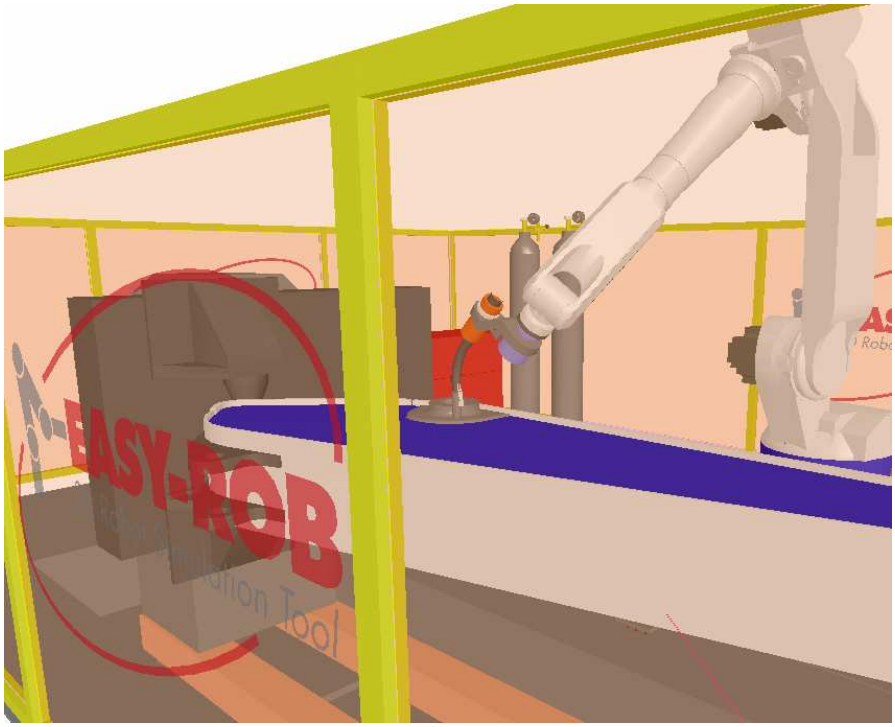
Die neue Version

EASY-ROB™ V5.3



Januar 2010

Version 1.04

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™

Inhaltsverzeichnis

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™ V5.3

Die neue Version 5.3

Wir freuen uns, Ihnen die neue EASY-ROB™ Version 5.3 mit den Neuerungen und Verbesserungen vorzustellen. Folgendes Dokument vermittelt Ihnen einen ersten Überblick über die wichtigsten Änderungen. Eine detaillierte Beschreibung finden Sie wie immer in den Bedienhinweisen.

- **CAD Daten-Import**
  Unterstützung des neutralen Formats 3D-Studio 3DS File Format
- **Transparente Geometrien**
  Geometrien lassen sich transparent darstellen
- **Neuer Kollisions-Algorithmus**
  Ein neu implementierter Kollisionsalgorithmus erlaubt es Toleranzen zu definieren
- **Robotertausch**
  Schnelles Austauschen von Robotern und Geräten vereinfacht die Layoutplanung
- **SNAP-Funktion**
  Einfaches platzieren von Robotern, Geräten, Geometrien und Tags auf Flächen oder an Punkte
- **Vereinfachte TOOL Definition**
  Die Festlegung bzw. Übernahme der TCP-Daten wurde erheblich vereinfacht
- **TAG Attribute**
  Zusätzliche TAG Attribute für übersichtliche ERPL-Programme
- **Bewegungsplanung mit Roboter TURNS**
  Die Bewegungsplanung berücksichtigt bei PTP-Bewegungen TURN-Vorgaben
- **Messfunktionen**
  Vermessen von Kreisen und Abstände zwischen Geräten, Kreismittelpunkten und Flächen
- **Schriftgröße**
  Einstellmöglichkeiten der Schriftgröße für Namen und Tags für verbesserte Lesbarkeit
- **Erweitertes API**
  Die exportierte Klasse ER_CAPI strukturiert API-Funktionen und vereinfacht deren Anwendung
- **Windows® 7**
  EASY-ROB™ kompatibel auf 32- und 64-Bit

Die neue Version steht für alle Kunden mit einer gültigen Lizenz für EASY-ROB™ V5.3 kostenfrei zur Verfügung. Für Kunden älterer Versionen besteht die Möglichkeit ein Update zu erwerben. Für Ihre Anregungen und Verbesserungsvorschläge bedanken wir uns schon jetzt bei Ihnen.

Vielen Dank

Stefan Anton

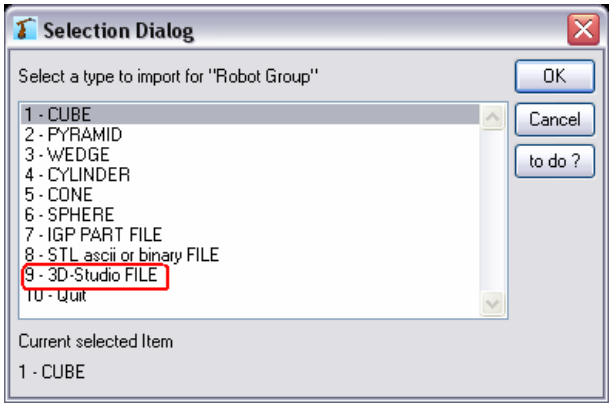
EASY-ROB
3D Robot Simulation Tool

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

3D-Studio CAD Daten-Import Format

EASY-ROB™ unterstützt neben den CAD-Formaten IGP Part File, STL (ASCII und binär) nun auch das neutrale 3D - Studio 3DS File Format. Dieses Format wird direkt gelesen, muss also nicht vorab konvertiert werden. Das 3DS Format wird von vielen CAD Systemen als Exportmöglichkeit angeboten und zeichnet sich durch seine Kompaktheit aus, wodurch es schnell gelesen werden kann.



3D CAD Menu > Create/Import new 3D CAD Body

Für weitere neutrale Formate wie STEP, IGES, VDA, VRML II und JT-Open stehen die Möglichkeiten im CAD-Import Window zur Verfügung. Die aktuelle Version des verwendeten CAD Daten-Import Kernels (3D_Evolution® API) ist noch leistungsfähiger. Insbesondere bei der Reduzierung der Dreiecke.

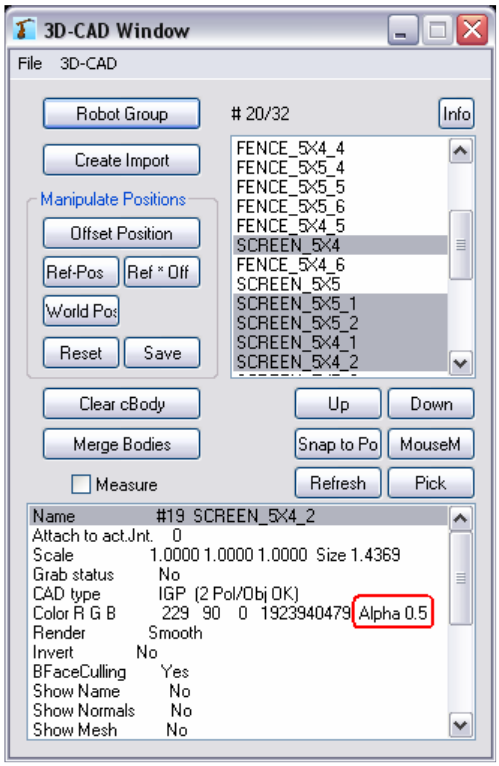
Der CAD Daten-Import Kernel erlaubt es auch native Formate wie CATIA V4 und CATIA V5 R19, Pro/E, UG, SolidWorks, Robface und weitere Formate einzulesen.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

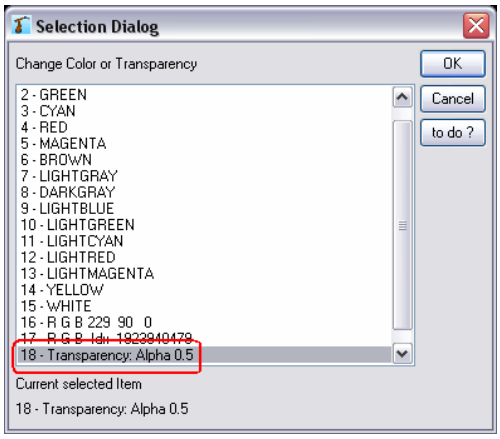
Transparente Geometrien

Sämtliche Geometrien lassen sich durch Angabe eines Alpha Blend Wertes transparent darstellen.

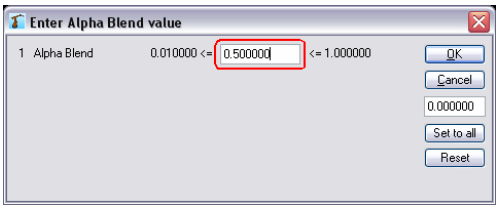
Öffnen Sie das 3D-CAD Window und selektieren Sie ein oder mehre Bodies. Doppelklicken Sie auf die Zeile „Color R G B“



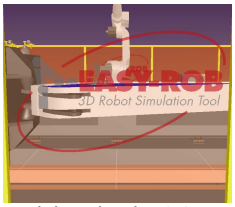
„Multi-Selection“

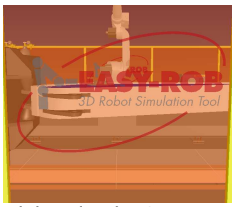


Selektieren Sie „18 - Transparency: Alpha 0.5“ und geben einen neuen Alpha Blend Wert im Bereich zwischen 0 bis 1 ein.



Alpha Blend Wert
= 0 unsichtbar
]0,1[transparente Darstellung
= 1 Transparenz abgeschaltet


Alpha Blend = 0.2


Alpha Blend = 0.5

ERCL - Befehl

```
ERC TRANSPARENCY
      BODY [ROBOT,TOOL]
      bodyname alpha
ERC TRANSPARENCY
      BODY_GRP [ROBOT_GRP,TOOL_GRP]
      alpha

mit alpha = [0,1]
```


EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

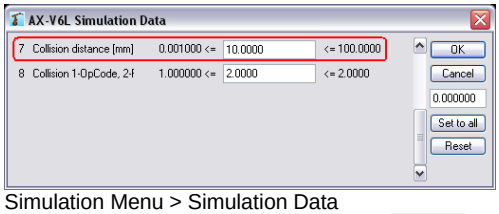
Neuer Kollisions-Algorithmus

Der neue Kollisions-Algorithmus erlaubt es Toleranzen bzw. einen minimalen Abstand zu definieren.

Während der Simulation wird eine Kollision angezeigt, wenn der Abstand zweier Geometrien einen minimalen Kollisionsabstand unterschreitet. Somit kann die Planungssicherheit gegenüber Bauteiltoleranzen und weiteren realen Abweichungen erheblich gesteigert werden.

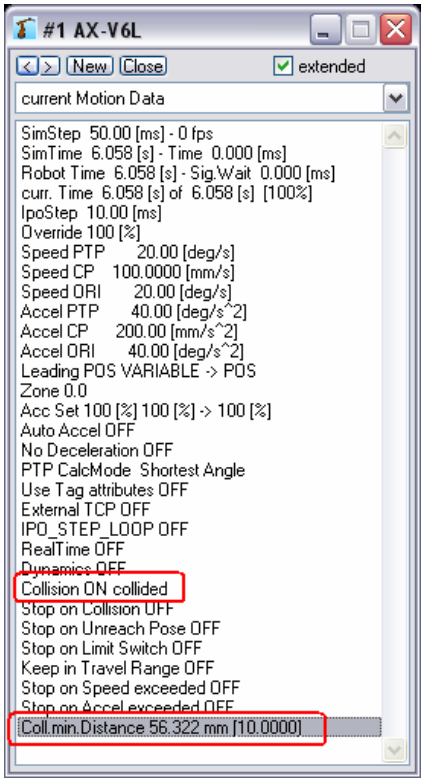
Schalten Sie die Kollision ein 

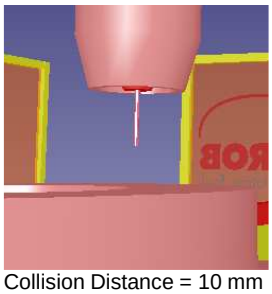
Geben Sie einen minimalen Kollisionsabstand ein, z.B. 10 mm



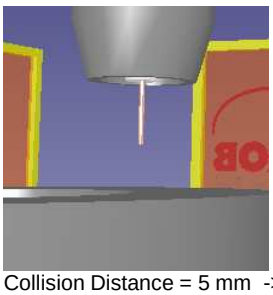
Simulation Menu > Simulation Data

Öffnen Sie Online Output Data 

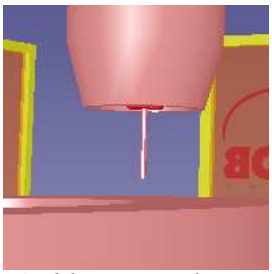




Collision Distance = 10 mm -> Kollision



Collision Distance = 5 mm -> keine Kollision



TCP fährt 5 mm nach unten
Collision Distance = 5 mm -> Kollision

ERCL - Befehle

```
ERC COLLISION ON/OFF
ERC COLLISION DISTANCE distance
mit distance in [mm]
```

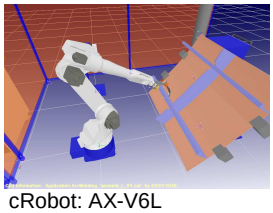

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

Tauschen von Robotern und Geräten

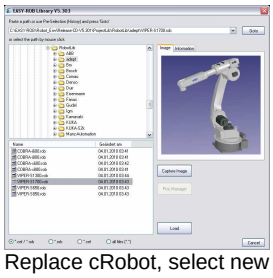
Das Austauschen von Robotern oder Geräten wurde erheblich vereinfacht. Roboter in der Arbeitszelle sind in der Regel mit ihrer Umgebung verknüpft, haben eine Lage bzgl. ihres Referenzkoordinatensystems, mehrere Tool/TCP-Definitionen und ein ERPL-Programm.

Beim Austausch von Robotern werden diese Eigenschaften übernommen. Somit können Kinematiken die beispielsweise eine zu geringe Reichweite habe schnell gewechselt werden, was die Layoutplanung erheblich vereinfacht.

File Menu > Load > Replace cRobot



cRobot: AX-V6L



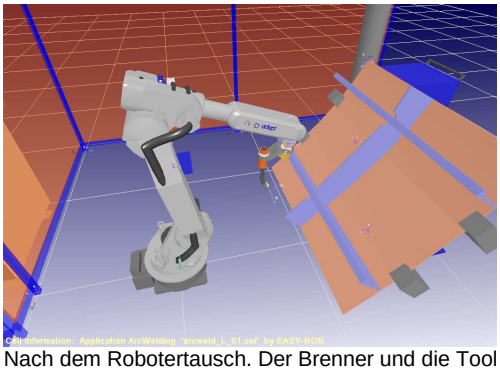
Replace cRobot, select new one

Ja, wenn im ERPL-Programm der Name des Roboters verwendet wird.
Nein, wenn auf auf den Namen des Roboters kein Bezug genommen wird.

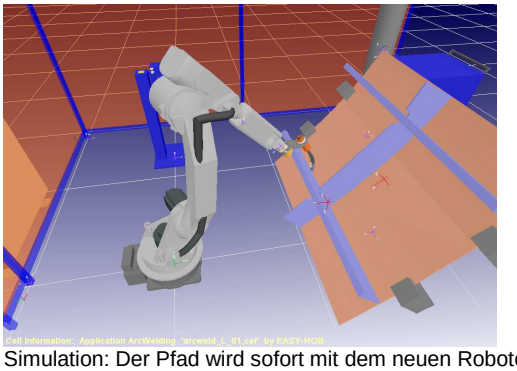
Im Beispiel wird im Programm der Befehl „ERC CURRENT_DEVICE SET AX-V6L“ verwendet, deshalb wählen wir „Ja“

Ja, wenn die Tool-Daten des alten Roboters übernommen werden sollen. Sinnvoll bei Robotertausch.
Nein, macht nur Sinn wenn es sich z.B. um ein Werkzeug, einen anderen Brenner, handelt.

Im Beispiel tauschen wir einen Roboter und wollen die Tool-Definitionen des alten Roboters übernehmen. Wir wählen „Ja“



Nach dem Robotertausch. Der Brenner und die Tool Definitionen wurden übernommen



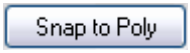
Simulation: Der Pfad wird sofort mit dem neuen Roboter abgefahren.

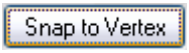
EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

SNAP-Funktion

Die neue SNAP-Funktion ermöglicht es Roboter, Geräte, Tools, Geometrien und Tags auf Flächen oder an Punkte zu platzieren.

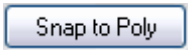
1. SNAP Devices/Geräte
   - Öffnen Sie das Kinematics Window (Robotics Menu > Open Kinematics Window oder Ctrl+K) und wählen Sie mit 

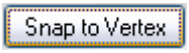
ein Gerät aus.
   - Wählen Sie 

oder 

aus und klicken sie anschließend eine Fläche oder einen Punkt/Vertex in der Arbeitszelle.
   - Die Basis des Gerätes „springt“ an die gewählte Fläche bzw. an den gewählten Punkt. Bei Snap to Vertex wird die aktuelle Orientierung beibehalten.

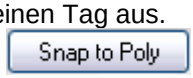
2. SNAP Bodies/Geometrien
   - Öffnen Sie das 3D CAD Window (3d CAD Menu > Open 3D CAD Window) und wählen Sie mit 

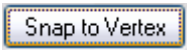
eine Geometrie aus.
   - Wählen Sie 

oder 

aus und klicken sie anschließend eine Fläche oder einen Punkt/Vertex in der Arbeitszelle.
   - Das Bezugskoordinatensystem der Geometrie „springt“ an die gewählte Fläche bzw. an den gewählten Punkt. Bei Snap to Vertex wird die aktuelle Orientierung beibehalten.

3. SNAP Tags
   - Öffnen Sie das Tag Window (Tags Menu > Open Tag Window) und wählen Sie mit 

einen Tag aus.
   - Wählen Sie 

oder 

aus und klicken sie anschließend eine Fläche oder einen Punkt/Vertex in der Arbeitszelle.
   - Der Tag „springt“ an die gewählte Fläche bzw. an den gewählten Punkt. Bei Snap to Vertex wird die aktuelle Orientierung beibehalten.

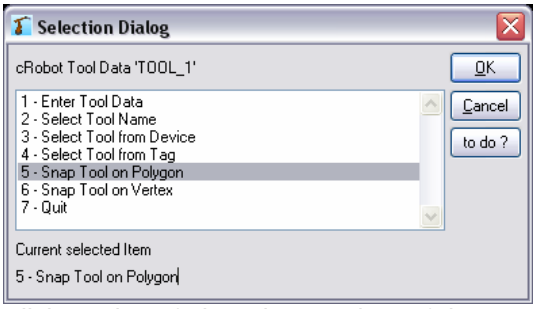
Tipp: Platzieren Sie zuerst auf eine Fläche um die Flächennormale zu erhalten und dann an einen gewünschten Punkt.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

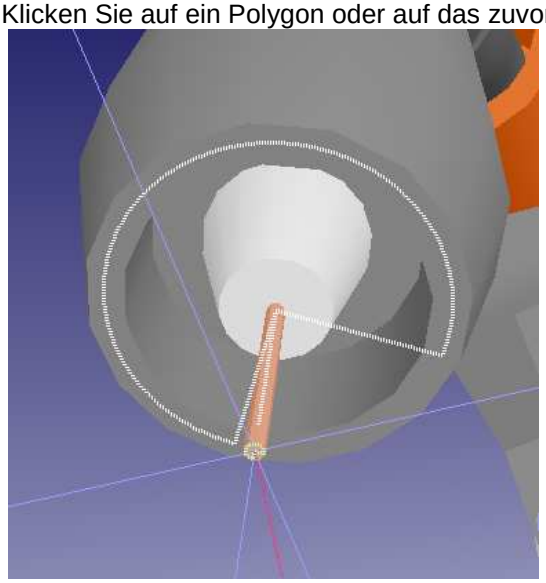
Vereinfachte Tool Definition

Beim Erzeugen von Werkzeugen wird in der Regel die Werkzeuggeometrie zum „Tool-Gerät“ geladen. Anschließend wird der TCP festgelegt und das Werkzeug als „.rob“ Geräte-Datei mit Bild gespeichert.

1. Tool Definition an einer Geometrie
   - Zoomen Sie die Geometrie in einen „guten“ sichtbaren Bereich.
   - Wählen aus dem Robotics Menu > cRobot Tool Data > 5 - Snap Tool on Polygon



   - Klicken Sie auf ein Polygon oder auf das zuvor gemessene Kreissegment (hier weiß oder grün).



   Der neue TCP „springt“ an die gewählte Fläche.
   Tipp: Platzieren Sie zuerst auf eine Fläche um die Flächennormale zu erhalten und dann an einen gewünschten Punkt.

2. Roboter TCP Definition von einem Tool-Gerät
   - Haben Sie einen Roboter geladen und ein Tool-Gerät an den Flansch befestigt, so können die Tool-Werte übernommen werden, indem Sie aus dem
     Robotics Menu > cRobot Tool Data > 3 - Select Tool from Device wählen
     und anschließend das entsprechende Tool-Gerät selektieren und „1 - Use cTool“ wählen.
   - Der Roboter hat nun die Tool-Werte vom Tool-Gerät übernommen.

Entsprechend können die Tool-Werte von Tag-Punkten übernommen werden.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

Zusätzliche TAG Attribute

Zusätzliche TAG Attribute ermöglichen es Informationen wie Verfahrart, Geschwindigkeiten, Beschleunigungen, Interpolationsverhalten, Turns, Wartezeiten sowie weitere Parameter auf der Bahn bzw. im Ziel im Tag abzulegen.

Durch die Verwendung von TAG Attributen vereinfacht sich das ERPL-Programm des Roboters und trägt zur Übersichtlichkeit erheblich bei. Weiterhin lassen sich die Attribute schnell ändern und so verschiedene Möglichkeiten durchspielen. Das Programm bleibt dabei unverändert.

TAG-Attribute:

- Motype	Verfahrt: PTP, LIN, VIA, CIRC
- Speed PTP Ov [%]	Prozentuale Geschwindigkeit für PTP
- Speed CP [mm/s]	CP Geschwindigkeit für LIN, CIRC
- Speed ORI [deg/s]	Orientierungs-Geschwindigkeit für LIN, CIRC
- Speed PTP Ax Ov [%]	Prozentuale Geschwindigkeit für PTP für jede Achse
- Accel PTP Ov [%]	Prozentuale Beschleunigung für PTP
- Accel CP [mm/s]	CP Beschleunigung für LIN, CIRC
- Accel ORI [deg/s]	Orientierungs- Beschleunigung für LIN, CIRC
- Accel PTP Ax Ov [%]	Prozentuale Beschleunigung für PTP für jede Achse
- AutoAccel ON/OFF	Automatische Berechnung der Beschleunigung in Abhängigkeit der programmierten Geschwindigkeit
- AccSet Acc [%]	Verzögerung der Beschleunigung [20% - 100%]
- AccSet Ramp [%]	Anstiegswert von Beschleunigung und Verzögerung [20% - 100%]
-	
- Configuration	Roboterkonfiguration, wird nur bei PTP verwendet
- No_Decel ON/OFF	ON – kein Abbremsen, Geschwindigkeit im Ziel ist gleich der programmierten Bahngeschwindigkeit. OFF – Abbremsen, Geschwindigkeit im Ziel ist 0 (default)
- Zone [mm,%]	Überschleifparameter 0-Finepunkt (default), >0 Überschleif-Wert
- Leading Position	0 - leading Orientierung, 1 - leading Position, 2-Variable (default)
- Lead Time [s]	Wartezeit vor der Bewegung
- Lag Time [s]	Wartezeit nach der Bewegung
- LIN Ori Mode	0-Var, 1-Fix, 2-Tang, 3-Aux, 4-Quaternionen (default)
- CIRC Ori Mode	0-Var, 1-Fix, 2-Tang, 3-Aux, 4-Var2, 5-Quaternionen (default)
- Turn Use ON/OFF	Use Turns if defined
- Turn_Ax	Turn-Wert für jede Achse des Roboters entsprechend dem definierten Turn-Intervall
- UserString	Benutzerdefinierter String für API

Damit die TAG Attribute im Programm verwendet werden, muss „Use Tag Attributes“ aktiv sein.

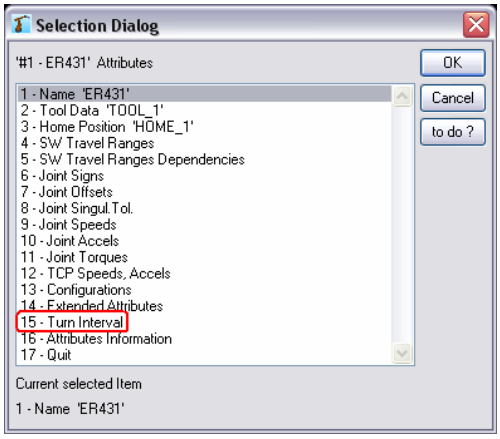
```
ERC USE_TAG_ATTRIBUTES ON/OFF
```


EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

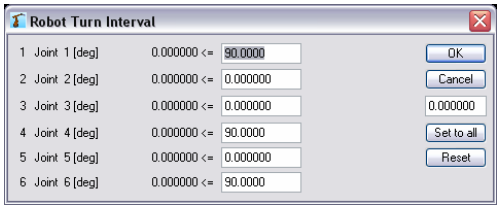
Bewegungsplanung mit Roboter TURNS

Die Bewegungsplanung berücksichtigt bei PTP Bewegungen TURN Vorgaben. Für Drehachsen, deren Verfahrbereich größer als 360° oder gar endlos ist, kann der entsprechende Quadrant vorgegeben werden, womit das so genannte Vordrehen oder Vorspannen der Achsen möglich wird. Entscheidend sind die definierten TURN-Intervalle des Roboters, die z.B. bei ABB-Robotern 90° und bei KUKA- oder Fanuc-Robotern 180° sind.

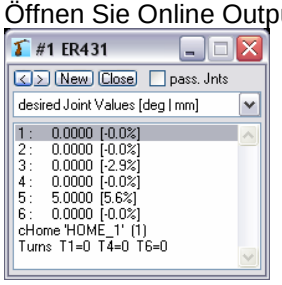
Laden Sie den Roboter ER431 aus der Standard Roboter Bibliothek.
Öffnen Sie das Kinematics Window (Ctrl+K) und klicken Sie auf Button „Attributes“ und wählen Sie „15 - Turn Interval“ aus



Definieren Sie für die Achsen 1,4 und 6 ein Turn-Intervall von jeweils 90°.

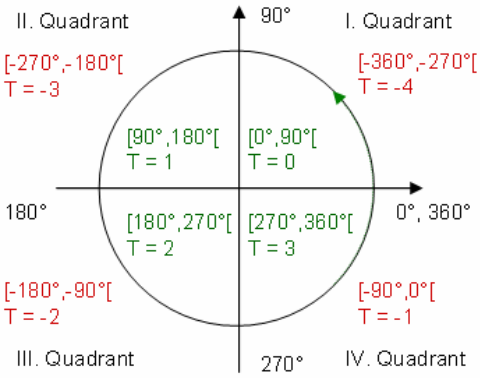


Öffnen Sie Online Output Data 



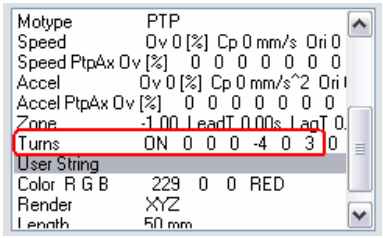
Die Achswinkel und Turn-Werte der Achsen 1,4 und 6 werden angezeigt: T1=0 T4=0 T6=0

Verfahren Sie Achse 4 z.B. nach -60° und Achse 6 nach +120° ergeben sich die Turns T4=-1 T6 = 1



Beispiel, Turn-Intervall 90°

Turns können in Tags abgelegt oder programmiert werden. Beispiel:
„`Tutorial\Proj_example_erpl\functions_turns.cel`“



Tag „T6“ verwendet Turns (ON) T4=-4, T6=3 (Achse 4 = -360°, Achse 6 = 360°)

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

ERCL - Befehle

```
ERC TURN_INTERVAL Ax1 ...Axn [deg]

PTP_CALC_MODE calc_mode

mit calc_mode =
       SHORTEST_ANGLE,
       MATH,
       IN_TRAVEL_RANGE,TRAVEL_RANGE
       TURN

ERC TURN Turn_Ax1 ... Turn_Axn
```

Per default steht der PTP_CALC_MODE auf SHORTEST_ANGLE. Die Drehwinkel im Ziel werden in Abhängigkeit der aktuellen Drehwinkelstellung berechnet.

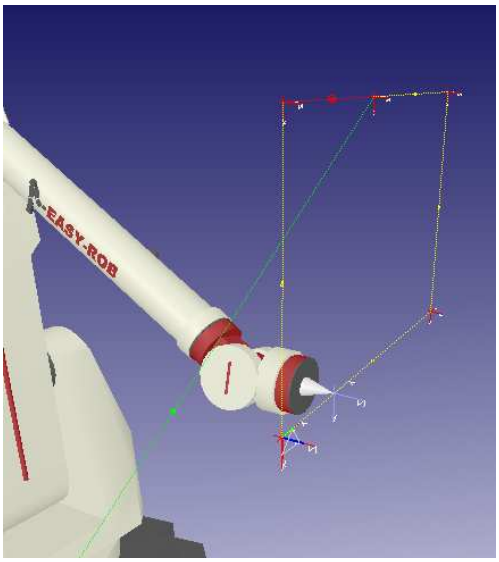
Bei MATH werden die Drehwinkel zwischen [-180,180°] berechnet.

Bei IN_TRAVEL_RANGE oder TRAVEL_RANGE werden die Drehwinkel im gültigen Verfahrbereich gerechnet, sofern möglich.

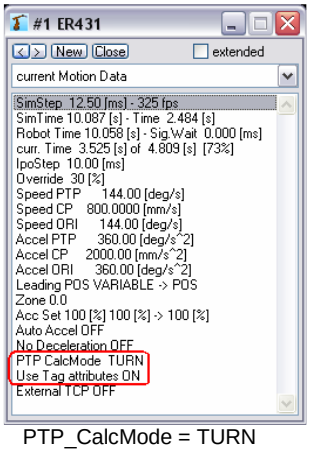
Bei TURN werden die vorgegebenen Turn Werte zur Berechnung der Drehwinkel im Ziel verwendet. Sind die Turn-Werte ungültig, wird die PTP Position nicht angefahren.
Fehler: UNREACH

ERCL-Beispiel:

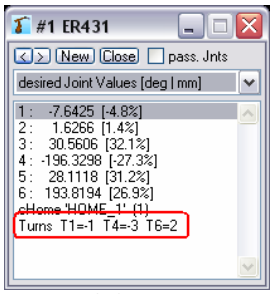
```
ERC TURN_INTERVAL 90 0 0 90 0 90
PTP_CALC_MODE TURN
TURN 0 @ @ -2 0 1
PTP T_1
PTP_CALC_MODE SHORTEST_ANGLE
PTP T_2
```



Beispiel: "functions_turns.cel"



PTP_CalcMode = TURN
Use Tag Attributes = ON



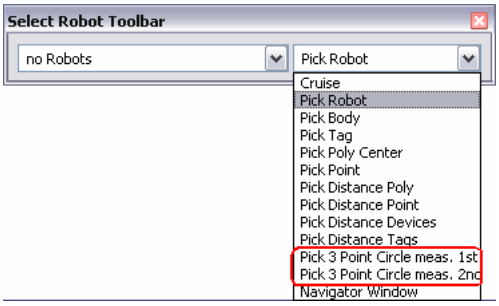
Ax1 = -7.64° -> T1 = -1
Ax4 = -196.3° -> T4 = -3
Ax6 = 193.8° -> T6 = 2

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

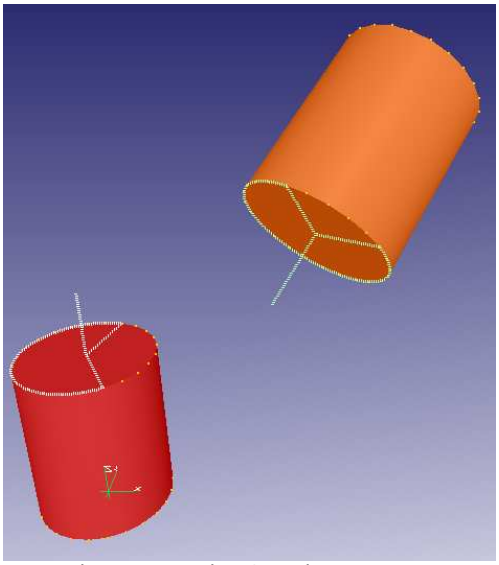
Messfunktionen

Das Messen von Kreisen und Abstände zwischen Geräten, Kreismittelpunkten, Flächen und Punkten wurde verbessert. Insbesondere ist das Vermessen von Kreisen bei der Erstellung von Kinematiken hilfreich, wenn nur Geometrien und keine technischen Daten (Abstände zwischen den Drehachsen) vorliegen.

Die Aktion kann vom „Select Robot Toolbar“ oder vom Navigator Window aus gestartet werden.

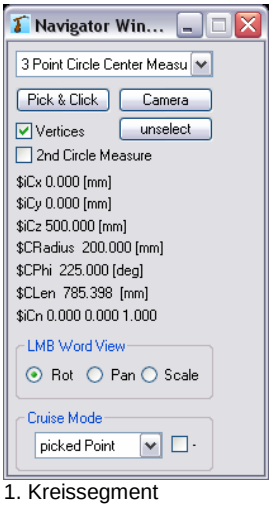


Um den Abstand zweier Kreise zu messen, wählen Sie zur Bestimmung des ersten Kreises:„Pick 3 Point Circle meas. 1st“ und im Anschluss für den zweiten Kreis „Pick 3 Point Circle meas. 2nd“ aus.

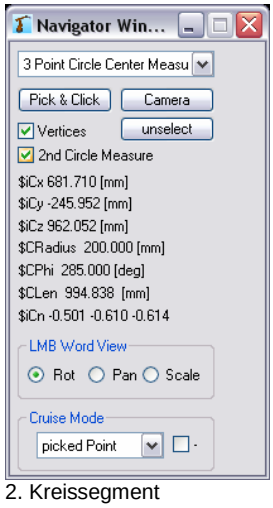


1. Kreissegment weiss, 2. Kreissegment grün

Die Ergebnisse können dem Navigator Window entnommen werden

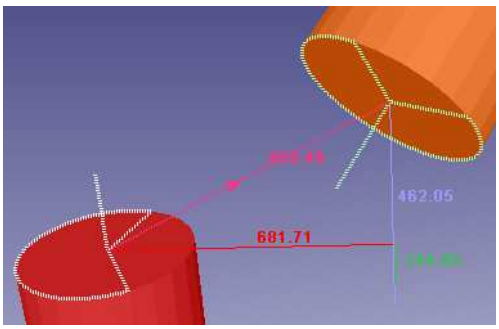


1. Kreissegment



2. Kreissegment

Um den Abstand zu messen wählen Sie z.B. „Pick Distance Poly“ und wählen die Kreissegmente an.



Abstand zweier Kreissegmente.

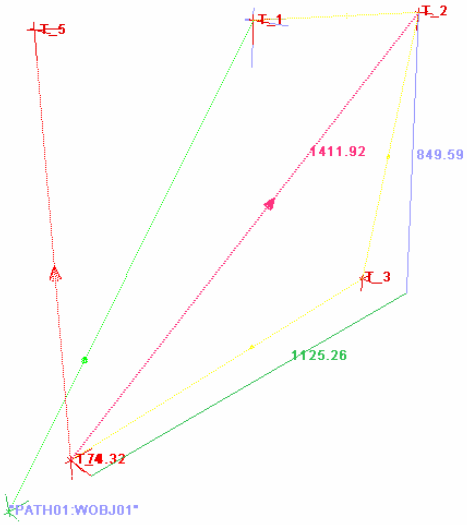
Kreissegmente können auch für die SNAP-Funktion angewählt werden, um Devices, Geräte, Geometrien oder Tags zu platzieren. Hierbei wird die Flächennormale übernommen.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

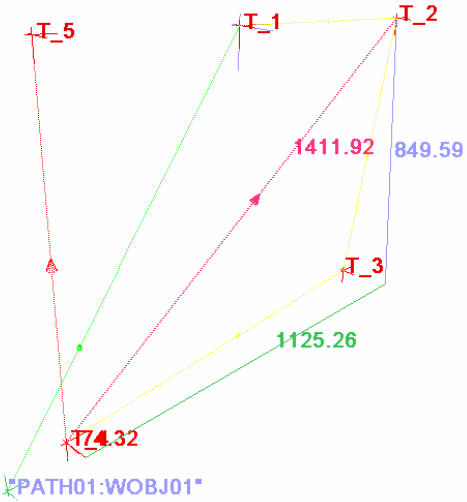
Einstellen der Schriftgröße

Bildschirme und höherwertige Notebooks haben teilweise eine sehr hohe Auflösung, so dass Beschriftungen von Geometrien und Tags in der 3d Szene kaum lesbar sind. In der neuen Version läßt sich die Schriftgröße für Geometrien und Tags/Pfade unabhängig voneinander einstellen und in der Environment Datei „easy-rob.env“ speichern.

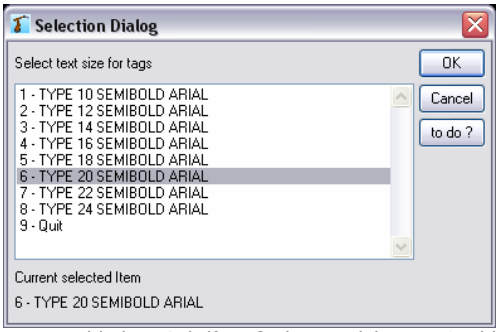
Wählen sie im View Menu:
Render > Text Size Standard
oder
Render > Text Size für Tags & Pathes



Beispiel: Standard-Schriftgröße, TYPE 12 SEMIBOLD ARIAL



Beispiel: Schriftgröße, TYPE 20 SEMIBOLD ARIAL



Auswahl einer Schriftgröße im Bereich von 10pt bis 24pt

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

ERPL-Befehle

PTP_CALC_MODE SHORTEST_ANGLE [TURN, MATH, IN_TRAVEL_RANGE]

Berechnungsvorschrift bei PTP-Bewegungen

SHORTEST_ANGLE	– kürzester Winkel
TURN	– nach TURN-Vorgabe
MATH	– mathematisch innerhalb [-180°,180°]
IN_TRAVEL_RANGE	– innerhalb gültiger Verfahrbereiche wenn möglich

TURN Turn_Ax1 ... Turn_Axn

Vorgabe der TURN-Werte für jede Achse Ax1…Axn, für die folgende PTP Ziel-Position

ERCL-Befehle

ERC TURN_INTERVAL Ax1 ...Axn [deg]

TURN -Intervalle für jede Achse Ax1…Axn, im Bereich zwischen [0°,360°[

ERC TRANSPARENCY BODY [ROBOT,TOOL] bodyname alpha

Setzt / modifiziert die Transparenz für ein einzelnes Part bzw. einem Body. alpha = [0,1]

ERC TRANSPARENCY BODY_GRP [ROBOT_GRP,TOOL_GRP] alpha

Setzt / modifiziert die Transparenz für alle Parts in einer Gruppe. alpha = [0,1]

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

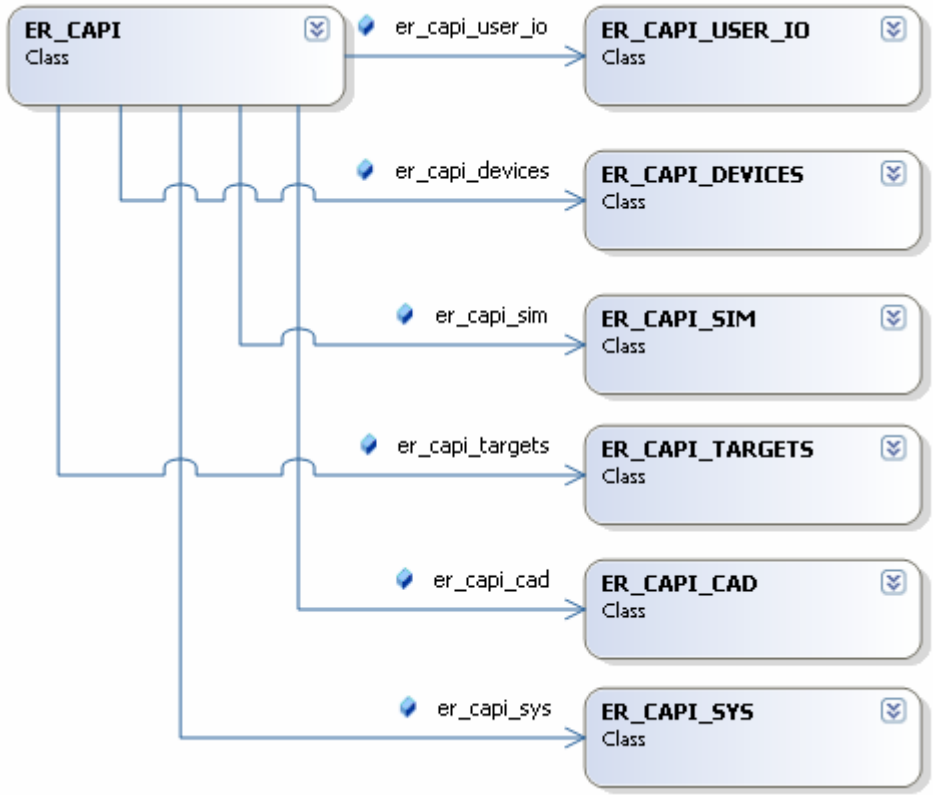
Neue Methoden-Klasse ER_CAPI

ER_CAPI

Die neue exportierte Klasse ER_CAPI strukturiert sämtliche EASY-ROB™ API-Funktionen und vereinfacht die Verwendung. Die Klasse ER_CAPI ist eine reine Methoden-Klasse, die in der Header-Datei „./er_dvlp/**er_capi.h**“ definiert sind. Sämtliche Methoden sind Standard ANSI C.

„Alte“ ANSI C Funktionen, die in den Header-Dateien „./er_dvlp/**er_dvlp.h**“ und „./er_dvlp/**er_dvlp_ext.h**“ definiert sind, sind aus Kompatibilitätsgründen weiterhin verfügbar.

Die Methoden-Klasse ER_CAPI ist in folgende Klassen, grobe Gliederung, aufgeteilt.



ER_CAPI_USER_IO: Methoden-Klassen zur Interaktion mit EASY-ROB™

ER_CAPI_USER_IO_FILE	Laden, speichern von Zellen, Robotern, etc.
ER_CAPI_USER_IO_DIALOG	Stellt Dialoge zur Verfügung, um z.B. Werte anzuzeigen
ER_CAPI_USER_IO_PICK	Methoden zum „Picken“ von Objekten etc.
ER_CAPI_USER_IO_CRUISE	Methoden zum Manipulieren der 3D Szene

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

ER_CAPI_DEVICES: Methoden-Klassen zum Erzeugen, Verknüpfen und Aktualisieren von Geräten, sowie für kinematische Berechnungen, Bewegungsplanung und -ausführung

ER_CAPI_ROB	Kinematik und Transformation
ER_CAPI_MOP	Bewegungsplanung und -ausführung

ER_CAPI_ROB: Methoden-Klassen für Kinematik und Transformation

ER_CAPI_ROB_KIN	Vorwärts-, Inverse-Kinematik, Sollwerte, Tools, Position bzgl. Welt und Referenz-System
ER_CAPI_ROB_ATRIBUTES	Roboter/Geräte-Attribute, Verfahrbereiche, max. Geschwindigkeiten, HomePositionen, etc.
ER_CAPI_ROB_DYN	Dynamik, Reglerparameter, Abtastzeiten, Istwerte

ER_CAPI_MOP: Methoden-Klassen für Bewegungsplanung und -ausführung

ER_CAPI_MOP_DATA	Start-, Zielwerte, Zeiten, etc.
ER_CAPI_MOP_PATH	Bahnvorgaben, Bewegungsart (PTP,LIN,CIRC), Geschwindigkeiten, Beschleunigungen, Waits, etc.
ER_CAPI_MOP_PREP	Bewegungsplanung
ER_CAPI_MOP_EXEC	Bewegungsausführung

ER_CAPI_SIM: Methoden-Klassen für Simulationseinstellungen

ER_CAPI_SIM_MONITORING	Überwachung z.B. Verfahrbereiche, etc.
ER_CAPI_SIM_COLLISION	Kollision, Toleranz, etc.
ER_CAPI_SIM_CAMERA	Kameraeinstellungen
ER_CAPI_SIM_ERPL	Ausführen von ERPL- und ERCL-Befehlen
ER_CAPI_SIM_TRACK	TCP-Spur, Referenz und Visualisierung

ER_CAPI_TARGETS: Methoden-Klassen für Pfade und Tags

ER_CAPI_TARGETS_TAG	Manipulation von Tags, Zugriff auf Tag Attribute
ER_CAPI_TARGETS_PATH	Erzeugen, Löschen und Manipulation von Pfaden

ER_CAPI_CAD: Methoden-Klassen für 3D CAD Daten-Import und -Export

ER_CAPI_CAD_IO	Zugriff auf Geometrien, deren Lage, Tesselierung, etc.
ER_CAPI_CAD_CREATE	Erzeugen von Primitiven (Würfel, Zylinder, Kugel, etc.)
ER_CAPI_CAD_IMPORT	Importieren vorhandener 3D Geometrien (IGP, STL, 3DS)
ER_CAPI_CAD_CONVERT	Konvertieren von externen neutralen und nativen CAD Formaten (STEP, IGES, JT-Open, CATIA, Pro/E, etc.) nach IGP Part File Format.

ER_CAPI_SYS: Methoden-Klassen für mathematische Berechnungen, Simulationsstatus, Einheiten

ER_CAPI_SYS_UTILITIES	Hilfsfunktionen
ER_CAPI_SYS_MATHEMATICS	Mathematische Berechnungen, Multiplikation von homogenen Matrizen, Umrechnung Eulerwinkel, Dreiecksberechnungen, Formel-Parser, etc.
ER_CAPI_SYS_VIEW	Grafischer Update der 3D Szene, Refreshen von Dialogen
ER_CAPI_SYS_PREVIEW	CAD-Preview
ER_CAPI_SYS_STATUS	Entladen von Objekten (Zelle, Robotern, Tools, Programme, etc.). Simulationsstatus
ER_CAPI_SYS_UNITS	Setzen und Verrechnen von Einheiten

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

Anwendung

Um die Klasse ER_CAPI zu verwenden, sind die folgenden drei Zeilen einzubinden.

```
#define ER_DllExport
#include "../er_dvlp/er_capi_types.h"
#include "../er_dvlp/er_capi.h"
```

Für den schnellen Zugriff auf die Methoden der einzelnen Klassen können die im Header „er_capi.h“ vordefinierten statischen Variablen verwendet werden.

```
// main class
static ER_CAPI                    er_capi;                // 0        CAPI

// class ER_CAPI                                          // 0
static ER_CAPI_USER_IO            er_user_io;             // 1.
static ER_CAPI_DEVICES            er_devices;             // 2.
static ER_CAPI_SIM                er_sim;                 // 3.
static ER_CAPI_TARGETS            er_targets;             // 4.
static ER_CAPI_CAD                er_cad;                 // 5.
static ER_CAPI_SYS                er_sys;                 // 6.

// class ER_CAPI_USER_IO                                  // 1.       USER_IO
static ER_CAPI_USER_IO_FILE       er_user_io_file;        // 1.1.
static ER_CAPI_USER_IO_DIALOG     er_user_io_dialog;      // 1.2.
static ER_CAPI_USER_IO_PICK       er_user_io_pick;        // 1.3.
static ER_CAPI_USER_IO_CRUISE     er_user_io_cruise;      // 1.4.

// class ER_CAPI_DEVICES                                  // 2.       DEVICES
static ER_CAPI_ROB                er_rob;                 // 2.1.
static ER_CAPI_MOP                er_mop;                 // 2.2.

// class ER_CAPI_ROB                                      // 2.1  ROB
static ER_CAPI_ROB_KIN            er_rob_kin;             // 2.1.1
static ER_CAPI_ROB_ATRIBUTES      er_rob_attributes;      // 2.1.2
static ER_CAPI_ROB_DYN            er_rob_dyn;             // 2.1.3

// class ER_CAPI_MOP                                      // 2.2  MOP
static ER_CAPI_MOP_DATA           er_mop_data;            // 2.2.1.
static ER_CAPI_MOP_PATH           er_mop_path;            // 2.2.2.
static ER_CAPI_MOP_PREP           er_mop_prep;            // 2.2.3.
static ER_CAPI_MOP_EXEC           er_mop_exec;            // 2.2.4.

// class ER_CAPI_SIM                                      // 3.       SIM
static ER_CAPI_SIM_MONITORING     er_sim_monitoring;      // 3.1.
static ER_CAPI_SIM_COLLISION      er_sim_collision;       // 3.2.
static ER_CAPI_SIM_CAMERA         er_sim_camera;          // 3.3.
static ER_CAPI_SIM_ERPL           er_sim_erpl;            // 3.4.
static ER_CAPI_SIM_TRACK          er_sim_track;           // 3.5.

// class ER_CAPI_TARGETS                                  // 4.       TARGETS
static ER_CAPI_TARGETS_TAG        er_targets_tag;         // 4.1.
static ER_CAPI_TARGETS_PATH       er_targets_path;        // 4.2.
```


EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

```
// class ER_CAPI_CAD                                    // 5.       CAD
static ER_CAPI_CAD_IO              er_cad_io;           // 5.1.
static ER_CAPI_CAD_CREATE          er_cad_create;       // 5.2.
static ER_CAPI_CAD_IMPORT          er_cad_import;       // 5.3.
static ER_CAPI_CAD_CONVERT         er_cad_convert;      // 5.4.

// class ER_CAPI_SYS                                    // 6.       SYS
static ER_CAPI_SYS_UTILITIES       er_sys_utilities;    // 6.1.
static ER_CAPI_SYS_MATHEMATICS     er_sys_mathematics;  // 6.2.
static ER_CAPI_SYS_VIEW            er_sys_view;         // 6.3.
static ER_CAPI_SYS_PREVIEW         er_sys_preview;      // 6.4.
static ER_CAPI_SYS_STATUS          er_sys_status;       // 6.5.
static ER_CAPI_SYS_UNITS           er_sys_units;        // 6.6.
```

Einfache Beispiele

1. Lesen und Ausgeben der aktuellen Roboterwinkel

```
float *q = er_rob_kin.inq_q_solut(); // pointer to desired joint data
int n_dof = er_rob_kin.inq_num_active_jnts(); // number of robot joints
// show cJoint in Message Window
er_user_io_dialog._info_line_msg_q(0,"Joints",q);
```

2. Ausführen eines Einzelsatzes “Home 1”

```
// Check if a Cell or Robot is loaded
if (er_sys_status.ChkRobotCellLoaded(1)==1) return;
// Check if a program is currently running
if (er_sim_monitoring.ChkPrgRunning(0)) return;
er_sim_erpl.RunProgram("Home 1");
```

3. Ausführen eines Programms

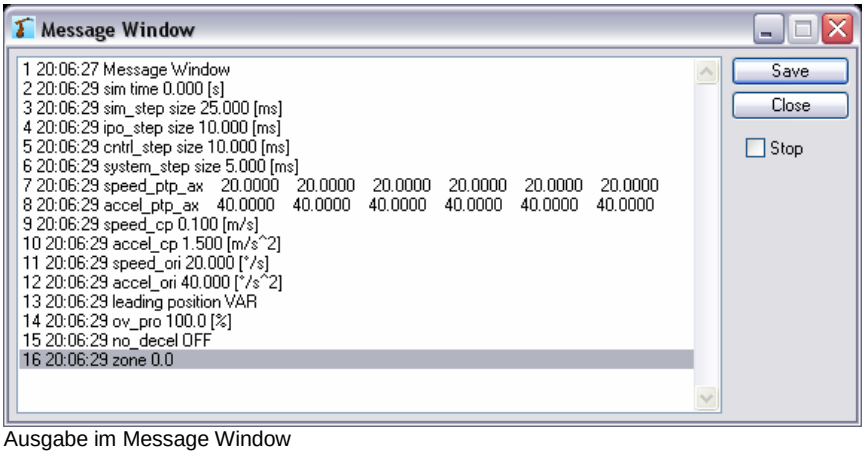
```
// Check if a Cell or Robot is loaded
if (er_sys_status.ChkRobotCellLoaded(1)==1) return;
// Check if a program is currently running
if (er_sim_monitoring.ChkPrgRunning(0)) return;
er_sim_erpl. RunProgram();
```


EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

4. Lesen der aktuellen Bahnvorgaben des Motion-Planners

```
er_user_io_dialog._info_line_msg(0,"sim time %.3f [s]",
       *er_sim.inq_sim_time());
er_user_io_dialog._info_line_msg(0,"sim_step size %.3f [ms]",
       *er_sim.inq_sim_step()*m2mm);
er_user_io_dialog._info_line_msg(0,"ipo_step size %.3f [ms]",
       *er_mop_data.inq_ipo_data_dt_ipo()*m2mm);
er_user_io_dialog._info_line_msg(0,"cntrl_step size %.3f [ms]",
       *er_rob_dyn.inq_cntrl_step()*m2mm);
er_user_io_dialog._info_line_msg(0,"system_step size %.3f [ms]",
       *er_rob_dyn.inq_system_step()*m2mm);

int n_dof = er_rob_kin.inq_num_active_jnts();
er_user_io_dialog._info_line_msg_v(0,"speed_ptp_ax",
       er_mop_path.inq_ipo_path_vq_axis(),n_dof,1);
er_user_io_dialog._info_line_msg_v(0,"accel_ptp_ax",
       er_mop_path.inq_ipo_path_aq_axis(),n_dof,1);
er_user_io_dialog._info_line_msg(0,"speed_cp %.3f [m/s]",
       *er_mop_path.inq_ipo_path_vx());
er_user_io_dialog._info_line_msg(0,"accel_cp %.3f [m/s^2]",
       *er_mop_path.inq_ipo_path_ax());
er_user_io_dialog._info_line_msg(0,"speed_ori %.3f [°/s]",
       *er_mop_path.inq_ipo_path_vx_ori());
er_user_io_dialog._info_line_msg(0,"accel_ori %.3f [°/s^2]",
       *er_mop_path.inq_ipo_path_ax_ori());
er_user_io_dialog._info_line_msg(0,"leading position %s",
       *er_mop_path.inq_ipo_path_leading_position()==IPO_LEADING_POS?"POS" :
       *er_mop_path.inq_ipo_path_leading_position()==IPO_LEADING_ORI?"ORI" :
       "VAR");
er_user_io_dialog._info_line_msg(0,"ov_pro %.1f [%%]",
       *er_mop_path.inq_ipo_path_ov_pro());
er_user_io_dialog._info_line_msg(0,"no_decel %s",
       *er_mop_path.inq_ipo_path_no_decel()?"ON":"OFF");
er_user_io_dialog._info_line_msg(0,"zone %.1f",
       *er_mop_path.inq_ipo_path_zone());
```



Ausgabe im Message Window

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

API-Methoden

USER_IO_PICK

- **int ER_CAPI_USER_IO_PICK::PickSnapDeviceTo(int SnapToMode=0)**

  // Enables the user to snap a device to a polygon or vertex
  // SnapToMode 0-polygon, 1-vertices

- **int ER_CAPI_USER_IO_PICK::PickSnapToolTo(int SnapToMode=0)**

  // Enables the user to snap a tool to a polygon or vertex
  // SnapToMode 0-polygon, 1-vertices

- **int ER_CAPI_USER_IO_PICK::PickSnapTagTo(int SnapToMode=0)**

  // Enables the user to snap a tag to a polygon or vertex
  // SnapToMode 0-polygon, 1-vertices

- **int ER_CAPI_USER_IO_PICK::PickSnapBodyTo(int SnapToMode=0)**

  // Enables the user to snap a body to a polygon or vertex
  // SnapToMode 0-polygon, 1-vertices

- **int ER_CAPI_USER_IO_PICK::PickDistanceDeviceMeasure(int hold_first_point=0)**

  // Enables the user to measure the distance between two devices
  // hold_first_point 1-holds first selected point

- **int ER_CAPI_USER_IO_PICK::PickDistanceTagMeasure(int hold_first_point=0)**

  // Enables the user to measure the distance between two tags
  // hold_first_point 1-holds first selected point

MOP_PATH_DATA

- **int *ER_CAPI_MOP_PATH::inq_ipo_path_path_motion_idx(void)**

  // Aktueller Path-Index
  // 0 not a tag motion,
  // >0 Index of cPath containing the TargetTag
  Return Pointer to value

- **int *ER_CAPI_MOP_PATH::inq_ipo_path_tag_motion_idx(void)**

  // Aktueller Tag-Index
  // 0 not a tag motion,
  // >0 Index of TargetTag
  Return Pointer to value

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

- **int ER_CAPI_USER_IO_PICK::Pick3PointCircleMeasure (int vertices=1, int second_circle_measure=0)**

  // Enables the user to measure the center of a circle by picking three points
  // vertices 0-pcik polygon, 1-pick vertex
  // second_circle_measure 0-measure 1st-, 1-measure 2nd circle

TARGETS_TAG

Methods to read and set tag attributes, such as name, speed and accels.

- **int ER_CAPI_TARGETS_TAG::cTagClone(int create_new=0)**

  // clones current Tag in current path at Tcp of cRobot
  // if cPath is empty a new tag will be created on TCP if create_new is set 1

- **int ER_CAPI_TARGETS_TAG::chk_rob_tag_idx(int idx)**

  // check for valid tag_idx, 0-OK, 1-idx does not exist

- **char *ER_CAPI_TARGETS_TAG::inq_rob_tag_prefix_idx(int idx)**

  // tag prefix name

- **char *ER_CAPI_TARGETS_TAG::inq_rob_tag_name_idx(int idx)**

  // tag name

- **char *ER_CAPI_TARGETS_TAG::inq_rob_tag_prefix_name_idx(int idx)**

  // complete tag name

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_motype_idx(int idx)**

  // tag motion type: TAG_MOTYPE_DEF, TAG_MOTYPE_PTP, TAG_MOTYPE_LIN
  // TAG_MOTYPE_CIRC, TAG_MOTYPE_VIA

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ptp_idx(int idx)**

  // ptp speed, 0-def, [rad/s] for tag_idx

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_cp_idx(int idx)**

  // cp speed, 0-def, [m/s] for tag_idx

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ori_idx(int idx)**

  // ori speed, 0-def, [rad/s] for tag_idx

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_confdata_idx(int idx)**

  // robot configuration, 0-def

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_zone_idx(int idx)**

  // programmed zone value [mm,%]

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ptp_idx(int idx)**
  // ori accel, 0-def, [rad/s^2] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_cp_idx(int idx)**
  // cp accel, 0-def, [rad/s^2] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ori_idx(int idx)**
  // ori accel, 0-def, [rad/s^2] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_ov_pro_idx(int idx)**
  // programmed override, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ptp_ov_idx(int idx)**
  // programmed override PTP speed, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_cp_ov_idx(int idx)**
  // programmed override CP speed, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ori_ov_idx(int idx)**
  // programmed override ORI speed, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ptp_ov_idx(int idx)**
  // programmed override PTP accel, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_cp_ov_idx(int idx)**
  // programmed override CP accel, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ori_ov_idx(int idx)**
  // programmed override ORI accel, 0-def, [0-200%] for tag_idx
- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_auto_accel_idx(int idx)**
  // auto acceleration, 0-def, 1-ON for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_acc_set_acc_idx(int idx)**
  // AccSet Acc, 100-def [20-100%] for tag_idx

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

- **`float *ER_CAPI_TARGETS_TAG::inq_rob_tag_acc_set_ramp_idx(int idx)`**

  // AccSet Ramp, 100-def [20-100%] for tag_idx

- **`int *ER_CAPI_TARGETS_TAG::inq_rob_tag_no_decel_idx(int idx)`**

  // No decelaration at target, 0-def, 1-enabled for tag_idx

- **`int *ER_CAPI_TARGETS_TAG::inq_rob_tag_leading_pos_idx(int idx)`**

  // 2-def, 1-Pos, 2-VAR, 0-Ori for tag_idx

- **`float *ER_CAPI_TARGETS_TAG::inq_rob_tag_lead_time_idx(int idx)`**

  // wait time before move begins, 0-def [s] for tag_idx

- **`float *ER_CAPI_TARGETS_TAG::inq_rob_tag_lag_time_idx(int idx)`**

  // wait time when target reached, 0-def [s] for tag_idx

- **`int *ER_CAPI_TARGETS_TAG::inq_rob_tag_lin_ori_mode_idx(int idx)`**

  // 4-def, 0-LIN_VARIABLE, 1-LIN_FIX, 2-LIN_TANGENTIAL, 3-LIN_AUX,
  // 4-LIN_QUATERNION for tag_idx

- **`int *ER_CAPI_TARGETS_TAG::inq_rob_tag_circ_ori_mode_idx(int idx)`**

  // 5-def, 0-CIRC_VARIABLE, 1-CIRC_FIX, 2-CIRC_TANGENTIAL, 3-CIRC_AUX,
  // 4-CIRC_VARIABLE2, 5-CIRC_QUATERNION for tag_idx

- **`float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ptp_ax_ov_idx(int idx)`**

  // ptp_ax override speed 0-def [0-200%] for tag_idx

- **`float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ptp_ax_ov_idx(int idx)`**

  // ptp_ax override accel 0-def [0-200%] for tag_idx

- **`int *ER_CAPI_TARGETS_TAG::inq_rob_tag_turn_use_idx(int idx)`**

  // 0-do not use turn values (def), 1-use turns for tag_idx

- **`int *ER_CAPI_TARGETS_TAG::inq_rob_tag_turn_ax_idx(int idx)`**

  // turn values depending on current robots turn interval for tag_idx

- **`char *ER_CAPI_TARGETS_TAG::inq_rob_tag_user_string_idx(int idx)`**

  // user defined Tag string for tag_idx

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

TARGETS_PATH

Methods to reattach a path with index 'path_idx' by device idx, name or unique robot ID ‘uid’

path_idx	idx of path to reattach to
new_reference_type	new referecne type = REF_BASE=2, REF_TOOL=3, REF_WORLD=4, REF_CAD=7, REF_TIP=8, 9-REF_JNT
new_reference_device_idx	get reference device by idx
new_reference_cad_grp_type	reference grp_type = (UNDEF_GRP,ROBOT_GRP, TOOL_GRP,BODY_GRP) if new_reference_type = REF_CAD
new_reference_cad_name	reference cad name if new_reference_type = REF_CAD
new_reference_jnt_idx	reference jnt idx if new_reference_type = REF_JNT
keep_world_position	True will keep the world position of path
Return:	0-OK, 1-Error

- **int Path_ReAttach_by_name(int path_idx, int new_reference_type,**
  char *new_reference_device_name,
  int new_reference_cad_grp_type=ROBOT_GRP,
  char *new_reference_cad_name=NULL, int new_reference_jnt_idx=0,
  bool keep_world_position=false)

  // Reattaches the path with index 'path_idx' by device name

- **int Path_ReAttach_by_idx (int path_idx, int new_reference_type,**
  int new_reference_device_idx,
  int new_reference_cad_grp_type=ROBOT_GRP,
  char *new_reference_cad_name=NULL, int new_reference_jnt_idx=0,
  bool keep_world_position=false)

  // Reattaches the path with index 'path_idx' by device idx

- **int Path_ReAttach_by_uid (int path_idx, int new_reference_type,**
  long new_reference_device_uid,
  int new_reference_cad_grp_type=ROBOT_GRP,
  char *new_reference_cad_name=NULL, int new_reference_jnt_idx=0,
  bool keep_world_position=false)

  // Reattaches the path with index 'path_idx' by device uid

CAD_IO

- **int *ER_CAPI_CAD_IO::inq_body_show_transparent(void *body_handle)**

  // return pointer to show_transparent flag
  // body_handle geometry handle

- **float *ER_CAPI_CAD_IO::inq_body_transparency(void *body_handle)**

  // return pointer to transparency value
  // body_handle geometry handle

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

Windows® 7 32- und 64-Bit

EASY-ROB™ ist kompatibel mit Windows® 7 Professional Ultimate & Enterprise (32-Bit und 64-Bit). Entscheidend für eine gute Performance ist eine gute Grafikkarte. Wir empfehlen Grafikkarten mit nVIDIA GeForce oder ATI Grafikchip mit OpenGL™ 2.0 Treiber und mindestens 256MB VRAM.

Weiterhin empfehlen wir 4GB RAM, so dass Windows® als 32-Bit Version den maximal möglichen Arbeitsspeicher für die Applikationen bereitstellen kann. Bei Windows® 7 werden bis zu 3.25 GB bereitgestellt.

Aktuell kann festgestellt werden, dass CPU und Grafikperformance durchaus ausreichend sind. Probleme bereitet meistens der zu geringe Arbeitsspeicher bei 32-Bit Applikationen, wie derzeit auch EASY-ROB™. Das Problem macht sich besonders bei Geometrien mit einem hohen Detaillierungsgrad bemerkbar. Umso wichtiger sind die Möglichkeiten des CAD Daten-Imports die Tesselierung beeinflussen zu können.

Um das Speicher-Ressourcen-Problem in den Griff zu bekommen, wird zukünftig an eine 64-Bit Lösung gearbeitet.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

Kontakt

EASY-ROB 3D Robot Simulation Tool

Stefan Anton
Hans - Thoma - Str. 26a, 60596 Frankfurt/Main, Germany
Tel. +49 (0) 69 677 24 287
Fax. +49 (0) 69 677 24 320

Email: contact@easy-rob.com
sales@easy-rob.com

Web: www.easy-rob.com

EASY-ROB Kundenbereich

Online verfügbar: Programm-Updates und Roboterbibliotheken

Web: www.easy-rob.com/special/kundenbereich

Zugangsdaten:

Benutzer: customer
Passwort: **********

Eigene Notizen

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com