Tutorial

EASY-ROB™ V8.6

August 2022

Version 3.5
EASY-ROB™

Table of content

Table of content........................................................................................................................................3
1 Your guide..................................................................................................................................................5
  1.1 Product Overview..............................................................................................................................5
1 Your guide................................................................................................................................................6
  1.2 EASY-ROB™ Multi-Robot Version ...................................................................................................6
1 Your guide................................................................................................................................................9
  1.3 EASY-ROB™ Options .........................................................................................................................9
1 Your guide.................................................................................................................................................10
  1.4 EASY-ROB™ API................................................................................................................................10
1 Your guide...............................................................................................................................................12
  1.5 EASY-ROB™ Remote Support ..........................................................................................................12
2 Basics.....................................................................................................................................................13
  2.1 Terms..................................................................................................................................................13
  2.2 Mouse Navigation ...............................................................................................................................15
3 Setting up a work cell ................................................................................................................................16
  3.1 Load and place existing components ...............................................................................................16
  3.2 Creating paths and tag points ...........................................................................................................18
  3.3 Creating a program .............................................................................................................................21
  3.4 Practice ...............................................................................................................................................24
4 Creating components ...............................................................................................................................25
  4.1 Creating simple geometry in EASY-ROB™ ......................................................................................25
  4.2 Import geometry from VRML.............................................................................................................28
5 Extend a work cell ....................................................................................................................................30
  5.1 Add components ................................................................................................................................30
  5.2 Creating paths and tag points at devices ...........................................................................................34
  5.3 Creating a program (welding simulation) .........................................................................................41
  5.4 Practice ...............................................................................................................................................46
6 Setup kinematics ......................................................................................................................................47
  6.1 Creating a positioner with simple kinematics ....................................................................................47
  6.2 Creating a robot with kinematics .......................................................................................................50
7 Creating a Multi-Robot Work cell ...........................................................................................................53
  7.1 Creating the programs .........................................................................................................................53
  7.2 Connecting the programs with signals ...............................................................................................59
8 Appendix ................................................................................................................................................62
  8.1 Most important dialogues ..................................................................................................................62
9 Contact ..................................................................................................................................................66
10 Notes ......................................................................................................................................................67
EASY-ROB™

1 Your guide

1.1 Product Overview

An overview of the most important EASY-ROB™-documentations

This tutorial is intended to explain and to show you the most fundamental EASY-ROB™-functionalities, such as creating and loading a work cell with its associated simulation program. It contains lessons with step-by-step examples, which should be exercised carefully. Regardless of this tutorial you should become familiar with the 3D-simulation software in advance. To help you, you have access to a lot of important and detailed documentations. This guide shows you, which documentations are available.

There are 3 ways to access these documentations:

1) Online:

2) In the CD-directory under Manual

3) In the Installation-directory under Manual

1. Product Description

Are you interested in other EASY-ROB™ products? Or do you just want to get information about the features of the single versions?

Take a look into the product description

>> EASY-ROB-ProductDescription.pdf
# EASY-ROB™

1. **Your guide**

## 1.2 EASY-ROB™ Multi-Robot Version

1. **Tutorial**

This tutorial is intended to explain and implement the most fundamental EASY-ROB™ functionalities. In a step-by-step instruction with a reference to examples you will learn here how to use EASY-ROB™.

This guide is part of the TrainLib-Tutorial.

>> TrainLib-Tutorial ENG.pdf

2. **Operation References**

The Operation References will be a helpful assistant while working with EASY-ROB™. It contains detailed basic information which is required for to use all software features from beginning in proper way.

The user will find information about the system files for the start configuration and the environment, description about short keys or the main dialogs as well as short tutorials

>> Operation-References.pdf

3. **Operation References – Special Functions & Plug-Ins**

The Operation References – Special Functions & Plug-Ins are an extension to the Operation References. They contain detailed information to the EASY-ROB™-Plug-Ins as well as the special functions.

>> Operation-References-Functions-PlugIns.pdf

4. **ERPL- / ERCL-Programming Language**

This description gives an overview about the EASY-ROB™ program structure and the available robot motion commands

>> EASY-ROB-ERPL ENG.pdf

---

**EASY-ROB™ Multi-Robot Version**
5. **Update-Descriptions**

   All new features of an updated EASY-ROB™ version are documented and described in the associated update descriptions.

   Use the Update descriptions to gain an overview of the new features of current or older versions and to familiarize yourself with their function.

   >> Update-ER_v8003-2019 ENG.pdf

6. **EASY-ROB™ Short Keys**

   The EASY-ROB™ user interface can be operated by using the mouse and or the keyboard by so-called short keys. This document provides an overview of all buttons and short keys and describes their function.

   >> EASY-ROB-ShortKeys ENG.pdf

7. **EASY-ROB™ Dialogues**

   The most important EASY-ROB-dialogues as overview and how to open them.

   >> EASY-ROB-Dialogues.pdf

8. **Adding a 7th tracking axis**

   Learn how to add a 7th tracking axis.

   >> Tracking-Axis.pdf
9. ERC Command Searcher

The ERC Command Searcher will support the user to search for a specific ERC commands in the example library "Proj_example_erpl", which is normally installed in the directory:
```
:\EASY-ROB\Tutorial\Proj_example_erpl
```

This description shows how to use the ERC Command Searcher.

>> ERC-Command-Searcher_ENG.pdf

10. Index

The index makes it easy to search for terms and to seek the corresponding detailed explanations in the operation references, update descriptions of older versions and eLearning modules.

>> EASY-ROB-Index.pdf

11. Glossar EASY-ROB™ eLearning

All keywords and terms that are used in EASY-ROB™ are explained in the glossary. They are also provided with a reference to the corresponding e-learning module.

>> Glossar_DE_ENG_CHS.pdf

12. eLearning platform

In cooperation with the Handwerkskammer Münster an eLearning platform with a total number of 25 modules for EASY-ROB™ has been created. This feature is only available in german language.

You can access the e-learning modules from the customer area of our website at https://www.easy-rob.com/en/service/training.html. As part of your EASY-ROB™ software maintenance and support contract, you can use this platform free of charge.

>> https://www.hbz-online.de/login/index.php

13. Autodesk® Inventor - Tutorial: Exporting STEP-files into STL-format

This documentation shows step-by-step how to convert a STEP-file (*.stp) into a STL-file (optionally with colours) using Autodesk® Inventor 2013. This STL-file can be opened directly with EASY-ROB™.

>> Autodesk-STEP2STL-Tutorial_ENG.pdf
1. **Conversion of NC-code into a robot program**

This description shows you how to create a path with tagpoints out of a NC-code and how to convert it into a manufacturer-specific robot program, by using EASY-ROB™.

>> EASY-ROB-NC-Import-Description.pdf

2. **EASY-ROB™ Remote Control**

The EASY-ROB™ remote control feature with TCP sockets allows it to interchange data with other programs, such as Robot or NC machine controller. An Application Program Interface (API) is provided with the Dynamic link library (DII) “er_remote.dll”.

This document describes the function and how to use the remote control feature.

>> Remote-Description-Ext_ENG.pdf
### 1.4 EASY-ROB™ API

#### 1. API-Description

The Application Program Interface (API) provides the user the ability to develop own functions written in C programming language for user defined inverse kinematics, motion planning and dynamics.

**CD-directory:** `:\Manual\API\API-Description.pdf`

**Installation-directory:** `:\Manual\API-Description.pdf`

#### 2. API-Sensor: Programmable Sensor interface

The sensor interface allows the user to connect an external device - like a SpaceMouse or a digitizer to capture axis angle and cartesian positions - and visualize the data direct in EASY-ROB™.

**CD-directory:** `:\Product-Info\API-Sensor_DE.pdf`

**Installation-directory:** `:\Manual\API-Sensor_ENG.pdf`

#### 3. API-UserDll

The optional API-UserDll is an enhancement of the EASY-ROB™ programming interface and enables you to create user specific dialogs that can be loaded automatically at the startup of EASY-ROB™. By using the API UserDll EASY-ROB™ can easily be customized to fit the customer’s needs.

**CD-directory:** `:\Product-Info\API-UserDll_ENG.pdf`

**Installation-directory:** `:\Manual\API-UserDll_ENG.pdf`
4. API - Doxygen documentation of methods class ER_CAPI

For the control of EASY-ROB™ and for individual product customization the methods class ER_CAPI provides a variety of functions that are documented with doxygen.

The documentation is available as Browser-version (index.html) and as compiled HTML-file (*.chm).


CD-directory: \Manual\API\EASY-ROB ER_CAPI.chm or \Manual\doxygen_ER_CAPI\html\index.html

Installation-directory: \Manual\EASY-ROB ER_CAPI.chm

5. EASY-ROB™ Kernel - Example Documentation

This documentation shows and describes examples for the „EASY-ROB™ Kernel“.


CD-directory: \Manual\API\ERK-Example-Documentation.pdf

Installation-directory: \Manual\ ERK-Example-Documentation.pdf

6. API - Doxygen Dokumentation der Klasse ERK_CAPI

For the integration of the EASY-ROB™ Kernel in your own technology-based software solutions the methods class ERK_CAPI provides a variety of functions that are documented with doxygen.

The documentation is available as Browser-version (index.html) and as compiled HTML-file (*.chm).


CD-directory: \Manual\API\ EASY-ROB Kernel.chm or \Manual\doxygen_ERK_CAPI\html\index.html

Installation-directory: \Manual\EASY-ROB Kernel.chm
1. Remote Support with TeamViewer

In order to support our customers much better, we provide you with the EASY-ROB™ TeamViewer Quick Support. A lot of problems can be solved much faster, by joining and watching either your or our EASY-ROB™ session.

Online: [https://easy-rob.com/en/contact/teamviewer/](https://easy-rob.com/en/contact/teamviewer/)

CD-directory: \TeamViewer\TeamViewerQSen.exe

Installation-directory: \TeamViewer\TeamViewerQSen.exe
2 Basics

2.1 Terms

Target of the chapter:
Introduction of the terms and the environment in EASY-ROB™

The screen consist of the following components:
- Menu
- Toolbars
- World

The Menu and Toolbars are providing different functions.

Toolbars can be placed everywhere in the world.

Cell file or Work cell (*.cel)
The Work cell contains all into simulation involved devices (e.g. robots, tools, tables,...) and the programs.

Robot/Device file (*.rob)
A robot file is device kinematics – e.g. a robot, a weld gun or a positioner.

Robot/Device Assembly file (*.ras)
Robot assembly consisting of several robots/devices, that can be linked together.

Tool file (*.tol)
A simple tool - TCP be set automatically
### Terms

<table>
<thead>
<tr>
<th>File Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body file (*.bod)</td>
<td>A simple geometry without kinematics</td>
</tr>
<tr>
<td>Program file (*.prg, *.nc)</td>
<td>A program in EASY-ROB-language ERPL and ERCL for „Robot files“</td>
</tr>
<tr>
<td>Mimic file (*.mmc)</td>
<td>A Machine-Interface file, containing ERPL and ERCL commands</td>
</tr>
<tr>
<td>View file (*.vie)</td>
<td>File the store a position for a view, defined by the user</td>
</tr>
<tr>
<td>Camera file (*.cam)</td>
<td>A file to store the current camera settings like the focus</td>
</tr>
<tr>
<td>Tagpoint file (*.tag)</td>
<td>A file to store the tagpoints in a separate place</td>
</tr>
</tbody>
</table>

### System files

<table>
<thead>
<tr>
<th>File Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Environment file (easy-rob.env)</td>
<td>File to set the variables for the environment – such as background color, floor on/off, etc ….</td>
</tr>
<tr>
<td>Config file (config.dat)</td>
<td>System file: sets the location of the license.dat and paths for temporary and user files</td>
</tr>
<tr>
<td>License file (license.dat)</td>
<td>System file: contains your license information</td>
</tr>
<tr>
<td>Er_load (er_LoadFromLibPb_prefered.ini)</td>
<td>System file: contains preferred paths for the Device Manager</td>
</tr>
<tr>
<td>Working Pathes File (easy-rob.pth)</td>
<td>System file: sets paths for the working and geometry directories with &quot;WORKDIR =&quot; and &quot;IGPDIR =&quot;</td>
</tr>
<tr>
<td>Localization file (easy-rob-localizationx64.ini)</td>
<td>System file: sets the language of the GUI (English / German / Chinese)</td>
</tr>
</tbody>
</table>
### 2.2 Mouse Navigation

**Cruise, Zoom and moving the world**

„Modify World view“ – activate by click

<table>
<thead>
<tr>
<th>Mode</th>
<th>Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cruise Mode</strong></td>
<td>Press and hold Left Mouse Button (LMB) and move the cursor to the <strong>left</strong> respectively to the <strong>right</strong> -&gt; the world will rotate around Z-axis. Press and hold Left Mouse Button (LMB) and move the cursor <strong>up</strong> or <strong>down</strong> -&gt; the world will rotate around the X-axis</td>
</tr>
<tr>
<td><strong>Zoom Mode</strong></td>
<td>Press Middle Mouse Button (MMB) [or RMB if you are using a 2button mouse] and move the cursor to the <strong>up-right</strong> corner -&gt; Zoom-In, Press Middle Mouse Button (MMB) [or RMB if you are using a 2button mouse] and move the cursor to the <strong>down-left</strong> corner -&gt; Zoom-Out.</td>
</tr>
<tr>
<td><strong>Pan Mode</strong></td>
<td>Press Left und Right Mouse Button (LMR+RMB) at the same time and move the cursor -&gt; the whole world will move</td>
</tr>
</tbody>
</table>
### Target of this chapter:
How to load and place an existing device from the library into the work cell.

### Note:
Please read as well the Operation-References, topic Kinematics Window and Frame Dialog.

<table>
<thead>
<tr>
<th>Step</th>
<th>Task</th>
</tr>
</thead>
</table>
| 1.   | Load a robot into the world by using the button Starte Device Manager  
      Directory: ../EASY-ROB/TRAInlib/...  
      Robot: AX-V6.rob  
      or shortcut „Ctrl+Shift+O“ |
| 2.   | Open the Kinematics Window by double click the button Robot Base |
| 3.   | Double click the headword Base to open the Frame Dialog |
| 4.   | Place the robot in X-direction -500 by inserting the numbers  
      or  
      by buttons.  
      The “delta” you can adjust in the field „dist“.  
      Robot Base Position  
      \[x: 500,000 \]  
      Rx  
      -dz | +dz  
      -dy | +dy  
      dist: 100,000 dR: 90,000 |
**Load and place existing components**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>5.</td>
<td>Confirm with OK.</td>
</tr>
<tr>
<td>6.</td>
<td>Save the new position as “Start Condition”</td>
</tr>
<tr>
<td>7.</td>
<td>Save the work cell – use the button <em>Save Cell File</em> or shortcut</td>
</tr>
</tbody>
</table>

Answer the question „Reset all Positions and Joints to Start Condition“ with Yes

Save the work cell with the name *tutorial_01.cell* in the directory: ../EASY-ROB / TRAINlib /

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>8.</td>
<td>Clear all by shortcut: „Ctrl+U“</td>
</tr>
</tbody>
</table>

Answer the question „Unload Cell“ with Yes
EASY-ROB™
Setting up a work cell

3.2 Creating paths and tag points

Target of the chapter:
How to create a path with tag points and how to move along the path with different motion types.

Note:
Please read as well the Operation-References, topic Tag Window

1. Load the work cell tutorial_01.cel by using the button Start Device Manager from directory: ../EASY-ROB / TRAINlib / or shortcut „Ctrl+Shift+O“

2. Save the work cell by using the button Save Cell File with the name tutorial_02.cel in directory: ../EASY-ROB / TRAINlib / or shortcut „Ctrl+S“

3. Move the robot to home position by using the button Home

4. Open the Tag Window – click on button Sel Tag

5. Information:
EASY-ROB always creates automatically one default path with the name PATH01. This path will be located in world zero and has to be assigned (or attached) to a device – e.g. the workpiece.

In this case we attach the path to robots base

Re-Attach the path PATH01 to the robot base

Attach cPath 'PATH01' to
1 - World
2 - cRobot 'AX-V6' Base
3 - cRobot 'AX-V6' Tip
Creating paths and tag points

Answer the question to “keep the world position” with  No

6. Create a new tagpoint at the TCP of the robot with button New TCP Tag

Information:
- the new point will get the default name T_1 (but you can change that)
- the new tagpoint will be assigned automatically to the default path
- and will be located on the TCP of the robot

7. Double click on  Tag Position  to open the Frame Dialog

and move the tag point to Y = -300 by inserting the coordinates

or use the buttons.

8. Repeat step 6 and 7 for three more tag points and place them like in the table:

<table>
<thead>
<tr>
<th>TAG</th>
<th>x</th>
<th>y</th>
<th>z</th>
</tr>
</thead>
<tbody>
<tr>
<td>T_2</td>
<td>810</td>
<td>300</td>
<td>1035</td>
</tr>
<tr>
<td>T_3</td>
<td>810</td>
<td>300</td>
<td>800</td>
</tr>
<tr>
<td>T_4</td>
<td>810</td>
<td>-300</td>
<td>800</td>
</tr>
</tbody>
</table>
Creating paths and tag points

9. Double click onto the tag point name and the robot will move to the location

or use buttons next and prev. to select the tagpoint and move by using the button Move cTag

10. Move to Tag T_1

Switch on the TCP-Trace with button:

Change the Motion Type to PTP

Move to every tag point - one after the other – and at the end move again to T_1

Change the Motion Type to LIN

and move again to every tag point

11. To move along all tag points just hold „Shift“, select all tag points and use the *Move Along-button*

Move the robot to home position by clicking on HOME

12. Save the cell with the name tutorial_02.cel in directory: ../EASY-ROB / TRAINlib /
(Answer the question „Reset all Positions and Joints to Start Condition“ with No)
### Setting up a work cell

#### 3.3 Creating a program

**Target of the chapter:**
How to create a program with all needed information like speed, motion type, etc.

**Note:**
Please read as well the Operation-References, topic *Teach Window*

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Load the work cell <code>tutorial_02.cel</code> by using the button <em>Start Device Manager</em> from directory: <code>../EASY-ROB / TRAINlib /</code> or shortcut „Ctrl+Shift+O“</td>
</tr>
<tr>
<td>2.</td>
<td>Save the cell with <em>Save Cell File</em>, name: <code>tutorial_03.cel</code> in directory: <code>../EASY-ROB / TRAINlib /</code> or shortcut „Ctrl+S“</td>
</tr>
<tr>
<td>3.</td>
<td>Open the <em>Teach Window</em> by clicking on the button <em>Open Program Teach Window</em></td>
</tr>
<tr>
<td>4.</td>
<td>Create a new program with the button <em>New</em> and save the program with <em>Save as</em> and the suggested name „tutorial_03-AX-V6.prg“ in directory: <code>../EASY-ROB / TRAINlib /</code> answer the question to get the program loaded with YES</td>
</tr>
<tr>
<td>5.</td>
<td>Open the program line window and set the cursor to line 10</td>
</tr>
</tbody>
</table>
### Creating a program

6. **Click > Control** to open the panel for setting **Control Commands**

   set **Default SPEEDs and ACCELs** for speeds and acceleration

   **Note:**
   To initialize the speed and acceleration it is very important to get correct results in a simulation.

7. **First insert the “Home”-Command to let the robot move to the home position.**

8. **Open Tag Window** by double click on the button **Sel Tag**

9. **Select the first tag point T_1** from the list in the **Tag Window**

10. **In the Teach Window click on PTP to insert the PTP-command into the program.**

   Repeat step 9. and 10. for every tag point and at the end one more time for tag T_1

   **Move the robot to the home position**

   Now your program should look like this:

11. **Save the program with Save**

   and reload the program into the robot by clicking on **Reload**
## Creating a program

### 12. Start the simulation with the button Run Program

Start the simulation with the button Run Program or shortcut „Ctrl+R”

### 13. Place the cursor after the second Home1 command and extend the program like shown:

Place the cursor after the second Home1 command and extend the program like shown:

1. Open the Tag Window with double click on button Sel Tag
2. Select the 2. tagpoint T_2 in the list Tag Window

### 14. Click on LIN in the Teach Window to insert the LIN-Command into the program.

Click on LIN in the Teach Window to insert the LIN-Command into the program.

1. Repeat the step for every tag point and at the end one more time for tagpoint T_1

Now your program should look like this:

![Program View](image)

### 15. Save the program with Save

Save the program with Save and reload the program into the robot

### 16. Switch the trace on and start the simulation

Switch the trace on and start the simulation

### 17. Save the cell as tutorial_03.cel

Save the cell as tutorial_03.cel in directory: ../EASY-ROB / TRAINlib /
3.4 Practice

Practice:
Create a program that enables the robot to “draw with the trace” the figure shown below.

Remark:
It's not allowed to draw a line 2 times and not allowed to “jump” to another position.

It has to be done in one go.
**EASY-ROB™**

4 Creating components

### 4.1 Creating simple geometry in EASY-ROB™

**Target of the chapter:**
How to create a simple geometry in EASY-ROB™.

**Remark:**
EASY-ROB™ is a tool to plan and simulate work cells with robots or other kinematics. It’s not a 3D-CAD-System. Therefore EASY-ROB™ only provides functions to create simple geometries like cubes, cylinder, cones and spheres. More complex parts should be generated in a 3D-CAD-program and imported to EASY-ROB™ afterwards.

**Note:**
Please read as well the Operation-References, topic 3D-CAD Window

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1.   | Create a new device (Rob-file) :
Robotics | cRobot Kinematics | Create new Robot | Universal Coordinates (1-12 dof) |
Answer the question to create simple geometry with a click on No |
| 2.   | Double click on the button 3D-CAD Window to open the 3D-CAD Window and select Create Import to get the creation dialog |
| 3.   | Create a cylinder with the numbers shown below |

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Radius</td>
</tr>
<tr>
<td>2</td>
<td>Height 1</td>
</tr>
<tr>
<td>3</td>
<td>y Scal</td>
</tr>
<tr>
<td>4</td>
<td>Radius top</td>
</tr>
<tr>
<td>5</td>
<td>Height 2</td>
</tr>
<tr>
<td>6</td>
<td>y Scal top</td>
</tr>
<tr>
<td></td>
<td>1.000000 &lt;= 200.0000</td>
</tr>
<tr>
<td></td>
<td>1.000000 &lt;= 20.0000</td>
</tr>
<tr>
<td></td>
<td>0.010000 &lt;= 1.000000</td>
</tr>
<tr>
<td></td>
<td>1.000000 &lt;= 200.0000</td>
</tr>
<tr>
<td></td>
<td>0.000000 &lt;= 20.0000</td>
</tr>
<tr>
<td></td>
<td>0.010000 &lt;= 1.000000</td>
</tr>
</tbody>
</table>

and save it with name: r_stat_00
Creating simple geometry in EASY-ROB™

4. Create a second cylinder:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Radius</td>
<td>1.000000 &lt;= 50.0000</td>
</tr>
<tr>
<td>2</td>
<td>Height 1</td>
<td>1.000000 &lt;= 500.0000</td>
</tr>
<tr>
<td>3</td>
<td>y Scal</td>
<td>0.010000 &lt;= 1.000000</td>
</tr>
<tr>
<td>4</td>
<td>Radius top</td>
<td>1.000000 &lt;= 50.0000</td>
</tr>
<tr>
<td>5</td>
<td>Height 2</td>
<td>0.000000 &lt;= 500.0000</td>
</tr>
<tr>
<td>6</td>
<td>y Scal top</td>
<td>0.010000 &lt;= 1.000000</td>
</tr>
</tbody>
</table>

and save it with the name: *r_stat_01*

5. Create a cube:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X</td>
<td>1.000000 &lt;= 200.0000</td>
</tr>
<tr>
<td>2</td>
<td>Y</td>
<td>1.000000 &lt;= 200.0000</td>
</tr>
<tr>
<td>3</td>
<td>Height</td>
<td>1.000000 &lt;= 150.0000</td>
</tr>
</tbody>
</table>

and save it with the name: *r_stat_02*

6. Change the position of the cube:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>-100.000</td>
</tr>
<tr>
<td>Y</td>
<td>-100.000</td>
</tr>
<tr>
<td>Z</td>
<td>500.000</td>
</tr>
</tbody>
</table>

7. Save the current position of all bodies with *Save all Body positions*

Subject to change or improve without prior notice
8. Change the color of the cube: 
   double click on Color RGB in the properties in the 3D-CAD Window 
   select 1- BLUE 

Answer the question "Copy color for whole robot" with No 

9. Double click the button Robot Base to open the Kinematics Window 
   Double click on Name to open the String Dialog 
   and rename the DEVICE to „cleaningstation“ 

10. Save the DEVICE with a click on Save in Kinematics Window 
    accept the suggested name 
    and save the DEVICE under the name CLEANINGSTATION.rob 
    in directory: ../EASY-ROB / TRAINlib / 

11. Clear the work cell: „Ctrl+U“ 
    Answer the question „Unload Cell“ with Yes 
    Save cell: No
# EASY-ROB™
## Creating components

### 4.2 Import geometry from VRML

**Target of the chapter:**
How to import 3D-CAD geometry from VRML.

The imported VRML files will be stored in EASY-ROB™ as IGP-Format (*.igp). The importer creates automatically a *.rob-file besides the *.igp-file.

**Note:**
Please read as well the Operation-References, topic **CAD Import - VRML**

1. Select Menu:
   - File | Load | Import / Convert | Convert VRML1.0 /2.0 into CAD Preview
   or use Drag & Drop to import the file `workpiece_panel.wrl`
   from directory: `../EASY-ROB / TRAINlib / igp`

2. Save the file `workpiece_panel.igp`
   in the directory: `../EASY-ROB / TRAINlib / igp`

3. Don’t change the default settings and start the conversion.
   - Answer the question to load the file with Yes to view the result.
   - Answer the question if the “file” should be “merged” with Yes
   - Answer the question if the “lines” should be “skipped” with Yes
   - The importer saves automatically the *.ROB- and the *.IGP-file.
Creating components

4. Switch to Windows Explorer and open the directory
   ../EASY-ROB / TRAINlib / igp

   now move the created *.ROB-file one directory up.

Remark:
Import of other CAD formats from other CAD Systems

EASY-ROB™ uses the 3D_Evolution® API of CT Core Technology GmbH to import CAD data. The CT Kernel_IO was especially designed for CAD data import and allows importing and optimizing CAD data formats generated by different CAD-Systems.

Please read the chapter CAD Import – CT Kernel_IO in the Operation References.

Start CAD2ER as x86 Version
   \cad2er\Cad2ErExe.exe
1. Select file
   "workpiece_panel.wrl"
2. Choose from list
   "One IGP-File per assembly component"
3. Choose "Start Conversion"
   to create "workpiece_panel.igp" and "workpiece_panel.rob".

![Image of CAD2ER GUI]
5 Extend a work cell

5.1 Add components

Target of the chapter:
How to load and place additional DEVICES like a welding gun, a cleaning station, a positioner and workpiece.

Note:
Please read as well the Operation-References, topic Kinematics Window und Frame Dialog.

1. Load the work cell tutorial_03.cel
   Start Device Manager
   from directory: ../EASY-ROB/TRAINlib/
   or shortcut „Ctrl+Shift+O“

2. Save the cell with Save Cell File
   name it: tutorial_04.cel
   in directory: ../EASY-ROB/TRAINlib/
   or shortcut „Ctrl+S“

3. Double click on Robot Base to open the Kinematics Window

   Load the welding gun torch_04.rob
   from directory: ../EASY-ROB/TRAINlib/

4. Re-attach the gun with Re-Attach at robot tip

   Answer the question with No to keep the position
Add components

5. Click *Pick Robot* and select the robot by a mouse click

Open the Frame Dialog by a mouse click on *Tool* in the properties to set a new tool

Set the following values to:

- **Base Pos.** : $-500.0 \ 0.0 \ 0.0$ [m]
- **Tool data**: $0.0 \ 0.0 \ 0.0$ [mm]
- **Cn. -->** : $0.0^\circ \ 0.0^\circ \ 0.0^\circ$
- **Tcp Pos.** : $810.0 \ 0.0 \ 1035.0$

- $X \ 0.000 \ Rx \ 0.000$
- $Y \ 0.000 \ Ry \ 45.000$
- $Z \ 381.000 \ Rz \ 0.000$

6. Load the cleaning station *cleaningstation.rob* (that one you created and saved in the previous chapter) from directory: ..//EASY-ROB/TRAINlib/

Place the cleaning station by using the Frame Dialog (open the dialog by mouse click *Base*)

Set the following values:

- **Attached to** : *Reference*
- **Ref World** : $0.0 \ 0.0 \ 0.0$ [mm]
- **Nri -->** : $0^\circ \ 0^\circ \ 0^\circ$
- $X \ -200.000$
- $Y \ 900.000$

7. Save the current position of all existing devices by using "Save Start Condition"
### Add components

8. Load the positioner *drehtisch.rob*
   
   from directory: ../*EASY-ROB / TRAINlib /
   
   Open the Frame Dialog (double click on Base) to place the positioner
   
   Set the following value:

<table>
<thead>
<tr>
<th>Attached to</th>
<th>Ref. World</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base</td>
<td>[0.0 0.0 0.0 (mm)]</td>
</tr>
<tr>
<td></td>
<td>[1200.0000]</td>
</tr>
</tbody>
</table>

9. Load with button *Load* the workpiece *workpiece_panel.rob*
   
   from directory: ../*EASY-ROB / TRAINlib /

10. Attach the workpiece with button *Re-Attach* to the positioner tip
    
    Don’t keep the position in the world
    
    Place the workpiece with Frame Dialog
    
    Set the following values:

    | Attached to | Ref. World |
    |-------------|------------|
    | Base        | [0.0 0.0 0.0 (mm)] |
    |             | [-700.000 0.0 0.0] |
    |             | [695.000 0.0 0.0] |
    |             | [100.000 0.0 0.0] |
### Add components

<table>
<thead>
<tr>
<th>11. Save cell with <em>Save Cell File</em> name it <code>tutorial_04.cel</code> in the directory: <code>../EASY-ROB / TRAINlib /</code></th>
</tr>
</thead>
<tbody>
<tr>
<td>IMPORTANT: Question to &quot;Reset all Positions and Joints to Start Condition&quot; : <em>No</em></td>
</tr>
<tr>
<td>Save position as start position: <em>Yes</em></td>
</tr>
</tbody>
</table>

or shortcut „Ctrl+S“
EASY-ROB™
Extend a work cell

5.2 Creating paths and tag points at devices

Target of the chapter:
How to create paths and tag points at the workpiece.

Note:
Please read as well the Operation-References, topic Tag Window und Kinematics Window

1. Load the cell tutorial_04.cel
   Start Device Manager
   from directory: ../EASY-ROB / TRAINlib /
   or shortcut „Ctrl+Shift+O“

2. Move the robot to home position by clicking on Home

3. Open the Tag Window (double click on Sel Tag)
   Delete all existing paths by clicking on Delete
Creating paths and tag points at devices

4. Save cell with *Save Cell File* name it: *tutorial_05.cel* in directory: `../EASY-ROB / TRAINlib /` or shortcut „Ctrl+S“

5. Open the Kinematics Window (double click on *Robot Base*)
   Select robot  *(Pick Robot)*
   Open Joint Values and set the following values
   Axis 2 = -30 Grad
   Axis 5 = 60 Grad

6. Rename the path *PATH01* to *start_pos* in the *Tag Window*
   Re-Attach the part *start_pos* robots base
   Keep position: No

7. Create a new tag point with *New TCP Tag*
   Rename the tag point to *T_start_pos*

8. Create a new path with *New*
   Rename the *PATH02* to *clean_stat*
   Re-Attach the path *clean_stat* by using the option *Pick Body* to the base of cleaning station
   Keep position: No
   Create a new tag point *(New TCP Tag)*
   Rename the tag point to *T_clean_stat01*
Creating paths and tagpoints at devices

### 9. Open the Frame Dialog by clicking on Tag Position

and shift the tag point to the position:

<table>
<thead>
<tr>
<th>X</th>
<th>Rx</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Y</th>
<th>Ry</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000</td>
<td>130.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Z</th>
<th>Rz</th>
</tr>
</thead>
<tbody>
<tr>
<td>710.000</td>
<td>-90.000</td>
</tr>
</tbody>
</table>

### 10. Clone the tag point T_clean_stat01

and place the new tag point in z = 600

Rename the tag point to T_clean_stat02

### 11. Select the robot with Pick Robot in Kinematics Window

and move to the tag points by double click onto the tag point's name

### 12. Change the path and move to start position by double click onto the tag point T_start_pos

### 13. Save cell and name it tutorial_05.cel

in the directory: ../EASY-ROB / TRAINlib /

### 14. Select the positioner “drehtisch” with Pick Robot in the Kinematics Window

Open the Joint Values and set them to:

Axis 1 = -90 Grad
## Creating paths and tagpoints at devices

15. Create a new path in Tag Window
   Rename the path *PATH03 to seam*
   Re-Attach the path *seam* with option *Pick Body* to the workpiece
   Keep position: *No*

16. Select the robot (*Pick Robot* Kinematics Window)
   Create a new tag point with *New TCP Tag*
   Open the *Frame Dialog* by click on *Position*
   and shift the tag point to:

   ![Frame Dialog](image)

   - **X**: 140.000
   - **Y**: 170.000
   - **Z**: 1240.000
   - **Fx**: 140.0000
   - **Fy**: 0.0000
   - **Fz**: 90.0000

17. Move the tag point
   switch *TCP Jog* on
   Create a new tag point (*New TCP Tag*)
   Select *MouseMode Tag*
and shift the tag point with your mouse by moving it to the start position of the welding seam.
Creating paths and tagpoints at devices

<table>
<thead>
<tr>
<th>18. Create a new tag point (New TCP Tag)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select <em>MouseMode Tag</em></td>
</tr>
<tr>
<td>and shift the tag point with your mouse by moving it to the end of the welding seam.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>19. Create a new tag point with New TCP Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select <em>MouseMode Tag</em></td>
</tr>
<tr>
<td>and shift the tag point with your mouse by moving it away from the “welding seam”.</td>
</tr>
</tbody>
</table>
Creating paths and tagpoints at devices

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>20.</td>
<td>Move along all tag points</td>
</tr>
<tr>
<td>21.</td>
<td>Move robot to start position</td>
</tr>
<tr>
<td>22.</td>
<td>Select the positioner “drehtisch” with Pick Robot in Kinematics Window move the positioner to home position</td>
</tr>
<tr>
<td>23.</td>
<td>Save cell as tutorial_05.cel in the directory: ../EASY-ROB / TRAINlib / Reset all positions: No Save current positions as start positions: Yes</td>
</tr>
</tbody>
</table>
**EASY-ROB™**
Extend a work cell

### 5.3 Creating a program (welding simulation)

**Target of the chapter:**
How to create a program to simulate welding

**Note:**
Please read as well the Operation-References, topic *Teach Window*

1. Load cell `tutorial_05.cel`  
   Start Device Manager  
   from directory: `../EASY-ROB / TRAINlib /`  
   or shortcut „Ctrl+Shift+O“

2. Save cell (Save Cell File)  
   name it `tutorial_06.cel`  
   in directory: `../EASY-ROB / TRAINlib /`  
   or shortcut „Ctrl+S“

3. Select the robot (*Pick Robot*) in Kinematics Window

4. Open the *Teach Window* by a mouse click on the Button *Open Program Teach Window*

5. Create a new program (*New*)  
   and save the program with the suggested name „tutorial_06-AX-V6.prg“  
   in the directory: `../EASY-ROB / TRAINlib /`  
   Reload the program: *Yes*

6. Open the program line window and set the cursor to line 10
### Creating a program (welding simulation)

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.</td>
<td>Click &gt; Control to open the panel to set Control Commands and set Default SPEEDs and ACCELs.</td>
</tr>
<tr>
<td>8.</td>
<td>Move the robot to home position.</td>
</tr>
<tr>
<td>9.</td>
<td>Open Tag Window (double click Sel Tag)</td>
</tr>
<tr>
<td>10.</td>
<td>Select the tag point T_START_POS from path START_POS. Click on PTP in Teach Window to insert the tag point with command.</td>
</tr>
<tr>
<td>11.</td>
<td>Select the tag point T_REIN_STAT01 from path REIN_STAT. Click on PTP in Teach Window to insert the tag point with command.</td>
</tr>
<tr>
<td>12.</td>
<td>Select the tag point T_REIN_STAT02 from path REIN_STAT. Click on PTP in Teach Window to insert the tag point with command.</td>
</tr>
<tr>
<td>13.</td>
<td>Select the tag point T_REIN_STAT01 from path REIN_STAT. Click on PTP in Teach Window to insert the tag point with command.</td>
</tr>
</tbody>
</table>
Creating a program (welding simulation)

14. Save the program (Save) and reload the program into the robot

15. Run the simulation (Run Program) or shortcut „Ctrl+R“

16. Set the cursor after the last PTP-command and continue programming:

17. Deactivate the robot: click >ERCL, then Set., Unset cRobot and select UNSET

18. Activate the positioner: click >ERCL, then Set., Unset cRobot and select SET (select the positioner “DREHTISCH“)

19. Deactivate the positioner: click >ERCL, then Set., Unset cRobot and select UNSET
Creating a program (welding simulation)

20. Activate the robot: click ERC, then Set., Unset cRobot and select SET (select the robot)

21. Select the tagpoint T_1 from path SEAM in Tag Window
   click on LIN in the Teach Window to insert the tagpoint with the command

   Repeat this step for every tagpoint from path SEAM

22. Select the tagpoint T_START_POS from path START_POS
   click on PTP in the Teach Window to insert the tagpoint with the command

23. Deactivate the robot: click ERC, then Set., Unset cRobot and select UNSET

   Activate the positioner: click ERC, then Set., Unset cRobot and select SET (select “drehtisch”)
   open Joint Values and set axis 1 = 0 deg
   Select Motion, click PTP AX (to insert the axis value)

   Deactivate the positioner: click ERC, then Set., Unset cRobot and select UNSET

   Activate the robot: click ERC, then Set., Unset cRobot and select SET (select the robot)

24. Add the Home-command to the program
Creating a program (welding simulation)

<table>
<thead>
<tr>
<th>25.</th>
<th>Save the program (Save) and reload the program into the robot</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image" alt="Save" /> <img src="image" alt="Reload" /></td>
</tr>
<tr>
<td>26.</td>
<td>Save the work cell as <em>tutorial_06.cel</em> in the directory: <code>../EASY-ROB / TRAINlib /</code></td>
</tr>
<tr>
<td></td>
<td><img src="image" alt="Save" /> <img src="image" alt="Reload" /></td>
</tr>
<tr>
<td>27.</td>
<td>Run the simulation</td>
</tr>
<tr>
<td></td>
<td><img src="image" alt="Play" /></td>
</tr>
<tr>
<td>28.</td>
<td>Switch collisions control ON and run the simulation again, to check for collisions. In case of collisions modify your path.</td>
</tr>
<tr>
<td></td>
<td><img src="image" alt="Play" /> <img src="image" alt="Play" /></td>
</tr>
</tbody>
</table>
5.4 Practice

**Practice:**
Extend the program of the welding simulation and weld all 4 outer seams.

Create all tag points and make sure that the program runs without any collision.
**EASY-ROB™**

6 Setup kinematics

### 6.1 Creating a positioner with simple kinematics

**Target of the chapter:**
Creating a simple kinematics for a positioner with standard CAD components. All required properties like travel limits or home position will be set.

**Note:**
Please read as well the Operation-References, topic Kinematics Window

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1.   | Create a new device (Rob-file) :
Robotics | cRobot Kinematics | Create new Robot | Universal Coordinates (1-12 dof) |
|      | Answer for creating simple geometry: No |
| 2.   | Open kinematics Window (double click on Robot Base) |
| 3.   | Open the String Dialog (double click on Name) and change the name of the device to „tutorial_positioner“ |
| 4.   | Open the Selection Dialog to change the kinematics properties:
|      | and set the number of active joints |
|      | in this case leave the default setting to 1 |
|      | Set the direction for the axis: click on active Jnt and press OK |
|      | select rotation around Z |
Creating a positioner with simple kinematics

in Frame Dialog set the position of the joint \( z=700 \)

| \( X \)  | 0.000 |
| \( Y \)  | 0.000 |
| \( Z \)  | 700.000 |

confirm with OK and close the dialog

5. Open the 3D-CAD Window (double click 3D-CAD Window)

and select Create Import to get the Selection Dialog to create basic geometries

Create a cylinder

like shown in the table:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Radius</td>
<td>( 1.000000 \leq 200.00000 )</td>
</tr>
<tr>
<td>2</td>
<td>Height 1</td>
<td>( 1.000000 \leq 700.00000 )</td>
</tr>
<tr>
<td>3</td>
<td>( r ) Scal</td>
<td>( 0.010000 \leq 1.000000 )</td>
</tr>
<tr>
<td>4</td>
<td>Radius top</td>
<td>( 1.000000 \leq 200.00000 )</td>
</tr>
<tr>
<td>5</td>
<td>Height 2</td>
<td>( 0.000000 \leq 700.00000 )</td>
</tr>
<tr>
<td>6</td>
<td>( r ) Scal top</td>
<td>( 0.010000 \leq 1.000000 )</td>
</tr>
</tbody>
</table>

and save the part with the name tut_positioner_00

6. Create a second cylinder like shown below:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Radius</td>
<td>( 1.000000 \leq 500.00000 )</td>
</tr>
<tr>
<td>2</td>
<td>Height 1</td>
<td>( 1.000000 \leq 50.00000 )</td>
</tr>
<tr>
<td>3</td>
<td>( r ) Scal</td>
<td>( 0.010000 \leq 1.000000 )</td>
</tr>
<tr>
<td>4</td>
<td>Radius top</td>
<td>( 1.000000 \leq 500.00000 )</td>
</tr>
<tr>
<td>5</td>
<td>Height 2</td>
<td>( 0.000000 \leq 50.00000 )</td>
</tr>
<tr>
<td>6</td>
<td>( r ) Scal top</td>
<td>( 0.010000 \leq 1.000000 )</td>
</tr>
</tbody>
</table>

and save the part with the name tut_positioner_01
Creating a positioner with simple kinematics

7. Change the position of the part `tut_positioner_01` with Offset Position to Z=700

<table>
<thead>
<tr>
<th>Offsets Position</th>
<th>Offsets Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>X: 0.000</td>
<td>Y: 0.000</td>
</tr>
<tr>
<td>Z: 700.000</td>
<td></td>
</tr>
</tbody>
</table>

8. Select `Attach to Jnt` in the properties of the part `tut_positioner_01` and attach the part to Joint 1

<table>
<thead>
<tr>
<th>Attach to Jnt</th>
<th>Attach to Jnt</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

9. Open the attributes of the positioner with a click on `Attribute` in the Kinematics Window

Select `Homeposition` then select „Modify cHome position „Home1““ and confirm the question to save the current position as Home Position with Yes

change the travel limits to -360 and +360 degree

close the dialog

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>2 - Homeposition</td>
<td>3 - Modify cHome position HOME „1“</td>
</tr>
<tr>
<td>3 - SW- Travel Range</td>
<td>4 - SW+ Travel Range</td>
</tr>
</tbody>
</table>

10. Change the color of the base `tut_positioner_00` in the 3D-CAD Window to grey

<table>
<thead>
<tr>
<th>Body type</th>
<th>Cylinder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color R G B</td>
<td>255 255 255</td>
</tr>
<tr>
<td>Fender</td>
<td>Smooth</td>
</tr>
</tbody>
</table>

11. Save the positioner (Save) – confirm the suggested name

<table>
<thead>
<tr>
<th>Save</th>
</tr>
</thead>
</table>

12. Select `Robot Joints` and move the mouse (hold LMB down)

or set axis values in `Joint`

to rotate the table.
Target of the chapter:
How to create a kinematics for a robot. All required properties like travel limits or home position will be set.

Note:
Please read as well the Operation-References, topic Kinematics Window

1. Create a new device (Rob-file):
   *Robotics | cRobot Kinematics | Create new Robot | Universal Coordinates (1-12 dof)*
   Answer for creating simple geometries: No

2. Open kinematics Window (double click on Robot Base)

3. Open the String Dialog (double click on Name) and change the name of the device to "tutorial_robot"

4. Open the Selection Dialog to change the kinematics properties: click on Kinematics and set the number of active joints set axis to 6
Creating a robot with kinematics

Set the direction and the positions for the axis:

<table>
<thead>
<tr>
<th>Joint</th>
<th>Axis</th>
<th>Geometry</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>rot</td>
<td>Z = 300</td>
</tr>
<tr>
<td>3</td>
<td>rot</td>
<td>all values = 0</td>
</tr>
<tr>
<td>4</td>
<td>rot</td>
<td>Z = 400</td>
</tr>
<tr>
<td>5</td>
<td>rot</td>
<td>all values = 0</td>
</tr>
<tr>
<td>6</td>
<td>rot</td>
<td>Z = 100</td>
</tr>
<tr>
<td>1</td>
<td>rot</td>
<td>Z = 500</td>
</tr>
</tbody>
</table>

5. Assign the kinematics

5. Assign the kinematics

6. Select:

Robotics | cRobot Kinematics | Create new Robot | Create Simple Geometries for cRobot

and create simple geometry for the robot

7. Jog the robot by mouse either by using TCP TOOL or by joints (Robot Joints) to different positions

Try as well the home position

8. Jog the robot by Joint to this position:

<table>
<thead>
<tr>
<th>Joint</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>50</td>
</tr>
<tr>
<td>3</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>55</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
</tr>
</tbody>
</table>
## Creating a robot with kinematics

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.</td>
<td>Save the current position with <em>Attributes</em> as home position</td>
</tr>
<tr>
<td>10.</td>
<td>Change the travel limits in <em>Attributes</em> like shown below:</td>
</tr>
<tr>
<td></td>
<td>Joint 1  +/- 180</td>
</tr>
<tr>
<td></td>
<td>Joint 2  +/- 145</td>
</tr>
<tr>
<td></td>
<td>Joint 3  +/- 140</td>
</tr>
<tr>
<td></td>
<td>Joint 4  +/- 180</td>
</tr>
<tr>
<td></td>
<td>Joint 5  +/- 130</td>
</tr>
<tr>
<td></td>
<td>Joint 6  +/- 180</td>
</tr>
<tr>
<td>11.</td>
<td>Save the robot (<em>Save</em>) under suggested name</td>
</tr>
</tbody>
</table>
# EASY-ROB™
## 7 Creating a Multi-Robot Work cell

### 7.1 Creating the programs

**Target of the chapter:**
To build up a simulation with two communicating devices (each one has a single program) by using digital signals. It is a robot and a conveyor.
The robot will place the workpiece (blue box) onto the conveyor, which will move the workpiece forward and backward. At the origin the robot will take the workpiece and put it back to the starting place.
An existing work cell load from the library will be used for this exercise.

**Note:**
The option „Multi-Robot“ is required for this exercise!

| 1. | Load the work cell tutorial_multi_program_01.cel by using the button Start Device Manager from directory: ../EASY-ROB / TRAINlib / | ![Folder](#) or shortcut „Ctrl+Shift+O“ |
| 2. | Save the work cell by using the button Save Cell File with the name tutorial_multi_program_02.cel in: ../EASY-ROB / TRAINlib / | ![Save](#) or shortcut „Ctrl+S“ |
| 3. | Select the robot (pick robot) and open the Teach Window by a mouse click on the button Open Program Teach Window | ![Teach Window](#) |
| 4. | Create a new program by using the button New and save the program with Save as under the suggested name „tutorial_multi_program_01-ER431.prg“ in the directory: ../EASY-ROB / TRAINlib / | ![New](#) ![Save as](#) |

Answer the following question (reload the program) with “Yes”. 

Subject to change or improve without prior notice
### Creating the programs

5. Open the line number window and set the cursor to line 10

6. Open the panel for the Control Commands by using the button > Control and set the standard speeds and accelerations by using the button “Set Default SPEEDs and ACCELs”

**Remark:**
This step is very important – the initialization of the speeds and accelerations is the base for proper simulation results.

7. At the beginning the robot has to be moved to the Home position. Add the Home command to the program by using the button Home

8. Open the Tag Window by double click on the button Sel Tag

9. Select the tag point T_LOAD_2 from the path LOAD_WP by double click, to make sure that the robot is moving to the tag point.

10. Click on PTP in the Teach Window to add the tag point with a PTP-command to the program

Repeat the Step 9. and 10. for the tag point: T_LOAD_1

11. Let the robot grab the workpiece:

and select the workpiece
Creating the programs

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.</td>
<td>Select the tag point T_LOAD_2 from the path LOAD_WP by <strong>double click</strong>, to make sure that the robot is moving to the tag point.</td>
</tr>
<tr>
<td>13.</td>
<td>Click on PTP in the Teach Window to add the tag point with a PTP-command to the program. Repeat the Step 9. and 10. for the tag point: T_LOAD_3, T_LOAD_4. (the workpiece should be on the conveyor now)</td>
</tr>
<tr>
<td>14.</td>
<td>Now the robot has to release the workpiece: <strong>Select</strong> the workpiece. <strong>Select</strong> EFC RELEASE Command: 1. RELEASE 'bodyname'. 2. RELEASE 'body_name'. 3. RELEASE DEVICE 'robotname'. <strong>Save</strong> the program with Save. (Reset all Positions and Joints to Start condition : Yes)</td>
</tr>
<tr>
<td>15.</td>
<td>Select the tag point T_BOX_TOP_2 from the path BOX_TOP by <strong>double click</strong>, to make sure that the robot is moving to the tag point.</td>
</tr>
<tr>
<td>16.</td>
<td>Click on PTP in the Teach Window to add the tag point with a PTP-command to the program.</td>
</tr>
<tr>
<td>17.</td>
<td>Remark: Now the robot is done for the first part and it is the turn of the conveyor. But we continue programming the second part for the robot and we will do later on the program for the conveyor.</td>
</tr>
</tbody>
</table>
## Creating the program

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
</table>
| 18.  | Add the comment to the current line of the program:  
(important: don’t forget the exclamation mark)  
! Position for signal  |
| 19.  | Select the tag point T_BOX_TOP_1 from the path BOX_TOP by  
double click  |
| 20.  | Click on PTP in the Teach Window to add the tag point with a PTP-command to the program.  |
| 21.  | Let the robot grab the workpiece.  |
| 22.  | Select the tag point T_LOAD_3 from the path LOAD_WP by  
double click  |
| 23.  | Click on PTP in the Teach Window to add the tag point with a PTP-command to the program  
Repeat step 9. and 10. for tag point:  
T_LOAD_2  
T_LOAD_1  
(the work piece should be on the floor again)  |
| 24.  | Now the robot has to release the work piece again.  |
| 25.  | Select the tagpoint T_BOX_TOP_2 from the path BOX_TOP and  
add the tag point by click onto PTP into the program.  |
| 26.  | Drive robot „HOME“ (add the Home command to the program).  |
Creating the programs

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>27.</td>
<td>Save the program with Save</td>
</tr>
<tr>
<td>28.</td>
<td>Save the work cell with the name tutorial_multi_program_02.cell in the directory: ../EASY-ROB / TRAINlib /</td>
</tr>
<tr>
<td>29.</td>
<td>Start the simulation</td>
</tr>
<tr>
<td><strong>Remark:</strong></td>
<td>Now we have to create the program for the conveyor and therefore the workpiece should be placed onto the conveyor.</td>
</tr>
<tr>
<td>30.</td>
<td>Select the conveyor in the Pull Down Menu from the Teach Window</td>
</tr>
<tr>
<td>31.</td>
<td>Create a new program by using the button New and save the program with Save as under the suggested name „tutorial_multi_program_01-CONVEYOR_BIG.prg“ in directory: ../EASY-ROB / TRAINlib /</td>
</tr>
<tr>
<td>32.</td>
<td>Open the line number window and set the cursor to line 10</td>
</tr>
</tbody>
</table>

Subject to change or improve without prior notice
Creating the programs

33. Open the panel for the **Control Commands** by clicking on >**Control**
and set the standard speeds and accelerations by using the button
“Set Default SPEEDs and ACCELs”

34. Insert the command Home into the program by using the button
**Home**

35. Now the conveyor has to grab the workpiece:
(like the robot earlier)

36. Select the tag point T_CONV_2 from the path CONV by double
**click**, to make sure that the conveyor is moving to the point.

37. Click on PTP in the Teach Window to add the tag point with a PTP-
command to the program
Repeat the Step 36. and 37. for the tagpoint:
T_CONV_1

38. The conveyor has to release the workpiece:

39. Save the program with **Save**

40. Save the work cell with the name **tutorial_multi_program_02.cel**
in the directory: ..:/EASY-ROB / TRAINlib /
## Target of the chapter:
To connect the created programs from previous chapter by using digital signals.

### Note: The option „Multi-Robot“ is required for this exercise!

1. Load the work cell `tutorial_multi_program_02.cel` by using the button *Start Device Manager* from the directory: `../EASY-ROB / TRAINlib /`

2. Select both devices by the Pull Down Menu in the Teach Window and check if the programs are loaded.
   - If the programs are not loaded, load them by using „LOAD“

3. Save the work cell with the name `tutorial_multi_program_03.cel` in the directory: `../EASY-ROB / TRAINlib /`

4. Select first the conveyor by using the Pull Down Menu from the Teach Window

5. To initialize the outgoing signal of the conveyor, set the cursor in the Teach Window to line 5 (before of the “EndInit”-command) and add the command: `conv_out=0`
### Connecting the programs

6. Place the cursor onto the command 
   "ERC GRAB DEVICE BOX_TUTORIAL" and add the 
   "Wait_Until_Signal_Set" command with the signal name 
   "rob01_out"

Then unset the signal (by editing in the command line) of the robot:

**Remark:**
The conveyor will wait until the robot will set the signal gibt and immediately after that the conveyor will unset the signal again.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; Control</td>
<td>Wait for Signal</td>
</tr>
<tr>
<td>[Set]</td>
<td>[Unset]</td>
</tr>
<tr>
<td>WAIT_UNTIL_SIGNAL_SET rob01_out</td>
<td>[rob01_out=0]</td>
</tr>
</tbody>
</table>

7. Place the cursor after the command 
   "ERC RELEASE DEVICE BOX_TUTORIAL" and add the 
   "Wait_Until_Signal_Unset" command with the signal name 
   "conv_out"

**Remark:**
The conveyor will only go on when its signal is unset.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WAIT_UNTIL_SIGNAL_UNSET conv_out</td>
<td>[conv_out=1]</td>
</tr>
</tbody>
</table>

8. Set the signal of the conveyor to "conv_out=1"

**Remark:**
The conveyor will tell the robot that the workpiece is back.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv_out=1</td>
<td>[conv_out=1]</td>
</tr>
</tbody>
</table>

9. Save the program with Save

10. Select the robot by using the Pull Down Menu from the Teach Window.

11. To initialize the outgoing signal of the robot, set the cursor in the TeachWindow to line 5 (before of the "EndInit"-command) and add the command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rob01_out=0</td>
<td>[rob01_out=0]</td>
</tr>
</tbody>
</table>
### Connecting the programs

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.</td>
<td>Set the cursor after your comment „! Position for signal“ and add the „Wait_Until_Signal_Unset“ - command with the signal name „rob01_out“</td>
<td>WAIT_UNTIL_SIGNAL_UNSET rob01_out</td>
</tr>
<tr>
<td></td>
<td>Remark: The robot will only go on when its signal is unset.</td>
<td></td>
</tr>
<tr>
<td>13.</td>
<td>Set the signal (by editing the command line) of the robot to „rob01_out=1“</td>
<td>rob01_out=1</td>
</tr>
<tr>
<td></td>
<td>Remark: The robot will tell the conveyor that the workpiece is standing onto the conveyor.</td>
<td></td>
</tr>
<tr>
<td>14.</td>
<td>Add a „Wait_Until_Signal_SET“ - command with the signal name „conv_out“</td>
<td>WAIT_UNTIL_SIGNAL_SET conv_out</td>
</tr>
<tr>
<td></td>
<td>Remark: The robot will wait until the conveyor tells that the workpiece is back and then the robot will immediately unset (reset) the signal of the conveyor.</td>
<td></td>
</tr>
<tr>
<td>15.</td>
<td>Save both programs with Save</td>
<td><img src="save_icon.png" alt="Save" /></td>
</tr>
<tr>
<td>16.</td>
<td>Save the work cell with the name <code>tutorial_multi_program_03.cel</code> in the directory: <code>../EASY-ROB / TRAINlib</code></td>
<td><img src="folder_icon.png" alt="folder_icon" /></td>
</tr>
<tr>
<td>17.</td>
<td>Start the simulation</td>
<td><img src="play_icon.png" alt="play_icon" /></td>
</tr>
<tr>
<td></td>
<td>To run the simulation in a loop, just push the loop button straight after starting the simulation.</td>
<td></td>
</tr>
</tbody>
</table>
8.1 Most important dialogues

**Menu:**
- Robotics > cRobot Program > Open Teach Window
- Robotics > cRobot Program > Open Program Output Window
- View > Open Online Output Data

**Toolbars:**
- Shows you actual executed program line
- Shows you Joint values, TCP position, motion data
Most important dialogues

**Menu:**

- File
- Robotics
- Simulation
- 3D-CAD
- Tags
- View
- Aux
- ?

**Toolbars:**

Mouse Mode Toolbar:

- TCP
- TCP BASE
- TCP WORLD
- JOINTS
- BASE
- SEL
- WORK
- OBJ
- ABS
- TCP
- JBG
- ZOOM

**Kinematics Window**

**Menu:**

Robotics > Open Kinematics Window  Ctrl+K

**Toolbar:**

Create and change kinematics

**3D-CAD Window**

**Menu:**

3D-CAD > Open 3D-CAD Window

**Toolbar:**

Create, import and change geometry

**Tag Window**

**Menu:**

Tags > Open Tag Window

**Toolbar:**

Create and Change paths and tagpoints
### Most important dialogues

**Menu:**
- File  Robotics  Simulation  3D-CAD  Tags  View  Aux  ?

**Toolbars:**

<table>
<thead>
<tr>
<th>Mouse Mode Toolbar</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP TOOL  TCP BASE  TCP WORLD  TCP ROBOT  ROBOT BASE  SEI TAG  WORK OBJ  ABC COORD  TCP JOB  ZOOM</td>
</tr>
</tbody>
</table>

### Navigator Window

**Menu:**
- View > Navigator Window  Ctrl+N

**Toolbar:**

![Navigator Window toolbar](image)

- Selecting by Pick & Click

### Jog Window

**Menu:**
- View > Open Window Dialogs > Jog Window

**Toolbar:**

![Jog Window toolbar](image)

- Jog joints

### Message Window

**Menu:**
- View > Message Window  Ctrl+M

**Toolbar:**

- Shows you important messages, status etc.
EASY-ROB™

9 Contact

EASY-ROB Software GmbH

Address: Hauptstr. 42
65719 Hofheim am Taunus
Germany

Contact: Mr. Stefan Anton

Phone: +49 6192 921 70 77
FAX: +49 6192 921 70 66

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

Url: www.easy-rob.com

EASY-ROB customer area

Content: Program updates and robot libraries


Log in data:
User name: customer
Password: **********
EASY-ROB™

10 Notes