die Grenzen für ein Gelenk mit einem Positionsbereich konfiguriert, der sich vom Standard (siehe 10.11) unterscheidet, wird dieser Bereich auf der horizontalen Leiste rot dargestellt.

---

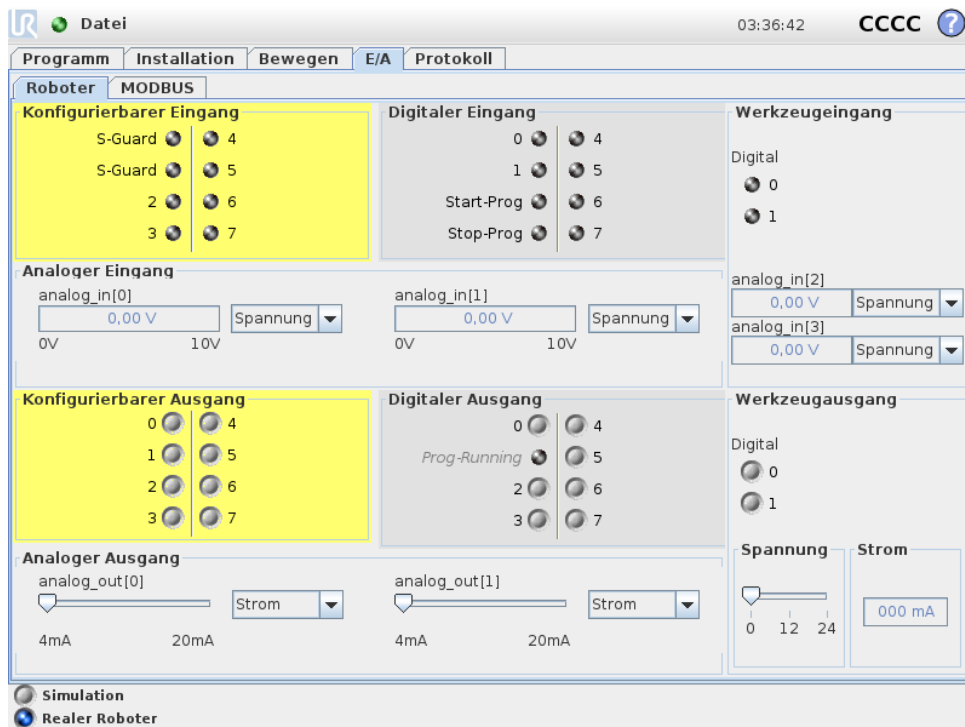
### 13.1.5 Freedrive

Während die *Freedrive*-Taste gedrückt ist, kann der Roboterarm festgehalten und an die gewünschte Stelle gezogen werden. Ist die Gravitationseinstellung (siehe 13.7) im Tab **Einstellung** falsch oder trägt der Roboterarm eine schwere Last, kann sich der Roboterarm bewegen/herabfallen, während die **Freedrive**-Taste gedrückt wird. Lassen Sie die **Freedrive**-Taste in diesem Fall einfach los.

**WARNUNG:**

1. Stellen Sie sicher, dass Sie die richtigen Installationseinstellungen verwenden (z. B. Robotermontagewinkel, Gewicht in TCP, TCP-Ausgleich). Speichern und laden Sie die Installationsdateien zusammen mit dem Programm.
2. Stellen Sie sicher, dass die TCP-Einstellungen und die Robotermontageeinstellungen korrekt eingestellt sind, bevor die **Freedrive**-Taste bedient wird. Sind diese Einstellungen falsch, bewegt sich der Roboterarm, sobald die **Freedrive**-Taste aktiviert wird.
3. Die Freedrive-Funktion (**Impedanz/Zurückfahren**) sollte nur bei Installationen verwendet werden, in denen die Risikobewertung dies zulässt. Werkzeuge und Hindernisse sollten keine scharfen Kanten oder Klemmpunkte haben. Stellen Sie sicher, dass sich niemand in der Reichweite des Roboterarms befindet.

## 13.2 E/A-Tab



In diesem Bildschirm können Sie die spannungsführenden E/A-Signale von/zur Control-Box stets überwachen und einstellen. Der Bildschirm zeigt den aktuellen Zustand der Ein- und Ausgänge an, auch während der Programmausführung. Werden während der Ausführung des Programms Änderungen vorgenommen, so stoppt das Programm. Wenn ein Programm stoppt, behalten alle Ausgangssignale ihren Status bei. Der Bildschirm wird bei nur 10 Hz aktualisiert, sodass ein sehr schnelles Signal eventuell nicht richtig angezeigt wird.

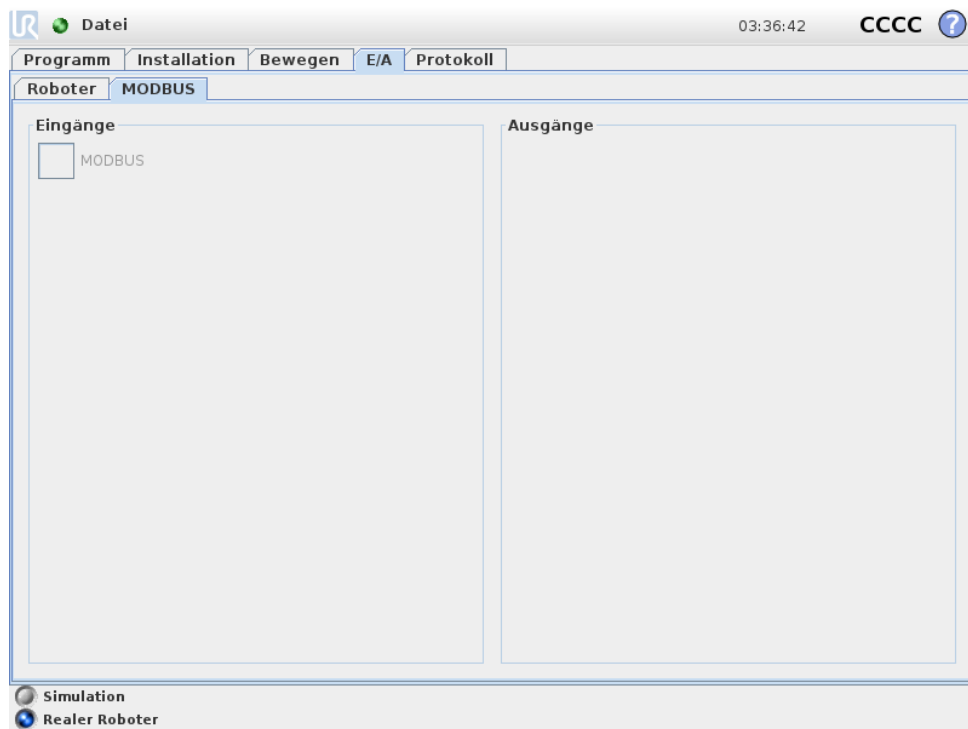
Konfigurierbare E/A können für spezielle Sicherheitseinstellungen reserviert werden, die im Abschnitt Sicherheits-E/A-Konfiguration der Installation definiert sind (siehe 10.13). Reservierte E/A tragen den Namen der Sicherheitsfunktion statt des Standardnamens oder eines benutzerdefinierten Namens. Konfigurierbare Ausgänge, die für Sicherheitseinstellungen reserviert sind, können nicht bedient werden und werden nur als LEDs angezeigt.

Die elektrischen Angaben der Signale sind im Kapitel ?? beschrieben.

**Einstellung Analogdomäne** Die analogen E/A können entweder auf Stromausgang [4-20 mA] oder Spannungsausgang [0-10 V] eingestellt werden. Die Einstellungen werden für mögliche spätere Neustarts des Controllers bei der Speicherung eines Programms gespeichert.

### 13.3 MODBUS-Client-E/A

Hier werden die digitalen MODBUS-Client-E/A-Signale des MODBUS-Client angezeigt, wie sie bei der Installation eingegeben wurden. Bei einem Verlust der Signalverbindung wird der entsprechende Eintrag auf dem Bildschirm deaktiviert.



#### Eingänge

Rufen Sie den Status der digitalen MODBUS-Client-Eingänge auf.

#### Ausgänge

Rufen Sie den Status der digitalen MODBUS-Ausgänge auf und schalten Sie zwischen ihnen um. Ein Signal kann nur dann umgeschaltet werden, wenn die Auswahl für E/A Tab-Steuerung (siehe 13.8) dies zulässt.

## 13.4 AutoMove-Tab

Der Tab „AutoMove“ wird eingesetzt, wenn sich der Roboter in eine bestimmte Position innerhalb seines Wirkungsbereichs bewegen muss. Dazu gehört beispielsweise, wenn sich der Roboterarm laut Programm in eine Startposition bewegen muss, um dieses auszuführen oder wenn er sich während einer Programmänderung zu einem Wegpunkt bewegt.



### Animation

Die Animation zeigt die Bewegung, die der Roboterarm ausführen wird.



#### **VORSICHT:**

Vergleichen Sie die Animation mit der Position des echten Roboterarms und stellen Sie sicher, dass der Roboterarm die Bewegung sicher ausführen kann, ohne auf Hindernisse zu treffen.



#### **VORSICHT:**

Mit der AutoMove-Funktion wird der Roboter entlang der Schattenbahn bewegt. Kollisionen können den Roboter oder andere Geräte beschädigen.

### Auto

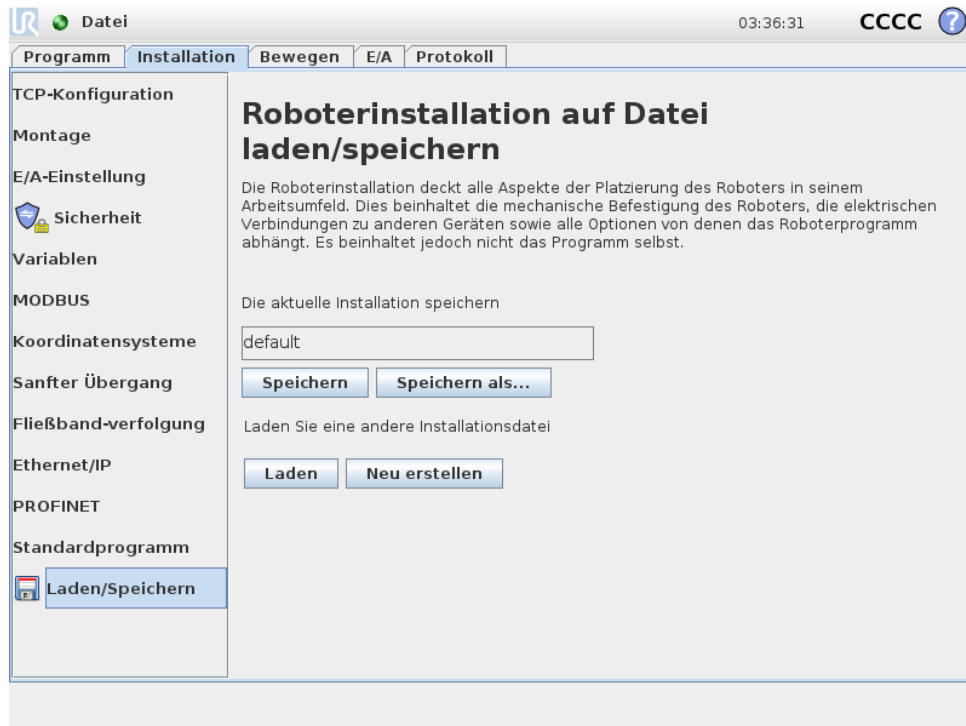
Halten Sie die Schaltfläche **Auto** gedrückt, um den Roboterarm wie in der Animation zu bewegen.

Hinweis: Lassen Sie die Schaltfläche los, um die Bewegung jederzeit zu stoppen.

## Manuell

Drücken Sie die Schaltfläche **Manuell**, um zum **Move**-Tab zu gelangen, wo der Roboter manuell bewegt werden kann. Dies ist nur erforderlich, wenn eine andere Bewegung als die der Animation gewünscht ist.

## 13.5 Installation → Laden/Speichern



Die Roboterinstallation deckt alle Aspekte dessen ab, wie der Roboterarm und die Control-Box in ihrem Arbeitsumfeld platziert werden. Dies beinhaltet die mechanische Befestigung des Roboterarms, die elektrischen Verbindungen zu anderen Geräten sowie alle Optionen, von denen das Roboterprogramm abhängt. Es beinhaltet jedoch nicht das Programm selbst.

Diese Einstellungen können mithilfe der verschiedenen Bildschirme unter **Installation** vorgenommen werden. Davon ausgenommen sind die E/A-Bereiche, die im Register **E/A** festgelegt werden (siehe 13.2).

Es ist möglich, mehr als eine Installationsdatei für den Roboter zu haben. Die erstellten Programme verwenden die aktive Installation und laden diese automatisch, wenn sie verwendet wird.

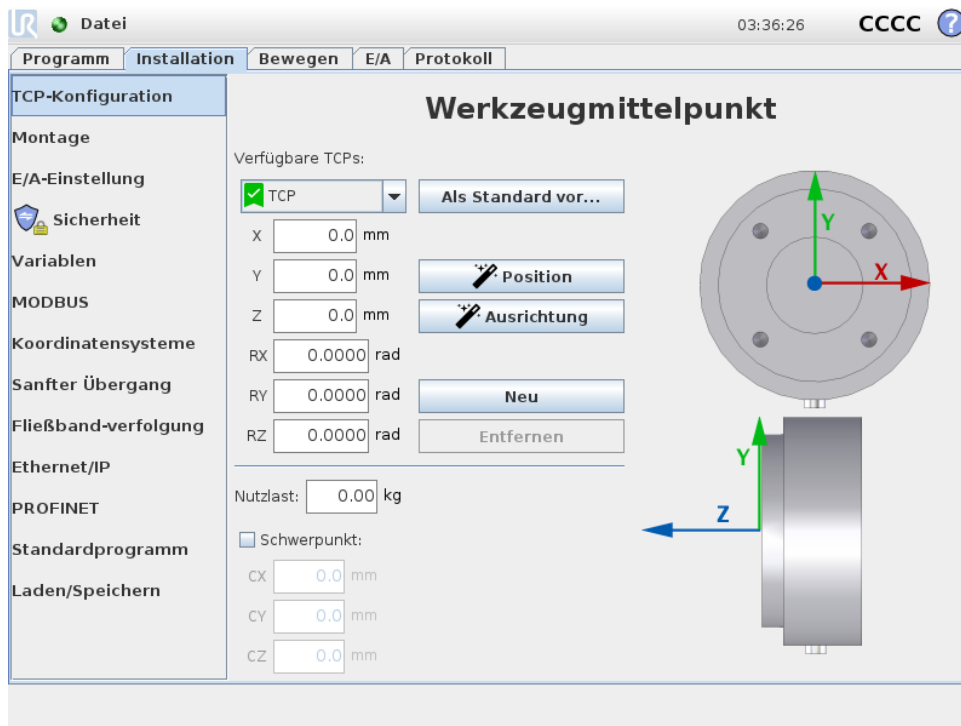
Alle Änderungen an einer Installation müssen gespeichert werden, um nach dem Herunterfahren erhalten zu bleiben. Wenn es während der Installation nicht gespeicherte Änderungen gibt, wird ein Diskettensymbol neben dem **Laden/Speichern**-Text auf der linken Seite des Tab **Installation** angezeigt.

Eine Installation kann durch Drücken der Tasten **Speichern** oder **Speichern als...** gespeichert werden. Alternativ wird die aktive Installation durch das Speichern eines Programms gespeichert. Nutzen Sie die Taste **Laden**, um eine andere Installationsdatei zu laden. Die Taste **Neu erstellen** setzt alle Einstellungen in der Roboterinstallation auf die Werkseinstellungen zurück.

**VORSICHT:**

Die Verwendung des Roboters mit einer von einem USB-Laufwerk geladenen Installation wird nicht empfohlen. Um eine Installation, die auf einem USB-Laufwerk gespeichert ist, auszuführen, laden Sie sie zuerst und speichern Sie sie dann im lokalen Ordner **Programme** mithilfe der Schaltfläche **Speichern als...**

## 13.6 Installation → TCP-Konfiguration



Ein **Tool Center Point (TCP)** ist ein Punkt auf dem Roboterwerkzeug. Der TCP ist im Installations-Tab im Bildschirm **Einrichtung für den Werkzeugmittelpunkt** definiert und benannt (oben abgebildet). Jeder TCP enthält eine Verschiebung und Drehung bezogen auf die Mitte des Werkzeugausgangsflanschs

. Bei der Programmierung zur Rückkehr zu einem zuvor gespeicherten Wegpunkt bewegt ein Roboter den TCP zu der Position und Orientierung, die im Wegpunkt gespeichert ist. Wenn der TCP für lineare Bewegungen programmiert ist, bewegt er sich linear.

Die Koordinaten  $X$ ,  $Y$  und  $Z$  bestimmen die Position eines TCP, während die Koordinaten  $RX$ ,  $RY$  und  $RZ$  seine Ausrichtung angeben. Wenn alle Werte Null sind, fällt der TCP mit dem Mittelpunkt des Werkzeugflanschs zusammen und nimmt das auf dem Bildschirm dargestellte Koordinatensystem an.

### 13.6.1 Hinzufügen, Ändern und Entfernen von TCPs

Um einen neuen TCP zu definieren, tippen Sie auf die Taste **Neu**. Der so erstellte TCP erhält dann automatisch einen eindeutigen Namen und wird im Dropdown-Menü ausgewählt. Die Verschiebung und Rotation des gewählten TCP kann durch Antippen der jeweiligen weißen Textfelder und Eingabe neuer Werte geändert werden. Um den ausgewählten TCP zu entfernen,

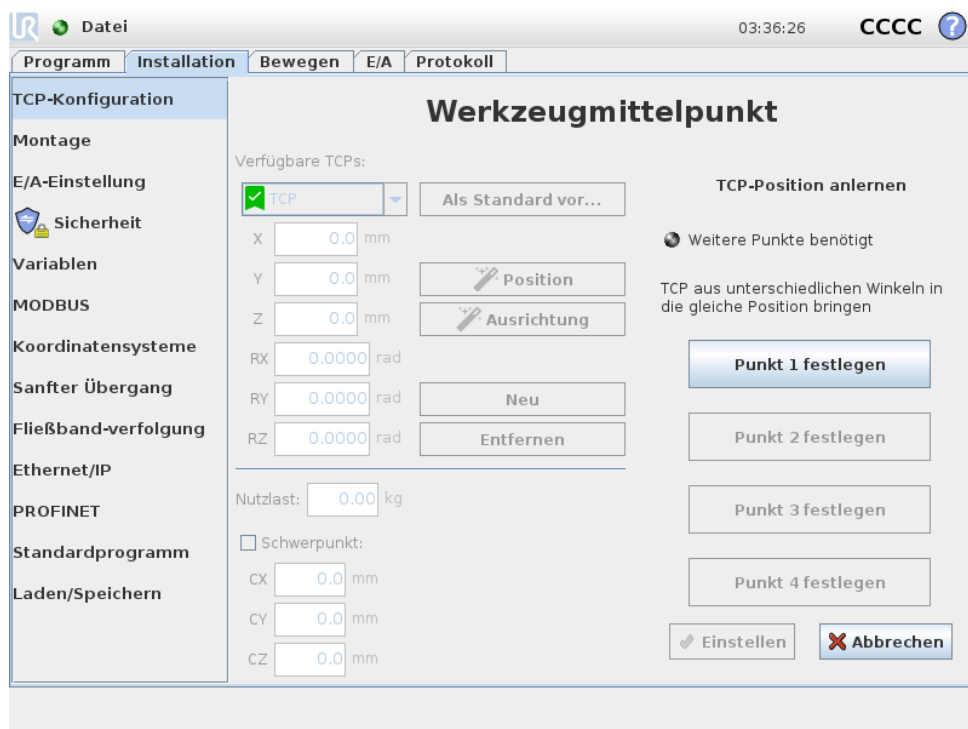
tippen Sie einfach auf die **Entfernen**-Taste. Der letzte verbleibende TCP kann nicht gelöscht werden.

### 13.6.2 Standard-TCP und aktiver TCP

Es ist ein Standard-TCP konfiguriert, der durch ein grünes Symbol links von seinem Namen im **Verfügbares TCP-Dropdown-Menü** markiert ist. Um einen TCP als Standard einzustellen, wählen Sie den gewünschten TCP und tippen Sie auf **Als Standard festlegen**.

Ein TCP-Offset wird als *Aktiv* festgelegt, um alle linearen Bewegungen im kartesischen Koordinatensystem zu bestimmen. Die Bewegung des aktiven TCP im Grafik-Tab veranschaulicht (siehe 14.29). Bevor ein Programm ausgeführt wird, wird der standardmäßige TCP als aktiver TCP eingestellt. Innerhalb eines Programms kann jeder der angegebenen TCPs für eine bestimmte Bewegung des Roboters als *aktiv* gesetzt werden (siehe 14.5 und 14.10).

### 13.6.3 Anlernen (Teaching) der TCP-Position

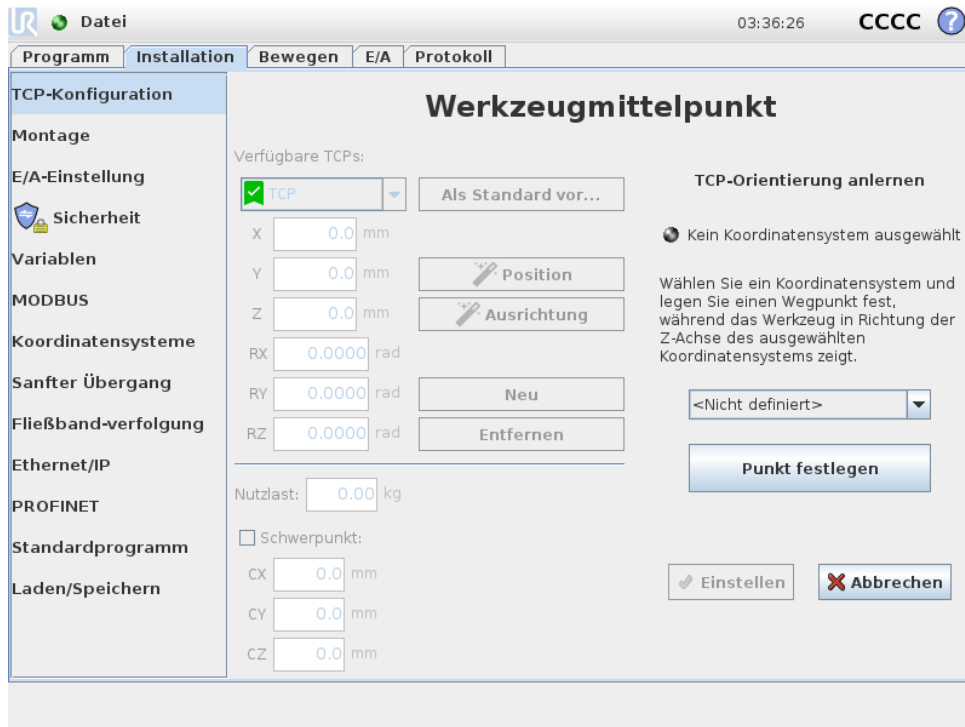


TCP-Positionskoordinaten können wie folgt automatisch berechnet werden:

1. Tippen Sie auf **TCP-Positionsassistent**.
2. Wählen Sie einen festen Punkt im Wirkungsbereich des Roboters.
3. Verwenden Sie die Positionspfeile auf der rechten Seite des Bildschirms, um den TCP aus mindestens drei verschiedenen Winkeln zu bewegen, und um die entsprechenden Positionen des Werkzeugausgangsflanschs zu speichern.
4. Verwenden Sie die Taste **Set**, um die verifizierten Koordinaten auf den entsprechenden TCP anzuwenden. Die Positionen müssen ausreichend vielfältig sein, damit die Berechnung richtig funktioniert. Sind sie nicht ausreichend vielfältig, leuchtet eine rote Status-LED über den Tasten.

Obwohl drei Positionen ausreichend sind, um den TCP zu bestimmen, kann die vierte Position dazu beitragen, sicherzustellen, dass die Berechnung korrekt ist. Die Qualität jedes gespeicherten Punktes in Bezug auf den berechneten TCP wird mit einer grünen, gelben oder roten LED auf der jeweiligen Taste signalisiert.

### 13.6.4 Anlernen (Teaching) der TCP-Ausrichtung



1. Tippen Sie auf **TCP-Ausrichtungsassistent**.
2. Wählen Sie eine Funktion aus der Dropdown-Liste. (Siehe 13.12) für weitere Informationen über das Definieren von neuen Funktionen
3. Tippen Sie auf **Punkt auswählen** und navigieren Sie mit den **Werkzeuggesteuerungs** Pfeilen zu einer Position, in der die Ausrichtung des Werkzeugs und der entsprechende TCP mit den ausgewählten Funktionen des Koordinatensystems zusammenfallen.
4. Überprüfen Sie die berechnete TCP-Ausrichtung und übertragen Sie diese auf den ausgewählten TCP durch Antippen der Taste **Festlegen**.

### 13.6.5 Nutzlast

Das Gewicht des Roboterwerkzeugs ist im unteren Teil des Bildschirms angegeben. Um diese Einstellung zu ändern, tippen Sie einfach auf das weiße Textfeld und geben Sie ein neues Gewicht ein. Die Einstellung gilt für alle definierten TCPs. Weitere Informationen zur maximal zulässigen Nutzlast finden Sie im Hardware-Installationshandbuch.

### 13.6.6 Schwerpunkt

Der Schwerpunkt wird mit Hilfe der Felder *CX*, *CY* und *CZ* festgelegt. Wenn nichts anderes angegeben ist, wird davon ausgegangen, dass der TCP der Schwerpunkt ist. Die Einstellungen gelten für alle definierten TCPs.



**WARNUNG:**

Verwenden Sie die korrekten Installationseinstellungen. Speichern und laden Sie die Installationsdateien zusammen mit dem Programm.

## 13.7 Installation/ → Montage



Die Angabe der Befestigung des Roboterarms dient zwei Zwecken:

1. Die richtige Darstellung des Roboterarms auf dem Bildschirm.
2. Der Controller wird über die Richtung der Gravitationskraft informiert.

Ein erweitertes dynamisches Modell gibt dem Roboterarm glatte und präzise Bewegungen und ermöglicht es dem Roboterarm sich selbst im **Modus Freedrive** zu halten. Aus diesem Grund ist es wichtig, dass die Montage des Roboters korrekt erfolgt.



**WARNUNG:**

Werden die Einstellungen des Roboterarms nicht richtig durchgeführt, kann dies zu häufigen Schutzstopps führen und/oder eine Bewegung des Roboterarms beim Drücken der **Freedrive**-Taste zur Folge haben.

Wenn der Roboterarm auf einem flachen Tisch oder Untergrund montiert ist, sind keine Änderungen auf diesem Bildschirm erforderlich werden. Wird der Roboterarm jedoch **an der Decke, an der Wand** oder **in einem Winkel** montiert, muss dies mithilfe der Tasten angepasst werden.

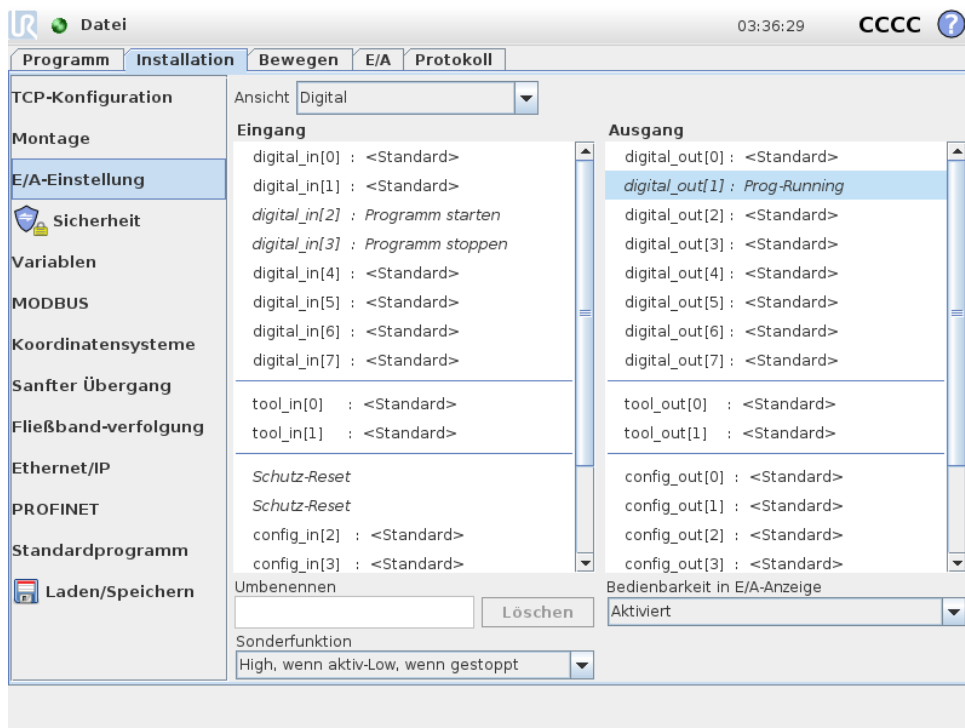
Die Schaltflächen auf der rechten Seite des Bildschirms dienen der Einstellung des Winkels der Roboterarmmontage. Die drei Schaltflächen auf der rechten oberen Seite stellen den Winkel für **Decke** (180°), **Wand** (90°), **Boden** (0°). Die Schaltflächen **Neigen** können stellen einen willkürlichen Winkel ein.

Die Schaltflächen im unteren Teil des Bildschirms werden zur Drehung der Montage des Roboterarms eingesetzt, um der eigentlichen Montage zu entsprechen.

**WARNUNG:**

Verwenden Sie die korrekten Installationseinstellungen. Speichern und laden Sie die Installationsdateien zusammen mit dem Programm.

## 13.8 Installation → E/A-Einstellung



Auf dem E/A-Einrichtungsbildschirm kann der Benutzer E/A-Signale und Aktionen mit der E/A-Tab-Steuerung definieren.

Die Spalten **Eingänge** und **Ausgänge** beinhalten folgende Typen von E/A-Signalen:

- Digitalstandard, Mehrzweck, konfigurierbar und Werkzeug
- Analogstandard, Mehrzweck und Werkzeug
- MODBUS
- Allgemeine Tabs (Boole, Integer und Float) Auf die allgemeinen Tabs kann von einem Feldbus zugegriffen werden (wie z. B. Profinet und EtherNet/IP).

### 13.8.1 E/A Signaltyp

Um die Anzahl der angezeigte Signale in den Abschnitten **Eingänge** und **Ausgänge** zu begrenzen, verwenden Sie das Drop-down-Menü **Ansicht** oben auf dem Bildschirm und ändern Sie die angezeigten Inhalte durch Auswahl der Signaltypen.

### 13.8.2 Zuordnen von benutzerdefinierten Namen

Um sich einfach an das zu erinnern, was die Signale bei der Arbeit mit dem Roboter bewirken, können Benutzer den Eingangs- und Ausgangssignalen Namen zuordnen.

1. Wählen Sie das gewünschte Signal
2. Tippen Sie auf das Textfeld im unteren Teil des Bildschirms, um den Namen festzulegen.
3. Um die Namen auf die Standardeinstellungen zurückzusetzen, tippen Sie auf **Löschen**.

Einem Mehrzweck-Tab muss ein benutzerdefinierter Name zugeordnet werden, um ihn im Programm verfügbar zu machen (d.h. für einen **Warten** Befehl oder den bedingten Ausdruck eines **If**-Befehls. **Warten**- und **If**-Befehle werden entsprechend in (14.9) und (14.18) beschrieben. Benannte, allgemeine Tabs befinden sich in der Auswahl der **Eingänge** oder **Ausgänge** in der **Ausdruckseditor**-Anzeige.

### 13.8.3 E/A-Aktionen und E/A-Tabsteuerung

*Funktionen für Eingänge:* Die allgemeinen Standard-Eingänge, die beiden digitalen Eingänge für Werkzeuge, die allgemeinen Boole'schen Eingangsregister und die konfigurierbaren Eingänge, die nicht für Sicherheitsfunktionen verwendet werden, können eingerichtet werden, um eine Funktion auszulösen. Verfügbare Funktionen:

- Starten des aktuellen Programms
- Stoppen des aktuellen Programms
- Unterbrechen des aktuellen Programms

Darüber hinaus kann eine Aktion so konfiguriert werden, dass Sie den Freedrive Modus aktivieren oder verlassen können, wenn der Eingang HIGH/LOW ist. (Ähnlich der Verwendung der Schaltfläche **Freedrive** auf der Rückseite des Teach Pendant).

*Funktionen für Ausgänge:* Ausgänge bewahren standardmäßig ihre Werte, auch nachdem ein Programm angehalten ist. Diese digitalen Ausgänge können Ausgabefunktionen haben:

- Digitaler Ausgang
- Werkzeugausgang
- Konfigurierbarer Ausgang
- Allgemeiner Boole'scher Ausgang
- Digitaler ModBus-Ausgang
- Analoger Ausgang (Ausgang geht auf Maximum oder Minimum)

Verfügbare Funktionen:

LOW, wenn nicht aktiv : Ausgang ist LOW, wenn der Programmzustand „gestoppt“ oder „unterbrochen“ ist.

HIGH, wenn nicht aktiv : Ausgang ist HIGH, wenn der Programmzustand „gestoppt“ oder „unterbrochen“ ist.

### 13.9 Installations → sicherheit

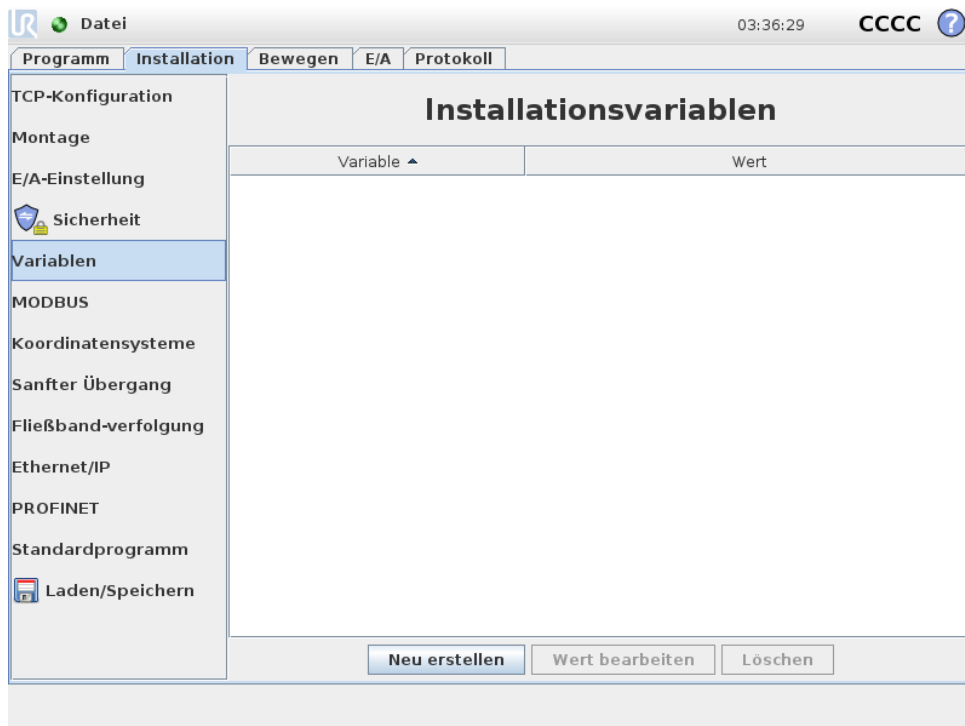
HIGH, wenn aktiv. LOW, wenn gestoppt. : Ausgang ist LOW, wenn der Programmzustand „gestoppt“ oder „unterbrochen“ ist und HIGH, wenn aktiv.

*E/A-Tab-Steuerung:* Es kann darüber hinaus vorgegeben werden, ob ein Ausgang über Tab E/A (entweder durch Programmierer oder Bediener und Programmierer) oder ob der Ausgangswert nur über Roboterprogramme geändert werden kann.

## 13.9 Installations → sicherheit

Siehe Kapitel 10.

## 13.10 Installations → variablen



Auf dem Variablen-Bildschirm erstellte Variablen werden Installationsvariablen genannt und können wie normale Programmvariablen verwendet werden. Installationsvariablen sind eindeutig, da sie ihren Wert beibehalten, selbst wenn ein Programm gestoppt und dann wieder gestartet wird und wenn der Roboterarm und/oder die Control-Box aus- und dann wieder eingeschaltet wird. Ihre Namen und Werte werden mit der Installation gespeichert. Deshalb ist es möglich, die gleiche Variable in mehreren Programmen zu verwenden.



Durch Betätigen von **Neu erstellen** wird ein Feld mit einem Namensvorschlag für die neue Variable geöffnet. Der Name kann geändert bzw. sein Wert eingegeben werden, indem das Textfeld

berührt wird. Die Taste **OK** kann nur angetippt werden, wenn der neue Name nicht bereits in dieser Installation verwendet wird.

Es ist möglich, den Wert einer Installationsvariablen zu ändern, indem die Variable in der Liste markiert und dann auf **Wert bearbeiten** geklickt wird.

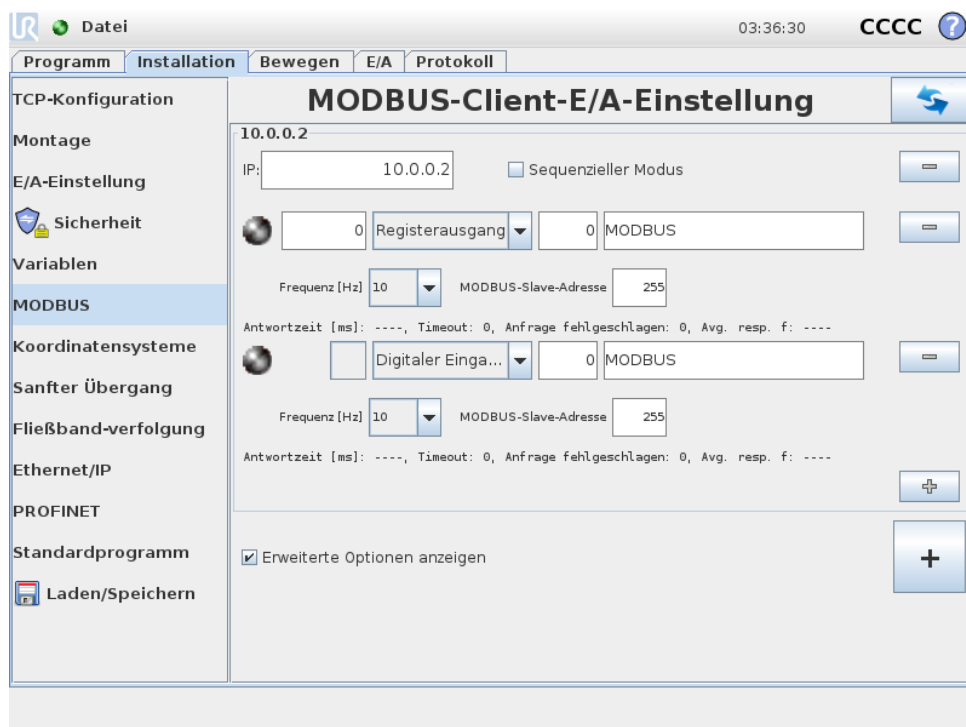
Wählen Sie zum Löschen einer Variablen diese aus der Liste aus, und tippen Sie auf **Löschen**.

Nach dem Konfigurieren der Installationsvariablen muss die Installation eigens gespeichert werden, um diese Konfiguration beizubehalten, (siehe 13.5).

Die Installationsvariablen und deren Werte werden etwa alle 10 Minuten automatisch gespeichert.

Wird ein Programm oder eine geladen und eine oder mehrere der Programmvariablen haben denselben Namen wie die Installationsvariablen, so werden dem Benutzer zwei Optionen zur Behebung dieses Problems angeboten: er kann entweder die Installationsvariablen desselben Namens anstelle der Programmvariablen verwenden oder die in Konflikt stehenden Variablen automatisch umbenennen lassen.

## 13.11 Installation → MODBUS-Client-E/A-Einstellung



Hier können die Signale des MODBUS-Client (Master) eingestellt werden. Verbindungen zu MODBUS-Servern (oder Slaves) auf angegebenen IP-Adressen können mit Eingangs-/Ausgangssignalen (Register oder digital) erstellt werden. Jedes Signal hat einen einmaligen Namen, damit es in Programmen verwendet werden kann.

### Aktualisieren

Drücken Sie auf diese Schaltfläche, um alle MODBUS-Verbindungen zu aktualisieren. Das Aktualisieren trennt alle Modbus-Einheiten und verbindet sie erneut. Alle Statistik wird gelöscht.

---

## Einheit hinzufügen

Drücken Sie auf diese Schaltfläche, um eine neue MODBUS-Einheit hinzuzufügen.

---

## Einheit löschen

Drücken Sie auf diese Schaltfläche, um die MODBUS-Einheit und alle Signale dieser Einheit zu löschen.

---

## Einstellung IP-Adresse Einheit

Hier wird die IP-Adresse der MODBUS-Einheit angezeigt. Drücken Sie auf die Schaltfläche, um diese zu ändern.

---

## Sequenzieller Modus

Nur verfügbar, wenn „Erweiterte Optionen anzeigen“ (siehe 13.11) ausgewählt ist. Dieses Kontrollkästchen zwingt den Modbus-Client auf eine Antwort zu warten, bevor er die nächste Anforderung sendet. Dieser Modus ist für einige Feldbus-Einheiten erforderlich. Das Aktivieren dieser Option kann bei Mehrfach-Signalen hilfreich sein und steigende Anfragehäufigkeit resultiert in Signaltrennungen. Beachten Sie, dass die tatsächliche Signalfrequenz möglicherweise niedriger als angefordert sein könnte, wenn mehrere Signale im sequenziellen Modus definiert sind. Die tatsächliche Signalfrequenz kann in Signalstatistiken festgestellt werden (siehe Abschnitt 13.11). Die Signalanzeige wird gelb, wenn die tatsächliche Signalfrequenz weniger als die Hälfte des aus der Dropdown-Liste „Frequenz“ ausgewählten Wertes beträgt.

---

## Signal hinzufügen

Drücken Sie auf diese Schaltfläche, um der entsprechenden MODBUS-Einheit ein Signal hinzuzufügen.

---

## Signal löschen

Drücken Sie auf diese Schaltfläche, um ein MODBUS-Signal der entsprechenden MODBUS-Einheit zu löschen.

---

## Signaltyp einstellen

Verwenden Sie dieses Auswahlmü, um den Signaltyp auszuwählen. Die folgenden Typen stehen zur Verfügung:

*Digitaleingang:* Ein digitaler Eingang (Coil) ist eine Ein-Bit-Menge, die von der MODBUS-Einheit aus dem Coil abgelesen wird und im Adressfeld des Signals angegeben ist. Funktionscode 0x02 (diskrete Ausgänge lesen) wird eingesetzt.

*Digitalausgang:* Ein digitaler Ausgang (Coil) ist eine Ein-Bit-Menge, die auf high oder low eingestellt werden kann. Bevor der Wert dieses Ausgangs durch den Benutzer eingestellt wurde, wird der Wert von der dezentralen MODBUS-Einheit abgelesen. Das bedeutet, dass der Funktionscode 0x01 (Read Coils) verwendet wird. Wenn der Ausgang entweder durch ein Roboterprogramm oder durch Betätigung der Schaltfläche **Signalwert bestimmen** festgelegt wurde, wird ab diesem Zeitpunkt der Funktionscode 0x05 (Write Single Coil) eingesetzt.

**Registereingang:** Ein Registereingang ist eine 16-Bit-Menge, die von der Adresse abgelesen wird, die im Adressfeld angegeben ist. Der Funktionscode 0x04 (Read Input Registers) wird eingesetzt.

**Registerausgang:** Ein Registerausgang ist eine 16-Bit-Menge, die durch den Benutzer eingestellt werden kann. Bevor der Wert dieses Registers eingestellt wurde, wird der Wert von der dezentralen MODBUS-Einheit abgelesen. Das bedeutet, dass der Funktionscode 0x03 (Read Holding Registers) verwendet wird. Wenn der Ausgang entweder durch ein Roboterprogramm oder durch Betätigung der Schaltfläche **Signalwert bestimmen** festgelegt wurde, wird der Funktionscode 0x06 (Einzelnes Register schreiben) eingesetzt, um den Wert auf der dezentralen MODBUS-Einheit festzulegen.

---

## Signaladresse einstellen

Dieses Feld zeigt die Adresse des dezentralen MODBUS-Servers. Verwenden Sie die Bildschirmtastatur, um eine andere Adresse auszuwählen. Gültige Adressen hängen von Hersteller und Konfiguration der dezentralen MODBUS-Einheit ab.

---

## Signalname einstellen

Durch Verwendung der Bildschirmtastatur kann der Benutzer das Signal benennen. Dieser Name wird verwendet, wenn das Signal in Programmen eingesetzt wird.

---

## Signalwert

Hier wird der Istwert des Signals angezeigt. Bei Registersignalen wird der Wert als vorzeichenlose ganze Zahl ausgedrückt. Bei Ausgangssignalen kann der gewünschte Signalwert mit der Schaltfläche eingestellt werden. Für den Registerausgang muss der an die Einheit zu schreibende Wert als vorzeichenlose ganze Zahl bereitgestellt werden.

---

## Status Signalkonnektivität

Dieses Symbol zeigt an, ob das Signal korrekt gelesen/geschrieben (grün) werden kann oder ob die Einheit unerwartet antwortet oder nicht erreichbar ist (grau). Wird eine MODBUS-Ausnahmeantwort empfangen, wird der Antwortcode angezeigt. Die MODBUS-TCP-Ausnahmeantworten lauten wie folgt:

- E1:* UNZULÄSSIGE FUNKTION (0x01) Der in der Abfrage empfangene Funktionscode ist keine zulässige Aktion für den Server (oder Slave).
- E2:* UNZULÄSSIGE DATENADRESSE (0x02) Der in der Abfrage empfangene Funktionscode ist keine zulässige Aktion für den Server (oder Slave). Prüfen Sie, ob die eingegebenen Signaladressen mit der Einstellung des dezentralen MODBUS-Servers übereinstimmen.
- E3:* UNZULÄSSIGER DATENWERT (0x03) Ein im Abfragedatenfeld enthaltener Wert ist für den Server (oder Slave) unzulässig. Prüfen Sie, ob der eingegebene Signalwert für die angegebenen Adressen auf dem dezentralen MODBUS-Server gültig ist.
- E4:* FEHLER IM SLAVE-GERÄT (0x04) Ein nicht wiederherstellbarer Fehler ist aufgetreten, während der Server (oder Slave) versucht hat, die angeforderte Aktion auszuführen.
- E5:* BESTÄTIGEN (0x05) Spezielle Verwendung in Verbindung mit Programmierbefehlen, die an die dezentrale MODBUS-Einheit gesendet werden.

*E6: SLAVE-GERÄT MOMENTAN NICHT VERFÜGBAR (0x06)* Spezielle Verwendung in Verbindung mit Programmierbefehlen, die an die dezentrale MODBUS-Einheit gesendet werden; der Slave (Server) kann im Moment nicht antworten.

---

## Erweiterte Optionen anzeigen

Dieses Kontrollkästchen zeigt die erweiterten Optionen für jedes Signal bzw. blendet diese aus.

---

## Erweiterte Optionen

*Update-Häufigkeit:* Mit diesem Menü kann die Aktualisierungsfrequenz des Signals geändert werden. Dies gilt für die Frequenz, mit der Anfragen an das dezentrale MODBUS-Steuergesetz geschickt werden, um den Signalwert entweder zu lesen oder zu schreiben. Ist die Frequenz auf 0 gesetzt, so werden Modbus-Anfragen auf Anforderung unter Verwendung von *Modbus\_erhalte\_Signal\_Status*, *Modbus\_setze\_Ausgangs\_Register* und *Modbus\_setze\_Ausgangs\_Signal*-Scriptfunktionen angestoßen.

*Slave-Adresse:* Dieses Textfeld kann verwendet werden, um eine spezifische Slave-Adresse für Anfragen im Zusammenhang mit einem spezifischen Signal einzustellen. Der Wert muss im Bereich von 0-255 liegen. Der Standardwert ist 255. Wenn Sie diesen Wert ändern, wird empfohlen, das Handbuch des dezentralen MODBUS-Geräts hinzuzuziehen, um seine Funktion zu prüfen, wenn die Slave-Adresse geändert wird.

*Zählvariable neu verbinden:* Anzahl der beendeten und neu hergestellten TCP-Verbindungen

*Verbindungsstatus:* TCP-Verbindungsstatus

*Antwortzeit [ms]:* Zeitspanne zwischen gesendeter Modbus-Anforderung und empfangener Antwort - wird nur bei aktiver Kommunikation aktualisiert.

*Modbus-Paket-Fehler:* Anzahl der empfangenen Pakete, die Fehler enthielten. (Z. B. ungültige Länge, fehlende Daten, TCP-Socket-Fehler)

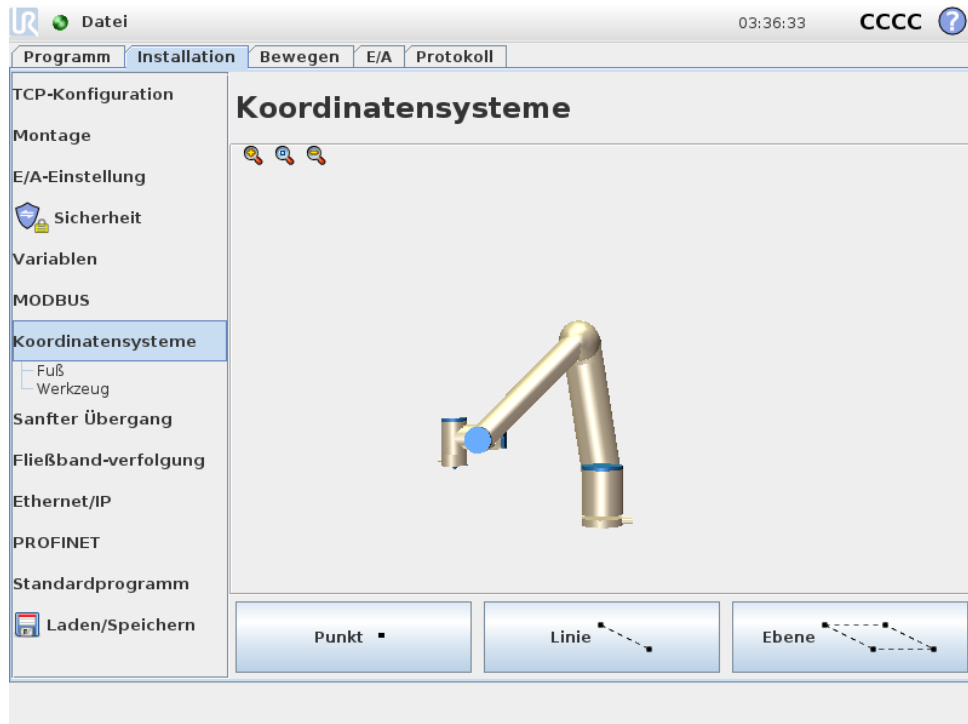
*Timeout:* Anzahl von Modbus-Anfragen ohne Antwort.

*Anfrage fehlgeschlagen:* Anzahl der Pakete, die aufgrund eines ungültigen Socket-Status nicht gesendet werden konnten.

*Istfrequenz:* Die durchschnittliche Häufigkeit von Client (Master)-Signal-Status-Updates. Dieser Wert wird jedes Mal neu berechnet, wenn das Signal eine Antwort vom Server (oder Slave) erhält.

Alle Zähler zählen bis 65535 und springen dann wieder auf 0.

## 13.12 Installation → Funktionen



Die **Funktion** ist eine Darstellung eines solchen Objekts, das mit einem Namen für zukünftige Referenzzwecke und einer sechsdimensionalen Pose (Position und Orientierung) in Bezug auf die Roboterbasis definiert wurde.

Einige Unterkomponenten eines Roboterprogramms bestehen aus Bewegungen, die sich nicht auf die Basis des Roboterarms beziehen, sondern relativ zu bestimmten Punkten auszuführen sind. Dabei kann es sich um Tische, andere Maschinen, Werkstücke, Fließbänder, Paletten, Kamerasysteme, Rohteile oder Begrenzungen handeln, die in der Umgebung des Roboterarms vorhanden sind. Zwei vordefinierte Funktionen existieren immer für den Roboter. Jede Funktion hat eine von der Konfiguration des Roboterarms selbst definierte Pose:

- Die Basisfunktion hat ihren Ursprung im Zentrum der Roboterbasis (siehe Abbildung 13.1)
- Die Werkzeugfunktion Der Ursprung liegt im Zentrum des aktuellen TCP (siehe Abbildung 13.2)

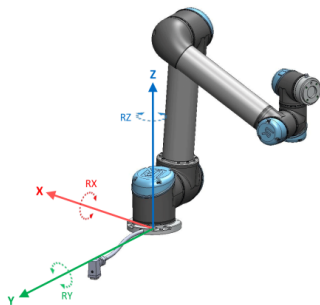


Abbildung 13.1: Basisfunktion

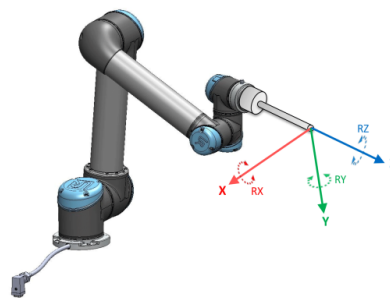


Abbildung 13.2: Tool (TCP)-Funktion

Benutzerdefinierte Funktionen werden über eine Methode positioniert, die die aktuelle Pose des TCP im Arbeitsbereich verwendet. Dies bedeutet: Der Benutzer kann mithilfe der Position und Lage von Funktionen anlernen oder den Roboter mit dem manuellen Tippbetrieb in die gewünschte Position bringen.

Für die Definition einer benutzerdefinierten Funktion stehen drei verschiedene Methoden zur Auswahl (**Punkt**, **Linie** und **Ebene**). Die beste Strategie für eine bestimmte Anwendung hängt von der Art des verwendeten Objekts und den Anforderungen an Genauigkeit ab. Im Allgemeinen ist eine Funktion zu bevorzugen, die auf mehreren Eingangspunkten (**Linie** und **Ebene**) basiert, wenn diese auf das spezifische Objekt zutrifft.

Um die Richtung eines Linear-Fließbands genau definieren zu können, definieren Sie zwei Punkte einer Linienfunktion mit möglichst hoher physikalischer Trennung. Die Punktfunktion kann auch verwendet werden, um ein Linear-Fließband zu definieren, allerdings muss der Anwender den TCP in die Richtung der Bandbewegung weisen.

Werden mehr Punkte für die Definition von Position und Lage z. B. eines Tisches verwendet, bedeutet dies, dass die Ausrichtung eher auf Positionen anstatt auf der Ausrichtung eines einzelnen TCP basiert. Eine einzelne TCP-Ausrichtung ist mit hoher Präzision schwerer zu konfigurieren.

Erfahren Sie mehr über die verschiedenen Methoden zum Definieren einer Funktion in (Abschnitte: 13.12.2), (13.12.3) und (13.12.4).

### 13.12.1 Verwenden einer Funktion

Wurde eine Funktion bei der Installation definiert, können Sie diesen Funktionen Roboterbewegungen aus dem Roboterprogramm (z. B. **MoveL** und **MoveP**-Befehle) zuweisen (siehe Abschnitt 14.5). Dies ermöglicht eine einfache Anpassung eines Roboterprogramms (z. B. bei mehreren Roboterstationen oder wenn ein Objekt während der Programmlaufzeit verschoben oder permanent in der Szene verschoben wird). Durch einfaches Anpassen der Funktion eines Objekts, werden alle Bewegungen innerhalb des Programms relativ mit dem Objekt verschoben. Weitere Beispiele finden Sie unter (Abschnitte 13.12.5) und (13.12.6).

Funktionen mit aktivierbarem Kontrollfeld „Tippbetrieb“ sind auch ein nützliches Tool, wenn der Roboter im Move-Tab (Abschnitt 13.1) oder **Posen-Editor**-Bildschirm (siehe 12.2) manuell bewegt wird. Ist eine Funktion als Bezug ausgewählt, werden die Tasten für die Bewegung des Werkzeugs für Translationen und Rotationen im ausgewählten Funktionsbereich angewendet (siehe 13.1.2) und (13.1.3), ebenso wie die aktuelle Anzeige der TCP-Koordinaten. Wenn z. B. ein Tisch als eine Funktion definiert und als Referenz im Tab „Move“ ausgewählt ist, bewegen die Pfeile (nach oben/unten, links/rechts, vorwärts/rückwärts) den Roboter in diese Richtungen relativ zum Tisch. Zusätzlich befinden sich die TCP-Koordinaten im Rahmen des Tisches.

#### Umbenennen

Diese Taste benennt eine Funktion um.

#### Löschen

Diese Schaltfläche löscht eine ausgewählte Funktion und alle Unterfunktionen.

## Achsen zeigen

Wählen Sie, ob die Koordinatenachsen der ausgewählten Funktion in der 3D-Grafik sichtbar sein sollen. Die Auswahl gilt für diese Anzeige und den Bewegen-Bildschirm.

## Ändern des Punkts

Verwenden Sie die Schaltfläche **Diesen Punkt ändern**, um die ausgewählte Funktion einzustellen oder zu ändern. Der **Move**-Tab (Abschnitt 13.1) erscheint und eine neue Position der Funktion kann eingestellt werden.

## Tippbetrieb

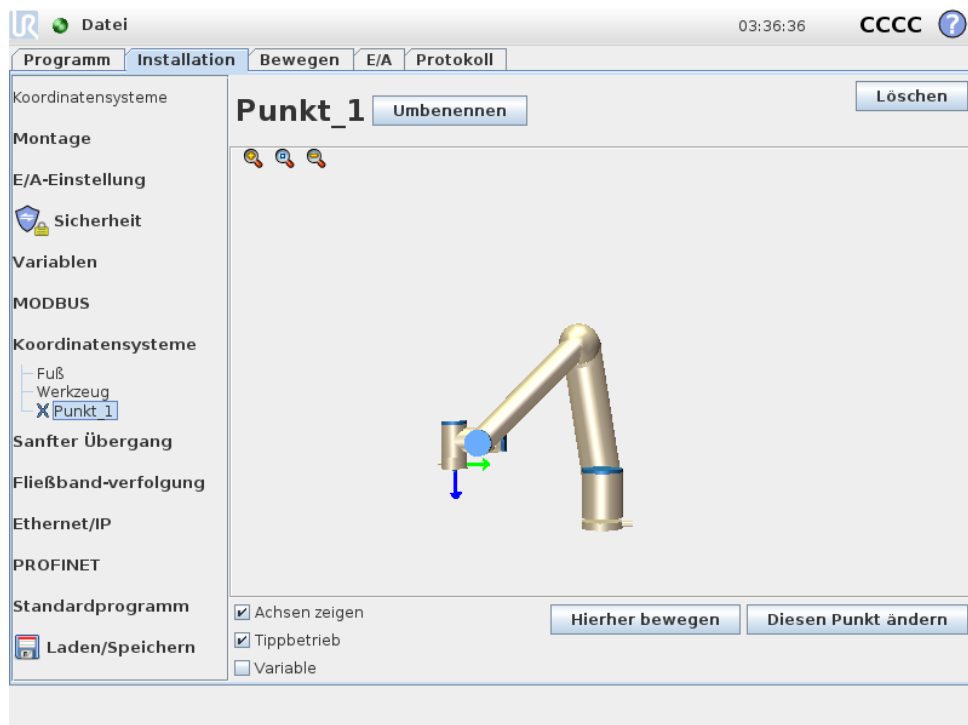
Wählen Sie, ob ein Tippbetrieb für die gewählte Funktion möglich sein soll. Dadurch wird festgelegt, ob die jeweilige Funktion auch im Bewegen-Bildschirm als auswählbare Referenzfunktion für manuelle Bewegungen angezeigt wird.

## Verwenden von Roboter hierher bewegen

Drücken Sie die Schaltfläche **Roboter hierher bewegen** um den Roboterarm in Richtung der ausgewählten Funktion zu bewegen. Am Ende dieser Bewegung stimmen die Koordinatensysteme der Funktion und des TCP überein.

### 13.12.2 Neuer Punkt

Betätigen Sie die Schaltfläche **Punkt**, um eine Punkt-Funktion zur Installation hinzuzufügen. Die Punkt-Funktion definiert Sicherheitsgrenzen oder eine globalen Grundposition des Roboterarms. Die Punkt-Funktion Pose wird als die Position und Ausrichtung des TCP definiert.



### 13.12.3 Neue Linie

Betätigen Sie die Schaltfläche **Linie**, um eine Linien-Funktion zur Installation hinzuzufügen. Die Linienfunktion definiert Linien, denen der Roboter folgen muss. (z. B. bei Fließband-Tracking). Eine Linie  $l$  ist als eine Achse zwischen zwei Punkt-Funktionen  $p1$  und  $p2$  definiert, wie in Abbildung 13.3 gezeigt.

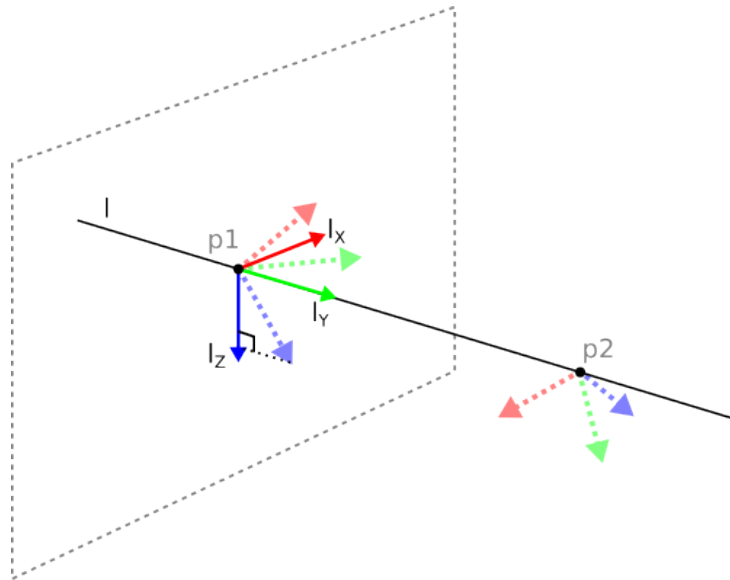
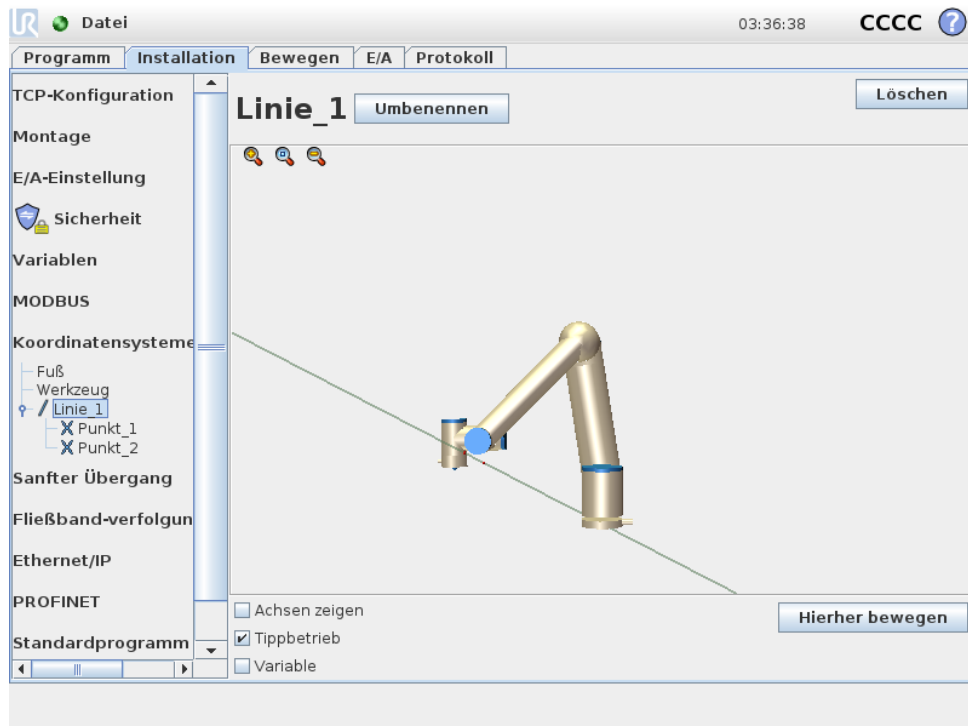


Abbildung 13.3: Definition der Zeilenfunktion

In Abbildung 13.3 ist die Achse vom ersten zum zweiten Punkt gerichtet und beschreibt die y-Achse des Koordinatensystems der Linien-Funktion. Die z-Achse wird durch die Orientierung der z-Achse von  $p1$  definiert und steht senkrecht auf der Linie. Die Position des Koordinatensystems der Linie ist dieselbe wie die Position von  $p1$ .



### 13.12.4 Funktion Ebene

Wählen Sie die Ebenenfunktion, wenn ein Koordinatensystem mit hoher Präzision erforderlich ist, z. B. bei der Arbeit mit einem Sichtsystem oder bei Bewegungen relativ zu einem Tisch.

#### Hinzufügen einer Ebene

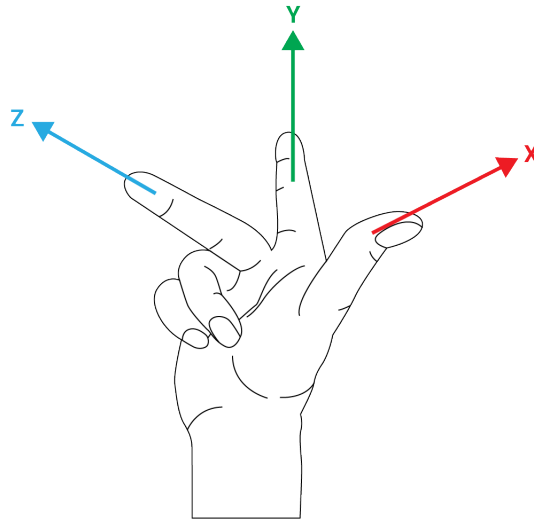
1. Unter Installation wählen Sie **Funktionen**.
2. Unter Funktionen wählen Sie **Ebene**.

#### Anlernen (Teaching) einer Ebene

Wenn Sie die Schaltfläche Ebene zum Erstellen einer neuen Ebene klicken, führt Sie der Assistent auf dem Bildschirm durch das Erstellen einer Ebene.

1. Wählen Sie Origo
2. Bewegen Sie den Roboter, um die Richtung der positiven X-Achse der Ebene zu definieren
3. Bewegen Sie den Roboter, um die Richtung der positiven Y-Achse der Ebene zu definieren

Die Ebene wird mit der Regel der rechten Hand definiert, sodass die Z-Achse das Kreuzprodukt der X-Achse und der Y-Achse ist, wie unten dargestellt.



**HINWEIS:**

Sie können die Ebene erneut in entgegengesetzter Richtung der X-Achse anlernen, wenn Sie wollen, dass die Ebene in entgegengesetzter Richtung normal ist.

Ändern Sie eine vorhandene Ebene durch die Auswahl einer Ebene und drücken Sie „Ebene ändern“. Damit verwenden Sie den gleichen Leitfaden wie für das Anlernen einer neuen Ebene.

**13.12.5 Beispiel: Manuelle Anpassung einer Funktion zur Anpassung eines Programms**

Stellen Sie sich eine Anwendung vor, in welcher mehrere Teile eines Roboterprogramms relativ zu einem Tisch definiert sind. In Abbildung 13.4 wird dies als Bewegung der Wegpunkte wp1 bis wp4 dargestellt.

Roboterprogramm

```

MoveJ
  s1
MoveL # Funktion: P1_var
  wp1
  wp2
  wp3
  wp4
    
```

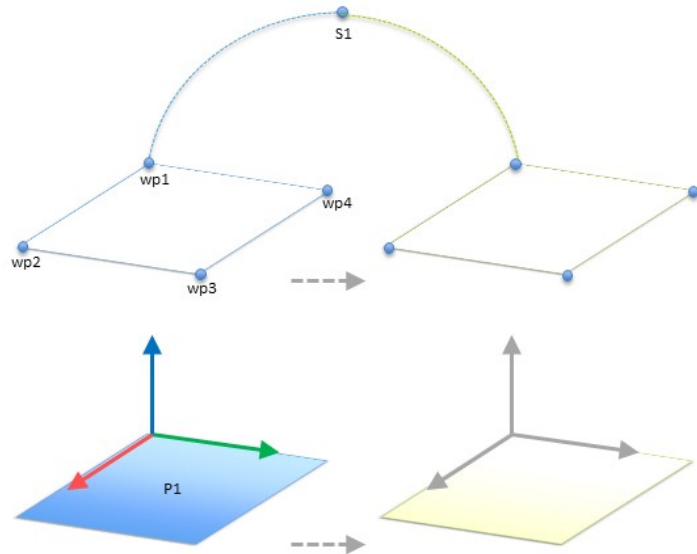


Abbildung 13.4: Einfaches Programm mit vier Wegpunkten in Relation zu einer Funktionsebene, manuell aktualisiert durch Ändern der Funktion

Die Anwendung erfordert, dass das Programm für mehrere Roboterinstallationen verwendet werden soll, in welchen nur die Positionen des Tisches leicht variieren. Die Bewegung relativ zum Tisch ist identisch. Durch Definition der Tischposition als Funktion *P1* in der Installation kann das Programm mit einem *MoveL*-Befehl, welcher relativ zu Ebene konfiguriert ist, einfach für weitere Roboter angewendet werden, indem lediglich die Installation mit der tatsächlichen Position des Tisches aktualisiert wird.

Das Konzept gilt für eine beliebige Anzahl von Funktionen einer Applikation, um ein flexibles Programm zu erhalten, welches die gleiche Aufgabe auf mehreren Robotern ausführen kann. Und dies selbst dann, wenn andere Bereiche der Arbeitsfläche zwischen den Installationen variieren.

### 13.12.6 Beispiel: Dynamisches Aktualisieren einer Funktion

Stellen Sie sich eine ähnliche Anwendung vor, in welcher der Roboter seinen TCP ebenfalls in einem bestimmten Muster über den Tisch bewegt, um eine spezifische Aufgabe zu lösen (wie in Abbildung 13.5 gezeigt).

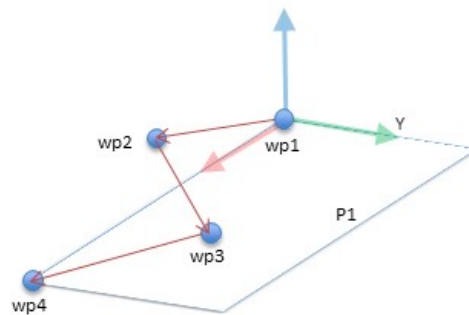


Abbildung 13.5: Ein *MoveL* Befehl mit vier Wegpunkten relativ zu einer Ebenenfunktion

Die Bewegung relativ zu *P1* wird mehrmals wiederholt, jeweils durch einen Offset von *o*. In diesem Beispiel ist der Offset auf 10 cm in Y-Richtung festgelegt (siehe Abbildung 13.6, Offsets *O1* und *O2*). Dies kann mit den Skriptfunktionen *pose\_add()* oder *pose\_trans()* erreicht werden, mit denen die Variable beeinflusst wird.

```

Roboterprogramm
MoveJ
  wp1
y = 0,01
o = p[0,y,0,0,0,0]
P1_var = pose_trans(P1_var, o)
MoveL # Funktion: P1_var
  wp1
  wp2
  wp3
  wp4
    
```

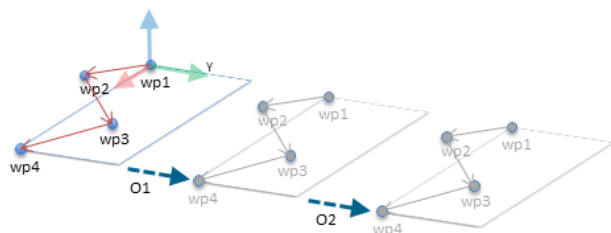


Abbildung 13.6: Anwenden einer Verschiebung bei der Ebenenfunktion

Anstelle eines Offset ist es auch möglich, auf eine andere Funktion zu wechseln. Dies wird im Beispiel unten gezeigt (siehe Abbildung 13.7), in dem die Bezugsfunktion für den *MoveL*-Befehl *P1\_var* zwischen zwei Ebenen *P1* und *P2* wechseln kann.

Roboterprogramm

```

MoveJ
  S1
  if (digital_input[0]) then
    P1_var = P1
  else
    P1_var = P2
MoveL # Funktion: P1_var
  wp1
  wp2
  wp3
  wp4

```

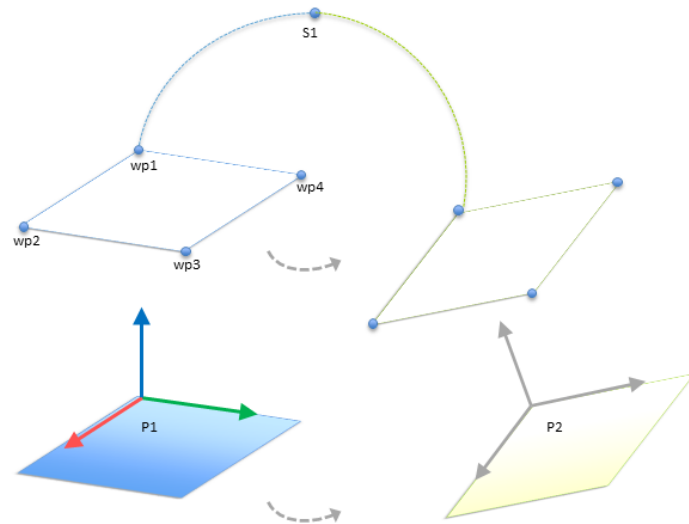


Abbildung 13.7: Umschalten von einer Ebenenfunktion zu einer anderen

## 13.13 Einrichtung der Fließbandverfolgung

Wird ein Fließband verwendet, kann der Roboter so konfiguriert werden, dass er dessen Bewegung verfolgt. Die Fließband-Tracking-Einrichtung bietet Robotereinstelloptionen für den Betrieb mit absoluten und relativen Encodern sowie einem geraden oder kreisförmigen Fließband.

### Fließband-Parameter

*Relative:* Encoder können an die digitalen Eingänge 0 bis 3 angeschlossen werden. Das Decodieren von digitalen Signalen läuft mit 40 kHz. Mit einem **Quadratur** Encoder (erfordert zwei Eingänge) ist der Roboter in der Lage, die Geschwindigkeit sowie Richtung des Fließbands zu bestimmen. Ist die Richtung des Fließbands konstant, kann ein einzelner Eingang dazu verwendet werden, die Geschwindigkeit des Fließbands über die Erkennung einer Steigenden, Fallenden oder von Steigenden und Fallenden Signalfanken zu bestimmen.

*Absolute:* Encoder können über ein MODBUS-Signal verbunden werden. Dies erfordert eine Vorkonfigurierung des digitalen MODBUS-Eingangsregisters in Abschnitt 13.11).

### Linear-Fließband

Wurde ein Linear-Fließband ausgewählt, muss in der Installation unter **Funktionen** eine Linienfunktion konfiguriert werden, um die Richtung des Fließbands zu ermitteln. Die Linienfunktion sollte parallel zu der Richtung des Fließbands verlaufen und es sollte ein großer Abstand zwischen den beiden Punkten bestehen, die die Linienfunktion bestimmen. Konfigurieren Sie die Linienfunktion so, dass Sie das Werkzeug beim Anlernen fest gegen die Seite des Fließbands stemmen. Verläuft die Richtung der Linienfunktion entgegen der Fließbandbewegung, verwenden Sie die Schaltfläche **Richtung umkehren**.

Das Feld **Inkrement pro Meter** zeigt die Anzahl der Inkremente, die der Encoder während

eines Meters Fahrstrecke des Fließbands erzeugt.

$$\text{Inkmente pro Meter} = \frac{\text{Inkmente pro Umdrehung des Encoder}}{2\pi \cdot \text{Radius Encoderscheibe[m]}} \quad (13.1)$$

### Kreisförmiger Fließband

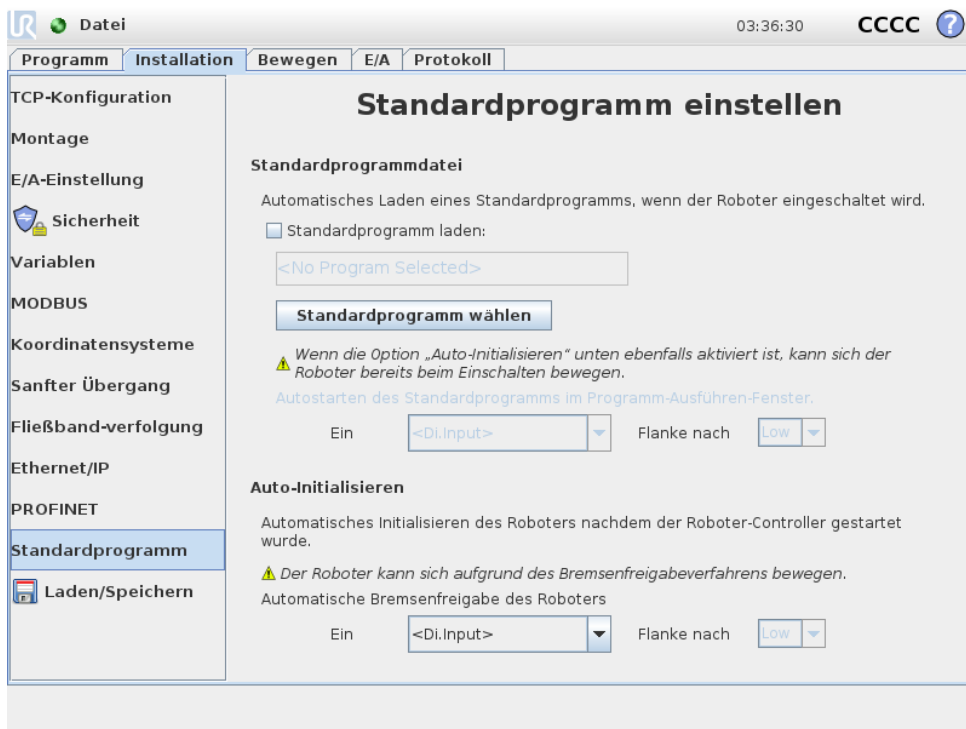
Bei der Verfolgung eines kreisförmigen Fließbands, muss der Mittelpunkt des Fließband definiert sein.

1. Definieren Sie den Mittelpunkt im Teil **Funktionen** der Installation. Der Wert **Inkmente pro Meter** wird als die Anzahl der Inkmente verwendet, die der Encoder während einer vollen Umdrehung des Fließbands erzeugt.
2. Wählen Sie das Kontrollkästchen **Werkzeug mit Fließband drehen**, wenn sich die Werkzeugausrichtung am Fließband orientieren soll (z. B., w wenn das Werkzeug senkrecht zum Fließband ist, wird es während der Bewegung senkrecht bleiben).
3. Deaktivieren Sie das Kontrollkästchen **Werkzeug mit dem Fließband drehen**, wenn die Orientierung durch die Bahn gesteuert werden soll.

### 13.14 Sanfter Übergang zwischen Sicherheitsmodi

Beim Umschalten zwischen Sicherheitsmodi bei Ereignissen (z. B. Eingang Reduzierter Modus, Auslöseebenen für Reduzierter Modus, Schutzstopp und 3-Stellungs-Zustimmsschalter) ist der Roboterarm bestrebt, einen „sanften“ Übergang mit 0,4 s zu bewerkstelligen. Vorhandene Anwendungen bleiben davon unberührt, was einer „harten“ Einstellung entspricht. Neue Installationsdateien erhalten standardmäßig die „sanfte“ Einstellung.

### 13.15 Installation → Standardprogramm



Dieser Startbildschirm enthält Einstellungen für das automatische Laden und Starten eines Standardprogramms und für die Auto-Initialisierung des Roboterarms beim Einschalten.

**WARNUNG:**

1. Sind automatisches Laden, automatisches Starten und automatisches Initialisieren aktiviert, führt der Roboter das Programm aus, sobald die Kontrolleinheit eingeschaltet ist und solange die Eingangssignale mit dem gewählten Signalpegel übereinstimmen. Der Flankenübergang für den gewählten Signalpegel ist in diesem Fall beispielsweise nicht erforderlich.
2. Seien Sie vorsichtig, wenn der Signalpegel auf LOW eingestellt ist. Eingangssignale sind standardmäßig LOW, sodass das Programm automatisch ausgeführt wird, ohne durch ein externes Signal ausgelöst zu werden.

---

**Laden eines Standardprogramms**

Ein Standard-Programm wird geladen, nachdem die Kontrolleinheit eingeschaltet wurde. Darüber hinaus wird das Standardprogramm auch automatisch geladen, wenn die **Programm ausführen**-Anzeige (siehe 11.4) geöffnet wird und kein Programm geladen ist.

---

**Starten eines Standardprogramms**

Das Standardprogramm kann auf dem Bildschirm **Programm ausführen** automatisch gestartet werden. Wird das Standardprogramm geladen und der spezifizierte Flankenübergang eines externen Eingangssignals erkannt, startet das Programm automatisch.

Beim Programmstart ist das aktuelle Eingangssignal nicht definiert. Das Wählen eines Übergangs, der dem Signalpegel beim Start entspricht, startet das Programm sofort. Darüber hinaus wird die Auto-Startfunktion beim Verlassen des Bildschirms **Programm ausführen** oder beim Antippen der Stopptaste im Dashboard solange deaktiviert, bis die Taste „Ausführen“ noch einmal gedrückt wird.

---

**Auto-Initialisierung**

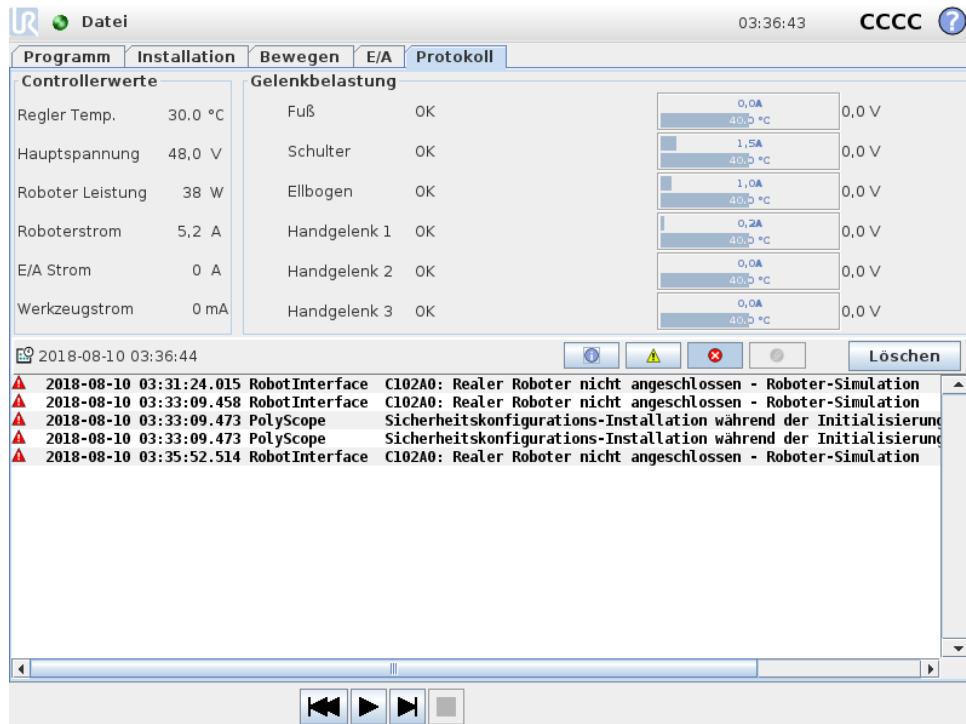
Der Roboterarm wird automatisch initialisiert. Auf dem spezifizierten Flankenübergang eines externen Eingangssignals wird der Roboterarm vollständig initialisiert, unabhängig vom sichtbaren Bildschirm.

Das Lösen der Bremse ist die endgültige Initialisierungsphase. Beim Lösen der Bremse macht der Roboterarm eine kleine Bewegung und ein klickendes Geräusch. Darüber hinaus können die Bremsen nicht automatisch freigegeben werden, wenn die Konfiguration nicht dem erkannten Aufbau (basierend auf den Sensordaten) entspricht. In diesem Fall muss der Roboter auf dem Initialisierungsbildschirm manuell initialisiert werden (siehe 11.5).

Beim Programmstart ist das aktuelle Eingangssignal nicht definiert. Das Wählen eines Übergangs, der dem Signalpegel beim Start entspricht, initialisiert den Roboterarm sofort.

Das automatische Initialisierungsmerkmal funktioniert nur bei ausgeschaltetem Roboterarm.

## 13.16 Der Tab „Protokoll“



**Zustand des Roboters** Die obere Hälfte des Bildschirms zeigt den „Zustand“ des Roboterarms und der Control-Box an.

Der linke Teil des Bildschirms zeigt Informationen im Zusammenhang mit der Control-Box an, während auf der rechten Bildschirmseite Informationen zu den Robotergelenken angezeigt werden. Jedes Robotergelenk zeigt Informationen zur Motortemperatur und Elektronik, zur Belastung am Gelenk und zur elektrischen Spannung an.

**Roboterprotokoll** In der unteren Bildschirmhälfte werden Meldungen angezeigt. Die erste Spalte kategorisiert den Schweregrad des Protokolleintrags. In der zweiten Spalte ist der Zeitpunkt des Eintreffens der jeweiligen Meldungen eingetragen. Die folgende Spalte zeigt den Sender einer Meldung. Die letzte Spalte zeigt die eigentliche Meldung. Meldungen können durch Auswahl der Schaltflächen, die zu dem Schweregrad des Protokolleintrags gehören, gefiltert werden. Die Abbildung oben zeigt, dass Fehler angezeigt werden, während Informations- und Warnmeldungen gefiltert werden. Einige Protokollmeldungen sind darauf ausgelegt, weitere Informationen zu bieten, auf die durch Auswahl des Protokolleintrags zugegriffen werden kann.

### 13.16.1 Fehlerberichte speichern

Tritt in PolyScope ein Fehler auf, wird ein Protokolleintrag für den Fehler generiert. Im Register Protokoll können Sie generierte Berichte nachverfolgen und/oder an ein USB-Laufwerk exportieren (siehe 13.16). Die folgende Liste mit Fehlern kann nachverfolgt und exportiert werden:

- Störung

### 13.17 „Laden“ - Anzeige

- Interne PolyScope Ausnahmen
- Sicherheitsstopp
- Nicht abgefangener Ausnahmefehler in URCap
- Verletzung

Der exportierte Bericht enthält ein Benutzerprogramm, ein Journalprotokoll, eine Installation und eine Liste mit ausgeführten Diensten.

**Fehlerbericht** Wenn ein Büroklammersymbol in der Protokollzeile erscheint, steht ein ausführlicher Statusbericht zur Verfügung.

- Wählen Sie die Protokollzeile aus und tippen Sie auf die Option Bericht speichern, um den Bericht auf einem USB-Laufwerk zu speichern.
- Der Bericht kann bei laufendem Programm gespeichert werden.



**HINWEIS:**

Der jeweils älteste Bericht wird gelöscht wenn ein neuer generiert wird. Nur die aktuellsten fünf Berichte werden gespeichert.

### 13.17 „Laden“ - Anzeige

Mithilfe dieses Bildschirms wählen Sie, welches Programm Sie laden wollen. Es gibt zwei Versionen dieses Bildschirms: eine, die verwendet wird, wenn Sie ein Programm laden und ausführen möchten und eine, die verwendet wird, wenn Sie ein Programm bearbeiten möchten.



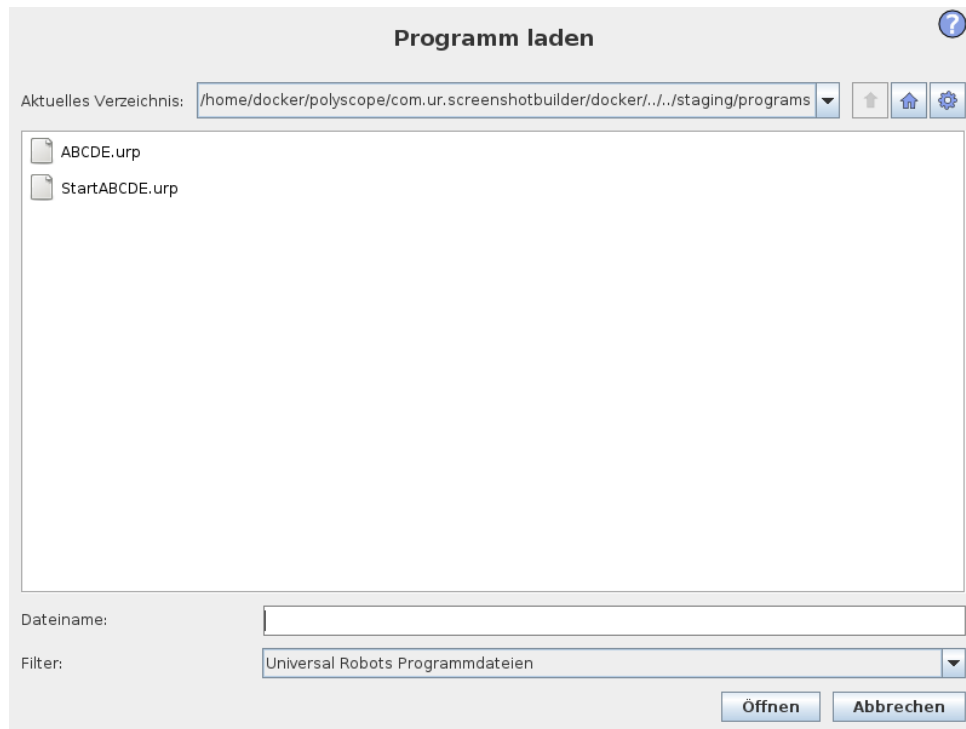
**HINWEIS:**

Das Ausführen eines Programms von einem USB-Laufwerk aus wird nicht empfohlen. Um ein Programm auszuführen, das auf einem USB-Laufwerk gespeichert ist, laden Sie es zuerst und speichern Sie es dann im lokalen Ordner `Programme` mithilfe der Option **Speichern als...** im **Datei**-Menü.

Der Hauptunterschied liegt darin, welche Aktionen dem Benutzer zur Verfügung stehen. Im Grundbildschirm „Laden“ kann der Benutzer lediglich auf Dateien zugreifen, sie jedoch nicht bearbeiten oder löschen. Weiterhin kann der Benutzer die Verzeichnisstruktur, die vom Ordner `Programme` ausgeht, nicht verlassen. Der Benutzer kann in ein Unterverzeichnis wechseln, aber er kann nicht über den Ordner `Programme` hinaus gelangen.

Deshalb sind alle Programme in den Ordner „Programme“ und/oder in Unterordner unter dem Ordner „Programme“ zu speichern.

## Layout des Bildschirms



Die Abbildung zeigt den eigentlichen Bildschirm „Laden“. Er besteht aus den folgenden wichtigen Bereichen und Schaltflächen:

**Pfadhistorie** Die Pfadhistorie zeigt eine Liste der Pfade, die zum aktuellen Ort führen. Das bedeutet, dass alle übergeordneten Verzeichnisse bis zum Root-Verzeichnis des Computers angezeigt werden. Sie werden hier feststellen, dass Sie vielleicht nicht auf alle Verzeichnisse über dem Ordner „Programme“ zugreifen können.

Mit der Auswahl eines Ordernamens in der Liste, wechselt der Ladedialog zu diesem Verzeichnis und zeigt es im Dateiauswahlbereich (siehe 13.17) an.

**Dateiauswahlbereich** In diesem Bereich des Dialogfensters werden die Inhalte des eigentlichen Bereiches angezeigt. Es gibt dem Benutzer die Möglichkeit, eine Datei durch einfachen Klick auf ihren Namen auszuwählen oder eine Datei durch Doppelklick auf ihren Namen zu öffnen.

Verzeichnisse werden durch längeres Drücken von ungefähr 0,5 Sek. ausgewählt. Zugriff auf einen Ordner und seinen Inhalt erfolgt durch Einzelklick.

**Dateifilter** Durch die Verwendung des Dateifilters kann man die angezeigten Dateien so begrenzen, dass nur die gewünschten Dateitypen angezeigt werden. Durch Auswahl von **Backup-Dateien** erscheint die Anzeige des Dateiauswahlbereichs der neuesten 10 gespeicherten Versionen der einzelnen Programme, wobei `.old0` die neueste und `.old9` die älteste ist.

### 13.18 Der „Ausführen“-Tab

**Dateifeld** Hier wird die aktuell ausgewählte Datei angezeigt. Der Benutzer hat die Option, den Dateinamen per Hand einzugeben, indem er auf das Tastatursymbol rechts auf dem Feld klickt. Dadurch wird eine Bildschirmtastatur angezeigt, mit der man den Dateinamen direkt auf dem Bildschirm eingeben kann.

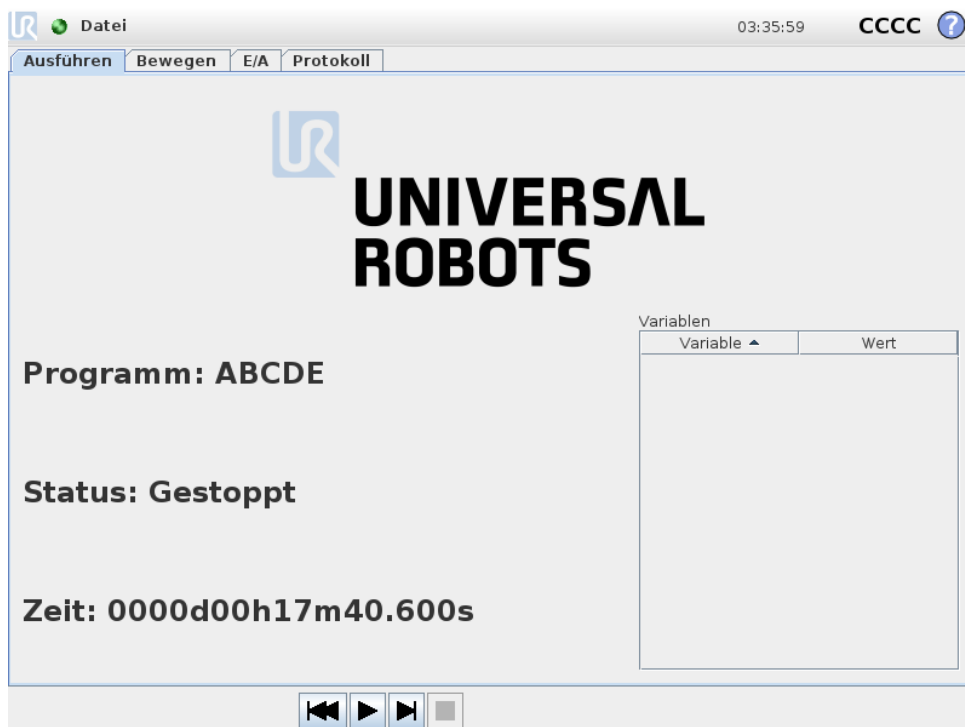
**Schaltfläche „Öffnen“** Durch Anklicken der Schaltfläche „Öffnen“ öffnet sich die aktuell ausgewählte Datei und das System kehrt zum vorhergehenden Bildschirm zurück.

**Schaltfläche „Abbrechen“** Durch Anklicken der Schaltfläche „Abbrechen“ wird der aktuelle Ladevorgang abgebrochen und der Bildschirm wechselt auf die vorhergehende Ansicht.

**Aktionsschaltflächen** Eine Reihe von Schaltflächen bietet dem Benutzer die Möglichkeit, die Handlungen vorzunehmen, die in der Regel durch Rechtsklick auf einen Dateinamen in einem herkömmlichen Dateidialog verfügbar sind. Zusätzlich gibt es die Möglichkeit zum Wechsel nach oben in die Verzeichnisstruktur und direkt in den Ordner „Programme“.

- Parent: Wechsel nach oben in der Verzeichnisstruktur. Die Schaltfläche ist in zwei Fällen nicht aktiviert: wenn das aktuelle Verzeichnis das oberste Verzeichnis ist oder wenn der Bildschirm im begrenzten Modus läuft und das aktuelle Verzeichnis der Ordner „Programm“ ist.
- Gehe zu Ordner „Programme“: Zum Ausgangsbildschirm zurück
- Aktionen: Aktionen wie beispielsweise Verzeichnis erstellen, Datei löschen usw.

### 13.18 Der „Ausführen“-Tab



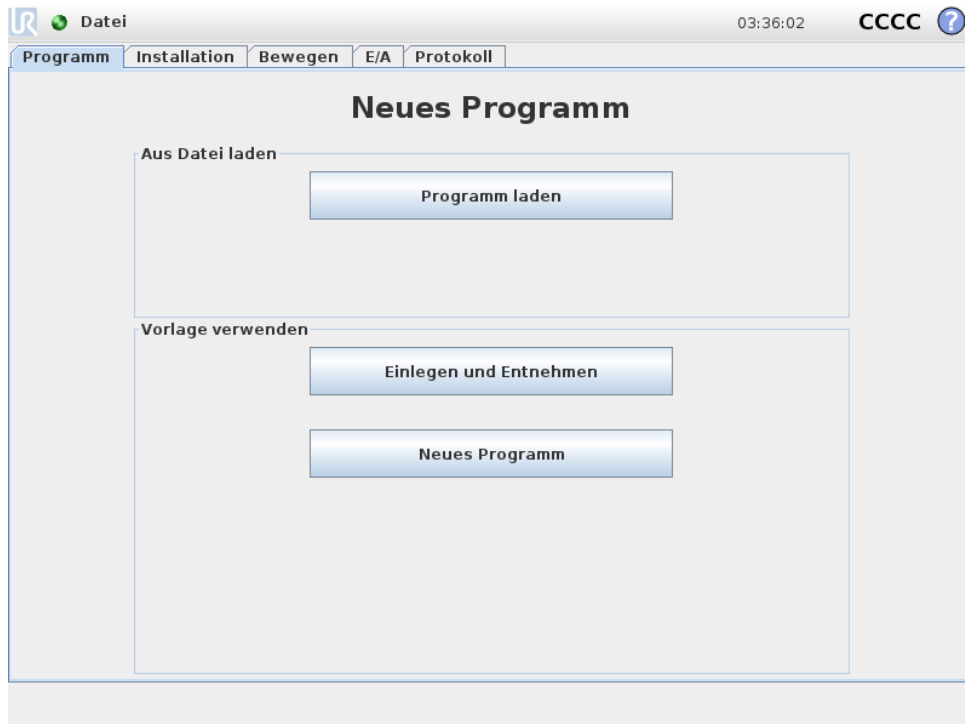
Dieser Tab bietet einen sehr einfachen Weg zur Bedienung des Roboterarms und des Steuergeräts mit so wenig Schaltflächen und Optionen wie möglich. Dies kann sinnvoll mit einem

Passwort kombiniert werden, das den Programmiererteil von PolyScope schützt (siehe 15.3), um den Roboter zu einem Werkzeug zu machen, das ausschließlich vorab geschriebene Programme ausführt.

Des Weiteren kann ein Standardprogramm in diesem Tab, basierend auf dem Flankenübergang eines externen Eingangssignals, automatisch geladen und gestartet werden (siehe 13.15). Die Kombination von automatischem Laden und Starten eines Standardprogramms und der Auto-Initialisierung beim Einschalten kann beispielsweise verwendet werden, um den Roboterarm in andere Maschinen zu integrieren.

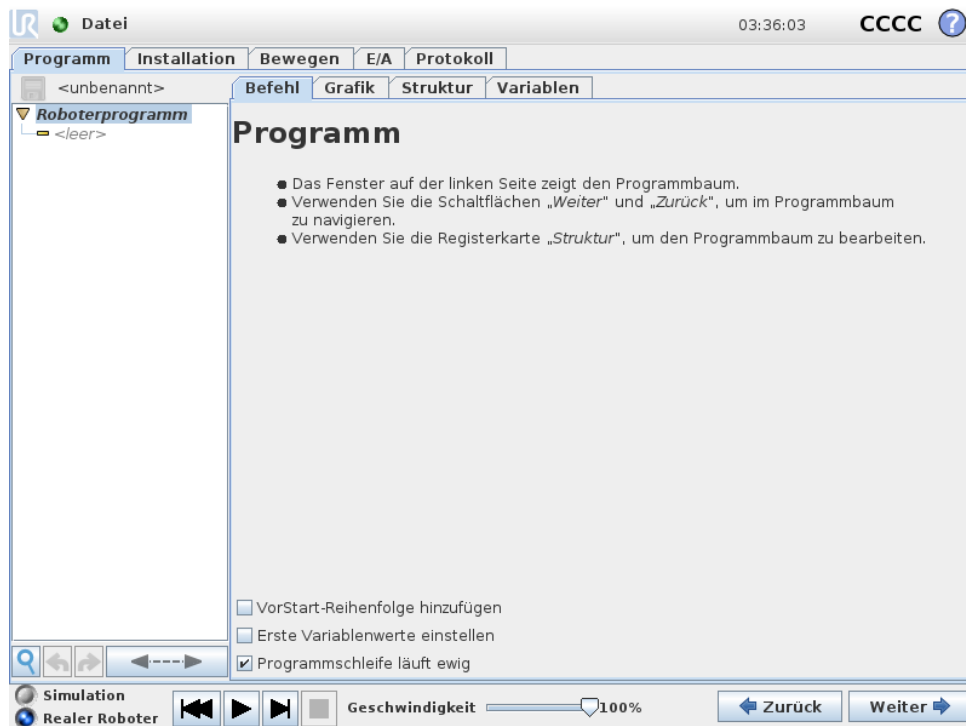
# 14 Programmierung

## 14.1 Neues Programm



Ein neues Roboterprogramm kann entweder von einer *Vorlage* oder von einem vorhandenen (gespeicherten) Roboterprogramm aus gestartet werden. Eine *Vorlage* kann die Gesamtprogrammstruktur bieten, sodass nur die Details des Programms ausgefüllt werden müssen.

## 14.2 Programm - Tab



Der Tab „Programm“ zeigt das aktuell bearbeitete Programm an.

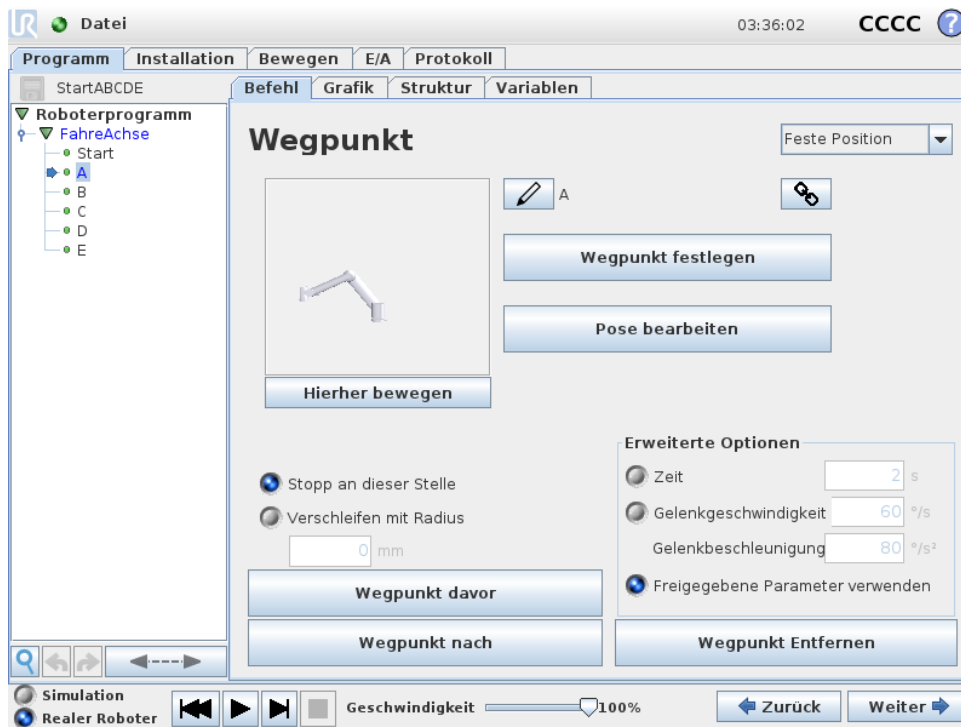
### 14.2.1 Programmstruktur




Die **Programmstruktur** auf der linken Bildschirmseite zeigt das Programm als Auflistung von Befehlen, während der Bereich auf der rechten Bildschirmseite Informationen im Zusammenhang mit dem aktuellen Befehl anzeigt.


Der aktuelle Befehl wird durch Anklicken der Befehlsliste bzw. über die Schaltflächen **Zurück** und **Weiter** unten rechts auf dem Bildschirm ausgewählt. Befehle können mithilfe des Tab **Struktur** eingegeben oder entfernt werden. Der Programmname erscheint direkt über der Befehlsliste mit einem kleinen Symbol, das zur schnellen Speicherung des Programms angeklickt werden kann.

In der Programmstruktur ist der Befehl markiert, der gerade ausgeführt wird. (wie in 14.2.2 beschrieben).



## 14.2.2 Programmausführungsanzeige





Die **Programmstruktur** enthält visuelle Hinweise hinsichtlich des Befehls, den der Controller des Roboters gerade ausführt. Ein kleines  Anzeigesymbol auf der linken Seite des Befehlsymbols wird angezeigt und der Name des gerade ausgeführten Befehls inkl. aller Befehle, von denen dieser Befehl ein Teilbefehl ist (in der Regel durch die Befehlssymbole  /  erkennbar) ist blau markiert. Dies hilft dem Anwender den ausgeführten Programmbefehl in der Struktur zu identifizieren.

Bewegt sich beispielsweise der Roboterarm in Richtung eines Wegpunktes, so wird der entsprechende Teilbefehl des Wegpunktes mit dem -Symbol markiert und sein Name wird, zusammen mit dem zugehörigen Namen des Move-Befehls (siehe 14.5) blau angezeigt.



Wenn das Programm angehalten wird, markiert die Programmausführungsanzeige den letzten Befehl, der gerade durchgeführt wurde.

Durch Klicken auf die untere Schaltfläche mit dem Symbol  springt die Programmstruktur auf den aktuell oder zuletzt ausgeführten Befehl in der Struktur. Wenn ein Befehl angeklickt wird, während ein Programm läuft, zeigt der **Befehl**-Tab die Information zu dem ausgewählten Befehl an. Durch Drücken der Taste  zeigt der **Befehl**-Tab weitere fortlaufende Informationen über die aktuell ausgeführten Befehle.

### 14.2.3 Schaltfläche „Suchen“

Tippen Sie auf , um eine Textsuche in der Programmstruktur durchzuführen. Nach einem Klick kann ein Suchtext eingegeben werden, woraufhin die entsprechenden Programmknoten gelb hervorgehoben erscheinen. Zusätzlich stehen Navigationstasten zur Verfügung, um durch die Treffer zu navigieren. Drücken Sie das Symbol , um die Suchfunktion zu verlassen. Hinweis: Die Programmstruktur muss erweitert werden, um auf die zusätzlichen Navigationstasten zuzugreifen.

## 14.2.4 Rückgängig/Erneut ausführen - Taste

Die Tasten mit den Symbolen  und  in der Werkzeugleiste unterhalb der Programmstruktur dienen dazu, in der Programmstruktur vorgenommene Änderungen und darin enthaltene Befehle rückgängig zu machen bzw. erneut auszuführen.

## 14.2.5 Programm-Dashboard

Der unterste Teil des Bildschirms ist das *Dashboard*. Das *Dashboard* verfügt über Schaltflächen, die einem traditionellen Kassettenrekorder ähneln, mit denen Programme gestartet und gestoppt, einzeln durchgegangen und neu gestartet werden können. Der *Geschwindigkeitsregler* ermöglicht Ihnen die Anpassung der Programmgeschwindigkeit zu jeder Zeit, was sich direkt auf die Geschwindigkeit auswirkt, mit der sich der Roboterarm bewegt. Zusätzlich zeigt der *Geschwindigkeitsregler* in Echtzeit und unter Einbeziehung der Sicherheitseinstellungen die relative Geschwindigkeit an, in der sich der Roboterarm bewegt. Der angezeigte Prozentsatz im laufenden Programm zeigt die maximal mögliche Geschwindigkeit an, ohne die Grenzwerte des Sicherheitssystems zu überschreiten.

Mit den Tasten links vom *Dashboard* kann zwischen der Ausführung des Programms in einer *Simulation* oder dem echten *Roboter* hin- und hergeschaltet werden. Bei einer *Simulation* bewegt sich der Roboterarm nicht und kann deshalb keinerlei Schäden verursachen. Verwenden Sie die *Simulationsfunktion* zum Testen von Programmen, wenn Sie sich bzgl. der Bewegungen des Roboterarms unsicher sind.



### GEFAHR:

1. Stellen Sie sicher, dass Sie sich außerhalb des Wirkungsbereichs des Roboters befinden, wenn die Taste *Abspielen* gedrückt wird. Die von Ihnen programmierte Aktivität könnte von der erwarteten Bewegung abweichen.
2. Verwenden Sie die Taste *Schritt* nur, wenn absolut notwendig. Stellen Sie sicher, dass Sie sich außerhalb des Wirkungsbereichs des Roboters befinden, wenn die Taste *Step* gedrückt wird.
3. Stellen Sie sicher, dass Sie Ihr Programm immer prüfen, indem Sie die Geschwindigkeit mithilfe des Geschwindigkeitsreglers reduzieren. Logische Programmierfehler des Integrators können unerwartete Bewegungen des Roboterarms verursachen.
4. Nachdem eine Notabschaltung oder ein Schutzstopp aufgetreten ist, stoppt das Roboterprogramm. Es kann fortgesetzt werden, solange sich kein Gelenk mehr als 10° bewegt hat. Mit dem Fortsetzen fährt der Roboter langsam auf seine Bahn zurück und führt das Programm fort.

Während das Programm geschrieben wird, wird die daraus folgende Bewegung des Roboterarms mithilfe einer 3D-Zeichnung im Tab *Grafik* dargestellt (wie in 14.29 beschrieben).

## 14.3 Variablen

Neben jedem Programmbefehl befindet sich ein kleines rotes, gelbes oder grünes Symbol. Ein rotes Symbol deutet auf einen Fehler in diesem Befehl, gelb weist darauf hin, dass der Befehl nicht abgeschlossen ist und grün steht für eine ordnungsgemäße Eingabe. Ein Programm kann erst ausgeführt werden, wenn alle Befehle grün angezeigt sind.

### 14.3 Variablen

Ein Roboterprogramm kann Variablen nutzen, um während der Laufzeit verschiedene Werte zu aktualisieren. Es stehen zwei Arten von Variablen zur Verfügung:

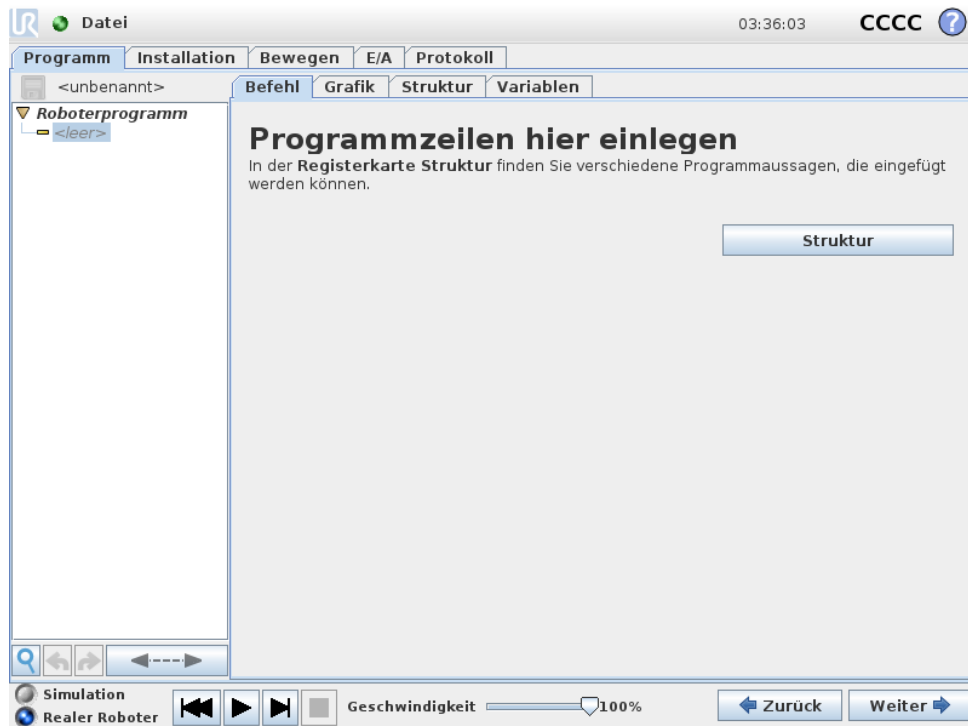
*Installationsvariablen:* Diese können von mehreren Programmen verwendet werden und ihre Namen und Werte bleiben zusammen mit der Roboterinstallation bestehen (siehe 13.10). Installationsvariablen behalten ihren Wert, auch nachdem Roboter und die Kontrolleinheit neu gestartet wurden.

*Normale Programmvariablen:* Diese stehen nur dem laufenden Programm zur Verfügung und ihre Werte gehen verloren, sobald das Programm gestoppt wird.

*Variablentypen:*

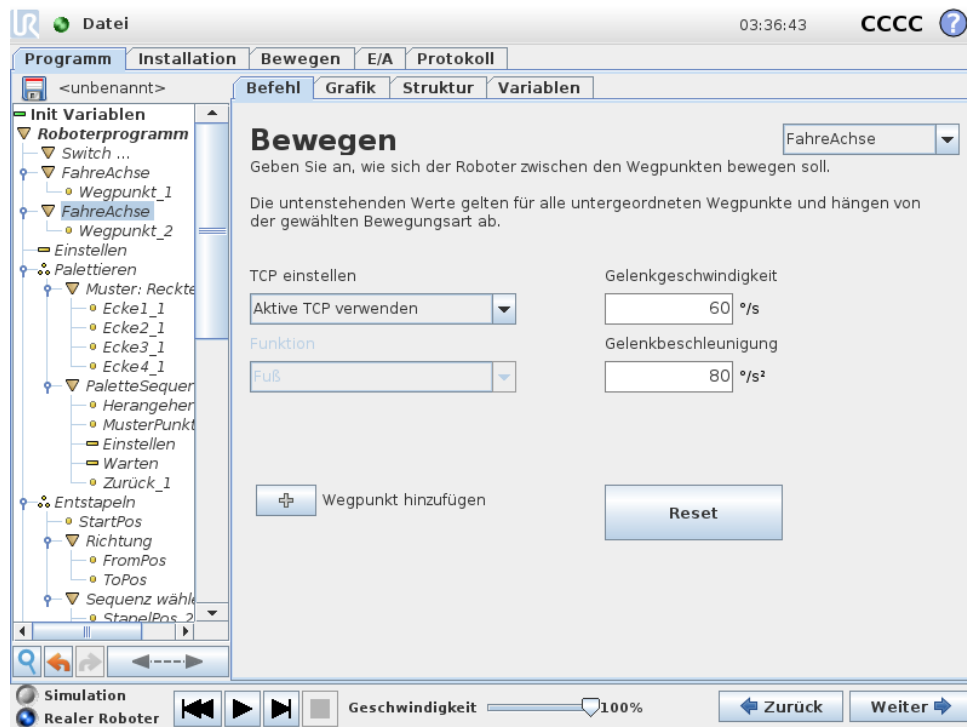
<i>Boole</i>	Eine Boole'sche Variable, deren Wert entweder <code>True</code> (wahr) oder <code>False</code> (falsch) ist.
<i>int</i>	Eine Ganzzahl im Bereich von $-2147483648$ bis $2147483647$ (32 Bit).
<i>Float</i>	Eine Gleitkommazahl (dezimal) (32 Bit).
<i>String</i>	Eine Sequenz von Zeichen.
<i>Pose</i>	Ein Vektor, der die Lage und Ausrichtung im Kartesischen Raum beschreibt. Er ist eine Kombination aus einem Positionsvektor $(x, y, z)$ und einem Rotationsvektor $(rx, ry, rz)$ , der die Ausrichtung darstellt; Schreibweise ist <code>p[x, y, z, rx, ry, rz]</code> .
<i>List</i>	Eine Sequenz von Variablen.

## 14.4 Befehl: Leer



Programmbefehle müssen hier eingegeben werden. Drücken Sie auf die Schaltfläche „**Struktur**“, um zum Tab „Struktur“ zu gelangen, in der die verschiedenen auswählbaren Programmzeilen zu finden sind. Ein Programm kann erst ausgeführt werden, wenn alle Zeilen vorgegeben und festgelegt sind.

## 14.5 Befehl: Move



Der **Move**-Befehl steuert die Roboterbewegung durch die zugrunde liegenden Wegpunkte. Wegpunkte müssen unter einem Move-Befehl vorhanden sein. Der Befehl „Move“ definiert die Beschleunigung und die Geschwindigkeit, mit der sich der Roboterarm zwischen diesen Wegpunkten bewegen wird.

### Bewegungsarten

Folgende drei Bewegungsarten stehen zur Auswahl: **MoveJ**, **MoveL** und **MoveP**. Jede Bewegungsart wird weiter unten erklärt.

- **moveJ** führt Bewegungen aus, die im **Gelenkraum** des Roboterarms berechnet werden. Jedes Gelenk wird so gesteuert, dass alle Gelenke die gewünschte Stellung gleichzeitig erreichen. Diese Bewegungsart sorgt für eine gekrümmte Bewegung des Werkzeugs. Die gemeinsamen Parameter, die für diese Bewegungsart gelten, sind die maximale Gelenkgeschwindigkeit und die Gelenkbeschleunigung und werden in  $deg/s$  bzw.  $deg/s^2$  angegeben. Wird gewünscht, dass sich der Roboterarm (ungeachtet der Bewegung des Werkzeugs zwischen diesen Wegpunkten) zwischen Wegpunkten schneller bewegt, ist diese Bewegungsart zu bevorzugen.
- **MoveL** sorgt für eine lineare Bewegung des Werkzeugmittelpunkts (TCP) zwischen Wegpunkten. Dies bedeutet, dass jedes Gelenk eine komplexere Bewegung ausführt, um die lineare Bewegung des Werkzeugs sicherzustellen. Die gemeinsamen Parameter, die für diese Bewegungsart eingestellt werden können, sind die gewünschte Werkzeuggeschwindigkeit und die Werkzeugbeschleunigung, angegeben in  $mm/s$  bzw.  $mm/s^2$ , und auch ein Merkmal.
- **MoveP** bewegt das Werkzeug linear bei konstanter Geschwindigkeit und kreisrunden Blending-Bewegungen und ist für Abläufe wie beispielsweise Kleben oder Ausgeben konzipiert. Die

Größe des Blending-Radius ist standardmäßig ein gemeinsamer Wert zwischen allen Wegpunkten. Ein kleinerer Wert sorgt für eine engere Kurve und ein größerer Wert sorgt für eine länger gezogene Kurve. Während sich der Roboterarm bei konstanter Geschwindigkeit durch die Wegpunkte bewegt, kann die Kontrolleinheit weder auf die Betätigung eines E/A, noch auf eine Eingabe durch den Bediener warten. Dadurch kann die Bewegung des Roboterarms eventuell angehalten oder ein Schutzstopp ausgelöst werden.

- **Circle move** kann zu einem **moveP** hinzugefügt werden, um eine Kreisbewegung zu bewirken. Der Roboter beginnt die Bewegung von seiner aktuellen Position oder seinem Startpunkt aus, bewegt sich durch einen auf der Kreisbahn definierten **Zwischenpunkt** und einen **Endpunkt**, der die Kreisbewegung vollendet.

Ein Modus wird verwendet, um die Werkzeugausrichtung durch den Kreisbogen zu berechnen. Mögliche Moduseinstellungen:

- Fest: Werkzeugausrichtung wird nur durch den Startpunkt definiert
- Uneingeschränkt: Der Startpunkt geht in den **Endpunkt** über, um die Werkzeugausrichtung festzulegen



Copyright ©2009–2018 by Universal Robots A/S. Alle Rechte vorbehalten.

### Gemeinsame Parameter

Die Einstellungen der gemeinsamen Parameter (unten rechts auf dem Move-Bildschirm) gelten für den Weg zwischen der vorherigen Position des Roboterarms und dem ersten Wegpunkt unter dem Befehl und von dort zu jedem weiteren der nachfolgenden Wegpunkte. Die Einstellungen des Move-Befehls gelten nicht für den Weg *von* letzten Wegpunkt unter diesem „Move“-Befehl.

### TCP-Auswahl

Der für die Wegpunkte im Rahmen dieses Fahrbefehls verwendete TCP kann aus dem Dropdown-Menü ausgewählt werden. Es können benutzerdefinierte TCPs aus der Installation, der aktive TCP oder einfach der Werkzeugflansch ausgewählt werden. Wenn ein benutzerdefinierter TCP

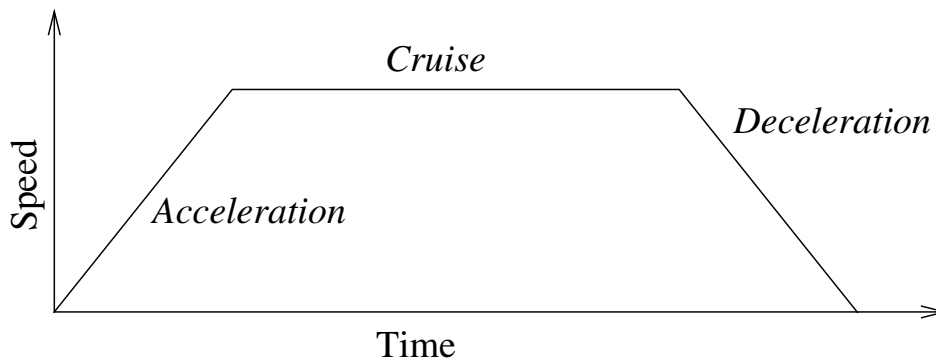


Abbildung 14.1: Geschwindigkeitsprofil für eine Bewegung. Die Kurve wird in drei Segmente unterteilt: *Beschleunigung*, *konstante Bewegung* und *Verlangsamung*. Die Ebene der *konstanten Bewegung* wird durch die Geschwindigkeitseinstellung der Bewegung vorgegeben, während der Anstieg und Abfall der Phasen in *Beschleunigung* und *Verlangsamung* durch den Beschleunigungsparameter vorgegeben wird.

oder der aktive TCP ausgewählt ist, wird die Bewegung im Rahmen dieses Fahrbefehls entsprechend angepasst. Wird **Werkzeugflansch verwenden** ausgewählt, so wird kein TCP verwendet und die Bewegung im Rahmen dieses Fahrbefehls erfolgt in Bezug auf den Werkzeugflansch (d. h., es erfolgt keine Anpassung der Bewegung).

Wird der aktive TCP für diese Bewegung während der Laufzeit des Programms festgelegt, muss er über den Set-Befehl dynamisch eingestellt werden (siehe 14.10) oder mittels Verwendung von Script-Befehlen. Weitere Informationen zu benannten TCP-Konfigurationen finden Sie hier (siehe 13.6).

### Auswahl von Funktionen

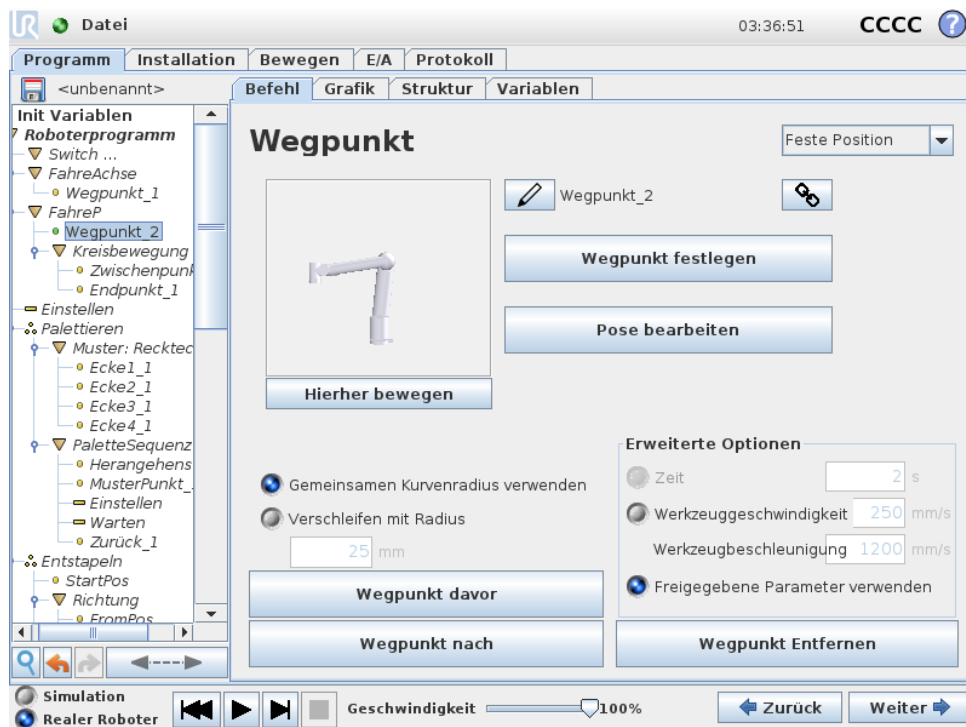
Die Funktion vergibt die abzubildenden Wegpunkte unter dem Befehl „Move“, wenn diese Wegpunkte festgelegt werden (siehe Abschnitt 13.12). Dies bedeutet, dass sich das Programm bei der Bestimmung eines Wegpunktes an die Werkzeugkoordinaten im Koordinatensystem der gewählten Funktion erinnert. Es gibt nur einige wenige Umstände, die einer detaillierteren Erläuterung bedürfen:

*Relative Wegpunkte:* Die ausgewählte Funktion hat keinen Einfluss auf die relativen Wegpunkte. Die relative Bewegung ist immer hinsichtlich der Orientierung zur **Basis** ausgerichtet.

*Variable Wegpunkte:* Bewegt sich der Roboterarm zu einem variablen Wegpunkt, wird der Werkzeugmittelpunkt (TCP) als die Koordinaten der Variablen im Raum der ausgewählten Funktion berechnet. Deshalb ändert sich die Roboterarmbewegung für einen variablen Wegpunkt, sobald eine andere Funktion ausgewählt wird.

*Variablen-Funktion:* Wird eine der Funktionen in der aktuell geladenen Installation als variabel ausgewählt, sind die entsprechenden Variablen ebenfalls im Menü zur Auswahl der Funktion wählbar. Wird eine Funktionsvariable (bezeichnet mit dem Namen der Funktion und nachgestelltem „\_var“) ausgewählt, sind die Roboterarmbewegungen (ausgenommen von **relativen Wegpunkten**) relativ zum Istwert der Variablen, solange das Programm läuft. Der Anfangswert einer Funktionsvariablen ist der Wert der eigentlichen Funktion, wie bei der Installation konfiguriert. Wenn dieser Wert verändert wird, ändern sich auch die Bewegungen des Roboters.

## 14.6 Befehl: Fester Wegpunkt



Ein Punkt auf der Bahn des Roboters. Wegpunkte sind der wichtigste Faktor eines Roboterprogramms, da sie die Positionen des Roboterarms bestimmen. Ein Wegpunkt mit einer festen Position wird angelernt, indem der Roboterarm physisch in die entsprechende Position bewegt wird.

### Festlegung des Wegpunktes

Betätigen Sie diese Taste, um zum Move-Bildschirm zu gelangen, über den Sie die Roboterarm-Position für diesen Wegpunkt vorgeben können. Wird der Wegpunkt unter einen Move-Befehl (**MoveL** oder **MoveP**) im linearen Raum gesetzt, muss ein Bezugskoordinatensystem für diesen Move-Befehl ausgewählt werden, damit diese Schaltfläche betätigt werden kann.

### Namen der Wegpunkte

Wegpunkte erhalten automatisch einen eindeutigen Namen. Der Name kann durch den Benutzer geändert werden. Wenn ein Link-Symbol ausgewählt wird, werden Wegpunkte verknüpft und Positionsinformationen geteilt. Andere Wegpunktinformationen wie Blending-Radius, Werkzeug-/Gelenkgeschwindigkeit und Werkzeug-/Gelenkbeschleunigung werden für jeden einzelnen Wegpunkt konfiguriert, auch wenn sie verknüpft sein könnten.

### Blending

Blending ermöglicht dem Roboter einen sanften Übergang zwischen zwei Bewegungsabläufen ohne am dazwischenliegenden Wegpunkt zu stoppen.

**Beispiel** Betrachten wir beispielsweise eine Pick-and-Place-Anwendung (siehe Abbildung 14.2), bei der sich der Roboter aktuell am Wegpunkt 1 (WP\_1) befindet und ein Objekt am Weg-

punkt 3 greifen (WP\_3) soll. Um Kollisionen mit dem Objekt und anderen Hindernissen (O) zu vermeiden, muss sich der Roboter WP\_3 aus der Richtung von Wegpunkt 2 kommend (WP\_2) nähern. Es werden also drei Wegpunkte für die Bahn einbezogen, um die Anforderungen zu erfüllen.

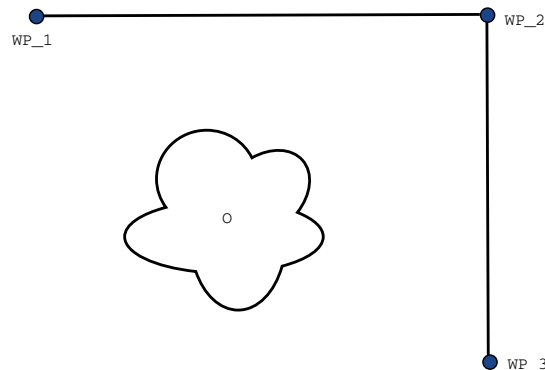


Abbildung 14.2: WP\_1: Ausgangsstellung, WP\_2: Zwischenziel, WP\_3: Aufnahmeposition, O: Hindernis.

Ohne die Konfiguration weiterer Einstellungen führt der Roboter an jedem Wegpunkt einen Stopp aus, bevor er seinen Bewegungsablauf fortsetzt. Für diese Aufgabenstellung ist ein Stopp bei WP\_2 nicht erwünscht, da mit einer reibungslosen Bewegung Zeit und Energie eingespart und die Anforderungen dennoch erfüllt werden. Es ist sogar zulässig, dass der Roboter WP\_2 nicht genau erreicht, solange der Übergang von Bewegungsablauf eins zu zwei nahe dieser Position stattfindet.

Der Stopp bei WP\_2 kann durch Konfigurieren eines Blending für den Wegpunkt vermieden werden und ermöglicht dem Roboter die Berechnung für einen reibungslosen Übergang zur nächsten Bewegung. Der primäre Parameter für das Blending ist ein Radius. Wenn sich der Roboter innerhalb des Blending-Radius des Wegpunktes befindet, kann er von der ursprünglichen Bahn abweichen. Dies ermöglicht schnellere und gleichmäßigere Bewegungen, da der Roboter weder abbremsen noch beschleunigen muss.

**Blending-Parameter** Neben den Wegpunkten beeinflussen mehrere Parameter den Bewegungsablauf im Blending-Bereich (siehe Abbildung 14.3):

- der Blending-Radius ( $r$ )
- die Anfangs- und Endgeschwindigkeit des Roboters (an Position  $p_1$  und entsprechend an  $p_2$ )
- die Bewegungsdauer (z. B. wenn eine bestimmte Dauer für einen Bewegungsablauf vorgegeben wird, beeinflusst dies die Anfangs-/Endgeschwindigkeit des Roboters)
- die Bewegungsart im Blending von bzw. zu ( $MoveL$ ,  $MoveJ$ )

Wird ein Blending-Radius eingestellt, so wird der Roboterarm um den Wegpunkt geführt, so dass der Roboterarm an dem Punkt nicht anhalten muss.

Blending-Bereiche können nicht überlappen, womit ausgeschlossen wird, dass ein eingestellter Blending-Radius mit einem Blending-Radius für einen vorhergehenden oder nachfolgenden Wegpunkt überlappt (siehe Abb. 14.4).

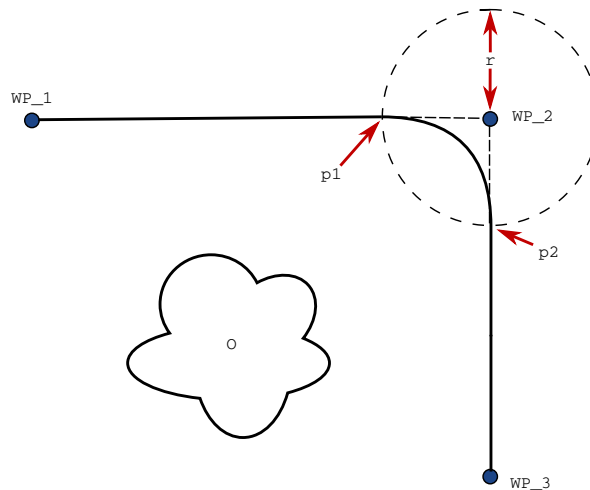


Abbildung 14.3: Blending über WP\_2 mit Radius  $r$ , ursprüngl. Blending-Position bei p1 und letzte Blending-Position bei p2. o ist ein Hindernis.

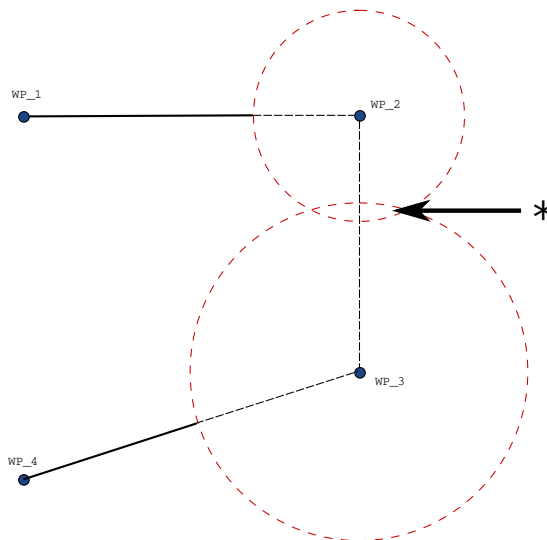


Abbildung 14.4: Blending-Radius-Überlappung nicht zulässig (\*).

**Bedingte Bewegungsabläufe im Blending-Bereich** Bewegungsabläufe im Blending-Bereich sind sowohl vom Wegpunkt, in dem der Blending-Radius festgelegt ist, als auch dem in der Programmstruktur nachfolgenden Wegpunkt abhängig. Das heißt, im Programm in Abbildung 14.5 ist der Blendingradius um WP\_1 von WP\_2 abhängig. Die Folge davon wird offensichtlicher, wenn das Blending wie in diesem Beispiel um WP\_2 stattfindet. Es gibt zwei mögliche Endpositionen. Um den nächsten Wegpunkt für das Blending zu bestimmen, muss der Roboter den aktuellen Wert von `digital_input[1]` bereits beim Eintritt in den Blending-Radius berechnen. Dies bedeutet, dass der Ausdruck `if...then` oder andere notwendige Anweisungen, die den folgenden Wegpunkt bestimmen (z. B. variable Wegpunkte) bereits ausgewertet werden, bevor wir bei WP\_2 tatsächlich ankommen. Bei Betrachtung des Programmablaufs klingt dies ein wenig unlogisch. Wenn es sich bei einem Wegpunkt um einen Wegpunkt ohne Blendingradius

handelt auf welchen beispielsweise einem If-else-Befehl folgt durch welchen (z. B. mit einem E/A-Befehl) der nächste Wegpunkt bestimmt wird, so wird die Prüfung ausgeführt, sobald der Roboterarm am Wegpunkt anhält.

```

MoveL
  WP_I
  WP_1 (Blending)
  WP_2 (Blending)
  if (digital_input[1]) then
    WP_F_1
  else
    WP_F_2

```

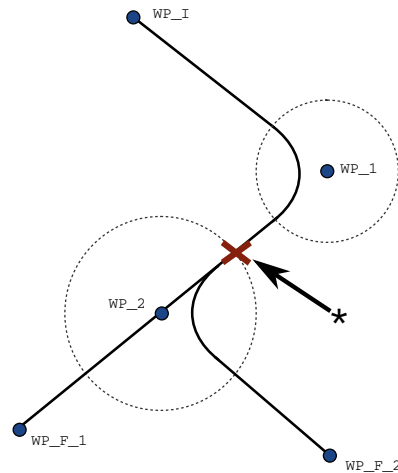


Abbildung 14.5: WP\_I ist der Ausgangswegpunkt und es gibt zwei mögliche endgültige Wegpunkte WP\_F\_1 und WP\_F\_2, je nach einem bedingten Ausdruck (if ... then). Die Bedingung if wird ausgewertet, sobald der Roboterarm den zweiten Übergang (\*) erreicht.

**Bewegungsarten in Kombinationen** Es ist möglich, alle vier Bewegungsarten von **MoveJ** und **MoveL** beim Blending zu kombinieren, hierbei hat die spezifische Kombination die entscheidende Auswirkung auf den berechneten Bewegungsablauf im Blending. Es gibt vier mögliche Kombinationen:

1. **MoveJ** zu **MoveJ** (Reines Blending im Gelenkraum)
2. **MoveJ** zu **MoveL**
3. **MoveL** zu **MoveL** (Reines Blending im kartesischen Raum)
4. **MoveL** zu **MoveJ**

Ein reines Blending im Gelenkraum (Punkt 1) im Vergleich zu einem auf den kartesischen Raum beschränkten Blending (Punkt 3) ist in Abbildung 14.6 dargestellt. Sie zeigt zwei mögliche Bahnen des Werkzeugs für jeweils identische Reihen von Wegpunkten auf.

Von den verschiedenen Kombinationen führen die Punkte 2, 3 und 4 zu Bewegungsabläufen, die innerhalb der Grenzen der ursprünglichen Bahn im kartesischen Koordinatensystem erfolgen. Ein Beispiel für ein Blending zwischen verschiedenen Bewegungsarten (Punkt 2) ist in Abbildung 14.7 zu sehen.

Ein Blending im Gelenkraum (Punkt 1) verhält sich jedoch in einer weniger intuitiven Weise, da der Roboter versuchen wird, der reibungslosesten Bahn im Gelenkraum unter Berücksichtigung von Geschwindigkeit und zeitlichen Anforderungen zu folgen. Aus diesem Grund können Bewegungen vom Kurs abweichen, der durch die Wegpunkte vorgeben ist. Dies ist insbesondere

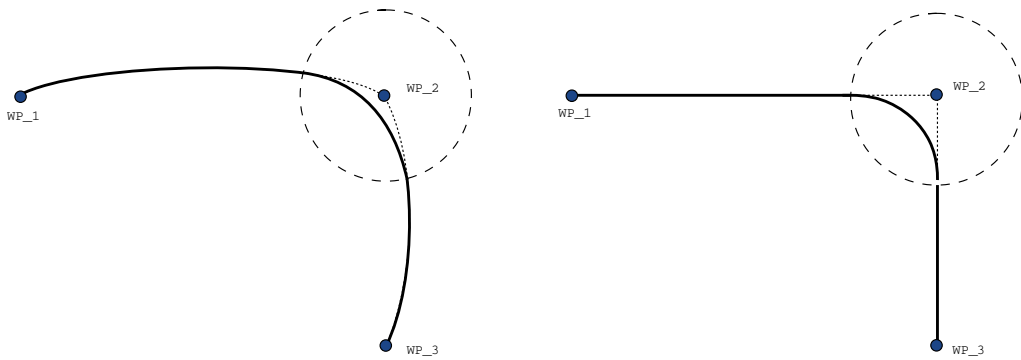


Abbildung 14.6: Bewegung und Blending im Gelenkraum (MoveJ) im Vgl. zum kartesischen Raum (MoveL) .

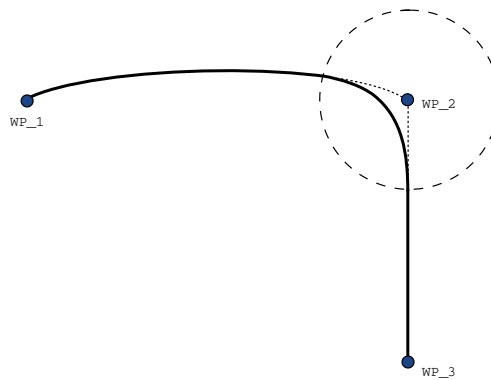


Abbildung 14.7: Blending von einer Bewegung im Gelenkraum (MoveJ) zu linearer Werkzeugbewegung (MoveL).

dann der Fall, wenn erhebliche Unterschiede zwischen den Gelenkgeschwindigkeiten der beiden Bewegungsabläufe bestehen. *Vorsicht:* Wenn sich die Geschwindigkeiten stark unterscheiden (z. B. durch die Angabe erweiterter Einstellungen von Geschwindigkeit oder Zeit für einen bestimmten Wegpunkt) so können dadurch größere Abweichungen vom ursprünglichen Bewegungsablauf (wie in Abbildung 14 . 8 dargestellt) die Folge sein. Falls verschiedene Geschwindigkeiten im Blending-Bereich erforderlich aber Bahnabweichungen nicht akzeptabel sind, sollte das Blending im kartesischen Raum mittels **MoveL** erfolgen.

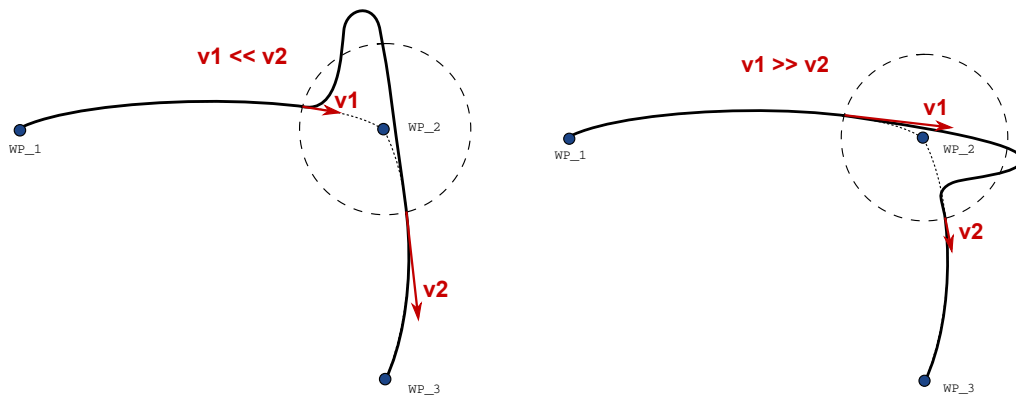


Abbildung 14.8: Blending im Gelenkraum bei erheblich niedrigerer Ausgangsgeschwindigkeit  $v_1$  im Vergleich zur Endgeschwindigkeit  $v_2$  oder umgekehrt.

## 14.7 Befehl: Relativer Wegpunkt

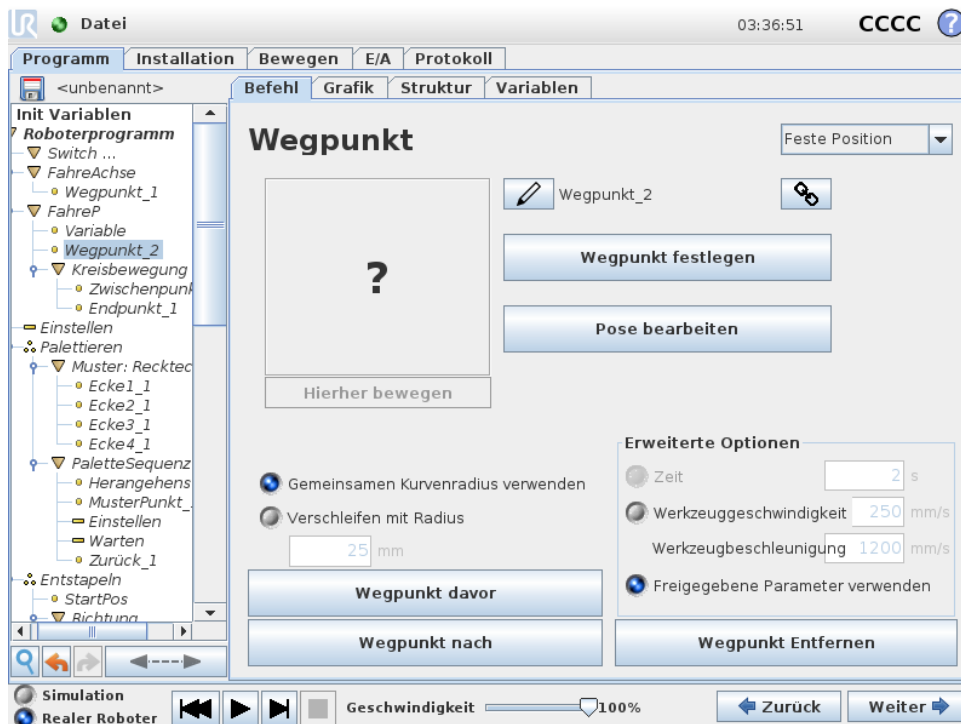


Ein Wegpunkt, dessen Position in Relation zur vorhergehenden Position des Roboterarms angegeben wird, wie z. B. „zwei Zentimeter nach links“. Die relative Position wird als Unterschied zwischen den beiden gegebenen Positionen festgelegt (links nach rechts).

Hinweis: Bitte beachten Sie, dass wiederholte relative Positionen den Roboterarm aus dessen Wirkungsbereich heraus bewegen können.

Der Abstand hier ist der kartesische Abstand zwischen dem TCP an beiden Positionen. Der Winkel gibt an, wie sehr die Ausrichtung des TCP sich zwischen beiden Positionen ändert. Genauer gesagt handelt es sich um die Länge des Rotationsvektors, welche die Ausrichtungsänderung angibt.

## 14.8 Befehl: Variabler Wegpunkt



Ein Wegpunkt, dessen Position durch eine Variable angegeben wird, in diesem Fall `berechnete_Pos`.

Die Variable muss eine *Pose* sein, wie beispielsweise

`var=p[0.5,0.0,0.0,3.14,0.0,0.0]`. Die ersten drei sind  $x,y,z$  und die letzten drei beschreiben die Ausrichtung als *Rotationsvektor*, der durch den Vektor  $rx,ry,rz$  vorgegeben wird. Die Länge der Achse entspricht dem zu drehenden Winkel in Radianten, und der Vektor selbst gibt die Achse an, um die die Drehung erfolgt. Die Position wird immer in Bezug auf einen Bezugsrahmen oder ein Koordinatensystem angegeben, definiert durch die ausgewählte Funktion. Wird ein Übergangsradius auf einen festen Wegpunkt gelegt, wobei der vorangegangene und nachfolgende Wegpunkt variabel ist, oder wird ein Übergangsradius auf einen Variable Wegpunkt gelegt, so wird der Übergangsradius nicht auf Überschneidungen geprüft (siehe 14.6). Überschneidet der Übergangsradius bei der Ausführung des Programms einen Punkt, so ignoriert der Roboter diesen und bewegt sich zum nächsten Punkt.

Beispielsweise, um den Roboter 20 mm entlang der z-Achse des Werkzeugs zu bewegen:

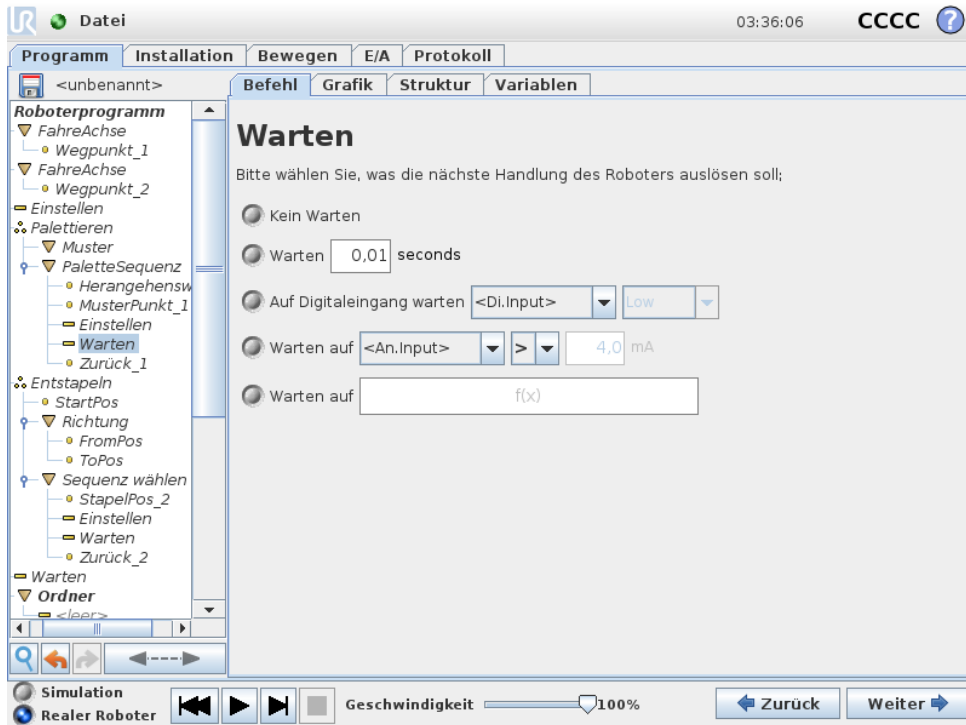
```
var_1=p[0,0,0.02,0,0,0]
```

```
MoveL
```

```
Wegpunkt_1 (Variableposition):
```

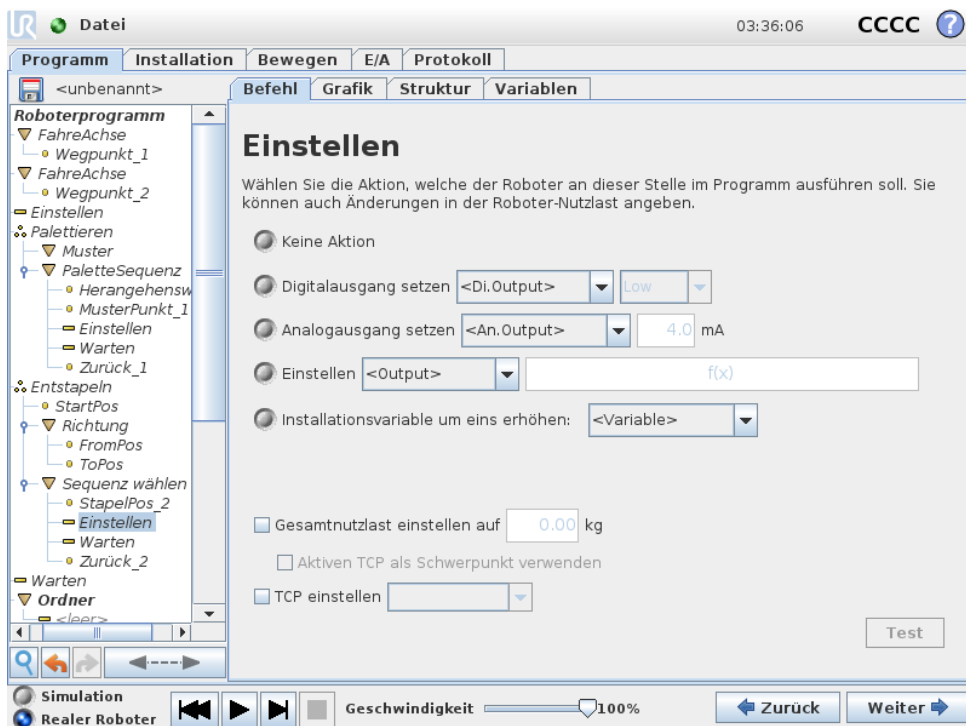
```
Verwenden Sie Variable=var_1, Funktion=Werkzeug
```

## 14.9 Befehl: Warten



**Warten** unterbricht das E/A-Signal oder den Ausdruck für eine bestimmte Zeit. Wird **Nicht warten** ausgewählt, erfolgt keine Maßnahme.

## 14.10 Befehl: Einstellen



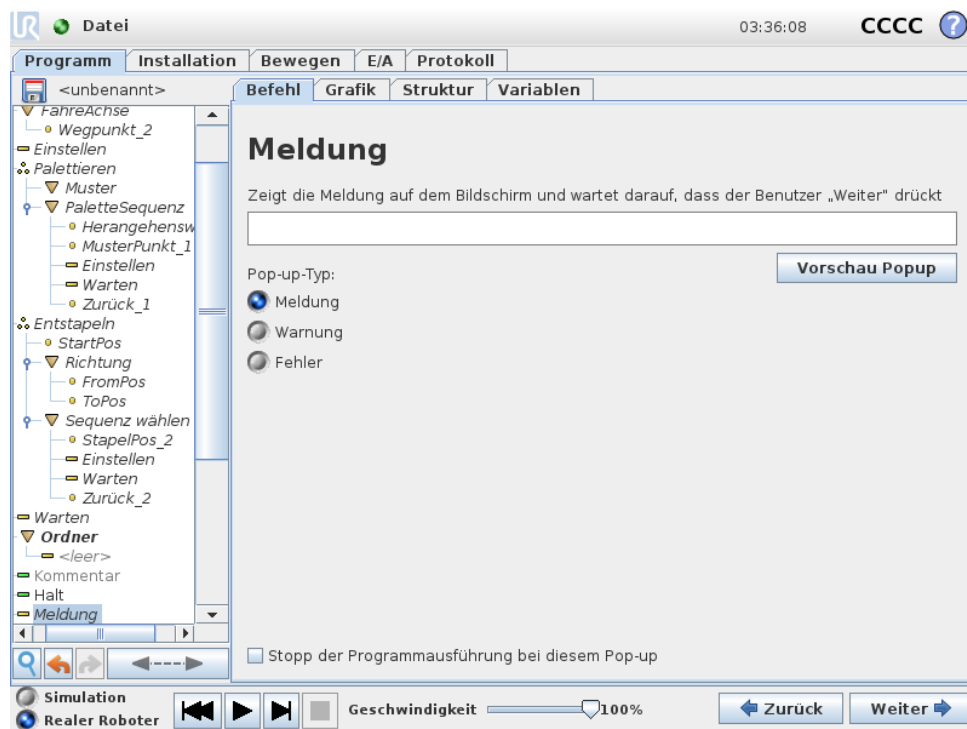
## 14.11 Befehl: Meldung

Setzt entweder digitale oder analoge Ausgänge auf einen vorgegebenen Wert.

Der Befehl kann ebenso zur Einstellung der Tragfähigkeit des Roboterarms eingesetzt werden. Eine Anpassung der Tragfähigkeit könnte erforderlich sein, um zu verhindern, dass der Roboter einen Schutzstopp auslöst, falls das Gewicht am Werkzeug vom erwarteten Gewicht abweicht. Der aktive TCP wird auch als Standardeinstellung für den Schwerpunkt verwendet. Soll der aktive TCP nicht den Schwerpunkt ausmachen, ist der Haken aus dem Kontrollkästchen zu entfernen.

Der aktive TCP kann auch über den Befehl **Einstellen** verändert werden. Setzen Sie einfach den Haken im Kontrollkästchen und wählen Sie eine der TCP-Offsets aus dem Menü. Ist das aktive TCP für eine bestimmte Bewegung zum Zeitpunkt der Programmierung bekannt, können Sie stattdessen die Verwendung der TCP-Auswahl auf der **Move**-Karte in Betracht ziehen (siehe 14.5). Weitere Informationen zu benannten TCP-Konfigurationen finden Sie hier (siehe 13.6).

## 14.11 Befehl: Meldung

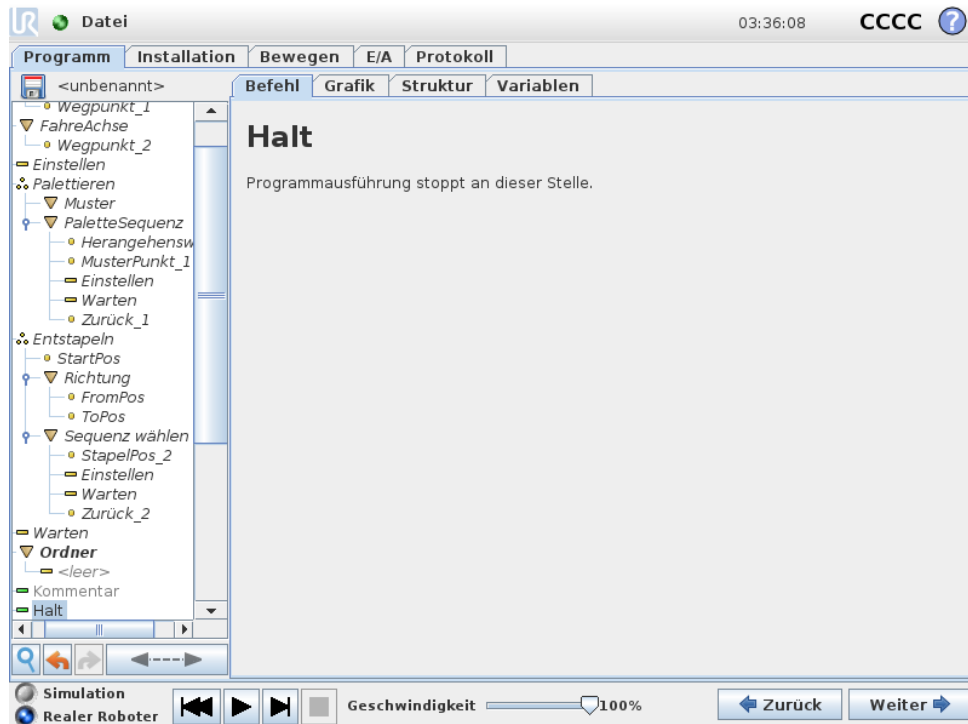


Eine Meldung ist ein Pop-up, das auf dem Bildschirm angezeigt wird, wenn das Programm diesen Befehl erreicht. Der Meldungsstil ist wählbar und der Text kann mithilfe der Tastatur auf dem Bildschirm eingegeben werden. Der Roboter wartet, bis der Benutzer/Bediener die Schaltfläche „O.K.“ unter dem Pop-up betätigt, bevor er mit dem Programm fortfährt. Wenn der Punkt „Programmausführung stoppen“ ausgewählt ist, hält das Programm bei dieser Meldung an.

Hinweis: Nachrichten werden auf maximal 255 Zeichen beschränkt.

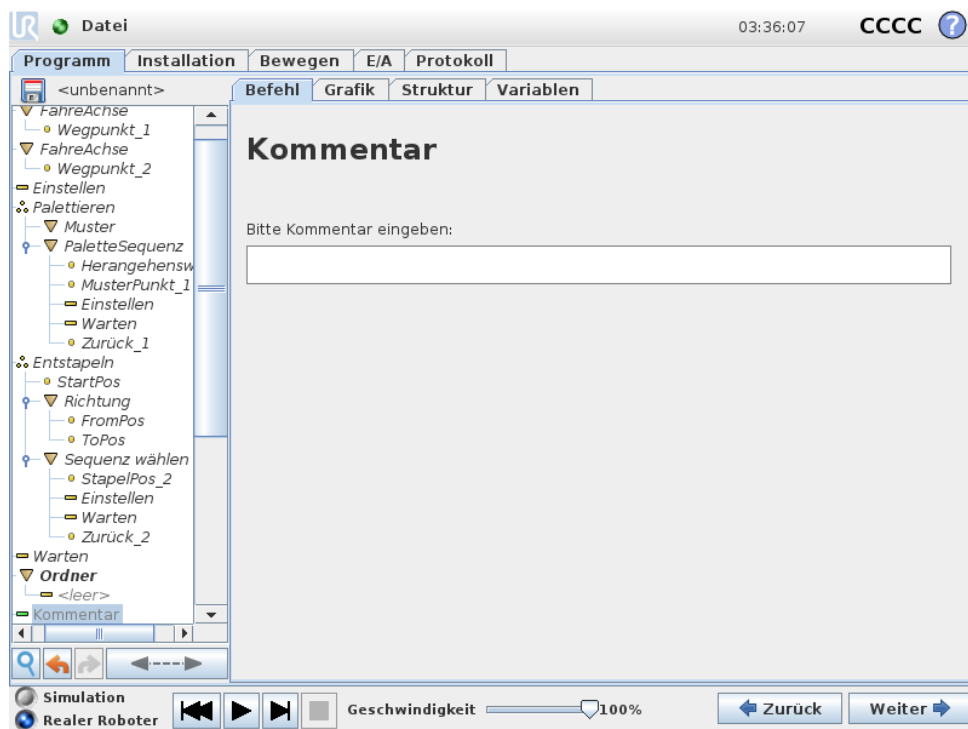
Copyright ©2009–2018 by Universal Robots A/S. Alle Rechte vorbehalten.

## 14.12 Befehl: Halt



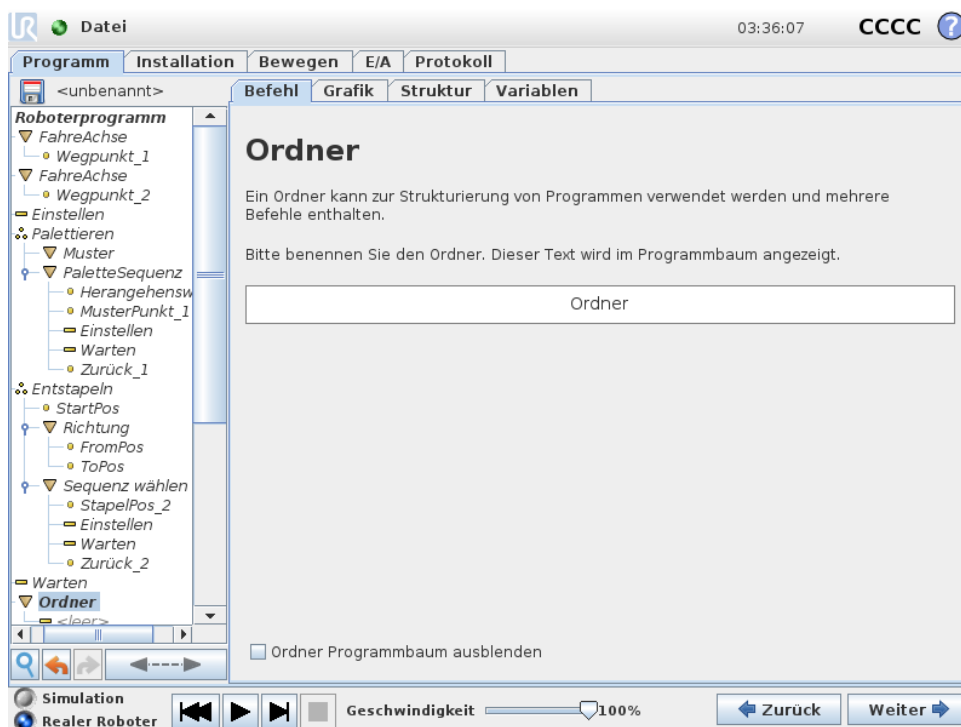
Die Ausführung des Programms wird an dieser Stelle angehalten.

## 14.13 Befehl: Kommentar



Hier erhält der Programmierer die Möglichkeit, das Programm durch eine Textzeile zu ergänzen. Diese Textzeile hat auf die Ausführung des Programms keinerlei Auswirkung.

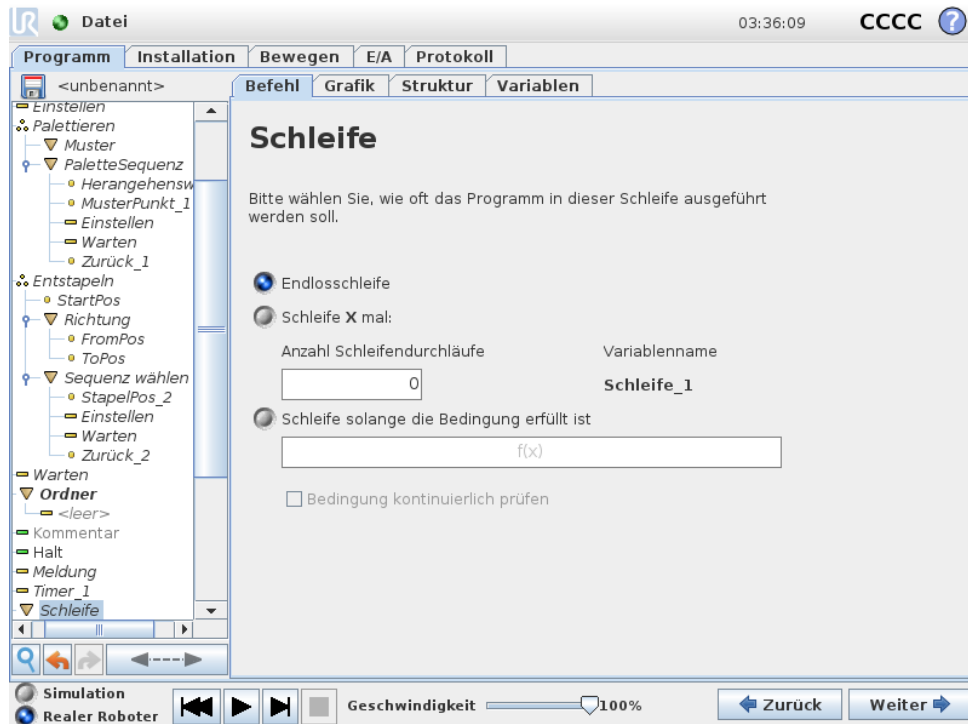
## 14.14 Befehl: Ordner



Ein **Ordner** wird zur Organisation und Kennzeichnung bestimmter Programmteile, zur Bereinigung der Programmstruktur und zur Vereinfachung des Lesens und Navigierens im Programm eingesetzt.

**Ordner** haben keine Auswirkungen auf das Programm und seine Ausführung.

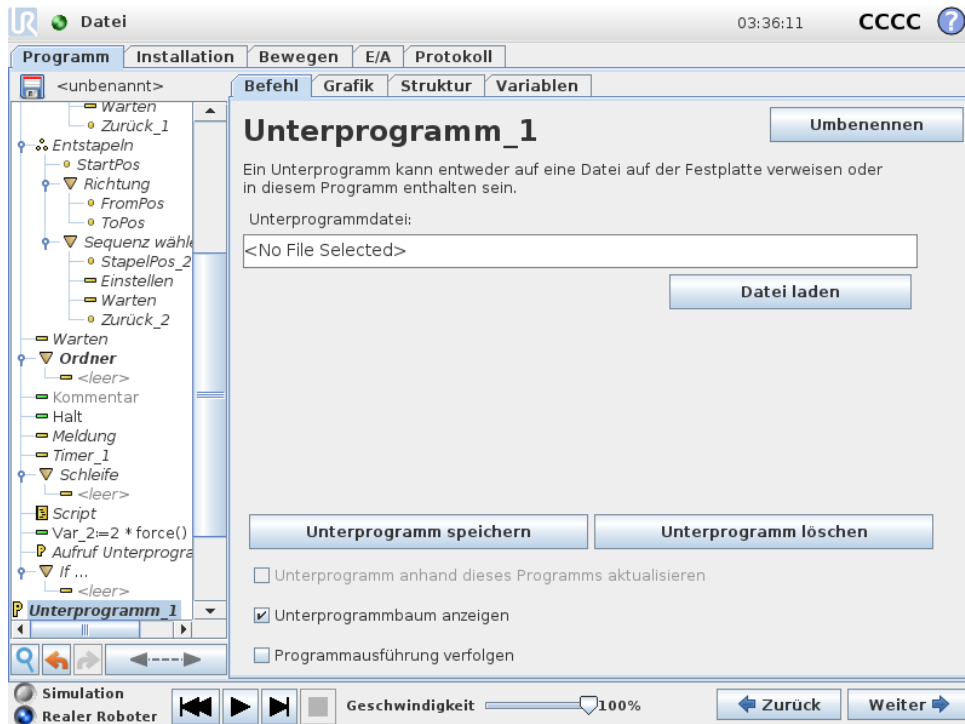
## 14.15 Befehl: Schleife



Die zugrunde liegenden Programmbefehle befinden sich in einer Schleife. In Abhängigkeit von der Auswahl werden die zugrunde liegenden Befehle entweder unbegrenzt, eine gewisse Anzahl oder solange wiederholt wie die vorgegebene Bedingung wahr ist. Bei der Wiederholung für eine bestimmte Anzahl wird eine fest zugeordnete Schleifenvariable (im vorherigen Screenshot `loop_1` genannt) erstellt, die in Ausdrücken innerhalb der Schleife eingesetzt werden kann. Die Schleifenvariable zählt von 0 bis  $N - 1$ .

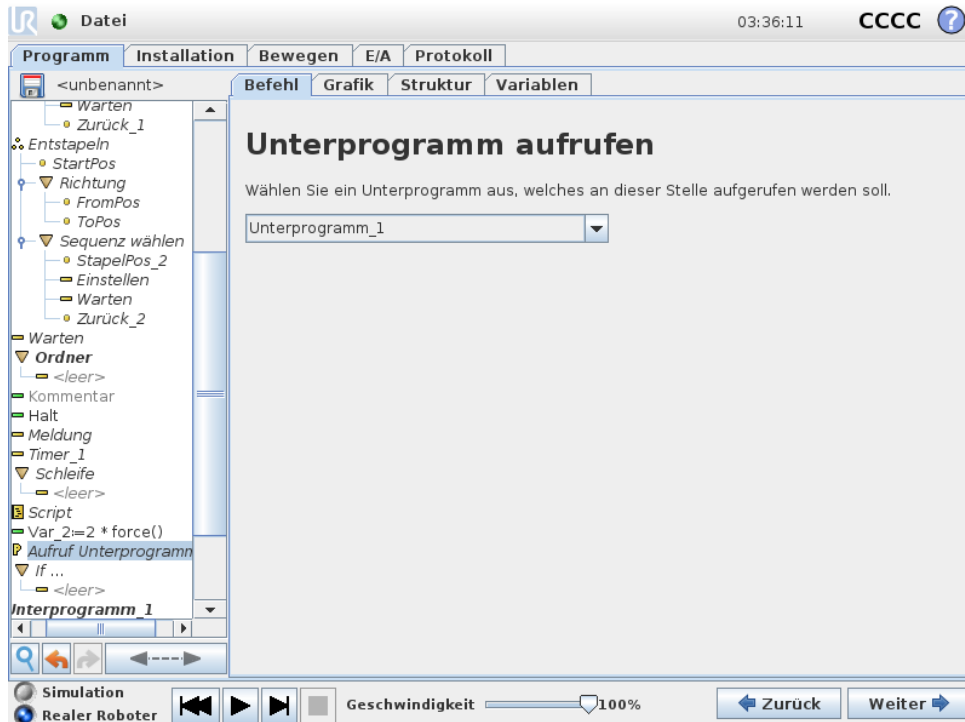
Bei der Erstellung von Schleifen mit einem Ausdruck als Endbedingung bietet PolyScope eine Option zur kontinuierlichen Bewertung dieses Ausdrucks, sodass der „loop“ jederzeit während der Ausführung unterbrochen werden kann, anstelle nach jedem Durchlauf.

## 14.16 Befehl: Unterprogramm



Ein Unterprogramm kann Programmteile enthalten, die an mehreren Stellen erforderlich sind. Ein Unterprogramm kann eine separate Datei auf der Diskette oder auch versteckt sein, um sie gegen ungewollte Änderungen am Unterprogramm zu schützen.

### Befehl: Unterprogramm aufrufen



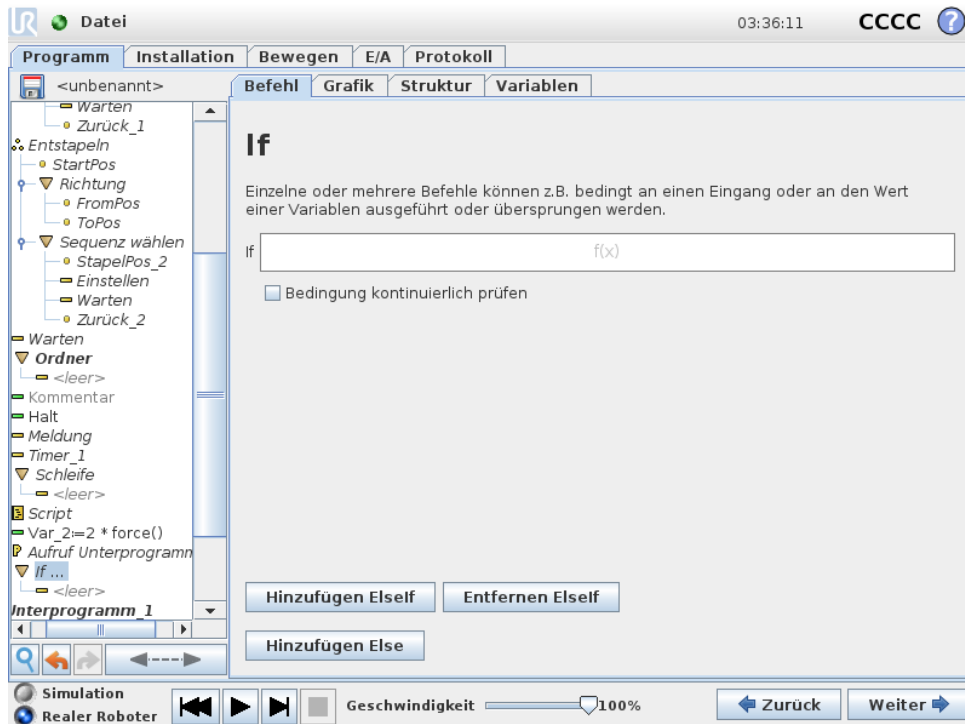
Wenn Sie ein Unterprogramm aufrufen, werden die Programmzeilen im Unterprogramm ausgeführt, bevor zur nächsten Zeile übergegangen wird.

## 14.17 Befehl: Zuordnung



Weist den Variablen Werte zu. Der berechnete Wert auf der rechten Seite wird der Variablen auf der linken Seite zugeordnet. Dies kann sich bei komplexen Programmen als nützlich erweisen.

## 14.18 Befehl: If

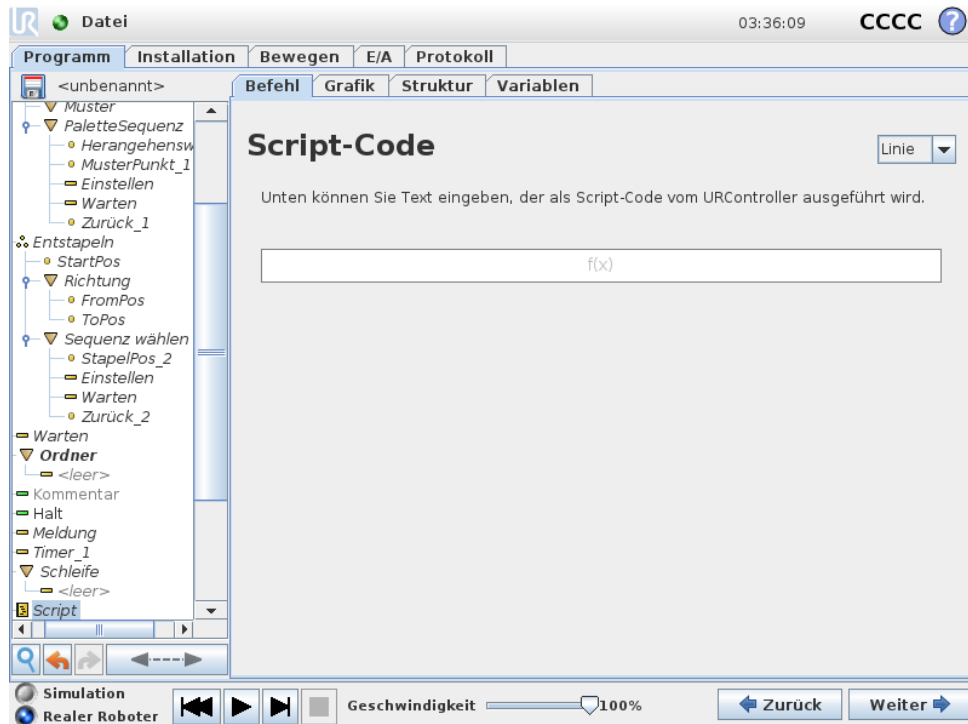


Durch einen **If...else**-Befehl kann der Roboter sein Verhalten aufgrund von Sensoreingängen oder Variablewerten ändern. Verwenden Sie den Ausdruckseditor, um die Bedingung zu beschreiben, in der der Roboter mit den Anweisungen dieses **If** Befehls fortfahren soll. Wenn die Bedingung mit True bewertet wird, werden die Anweisungen in diesem **If** Befehl ausgeführt.

Ein **If** Befehl kann mehrere **ElseIf**-Anweisungen haben, die über die Tasten **ElseIf hinzufügen** und **ElseIf entfernen**- hinzugefügt und entfernt werden können. Jedoch kann ein **If** Befehl nur eine **Else** Anweisung haben.

Hinweis: Durch Anklicken des Kontrollkästchens **Ausdruck kontinuierlich prüfen** wird die Bewertung der Bedingungen der **If** Befehle und **ElseIf** Anweisungen ermöglicht, während die darin enthaltenen Zeilen ausgeführt werden. Wenn ein Ausdruck innerhalb des **If** Befehls als Falsch bewertet wird, werden die Anweisungen **ElseIf** oder **Else** befolgt.

## 14.19 Befehl: Script



Dieser Befehl ermöglicht den Zugang zur zugrundeliegenden Echtzeit-Scriptsprache, die vom Controller des Roboters ausgeführt wird. Er ist nur für erfahrene Benutzer bestimmt und Anweisungen zu seiner Verwendung finden Sie im Skripthandbuch auf der Support-Webseite (<http://www.universal-robots.com/support>).

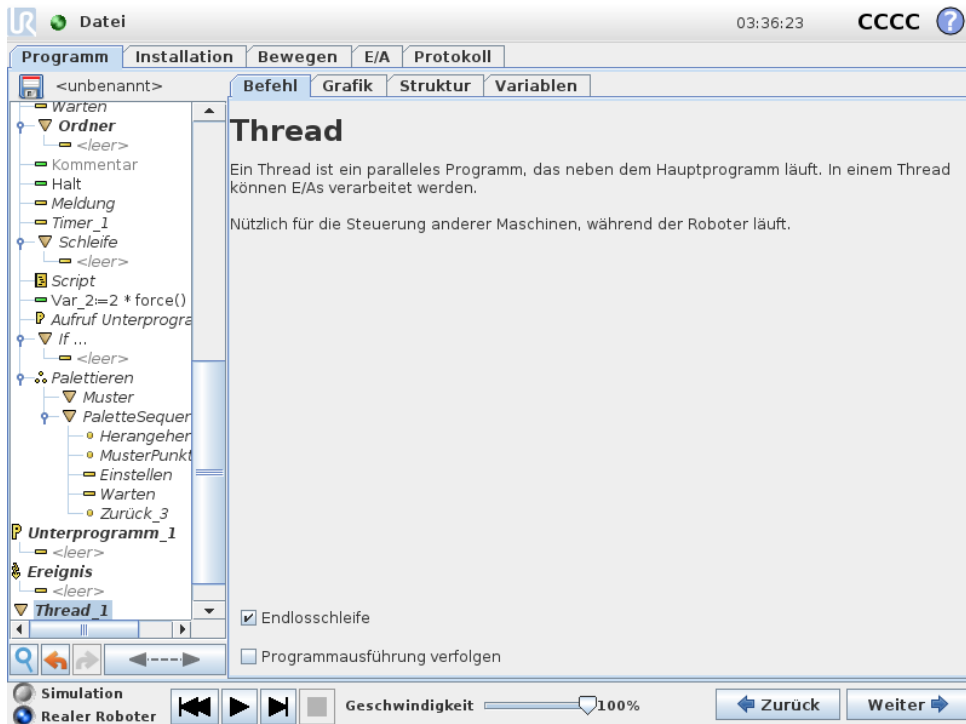
Mit der Option „File“ oben links können Skript-Programmdateien erstellt und bearbeitet werden. So können lange und komplexe Skript-Programme zusammen mit der bedienerfreundlichen Programmierung von PolyScope eingesetzt werden.

## 14.20 Befehl: Ereignis



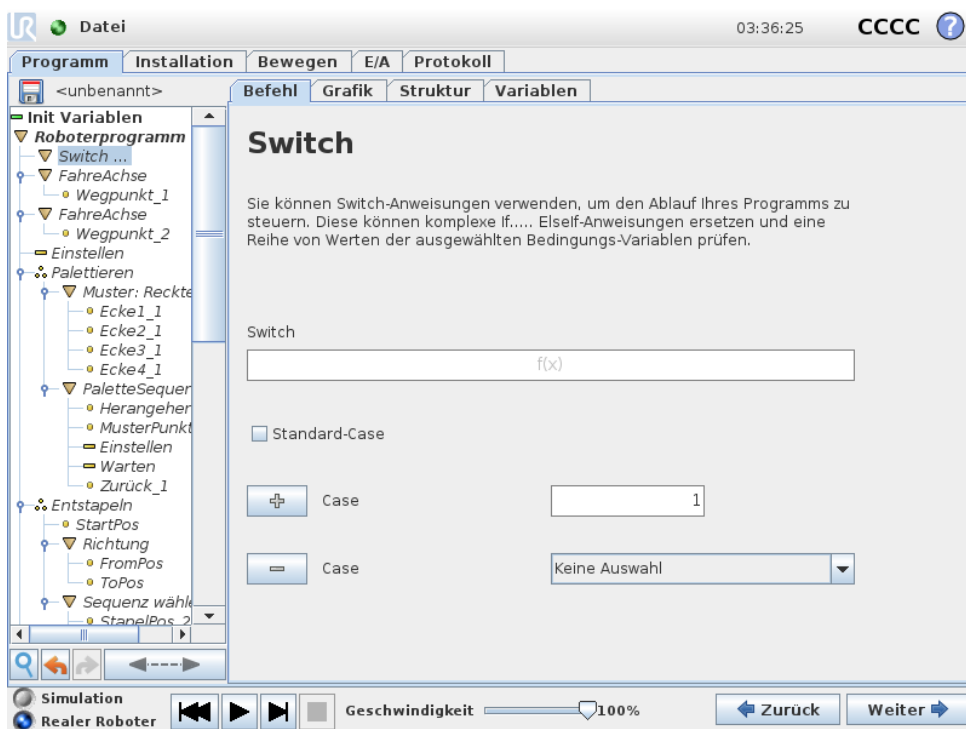
Ein Ereignis kann zur Überwachung eines Eingangssignals eingesetzt werden und eine Maßnahme durchführen oder eine Variable einstellen, wenn dieses Eingangssignal auf HIGH wechselt. Wechselt beispielsweise ein Ausgangssignal auf HIGH, kann das Ereignisprogramm 200 ms warten, bevor es das Signal anschließend wieder auf LOW zurücksetzt. Dadurch kann der Hauptprogrammcode erheblich vereinfacht werden, falls eine externe Maschine durch eine ansteigende Flanke anstelle eines HIGH-Pegels ausgelöst wird. Ein Ereignis wird einmal pro Kontrollzyklus (8 ms) überprüft.

### 14.21 Befehl: Thread



Ein Thread ist ein paralleler Prozess zum Roboterprogramm. Ein Thread kann zur Steuerung einer externen Maschine, unabhängig vom Roboterarm, eingesetzt werden. Ein Thread kann mithilfe von Variablen und Ausgangssignalen mit dem Roboterprogramm kommunizieren.

### 14.22 Befehl: Switch



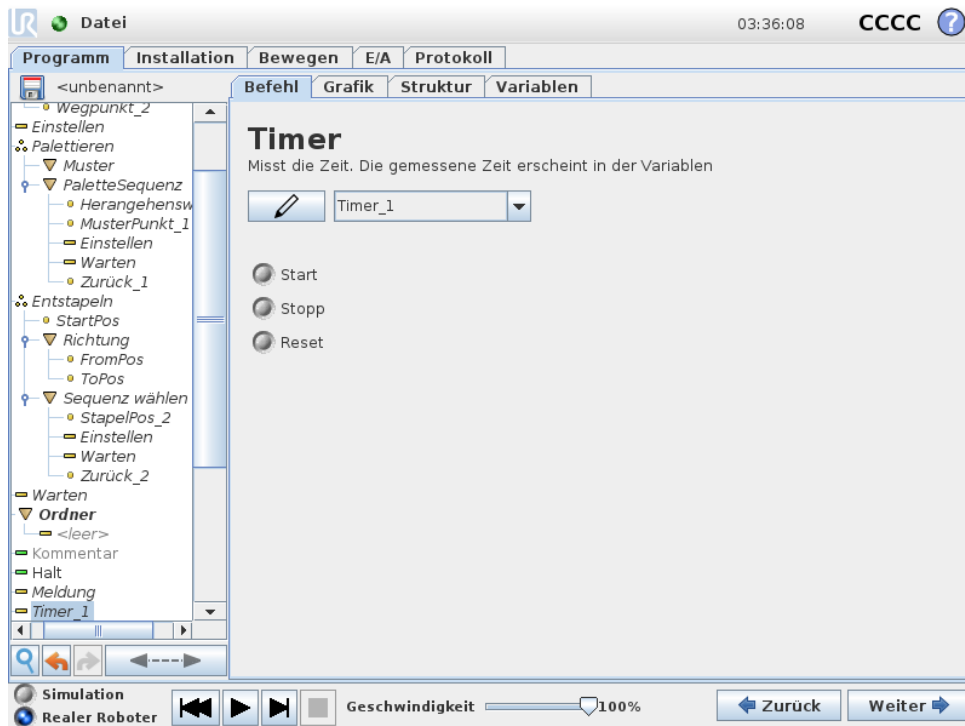
Copyright ©2009–2018 by Universal Robots A/S. Alle Rechte vorbehalten.

## 14.22 Befehl: Switch

Durch einen **Switch Case**-Befehl kann der Roboter sein Verhalten aufgrund von Sensoreingängen oder Variablewerten verändern. Verwenden Sie den **Ausdruckseditor**, um die Bedingung zu beschreiben, in welcher der Roboter mit den Unterbefehlen dieses **Switch** fortfahren soll. Wenn die Bedingung einen dieser Fälle erfüllt, werden die Zeilen in dem jeweiligen **Case** ausgeführt. Wurde ein **Default Case** festgelegt, werden die Linien nur dann ausgeführt, wenn keine anderen passenden Fälle wurden gefunden.

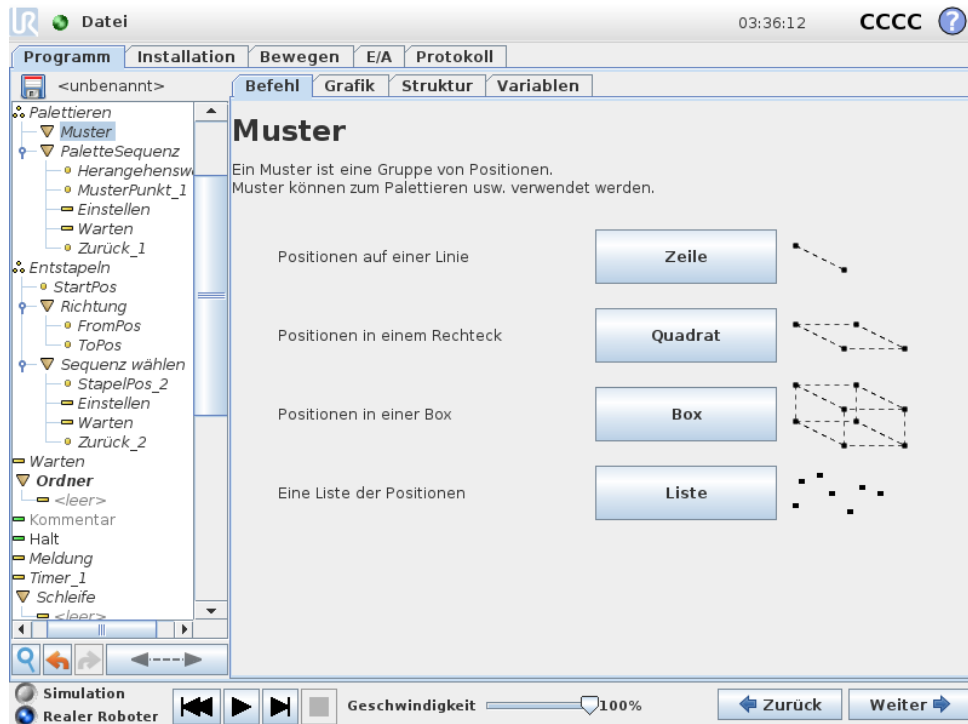
Jeder **Switch** kann mehrere **Cases** sowie einen **Default Case** haben. In einem **Switch** kann immer nur eine Instanz pro **Case-Wert** definiert sein. **Cases** können mithilfe der Schaltflächen auf dem Bildschirm hinzugefügt werden. Ein **Case-Befehl** kann für diesen **Switch** vom Bildschirm entfernt werden.

### 14.22.1 Timer



Ein **Timer** misst die benötigte Zeitdauer, die bestimmte Teile des Roboters für die Ausführung benötigen. Eine Programmvariable

## 14.23 Befehl: Muster



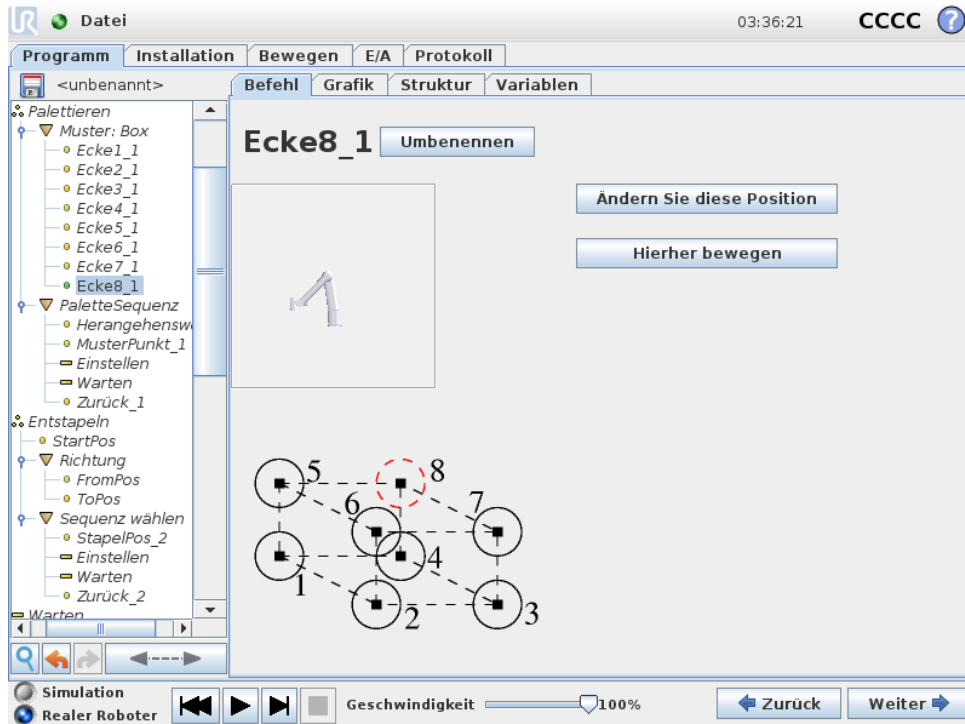
Der Befehl **Muster** kann eingesetzt werden, um die Positionen im Roboterprogramm zu durchlaufen. Der Befehl **Muster** entspricht bei jeder Ausführung einer Position.

Ein Muster kann aus Punkten in einer Linie, in einem Quadrat, in einer Box oder nur aus einer Liste aus Punkten bestehen. Die ersten drei, nämlich **Linie**, **Quadrat** oder **Box** können für Positionen in einem regelmäßigen Muster verwendet werden. Die regelmäßigen Muster werden von einer Anzahl an charakteristischen Punkten definiert, wobei die Punkte die Kanten/Ecken des Musters definieren. Für **Linie** sind dies die beiden Endpunkte, für **Quadrat** sind es drei der vier Eckpunkte, während es für **Box** vier der acht Eckpunkte sind. Der Programmierer gibt die Anzahl der Positionen entlang jeder der Ecken im Muster ein. Die Robotersteuerung errechnet dann die einzelnen Musterpositionen, indem die Kantenvektoren proportional addiert werden.

Fallen die durchlaufenen Positionen nicht in ein regelmäßiges Muster, kann man **Liste** wählen und sich der von den Programmierern bereitgestellten Positionen bedienen. Auf diese Weise kann jede Art von Positionierung realisiert werden.

### Muster definieren

Bei Wahl des **Box**-Musters wechselt der Bildschirm zu dem unten gezeigten.



Ein **Boxmuster** verwendet drei Vektoren, um die Seiten der Box zu definieren. Diese drei Vektoren sind als vier Punkte gegeben, wobei der erste Vektor von Punkt ein bis Punkt zwei, der zweite von Punkt zwei bis Punkt drei und der dritte von Punkt drei bis Punkt vier geht. Jeder Vektor wird durch die Anzahl der Punkte in dem angegebenen Intervall dividiert. Jede Position im Muster wird durch das proportionale Addieren der Intervektoren berechnet.

Das **Linien-** und das **Quadrat-**Muster funktioniert ähnlich.

Eine Zähler-Variable wird beim Durchgehen der Positionen im Muster verwendet. Der Name der Variablen wird auf dem Befehlsbildschirm für **Muster** angezeigt. Die Variable durchläuft die Zahlen von 0 bis  $x * y * z - 1$ , die Anzahl der Punkte in den Mustern. Diese Variable kann mit Zuweisungen manipuliert und in Ausdrücken verwendet werden.

## 14.24 Befehl: Kraft

Im Wirkungsbereich des Roboters erlaubt der **Kraftmodus** eine Anpassung und Kraftanwendung in konformen Achsen. Alle Roboterarmbewegungen im Rahmen eines **Kraft-**Befehls erfolgen im **Kraftmodus**. Bei Bewegung des Roboterarms im **Kraftmodus** können eine oder mehrere Achsen ausgewählt werden, in denen sich der Roboterarm konform verhält. Der Roboterarm geht mit der Umgebung entlang einer vorgegebenen Achse konform. d. h. der Roboterarm passt seine Position automatisch an, um die vorgegebene Kraft zu erreichen. Der Roboterarm kann auch selbst auf seine Umgebung (z. B. ein Werkstück) Kraft ausüben.

Der **Kraftmodus** eignet sich für Anwendungen, bei denen die eigentliche TCP-Position entlang einer vorgegebenen Achse keine ausschlaggebende Bedeutung hat, sondern eher eine bestimmte Kraft entlang dieser Achse benötigt wird. Dies ist beispielsweise der Fall, wenn der Roboter-TCP auf eine gekrümmte Oberfläche trifft oder beim Schieben oder Ziehen eines Werkstücks. Der **Kraftmodus** lässt sich auch auf bestimmte Drehmomente um vorgegebene Achsen anwenden.

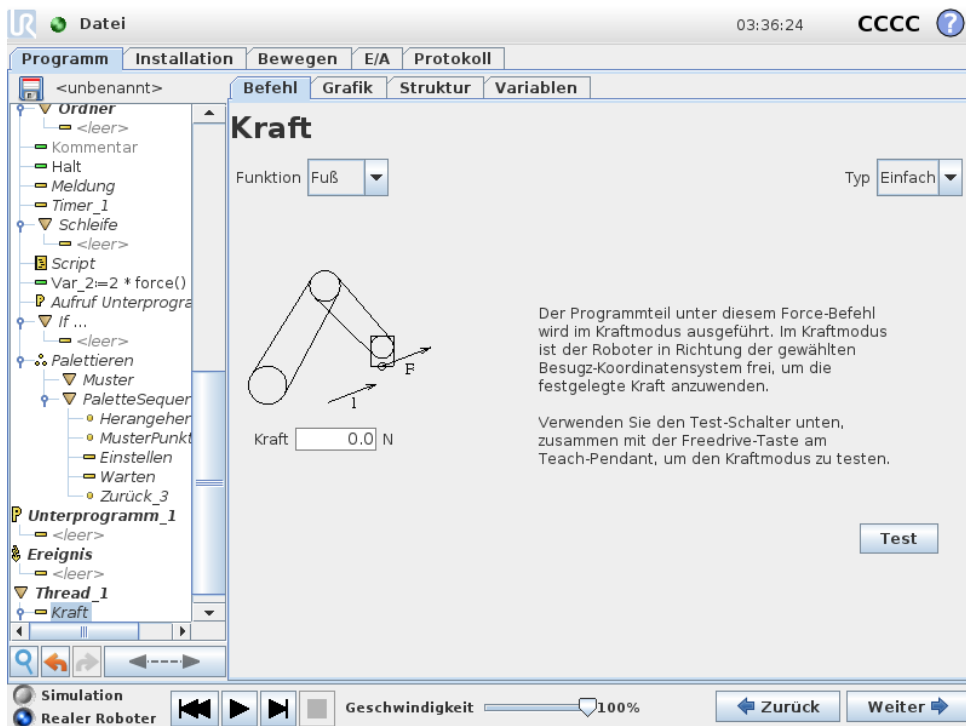
Hinweis: Trifft der Roboterarm auf einer Achse mit Kräfteinstellung ungleich null auf keinerlei

Hindernis, so tendiert er entlang/ an dieser Achse zur Beschleunigung. Auch wenn eine Achse als konform ausgewählt wurde, versucht das Roboterprogramm den Roboter entlang dieser Achse zu bewegen. Mithilfe der Kraftregelung ist jedoch sichergestellt, dass der Roboterarm die vorgegebene Kraft dennoch erreicht.



**WARNUNG:**

1. Vermeiden Sie zu starke Verlangsamung kurz vor Eintritt in den Kraftmodus.
2. Vermeiden Sie zu starke Beschleunigung im Kraftmodus, da dies zu Genauigkeitsverlusten bei der Kraftregelung führt.
3. Vermeiden Sie parallele Bewegungen zu konformen Achsen vor Eintritt in den Kraftmodus.



Copyright ©2009–2018 by Universal Robots A/S. Alle Rechte vorbehalten.

**Auswahl von Funktionen**

Im **Funktionsmenü** wird das vom Roboter während des Betriebs im Kraftmodus zu verwendende Koordinatensystem (Achsen) ausgewählt. Die im Menü enthaltenen Funktionen sind diejenigen, die bei der Installation festgelegt wurden, (siehe 13.12).

**Kraftmodustyp**

Es gibt vier verschiedene Kraftmodustypen, die bestimmen, wie die ausgewählte Funktion jeweils zu interpretieren ist.

- **Einfach:** In diesem Kraftmodus ist nur eine Achse konform. Die Kraftanwendung entlang dieser Achse ist anpassbar. Die gewünschte Kraft wird immer entlang der z-Achse der

ausgewählten Funktion angewendet. Bei Linienfunktionen geschieht dies entlang der y-Achse.

- **Rahmen:** Der Rahmen-Kraftmodus ermöglicht eine erweiterte Anwendung. Die Positionsanpassung und die Kräfte in allen sechs Freiheitsgraden können hier unabhängig voneinander eingestellt werden.
- **Punkt:** Bei Auswahl des Punkt-Kraftmodus verläuft die y-Achse des Task-Rahmens vom Roboter-TCP zum Ursprung der ausgewählten Funktion. Der Abstand zwischen dem Roboter-TCP und dem Ursprung der ausgewählten Funktion muss mindestens 10 mm betragen. Bitte beachten Sie, dass sich der Task-Rahmen während der Ausführung mit der Position des Roboter-TCPs ändert. Die x- und z-Achse des Task-Rahmens sind von der ursprünglichen Ausrichtung der ausgewählten Funktion abhängig.
- **Bewegung:** Bewegung bedeutet, dass sich der Task-Rahmen mit der Richtung der TCP-Bewegung verändert. Die x-Achse des Task-Rahmens ist eine Projektion der TCP-Bewegungsrichtung auf der Ebene zwischen x- und y-Achse der ausgewählten Funktion. Die y-Achse ist orthogonal zur Bewegung des Roboterarms gerichtet und in der x-y-Ebene der ausgewählten Funktion. Dies kann beim Entgraten entlang eines komplexen Pfades hilfreich sein, bei dem eine zur TCP-Bewegung senkrechte Kraft benötigt wird.

Hinweis: Im Falle eines Stillstandes des Roboterarms: Wird in den Kraftmodus übergegangen, wenn der Roboterarm stillsteht, so gibt es keine konformen Achsen bis die TCP-Geschwindigkeit über null liegt. Wenn der Roboterarm später (immer noch im Kraftmodus) wieder stillsteht, hat der Task-Rahmen die gleiche Ausrichtung wie zu dem Zeitpunkt, als die TCP-Geschwindigkeit das letzte Mal über Null lag.

Für die letzten drei Kraftmodustypen wird der tatsächliche Task-Rahmen während der Ausführung im Tab (siehe 14.29) angezeigt, wenn der Roboter im Kraftmodus betrieben wird.

## Auswahl des Kraftwertes

- Der Kraft- oder Drehmomentwert kann für konforme Achsen eingestellt werden, so dass der Roboterarm seine Position anpasst, um die ausgewählte Kraft zu erreichen.
- Bei nichtkonformen Achsen folgt der Roboterarm der Bahn, die mit dem Programm festgelegt wurde.

Für Übersetzungsparameter wird die Kraft in Newton [N] angegeben, für Rotationsparameter wird das Drehmoment in Newtonmeter [Nm] angegeben.



### HINWEIS:

Folgende Schritte sind erforderlich:

- Verwenden Sie die Scriptfunktion `Festlegung_TCP-Kraft()` in einem separaten Thread, um Ist-Kraft und -Drehmoment auszulesen.
- Korrigieren Sie den Vektor für den Schlüssel, falls die tatsächliche Kraft und/oder Drehmoment niedriger sein sollte als benötigt.

## Grenzwertauswahl

Für alle Achsen können Grenzwerte eingegeben werden, die allerdings, je nach Konformität der Achse, verschiedene Bedeutung haben.

- **Konform:** Der Grenzwert gibt die maximal zulässige Geschwindigkeit des TCP entlang/an der Achse an. Die Einheiten sind [mm/s] und [deg/s].
- **Nicht konform:** Der Grenzwert gibt die maximal zulässige Abweichung von der vom Programm vorgegebenen Bahn an, über welcher ein Sicherheitsstopp des Roboters ausgelöst wird. Die Einheiten sind [mm] und [deg].

---

## Testeinstellungen für Kraft

Über den als **Test** gekennzeichneten Ein-/Aus-Schalter wird die **Freedrive**-Taste hinten am Teach Pendant vom normalen Freedrive-Modus auf das Testen der Kraft umgeschaltet.

Wird bei eingeschaltetem **Testschalter** die **Freedrive**-Taste hinten am Teach Pendant gedrückt, führt der Roboter den Kraftbefehl ohne Durchlauf des Programms direkt aus, sodass die Einstellungen vor der eigentlichen Ausführung des Programms geprüft werden können. Diese Funktion ist besonders nützlich, um sicherzustellen, dass konforme Achsen und Kräfte korrekt ausgewählt und eingestellt wurden. Halten Sie den Roboter-TCP einfach mit einer Hand, drücken Sie mit der anderen Hand die **Freedrive**-Taste und beobachten Sie, in welche Richtungen der Roboterarm bewegt oder nicht bewegt werden kann. Nach Verlassen dieses Bildschirms wird der Test-Schalter automatisch abgeschaltet, so dass die **Freedrive**-Taste hinten am Teach Pendant wieder für den freien **Freedrive**-Modus genutzt werden kann.

Hinweis: Die **Freedrive**-Taste ist nur wirksam, wenn eine gültige Funktion für den Kraft-Befehl ausgewählt wurde.

## 14.25 Befehl: Palettieren



Ein Palettierbetrieb kann eine Reihe von Bewegungen an bestimmten Stellen beinhalten, die als Muster vorgegeben sind (siehe 14.23). An jeder Stelle im Muster wird die Abfolge von Bewegungen in Relation zur Position im Muster durchgeführt.

### Programmierung eines Palettierbetriebs

Die durchzuführenden Schritte lauten wie folgt:

1. Festlegung eines Musters.
2. Führen Sie eine **Palettenabfolge** für die Aufnahme/das Ablegen an jeder einzelnen Stelle durch. Die Abfolge beschreibt, was an jeder Position im Muster durchzuführen ist.
3. Verwenden Sie das Auswahlwerkzeug im Menü Abfolgebefehl, um festzulegen, welcher der Wegpunkte in der Abfolge welcher Position im Muster entsprechen soll.

### Palettenabfolge/Verankerbare Abfolge

In einer **Palettenabfolgelinie** sind die Bewegungen des Roboterarms auf die Position der Palette bezogen. Das Verhalten einer Abfolge ist so, dass sich der Roboterarm an der durch das Muster vorgegebenen Position, in der **Verankerungs-/Musterposition** befinden wird. Damit diese übereinstimmen, werden alle verbleibenden Positionen verschoben.

Verwenden Sie bitte NICHT den Befehl **Move** innerhalb einer Abfolge, da dieser nicht relativ zur Verankerungsposition erfolgen wird.

### „VorStart“

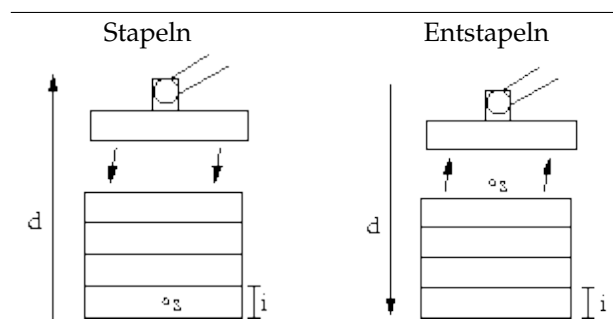
Die optionale **VorStart**-Abfolge wird kurz vor Beginn des Stapelvorgangs ausgeführt. Dies kann genutzt werden, um auf Freigabesignale zu warten.

„NachEnde“

Die optionale **NachEnde**-Abfolge wird kurz nach Ende des Stapelvorgangs ausgeführt. Diese kann dafür eingesetzt werden, um zu signalisieren, dass die Bewegung des Conveyers in Vorbereitung auf die nächste Palette beginnen kann.

14.26 Befehl: Suchen

Die Suchfunktion verwendet einen Sensor, um zu bestimmen, wann die korrekte Position erreicht ist, um ein Element zu fassen oder loszulassen. Der Sensor kann ein Drucktastenschalter, ein Drucksensor oder ein kapazitiver Sensor sein. Diese Funktion ist für Arbeiten an Stapeln aus Artikeln mit unterschiedlicher Dicke konzipiert, oder wenn die genauen Positionen der Artikel nicht bekannt oder schwierig zu programmieren sind.



Bei der Programmierung eines Stapelvorgangs ist der Ausgangspunkt  $s$ , die Stapelrichtung  $d$  und die Dicke der Elemente auf dem Stapel  $i$  zu definieren.

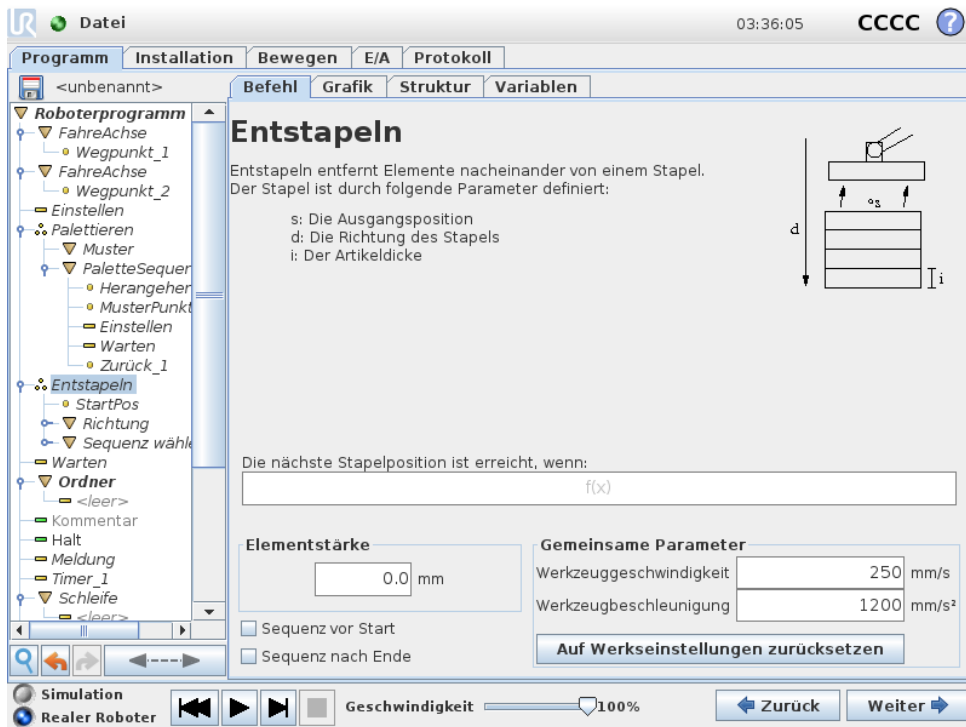
Dazu ist die Voraussetzung für die nächste Stapelposition sowie eine spezielle Programmabfolge, die an jeder Stapelposition ausgeführt wird, zu definieren. Auch Geschwindigkeit und Beschleunigungen müssen für die Bewegung im Stapel bestimmt werden.

## Stapeln



Beim Stapeln bewegt sich der Roboterarm in die Ausgangsposition und dann in die *Gegenrichtung*, um die nächste Stapelposition zu suchen. Wenn gefunden, merkt sich der Roboter die Position und führt die spezielle Abfolge aus. Das nächste Mal startet der Roboter die Suche aus dieser Position, erweitert um die Stärke des Elements in der Stapelrichtung. Das Stapeln ist beendet, wenn die Stapelhöhe eine bestimmte Anzahl erreicht hat oder der Sensor ein Signal gibt.

## Entstapeln

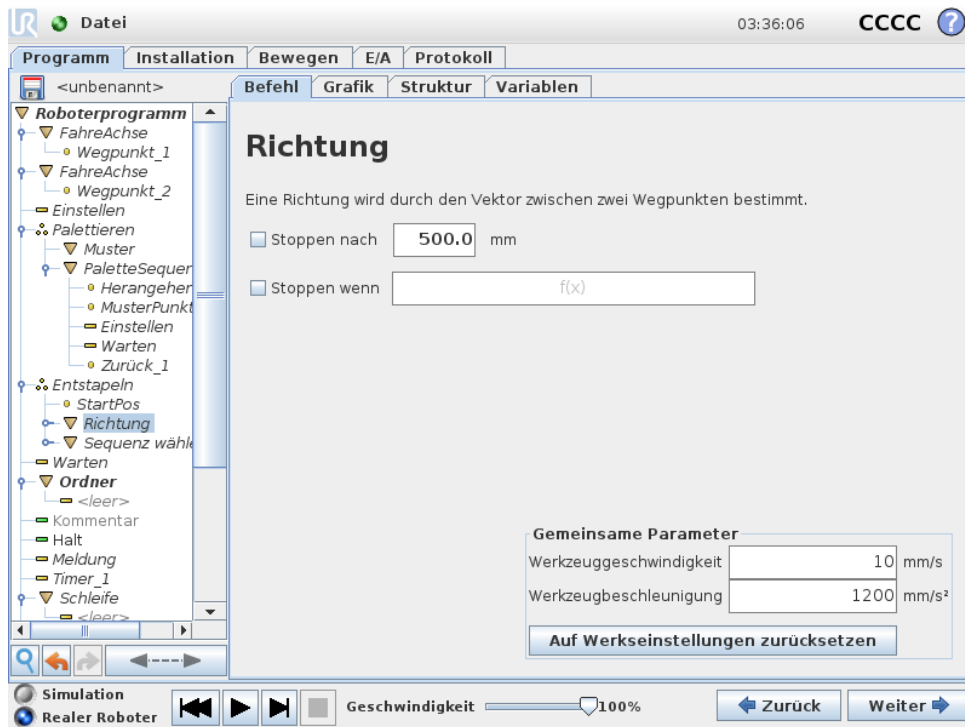


Beim Entstapeln bewegt sich der Roboterarm von der Ausgangsposition in die angegebene Richtung, um nach dem nächsten Element zu suchen. Die Voraussetzung auf dem Bildschirm bestimmt, wann das nächste Element erreicht wird. Wenn die Voraussetzung erfüllt wird, merkt sich der Roboter die Position und führt die spezielle Abfolge aus. Das nächste Mal startet der Roboter die Suche aus dieser Position, erweitert um die Stärke des Elements in der Stapelrichtung.

### Ausgangsposition

Das Stapeln beginnt mit der Ausgangsposition. Wird die Ausgangsposition weggelassen, fängt das Stapeln an der aktuellen Position des Roboterarms an.

## Richtung



Die Richtung wird durch zwei Punkte angezeigt und wird als Differenz der TCP-Punkt 1 und 2 ermittelt.

Hinweis: Eine Richtung berücksichtigt nicht die Ausrichtung der Punkte.

### Ausdruck der nächsten Stapel-Position

Der Roboterarm bewegt sich entlang des Richtungsvektors, während er fortlaufend bewertet, ob die nächste Stapel-Position erreicht worden ist. Wenn der Ausdruck als `Wahr` bewertet wird, wird die spezielle Abfolge ausgeführt.

### „VorStart“

Die optionale `VorStart`-Abfolge wird kurz vor Anfang des Stapelvorgangs ausgeführt. Dies kann genutzt werden, um auf Freigabesignale zu warten.

### „NachEnde“

Die optionale `NachEnde`-Abfolge wird kurz nach Ende des Stapelvorgangs ausgeführt. Dies kann genutzt werden, um dem Conveyer ein Signal zur Vorbereitung auf den nächsten Stapel zu geben.

### Einlege/Entnahme-Sequenz

Wie beim Palettierbetrieb (14.25) wird an jeder Stapelposition eine spezielle Programmabfolge ausgeführt.

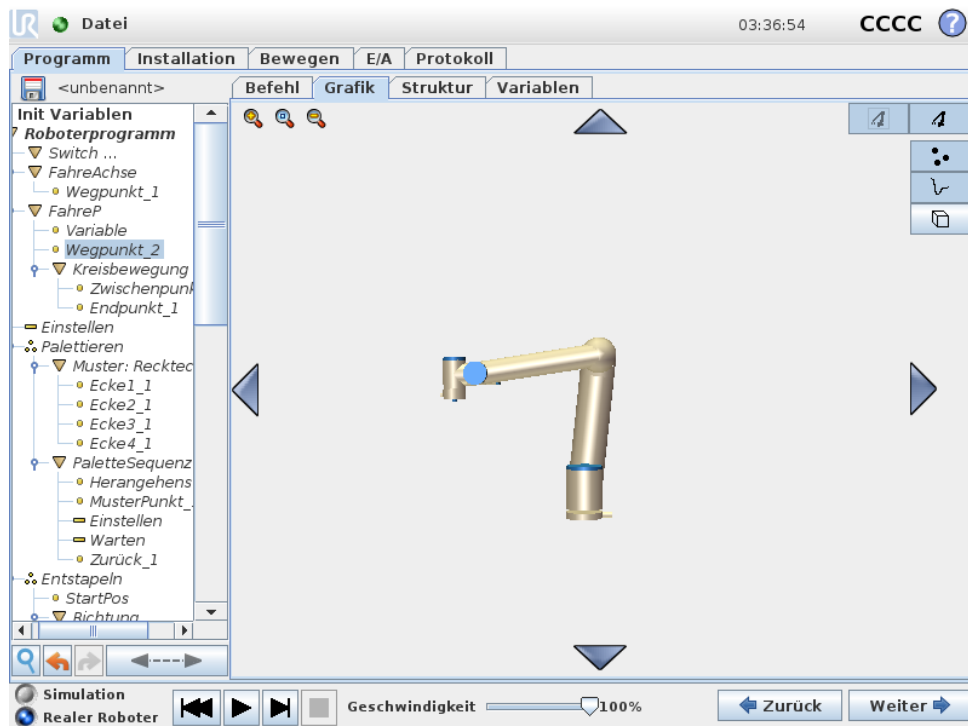
## 14.27 Befehl: Fließband-Tracking

Wird ein Fließband verwendet, kann der Roboter so konfiguriert werden, dass er dessen Bewegung verfolgt. Ist das **Fließband-Tracking** in der Installation korrekt konfiguriert, folgt der Roboter mit seinen Bewegungsabläufen dem Band. Der Programmknoten **Fließband-Tracking** steht im Tab **Assistenten** unter dem Tab **Struktur** zur Verfügung. Alle Bewegungen unter diesem Knoten sind beim Fließband-Tracking erlaubt, stehen aber im Verhältnis zu der Fließbandbewegung. Die **Fließband-Tracking**-Einstellung im Tab **Installation** (siehe Abschnitt 13.13) enthält Optionen, mit denen der Roboter für den Einsatz absoluter und relativer Encoder sowie linearer oder kreisförmiger Fließbänder eingestellt werden kann.

## 14.28 Befehl: Unterdrücken

Unterdrückte Programmzeilen werden bei der Programmausführung übersprungen. Die Unterdrückung einer Zeile kann zu einem späteren Zeitpunkt wieder aufgehoben werden. Dies ist eine einfache Methode, um Änderungen an einem Programm vorzunehmen, ohne die ursprünglichen Inhalte zu zerstören.

## 14.29 Grafik-Tab



Grafische Darstellung des aktuellen Roboterprogramms. Der Weg des TCP wird in einer 3D-Ansicht gezeigt, mit schwarzen Bewegungssegmenten und grünen Übergangsegmenten (Übergänge zwischen den Bewegungssegmenten). Die grünen Punkte bestimmen die Positionen des TCP an jedem der Wegpunkte im Programm. Die 3D-Zeichnung des Roboterarms zeigt die aktuelle Position des Roboterarms, während der *Schatten* des Roboterarms verdeutlicht, wie der Roboterarm beabsichtigt, die auf der linken Bildschirmseite gewählten Wegpunkte zu erreichen.

Nähert sich die aktuelle Position des Roboter-TCP einer Sicherheits- oder Auslöseebene oder befindet sich die Ausrichtung des Roboterwerkzeugs nahe einer Werkzeugausrichtungsgrenze (siehe 10.12), so wird eine 3D-Darstellung der angenäherten Bewegungsgrenze angezeigt.

Hinweis: Beachten Sie, dass die Visualisierung der Begrenzungen deaktiviert wird, während der Roboter ein Programm ausführt.

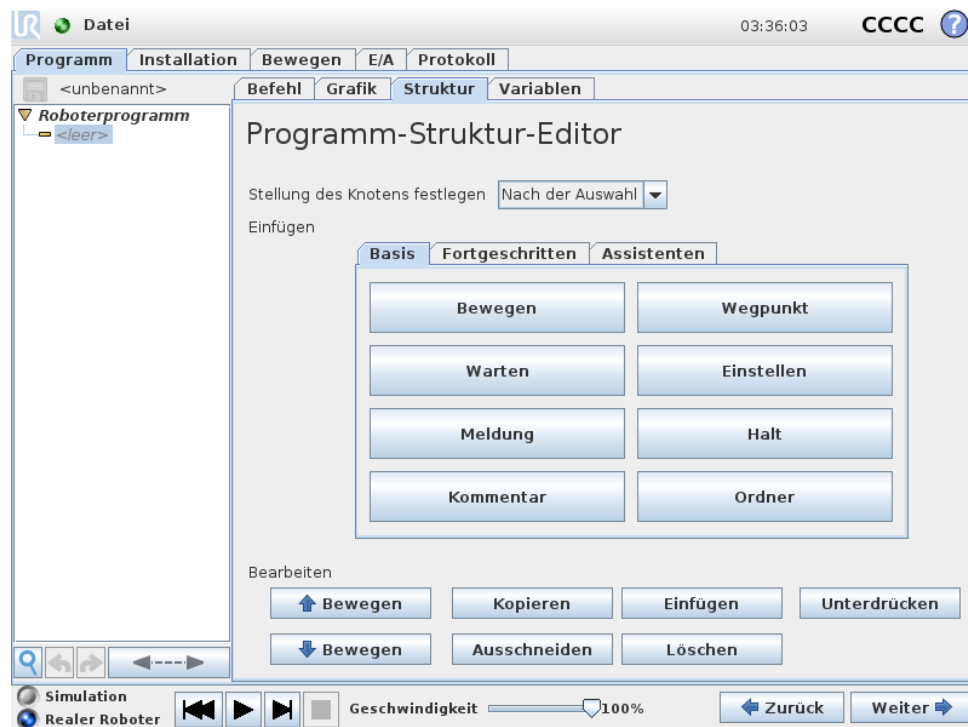
Sicherheitsebenen werden in Gelb und Schwarz zusammen mit einem kleinen Pfeil angezeigt, der für die Normal-Ebene steht, was angibt, auf welcher Seite der Ebene der Roboter-TCP positioniert werden darf. Auslöseebenen werden in Blau und Grün zusammen mit einem kleinen Pfeil dargestellt, der auf die Seite der Ebene zeigt, bei der die Grenzen des **Normalmodus** aktiv sind (siehe 10.6). Das Limit der Werkzeugausrichtungsgrenze wird anhand eines sphärischen Kegels visualisiert, wobei ein Vektor die aktuelle Ausrichtung des Roboterwerkzeugs anzeigt. Das Innere des Kegels repräsentiert den zulässigen Bereich für die Werkzeugausrichtung (Vektor).

Wenn der Zielroboter-TCP sich nicht mehr in Nähe zum Limit befindet, verschwindet die 3D-Darstellung. Wenn der TCP einen Grenzwert überschreitet oder dem sehr nahe ist, ändert sich die Limitanzeige zu rot.

Die 3D-Ansicht kann vergrößert und gedreht werden, um den Roboterarm besser sehen zu können. Die Schaltflächen oben rechts im Bildschirm können die verschiedenen grafischen Komponenten in der 3D-Ansicht deaktivieren. Die Schaltfläche unten schaltet die Visualisierung von Limits von Näherungsgrenzen ein/aus.

Die gezeigten Bewegungssegmente hängen vom gewählten Programmknoten ab. Wenn ein **Move**-Knoten ausgewählt wird, ist der angezeigte Pfad die mit diesem Move definierte Bewegung. Wird ein **Wegpunkt**-Knoten gewählt, zeigt das Display die folgenden  $\sim 10$  Bewegungsschritte:

## 14.30 Struktur-Tab



Im Tab „Struktur“ kann man die verschiedenen Befehlsarten einfügen, verschieben, kopieren und/oder entfernen.

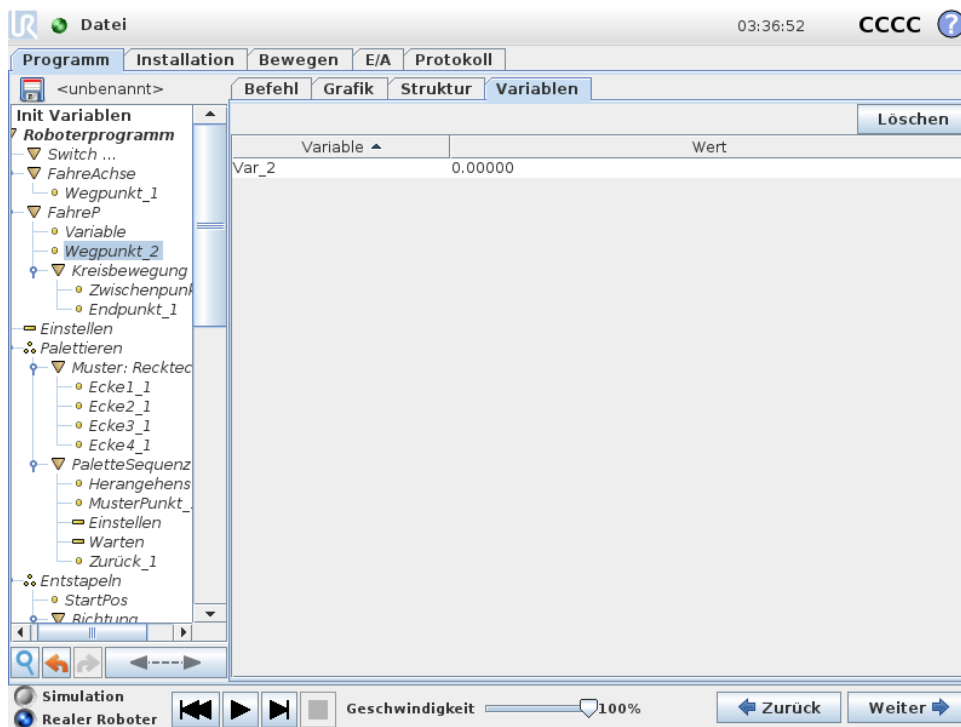
Um neue Befehle einzufügen, gehen Sie wie folgt vor:

1. Wählen Sie einen vorhandenen Programmbefehl.
2. Wählen Sie, ob der neue Befehl über oder unter dem gewählten Befehl eingefügt werden soll.
3. Drücken Sie die Schaltfläche für die Befehlsart, die Sie einfügen möchten. Öffnen Sie zur Einstellung der Details des neuen Befehls den Tab `Befehl`.

Befehle können mit Hilfe der Schaltflächen im Bearbeitungsrahmen verschoben/kopiert/gelöscht werden. Wenn ein Befehl über Unterbefehle verfügt (ein Dreieck neben dem Befehl) werden alle Unterbefehle ebenfalls verschoben/kopiert/gelöscht.

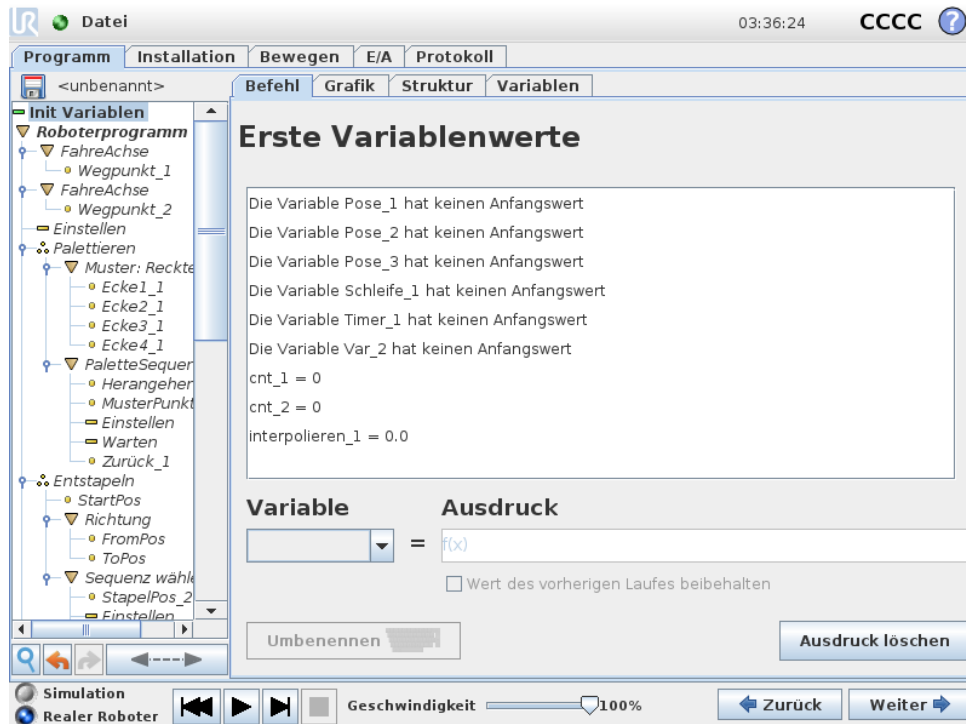
Nicht alle Befehle passen an alle Stellen in einem Programm. `Wegpunkte` müssen unter einem `Move`-Befehl stehen (nicht notwendigerweise direkt darunter). `ElseIf`- und `Else`-Befehle müssen nach dem Befehl `If` folgen. Im Allgemeinen kann die Verschiebung von `ElseIf`-Befehlen zu Verwirrungen führen. Variablen müssen Werte zugeordnet werden, bevor diese verwendet werden.

## 14.31 Der „Variablen“ -Tab



Der **Variablen** -Tab zeigt die aktuellen Werte von Variablen im laufenden Programm und führt eine Liste von Variablen und Werten zwischen Programmverläufen auf. Er erscheint, wenn er anzuzeigende Informationen enthält. Alle Variablen sind alphabetisch nach ihren Namen geordnet. Die Variablenbezeichnungen werden in diesem Bildschirm mit höchstens 50 Zeichen und die Variablenwerte mit höchstens 500 Zeichen angezeigt.

## 14.32 Befehl: Variablen-Initialisierung



Dieser Bildschirm ermöglicht die Einstellung von Variablen-Werten, bevor das Programm (mit einem Thread) ausgeführt wird.

Wählen Sie eine Variable aus der Liste der Variablen, indem Sie darauf klicken oder indem Sie die Variablen-Auswahlbox verwenden. Für eine ausgewählte Variable kann ein Ausdruck eingegeben werden, mit dem der Variablen-Wert bei Programmstart festgelegt wird.

Bei Wahl des Kontrollkästchens **Wert von der vorherigen Ausführung bewahren** wird die Variable auf den Wert initialisiert, der aus der Registerkarte **Variablen** hervorgeht (siehe 14.31). So können Variable ihre Werte zwischen Programmausführungen beibehalten. Die Variable erhält ihren Wert von dem Ausdruck bei erstmaliger Ausführung des Programms oder wenn der Tab-Wert gelöscht wurde.

Eine Variable kann aus dem Programm gelöscht werden, indem ihr Namensfeld leer gelassen wird (nur Leerschritte).

Copyright ©2009–2018 by Universal Robots A/S. Alle Rechte vorbehalten.

## 15 Set-up-Bildschirm



- **Roboter initialisieren** Führt Sie zum Initialisierungsbildschirm, siehe 11.5.
- **Sprache und Einheiten** Konfigurieren Sie die Sprache und die Maßeinheiten der Benutzeroberfläche, siehe 15.1.
- **Roboter aktualisieren** Aktualisiert die Robotersoftware auf eine neuere Version, siehe 15.2.
- **Passwort festlegen** Bietet die Möglichkeit zur Sperrung des Programmierteils des Roboters für Personen ohne Passwort, siehe 15.3.
- **Bildschirm kalibrieren** Kalibriert die „Oberfläche“ des Touch-Screens, siehe 15.4.
- **Netzwerk einrichten** Öffnet eine Schnittstelle zur Einrichtung des Ethernet-Netzwerks für den Controller, siehe 15.5.
- **Uhrzeit einstellen** Stellt die Uhrzeit und das Datum für das System ein und konfiguriert die Anzeigeformate für die Uhr, siehe 15.6.
- **URCaps-Einstellung** Übersicht über die installierten URCaps sowie Optionen für die Installation und Deinstallation, siehe 15.7.
- **Zurück** Führt Sie zum Startbildschirm zurück.

## 15.1 Sprachen und Einheiten



Auf diesem Bildschirm können die in PolyScope verwendeten Sprachen, Einheiten und die Tastatursprache ausgewählt werden.

Die ausgewählte Sprache wird für den sichtbaren Text auf den verschiedenen Bildschirmen von PolyScope sowie in der eingebetteten Hilfe verwendet. Aktivieren Sie „Englische Programmierung“, damit die Befehle in den Roboterprogrammen in Englisch angezeigt werden. PolyScope muss neu gestartet werden, um Änderungen wirksam zu machen.

Die ausgewählte Tastatursprache wird dann in allen Pop-up-Tastaturen in PolyScope verwendet.

## 15.2 Roboter aktualisieren



Softwareaktualisierungen können über USB-Sticks installiert werden. Stecken Sie einen USB-Stick ein und klicken Sie auf **Suchen**, um dessen Inhalt anzuzeigen. Um eine Aktualisierung durchzuführen, wählen Sie eine Datei, klicken Sie auf **Aktualisieren** und folgen Sie den Anweisungen auf dem Bildschirm.



### WARNUNG:

Prüfen Sie nach einer Softwareaktualisierung stets Ihre Programme. Die Aktualisierung könnte Bahnen in Ihrem Programm verändert haben. Die aktualisierten Softwarespezifikationen können durch Drücken der Schaltfläche „?“ rechts oben in der GUI aufgerufen werden. Hardwarespezifikationen bleiben unverändert und können dem Originalhandbuch entnommen werden.

## 15.3 Passwort festlegen

Zwei Passwörter sind verfügbar. Das erste ist ein *optionales* Systempasswort, das die Konfiguration des Roboters vor nicht autorisierten Änderungen schützt. Wenn ein Systempasswort eingerichtet ist, können Programme zwar ohne Passwort geladen und ausgeführt werden, aber zur Erstellung und Änderung von Programmen muss das Passwort eingegeben werden.

Das zweite ist ein *erforderliches* Sicherheitspasswort, das eingegeben werden muss, um die Sicherheitseinstellungen ändern zu können.



**HINWEIS:**

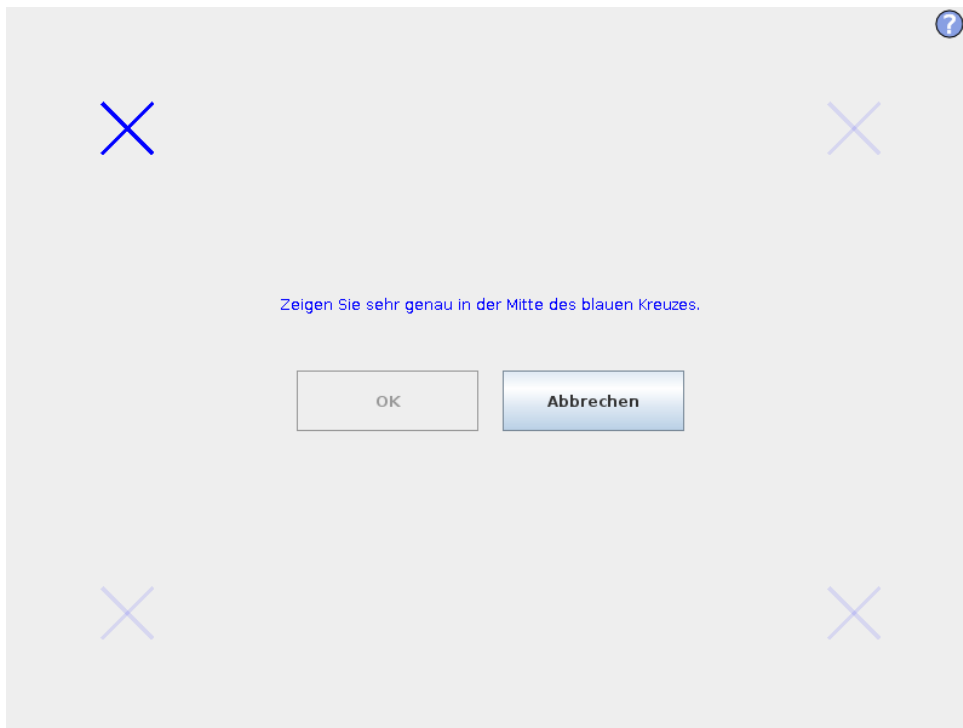
Um die Sicherheitskonfiguration ändern zu können, muss das Sicherheitspasswort festgelegt sein.



**WARNUNG:**

Fügen Sie ein Systempasswort hinzu, um zu verhindern, dass nicht autorisiertes Personal Änderungen an den Einstellungen des Roboters vornimmt.

## 15.4 Bildschirm kalibrieren



Kalibrieren des Touch-Screens. Befolgen Sie die Anleitung auf dem Bildschirm zur Kalibration des Touch-Screens. Verwenden Sie vorzugsweise einen spitzen, nicht metallischen Gegenstand, beispielsweise einen geschlossenen Stift. Durch Geduld und Sorgfalt lässt sich ein besseres Ergebnis erzielen.

## 15.5 Netzwerk einstellen

Fenster zur Einrichtung des Ethernet-Netzwerkes. Für die grundlegenden Roboterfunktionen ist keine Ethernet-Verbindung erforderlich, sodass diese standardmäßig deaktiviert ist.

## 15.6 Uhrzeit einstellen

Stellen Sie die Uhrzeit und das Datum für das System ein und konfigurieren Sie die Anzeigeformate für die Uhr. Die Uhr wird im oberen Bereich der Bildschirme *Programm ausführen* und *Roboter programmieren* angezeigt. Wenn Sie die Uhr antippen, wird das Datum kurz eingeblendet. Die GUI muss neu gestartet werden, um Änderungen wirksam zu machen.

## 15.7 URcaps-Einstellung






In der oberen Liste finden Sie eine Übersicht über alle installierten *URcaps*. Ein Klick auf ein URcap zeigt dessen Meta-Informationen (einschließlich des URcap-Namens, Version, Lizenz usw.) im Bereich URcap-Informationen unterhalb der Liste.

Klicken Sie auf die Schaltfläche + am unteren Bildschirmrand, um ein neues URcap zu installieren. Eine Dateiauswahl wird da angezeigt, wo `.urcap`-Dateien ausgewählt werden können. Klicken Sie auf *Öffnen* und PolyScope kehrt zum Einrichtungsbildschirm zurück. Das ausgewählte URcap wird installiert und ein entsprechender Eintrag erscheint kurz danach in der Liste. Neu installierte oder deinstallierte URcaps erfordern einen Neustart von PolyScope. Dazu wird die Schaltfläche *Restart* aktiviert.

Um ein URcap zu deinstallieren, wählen Sie es einfach in der Liste aus und klicken Sie auf die Schaltfläche -. Auch wenn das URcap jetzt bereits aus der Liste entfernt ist, muss dennoch anschließend ein Neustart durchgeführt werden.

Das Symbol neben einem Eintrag in der Liste zeigt den Zustand des URcap. Die verschiedenen Zustände sind nachstehend beschrieben:

-  *URcap o.k.*: Das URcap ist installiert und läuft ordnungsgemäß.
-  *URcap-Fehler*: Das URcap ist installiert, aber kann nicht ausgeführt werden. Kontaktieren Sie den Entwickler des URcaps.
-  *URcap-Neustart erforderlich*: Das URcap wurde gerade installiert und ein Neustart ist erforderlich.