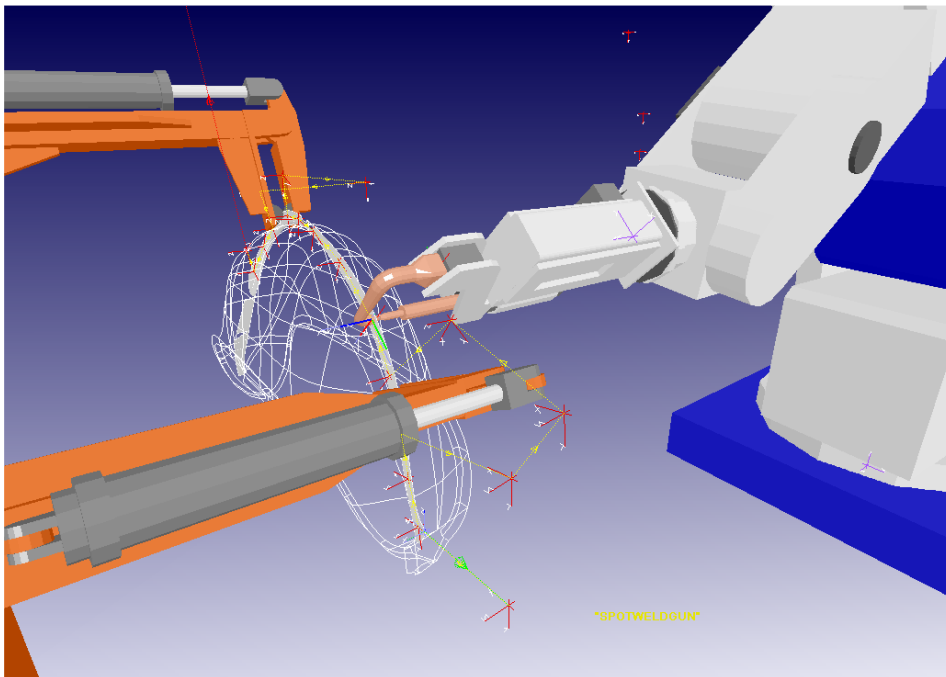


Update

EASY-ROB™ V4.007



September 2005

Version 1.1

EASY-ROB™ V4.007

Zur aktuellen Version

Die aktuelle Version EASY-ROB™ V4.007 beinhaltet eine Vielzahl neuer Funktionen die am Beispiel der Arbeitszelle „Spotweld_demo.cel“ erläutert werden sollen. Als wichtigste Weiterentwicklung sind die „Multiple Kinematiken“ zu nennen, die auch einen entscheidenden Einfluss auf das EASY-ROB™ Bedienkonzept haben, um die Vorteile zu nutzen. Darüber hinaus sind weitere Bedienelemente, ERPL/ERCL Sprachbefehle und API C-Funktionen hinzugekommen, die das Planen, Layouten und Simulieren von Roboterarbeitszellen erleichtern. Es ist jedoch so, dass auch mit dieser Version nur ein weiterer Meilenstein auf dem langen Weg zum Ziel erreicht ist. Die neuen Funktionen sind sicherlich hilfreich und von Vorteil, doch ergeben sich hieraus unmittelbar neue Anforderungen. An dieser Stelle möchten wir uns bei unseren Kunden und Anwendern bedanken, die mit Ihren guten Vorschlägen die Entwicklung in die richtige Spur halten.


Vielen Dank

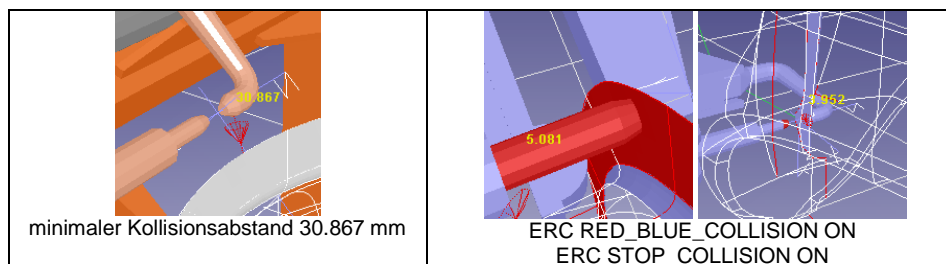
Stefan Anton


EASY-ROB
3D Robot Simulation Tool

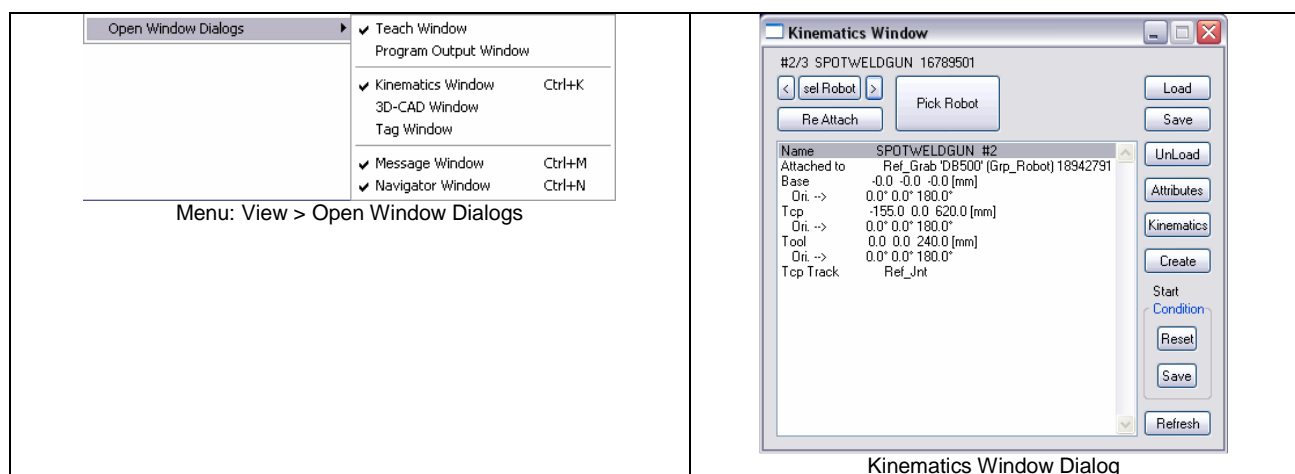
Arbeitszelle „Spotweld_demo.cel“

Das Zip-Archiv „Projects-v4007.zip“ beinhaltet mehrere Beispiele. Bitte entpacken Sie die Zip-Datei mit Unterverzeichnissen an einem geeigneten Ort auf Ihrer Festplatte.

Starten Sie nun EASY-ROB™ und laden Sie die Zelle „Spotweld_demo.cel“. Sobald die Zelle vollständig geladen ist, klicken Sie „Run“ , um das mit der Zelle verknüpfte Programm „spotweld_demo.prg“ und die Mimic-Datei „spotweld.mmc“ auszuführen. Sie sehen, wie der Roboter eine Punktschweißzange greift und mit dieser einige Punkte am Bauteil schweißt. Während der Simulation werden einige Perspektiven bzw. View-Dateien „spotweld_n.vie“ geladen, so dass das Szenario optimal beobachtet werden kann. Weiterhin wurde die Kollisionsprüfung aktiviert, die hier wissentlich nicht vermieden wurde. Tritt eine Kollision zwischen den Geometrien der Roboter und Kinematiken auf, verfärben sich die kollidierenden Geometrien rot und die nicht kollidieren Geometrien blau. So können die Orte der Kollision besser lokalisiert werden.



Die Sportweld-Zelle besteht aus insgesamt drei Kinematiken. Die erste Kinematik ist der 6 achsige Knickarmroboter mit dem Namen DB500. Die zweite Kinematik ist die Punktschweißzange „SPOTWELDGUN“, die dritte Kinematik die Fixture mit dem Bauteil „WELD_FIXTURE“. Einen guten Überblick sämtlicher geladenen Kinematiken gibt das „Kinematics Window“. Sie können das „Kinematics Window“ durch den Short-Key „Ctrl+K“, durch zweimaliges Klicken des  Icons oder vom Menu „View > Open Window Dialogs“ öffnen. Der Dialog kann während der Simulation geöffnet bleiben.

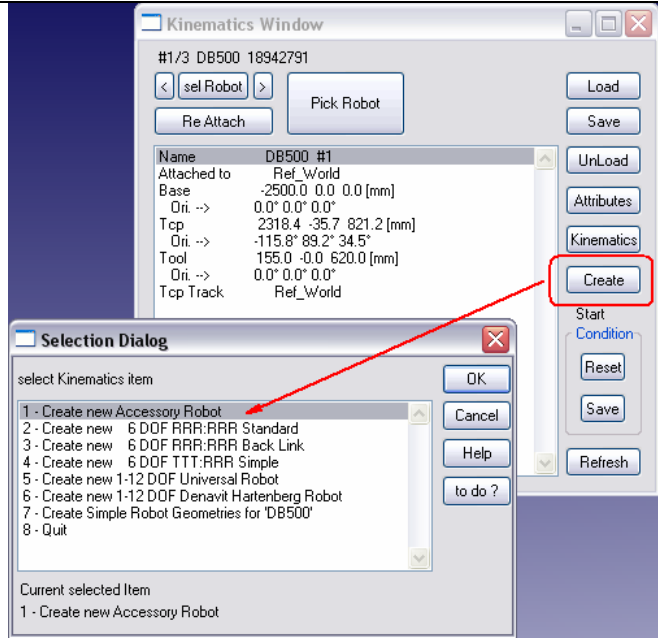


Weitere Informationen können dem Pdf-Dokument „Bedienungs-Hinweise_Multiple-Kin.pdf“ entnommen werden.

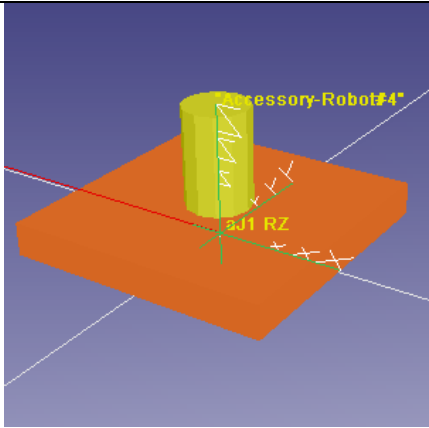
Im Unterschied zur Bedienphilosophie vorheriger EASY-ROB™ Versionen hat unsere Arbeitszelle eine leere Body- bzw. Umgebungsgruppe. Normalerweise hätten wir die zwei Geometrien der „WELD_FIXTURE“ Kinematik dieser ‚Body-Group‘ zugewiesen und hätten dann mit erhöhtem Aufwand beide Geometrien einzeln positionieren müssen. Mit EASY-ROB™ V4.007 gilt nun.

- Für jede Geometrie-Gruppe wird eine Kinematik erzeugt. Die erzeugte „Zubehör“-Kinematik hat per Grundkonfiguration eine Rotationsachse in z-Richtung. Dieser Kinematik können ‚beliebig‘ viele Geometrien zugewiesen werden. Durch Verschieben der Kinematik- bzw. Roboter-Basis werden alle zur Gruppe gehörenden Geometrien mit verschoben.


Eine neue Kinematik kann direkt vom ‚Kinematics Window‘ heraus erzeugt werden. Indem Sie auf den ‚Create-Button‘ klicken und ‚Accessory Kinematics‘ auswählen wird eine Kinematik mit Namen ‚Accessory-Robot#4‘ und einer Rz-Achse erzeugt. Weiterhin werden optional auch pro Achse vereinfachte Robotergeometrien (eine orange farbige Box der Kinematikbasis zugewiesen und ein gelber Zylinder der ersten Achse zugewiesen) erzeugt. Diese können im ‚3D-CAD Window‘ durch den Import geeigneter CAD Geometrien ersetzt werden.



Eine neue ‚Accessory Kinematik‘ erzeugen als „Universal Robot“ mit einem Freiheitsgrad Rot(z,q1)



Accessory Kinematik mit vereinfachten default Geometrien.

Mit dem Icons  werden die Namen, Koordinatensysteme, Achsnummern und -gelenkrichtungen sowie die Referenzverbindungen aller Kinematiken angezeigt.

Verketteten von Kinematiken

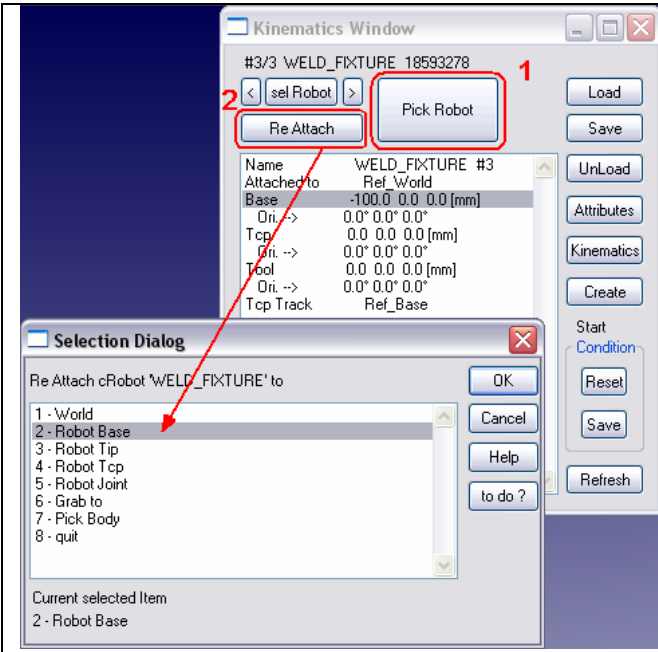
Kinematiken können verkettet werden. Betrachten wir den Roboter DB500, lässt sich aus der zweiten Zeile der Liste im ‚Kinematics Window‘ ablesen „**Attached to Ref_World**“. Somit ist der DB500 bzgl. Welt positioniert. Die Position des DB500 wird durch ‚Base‘ = (-2500 , 0 , 0), bezogen auf das Weltkoordinatensystem, beschrieben.

Tipp: Durch Doppelklick auf die dritte Zeile kann die Roboterbasis ‚Base‘ sofort mit dem ‚Frame Dialog‘ geändert werden.

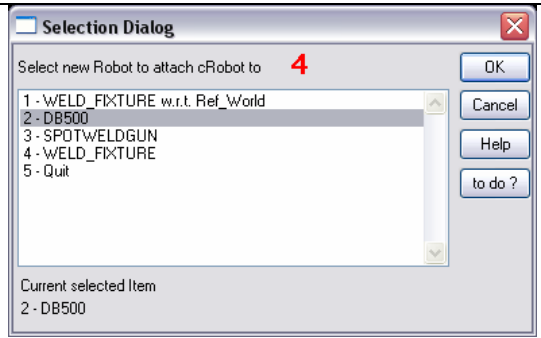
Gleiches gilt für die ‚WELD_FIXTURE‘, die wiederum bzgl. Welt beschrieben ist. Wollen wir nun, dass sich die ‚WELD_FIXTURE‘ beim verschieben des DB500 mitbewegt, muss die ‚WELD_FIXTURE‘ an die Basis des DB500 ‚attached‘ werden.

Um die ‚WELD_FIXTURE‘ an die Basis des DB500 zu „attachen“, sind folgende Schritte durchzuführen


1. Wählen sie als aktuelle Kinematik die ‚WELD_FIXTURE‘ aus, indem Sie im ‚Kinematics Window‘ auf den Button ‚Pick Robot‘ klicken und die ‚WELD_FIXTURE‘ mit der linken Maustaste auswählen.
2. Klicken Sie den Button ‚Re Attach‘ im ‚Kinematics Window‘.
3. Wählen Sie ‚2 - Robot Base‘ aus
=> „Re Attach cRobot ‚WELD_FIXTURE‘ to Robot Base
4. DB500 auswählen
=> Die ‚WELD_FIXTURE‘ bewegt sich nun mit der Basis des DB500
5. Für den „Ja“ Fall, dass die Position der ‚WELD_FIXTURE‘ im Raum unverändert bleiben soll, müssen die Koordinaten entsprechend transformiert werden.



Schritt 1, 2 und 3




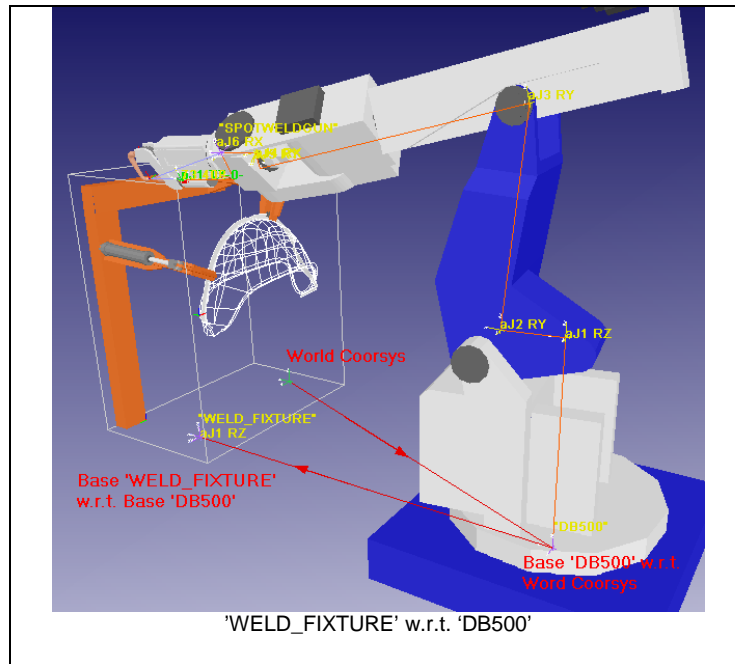
Schritt 4, DB500 auswählen.



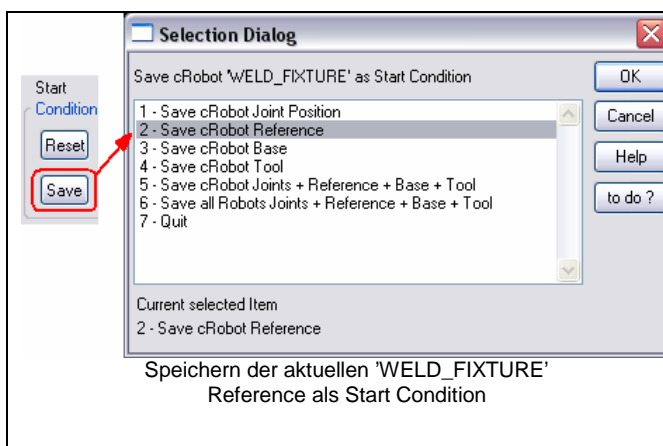
Schritt 5, Transformation der Basis, so dass die resultierende Position der ‚WELD_FIXTURE‘ im Raum beibehalten bleibt.
Die neu berechnete Base ist (2400 , 0 , 0)

Wichtig: Die Anzahl der Kinematiken in einer Arbeitszelle ist nicht begrenzt. Es können beliebig viele Kinematiken geladen und ‚attached‘ werden, mit der Ausnahme, dass keine geschlossenen Ketten gebildet werden dürfen.

Klicken Sie auf das Icon  um die Verkettung der Kinematiken darzustellen. Die roten Referenzverbindungen bzw. Verbindungslinien beschreiben die Abhängigkeiten der Roboterbasen zueinander.



Verkettungen sind jedoch temporär und bleiben nur so lange erhalten bis sie gespeichert werden. Das hat folgenden Grund. Während eines Simulationslaufes werden Kinematiken gegriffen (Beispiel Spotweldgun) und deren Positionen und Achsgelenkstellung verändert. Wird der Simulationslauf mit STOP abgebrochen und erneut gestartet werden, 'springen' sämtliche Kinematiken und Geometrien in die so genannte **'Start Condition'** zurück. Das heißt, die Verkettung zwischen der 'WELD_FIXTURE' und des 'DB500' wird wieder aufgelöst. Soll die Verkettung erhalten bleiben, muss sie zuvor als 'Start Condition' gespeichert werden.



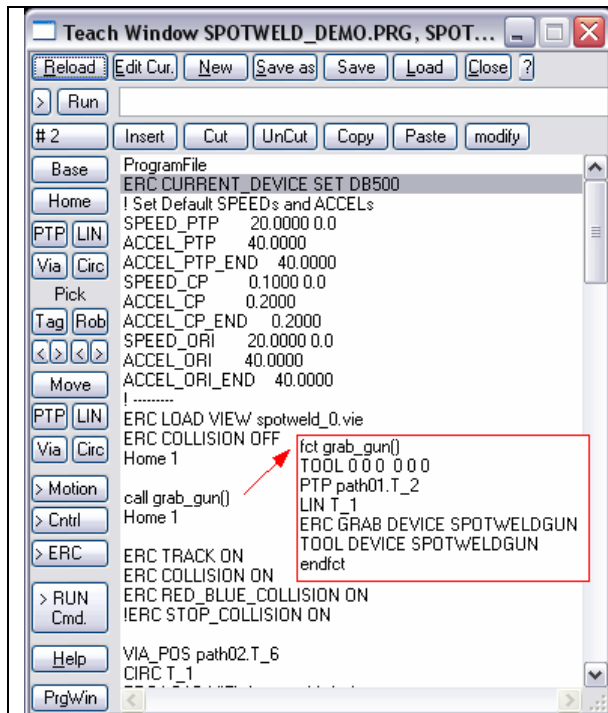
Tipp: Vergewissern Sie sich, dass die aktuelle Verkettung der Kinematiken gespeichert ist, bevor sie einen kompletten Simulationslauf mit RUN starten, wodurch sämtliche temporären Verkettungen auflöst werden.

Menu: Simulation >

Save Start Condition	Save all Robots Position and Joints as Start Condition	Ctrl+Shift+S
TCP Trace	ON/OFF	
Collision	ON/OFF	
RED-BLUE Collision	ON/OFF	
Dynamics	ON/OFF	
Simulation Data	Save all cRobot Body Positions as Start Condition	
Open Online Output Data	Save all cTool Body Positions as Start Condition	
Increase/Decrease Steps	Save all Environment Body Positions as Start Condition	
	Save cPath as Start Condition	
	Save all Path Positions as Start Condition	

Das 'Spotweld_Demo' Programm für den Simulationslauf

Das Programm ‚Spotweld_Demo.prg‘ und die zugehörige Mimic Datei ‚Spotweld.mmc‘ werden im folgenden erläutert, insbesondere die verwendeten Multi-KIN Befehle.



Mit dem ERCL-Befehl **"ERC CURRENT DEVICE SET DB500"** wird die DB500-Kinematik aktiv gesetzt.. Alle folgenden ERPL- und ERCL-Befehle beziehen sich auf die aktuelle Kinematik.

Um reproduzierbare Ergebnisse zu erhalten, werden Geschwindigkeiten und Beschleunigungen programmiert.

Die Funktion **grab_gun()** hat die Aufgabe den DB500 zum Greifpunkt T_2 des Pfades ,path01' zu fahren, die Kinematik ,SpotweldGun' zu greifen und den TCP-Punkt der Spotweldgun für den DB500 zu übernehmen.

Tool 000 000 setzt den TCP des DB500 an den Flansch
PTP path01.T_2 fährt den DB500 nach T_2 des Pfades
'path01'. Mit der Syntax 'path01.T_2' ist der
Tag-Punkt T_2 eindeutig bestimmt, denn
auch der Pfad 'path02' enthält einen Tag-
Punkt mit dem Namen 'T_2'

LIN T_1 führt den DB500 nach T_1 des aktuellen Pfades path01

ERC GRAB DEVICE SPOTWELDGUN

attached die Kinematik „SPOTWELDGUN“
an den Flansch des DB500.

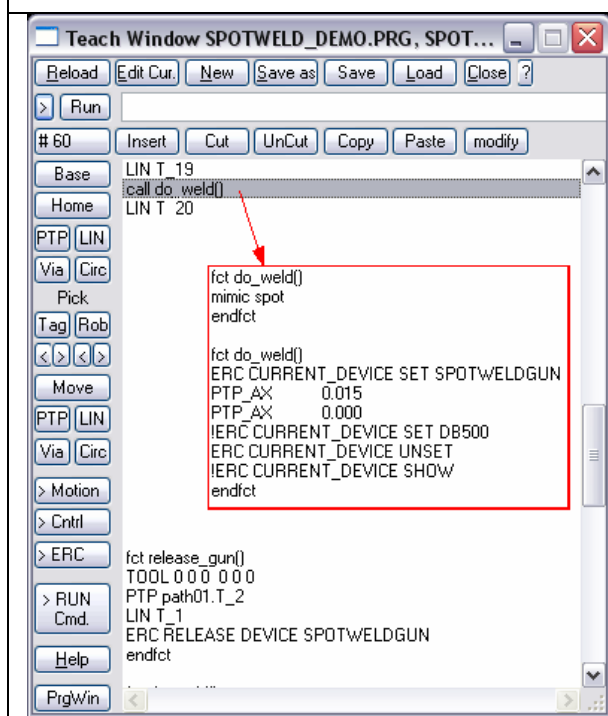
TOOL DEVICE SPOTWELDGUN

setzt den TCP des DB500 auf den TCP der SPOTWELDGUN

ERC COLLISION ON schaltet die Kollision ein

ERC RED BLUE COLLISION ON

- aktiviert die Red/Blue Kollision



Die Funktion **do_weld()** hat die Aufgabe die SPOTWELDGUN zu schließen und wieder zu öffnen.

Diese Kinematik wird achsspezifisch verfahren und hat keine inverse kinematische Lösung. Die folgenden ERC Befehle können alternativ auch in der Mimic-Datei mit dem KeyWord SPOT realisiert wenn. Der Befehl **mimic spot** lässt sich auch als Einzelsatzbefehl ausführen.

ERC CURRENT DEVICE SET SPOTWELDGUN

aktiviert die Spotweldgun

PTP_AX 0.015 fährt die erste Achse der S-Gun um 15mm, anschließend wieder auf 0mm.

ERC CURRENT DEVICE **UNSET**

Reaktiviert die **zuvor** aktive Kinematik, in dem Beispiel ist das der DB500.

Wichtig:

aktivieren Sie nicht den DB500 mit
ERC CURRENT DEVICE SET DB500
sonst ist es nicht möglich die Funktion
do_weld() für einen zweiten Roboter wie
DB500#2 zu verwenden. Den hier müsste
es plötzlich
ERC CURRENT DEVICE SET DB500#2
heissen!

Die Funktion **release_gun()** hat die Aufgabe den DB500 zum Greifpunkt T_2 des Pfades „path01“ zu fahren und die Kinematik „SpotweldGun“ abzulegen. Der TCP wird wieder an den Flansch des DB500 gesetzt.

Neue ERPL- und ERCL Befehle

Siehe auch "EASY-ROB-CMD.txt"

ERPL-, ERCL-Syntax	Beispiel
CALC 'math expression'	calc a= 0.5 calc b= a * sin (45*RAD) LIN_REL 0 0 0.1*a 0 0 0
CALC SHOW_VARS	Zeigt alle Variablen im Message Window an. Variablen mit dem Prefix \$ sind Systemvariablen
CALC SHOW_USER_VARS	Zeigt alle User-Variablen (ohne Prefix \$) im Message Window an.
TOOL DEVICE 'robotname'	Setzt den TCP der aktuellen Kinematik an die Position des TCP's der Kinematik die mit dem Namen 'robotname' bestimmt ist.
ERC VIEW_CHOREOGRAPHY ON,OFF	aktiviert / deaktiviert View Choreography ERC LOAD View Befehle werden unterdrückt.
ERC KUD row col value	Setzt den Wert der kinematischen Userdaten via Zeilen- und Spaltenangabe ERC KUD 1 1 1 ERC KUD 12 12 144
ERC KUD Name value	ERC KUD KUD_1_1 1 ERC KUD KUD_12_12 144
ERC SET_PARAMETER \$NAME_1 'name'	ERC SET_PARAMETER \$NAME_1 T_1 ERC SET_PARAMETER \$NAME_100 MYROBOT ERC CURRENT DEVICE SET \$NAME_100 LIN \$NAME_1
ERC SET_PARAMETER \$NAME_RESET	Löscht die Inhalte aller \$NAME-Variablen
ERC SET_PARAMETER \$NAME_INFO	Zeigt die Inhalte aller \$NAME-Variablen im Message Window an.
ERC CURRENT_DEVICE SET 'robotname'	Aktiviert die Kinematik 'robotname' Alle folgenden ERPL- und ERCL-Befehle beziehen sich auf die aktive Kinematik.
ERC CURRENT_DEVICE UNSET	Setzt die zuvor aktive Kinematik aktiv
ERC CURRENT_DEVICE CLEAR	Löscht den Kinematik-Stack
ERC CURRENT_DEVICE SHOW	Zeigt alle Kinematiken auf dem Kinematik-Stack
ERC GRAB DEVICE 'robotname'	Die aktuelle Kinematik greift die Kinematik mit dem Namen 'robotname'
ERC RELEASE DEVICE 'robotname'	Die aktuelle Kinematik löst sich von der Kinematik mit dem Namen 'robotname'

Neue ERPL- und ERCL Befehle

ERPL-, ERCL-Syntax	Beispiel
ERC COLLISION BODY [ROBOT,TOOL] bodyname OFF, ON=CONVEX, CONCAVE, CONVEX, BBOX	Setzt das Attribut für die Kollisionsprüfung der Geometrie mit dem Namen ‚bodyname‘ aus den Gruppen BODY, ROBOT oder TOOL auf OFF, ON=CONVEX, CONCAVE oder BBOX. ERC Collision Robot Part_1 Concave
ERC COLLISION BODY_GRP [ROBOT_GRP,TOOL_GRP] OFF, ON=CONVEX, CONCAVE, CONVEX, BBOX	Setzt das Attribut für die Kollisionsprüfung für alle Geometrien der Gruppe BODY_GRP, ROBT_GRP oder TOOL_GRP auf OFF, ON=CONVEX, CONCAVE oder BBOX
ERC COLLISION QUEUE BODY_ROBOT ON,OFF	aktiviert / deaktiviert die Kollisionsprüfung zwischen Gruppe BODY vs. ROBOT
ERC COLLISION QUEUE BODY_TOOL ON,OFF	aktiviert / deaktiviert die Kollisionsprüfung zwischen Gruppe BODY vs. Tool
ERC COLLISION QUEUE ROBOT_TOOL ON,OFF	aktiviert / deaktiviert die Kollisionsprüfung zwischen Gruppe Robot vs. Tool
ERC COLLISION QUEUE GRABBODY_ROBOT ON,OFF	aktiviert / deaktiviert die Kollisionsprüfung zwischen Gruppe grabbed BODY vs. ROBOT
ERC COLLISION QUEUE GRABBODY_BODY ON,OFF	aktiviert / deaktiviert die Kollisionsprüfung zwischen Gruppe grabbed BODY vs. Ungrabbed BODY
ERC COLLISION QUEUE ROBOT_ROBOT ON,OFF	aktiviert / deaktiviert die Kollisionsprüfung zwischen den Geometrien innerhalb einer ROBOT_GRP
ERC COLLISION QUEUE BODY_BODY ON,OFF	aktiviert / deaktiviert die Kollisionsprüfung zwischen den Geometrien innerhalb einer BODY_GRP
ERC COLLISION QUEUE ALL ON,OFF	aktiviert / deaktiviert Kollisionsprüfung zwischen allen Gruppen
ERC COLLISION DISTANCE x [mm]	Kollisionsschwellwert, Kollision wird erkannt, wenn der minimale Kollisionsabstand zweier Geometrien den Wert x unterschreitet.
ERC DISPLAY_ROBOT_COORSYS ON,OFF	aktiviert / deaktiviert die Visualisierung der Koordinatensysteme aller Roboterkinematiken
ERC DISPLAY_ROBOT_NAME ON,OFF	aktiviert / deaktiviert die Visualisierung der Namen aller Roboterkinematiken
ERC RED_BLUE_COLLISION ON, OFF	aktiviert / deaktiviert Red/Blue Kollision. Geometrien die kollidieren sind rot eingefärbt, nicht kollidierende Geometrien blau.
ERC LOAD TAGS ‘tag file name’	Lädt ein Tag-file ‘*.tag’
ERC LOAD MIMIC ‘mimic file name’	Lädt ein mimic file ‘*.mmc’
ERC LOAD CAMERA ‘camera file name’	Lädt ein camera file ‘*.cam’
ERC PAUSE	Hält die Simulation an