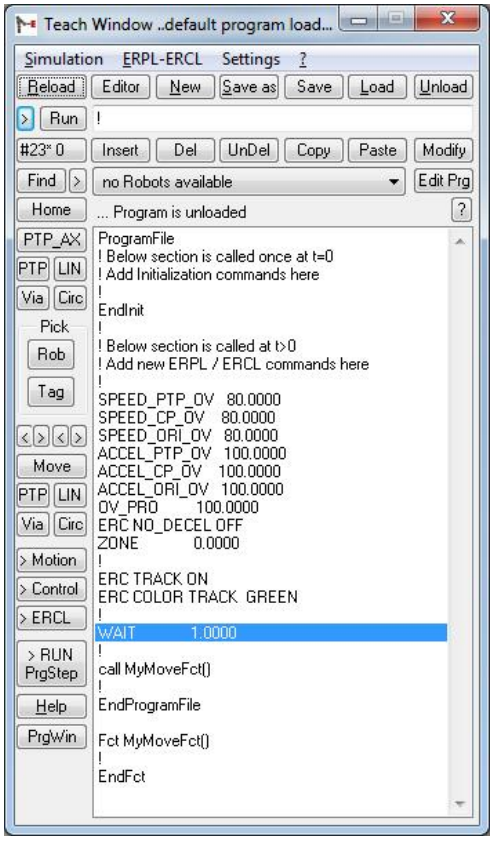
ERPL- / ERCL- Programmiersprache

EASY-ROB™ V8.0



November 2019

Version 3.4

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™

Inhaltsverzeichnis

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™

ERPL - EASY-ROB Programmier Sprache

Die untenstehenden Tabellen geben einen Überblick über die EASY-ROB Programmstruktur und die verfügbaren Roboterbewegungskommandos.

Grundsätzliche Definition des Syntax:

- Einheiten sind (Länge) Meter [m], (Winkel) [deg] oder Prozent [%]
- Geschwindigkeitseinheiten sind in Längeneinheit pro Sekunde angegeben, z.B. [m/s]
- Eine kartesische Position besteht aus einer Position mit X, Y und Z Wert und einer Orientierung mit A, B und C Winkel.
  Die Orientierungsdefinition für ABC Winkel ist:
  Rot (A,B,C) = Rot(X,A) * Rot(Y,B) * Rot(Z,C)
- Ein Tagpunktname ist z.B. 'T_1'. Um den Tagpunkt zu nutzen und mit einem Bewegungskommando abzufahren, muss der Punkt in der Arbeitszelle existieren.

Allgemeine Programm Struktur

Kommando und Syntax	**Beschreibung**
PROGRAMFILE	Anfang des Programms Bei Ausführung dieses Kommandos als Einzelschritt werden Status Daten wie BASE, BASE_PRG, etc. zurückgesetzt
ENDPROGRAMFILE oder END	Ende des Programms Programmausführung wird gestoppt
CALL fct_name()	Interner Funktionsaufruf fct_name() - Name des Funktion **Anm.**: Die Funktion muss in dem aktuellen Programm existieren
FCT fct_name()	Beginn der Funktionsdefinition fct_name() = Name der Funktion
ENDFCT	Ende der Funktion
! Text und Kommentare	Das '!' -Zeichen leitet einen Kommentar ein – alles nachfolgende in der Zeile wird ignoriert
EndInit	Ende des Initialisierungsbereichs (für alle „nicht-Zeit-beeinflussenden“ Kommandos

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Roboter-Bewegungs Kommandos

Kommando und Syntax	Beschreibung
! Text und Kommentare	Das '!' -Zeichen leitet einen Kommentar ein
OV_PRO x [%]	programmierbarer Override x - Prozentwert
SPEED_CP dx dxe [m/s]	Geschwindigkeit für lineare Bewegung dx - kartesische Geschwindigkeit [dxe] - kartesische Geschwindigkeit am Ziel
SPEED_ORI dx_ori dxe_ori [deg/s]	Orientierungs-Geschwindigkeit für lineare Bewegung dx_ori - Orientierungs-Geschwindigkeit [dxe_ori] - Orientierungs-Geschwindigkeit am Ziel
SPEED_PTP v ve [m/s, deg/s]	Geschwindigkeit für PTP Bewegung v - Joint Geschwindigkeit [ve] - Joint Geschwindigkeit am Ziel *Anm.: Befehl obsolet – verwenden Sie stattdessen SPEED_PTP_AX*
ACCEL_CP ax axe [m/s²]	Beschleunigung für lineare Bewegung ax - kartesische Beschleunigung [axe] - kartesische Beschleunigung am Ziel
ACCEL_ORI ax_ori axe_ori [deg/s²]	Orientierungs-Beschleunigung für lineare Bewegung ax_ori - Orientierungs-Beschleunigung axe_ori] - Orientierungs-Beschleunigung am Ziel
ACCEL_PTP aq aq_end [m/s², deg/s²]	Beschleunigung für PTP Bewegung aq - Joint Beschleunigung [aqe] - Joint Beschleunigung am Ziel *Anm.: Befehl obsolet – verwenden Sie stattdessen ACCEL_PTP_AX*
SPEED_PTP_AX v1 .. vn [m/s, deg/s]	Geschwindigkeiten für PTP Bewegung für jede Achse v1 - Joint(1) Geschwindigkeit [vn] – Joint(n) Geschwindigkeit
ACCEL_PTP_AX a1 .. an [m/s², deg/s²]	Beschleunigungen für PTP Bewegung für jede Achse a1 - Joint(1) Beschleunigung [an] - Joint(n) Beschleunigung
SPEED_PTP_OV x [%]	Prozentuale Geschwindigkeit (für PTP Bewegung)der maximalen Achsgeschwindigkeitswerte (für jede Achse) x - Prozentwert
ACCEL_PTP_OV x [%]	Prozentuale Beschleunigung (für PTP Bewegung) der maximalen Beschleunigungswerte (für jede Achse) x - Prozentwert
SPEED_CP_OV x [%]	Prozentuale Geschwindigkeit (für lineare Bewegung) der maximalen Achsgeschwindigkeitswerte (für jede Achse) x - Prozentwert
ACCEL_CP_OV x [%]	Prozentuale Beschleunigung (für lineare Bewegung) der maximalen Beschleunigungswerte (für jede Achse) x - Prozentwert
SPEED_ORI_CP_OV x [%]	Prozentuale Orientierungs-Geschwindigkeit (für lineare Bewegung) der maximalen Achsgeschwindigkeitswerte (für jede Achse) x - Prozentwert
ACCEL_ORI_CP_OV x [%]	Prozentuale Orientierungs-Beschleunigung (für lineare Bewegung) der maximalen Beschleunigungswerte (für jede Achse) x - Prozentwert

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Kommando und Syntax	Beschreibung
CONFIG n	Roboter Konfiguration n - Konfigurationsnummer
TOOL X Y Z A B C [m,deg]	Tooldaten (von Tip zu TCP) XYZ - Position ABC - Orientierung
TOOL tagname	Tooldaten (von Tip zu TCP) tagname - Name des Tag
TOOL DEVICE robname	Setzt den TCP für den aktuellen Roboter auf die Tip-Daten des Roboters ‘robname’. Das Kommando ist nützlich nachdem ein Gerät mit „grab“ gegriffen“ wurde.
TOOL toolname	Setzt das Tool / den TCP auf das Tool mit dem Namen ‚toolname’
TOOL DEVICE tool_dev_name toolname	Setzt den TCP für den aktuellen Roboter auf die Tool-Daten des Tools ‚toolname’ des gewählten Devices ‘tool_dev_name’. Das Kommando ist nützlich nachdem ein Gerät mit „grab“ gegriffen“ wurde.
EXT_TCP X Y Z A B C [m,deg]	Externer TCP XYZ - Position ABC - Orientierung
EXT_TCP tagname	Externer TCP tagname - Name des Tags
BASE X Y Z A B C [m,deg]	Verschiebt die Ziele um das BASE frame Das Ziel von allen BASE-Kommandos ist es Kommandos zu verschieben. Das BASE-Kommando ist immer bezogen auf die Roboterbasis. (siehe auch ERC BASE ...) d.h.: Alle folgenden Bewegungskommandos werden um das aktuelle Base Frame transformiert. XYZ - Position ABC - Orientierung **Anm.**: siehe dazu auch folgende Kommandos ERC BASE BODY bodyname ERC BASE TCP
BASE tagname	Programm BASE tagname - Name des Tags
BASE_REL dX dY dZ dA dB dC [m,deg]	Relative Programm BASE Verschiebt das aktuelle Baseframe um die relativen Werte dXdYdZ - delta Position, dAdBdC - delta Orientierung
BASE_PRG X Y Z A B C [m,deg]	Das BASE_PRG Kommando arbeitet in Bezug auf das aktuelle BASE frame. Das resultierende Bezugssystem für alle Bewegungen bzgl. Roboterbasis berechnet sich aus: T_base_final = T_base_prg * T_base (T - homogene 4x4 matrix) XYZ – Position, ABC - Orientierung
BASE_PRG_REL dX dY dZ dA dB dC [m,deg]	Relative Programm BASE_PRG dXdYdZ - delta Position, dAdBdC - delta Orientierung

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

LEADING_POSITION x	Position ist führend für lineare Bewegung ON - Position hat Priorität, hält die Geschwindigkeit OFF - Orientierung hat Priorität VAR - langsamstes Profil hat Priorität
LEADING_ORIENTATION x	Orientierung ist führend für lineare Bewegung (Gegenteil von LEADING_POSITION) ON - Orientierung hat Priorität OFF - Position hat Priorität, hält die Geschwindigkeit VAR - langsamstes Profil hat Priorität
HOME n	Homeposition n - Zahl der Homeposition
HOME homepositionname	Homeposition homepositionname - Name der Homeposition
HOME $NAME_N	Homeposition $NAME_N Name der Homeposition der mit dem Kommando ERC SET_PARAMETER $NAME_1 'name' definiert wird
SLEW X Y Z A B C [m,deg] [Extax1 Extax2 …]	SLEW (PTP), die Achsen erreichen zu unterschiedlichen Zeiten ihr Ziel XYZ – Position, ABC – Orientierung, Extax – externer Achswert
SLEW_REL dX dY dZ dA dB dC [m,deg]	Relative SLEW dXdYdZ - delta Position, dAdBdC - delta Orientierung
SLEW tagname	SLEW (PTP), die Achsen erreichen zu unterschiedlichen Zeiten ihr Ziel tagname - Name des Tags
SLEW_AX q1 .. qn [m,deg]	Joint spezifisch SLEW q1..qn - Ziel Joint/Axis
SLEW_AX_REL dq1..dqn [m,deg]	Relative joint spezifisch SLEW dq1..dqn - delta Joint/Axis
PTP X Y Z A B C [m,deg] [Extax1 Extax2 …]	Voll-Synchro PTP XYZ – Position, ABC – Orientierung, Extax – externer Achswert
PTP_REL dX dY dZ dA dB dC [m,deg]	Relative Voll-Synchro PTP dXdYdZ - delta Position dAdBdC - delta Orientierung
PTP tagname	Voll-Synchro PTP tagname - Name des Tags
PTP_AX q1 .. qn [m,deg]	Joint spezifisch Voll-Synchro PTP q1..qn - Ziel Joint/Axis
PTP_AX_REL dq1..dqn [m,deg]	Relative Joint spezifisch Voll-Synchro PTP dq1..dqn - delta Joint/Axis
LIN X Y Z A B C [m,deg] [Extax1 Extax2 …]	Linear CP Bewegung XYZ – Position, ABC - Orientierung Extax - external axis values
LIN_REL dX dY dZ dA dB dC [m,deg]	Relative Linear CP Bewegung dXdYdZ - delta Position, dAdBdC - delta Orientierung
LIN TagName	Lineare CP Bewegung Tagname - Name des Tag
LIN_ORI ori_type	Orientierungs- Interpolationstyp für lineare CP Bewegung ori_type - VARIABLE, FIX, TANGENTIAL, AUX, VARIABLE2, QUATERNION

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

MOVE TagNames[] MOVE TagIdx[]	Anfahren eines oder mehrerer Tagpunkte im Tag-Motiontype. Beispiel: Pfad mit 4 Tagpunkten { T1, T2, T3, T4}, T1 hat PTP motype T2 hat LIN motype T3 hat CIRC motype T4 hat VIA motype MOVE T1 T4 T3 T2 fährt zu T1 in ptp, zu T3 in circ via T4 und zu T2 im linear Motiontype. Die Nutzung von Tagindex, MOVE 1 4 3 2 bewirkt dieselbe Bewegung.
Along path TagNameStrt TagNameEnd	Abfahren eines Pfades path : Pfadname TagNameStrt – erster anzufahrender Tag TagNameEnd – letzter anzufahrender Tag TagNameStrt, TagNameEnd werden mit Namen “T_1” oder Index identifiziert Beispiele: along path01 T_1 T_5, fährt von Tag “T_1” zu “T_5” along path01 2 6, fährt vom 2. Tag zum 6. Tag im Pfad “path01” along path01 T_1 -1, fährt von Tag “T_1” bis zum letzten Tag im Pfad **Anm.**: Motion type, Geschwindigkeit, etc. sind abhängig von den Tagpunktattributen wenn ”ERC USE_TAG_ATTRIBUTES ON” gesetzt ist !.
CIRC X Y Z A B C [X2 Y2 Z2] [m,deg]	Circular CP Bewegung XYZ – Position, ABC - Orientierung [X2 Y2 Z2] – Via-Punkt
CIRC_REL dX dY dZ dA dB dC [dX2 dY2 dZ2] [m,deg]	Relative Circular CP Bewegung dXdYdZ - delta Position, dAdBdC - delta Orientierung [dX2 dY2 dZ2] - delta Via-Punktposition
CIRC tagname [TagName2]	Circular CP Bewegung tagname - Name des Ziel-Tagpunktes [tagname2] - Name des Via-Tagpunktes
CIRC_ORI ori_type	Orientierungs-Interpolationtyp für Circular CP Bewegung ori_type - VARIABLE, FIX, TANGENTIAL, AUX, VARIABLE2, QUATERNION
CIRC_ORI_QUAT_IPO ori_quat_type START_VIA_END [START_END, START_VIAORI_END, TANGENTIAL, FIX]	Orientierungs-Interpolationart für Circular CP Bewegung wenn CIRC_ORI = QUATERNION gesetzt START_VIA_END – Die Ausrichtung der X-Richtung des Via-Punktes bestimmt die Interpolation der Orientierung START_END – Die Interpolation der Orientierung wird nur durch die Orientierung im Start und Ziel bestimmt. START_VIAORI_END – Während CIRC-Interpolation wird auch die Orientierung im Via-Punkt erreicht. TANGENTIAL – Die X-Achse wird tangential am Kreisbogen geführt FIX – Die Orientierung bleibt unverändert, wobei die Orientierung im Ziel ignoriert wird

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

VIA_POS X Y Z A B C [m,deg]	Via Position für Circular CP Bewegung XYZ - Position, ABC - Orientierung
VIA_POS_REL dX dY dZ dA dB dC [m,deg]	Relative Via Position für Circular CP Bewegung dXdYdZ - delta Position, dAdBdC - delta Orientierung
VIA_POS tagname	Via Position für Circular CP Bewegung tagname - Name des Via-Tagpunktes
JUMP_TO X Y Z A B C [m,deg]	Springt zur Zielposition XYZ – Position, ABC - Orientierung
JUMP_TO tagname	Springt zur Zielposition tagname - Name des Ziel-Tagpunktes
JUMP_TO_AX q1 .. qn [m,deg]	Springt zur Ziel-Achswinkelstellung q1..qn - Ziel Joint/Axis
WAIT x [sec]	Wait Befehl x – Zeit in Sekunden
ZONE	Überschleifparameter. Das Kommando „Zone = 0“ ermöglicht das exakte Anfahren einer Zielstellung.
AUTO_ACCEL ON [OFF/AX/POS/ORI]	Automatische Berechnung der Beschleunigung in Abhängigkeit der programmierten Geschwindigkeit. AX – Berechnung für PTP Bewegungen POS – Berechnung für CP Bewegungen für Position ORI – Berechnung für CP Bewegungen für Orientierung ON – Berechnung für PTP, POS und ORI OFF – Berechnung deaktiviert
ACCSET Acc Ramp	Verzögerung der Beschleunigung. Die Argument Acc und Ramp werden prozentual im Bereich von 20% - 100% angegeben. Acc – Beschleunigung und Verzögerung als Prozentwert der normalen Werte Ramp – Anstiegswert von Beschleunigung und Verzögerung als Prozentwert der normalen Werte
PTP_CALC_MODE SHORTEST_ANGLE [TURN, MATH, IN_TRAVEL_RANGE, VAR_CONFIG]	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 VAR_CONFIG – variable Konfiguration. Die Berechnung kleinster Achswertänderungen zwischen Start und Ziel, kann zu einer neuen Roboterkonfiguration führen
TURN Turn_Ax1 ... Turn_Axn	Vorgabe der TURN-Werte für jede Achse Ax1…Axn, für die folgende PTP oder SLEW Ziel-Position, wenn Turn-Intervalle definiert sind
MSG Text	“Text“ wird im Program Window angezeigt
NATIVE command	Der native Befehl kann direkt in das Post-Processor API verwendet werden. Es kann keinen Einfluss auf die Simulation

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

IF, While und Goto Anweisungen

Kommando und Syntax	Beschreibung

IF *Bedingung*
 	
! ERPL,ERCL hier einfügen
 	
ELSEIF *Bedingung2*
 	
! ERPL,ERCL hier einfügen
 	
ELSEIF *Bedingung3*
 	
! ERPL,ERCL hier einfügen
 	
ELSE
 	
! ERPL,ERCL hier einfügen
 	
ENDIF
	IF Statement (prüft Bedingung auf „True“ und „False“)
	
	Die Bedingung ist ein mathematischer Ausdruck und wird auf “True” und “False” geprüft. Die Bedingung ist wahr, wenn größer Null.
	
	Beispiel: xx > yy, somit wird der Roboter zu Tag ‘T_1’ fahren
	xx=3; yy=1
	IF gt(xx,yy)
	LIN T_1
	ELSEIF lt(xx,yy)
	LIN T_2
	ELSEIF eq(xx,yy)
	LIN T_3
	ELSE
	! ERPL,ERCL hier einfügen
	ENDIF

WHILE *Bedingung* *! ERPL,ERCL hier einfügen* ENDWHILE	While-loop (prüft Bedingung auf „True“ und „False“) Die Bedingung ist ein mathematischer Ausdruck und wird auf “True” und “False” geprüft. Die Bedingung ist wahr, wenn größer Null. Beispiel: Der Roboter fährt den Pfad ‘path01’ dreimal entlang. xx=4; yy=1 WHILE gt(xx,yy) xx = xx-1 ALONG PATH01 T_1 T_5 ENDWHILE
GOTO LABEL *label_name* LABEL *label_name*	GOTO Statement (springt zu anderer Programmzeile) Beispiel: GOTO LABEL my_label … *! dieser Bereich wird übersprungen* … LABEL my_label Tipp: - nutze „labels“ nur wenn unbedingt nötig - ein „label“ kann überall stehen, aber nur einmal im Programm existieren - nutze „labels“ niemals um aus einem if-else-endif oder einer while-endwhile Schleife auszusteigen

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Kommandos für I/O Signale

Kommando und Syntax	Beschreibung
WAIT_UNTIL_SIGNAL_SET *my_signal*	Wartet solange, bis das entsprechende Signal gesetzt wurde Das Kommando prüft eine Bedingung auf „True“ und „False“. Dabei ist die Bedingung ein mathematischer Ausdruck und ist wahr, wenn größer 0.5 WAIT_UNTIL_SIGNAL_SET my_signal ! macht weiter wenn das Signal „my_signal“ gesetzt wurde ! wartet wenn das Signal „my_signal“ nicht gesetzt wurde
WAIT_UNTIL_SIGNAL_UNSET *my_signal*	Wartet solange, bis das entsprechende Signal **nicht** mehr gesetzt ist Das Kommando prüft eine Bedingung auf „True“ und „False“. Dabei ist die Bedingung ein mathematischer Ausdruck und ist wahr, wenn größer 0.5 WAIT_UNTIL_SIGNAL_SET my_signal ! macht weiter wenn das Signal „my_signal“ nicht mehr gesetzt ist ! wartet wenn das Signal „my_signal“ gesetzt ist
WAIT_FOR_CONDITION *condition*	Prüft ob eine angegebene Bedingung zutrifft WAIT_FOR_CONDITION gt(my_signal ,0) ! macht weiter wenn die Bedingung wahr ist ! wartet wenn die Bedingung falsch ist

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Mathematische Parser Konstanten

Kommando und Syntax	Beschreibung	Beispiel
PI	Kreiszahl 3.1415926	U = 2*pi*radius
e	Exponentialwert 2.718282	e = exp(1)
RAD	Konvertiert Grad → Radiant	RAD = PI / 180° = 0.017453
DEG	Konvertiert Radiant → Grad	DEG = 180° / PI = 57.295778
m2mm	Konvertiert m → mm	m2mm = 1000
mm2m	Konvertiert mm → m	m2mm = 0.001
m2inch	Konvertiert m → inch	m2inch = 39.37
inch2m	Konvertiert inch → m	inch2m = 0.0254
mm2inch	Konvertiert mm → inch	mm2inch = 0.03937
inch2mm	Konvertiert inch → mm	inch2mm = 25.4
mps2mmpmin	Konvertiert mps → mmpmin	mps2mmpmin = 60000
mmpmin2mps	Konvertiert mmpmin → mps	mmpmin2mps = 1/60000
TRUE	1	WHILE TRUE ENDWHILE
FALSE	0	done = FALSE
unit2m	Konvertiert userunit → m	scale = unit2m
m2unit	Konvertiert m → userunit	scale = m2unit
ans	Answer, letztes Resultat	
DIM	3	
DOF6	6	

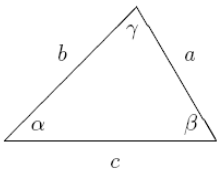
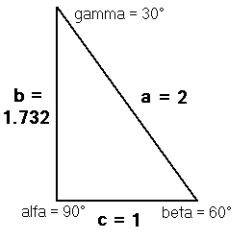
EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Mathematische Parser Funktionen

Kommando und Syntax	Beschreibung	Beispiel
abs (x)	Absoluter Wert	abs (-6) = 6.000000
asin (x)*DEG	Arcussinus	asin (0.5)*DEG = 30
acos (x)*DEG	Arcuskosinus	acos (0.5)*DEG = 60
atan (x)*DEG	Arcustangens	atan (1)*DEG = 45
atan2 (y,x)*DEG	Arcuskotangens	atan2 (2,-1)*DEG = -116.56501
cos (x*RAD)	Cosinus	cos (45*RAD) = 0.707107
cosh (x)	Cosinus hyperbolicus	cosh (0) = 1.000000
ceil (x)	Obergrenze	ceil (2.83) = 3.000000
exp (x)	Exponential	exp (0) = 1.000000
eq (x,y)	Gleich	eq (5,5) = 1.000000
fact (x)	Fakultät	fact (6) = 720.000000
floor (x)	Untergrenze	floor (2.83) = 2.000000
gt (x,y)	Größer als	gt (5,3) = 1.000000
ge (x,y)	Größer gleich	ge (5,5) = 1.000000
int (x)	Integer	int (2.83) = 2.000000
ln (x)	Natürlicher Logarithmus	ln (2) = 0.693147
Log (x)	Logarithmus	log (2) = 0.301030
lt (x,y)	Kleiner als	lt (3,5) = 1.000000
le (x,y)	Kleiner gleich	le (5,5) = 1.000000
min(1,2)	Minimum zweier Argumente	min(1,2) = 1.000000
max(1,2)	Maximum zweier Argumente	max(1,2) = 2.000000
ne (x,y)	Ungleich	ne (3,5) = 1.000000
pow (x,y)	zum Quadrat	pow (2,3) = 8.000000
rnd (x)	Zufall	Rnd (10)-5 = 3.772546
Sqrt (x)	Wurzel	sqrt (2) = 1.414214
sin (x*RAD)	Sinus	sin (45*RAD) = 0.707107
sinh (x)	Sinus-hyperbolicus	sinh (1) = 1.175201
sign (x)	Vorzeichenumkehrung	sign (-2.83) = -1.000000
tan (x*RAD)	Tangens	tan (45*RAD) = 1.000000
tanh (x)	Tangens-hyperbolicus	tanh (1) = 0.761594
trunk (x)	Kürzen	Trunk (-2.83) = -2.000000

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Dreiecks Parser Funktionen

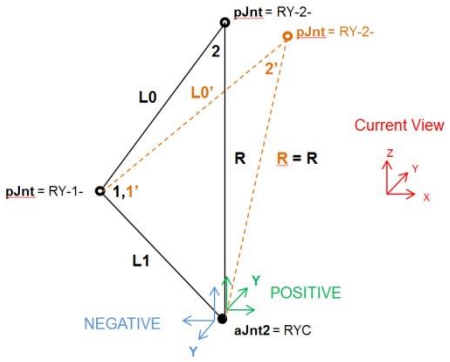
Kommando und Syntax	Beschreibung	Beispiel
beta = tr_assa (alfa, c, a) angle side side > angle	Dreiecksberechnung:  In: alfa, c, a Out: beta	 tr_assa (90*RAD, 1, 2) * DEG = 60° = beta
gamma = tr_assa2 (alfa, c, a) angle side side > angle2	In: alfa, c, a Out: gamma	tr_assa2 (90*RAD, 1, 2) * DEG = 30°= gamma
b = tr_asss (alfa, c, a) angle side side > side	In: alfa, c, a Out: b	tr_asss (90*RAD, 1, 2) = 1.732051 = b = sqrt (2^2-1^2)
b = tr_sass (c, beta, a) side angle side > side	In: c, beta, a Out: b	tr_sass (1, 60*RAD, 2) = 1.732051 = b = sqrt (2^2-1^2)
gamma = tr_sasa (c, beta, a) side angle side > angle	In: c, beta, a Out: gamma	tr_sasa (1, 60*RAD, 2)*DEG = 30° = gamma
alfa = tr_sssa (a, b, c) side side side > angle	In: a, b, c Out: alfa	tr_sssa (2,sqrt(3),1)*DEG = 90° = alfa
gamma = tr_sasssa (d, beta, a, b, c) side angle side side side > angle	4 – Winkelberechnung In: d, beta, a, b, c Out: gamma beta is between <d,a> gamma is between <a,b>	tr_sasssa (1,90*RAD,2,1,3) * DEG = 158.6955° = gamma
delta = tr_sasssa2 (d, beta, a, b, c) side angle side side side > angle	In: d, beta, a, b, c Out: delta beta is between <d,a> delta is between <b,c>	tr_sasssa2 (1,90*RAD,2,1,3)*DEG = 33.55731° = delta

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Roboterbezogene Parser Funktionen

Kommando und Syntax	Beschreibung	Beispiel
num_dof()	Anzahl der aktiven Achsen	num_dof() = 6
num_pdof()	Anzahl der passiven Achsen	num_pdof() = 1
dof(aJnt_number)	Achswert der Roboterachse mit der Nummer 'aJnt_number' [1..ndof] in Radiant [rad] oder Meter [m]	dof(1) * DEG = -0.132455
pdof(pJnt_number)	Achswert der passiven Roboterachse mit der Nummer 'pJnt_number' [1..npdof] in Radiant oder Meter	pdof(1) * DEG = -0.132455
dofoffsign(aJnt_number)	Beachte Joint offset und sign = dof(jn) * jntsign(jn) - jntoff(jn)	dofoffsign(1) = -0.132455
jntoff(aJnt _number)	Joint offset der Roboterachse	jntoff(2) = 0.174533
jntsign(aJnt _number)	Vorzeichen der Roboterachse	jntsign(1) = 1.000000
mtounit()	Meter-zu-Unit	mtounit() = 1000.000000
robb(direction)	Robotbase bezogen auf Referenzsystem in [m,rad] direction = [1..6] mit 1=x, 5=Ry	robb(1)*m2mm = 0.000000 robb(5)*deg = 45. 000000
swen(aJnt)	Negative Verfahrbereichsgrenze [rad,m]	swen(1) = -3.141593
swep(aJnt)	Positive Verfahrbereichsgrenze [rad,m]	swep(1) = 3.141593
swen_calc(aJnt)	Negative berechnete Verfahrbereichsgrenze [rad,m]	swen_calc(1) = -3.141593
swep_calc(aJnt)	Positive berechnete Verfahrbereichsgrenze [rad,m]	swep_calc(1) = 3.141593
tcp (direction)	TCP bezogen auf Robotbase in [m,rad] direction = [1..6] mit 1=x, 5=Ry	tcp(1)*m2mm = 1798.858523 tcp(5)* DEG = 95.000000
tcpi (direction)	TCP bezogen auf Welt 'i' in [m,rad] direction = [1..6] mit 1=x, 5=Ry	tcpi(1)*m2mm = 1798.858523 tcpi(5)* DEG = 95.000000
ctool (direction)	current Tool data [m,rad] direction = [1..6] mit 1=x, 5=Ry	ctool(3)*m2mm = 150.000000 ctool(5)* DEG = 30.000000
tool(tool_number,direction)	Tool data [m,rad] tool_number – number of tool direction = [1..6] mit 1=x, 5=Ry	tool(1,3)*m2mm = 150.000000 tool(1,5)* DEG = 30.000000
la (aJnt , direction)	Transformation to next active Joint aJnt - Nummer aktive Achse [0..ndof] direction = [1..6] mit 1=x, 5=Ry Hinweis: aJnt = 0 → Rbase to 1st Achse	la(0,1)*m2mm = 0.000000 la(1,1)*m2mm = 0.000000 la(6,5)* DEG = 90.000000

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Kommando und Syntax	Beschreibung	Beispiel
lp (pJnt , direction)	Transformation to next passive Joint pJnt - Nummer passiven Achse [1..npdof] direction = [1..6] mit 1=x, 5=Ry	lp(1,1)*m2mm = 0.000000 lp(1,5)* DEG = 0.000000
lp0 (pJnt , direction)	Transformation from last passive Joint pJnt - Nummer passiven Achse [1..npdof] direction = [1..6] mit 1=x, 5=Ry	lp0(1,1)*m2mm = 0.000000 lp0(1,5)* DEG = 0.000000
JNTDAMP_TRI(L0, L1, R, position, aJ2_ori, pjnt_angle)	Winkeländerung der pJnt-Achsen in Abhängigkeit vom aJnt2 [rad]  Hinweis: Nullstellung bzw. Ausgangs-position betrachten L0 = Abstand pJnt zu pJnt in [mm] L1 = Abstand aJnt2 RYC zu pJnt RY-1- in [mm] R = Abstand aJnt2 RYC zu pJnt RY-2- in [mm] position – Lage vom pJnt RY-2- zu aJnt2 RYC [DAMP_BELOW, DAMP_ABOVE] mit DAMP_BELOW = 0: unter aJnt2 mit DAMP_ABOVE = 1: über aJnt2 aJ2_ori – Orientierung der y-Achse (Drehachse) des aJnt2 RYC bei Betrachtung d. Ausgangslage (Current View) [POSITIVE, NEGATIVE] mit POSITIVE = 1: pos. Richtung (weg) mit NEGATIVE = -1: neg. Richtung (hin) pjnt_angle = [1..2] mit 1 = Funktion gibt Winkeländerung für den Winkel zwischen L0 und L1 aus mit 2 = Funktion gibt Winkeländerung für den Winkel zwischen L0 und R aus	L0=400.000 L1=300.000 R=500.000 **Example 1:** dof(2)*DEG=0.000° JNTDAMP_TRI(L0,L1,R, DAMP_ABOVE,POSITIVE,1)*DEG = 0.000° JNTDAMP_TRI(L0,L1,R, DAMP_ABOVE,POSITIVE,2)*DEG = 0.000° **Example 2:** dof(2)*DEG=10.000° JNTDAMP_TRI(L0,L1,R, DAMP_ABOVE,POSITIVE,1)*DEG = 9.42° JNTDAMP_TRI(L0,L1,R, DAMP_ABOVE,POSITIVE,2)*DEG = - 0.58°

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Kommando und Syntax	Beschreibung	Beispiel
unittom()	Unit-zu-Meter	unittom() = 0.001000
collision()	Rückgabewert=1 wenn Kollision gefunden, sonst 0	coll = collision()
collision_devices_idx (dev_idx1, dev_idx2)	Prüft ob das 1. Gerät mit dem Geräte-Idx dev_idx1 mit dem 2. Gerät dev_idx2 kollidiert. Die Parameter dev_idx1 und dev_idx2 können flexibel gesetzt werden, so dass z.B. das 1. Gerät gegen alle anderen Geräte dev_idx2 = 0 in der Arbeitszelle auf Kollision geprüft wird Rückgabewert=1 wenn Kollision gefunden, sonst 0	coll = collision_devices_idx(1,2) coll = collision_devices_idx(3,2) coll = collision_devices_idx(1,0) coll = collision_devices_idx(0,2)
collision_devices_name ("DeviceName1", "DeviceName2")	Prüft ob das 1. Gerät mit dem Gerätenamen "DeviceName1" mit dem 2. Gerät "DeviceName2" kollidiert. Die Parameter "DeviceName1" und "DeviceName2" können flexibel gesetzt werden, so dass z.B. das 1. Gerät gegen alle anderen Geräte "DeviceName2" = "0" in der Arbeitszelle auf Kollision geprüft wird. Rückgabewert=1 wenn Kollision gefunden, sonst 0	coll = collision_devices_name ("dev1","dev2") coll = collision_devices_name (dev1,dev2) coll = collision_devices_name ("dev1","0") coll = collision_devices_name (dev2,0)
get_device_idx ("DeviceName")	Gibt den Geräte-Idx im Bereich [1 bis n=Geräteanzahl] des Geräts mit dem Namen "DeviceName" zurück. Der Parameter "DeviceName" kann flexibel gewählt werden, so dass z.B. "" oder () den Geräte-Idx des aktuellen Geräts zurückgibt. Bei Fehler wird 0 zurückgegeben.	dev_idx = get_devices_idx ("dev1") dev_idx = get_devices_idx (dev1) cdev_idx = get_devices_idx ("") cdev_idx = get_devices_idx ()
sim_time()	Globale Simulationszeit in sec	simtime = sim_time()
sim_realtime()	Reale Simulationszeit in sec	simrealtime = sim_realtime()
sim_step()	Simulationsschrittweite in sec	simstep = sim_step()

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™

ERCL - EASY-ROB™ Kommando Sprache

ERCL ist eine Erweiterung zu ERPL, um nahezu alle Benutzerinteraktionen innerhalb eines Roboterprogramms zu automatisieren. Beispiele dafür sind:
TCP-Trace ON/OFF, Kollisionsprüfung ON, laden von Views, Rendering ändern zu flat, wire oder invisible, Farben ändern, ändern der Simulation Stepsize, Körper verschieben, Verschieben der Roboterbase, etc..
All diese Kommandos unterstützen Sie beim Erzeugen von fortschrittlichen und effektiven Simulationen.

Kommando und Syntax	**Beschreibung**
ERC SET_DEFAULTS	Setzen der Defaultwerte: Visualisiert Roboter, Werkzeug und Umgebung Schaltet Roboterkoordinatensysteme ab.
ERC SIM_STEP x [sec]	Setzen der Simulation Stepsize x - Simulationsabtastzeit **Achtung**: Dieses Kommando schaltet automatisch „Realtime OFF“
ERC CNTRL_STEP x [sec]	Setzen der Abtastrate des Positionsreglers x - Reglerabtastzeit
ERC SYSTEM_STEP x [sec]	Setzen der Abtastrate des dynamischen Robotermodells x - Modellabtastzeit
ERC IPO_STEP x [sec]	Setzen der Abtastrate des Interpolators x - Interpolationsabtastzeit
ERC IPO_LEAD_TIME x [sec]	Set IPO Lead time, die Bewegung startet nach dieser Zeit. x - ipo lead time
ERC IPO_LAG_TIME x [sec]	Set IPO Lag time, am Ende der Bewegung; der Roboter wir am Ende der Bewegung für die Zeit warten bevor er zur nächsten Position fährt. x - ipo lag time

ERCL - ON / OFF Kommandos

Kommando und Syntax	**Beschreibung**
ERC TRACK ON,OFF	Ein- / Ausschalten des TCP-Trace
ERC DYNAMICS ON,OFF	Ein- / Ausschalten der Dynamik-Option
ERC STOP_SWE ON,OFF	Ein- / Ausschalten der Überwachung der Softwareendschalter
ERC STOP_SPEED ON,OFF	Ein- / Ausschalten der Überwachung der Achsgeschwindigkeit
ERC STOP_ACCEL ON,OFF	Ein- / Ausschalten der Überwachung der Achsbeschleunigung
ERC STOP_CART_SPACE ON/OFF	Ein- / Ausschalten der Überwachung der kartesischen Raumbegrenzung.
ERC COLLISION ON,OFF	Ein- / Ausschalten der Kollisionsüberwachung
ERC STOP_COLLISION ON,OFF	Ein- / Ausschalten von „STOP bei Kollision“
ERC RED_BLUE_ COLLISION ON,OFF	Ein- / Ausschalten der Rot/Blau Kollision

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERC ROBOT_JOINTS * ERC ROBOT_POSITION * ERC ROBOT_MOTIONDATA * ERC ROBOT_PARSERVARS * * = ON,OFF	Ein- / Ausschalten der Roboter Online Output Data ROBOT_JOINTS: Soll-Achswerte des Roboters ROBOT_POSITION kartesische Soll-Position des Roboters ROBOT_MOTIONDATA Motion Data des Roboters ROBOT_PARSERVARS Parser Vars des Roboters
ERC FLOOR ON,OFF	Ein- / Ausschalten des Fußbodens
ERC FLOOR_RENDER ON,OFF	Ein- / Ausschalten der Schattierung des Fußbodens (Voll oder Draht)
ERC EXT_TCP ON,OFF	Ein- / Ausschalten der werkstückführenden Bewegung, externer TCP
ERC ORTHOGRAFIC ON,OFF	Ein- / Ausschalten des Orthographischen Ansicht
ERC DISPLAY DEVICE 'device_name' ON,OFF	Ein- / Ausschalten der Visualisierung des Geräts mit dem Namen 'DeviceName'
ERC DISPLAY_ROBOTS ON,OFF	Ein- / Ausschalten Visualisierung der Geometrien in allen Roboter/Device-Gruppen
ERC DISPLAY_ROBOT_COORSYS ON,OFF	Ein- / Ausschalten Visualisierung der gelb/grün kolorierten Roboter Joint/Axis Coorsys
ERC DISPLAY_TOOLS ON,OFF	Ein- / Ausschalten Visualisierung der Geometrien in allen Toolgruppen
ERC DISPLAY_BODIES ON,OFF	Ein- / Ausschalten Visualisierung aller Geometrien der Body / Umgebungsgruppe
ERC TCP_COORSYS ON,OFF	Ein- / Ausschalten Visualisierung von blau kolorierten Tool/TCP Coorsys
ERC IPO_COORSYS ON,OFF	Ein- / Ausschalten Visualisierung von rot kolorierten IPO Coorsys
ERC BASE_COORSYS ON/OFF	Ein- / Ausschalten Visualisierung von grün kolorierten Base Coorsys
ERC CREATE_TARGET_TAGS ON/OFF	Ein- / Ausschalten automatisches Erzeugen von Tags am Zielpunkt
ERC RESET_ALL_POSITIONS_JOINTS ON/OFF	Ein- / Ausschalten des Rücksetzens aller Positionen und Roboterjoints
ERC NO_DECEL ON/OFF	Ein- / Ausschalten der Geschwindigkeitsabnahme (Abbremsen) am Zielpunkt NO_DECEL=ON : setzt ZONE=0.1 wenn ZONE<> 0 NO_DECEL=OFF : setzt ZONE=0
ERC GRAFIC_UPDATE ON/OFF	Ein- / Ausschalten des grafischen Updates während des Programmablaufs. Das Kommando ist hilfreich um Hintergrundkommandos zu verstecken
ERC DISPLAY_TAGS ON/OFF	Ein- / Ausschalten Visualisierung aller Pfade und Tagpunkte
ERC VIEW_CHOREOGRAPHY ON,OFF	Ein- / Ausschalten der View Choreography, so werden “ERC LOAD View …”- Kommandos ignoriert
ERC DISPLAY_ROBOT_COORSYS ON, OFF	Ein- / Ausschalten Visualisierung von Roboterjointcoordinatensystemen für alle aktiven und passiven Joints
ERC DISPLAY_ROBOT_NAME ON, OFF	Ein- / Ausschalten Visualisierung des Roboternamens

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERC SHOW_CART_SPACE ON/OFF	Ein- / Ausschalten Visualisierung der kartesischen Raumbegrenzung
ERC STATUS_OUTPUT ON/OFF [1-at simstep,2-at target pose] [flnname] [fct# 0-12]	Ein- / Ausschalten des Statusoutput während des Programmablaufs. Diese Funktion ermöglicht das Speichern des kompletten Simulationsstatus in einem definierten Format. Typische Werte sind Joint- /Achsdaten und kartesische TCP-positionen. Parameter: 1st : 1 – speichert die Statusdaten bei jedem Simstep 2 - speichert die Statusdaten an jedem Zielpunkt 2nd : Dateiname, z.B. "out.dat" 3rd : Funktionnummer -1 - Defaultoutput in EASY-ROB™ für Matlab-Visualisierung 0 - Defaultoutput in EASY-ROB™ 1..12 – Benutzerdefinierter Output, siehe Beispiel API-DYN status_output_user_1() definiert in Datei *dyn_user.cpp* **Beispiel**: im Ordner ./proj/proj/ Status_Output.cel und Status_Output.prg
ERC INTERPOLATION ON/OFF	Ein- / Ausschalten lineare Interpolation, Roboter springt zum Zielpunkt
ERC REALTIME_SIM ON/OFF	Echtzeitsimulation – lässt das Programm in Echtzeit laufen. Anmerkung: „Realtime ON“ verändert die SIM_STEPSIZE“
ERC BACKFACES ON/OFF	Ein- / Ausschalten Visualisierung von „backfaces“ (Flächenrückseiten)
ERC CAMERA ON/OFF	Ein- / Ausschalten der Kamera.
ERC USE_TAG_ATTRIBUTES ON/OFF	Ein- / Ausschalten der Nutzung von Tagattributen, wie „motion type“, Geschwindigkeit, Beschleunigung, etc.
ERC MSG_WIN ON/OFF	Ein- / Ausschalten des Message-Fensters
ERC PRG_WIN ON/OFF	Ein- / Ausschalten des Programm-Fensters
ERC DISPLAY_CROBOT ON,OFF	Ein- / Ausschalten Visualisierung des aktuellen Roboters
ERC DISPLAY_CTOOL ON/OFF	Ein- / Ausschalten Visualisierung des aktuellen Tools
ERC COLLISION_CROBOT ON/OFF	Ein- / Ausschalten der Kollisionsprüfung des aktuellen Roboters
ERC COLLISION_CTOOL ON/OFF	Ein- / Ausschalten der Kollisionsprüfung des aktuellen Tools
ERC COLLISION_CROBOT_REF ON/OFF	Ein- / Ausschalten der Referenz-Kollisionsprüfung des aktuellen Roboters
ERC WORLD_COORSYS ON/OFF	Ein- / Ausschalten des Weltkoordinatensystems
ERC ALL_COORSYS ON/OFF	Ein- / Ausschalten aller Koordinatensystem in der Arbeitswelt
ERC HISTORY_DEVICE ON/OFF	Ein- / Ausschalten der Aufzeichnung von Daten (für ein Device) für das History-Diagramm
ERC HISTORY_DEVICE ALL_ON / ALL_OFF	Ein- / Ausschalten der Aufzeichnung von Daten (für alle Devices) für das History-Diagramm
ERC HISTORY_OUTPUT ON / ALL_ON [flnname]	Start der History-Diagramm Aufzeichnung flnname - Name der Datei in der die Daten gespeichert werden
ERC HISTORY_OUTPUT OFF	Beenden der History-Diagramm Aufzeichnung
ERC CELL_INFO_SHOW ON/OFF	Ein- und Ausschalten der Zellen-Informationszeile

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - Post Processor Kommandos

Kommando und Syntax	Beschreibung
ERC POST_PROCESS KEY filename	Startet den Post Processor. Parameter: KEY Language Key, definiert in er_post.dll z. B. ABB, KUKA, FANUC_RJ oder COMAU_C5G filename Name der erzeugten Roboterprogrammdatei
ERC POST_PROCESS OFF	Post Processing beendet

ERCL - Render Kommandos

Kommando und Syntax	Beschreibung
ERC RENDER SMOOTH ERC RENDER FLAT	Darstellung der gesamten Szene in FLAT oder SMOOTH (vollschattiert), Icon 
ERC RENDER WIRE	Darstellung der gesamten Szene in WIRE frame (Drahtmodell), Icon 
ERC RENDER POINT	Darstellung der gesamten Szene in POINTS (Punktdarstellung), Icon 
ERC RENDER BBOX ON,OFF	Darstellung der gesamten Szene als Bbox (bounding boxes), Icon 
ERC RENDER group bodyname render	Setzen / modifizieren des Renderings eines einzelnen Parts group - BODY, ROBOT, TOOL bodyname - Name des einzelnen Parts render - WIRE, FLAT, BBOXWIRE, BBOXFLAT, INVISIBLE, POINT, SMOOTH **Beispiel**: erc render ROBOT 'body_name' WIRE erc render ROBOT 'body_name' FLAT
ERC RENDER group render	Setzen / modifizieren des Renderings von allen Parts in einer Gruppe group - BODY_GRP, ROBOT_GRP, TOOL_GRP render - WIRE, FLAT, BBOXWIRE, BBOXFLAT, INVISIBLE, POINT, SMOOTH **Beispiel**: erc render ROBOT_GRP WIRE erc render ROBOT_GRP FLAT
ERC DISPLAY DEVICE 'device_name' ON,OFF	Ein- / Ausschalten der Visualisierung des Geräts mit dem Namen 'DeviceName'

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

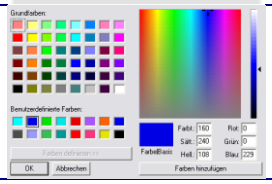
ERCL - Kamera Kommandos

Kommando und Syntax	**Beschreibung**
ERC CAMERA ON/OFF	Ein- / Ausschalten der Kamera.
ERC CAMERA FOCUS focus [mm]	Setzt den Kamerafocus focus – Focuslänge Bereich [12.25 mm bis 240 mm], Default. 50 mm (mittlere Maus)
ERC CAMERA Z_OFFSET z_offset [mm]	Setzt Kamera Offset in Z -Richtung z_offset – Z - Offset Bereich [-200 mm bis +500 mm], Default. 0 mm (linke Maus) - Wert zoomt heraus + Wert zoomt hinein
ERC CAMERA BODY [ROBOT,TOOL] bodyname	Befestigt die Kamera am Ursprung der CAD-Geometrie
ERC CAMERA POSITION XYZ ABC [m,deg]	Offsetposition bezogen auf den Ursprung der CAD-Geometrie an die die Kamera befestigt ist Anm.: Standardmäßig zeigt/sieht die Kamera in Z-Richtung (siehe auch OpenGL™) Eine 180°-drehung um die Y-Achse dreht die Kamera in den meisten Fällen in die richtige Richtung.

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - Farb- Kommandos

Vordefinierte Farbwerte

	1 BLUE 2 GREEN 3 CYAN 4 RED 5 MAGENTA 6 BROWN 7 LIGHTGRAY 8 DARKGRAY 9 LIGHTBLUE 10 LIGHTGREEN 11 LIGHTCYAN 12 LIGHTRED 13 LIGHTMAGENTA 14 YELLOW 15 WHITE
	32 Bit RGB Color

Kommando und Syntax	**Beschreibung**
ERC COLOR group name color	Setzt / modifiziert die Farbe für ein einzelnes Part bzw. ein Body group - BODY, ROBOT, TOOL name - Name des Body color vordefinierte Farbe **Beispiel**: erc color TOOL 'tool' RED
ERC COLOR group color	Setzt / modifiziert die Farbe für alle Parts in einer Gruppe group BODY_GRP, ROBOT_GRP, TOOL_GRP color vordefinierte Farbe **Beispiel**: erc color TOOL_GRP RED
ERC COLOR track color	Setzt / modifiziert die Farbe des Roboter TCP trace track TRACK, TRACK_DYN color vordefinierte Farbe **Beispiel**: erc color TRACK RED erc color TRACK -1, für wechselnde Farbe
ERC COLOR tag color	Setzt die vordefinierte TAG Farbe color vordefinierte Farbe **Anm.**: Beim Erzeugen eines neuen Tags wird dieser die gewählte Farbe haben

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - Transparenz- Kommandos

Kommando und Syntax	Beschreibung
ERC TRANSPARENCY group name alpha	Setzt / modifiziert die Transparenz für ein einzelnes Part bzw. ein Body group - BODY, ROBOT, TOOL name - Name des Body alpha alpha blend wert [0,1] **Beispiel**: erc transparency TOOL 'tool' 0.5
ERC TRANSPARENCY group alpha	Setzt / modifiziert die Transparenz für alle Parts in einer Gruppe group BODY_GRP, ROBOT_GRP, TOOL_GRP alpha alpha blend wert [0,1] **Beispiel**: erc transparency TOOL_GRP 0.5

ERCL - Reset und Save Kommandos

Kommando und Syntax	Beschreibung
ERC RESET JOINTPOSITION	Setzt die Joint-/Achsposition zurück auf Anfangzustand
ERC SAVE JOINTPOSITION	Speichert die Joint-/Achsposition als Startzustand

ERCL - Load Kommandos

Mit dem Load Kommando können Sie während des Programmablaufs weitere Dateien wie z.B. ein Tool oder eine andere Ansicht (View) laden.

Kommando und Syntax	Beschreibung
ERC LOAD TOOL filename	Lädt eine Tooldatei (*.tol)
ERC LOAD VIEW filename	Lädt eine Viewdatei (*.vie) **Anm.**: Die neue Ansicht wird mit der Anzahl „n“ der View Steps angefahren 'view steps' ERC VIEW steps n
ERC LOAD ROBOT filename	Lädt eine Roboterdatei (*.rob)
ERC LOAD BODY filename	Lädt eine Bodydatei (*.bod)
ERC LOAD TAGS filename	Lädt ein Tagpunktdatei (*.tag)
ERC LOAD MIMIC filename	Lädt eine Mimicdatei (*.mmc) ! Mimic = Machine Interface file
ERC LOAD CAMERA filename	Lädt eine Kameradatei (*.cam)
ERC LOAD ENVIRONMENT [filename]	Lädt eine Environment-datei (*.env)

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - Move Kommandos

Kommando und Syntax	Beschreibung
ERC MOVE BODY bodyname XYZ ABC [m,deg]	Verschiebt ein Part der BODY-Gruppe an eine neue Position bodyname - Name des Body XYZ - Position. ABC - Orientierung
ERC MOVE BODY bodyname TagName	Verschiebt ein Part der BODY-Gruppe an die Tagpunktposition bodyname - Name des Body TagName - Name des Tag
ERC MOVE TOOL bodyname XYZ ABC [m,deg]	Verschiebt ein Part der TOOL-Gruppe an eine neue Position bodyname - Name des Body XYZ - Position ABC - Orientierung
ERC MOVE TOOL bodyname TagName	Verschiebt ein Part der TOOL-Gruppe an die Tagpunktposition bodyname - Name des Body TagName - Name des Tag
ERC MOVE ROBOT bodyname XYZ ABC [m,deg]	Verschiebt ein Part der ROBOTER-Gruppe an eine neue Position bodyname - Name des Body XYZ - Position ABC - Orientierung
ERC MOVE ROBOT bodyname TagName	Verschiebt ein Part der ROBOTER-Gruppe an die Tagpunktposition bodyname - Name des Body TagName - Name des Tag
ERC MOVE_REL BODY bodyname dXdYdZ dAdBdC [m,deg]	Relatives Verschieben eines Parts aus der BODY-Gruppe bodyname - Name des Body dXdYdZ - delta Position. dAdBdC - delta Orientierung
ERC MOVE_REL TOOL bodyname dXdYdZ dAdBdC [m,deg]	Relatives Verschieben eines Parts aus der TOOL-Gruppe bodyname - Name des Body dXdYdZ - delta Position. dAdBdC - delta Orientierung
ERC MOVE_REL ROBOT bodyname dXdYdZ dAdBdC [m,deg]	Relatives Verschieben eines Parts aus der ROBOTER-Gruppe bodyname - Name des Body dXdYdZ - delta Position. dAdBdC - delta Orientierung
ERC MOVE_REL BODY_GRP bodyname dXdYdZ dAdBdC [m,deg]	Relatives Verschieben der kompletten BODY-Gruppe, bezogen auf den Reference Body bodyname - Name des Reference Body dXdYdZ - delta Position. dAdBdC - delta Orientierung
ERC MOVE_REL TOOL_GRP bodyname dXdYdZ dAdBdC [m,deg]	Relatives Verschieben der kompletten -Gruppe, bezogen auf den Reference Body bodyname - Name des Reference Body dXdYdZ - delta Position. dAdBdC - delta Orientierung

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERC MOVE_REL ROBOT_GRP bodyname dXdYdZ dAdBdC [m,deg]	Relatives Verschieben der kompletten -Gruppe, bezogen auf den Reference Body bodyname - Name des Reference Body dXdYdZ - delta Position. dAdBdC - delta Orientierung
ERC MOVE_REL LIST listname dXdYdZ dAdBdC [m,deg]	Relatives Verschieben aller Parts der LISTE listname - Name der Liste dXdYdZ - delta Position. dAdBdC - delta Orientierung

ERCL - Grab und Release Kommandos

Kommando und Syntax	**Beschreibung**
ERC GRAB BODY 'bodyname'	Greifen eines Body bodyname - Name des Body
ERC GRAB BODY_GRP	Greifen aller Parts der BODY_GRP
ERC RELEASE BODY 'bodyname'	Loslassen des Body bodyname - Name des Body
ERC RELEASE BODY_GRP	Loslassen aller Parts der BODY_GRP
ERC GRAB DEVICE devname	Der aktuelle Roboter greift den Roboter bzw. das Gerät mit dem Namen ‘devname’
ERC GRAB_TO DEVICE devname targetdevname	Das Gerät mit dem Namen ’devname’ wird vom Gerät mit dem Namen ‘targetdevname’ gegriffen
ERC RELEASE DEVICE devname	Das Gerät mit dem Namen ’devname’ wird losgelassen

ERCL - Roboter / Device Kommandos

Kommando und Syntax	**Beschreibung**
ERC CURRENT_DEVICE SET 'robotname’	Aktiviert den Roboter mit Namen ’robotname’ Anm.: Alle folgenden ERPL- und ERCL-Kommandos beziehen sich dann auf den aktuell aktivierten Roboter. Der aktuelle Roboter ist ‘cRobot’
ERC CURRENT_DEVICE UNSET	Deaktiviert den aktuellen Roboter und schaltet den vorigen Roboter wieder aktuell. Beispiel: Bei einer Funktion zum Schließen einer Spotweldgun benutzen Sie das Kommando wie folgt: (cRobot ist ‘MyRobot’) erc current_device set SpotWeldGun ptp_ax 0 0 ! move joint 1 und 2 nach 0 erc current_device unset ! disables ‘SpotWeldGun’, enables ‘MyRobot’
ERC CURRENT_DEVICE CLEAR	Löscht den gesamten Device Stack
ERC CURRENT_DEVICE SHOW	Zeigt den kompletten Device Stack im Messagefenster
TOOL DEVICE robotname	Setzt den TCP für den aktuellen Roboter auf die TCPwerte des Roboters ‘robotname’. Dieses Kommando ist nützlich nachdem ein anderer Roboter gegriffen wurde.

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERC ROBOT_BASE XYZ ABC [m,deg]	Verschiebt die Basis des aktuellen Roboters and die Position XYZ - Position ABC - Orientierung
ERC ROBOT_BASE tagname	Verschiebt die Base des aktuellen Roboters zur Tagpunktposition Tagname - Name des TAGs
ERC ROBOT_BASE_REL XYZ ABC [m,deg]	Verschiebt die Base des aktuellen Roboters relativ dXdYdZ - delta Position. dAdBdC - delta Orientierung

ERCL - TAG Kommandos

Kommando und Syntax	**Beschreibung**
ERC CREATE_TARGET_TAGS ON/OFF	Ein- / Ausschalten der Funktion „Tagpunkt am Zielpunkt erzeugen“
ERC TAGS PREFIX prefixname	Setzt das Prefix für Tagpunkte Prefixname - Name des Prefix **Anm.**:Ein neu erzeugter Tagpunkt erhält den Prefixnamen
ERC TAGS DELETE tagname	Löscht den Tagpunkt aus der Zelle Tagname - Name des Tag
ERC TAGS DELETE ALL	Löscht alle Tagpunkte aus der Zelle

ERCL - View Kommandos

Kommando und Syntax	**Beschreibung**
ERC VIEW steps n	Setzt die Anzahl View Steps (wichtig wenn eine *.vie-Datei geladen wird)
ERC VIEW hither x	Setzt den Wert für vordere (begrenzende) Betrachtungsebene (hither plane)
ERC VIEW yonder x	Setzt den Wert für hintere (begrenzende) Betrachtungsebene (yonder plane)
ERC VIEW zoom x	Zoomt die Szene auf den Wert „x“
ERC VIEW zoom_in x	Zoomt die Szene hinein um den Wert „x“
ERC VIEW zoom_out x	Zoomt die Szene heraus um den Wert „x“
ERC VIEW tcp_rot_tcp ABC	Rotiert die Szene um den TCP des aktuellen Roboters bezogen auf die TCP Orientierung ABC - relative Rotation
ERC VIEW tcp_rot_world ABC	Rotiert die Szene um den TCP des aktuellen Roboters bezogen auf die World frame Orientierung ABC - relative Rotation
ERC VIEW world_rot_base ABC	Rotiert die Szene um das Weltkoordinatensystem bezogen auf die Orientierung des Roboterbasisframes ABC - relative Rotation
ERC VIEW TOP [BOTTOM, LEFT, RIGHT, FRONT, REAR, ZOOM_WORLD]	Zoomt die Szene in vordefinierte Perspektiven (Top, Bottom, Left, etc.). ZOOM_WORLD zoomt die Szene in den sichtbaren Bereich

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERC VIEW world_rot_world ABC	Rotiert die Szene um das Weltkoordinatensystem bezogen auf die World frame Orientierung ABC - relative Rotation
ERC LOAD VIEW filename	Lädt eine Viewdatei (*.vie) **Anm.**: Die neue Ansicht wird mit der Anzahl „n“ der View Steps angefahren 'view steps' ERC VIEW steps n

ERCL - TCP Trace Kommandos

Kommando und Syntax	**Beschreibung**
ERC TRACK_TYPE type [sieze]	Setzt den Tracetyp und Stil Typen: POINT LINE LINE_Z_DIRECTION, Z_DIRECTION X_DIRECTION Y_DIRECTION [size] Länge in gewählter Richtung **Beispiele**: erc track_type line erc track_type line_z_direction 0.2 erc track_type line_z_direction -0.2
ERC TRACK ON,OFF	Ein- / Ausschalten des TCP Trace für aktuellen Roboter ‘cRobot’
ERC COLOR track color	Setzt/modifiziert die Farbe des TCP Trace track TRACK, TRACK_DYN color vordefinierte Farbe **Beispiel**: erc color TRACK RED erc color TRACK -1, für wechselnde Standardfarben [1..15]

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - Kollisions Kommandos

Kommando und Syntax	Beschreibung
ERC Collision ON/OFF	Ein- / Ausschalten der Kollisionsüberprüfung **Anm**.: Bei eingeschalteter Prüfung werden alle zum Roboter/Tool gehörenden Geometrien gegen andere Roboter und Tools geprüft. Dies ist die Standardkollisionskette.
ERC RED_BLUE_COLLISION ON, OFF	Ein- / Ausschalten der „Rot-Blau-Kollision“ Im Fall einer Kollision werden alle kollidierenden Geometrien in Rot dargestellt und nicht-kollidierenden Geometrien in Hellblau dargestellt.
ERC COLLISION BODY [ROBOT, TOOL] bodyname OFF, ON=CONCAVE, CONVEX, BBOX	Ein- / Ausschalten der Kollisionsattribute eines Bodys: OFF, ! Kollisionscheck aus für diese Geometrie ON = CONCAVE ! Kollisionscheck Concav CONVEX ! Kollisionscheck mit convexer Hülle BBOX ! Kollisionscheck mit bounded box Hülle
ERC COLLISION BODY_GRP [ROBOT_GRP,TOOL_GRP] OFF, ON= CONCAVE, CONVEX, BBOX	Ein- / Ausschalten der Kollisionsattribute für eine komplette Gruppe OFF, ON=CONCAVE, CONVEX oder BBOX **Anm.**: Robot_Grp und Tool_Grp haben Auswirkung auf Geometrien, die zum aktuell gewählten Roboter ‘cRobot’ gehören
ERC COLLISION QUEUE BODY_ROBOT [BODY_TOOL, ROBOT_TOOL, GRABBODY_ROBOT, GRABBODY_BODY, ROBOT_ROBOT BODY_BODY ALL] ON,OFF	Ein- / Ausschalten der vordefinierten Kollisionsketten BODY_ROBOT – prüft Kollision zwischen Body und Roboter BODY_TOOL – prüft Kollision zwischen Body und Tool ROBOT_TOOL – prüft Kollision zwischen Roboter und Tool GRABBODY_ROBOT – prüft Kollision zwischen gegriffenen Bodies und. Robotern GRABBODY_ BODY – prüft Kollision zwischen gegriffenen Bodies und nicht gegriffenen Bodies ROBOT_ROBOT – prüft Kollision zwischen allen Geometrien die zu einer Robot_Grp gehören BODY_BODY – prüft Kollision zwischen allen Geometrien die zu einer Body_Grp gehören ALL – Ein-/Ausschalten aller vordefinierten Ketten
ERC COLLISION_TOL BODY [ROBOT, TOOL] bodyname tolerance	Toleranzwert eines Bodys ‚bodyname’ aus Body-, Robot oder Tool-Gruppe beim Concave Kollisionscheck in [mm]
ERC COLLISION_TOL BODY_GRP [ROBOT_GRP, TOOL_GRP] tolerance	Toleranzwert für komplette Gruppe beim Concave Kollisionscheck in [mm]
ERC COLLISION DISTANCE tolerance [mm]	Setzt die globale Kollisionsschwelle ‘tolerance’ in mm. Kollision wird dann angezeigt, wenn die minimale Distanz zwischen zwei getesteten Geometrien kleiner ist als die Kollisionsschwelle ‘tolerance’.

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - Attach Kommandos

ERC ATTACH ROBOT 'bodyname' to_ROBOT ['bodyname2']	Reattached einen Body an einen anderen Roboterjoint innerhalb der Robot_Grp, benannt mit 'bodyname2' **Anm.**: Wenn kein 2. Body benannt wird, wird an der Basis attached
ERC ATTACH ROBOT 'bodyname' to_TOOL	Reattached einen Body am Tip des aktuellen Roboters

ERCL - Unit Kommandos

ERC UNIT M	Setzt die Einheit Meter [m] für alle folgenden ERPL & ERCL Kommandos Beispiel: speed_cp 0.1 is 100 mm/s
ERC UNIT MM	Setzt die Einheit Millimeter [mm] für alle folgenden ERPL & ERCL Kommandos Beispiel: speed_cp 100 is 100 mm/s
ERC UNIT INCH	Setzt die Einheit inch [inch] für alle folgenden ERPL & ERCL Kommandos Beispiel: speed_cp 1 is 25.4 mm/s
ERC UNIT DEG	Setzt die Rotationseinheit auf Degree [°] Beispiel: speed_ori 20 is 20°/s
ERC UNIT RAD	Setzt die Rotationseinheit auf Radiant [rad] Beispiel: speed_ori 3.1415 is 180°/s
ERC UNIT TRANS_SCAL x	Skaliert die interne Einheit [m] um den userdefinierten Wert x Beispiel: x=10 speed_cp 0.1 is 1 m/s = 1000 mm/s
ERC UNIT ROT_SCAL x	Skaliert die interne Einheit [°]um den userdefinierten Wert x Beispiel: x= 10 speed_ori 2 is 20°/s

CALC - Math Kommandos

CALC 'math expression'	Beispiel: calc a=0.5 calc b = a * sin (45.0*RAD) LIN_REL 0 0 0.1*a 0 0 0
CALC SHOW_VARS	Zeigt alle globalen Variablen in einem Messagefenster
CALC SHOW_USER_VARS	Zeigt alle userdefinierten globalen Variablen (ohne prefix '$') in einem Messagefenster
CALC CLEAR_VARS	Löscht alle globalen Variablen
CALC CLEAR_USER_VARS	Löscht alle userdefinierten globalen Variablen (ohne prefix '$')
= msg('math expression')	Wertausgabe im Messagefenster. Beispiel: = msg(a)
= 'math expression'	Ergebnisausgabe im Programmfenster. Beispiel: = a

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - PARAMETER Kommandos

ERC SET_PARAMETER $NAME_1 'name'	Kopiert „name" in den ersten von 100 möglichen globalen String-Parameter $NAME_1 = 'name' Beispiel: ERC SET_PARAMETER $NAME_1 T_1 ! Tag point ERC SET_PARAMETER $NAME_2 MYROB ! Devicename … ERC CURRENT DEVICE SET $NAME_2 ! setzt 'MYROB' als aktuell LIN $NAME_1 ! fahr zu Tag 'T_1'
ERC SET_PARAMETER $NAME_1 text_word [value]	Kopiert das Textwort plus die Zahl in den ersten von 100 möglichen globalen String-Parameter $NAME_1 = text_word [value] Beispiel: ii=1 ERC SET_PARAMETER $NAME T_ ii ! Tag point "T_1"
ERC SET_PARAMETER $NAME_RESET	Löscht alle globalen String Parameter
ERC SET_PARAMETER $NAME_INFO	Zeigt alle globalen String Parameter in einem Messagefenster

ERCL - KUD Kommandos

ERC KUD row column x	Setzt den Wert 'x' für die Kinematik User Data für den aktuellen Roboter mittels : row [1..12] column [1..12] Beispiel: ERC KUD 1 1 1 ERC KUD 12 12 144
ERC KUD name x	Setzt den Wert 'x' für die Kinematik User Data mit Namen ‚name' für den aktuellen Roboter Mögliche Namen sind: KUD_1_1 …. KUD_12_12 Anm.: Der User kann die KUD-Name verändern Beispiel: ERC KUD_1_1 1 ERC KUD_12_12 144

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - zusätzliche Kommandos

Kommando und Syntax	Beschreibung
ERC STOP	Stoppt (Abbruch) die Programmausführung
ERC PAUSE	Hält die Programmausführung an
ERC ESSI ON,OFF [speed scale value] [size scale value]	Ein- / Ausschalten der Interpretation von ESSI NC-Code Optionale Parameter: Speed scale value: skalieren die programmierte Geschwindigkeit. Size scale value: skaliert die Zielposition (Programmskalierung) **Anm.**: Dieses Kommando erfordert die lizenzierte NC-Option
ERC EIA ON,OFF [speed scale value] [size scale value]	Ein- / Ausschalten der Interpretation von EIA NC-Code, DIN 66025 Optional Parameter Speed scale value: skalieren die programmierte Geschwindigkeit. Size scale value: skaliert die Zielposition (Programmskalierung) **Anm.**: Dieses Kommando erfordert die lizenzierte NC-Option
ERC BASE BODY bodyname	Setzt die BASE auf den Body bodyname
ERC BASE TCP	Setzt die BASE auf den TCP des aktuellen Roboters
ERC SWE_NEG swe1 ... swen	Setzt die **negativen** Limits der Verfahrbereiche für jede Achse [m,deg] Die Anzahl der Werte sollte mit der Anzahl der Joints / Achsen übereinstimmen
ERC SWE_POS swe1 ... swen	Setzt die **positiven** Limits der Verfahrbereiche für jede Achse [m,deg] Die Anzahl der Werte sollte mit der Anzahl der Joints / Achsen übereinstimmen
ERC CART_LIMITS_MIN X Y Z	Setzt die **minimalen** kartesischen Begrenzungswerte [m] für die X, Y und Z-Koordinatenrichtung
ERC CART_LIMITS_MAX X Y Z	Setzt die **maximalen** kartesischen Begrenzungswerte [m] für die X, Y und Z-Koordinatenrichtung
ERC TURN_INTERVAL Ax1 ...Axn [deg]	TURN -Intervalle für jede Achse Ax1…Axn, im Bereich zwischen [0°,∞°[
ERC TURN_OFFSET Ax1 ...Axn [deg]	TURN - Offset für jede Achse Ax1…Axn, im Bereich zwischen]-∞°, ∞°[
ERC JOINT_WEIGHT 0 or 1 for number of joints	Setzt den „joint_weight vector“, zur Beeinflussung der numerischen Lösung Die Anzahl der Werte sollte mit der Anzahl der Joints / Achsen übereinstimmen
ERC MASK_VECTOR [0,1] for X Y Z A B C	Setzt den „mask vector“, zur Beeinflussung der numerischen Lösung
ERC CELL_INFO text	Text für Zellen-Informationszeile

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - 3D-PDF-Export Kommandos

Nachfolgenden Kommandos erfordern die lizensierte 3D-PDF Export-Option

Kommando und Syntax	Beschreibung
ERC _3D_PDF_EXPORT SCREENSHOT [flnname]	Erzeugt ein 3D-PDF-Dokument der aktuellen Szene (ohne Animation), d.h. ein Standbild mit allen Achswinkelstellungen zum Zeitpunkt des Befehls. Wird kein [flnname] eingegeben, wird der Name der Arbeitszelle für das 3D-PDF-Dokument übernommen. filename: Name des erzeugten 3D-PDF-Dokuments
ERC _3D_PDF_EXPORT ON / OFF [flnname]	Aktivieren/ Deaktivieren der Aufzeichnung des 3D-PDF-Dokumentes. Wird kein [flnname] eingegeben, wird der Name der Arbeitszelle für das 3D-PDF-Dokument übernommen. Beim Beenden der Aufzeichnung wird das 3D-PDF-Dokument automatisch erzeugt und unter dem angegebenen Namen gespeichert. **Wichtig:** Wurde zuvor ein 3D-PDF-Dokument erzeugt und geöffnet, so muss dieses geschlossen werden, damit die Aufzeichnung beendet und das Dokument erneut gespeichert werden kann.
ERC _3D_PDF_EXPORT SET_FILE flnname	Das 3D-PDF-Dokument wird mit dem unter „flnname“ angegebenen Namen erzeugt. Zuvor muss der Befehl _3D_PDF_EXPORT ON gesetzt werden. filename: Name des erzeugten 3D-PDF-Dokuments
ERC _3D_PDF_EXPORT SET_LABEL labelname	Setzt ein Label mit dem unter “labelname” angegebenen Namen. Der Name des Labels wird im geöffneten 3D-PDF-Dokument neben der realen Prozesszeit angezeigt und dient der individuellen Kennzeichnung einzelner Prozess-Abschnitte. Zuvor muss der Befehl _3D_PDF_EXPORT ON gesetzt werden. labelname: Name des labels
ERC _3D_PDF_EXPORT SET_PASSWORD passwordname	Das 3D-PDF-Dokument wird mit dem unter “passwordname” angegebenen Passwort geschützt. Dieses muss beim Öffnen des Dokuments eingegeben werden. Zuvor muss der Befehl _3D_PDF_EXPORT ON gesetzt werden. passwordname: Name des Passwords
ERC _3D_PDF_EXPORT PAUSE	Pausiert die Aufzeichnung des 3D-PDF-Dokuments. Ein erneutes Setzen des Befehls setzt die Aufnahme fort. Zuvor muss der Befehl _3D_PDF_EXPORT ON gesetzt werden.
ERC _3D_PDF_EXPORT DEACTIVATE	Die Aufnahme des 3D-PDF-Dokumentes wird abgebrochen und das Dokument verworfen.

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

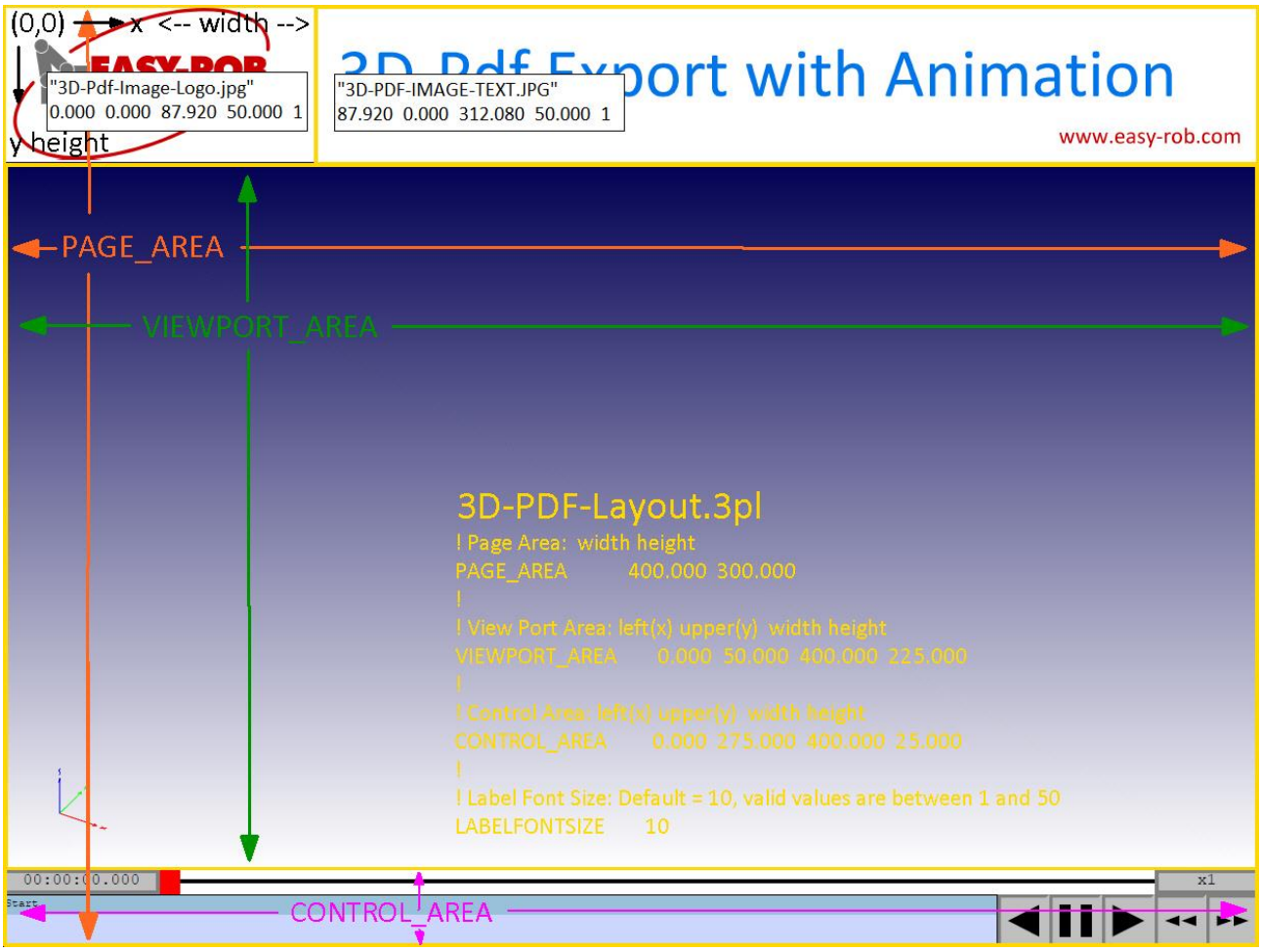
ERCL - 3D-PDF-Export Layout Definition Kommandos

Nachfolgenden Kommandos erfordern die lizensierte 3D-PDF-Export-Option

Das 3D-PDF Layout kann durch Laden einer 3D-Pdf Layout Datei (.3pl) festgelegt werden oder dynamisch und individuell während des Simulationslaufes.

Die Maße eines 3D-Pdf Layouts werden durch die Parameter PAGE_AREA, VIEWPORT_AREA und CONTROL_AREA bestimmt. Die Positionen der Bilder sollten außerhalb definiert werden. Bei Überlappung mit dem PAGE_AREA werden sie im Hintergrund dargestellt.

Beispiel eines 3D-Pdf Layouts



3D-Pdf Layouts mit 400mm x 300mm mit 2 Bildern

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Kommando und Syntax	Beschreibung
ERC _3D_PDF_EXPORT RESET_LAYOUT	3D-Pdf Layout auf Standard-Werte zurücksetzen
ERC _3D_PDF_EXPORT LOAD_LAYOUT filename.3pl	3D-Pdf Layout aus .3pl Datei "filename.3pl" laden
ERC _3D_PDF_EXPORT SAVE_LAYOUT filename.3pl	3D-Pdf Layout in .3pl Datei "filename.3pl" speichern
ERC _3D_PDF_EXPORT PAGE_AREA width height	Page Area des 3D-Pdf Layout festlegen Parameter: width height, siehe Bild
ERC _3D_PDF_EXPORT VIEWPORT_AREA left(x) upper(y) width height	Viewport Area des 3D-Pdf Layout festlegen Parameter: left(x) upper(y) width height, siehe Bild
ERC _3D_PDF_EXPORT CONTROL_AREA left(x) upper(y) width height	Control Area des 3D-Pdf Layout festlegen Parameter: left(x) upper(y) width height, siehe Bild
ERC _3D_PDF_EXPORT LABEL_FONT_SIZE size	Label Font Size im Control Area des 3D-Pdf Layout festlegen Parameter: size im Bereich [1-50], Default-Wert: size = 10

Bilder zum 3D-PDF Layout hinzufügen.

Der Pfad muss vorab mit dem Befehl IMAGE_PATH festgelegt werden. Pfad- und Bildangaben müssen in "" erfolgen.

ERC _3D_PDF_EXPORT IMAGE_PATH "path"	Pfadangabe aus denen die Bilder mit ADD_IMAGE geladen werden.
ERC _3D_PDF_EXPORT IMAGE_PATH USERPROFILE	Der Key "USERPROFILE" bestimmt den Pfad aus denen die Bilder mit ADD_IMAGE geladen werden z. B: c:\Users\MyLoginName
ERC _3D_PDF_EXPORT IMAGE_PATH WORKING_DIRECTORY	Der Key "WORKING_DIRECTORY" , current working directory bestimmt den Pfad aus denen die Bilder mit ADD_IMAGE geladen werden
ERC _3D_PDF_EXPORT IMAGE_PATH ""	Same as Key "WORKING_DIRECTORY" current working directory
ERC _3D_PDF_EXPORT IMAGE_PATH 3PL_FILE_FOLDER	Key "3PL_FILE_FOLDER " Der Pfad der in der geladenen 3D-Pdf Layout Datei .3pl definiert ist bestimmt den Pfad aus denen die Bilder mit ADD_IMAGE geladen werden
ERC _3D_PDF_EXPORT ADD_IMAGE "filename.jpg" left(x) upper(y) width height Scaling	Fügt ein jpg Bild an definierte Position ein Parameter 1: "Image file name" Parameter 2-5: left(x) upper(y) width height Parameter 6: Scaling is one of ISO_Stretch = 0 or ISO_CenterFit = 1

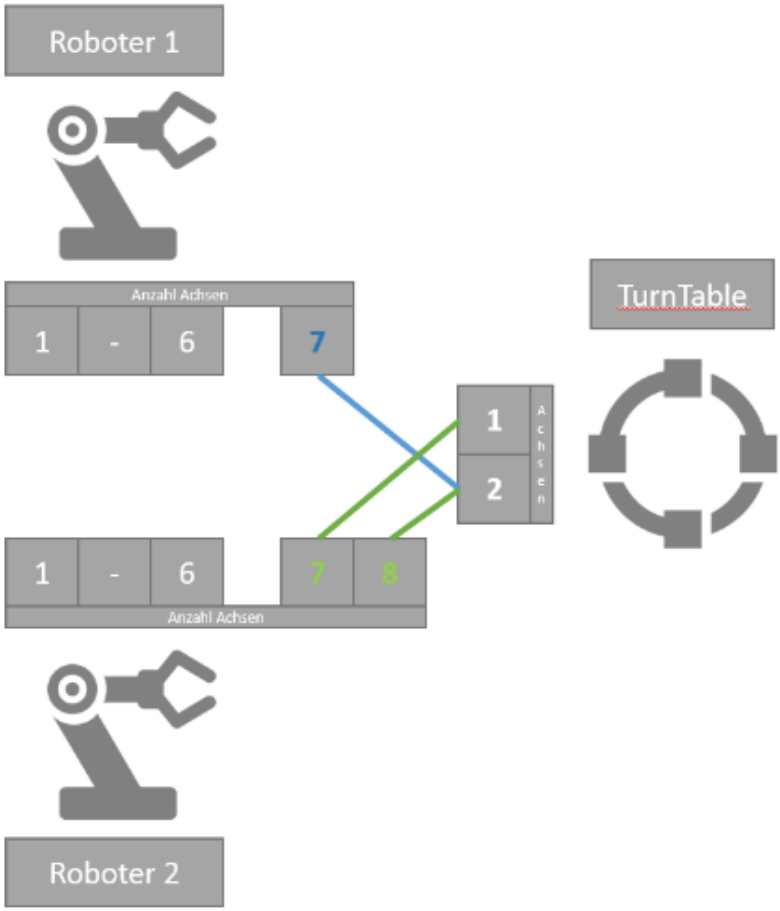
EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

ERCL - Linkage Befehle

Kopplung von Geräten über ERC Befehle mit der Möglichkeit des Mappings von Achsen.
Damit kann u.a. elegant ein Master – Slave Wechsel zum Simulationslauf realisiert werden.

ERC LINKAGE DEVICE SET ['DeviceName'] [AxIdx(1)] .. [AxIdx(n)]	Bewirkt die Kopplung des aktuellen Geräts zum Simulationslauf DeviceName = Name des Gerätes, mit welchen gekoppelt werden soll AxIdx(1) = erster Achsen-Index AxIdx(n) = n-ter Achsen- Index
ERC LINKAGE DEVICE UNSET ['DeviceName']	Aufhebung der Kopplung DeviceName = Name des Gerätes, mit welchen Kopplung aufgehoben werden soll

Schematische Darstellung des Achsen-Mappings



blau = ERC LINKAGE DEVICE SET TurnTable 0 7
grün = ERC LINKAGE DEVICE SET TurnTable 7 8

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™

Kontakt

EASY-ROB Software GmbH

Adresse: Hauptstr. 42
65719 Hofheim am Taunus
Germany

Kontaktperson: Stefan Anton, Patryk Lischka

Tel.: +49 (0) 6192 921 70 77
FAX.: +49 (0) 6192 921 70 66

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

Url: www.easy-rob.com

EASY-ROB Kundenbereich

Inhalte: Programm-Updates und Roboter Bibliotheken

Web-Adresse: www.easy-rob.com/downloads/kundenbereich/

Zugangsdaten:

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

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com

Platz für Ihre Notizen

EASY-ROB Software GmbH Hauptstr.42 65719 Hofheim am Taunus Germany Tel. +49 6192 921 70-77 Fax. +49 6192 921 70-66 info.er@easy-rob.com www.easy-rob.com