© Copyright 2011
KUKA Roboter GmbH
Zugspitzstraße 140
D-86165 Augsburg
Germany

This documentation or excerpts therefrom may not be reproduced or disclosed to third parties without
the express permission of KUKA Roboter GmbH.

Other functions not described in this documentation may be operable in the controller. The user has
no claims to these functions, however, in the case of a replacement or service work.

We have checked the content of this documentation for conformity with the hardware and software
described. Nevertheless, discrepancies cannot be precluded, for which reason we are not able to
guarantee total conformity. The information in this documentation is checked on a regular basis, how-
ever, and necessary corrections will be incorporated in the subsequent edition.

Subject to technical alterations without an effect on the function.

Translation of the original documentation
KIM-PS5-DOC

Publication: Pub KSS 8.2 END en
Bookstructure: KSS 8.2 END V1.1
Label: KSS 8.2 END V1 en
3.8.3 Start-up and recommissioning

3.8.2 Transportation

3.8.1 General safety measures

3.7 Overview of operating modes and safety functions

3.6.8 External safeguards

3.6.7 Labeling on the industrial robot

3.6.6 Release device (optional)

3.6.5 Axis range monitoring (optional)

3.6.4 Mechanical axis range limitation (optional)

3.6.3 Mechanical end stops

3.6.2 Software limit switches

3.6.1 Jog mode

3.6.0 Overview of the safety functions

3.5.12 Velocity monitoring in T1

3.5.11 External safety stop 1 and external safety stop 2

3.5.10 External safe operational stop

3.5.9 External enabling device

3.5.8 Enabling device

3.5.7 External EMERGENCY STOP device

3.5.6 Logging off the higher-level safety controller

3.5.5 EMERGENCY STOP device

3.5.4 Operator safety

3.5.3 Mode selection

3.5.2 Safety controller

3.5.1 Overview of the safety functions

3.5 Safety functions

3.4 Triggers for stop reactions

3.3 Workspace, safety zone and danger zone

3.2 Personnel

3.1.12 Terms used

3.1.11 Terms used

3.1.10 External safe operational stop

3.1.9 External enabling device

3.1.8 Enabling device

3.1.7 External EMERGENCY STOP device

3.1.6 Logging off the higher-level safety controller

3.1.5 EMERGENCY STOP device

3.1.4 Terms used

3.1.3 EC declaration of conformity and declaration of incorporation

3.1.2 Intended use of the industrial robot

3.1.1 Liability

3.1 General

3 Safety

2.3 Overview of KUKA System Software (KSS)

2.2 Overview of the software components

2.1 Overview of the industrial robot

2 Product description

1.2 Industrial robot documentation

1.1 Target group

1 Introduction

Contents

1 Introduction ........................................................................................................ 9

1.1 Target group ................................................................................................. 9

1.2 Industrial robot documentation ................................................................... 9

1.3 Representation of warnings and notes ........................................................... 9

1.4 Trademarks ..................................................................................................... 9

2 Product description .......................................................................................... 11

2.1 Overview of the industrial robot .................................................................... 11

2.2 Overview of the software components ............................................................ 11

2.3 Overview of KUKA System Software (KSS) ................................................ 11

3 Safety ............................................................................................................. 13

3.1 General ............................................................................................................ 13

3.1.1 Liability ...................................................................................................... 13

3.1.2 Intended use of the industrial robot ............................................................ 13

3.1.3 EC declaration of conformity and declaration of incorporation ................. 14

3.1.4 Terms used ................................................................................................ 14

3.2 Personnel ........................................................................................................ 16

3.3 Workspace, safety zone and danger zone ......................................................... 17

3.4 Triggers for stop reactions .............................................................................. 18

3.5 Safety functions .............................................................................................. 19

3.5.1 Overview of the safety functions ................................................................. 19

3.5.2 Safety controller ......................................................................................... 19

3.5.3 Mode selection ........................................................................................... 20

3.5.4 Operator safety .......................................................................................... 20

3.5.5 EMERGENCY STOP device ..................................................................... 21

3.5.6 Logging off the higher-level safety controller ............................................. 21

3.5.7 External EMERGENCY STOP device ......................................................... 22

3.5.8 Enabling device ........................................................................................ 22

3.5.9 External enabling device .......................................................................... 23

3.5.10 External safe operational stop ................................................................. 23

3.5.11 External safety stop 1 and external safety stop 2 .................................... 23

3.5.12 Velocity monitoring in T1 ....................................................................... 23

3.6 Additional protective equipment .................................................................. 23

3.6.1 Jog mode .................................................................................................... 23

3.6.2 Software limit switches ............................................................................ 23

3.6.3 Mechanical end stops .............................................................................. 24

3.6.4 Mechanical axis range limitation (optional) ............................................. 24

3.6.5 Axis range monitoring (optional) ............................................................... 24

3.6.6 Release device (optional) ......................................................................... 24

3.6.7 Labeling on the industrial robot ............................................................... 25

3.6.8 External safeguards ................................................................................ 25

3.7 Overview of operating modes and safety functions ....................................... 26

3.8 Safety measures ............................................................................................ 26

3.8.1 General safety measures .......................................................................... 26

3.8.2 Transportation .......................................................................................... 28

3.8.3 Start-up and recommissioning .................................................................. 28
3.8.3.1 Start-up mode ........................................................................................................ 30
3.8.4 Manual mode ........................................................................................................... 30
3.8.5 Simulation ............................................................................................................... 31
3.8.6 Automatic mode ..................................................................................................... 31
3.8.7 Maintenance and repair .......................................................................................... 32
3.8.8 Decommissioning, storage and disposal ................................................................. 33
3.8.9 Safety measures for “single point of control” ........................................................ 33
3.9 Applied norms and regulations .................................................................................. 35

4 Operation ....................................................................................................................... 37

4.1 KUKA smartPAD teach pendant ............................................................................... 37
4.1.1 Front view .............................................................................................................. 37
4.1.2 Rear view ................................................................................................................ 39
4.1.3 Disconnecting and connecting the smartPAD ......................................................... 40
4.2 KUKA smartHMI user interface ............................................................................... 41
4.2.1 Status bar ............................................................................................................... 42
4.2.2 “Submit interpreter” status indicator .................................................................... 43
4.2.3 Keypad .................................................................................................................... 44
4.3 Switching on the robot controller and starting the KSS ........................................... 44
4.4 Calling the main menu ............................................................................................... 44
4.5 Exiting or restarting KSS ............................................................................................ 45
4.6 Start types ................................................................................................................... 47
4.7 Switching the robot controller off ............................................................................... 47
4.8 Setting the user interface language .......................................................................... 47
4.9 Changing user group .................................................................................................. 48
4.10 Disabling the robot controller .................................................................................. 48
4.11 Changing operating mode ......................................................................................... 49
4.12 Coordinate systems ................................................................................................. 49
4.13 Jogging the robot ...................................................................................................... 51
4.13.1 “Jog options” window ............................................................................................ 52
4.13.1.1 “General” tab .................................................................................................... 52
4.13.1.2 “Keys” tab ......................................................................................................... 53
4.13.1.3 “Mouse” tab ...................................................................................................... 53
4.13.1.4 “KCP pos.” tab ................................................................................................ 54
4.13.1.5 “Cur. tool/base” tab ........................................................................................ 54
4.13.2 Activating the jog mode ......................................................................................... 55
4.13.3 Setting the jog override (HOV) ............................................................................ 55
4.13.4 Selecting the tool and base .................................................................................. 56
4.13.5 Axis-specific jogging with the jog keys ................................................................. 56
4.13.6 Cartesian jogging with the jog keys ..................................................................... 56
4.13.7 Configuring the Space Mouse ............................................................................. 56
4.13.8 Defining the alignment of the Space Mouse ........................................................ 58
4.13.9 Cartesian jogging with the Space Mouse ............................................................. 59
4.13.10 Incremental jogging ............................................................................................ 60
4.14 Jogging external axes ............................................................................................... 61
4.15 Bypassing workspace monitoring ........................................................................... 61
4.16 Monitor functions ...................................................................................................... 62
4.16.1 Displaying the actual position ............................................................................ 62
4.16.2 Displaying digital inputs/outputs ......................................................................... 63
6 Program management ......................................................... 107

6.1 Navigator file manager .................................................. 107
6.1.1 Selecting filters ......................................................... 108
6.1.2 Creating a new folder .................................................. 109
6.1.3 Creating a new program .............................................. 109
6.1.4 Renaming a file .......................................................... 109
6.2 Selecting or opening a program ....................................... 109
6.2.1 Selecting and deselecting a program ......................... 110
6.2.2 Opening a program .................................................... 111
6.2.3 Toggling between the Navigator and the program ........ 111
6.3 Structure of a KRL program .......................................... 112
6.3.1 HOME position .......................................................... 113
6.4 Displaying/hiding program sections ................................. 113
6.4.1 Displaying/hiding the DEF line ................................. 113
6.4.2 Activating detail view ................................................ 113
6.4.3 Activating/deactivating the line break function ............ 114
6.5 Starting a program ....................................................... 114
6.5.1 Selecting the program run mode ................................. 114
6.5.2 Program run modes .................................................... 114
6.5.3 Advance run ............................................................. 115
6.5.4 Setting the program override (POV) ......................... 115
6.5.5 Switching drives on/off .............................................. 115
6.5.6 Robot interpreter status indicator ............................... 115
6.5.7 Starting a program forwards (manual) ...................... 116
6.5.8 Starting a program forwards (automatic) ................. 116
6.5.9 Carrying out a block selection ................................. 117
6.5.10 Starting a program backwards ................................ 117
6.5.11 Resetting a program .............................................. 118
6.5.12 Starting Automatic External mode ......................... 118
6.6 Editing a program ....................................................... 118
6.6.1 Inserting a comment or stamp .................................. 120
6.6.2 Deleting program lines ............................................. 120
6.6.3 Additional editing functions ..................................... 121
6.7 Printing a program ...................................................... 121
6.8 Archiving and restoring data ......................................... 121
6.8.1 Archiving overview .................................................. 121
6.8.2 Archiving to a USB stick ......................................... 123
6.8.3 Archiving on the network ......................................... 123
6.8.4 Archiving the logbook ............................................. 124
6.8.5 Restoring data ......................................................... 124
6.8.6 Compressing data for error analysis at KUKA ........... 124

5.6.6.6 Entering the external tool numerically ........................ 104
5.7 Load data ..................................................................... 105
5.7.1 Checking loads with KUKA.Load ............................... 105
5.7.2 Calculating payloads with KUKA.LoadDataDetermination .... 105
5.7.3 Entering payload data .............................................. 105
5.7.4 Entering supplementary load data .............................. 106
## 7 Basic principles of motion programming

- **7.1 Overview of motion types** .......................................................... 127
- **7.2 Motion type PTP** ........................................................................ 127
- **7.3 Motion type LIN** ........................................................................ 127
- **7.4 Motion type CIRC** ....................................................................... 128
- **7.5 Approximate positioning** .......................................................... 128
- **7.6 Orientation control LIN, CIRC** ............................................... 130
- **7.7 Motion type “Spline”** ................................................................. 131
  - **7.7.1 Velocity profile for spline motions** ....................................... 133
  - **7.7.2 Block selection with spline motions** .................................... 134
  - **7.7.3 Modifications to spline blocks** ........................................... 135
  - **7.7.4 Approximate positioning of spline motions** ....................... 137
  - **7.7.5 Replacing an approximated motion with a spline block** .... 138
  - **7.7.5.1 SLIN-SPL-SLIN transition** ............................................. 141
- **7.8 Orientation control SPLINE** .................................................... 141
- **7.8.1 Combinations of “Orientation control” and “Circle orientation control”** ........................ 144
- **7.9 Singularities** ........................................................................... 145

## 8 Programming for user group “User” (inline forms)

- **8.1 Names in inline forms** ............................................................... 147
- **8.2 Programming PTP, LIN and CIRC motions** ............................. 147
  - **8.2.1 Programming a PTP motion** ............................................. 147
  - **8.2.2 Inline form “PTP”** ............................................................ 148
  - **8.2.3 Programming a LIN motion** ............................................. 148
  - **8.2.4 Inline form “LIN”** ............................................................ 149
  - **8.2.5 Programming a CIRC motion** .......................................... 149
  - **8.2.6 Inline form “CIRC”** .......................................................... 150
  - **8.2.7 Option window “Frames”** ................................................ 151
  - **8.2.8 Option window “Motion parameter” (PTP)** ....................... 151
  - **8.2.9 Option window “Motion parameters” (LIN, CIRC)** ............ 152
- **8.3 Spline motions** ........................................................................ 153
  - **8.3.1 Programming tips for spline motions** ................................. 153
  - **8.3.2 Programming a SLIN motion (individual motion)** ............. 154
    - **8.3.2.1 Inline form “SLIN”** ..................................................... 154
    - **8.3.2.2 Option window “Motion parameters” (SLIN)** ............... 155
  - **8.3.3 Programming a SCIRC motion (individual motion)** .......... 156
    - **8.3.3.1 Inline form “SCIRC”** .................................................. 156
    - **8.3.3.2 Option window “Motion parameters” (SCIRC)** ............ 157
  - **8.3.4 Programming a spline block** ............................................ 158
    - **8.3.4.1 Inline form for spline block** ...................................... 159
    - **8.3.4.2 Option window “Frames” (spline block)** ..................... 159
    - **8.3.4.3 Option window “Motion parameters” (spline block)** .... 160
    - **8.3.4.4 Programming an SPL or SLIN segment** ...................... 161
    - **8.3.4.5 Programming an SCIRC segment** ............................... 161
    - **8.3.4.6 Inline form for spline segment** .................................. 162
    - **8.3.4.7 Option window “Frames” (spline segment)** ............... 163
    - **8.3.4.8 Programming triggers in the spline block** .................. 164
- **8.4 Programming an SPL or SLIN segment** .................................. 128
- **8.5 Programming a SCIRC motion (individual motion)** ............... 147
8.3.4.10 Inline form for spline trigger type “Set output” ................................................................. 165
8.3.4.11 Inline form for spline trigger type “Set pulse output” ....................................................... 166
8.3.4.12 Inline form for spline trigger type “Trigger assignment” .................................................. 167
8.3.4.13 Inline form for spline trigger type “Trigger function call” ................................................ 167
8.3.4.14 Limits for functions in the spline trigger ......................................................................... 168
8.3.5 Copying spline inline forms ................................................................................................... 169
8.3.6 Converting spline inline forms from 8.1 ............................................................................... 170
8.4 Modifying motion parameters .................................................................................................. 170
8.5 Modifying the coordinates of a taught point ............................................................................ 170
8.6 Programming logic instructions ............................................................................................... 171
8.6.1 Inputs/outputs ....................................................................................................................... 171
8.6.2 Setting a digital output - OUT ............................................................................................. 171
8.6.3 Inline form “OUT” ................................................................................................................ 171
8.6.4 Setting a pulse output - PULSE ........................................................................................... 172
8.6.5 Inline form “PULSE” ............................................................................................................ 172
8.6.6 Setting an analog output - ANOUT ...................................................................................... 172
8.6.7 Inline form “ANOUT” (static) .............................................................................................. 173
8.6.8 Inline form “ANOUT” (dynamic) .......................................................................................... 173
8.6.9 Programming a wait time - WAIT ......................................................................................... 174
8.6.10 Inline form “WAIT” ............................................................................................................. 174
8.6.11 Programming a signal-dependent wait function - WAITFOR ............................................ 174
8.6.12 Inline form “WAITFOR” ...................................................................................................... 175
8.6.13 Switching on the path - SYN OUT .................................................................................... 176
8.6.14 Inline form “SYN OUT”, option “START/END” ................................................................. 176
8.6.15 Inline form “SYN OUT”, option “PATH” ............................................................................ 179
8.6.16 Setting a pulse on the path - SYN PULSE .......................................................................... 181
8.6.17 Inline form “SYN PULSE” .................................................................................................. 181
8.6.18 Modifying a logic instruction ............................................................................................... 182
9 Messages ................................................................................................................................... 183
  9.1 Automatic External error messages ......................................................................................... 183
10 KUKA Service ........................................................................................................................ 185
  10.1 Requesting support .............................................................................................................. 185
  10.2 KUKA Customer Support .................................................................................................... 185
Index ............................................................................................................................................. 193
1 Introduction

1.1 Target group

This documentation is aimed at users with the following knowledge and skills:

- Basic knowledge of the industrial robot

For optimal use of our products, we recommend that our customers take part in a course of training at KUKA College. Information about the training program can be found at www.kuka.com or can be obtained directly from our subsidiaries.

1.2 Industrial robot documentation

The industrial robot documentation consists of the following parts:

- Documentation for the manipulator
- Documentation for the robot controller
- Operating and programming instructions for the KUKA System Software
- Documentation relating to options and accessories
- Parts catalog on storage medium

Each of these sets of instructions is a separate document.

1.3 Representation of warnings and notes

Safety

These warnings are relevant to safety and must be observed.

- **DANGER** - These warnings mean that death or severe physical injury will occur, if no precautions are taken.
- **WARNING** - These warnings mean that death or severe physical injury may occur, if no precautions are taken.
- **CAUTION** - These warnings mean that minor physical injuries may occur, if no precautions are taken.
- **NOTICE** - These warnings mean that damage to property may occur, if no precautions are taken.

These warnings contain references to safety-relevant information or general safety measures. These warnings do not refer to individual hazards or individual precautionary measures.

Hints

These hints serve to make your work easier or contain references to further information.

- **Tip to make your work easier or reference to further information.**

1.4 Trademarks

- **Windows** is a trademark of Microsoft Corporation.
- **WordPad** is a trademark of Microsoft Corporation.
2 Product description

2.1 Overview of the industrial robot

The industrial robot consists of the following components:

- Manipulator
- Robot controller
- Teach pendant
- Connecting cables
- Software
- Options, accessories

Fig. 2-1: Example of an industrial robot

1 Manipulator  3 Robot controller
2 Connecting cables  4 Teach pendant

2.2 Overview of the software components

Overview

The following software components are used:

- KUKA System Software 8.2
- Windows XPe V3.0.0

2.3 Overview of KUKA System Software (KSS)

Description

The KUKA System Software (KSS) is responsible for all the basic operator control functions of the industrial robot.

- Path planning
- I/O management
- Data and file management
- etc.
Additional technology packages, containing application-specific instructions and configurations, can be installed.

**smartHMI**

The user interface of the KUKA System Software is called KUKA smartHMI (smart Human-Machine Interface).

Features:
- User administration
- Program editor
- KRL (KUKA Robot Language)
- Inline forms for programming
- Message display
- Configuration window
- etc.

(>>> 4.2 "KUKA smartHMI user interface" Page 41)

Depending on customer-specific settings, the user interface may vary from the standard interface.
3 Safety

3.1 General

3.1.1 Liability

The device described in this document is either an industrial robot or a component thereof.

Components of the industrial robot:

- Manipulator
- Robot controller
- Teach pendant
- Connecting cables
- External axes (optional)
  - e.g. linear unit, turn-tilt table, positioner
- Software
- Options, accessories

The industrial robot is built using state-of-the-art technology and in accordance with the recognized safety rules. Nevertheless, misuse of the industrial robot may constitute a risk to life and limb or cause damage to the industrial robot and to other material property.

The industrial robot may only be used in perfect technical condition in accordance with its intended use and only by safety-conscious persons who are fully aware of the risks involved in its operation. Use of the industrial robot is subject to compliance with this document and with the declaration of incorporation supplied together with the industrial robot. Any functional disorders affecting the safety of the industrial robot must be rectified immediately.

Safety information

Safety information cannot be held against KUKA Roboter GmbH. Even if all safety instructions are followed, this is not a guarantee that the industrial robot will not cause personal injuries or material damage.

No modifications may be carried out to the industrial robot without the authorization of KUKA Roboter GmbH. Additional components (tools, software, etc.), not supplied by KUKA Roboter GmbH, may be integrated into the industrial robot. The user is liable for any damage these components may cause to the industrial robot or to other material property.

In addition to the Safety chapter, this document contains further safety instructions. These must also be observed.

3.1.2 Intended use of the industrial robot

The industrial robot is intended exclusively for the use designated in the “Purpose” chapter of the operating instructions or assembly instructions.

Further information is contained in the “Purpose” chapter of the operating instructions or assembly instructions of the industrial robot.

Using the industrial robot for any other or additional purpose is considered impermissible misuse. The manufacturer cannot be held liable for any damage resulting from such use. The risk lies entirely with the user.

Operating the industrial robot and its options within the limits of its intended use also involves observance of the operating and assembly instructions for
the individual components, with particular reference to the maintenance specifications.

**Misuse**

Any use or application deviating from the intended use is deemed to be impermissible misuse. This includes e.g.:

- Transportation of persons and animals
- Use as a climbing aid
- Operation outside the permissible operating parameters
- Use in potentially explosive environments
- Operation without additional safeguards
- Outdoor operation

3.1.3 **EC declaration of conformity and declaration of incorporation**

This industrial robot constitutes partly completed machinery as defined by the EC Machinery Directive. The industrial robot may only be put into operation if the following preconditions are met:

- The industrial robot is integrated into a complete system.
  
  Or: The industrial robot, together with other machinery, constitutes a complete system.
  
  Or: All safety functions and safeguards required for operation in the complete machine as defined by the EC Machinery Directive have been added to the industrial robot.

- The complete system complies with the EC Machinery Directive. This has been confirmed by means of an assessment of conformity.

**Declaration of conformity**

The system integrator must issue a declaration of conformity for the complete system in accordance with the Machinery Directive. The declaration of conformity forms the basis for the CE mark for the system. The industrial robot must be operated in accordance with the applicable national laws, regulations and standards.

The robot controller is CE certified under the EMC Directive and the Low Voltage Directive.

**Declaration of incorporation**

The industrial robot as partly completed machinery is supplied with a declaration of incorporation in accordance with Annex II B of the EC Machinery Directive 2006/42/EC. The assembly instructions and a list of essential requirements complied with in accordance with Annex I are integral parts of this declaration of incorporation.

The declaration of incorporation declares that the start-up of the partly completed machinery remains impermissible until the partly completed machinery has been incorporated into machinery, or has been assembled with other parts to form machinery, and this machinery complies with the terms of the EC Machinery Directive, and the EC declaration of conformity is present in accordance with Annex II A.

The declaration of incorporation, together with its annexes, remains with the system integrator as an integral part of the technical documentation of the complete machinery.

3.1.4 **Terms used**

STOP 0, STOP 1 and STOP 2 are the stop definitions according to EN 60204-1:2006.
<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axis range</td>
<td>Range of each axis, in degrees or millimeters, within which it may move. The axis range must be defined for each axis.</td>
</tr>
<tr>
<td>Stopping distance</td>
<td>Stopping distance = reaction distance + braking distance. The stopping distance is part of the danger zone.</td>
</tr>
<tr>
<td>Workspace</td>
<td>The manipulator is allowed to move within its workspace. The workspace is derived from the individual axis ranges.</td>
</tr>
<tr>
<td>Operator (User)</td>
<td>The user of the industrial robot can be the management, employer or delegated person responsible for use of the industrial robot.</td>
</tr>
<tr>
<td>Danger zone</td>
<td>The danger zone consists of the workspace and the stopping distances.</td>
</tr>
</tbody>
</table>
| KCP                         | The KCP (KUKA Control Panel) teach pendant has all the operator control and display functions required for operating and programming the industrial robot. 
The KCP variant for the KR C4 is called KUKA smartPAD. The general term “KCP”, however, is generally used in this documentation. |
| Manipulator                 | The robot arm and the associated electrical installations                                                                                      |
| Safety zone                 | The safety zone is situated outside the danger zone.                                                                                           |
| Safe operational stop       | The safe operational stop is a standstill monitoring function. It does not stop the robot motion, but monitors whether the robot axes are stationary. 
If these are moved during the safe operational stop, a safety stop STOP 0 is triggered. 
The safe operational stop can also be triggered externally. 
When a safe operational stop is triggered, the robot controller sets an output to the field bus. The output is set even if not all the axes were stationary at the time of triggering, thereby causing a safety stop STOP 0 to be triggered. |
| Safety STOP 0               | A stop that is triggered and executed by the safety controller. The safety controller immediately switches off the drives and the power supply to the brakes. 
**Note:** This stop is called safety STOP 0 in this document. |
| Safety STOP 1               | A stop that is triggered and monitored by the safety controller. The braking process is performed by the non-safety-oriented part of the robot controller and monitored by the safety controller. 
As soon as the manipulator is at a standstill, the safety controller switches off the drives and the power supply to the brakes. 
When a safety STOP 1 is triggered, the robot controller sets an output to the field bus. 
The safety STOP 1 can also be triggered externally. 
**Note:** This stop is called safety STOP 1 in this document. |
| Safety STOP 2               | A stop that is triggered and monitored by the safety controller. The braking process is performed by the non-safety-oriented part of the robot controller and monitored by the safety controller. The drives remain activated and the brakes released. As soon as the manipulator is at a standstill, a safe operational stop is triggered. 
When a safety STOP 2 is triggered, the robot controller sets an output to the field bus. 
The safety STOP 2 can also be triggered externally. 
**Note:** This stop is called safety STOP 2 in this document. |
3.2 Personnel

The following persons or groups of persons are defined for the industrial robot:

- User
- Personnel

All persons working with the industrial robot must have read and understood the industrial robot documentation, including the safety chapter.

User

The user must observe the labor laws and regulations. This includes e.g.:

- The user must comply with his monitoring obligations.
- The user must carry out instruction at defined intervals.

Personnel

Personnel must be instructed, before any work is commenced, in the type of work involved and what exactly it entails as well as any hazards which may exist. Instruction must be carried out regularly. Instruction is also required after particular incidents or technical modifications.

Personnel includes:

- System integrator
- Operators, subdivided into:
  - Start-up, maintenance and service personnel
  - Operating personnel
  - Cleaning personnel

Installation, exchange, adjustment, operation, maintenance and repair must be performed only as specified in the operating or assembly instructions for the relevant component of the industrial robot and only by personnel specially trained for this purpose.

System integrator

The industrial robot is safely integrated into a complete system by the system integrator.
The system integrator is responsible for the following tasks:
- Installing the industrial robot
- Connecting the industrial robot
- Performing risk assessment
- Implementing the required safety functions and safeguards
- Issuing the declaration of conformity
- Attaching the CE mark
- Creating the operating instructions for the complete system

**Operator**

The operator must meet the following preconditions:
- The operator must be trained for the work to be carried out.
- Work on the industrial robot must only be carried out by qualified personnel. These are people who, due to their specialist training, knowledge and experience, and their familiarization with the relevant standards, are able to assess the work to be carried out and detect any potential hazards.

**Example**

The tasks can be distributed as shown in the following table.

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Operator</th>
<th>Programmer</th>
<th>System integrator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch robot controller on/off</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Start program</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Select program</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Select operating mode</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Calibration (tool, base)</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Master the manipulator</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Configuration</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Programming</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Start-up</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Maintenance</td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Repair</td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Decommissioning</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Transportation</td>
<td></td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>

Work on the electrical and mechanical equipment of the industrial robot may only be carried out by specially trained personnel.

### 3.3 Workspace, safety zone and danger zone

Workspaces are to be restricted to the necessary minimum size. A workspace must be safeguarded using appropriate safeguards.

The safeguards (e.g. safety gate) must be situated inside the safety zone. In the case of a stop, the manipulator and external axes (optional) are braked and come to a stop within the danger zone.
The danger zone consists of the workspace and the stopping distances of the manipulator and external axes (optional). It must be safeguarded by means of physical safeguards to prevent danger to persons or the risk of material damage.

### 3.4 Triggers for stop reactions

Stop reactions of the industrial robot are triggered in response to operator actions or as a reaction to monitoring functions and error messages. The following tables show the different stop reactions according to the operating mode that has been set.

<table>
<thead>
<tr>
<th>Trigger</th>
<th>T1, T2</th>
<th>AUT, AUT EXT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start key released</td>
<td>STOP 2</td>
<td>-</td>
</tr>
<tr>
<td>STOP key pressed</td>
<td>STOP 2</td>
<td></td>
</tr>
<tr>
<td>Drives OFF</td>
<td>STOP 1</td>
<td></td>
</tr>
<tr>
<td>&quot;Motion enable&quot; input drops out</td>
<td>STOP 2</td>
<td></td>
</tr>
<tr>
<td>Robot controller switched off (power failure)</td>
<td>STOP 0</td>
<td></td>
</tr>
<tr>
<td>Internal error in non-safety-oriented part of the robot controller</td>
<td>STOP 0 or STOP 1 (dependent on the cause of the error)</td>
<td></td>
</tr>
<tr>
<td>Operating mode changed during operation</td>
<td>Safety stop 2</td>
<td></td>
</tr>
<tr>
<td>Safety gate opened (operator safety)</td>
<td>-</td>
<td>Safety stop 1</td>
</tr>
<tr>
<td>Enabling switch released</td>
<td>Safety stop 2</td>
<td>-</td>
</tr>
<tr>
<td>Enabling switch pressed fully down or error</td>
<td>Safety stop 1</td>
<td>-</td>
</tr>
</tbody>
</table>

![Fig. 3-1: Example of axis range A1](image-url)
3.5 Safety functions

3.5.1 Overview of the safety functions

The following safety functions are present in the industrial robot:

- Mode selection
- Operator safety (= connection for the guard interlock)
- EMERGENCY STOP system
- Enabling device
- External safe operational stop
- External safety stop 1
- External safety stop 2
- Velocity monitoring in T1

The safety functions of the industrial robot have the following performance: **Category 3** and **Performance Level d** in accordance with EN ISO 13849-1:2008. This corresponds to **SIL 2** and **HFT 1** in accordance with EN 62061.

This performance only applies under the following conditions, however:

- The EMERGENCY STOP button is pressed at least once every 6 months.

The following components are involved in the safety functions:

- Safety controller in the control PC
- KUKA Control Panel (KUKA smartPAD)
- Cabinet Interface Board
- Resolver Digital Converter (RDC)
- KUKA Power Pack
- KUKA Servo Pack

There are also interfaces to components outside the industrial robot and to other robot controllers.

**Danger!**

In the absence of functional safety functions and safeguards, the industrial robot can cause personal injury or material damage. If safety functions or safeguards are dismantled or deactivated, the industrial robot may not be operated.

**During system planning, the safety functions of the overall system must also be planned and designed. The industrial robot must be integrated into this safety system of the overall system.**

3.5.2 Safety controller

The safety controller is a unit inside the control PC. It links safety-relevant signals and safety-relevant monitoring functions.

<table>
<thead>
<tr>
<th>Trigger</th>
<th>T1, T2</th>
<th>AUT, AUT EXT</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-STOP pressed</td>
<td>Safety stop 1</td>
<td></td>
</tr>
<tr>
<td>Error in safety controller or periphery of the safety controller</td>
<td>Safety stop 0</td>
<td></td>
</tr>
</tbody>
</table>
Safety controller tasks:
- Switching off the drives; applying the brakes
- Monitoring the braking ramp
- Standstill monitoring (after the stop)
- Velocity monitoring in T1
- Evaluation of safety-relevant signals
- Setting of safety-oriented outputs

### 3.5.3 Mode selection

The industrial robot can be operated in the following modes:
- Manual Reduced Velocity (T1)
- Manual High Velocity (T2)
- Automatic (AUT)
- Automatic External (AUT EXT)

---

<table>
<thead>
<tr>
<th>Operating mode</th>
<th>Use</th>
<th>Velocities</th>
</tr>
</thead>
</table>
| T1             | For test operation, programming and teaching | Program verification: Programmed velocity, maximum 250 mm/s  
Jog mode: Jog velocity, maximum 250 mm/s |
| T2             | For test operation                        | Program verification: Programmed velocity  
Jog mode: Not possible                       |
| AUT            | For industrial robots without higher-level controllers | Program mode: Programmed velocity  
Jog mode: Not possible                        |
| AUT EXT        | For industrial robots with higher-level controllers, e.g. PLC | Program mode: Programmed velocity  
Jog mode: Not possible                        |

---

**Operator safety**

The operator safety signal is used for interlocking physical safeguards, e.g. safety gates. Automatic operation is not possible without this signal. In the event of a loss of signal during automatic operation (e.g. safety gate is opened), the manipulator stops with a safety stop 1.

Operator safety is not active in the test modes T1 (Manual Reduced Velocity) and T2 (Manual High Velocity).
3.5.5 EMERGENCY STOP device

The EMERGENCY STOP device for the industrial robot is the EMERGENCY STOP button on the KCP. The button must be pressed in the event of a hazardous situation or emergency.

Reactions of the industrial robot if the EMERGENCY STOP button is pressed:

- The manipulator and any external axes (optional) are stopped with a safety stop 1.

Before operation can be resumed, the EMERGENCY STOP button must be turned to release it.

Warning!

- Following a loss of signal, automatic operation must not be resumed merely by closing the safeguard; it must first additionally be acknowledged. It is the responsibility of the system integrator to ensure this. This is to prevent automatic operation from being resumed inadvertently while there are still persons in the danger zone, e.g. due to the safety gate closing accidentally.
- The acknowledgement must be designed in such a way that an actual check of the danger zone can be carried out first. Acknowledgement functions that do not allow this (e.g. because they are automatically triggered by closure of the safeguard) are not permissible.
- Failure to observe this may result in death to persons, severe physical injuries or considerable damage to property.

3.5.6 Logging off the higher-level safety controller

If the robot controller is connected to a higher-level safety controller, switching off the robot controller inevitably terminates this connection. The KUKA safety controller generates a signal that prevents the higher-level controller from triggering an EMERGENCY STOP for the overall system.

Warning!

- Tools and other equipment connected to the manipulator must be integrated into the EMERGENCY STOP circuit on the system side if they could constitute a potential hazard.
- Failure to observe this precaution may result in death, severe physical injuries or considerable damage to property.

There must always be at least one external EMERGENCY STOP device installed. This ensures that an EMERGENCY STOP device is available even when the KCP is disconnected.

(>>>> 3.5.7 "External EMERGENCY STOP device" Page 22)
3.5.7 External EMERGENCY STOP device

There must be EMERGENCY STOP devices on every operator panel and anywhere else it may be necessary to trigger an EMERGENCY STOP. The system integrator is responsible for ensuring this.

There must always be at least one external EMERGENCY STOP device installed. This ensures that an EMERGENCY STOP device is available even when the KCP is disconnected.

External EMERGENCY STOP devices are connected via the customer interface. External EMERGENCY STOP devices are not included in the scope of supply of the industrial robot.

3.5.8 Enabling device

The enabling devices of the industrial robot are the enabling switches on the KCP.

There are 3 enabling switches installed on the KCP. The enabling switches have 3 positions:

- Not pressed
- Center position
- Panic position

In the test modes, the manipulator can only be moved if one of the enabling switches is held in the central position.

- Releasing the enabling switch triggers a safety stop 2.
- Pressing the enabling switch down fully (panic position) triggers a safety stop 1.

It is possible, for a short time, to hold 2 enabling switches in the center position simultaneously. This makes it possible to adjust grip from one enabling switch to another one. If 2 enabling switches are held simultaneously in the center position for a longer period of time, this triggers a safety stop after several seconds.

If an enabling switch malfunctions (jams), the industrial robot can be stopped using the following methods:

- Press the enabling switch down fully
- Actuate the EMERGENCY STOP system
- Release the Start key

Warning!
The enabling switches must not be held down by adhesive tape or other means or manipulated in any other way. Death, serious physical injuries or major damage to property may result.
3.5.9 External enabling device

External enabling devices are required if it is necessary for more than one person to be in the danger zone of the industrial robot. They are connected to the robot controller via the customer interface. External enabling devices are not included in the scope of supply of the industrial robot.

3.5.10 External safe operational stop

The safe operational stop can be triggered via an input on the customer interface. The state is maintained as long as the external signal is FALSE. If the external signal is TRUE, the manipulator can be moved again. No acknowledgement is required.

3.5.11 External safety stop 1 and external safety stop 2

Safety stop 1 and safety stop 2 can be triggered via an input on the customer interface. The state is maintained as long as the external signal is FALSE. If the external signal is TRUE, the manipulator can be moved again. No acknowledgement is required.

3.5.12 Velocity monitoring in T1

The velocity at the TCP is monitored in T1 mode. If, due to an error, the velocity exceeds 250 mm/s, a safety stop 0 is triggered.

3.6 Additional protective equipment

3.6.1 Jog mode

In the operating modes T1 (Manual Reduced Velocity) and T2 (Manual High Velocity), the robot controller can only execute programs in jog mode. This means that it is necessary to hold down an enabling switch and the Start key in order to execute a program.

- Releasing the enabling switch triggers a safety stop 2.
- Pressing the enabling switch down fully (panic position) triggers a safety stop 1.
- Releasing the Start key triggers a STOP 2.

3.6.2 Software limit switches

The axis ranges of all manipulator and positioner axes are limited by means of adjustable software limit switches. These software limit switches only serve as machine protection and must be adjusted in such a way that the manipulator/positioner cannot hit the mechanical end stops.

The software limit switches are set during commissioning of an industrial robot.

Further information is contained in the operating and programming instructions.
3.6.3  Mechanical end stops

The axis ranges of main axes A1 to A3 and wrist axis A5 of the manipulator are limited by means of mechanical end stops with buffers.

Additional mechanical end stops can be installed on the external axes.

**NOTICE**  If the manipulator or an external axis hits an obstruction or a buffer on the mechanical end stop or axis range limitation, this can result in material damage to the industrial robot. KUKA Roboter GmbH must be consulted before the industrial robot is put back into operation (>>> 10 "KUKA Service" Page 185). The affected buffer must be replaced with a new one before operation of the industrial robot is resumed. If a manipulator (or external axis) collides with a buffer at more than 250 mm/s, the manipulator (or external axis) must be exchanged or recommissioning must be carried out by KUKA Roboter GmbH.

3.6.4  Mechanical axis range limitation (optional)

Some manipulators can be fitted with mechanical axis range limitation in axes A1 to A3. The adjustable axis range limitation systems restrict the working range to the required minimum. This increases personal safety and protection of the system.

In the case of manipulators that are not designed to be fitted with mechanical axis range limitation, the workspace must be laid out in such a way that there is no danger to persons or material property, even in the absence of mechanical axis range limitation.

If this is not possible, the workspace must be limited by means of photoelectric barriers, photoelectric curtains or obstacles on the system side. There must be no shearing or crushing hazards at the loading and transfer areas.

This option is not available for all robot models. Information on specific robot models can be obtained from KUKA Roboter GmbH.

3.6.5  Axis range monitoring (optional)

Some manipulators can be fitted with dual-channel axis range monitoring systems in main axes A1 to A3. The positioner axes may be fitted with additional axis range monitoring systems. The safety zone for an axis can be adjusted and monitored using an axis range monitoring system. This increases personal safety and protection of the system.

This option is not available for all robot models. Information on specific robot models can be obtained from KUKA Roboter GmbH.

3.6.6  Release device (optional)

**Description**  The release device can be used to move the manipulator manually after an accident or malfunction. The release device can be used for the main axis drive motors and, depending on the robot variant, also for the wrist axis drive motors. It is only for use in exceptional circumstances and emergencies (e.g. for freeing people).
3 Safety

1. Switch off the robot controller and secure it (e.g. with a padlock) to prevent unauthorized persons from switching it on again.
2. Remove the protective cap from the motor.
3. Push the release device onto the corresponding motor and move the axis in the desired direction.
   The directions are indicated with arrows on the motors. It is necessary to overcome the resistance of the mechanical motor brake and any other loads acting on the axis.

**CAUTION** The motors reach temperatures during operation which can cause burns to the skin. Contact must be avoided. Appropriate safety precautions must be taken, e.g. protective gloves must be worn.

**WARNING** Moving an axis with the release device can damage the motor brake. This can result in personal injury and material damage. After using the release device, the affected motor must be exchanged.

### 3.6.7 Labeling on the industrial robot

All plates, labels, symbols and marks constitute safety-relevant parts of the industrial robot. They must not be modified or removed.

Labeling on the industrial robot consists of:

- Identification plates
- Warning labels
- Safety symbols
- Designation labels
- Cable markings
- Rating plates

Further information is contained in the technical data of the operating instructions or assembly instructions of the components of the industrial robot.

### 3.6.8 External safeguards

The access of persons to the danger zone of the industrial robot must be prevented by means of safeguards. It is the responsibility of the system integrator to ensure this.

Physical safeguards must meet the following requirements:

- They meet the requirements of EN 953.
- They prevent access of persons to the danger zone and cannot be easily circumvented.
- They are sufficiently fastened and can withstand all forces that are likely to occur in the course of operation, whether from inside or outside the enclosure.
- They do not, themselves, represent a hazard or potential hazard.
- The prescribed minimum clearance from the danger zone is maintained.

Safety gates (maintenance gates) must meet the following requirements:

- They are reduced to an absolute minimum.

Moving an axis with the release device can damage the motor brake. This can result in personal injury and material damage. After using the release device, the affected motor must be exchanged.
The interlocks (e.g. safety gate switches) are linked to the operator safety input of the robot controller via safety gate switching devices or safety PLC.

Switching devices, switches and the type of switching conform to the requirements of Performance Level d and category 3 according to EN ISO 13849-1.

Depending on the risk situation: the safety gate is additionally safeguarded by means of a locking mechanism that only allows the gate to be opened if the manipulator is safely at a standstill.

The button for acknowledging the safety gate is located outside the space limited by the safeguards.

Further information is contained in the corresponding standards and regulations. These also include EN 953.

Other safety equipment

Other safety equipment must be integrated into the system in accordance with the corresponding standards and regulations.

3.7 Overview of operating modes and safety functions

The following table indicates the operating modes in which the safety functions are active.

<table>
<thead>
<tr>
<th>Safety functions</th>
<th>T1</th>
<th>T2</th>
<th>AUT</th>
<th>AUT EXT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operator safety</td>
<td>-</td>
<td>-</td>
<td>active</td>
<td>active</td>
</tr>
<tr>
<td>EMERGENCY STOP device</td>
<td>active</td>
<td>active</td>
<td>active</td>
<td>active</td>
</tr>
<tr>
<td>Enabling device</td>
<td>active</td>
<td>active</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Reduced velocity during program verification</td>
<td>active</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Jog mode</td>
<td>active</td>
<td>active</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Software limit switches</td>
<td>active</td>
<td>active</td>
<td>active</td>
<td>active</td>
</tr>
</tbody>
</table>

3.8 Safety measures

3.8.1 General safety measures

The industrial robot may only be used in perfect technical condition in accordance with its intended use and only by safety-conscious persons. Operator errors can result in personal injury and damage to property.

It is important to be prepared for possible movements of the industrial robot even after the robot controller has been switched off and locked. Incorrect installation (e.g. overload) or mechanical defects (e.g. brake defect) can cause the manipulator or external axes to sag. If work is to be carried out on a switched-off industrial robot, the manipulator and external axes must first be moved into a position in which they are unable to move on their own, whether the payload is mounted or not. If this is not possible, the manipulator and external axes must be secured by appropriate means.
**KCP**

The user must ensure that the industrial robot is only operated with the KCP by authorized persons.

If more than one KCP is used in the overall system, it must be ensured that each KCP is unambiguously assigned to the corresponding industrial robot. They must not be interchanged.

**External keyboard, external mouse**

An external keyboard and/or external mouse may only be used if the following conditions are met:

- Start-up or maintenance work is being carried out.
- The drives are switched off.
- There are no persons in the danger zone.

The KCP must not be used as long as an external keyboard and/or external mouse are connected.

The external keyboard and/or external mouse must be removed as soon as the start-up or maintenance work is completed or the KCP is connected.

**Faults**

The following tasks must be carried out in the case of faults in the industrial robot:

- Switch off the robot controller and secure it (e.g. with a padlock) to prevent unauthorized persons from switching it on again.
- Indicate the fault by means of a label with a corresponding warning (tag-out).
- Keep a record of the faults.
- Eliminate the fault and carry out a function test.

**Modifications**

After modifications to the industrial robot, checks must be carried out to ensure the required safety level. The valid national or regional work safety regulations must be observed for this check. The correct functioning of all safety circuits must also be tested.

New or modified programs must always be tested first in Manual Reduced Velocity mode (T1).
After modifications to the industrial robot, existing programs must always be tested first in Manual Reduced Velocity mode (T1). This applies to all components of the industrial robot and includes modifications to the software and configuration settings.

3.8.2 Transportation

**Manipulator**
The prescribed transport position of the manipulator must be observed. Transportation must be carried out in accordance with the operating instructions or assembly instructions of the manipulator.

**Robot controller**
The robot controller must be transported and installed in an upright position. Avoid vibrations and impacts during transportation in order to prevent damage to the robot controller.

Transportation must be carried out in accordance with the operating instructions or assembly instructions of the robot controller.

**External axis (optional)**
The prescribed transport position of the external axis (e.g. KUKA linear unit, turn-tilt table, etc.) must be observed. Transportation must be carried out in accordance with the operating instructions or assembly instructions of the external axis.

3.8.3 Start-up and recommissioning

Before starting up systems and devices for the first time, a check must be carried out to ensure that the systems and devices are complete and operational, that they can be operated safely and that any damage is detected.

The valid national or regional work safety regulations must be observed for this check. The correct functioning of all safety circuits must also be tested.

The passwords for logging onto the KUKA System Software as “Expert” and “Administrator” must be changed before start-up and must only be communicated to authorized personnel.

**Danger!**
The robot controller is preconfigured for the specific industrial robot. If cables are interchanged, the manipulator and the external axes (optional) may receive incorrect data and can thus cause personal injury or material damage. If a system consists of more than one manipulator, always connect the connecting cables to the manipulators and their corresponding robot controllers.

**Warning!**
If additional components (e.g. cables), that are not part of the scope of supply of KUKA Roboter GmbH, are integrated into the industrial robot, the user is responsible for ensuring that these components do not adversely affect or disable safety functions.

**Caution!**
If the internal cabinet temperature of the robot controller differs greatly from the ambient temperature, condensation can form, which may cause damage to the electrical components. Do not put the robot controller into operation until the internal temperature of the cabinet has adjusted to the ambient temperature.

**Function test**
The following tests must be carried out before start-up and recommissioning:

**General test:**
It must be ensured that:

- The industrial robot is correctly installed and fastened in accordance with the specifications in the documentation.
- There are no foreign bodies or loose parts on the industrial robot.
- All required safety equipment is correctly installed and operational.
- The power supply ratings of the industrial robot correspond to the local supply voltage and mains type.
- The ground conductor and the equipotential bonding cable are sufficiently rated and correctly connected.
- The connecting cables are correctly connected and the connectors are locked.

**Test of the safety functions:**

A function test must be carried out for the following safety functions to ensure that they are functioning correctly:

- Local EMERGENCY STOP device (= EMERGENCY STOP button on the KCP)
- External EMERGENCY STOP device (input and output)
- Enabling device (in the test modes)
- Operator safety
- All other safety-relevant inputs and outputs used
- Other external safety functions

**Test of reduced velocity control:**

This test is to be carried out as follows:

1. Program a straight path with the maximum possible velocity.
2. Calculate the length of the path.
3. Execute the path in T1 mode with the override set to 100\% and time the motion with a stopwatch.
4. Calculate the velocity from the length of the path and the time measured for execution of the motion.

Control of reduced velocity is functioning correctly if the following results are achieved:

- The calculated velocity does not exceed 250 mm/s.
- The manipulator executes the path as programmed (i.e. in a straight line, without deviations).

**Machine data**

It must be ensured that the rating plate on the robot controller has the same machine data as those entered in the declaration of incorporation. The machine data on the rating plate of the manipulator and the external axes (optional) must be entered during start-up.

**Warning!**

The industrial robot must not be moved if incorrect machine data are loaded. Death, severe physical injuries or considerable damage to property may otherwise result. The correct machine data must be loaded.

Following modifications to the machine data, the safety configuration must be updated.
Further information is contained in the Operating and Programming Instructions for System Integrators.

Following modifications to the machine data, control of the reduced velocity must be checked.

### 3.8.3.1 Start-up mode

**Description**

The industrial robot can be set to Start-up mode via the smartHMI user interface. In this mode, the manipulator can be moved in T1 in the absence of the safety periphery.

If a connection to a higher-level safety system exists or is established, the robot controller prevents or terminates Start-up mode.

**Hazards**

Possible hazards and risks involved in using Start-up mode:
- A person walks into the manipulator’s danger zone.
- An unauthorized person moves the manipulator.
- In a hazardous situation, a disabled external EMERGENCY STOP device is actuated and the manipulator is not shut down.

Additional measures for avoiding risks in Start-up mode:
- Cover disabled EMERGENCY STOP devices or attach a warning sign indicating that the EMERGENCY STOP device is out of operation.
- If there is no safety fence, other measures must be taken to prevent persons from entering the manipulator's danger zone, e.g. use of warning tape.
- Use of Start-up mode must be minimized – and avoided where possible – by means of organizational measures.

**Use**

Intended use of Start-up mode:
- Only service personnel who have received safety instruction may use Start-up mode.
- Start-up in T1 mode when the external safeguards have not yet been installed or put into operation. The danger zone must be delimited at least by means of warning tape.
- Fault localization (periphery fault).

**Danger!**

Use of Start-up mode disables all external safeguards. The service personnel are responsible for ensuring that there is no-one in or near the danger zone of the manipulator.

**Misuse**

Any use or application deviating from the designated use is deemed to be impermissible misuse. This includes, for example, use by any other personnel.

KUKA Roboter GmbH accepts no liability for damage or injury caused thereby. The risk lies entirely with the user.

### 3.8.4 Manual mode

Manual mode is the mode for setup work. Setup work is all the tasks that have to be carried out on the industrial robot to enable automatic operation. Setup work includes:
- Jog mode
- Teaching
3 Safety

- Programming
- Program verification

The following must be taken into consideration in manual mode:

- If the drives are not required, they must be switched off to prevent the manipulator or the external axes (optional) from being moved unintentionally. New or modified programs must always be tested first in Manual Reduced Velocity mode (T1).
- The manipulator, tooling or external axes (optional) must never touch or project beyond the safety fence.
- Workpieces, tooling and other objects must not become jammed as a result of the industrial robot motion, nor must they lead to short-circuits or be liable to fall off.
- All setup work must be carried out, where possible, from outside the safeguarded area.

If the setup work has to be carried out inside the safeguarded area, the following must be taken into consideration:

In **Manual Reduced Velocity mode (T1):**

- If it can be avoided, there must be no other persons inside the safeguarded area.
  - If it is necessary for there to be several persons inside the safeguarded area, the following must be observed:
    - Each person must have an enabling device.
    - All persons must have an unimpeded view of the industrial robot.
    - Eye-contact between all persons must be possible at all times.
  - The operator must be so positioned that he can see into the danger area and get out of harm’s way.

In **Manual High Velocity mode (T2):**

- This mode may only be used if the application requires a test at a velocity higher than Manual Reduced Velocity.
- Teaching and programming are not permissible in this operating mode.
- Before commencing the test, the operator must ensure that the enabling devices are operational.
- The operator must be positioned outside the danger zone.
- There must be no other persons inside the safeguarded area. It is the responsibility of the operator to ensure this.

### 3.8.5 Simulation

Simulation programs do not correspond exactly to reality. Robot programs created in simulation programs must be tested in the system in **Manual Reduced Velocity mode (T1)**. It may be necessary to modify the program.

### 3.8.6 Automatic mode

Automatic mode is only permissible in compliance with the following safety measures:

- All safety equipment and safeguards are present and operational.
- There are no persons in the system.
- The defined working procedures are adhered to.
If the manipulator or an external axis (optional) comes to a standstill for no apparent reason, the danger zone must not be entered until an EMERGENCY STOP has been triggered.

### 3.8.7 Maintenance and repair

After maintenance and repair work, checks must be carried out to ensure the required safety level. The valid national or regional work safety regulations must be observed for this check. The correct functioning of all safety circuits must also be tested.

The purpose of maintenance and repair work is to ensure that the system is kept operational or, in the event of a fault, to return the system to an operational state. Repair work includes troubleshooting in addition to the actual repair itself.

The following safety measures must be carried out when working on the industrial robot:

- Carry out work outside the danger zone. If work inside the danger zone is necessary, the user must define additional safety measures to ensure the safe protection of personnel.
- Switch off the industrial robot and secure it (e.g. with a padlock) to prevent it from being switched on again. If it is necessary to carry out work with the robot controller switched on, the user must define additional safety measures to ensure the safe protection of personnel.
- If it is necessary to carry out work with the robot controller switched on, this may only be done in operating mode T1.
- Label the system with a sign indicating that work is in progress. This sign must remain in place, even during temporary interruptions to the work.
- The EMERGENCY STOP systems must remain active. If safety functions or safeguards are deactivated during maintenance or repair work, they must be reactivated immediately after the work is completed.

**Warning!**

Before work is commenced on live parts of the robot system, the main switch must be turned off and secured against being switched on again. The system must then be checked to ensure that it is deenergized.

It is not sufficient, before commencing work on live parts, to execute an EMERGENCY STOP or a safety stop, or to switch off the drives, as this does not disconnect the robot system from the mains power supply in the case of the drives of the new generation. Parts remain energized. Death or severe physical injuries may result.

Faulty components must be replaced using new components with the same article numbers or equivalent components approved by KUKA Roboter GmbH for this purpose.

Cleaning and preventive maintenance work is to be carried out in accordance with the operating instructions.

**Robot controller**

Even when the robot controller is switched off, parts connected to peripheral devices may still carry voltage. The external power sources must therefore be switched off if work is to be carried out on the robot controller.

The ESD regulations must be adhered to when working on components in the robot controller.

Voltages in excess of 50 V (up to 600 V) can be present in various components for several minutes after the robot controller has been switched off! To prevent life-threatening injuries, no work may be carried out on the industrial robot in this time.
Water and dust must be prevented from entering the robot controller.

Some robot variants are equipped with a hydropneumatic, spring or gas cylinder counterbalancing system.

The hydropneumatic and gas cylinder counterbalancing systems are pressure equipment and, as such, are subject to obligatory equipment monitoring. Depending on the robot variant, the counterbalancing systems correspond to category 0, II or III, fluid group 2, of the Pressure Equipment Directive.

The user must comply with the applicable national laws, regulations and standards pertaining to pressure equipment.

Inspection intervals in Germany in accordance with Industrial Safety Order, Sections 14 and 15. Inspection by the user before commissioning at the installation site.

The following safety measures must be carried out when working on the counterbalancing system:

- The manipulator assemblies supported by the counterbalancing systems must be secured.
- Work on the counterbalancing systems must only be carried out by qualified personnel.

The following safety measures must be carried out when handling hazardous substances:

- Avoid prolonged and repeated intensive contact with the skin.
- Avoid breathing in oil spray or vapors.
- Clean skin and apply skin cream.

To ensure safe use of our products, we recommend that our customers regularly request up-to-date safety data sheets from the manufacturers of hazardous substances.

3.8.8 Decommissioning, storage and disposal

The industrial robot must be decommissioned, stored and disposed of in accordance with the applicable national laws, regulations and standards.

3.8.9 Safety measures for “single point of control”

Overview

If certain components in the industrial robot are operated, safety measures must be taken to ensure complete implementation of the principle of “single point of control” (SPOC).

Components:

- Submit interpreter
- PLC
- OPC Server
- Remote control tools
- Tools for configuration of bus systems with online functionality
- KUKA.RobotSensorInterface
- External keyboard/mouse

The implementation of additional safety measures may be required. This must be clarified for each specific application; this is the responsibility of the system integrator, programmer or user of the system.
Since only the system integrator knows the safe states of actuators in the periphery of the robot controller, it is his task to set these actuators to a safe state, e.g. in the event of an EMERGENCY STOP.

**T1, T2**

In the test modes, the components referred to above (with the exception of the external keyboard/mouse) may only access the industrial robot if the following signals have the following states:

<table>
<thead>
<tr>
<th>Signal</th>
<th>State required for SPOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$USER_SAF</td>
<td>TRUE</td>
</tr>
<tr>
<td>$SPOC_MOTION_ENABLE</td>
<td>TRUE</td>
</tr>
</tbody>
</table>

**Submit interpreter, PLC**

If motions, (e.g. drives or grippers) are controlled with the Submit interpreter or the PLC via the I/O system, and if they are not safeguarded by other means, then this control will take effect even in T1 and T2 modes or while an EMERGENCY STOP is active.

If variables that affect the robot motion (e.g. override) are modified with the Submit interpreter or the PLC, this takes effect even in T1 and T2 modes or while an EMERGENCY STOP is active.

Safety measures:

- In the test modes, the system variable $OV_PRO must not be written to by the Submit interpreter or the PLC.
- Do not modify safety-relevant signals and variables (e.g. operating mode, EMERGENCY STOP, safety gate contact) via the Submit interpreter or PLC.
  
  If modifications are nonetheless required, all safety-relevant signals and variables must be linked in such a way that they cannot be set to a dangerous state by the Submit interpreter or PLC.

**OPC server, remote control tools**

These components can be used with write access to modify programs, outputs or other parameters of the robot controller, without this being noticed by any persons located inside the system.

Safety measures:

- KUKA stipulates that these components are to be used exclusively for diagnosis and visualization.
  
  Programs, outputs or other parameters of the robot controller must not be modified using these components.

- If these components are used, outputs that could cause a hazard must be determined in a risk assessment. These outputs must be designed in such a way that they cannot be set without being enabled. This can be done using an external enabling device, for example.

**Tools for configuration of bus systems**

If these components have an online functionality, they can be used with write access to modify programs, outputs or other parameters of the robot controller, without this being noticed by any persons located inside the system.

- WorkVisual from KUKA
- Tools from other manufacturers

Safety measures:

- In the test modes, programs, outputs or other parameters of the robot controller must not be modified using these components.

**External keyboard/mouse**

These components can be used to modify programs, outputs or other parameters of the robot controller, without this being noticed by any persons located inside the system.
Safety measures:
- Only use one operator console at each robot controller.
- If the KCP is being used for work inside the system, remove any keyboard and mouse from the robot controller beforehand.

3.9 Applied norms and regulations

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Edition</th>
</tr>
</thead>
<tbody>
<tr>
<td>EN ISO 13850</td>
<td>Safety of machinery: Emergency stop - Principles for design</td>
<td>2008</td>
</tr>
<tr>
<td>EN ISO 13849-1</td>
<td>Safety of machinery: Safety-related parts of control systems - Part 1: General principles for design</td>
<td>2008</td>
</tr>
<tr>
<td>EN ISO 12100-1</td>
<td>Safety of machinery: Basic concepts, general principles for design - Part 1: Basic terminology, methodology</td>
<td>2003</td>
</tr>
<tr>
<td>EN ISO 10218-1</td>
<td>Industrial robots: Safety</td>
<td>2008</td>
</tr>
<tr>
<td>EN 614-1</td>
<td>Safety of machinery: Ergonomic design principles - Part 1: Terminology and general principles</td>
<td>2006</td>
</tr>
<tr>
<td>EN 61000-6-2</td>
<td>Electromagnetic compatibility (EMC): Part 6-2: Generic standards; Immunity for industrial environments</td>
<td>2005</td>
</tr>
<tr>
<td>EN 61000-6-4</td>
<td>Electromagnetic compatibility (EMC): Part 6-4: Generic standards; Emission standard for industrial environments</td>
<td>2007</td>
</tr>
<tr>
<td>EN 60204-1</td>
<td>Safety of machinery: Electrical equipment of machines - Part 1: General requirements</td>
<td>2006</td>
</tr>
</tbody>
</table>
EN ISO 10218-1, Annex B, specifies the need for information about the stopping time and distance. These have not yet been determined in full for all robot types in conjunction with the KR C4 robot controller. In this respect, the industrial robot does not conform to the requirements of EN ISO 10218-1.
4  Operation

4.1  KUKA smartPAD teach pendant

4.1.1  Front view

Function
The smartPAD is the teach pendant for the industrial robot. The smartPAD has all the operator control and display functions required for operating and programming the industrial robot.

The smartPAD has a touch screen: the smartHMI can be operated with a finger or stylus. An external mouse or external keyboard is not necessary.

Overview

The general term "KCP" (KUKA Control Panel) is often used in this documentation for the smartPAD.

Fig. 4-1: KUKA smartPAD, front view
<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | Button for disconnecting the smartPAD  
        [>>> 4.1.3 "Disconnecting and connecting the smartPAD"  
        Page 40] |
| 2    | Keyswitch for calling the connection manager. The switch can only be turned if the key is inserted.  
        The connection manager is used to change the operating mode.  
        [>>> 4.11 "Changing operating mode" Page 49] |
| 3    | EMERGENCY STOP button. Stops the robot in hazardous situations. The EMERGENCY STOP button locks itself in place when it is pressed. |
| 4    | Space Mouse. For moving the robot manually.  
        [>>> 4.13 "Jogging the robot" Page 51] |
| 5    | Jog keys. For moving the robot manually.  
        [>>> 4.13 "Jogging the robot" Page 51] |
| 6    | Key for setting the program override |
| 7    | Key for setting the jog override |
| 8    | Main menu key. Shows the menu items on the smartHMI.  
        [>>> 4.4 "Calling the main menu" Page 44] |
| 9    | Technology keys. The technology keys are used primarily for setting parameters in technology packages. Their exact function depends on the technology packages installed. |
| 10   | Start key. The Start key is used to start a program. |
| 11   | Start backwards key. The Start backwards key is used to start a program backwards. The program is executed step by step. |
| 12   | STOP key. The STOP key is used to stop a program that is running. |
| 13   | Keyboard key  
        Displays the keyboard. It is generally not necessary to press this key to display the keyboard, as the smartHMI detects when keyboard input is required and displays the keyboard automatically.  
        [>>> 4.2.3 "Keypad" Page 44] |
4.1.2 Rear view

Overview

![KUKA smartPAD, rear view](image)

**Fig. 4-2: KUKA smartPAD, rear view**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identification plate</td>
<td>Identification plate</td>
</tr>
<tr>
<td>Start key</td>
<td>The Start key is used to start a program.</td>
</tr>
<tr>
<td>Enabling switch</td>
<td>The enabling switch has 3 positions:</td>
</tr>
<tr>
<td></td>
<td>- Not pressed</td>
</tr>
<tr>
<td></td>
<td>- Center position</td>
</tr>
<tr>
<td></td>
<td>- Panic position</td>
</tr>
<tr>
<td></td>
<td>The enabling switch must be held in the center position in operating modes T1 and T2 in order to be able to jog the robot. In the operating modes Automatic and Automatic External, the enabling switch has no function.</td>
</tr>
<tr>
<td>USB connection</td>
<td>The USB connection is used, for example, for archiving and restoring data. Only for FAT32-formatted USB sticks.</td>
</tr>
</tbody>
</table>
4.1.3 Disconnecting and connecting the smartPAD

Description
The smartPAD can be disconnected while the robot controller is running.

**WARNING** If the smartPAD is disconnected, the system can no longer be switched off by means of the EMERGENCY STOP button on the smartPAD. For this reason, an external EMERGENCY STOP must be connected to the robot controller.

The operator must ensure that disconnected smartPADs are immediately removed from the system and stored out of sight and reach of personnel working on the industrial robot. This serves to prevent operational and non-operational EMERGENCY STOP facilities from becoming interchanged. Failure to observe these precautions may result in death to persons, severe physical injuries or considerable damage to property.

Procedure

Disconnection:

1. Press the disconnect button on the smartPAD.
   A message and a counter are displayed on the smartHMI. The counter runs for 30 s. During this time, the smartPAD can be disconnected from the robot controller.

   If the smartPAD is disconnected without the counter running, this triggers an EMERGENCY STOP. The EMERGENCY STOP can only be canceled by plugging the smartPAD back in.

2. Disconnect the smartPAD from the robot controller.
   If the counter expires without the smartPAD having been disconnected, this has no effect. The disconnect button can be pressed again at any time to display the counter again.

Connection:

- Connect the smartPAD to the robot controller.

A smartPAD can be connected at any time. Precondition: Same smartPAD variant as the disconnected device. The EMERGENCY STOP and enabling switches are operational again 30 s after connection. The smartHMI is automatically displayed again. (This may take longer than 30 s.)

The connected smartPAD assumes the current operating mode of the robot controller.

The current operating mode is not, in all cases, the same as that before the smartPAD was disconnected: if the robot controller is part of a RoboTeam, the operating mode may have been changed after disconnection, e.g. by the master.

**WARNING** The user connecting a smartPAD to the robot controller must subsequently stay with the smartPAD for at least 30 s, i.e. until the EMERGENCY STOP and enabling switches are operational once again. This prevents another user from trying to activate a non-operational EMERGENCY STOP in an emergency situation, for example. Failure to observe this may result in death to persons, physical injuries or damage to property.
4.2 KUKA smartHMI user interface

**Fig. 4-3: KUKA smartHMI user interface**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Status bar (&gt;&gt;&gt; 4.2.1 &quot;Status bar&quot; Page 42)</td>
</tr>
</tbody>
</table>
| 2    | Message counter  
The message counter indicates how many messages of each message type are active. Touching the message counter enlarges the display. |
| 3    | Message window  
By default, only the last message is displayed. Touching the message window expands it so that all active messages are displayed.  
An acknowledgeable message can be acknowledged with OK. All acknowledgeable messages can be acknowledged at once with All OK. |
| 4    | Space Mouse status indicator  
This indicator shows the current coordinate system for jogging with the Space Mouse. Touching the indicator displays all coordinate systems, allowing a different one to be selected. |
4.2.1 Status bar

The status bar indicates the status of certain central settings of the industrial robot. In most cases, touching the display opens a window in which the settings can be modified.

Overview

Fig. 4-4: KUKA smartHMI status bar
4.2.2  “Submit interpreter” status indicator

<table>
<thead>
<tr>
<th>Icon</th>
<th>Color</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Yellow Icon" /></td>
<td>Yellow</td>
<td>Submit interpreter is selected. The block pointer is situated on the first line of the selected SUB program.</td>
</tr>
<tr>
<td><img src="image" alt="Green Icon" /></td>
<td>Green</td>
<td>Submit interpreter is running.</td>
</tr>
<tr>
<td><img src="image" alt="Red Icon" /></td>
<td>Red</td>
<td>Submit interpreter has been stopped.</td>
</tr>
<tr>
<td><img src="image" alt="Gray Icon" /></td>
<td>Gray</td>
<td>Submit interpreter is deselected.</td>
</tr>
</tbody>
</table>
4.2.3 Keypad

The smartPAD has a touch screen: the smartHMI can be operated with a finger or stylus.

There is a keypad on the smartHMI for entering letters and numbers. The smartHMI detects when the entry of letters or numbers is required and automatically displays the keypad.

The keypad only ever displays the characters that are required. If, for example, a box is edited in which only numbers can be entered, then only numbers are displayed and not letters.

![Example keypad](image)

Fig. 4-5: Example keypad

4.3 Switching on the robot controller and starting the KSS

**Procedure**

- Turn the main switch on the robot controller to ON.
  
  The operating system and the KSS start automatically.

If the KSS does not start automatically, e.g. because the Startup function has been disabled, execute the file StartKRC.exe in the directory C:\KRC.

If the robot controller is logged onto the network, the start may take longer.

4.4 Calling the main menu

**Procedure**

- Press “Main menu” key on the KCP. The Main menu window is opened.
  
  The display is always the same as that which was in the window before it was last closed.

**Description**

Properties of the Main Menu window:

- The main menu is displayed in the left-hand column.
- Touching a menu item that contains an arrow opens the corresponding submenu (e.g. Configure).
  
  Depending on how many nested submenus are open, the Main Menu column may no longer be visible, with only the submenus remaining visible.
- The arrow key in the top right-hand corner closes the most recently opened submenu.
- The Home key in the top right-hand corner closes all open submenus.
- The most recently selected menu items are displayed in the bottom section (maximum 6).
  This makes it possible to select these menu items again directly without first having to close other submenus that might be open.
- The white cross on the left-hand side closes the window.

![Main Menu](image)

Fig. 4-6: Example: Configure submenu is open.

### 4.5 Exiting or restarting KSS

**Precondition**
- User group “User”
- T1 or T2 operating mode

**Procedure**
1. Select the menu item **Shutdown** in the main menu.
2. Select the desired options.
3. Press **Shut down KRC**. Confirm the request for confirmation with **Yes**.

   The KSS is terminated.

---

**NOTICE**
If, on shutting down, the option with restart was selected, the main switch on the robot controller must not be pressed until the reboot has been completed. System files may otherwise be destroyed.

If this option was not selected on shutting down, the main switch can be pressed once the controller has shut down.

---

**If the robot controller detects a system error or modified data, the KSS always starts with a cold start – irrespective of the selected start type.**
Description

The following options are available:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Start type - Cold start</strong></td>
<td>After a power failure, the robot controller starts with a cold start. (A power failure and start are generally triggered by switching the main switch on the robot controller off and on.)&lt;br&gt;The setting can only be modified in the user group “Expert”.&lt;br&gt;(&gt;&gt;&gt; 4.6 &quot;Start types&quot; Page 47)</td>
</tr>
<tr>
<td><strong>Start type - Hibernate</strong></td>
<td>After a power failure, the robot controller starts with a start after Hibernate. (A power failure and start are generally triggered by switching the main switch on the robot controller off and on.)&lt;br&gt;The setting can only be modified in the user group “Expert”.&lt;br&gt;(&gt;&gt;&gt; 4.6 &quot;Start types&quot; Page 47)</td>
</tr>
<tr>
<td><strong>Power-off delay time</strong></td>
<td>Wait time before the robot controller is shut down. The purpose of the wait time is to ensure that the system does not immediately shut down, for example, in the event of a very sudden, brief power failure, but bridges the power failure for the duration of the wait time.&lt;br&gt;This value can only be changed in the user group “Expert”.</td>
</tr>
</tbody>
</table>
## 4.6 Start types

<table>
<thead>
<tr>
<th>Start type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cold start</td>
<td>After a cold start the robot controller displays the Navigator. No program is selected. The controller is completely reinitialized, e.g. all user outputs are set to FALSE.</td>
</tr>
<tr>
<td>Hibernate</td>
<td>After a start with Hibernate, the previously selected robot program can be resumed. The state of the kernel system: programs, block pointer, variable contents and outputs, is completely restored. Additionally, all programs that were open parallel to the robot controller are reopened and have the same state that they had before the system was shut down. The last state of Windows is also restored.</td>
</tr>
</tbody>
</table>

## 4.7 Switching the robot controller off

**Procedure**
- Turn the main switch on the robot controller to OFF.
- The robot controller automatically backs up data.

**NOTICE**
The main switch on the robot controller must not be operated if the KSS has been exited with the option **Reboot control PC** and the reboot has not yet been completed. System files may otherwise be destroyed.

## 4.8 Setting the user interface language

**Procedure**
1. In the main menu, select **Configuration > Miscellaneous > Language**.
2. Select the desired language. Click **OK** to confirm.

### 4.9 Changing user group

**Procedure**

1. Select **Configuration > User group** in the main menu. The current user group is displayed.
2. Press **Default** to switch to the default user group. (**Default** is not available if the default user group is already selected.)
   
   Press **Login...** to switch to a different user group. Select the desired user group.
3. If prompted: Enter password and confirm with **Log-on**.

**Description**

Different functions are available in the KSS, depending on the user group. The following user groups are available:

- **Operator**
  
  User group for the operator. This is the default user group.

- **User**
  
  User group for the operator. (By default, the user groups “Operator” and “User” are defined for the same target group.)

- **Expert**
  
  User group for the programmer. This user group is protected by means of a password.

- **Safety recovery**
  
  User group for the start-up technician. This user can activate and configure the safety configuration of the robot.
  
  This user group is protected by means of a password.

- **Safety maintenance**
  
  This user group is only relevant if KUKA.SafeOperation or KUKA.SafeRangeMonitoring is used. The user group is protected by means of a password.

- **Administrator**
  
  The range of functions is the same as that for the user group “Expert”. It is additionally possible, in this user group, to integrate plug-ins into the robot controller.
  
  This user group is protected by means of a password.

The default password is “kuka”.

When the system is booted, the default user group is selected.

If the mode is switched to AUT or AUT EXT, the robot controller switches to the default user group for safety reasons. If a different user group is desired, this must be selected subsequently.

If no actions are carried out in the user interface within a certain period of time, the robot controller switches to the default user group for safety reasons. The default setting is 300 s.

### 4.10 Disabling the robot controller

**Description**

The robot controller can be disabled. It is then disabled for all actions except logging back on.

The robot controller cannot be disabled in the default user group.

**Precondition**

- The default user group is not selected.

**Procedure**

1. Select **Configuration > User group** in the main menu.
2. Press **Lock**. The robot controller is then disabled for all actions except logging on. The current user group is displayed.

3. Log back on:
   - Log on as the default user: Press **Default**.
   - Log on as a different user: Press **Login**... Select the desired user group and confirm with **Log-on**.
     If prompted: Enter password and confirm with **Log-on**.

---

**4.11 Changing operating mode**

- Do not change the operating mode while a program is running. If the operating mode is changed during program execution, the industrial robot is stopped with a safety stop 2.

**Precondition**
- The robot controller is not executing a program.
- Key for the switch for calling the connection manager

**Procedure**
1. On the smartPAD, turn the switch for the connection manager. The connection manager is displayed.
2. Select the operating mode.
3. Return the switch for the connection manager to its original position.

   The selected operating mode is displayed in the status bar of the smartPAD.

<table>
<thead>
<tr>
<th>Operating mode</th>
<th>Use</th>
<th>Velocities</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>For test operation, programming and teaching</td>
<td>Program verification: Programmed velocity, maximum 250 mm/s &lt;br&gt; Jog mode: Jog velocity, maximum 250 mm/s</td>
</tr>
<tr>
<td>T2</td>
<td>For test operation</td>
<td>Program verification: Programmed velocity &lt;br&gt; Jog mode: Not possible</td>
</tr>
<tr>
<td>AUT</td>
<td>For industrial robots without higher-level controllers</td>
<td>Program mode: Programmed velocity &lt;br&gt; Jog mode: Not possible</td>
</tr>
<tr>
<td>AUT EXT</td>
<td>For industrial robots with higher-level controllers, e.g. PLC</td>
<td>Program mode: Programmed velocity &lt;br&gt; Jog mode: Not possible</td>
</tr>
</tbody>
</table>

---

**4.12 Coordinate systems**

**Overview**

The following Cartesian coordinate systems are defined in the robot controller:

- **WORLD**
- **ROBROOT**
Description

**WORLD**

The WORLD coordinate system is a permanently defined Cartesian coordinate system. It is the root coordinate system for the ROBROOT and BASE coordinate systems.

By default, the WORLD coordinate system is located at the robot base.

**ROBROOT**

The ROBROOT coordinate system is a Cartesian coordinate system, which is always located at the robot base. It defines the position of the robot relative to the WORLD coordinate system.

By default, the ROBROOT coordinate system is identical to the WORLD coordinate system. $ROBROOT allows the definition of an offset of the robot relative to the WORLD coordinate system.

**BASE**

The BASE coordinate system is a Cartesian coordinate system that defines the position of the workpiece. It is relative to the WORLD coordinate system.

By default, the BASE coordinate system is identical to the WORLD coordinate system. It is offset to the workpiece by the user.

(>>> 5.6.2 "Base calibration" Page 88)

**TOOL**

The TOOL coordinate system is a Cartesian coordinate system which is located at the tool center point.

By default, the origin of the TOOL coordinate system is located at the flange center point. (In this case it is called the FLANGE coordinate system.) The TOOL coordinate system is offset to the tool center point by the user.
4.13 Jogging the robot

Description

There are 2 ways of jogging the robot:

- Cartesian jogging
  The TCP is jogged in the positive or negative direction along the axes of a coordinate system.
- Axis-specific jogging
  Each axis can be moved individually in a positive and negative direction.

 Overview

<table>
<thead>
<tr>
<th>Angle</th>
<th>Rotation about axis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle A</td>
<td>Rotation about the Z axis</td>
</tr>
<tr>
<td>Angle B</td>
<td>Rotation about the Y axis</td>
</tr>
<tr>
<td>Angle C</td>
<td>Rotation about the X axis</td>
</tr>
</tbody>
</table>

Fig. 4-9: Axis-specific jogging

There are 2 operator control elements that can be used for jogging the robot:

- Jog keys
- Space Mouse

Cartesian jogging

<table>
<thead>
<tr>
<th>Jog keys</th>
<th>(&gt;&gt;&gt; 4.13.6 &quot;Cartesian jogging with the jog keys&quot; Page 56)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Space Mouse</td>
<td>(&gt;&gt;&gt; 4.13.9 &quot;Cartesian jogging with the Space Mouse&quot; Page 59)</td>
</tr>
</tbody>
</table>

Axis-specific jogging

<table>
<thead>
<tr>
<th>Jog keys</th>
<th>(&gt;&gt;&gt; 4.13.5 &quot;Axis-specific jogging with the jog keys&quot; Page 56)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Space Mouse</td>
<td>Axis-specific jogging with the Space Mouse is possible, but is not described here.</td>
</tr>
</tbody>
</table>
4.13.1 “Jog options” window

Description
All parameters for jogging the robot can be set in the Jogging Options window.

Procedure
Open the Jogging Options window:
1. Open a status indicator on the smartHMI, e.g. the POV status indicator.
   (Not possible for the Submit interpreter, Drives and Robot interpreter status indicators.)
   A window opens.
2. Press Options. The Jogging Options window is opened.

For most parameters, it is not necessary to open the Jogging Options window. They can be set directly via the smartHMI status indicators.

4.13.1.1 “General” tab

![General tab](image)

**Fig. 4-10: General tab**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Set program override</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 6.5.4 &quot;Setting the program override (POV)&quot; Page 115)</td>
</tr>
<tr>
<td>2</td>
<td>Set jog override</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 4.13.3 &quot;Setting the jog override (HOV)&quot; Page 55)</td>
</tr>
<tr>
<td>3</td>
<td>Select the program run mode</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 6.5.2 &quot;Program run modes&quot; Page 114)</td>
</tr>
</tbody>
</table>
4.13.1.2 “Keys” tab

**Fig. 4-11: Keys tab**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | Activate jog mode “Jog keys”  
      (>>> 4.13.2 "Activating the jog mode“ Page 55) |
| 2    | Select a kinematics group. The kinematics group defines the axes to which the jog keys refer.  
      Default: Robot axes (= A1 to A6)  
      Depending on the system configuration, other kinematics groups may be available.  
      (>>> 4.14 “Jogging external axes“ Page 61) |
| 3    | Select the coordinate system for jogging with the jog keys |
| 4    | Incremental jogging  
      (>>> 4.13.10 "Incremental jogging“ Page 60) |

4.13.1.3 “Mouse” tab

**Fig. 4-12: Mouse tab**
4.13.1.4 “KCP pos.” tab

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | Activate jog mode “Space Mouse”  
   (>>> 4.13.2 “Activating the jog mode” Page 55) |
| 2    | Configure the Space Mouse  
   (>>> 4.13.7 “Configuring the Space Mouse” Page 56) |
| 3    | Select the coordinate system for jogging with the Space Mouse |

Fig. 4-13: Kcp Pos. tab

4.13.1.5 “Cur. tool/base” tab

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
   (>>> 4.13.8 “Defining the alignment of the Space Mouse” Page 58) |

Fig. 4-14: Act. Tool/Base tab
4.13.2 Activating the jog mode

**Procedure**

1. Open the Jogging Options window.
   (>>> 4.13.1 "Jog options" window Page 52)

2. To activate the jog mode “Jog keys”:
   - On the Keys tab, activate the Activate Keys check box.
   - To activate the jog mode “Space Mouse”:
     - On the Mouse tab, activate the Activate Mouse check box.

**Description**

Both jog modes “Jog keys” and “Space Mouse” can be activated simultaneously. If the robot is jogged using the keys, the Space Mouse is disabled until the robot comes to a standstill. If the Space Mouse is actuated, the keys are disabled.

4.13.3 Setting the jog override (HOV)

**Description**

Jog override is the velocity of the robot during jogging. It is specified as a percentage and refers to the maximum possible jog velocity. This is 250 mm/s.

**Procedure**

1. Touch the POV/HOV status indicator. The Overrides window is opened.

2. Set the desired jog override. It can be set using either the plus/minus keys or by means of the slide controller.
   - Plus/minus keys: The value can be set to 100%, 75%, 50%, 30%, 10%, 3%, 1%
   - Slide controller: The override can be adjusted in 1% steps.

3. Touch the POV/HOV status indicator again. (Or touch the area outside the window.)
   - The window closes and the selected override value is applied.

**Alternative procedure**

Alternatively, the override can be set using the plus/minus key on the right-hand side of the KCP.

The value can be set to 100%, 75%, 50%, 30%, 10%, 3%, 1%.
4.13.4 Selecting the tool and base

Description
A maximum of 16 TOOL and 32 BASE coordinate systems can be saved in the robot controller. One tool (TOOL coordinate system) and one base (BASE coordinate system) must be selected for Cartesian jogging.

Procedure
1. Touch the Tool/base status indicator. The Act. Tool/Base window is opened.
2. Select the desired tool and base.
3. The window closes and the selection is applied.

4.13.5 Axis-specific jogging with the jog keys

Precondition
- The jog mode “Jog keys” is active.
- Operating mode T1

Procedure
1. Select Axes as the coordinate system for the jog keys.
2. Set jog override.
3. Hold down the enabling switch.
   Axes A1 to A6 are displayed next to the jog keys.
4. Press the Plus or Minus jog key to move an axis in the positive or negative direction.

4.13.6 Cartesian jogging with the jog keys

Precondition
- The jog mode “Jog keys” is active.
- Operating mode T1
- Tool and base have been selected.
  (>>> 4.13.4 "Selecting the tool and base" Page 56)

Procedure
1. Select World, Base or Tool as the coordinate system for the jog keys.
2. Set jog override.
3. Hold down the enabling switch.
   The following designations are displayed next to the jog keys:
   - X, Y, Z: for the linear motions along the axes of the selected coordinate system
   - A, B, C: for the rotational motions about the axes of the selected coordinate system
4. Press the Plus or Minus jog key to move the robot in the positive or negative direction.

4.13.7 Configuring the Space Mouse

Procedure
1. Open the Jogging Options window and select the Mouse tab.
  (>>> 4.13.1 "Jog options” window" Page 52)
2. Mouse Settings group:
- **Dominant** check box:
  Activate or deactivate dominant mode as desired.
- **6D/XYZ/ABC** option box:
  Select whether the TCP is to be moved using translational motions, rotational motions, or both.

3. Close the **Jogging Options** window.

### Description

**Fig. 4-15: Mouse settings**

**Dominant** check box:
Depending on the dominant mode, the Space Mouse can be used to move just one axis or several axes simultaneously.

<table>
<thead>
<tr>
<th>Check box</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Dominant mode is activated. Only the coordinate axis with the greatest deflection of the Space Mouse is moved.</td>
</tr>
<tr>
<td>Inactive</td>
<td>Dominant mode is deactivated. Depending on the axis selection, either 3 or 6 axes can be moved simultaneously.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| 6D     | The robot can be moved by pulling, pushing, rotating or tilting the Space Mouse.  
The following motions are possible with Cartesian jogging:  
  - Translational motions in the X, Y and Z directions  
  - Rotational motions about the X, Y and Z axes |
| XYZ    | The robot can only be moved by pulling or pushing the Space Mouse.  
The following motions are possible with Cartesian jogging:  
  - Translational motions in the X, Y and Z directions |
| ABC    | The robot can only be moved by rotating or tilting the Space Mouse.  
The following motions are possible with Cartesian jogging:  
  - Rotational motions about the X, Y and Z axes |
4.13.8 Defining the alignment of the Space Mouse

**Description**

The functioning of the Space Mouse can be adapted to the location of the user so that the motion direction of the TCP corresponds to the deflection of the Space Mouse.

The location of the user is specified in degrees. The reference point for the specification in degrees is the junction box on the base frame. The position of the robot arm or axes is irrelevant.

Default setting: 0°. This corresponds to a user standing opposite the junction box.
4 Operation

Precondition
- Operating mode T1

Procedure
1. Open the Jog options window and select the KCP pos. tab.
2. Drag the KCP to the position corresponding to the location of the user (in 45° steps).
3. Close the Jog options window.

Fig. 4-18: Space Mouse: 0° and 270°

Fig. 4-19: Defining the alignment of the Space Mouse

- Switching to Automatic External mode automatically resets the alignment of the Space Mouse to 0°.

4.13.9 Cartesian jogging with the Space Mouse

Precondition
- The jog mode “Space Mouse” is active.
- Operating mode T1
- Tool and base have been selected.
  (>>> 4.13.4 "Selecting the tool and base" Page 56)
- The Space Mouse is configured.
  (>>> 4.13.7 "Configuring the Space Mouse" Page 56)
The alignment of the Space Mouse has been defined.
(>>> 4.13.8 "Defining the alignment of the Space Mouse" Page 58)

**Procedure**

1. Select **World, Base** or **Tool** as the coordinate system for the Space Mouse.
2. Set jog override.
3. Hold down the enabling switch.
4. Move the robot in the desired direction using the Space Mouse.

The position of the robot during jogging can be displayed: select **Display > Actual position** in the main menu.

### 4.13.10 Incremental jogging

**Description**

Incremental jogging makes it possible to move the robot a defined distance, e.g. 10 mm or 3°. The robot then stops by itself.

Incremental jogging can be activated for jogging with the jog keys. Incremental jogging is not possible in the case of jogging with the Space Mouse.

**Areas of application:**

- Positioning of equidistant points
- Moving a defined distance away from a position, e.g. in the event of a fault
- Mastering with the dial gauge

The following options are available:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continuous</td>
<td>Incremental jogging is deactivated.</td>
</tr>
<tr>
<td>100 m / 10°</td>
<td>1 increment = 100 mm or 10°</td>
</tr>
<tr>
<td>10 mm / 3°</td>
<td>1 increment = 10 mm or 3°</td>
</tr>
<tr>
<td>1 mm / 1°</td>
<td>1 increment = 1 mm or 1°</td>
</tr>
<tr>
<td>0.1 mm / 0.005°</td>
<td>1 increment = 0.1 mm or 0.005°</td>
</tr>
</tbody>
</table>

Increments in mm:

- Valid for Cartesian jogging in the X, Y or Z direction.

Increments in degrees:

- Valid for Cartesian jogging in the A, B or C direction.
- Valid for axis-specific jogging.

**Precondition**

- The jog mode “Jog keys” is active.
- Operating mode T1

**Procedure**

1. Select the size of the increment in the status bar.
2. Jog the robot using the jog keys. Jogging can be Cartesian or axis-specific.

Once the set increment has been reached, the robot stops.

If the robot motion is interrupted, e.g. by releasing the enabling switch, the interrupted increment is not resumed with the next motion; a new increment is started instead.
4.14 Jogging external axes

External axes cannot be moved using the Space Mouse. If “Space Mouse” mode is selected, only the robot can be jogged with the Space Mouse. The external axes, on the other hand, must be jogged using the jog keys.

**Precondition**
- The jog mode “Jog keys” is active.
- Operating mode T1

**Procedure**
1. Select the desired kinematics group, e.g. **External axes**, on the **Keys** tab in the **Jog options** window.
   The type and number of kinematics groups available depend on the system configuration.
2. Set jog override.
3. Hold down the enabling switch.
   The axes of the selected kinematics group are displayed next to the jog keys.
4. Press the Plus or Minus jog key to move an axis in the positive or negative direction.

**Description**
Depending on the system configuration, the following kinematics groups may be available.

<table>
<thead>
<tr>
<th>Kinematics group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Robot Axes</strong></td>
<td>The robot axes can be moved using the jog keys. The external axes cannot be jogged.</td>
</tr>
<tr>
<td><strong>External Axes</strong></td>
<td>All configured external axes (e.g. external axes E1 to E5) can be moved using the jog keys.</td>
</tr>
<tr>
<td><strong>NAME / External Kinematics Groupn</strong></td>
<td>The axes of an external kinematics group can be moved using the jog keys. The name is taken from the system variable $ET_NAME (n = number of the external kinematicic system). If $ET_NAME is empty, the default name <strong>External Kinematics Groupn</strong> is displayed.</td>
</tr>
<tr>
<td>[User-defined kinematics group]</td>
<td>The axes of a user-defined kinematics group can be moved using the jog keys. The name corresponds to the name of the user-defined kinematics group.</td>
</tr>
</tbody>
</table>

4.15 Bypassing workspace monitoring

**Description**
Workspaces can be configured for a robot. Workspaces serve to protect the system.

There are 2 types of workspace:
- The workspace is an exclusion zone.
  The robot may only move outside the workspace.
- Only the workspace is a permitted zone.
  The robot may not move outside the workspace.

Exactly what reactions occur when the robot violates a workspace depends on the configuration.

One possible reaction, for example, is that the robot stops and an error message is generated. The workspace monitoring must be bypassed in such a case. The robot can then move back out of the prohibited workspace.
Precondition

- User group “Expert”
- Operating mode T1

Procedure

1. In the main menu, select **Configuration > Miscellaneous > Workspace monitoring > Override**.
2. Move the robot manually out of the prohibited workspace.
   Once the robot has left the prohibited workspace, the workspace monitoring is automatically active again.

4.16 Monitor functions

4.16.1 Displaying the actual position

Procedure

1. Select **Display > Actual position** in the main menu. The Cartesian actual position is displayed.
2. To display the axis-specific actual position, press **Axis-spec.**.
3. To display the Cartesian actual position again, press **Cartesian**.

Description

**Actual position, Cartesian:**

The current position (X, Y, Z) and orientation (A, B, C) of the TCP are displayed. In addition to this, the current TOOL and BASE coordinate systems and the Status and Turn are displayed.

**Actual position, axis-specific:**

The current position of axes A1 to A6 is indicated. If external axes are being used, the position of the external axes is also displayed.

The actual position can also be displayed while the robot is moving.

![Fig. 4-20: Actual position, Cartesian](image.png)
4.16.2 Displaying digital inputs/outputs

Procedure

1. In the main menu, select Display > Inputs/outputs > Digital I/O.
2. To display a specific input/output:
   - Select any cell in the No. column.
   - Enter the number using the keypad.
   - The display jumps to the input/output with this number.

Description

![Digital I/O display](image)

**Fig. 4-22: Digital inputs**
The following buttons are available:

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-100</td>
<td>Toggles back 100 inputs or outputs in the display.</td>
</tr>
<tr>
<td>+100</td>
<td>Toggles forward 100 inputs or outputs in the display.</td>
</tr>
<tr>
<td>Go to</td>
<td>The number of the input or output being searched for can be entered.</td>
</tr>
<tr>
<td>Value</td>
<td>Toggles the selected input/output between TRUE and FALSE. Precondition: The enabling switch is pressed. This button is not available in AUT EXT mode, and is only available for inputs if simulation is activated.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the selected input or output can be changed.</td>
</tr>
</tbody>
</table>

### 4.16.3 Displaying analog inputs/outputs

**Procedure**

1. In the main menu, select **Display > Inputs/outputs > Analog I/O**.
2. To display a specific input/output:
   - Select any cell in the **No.** column.
   - Enter the number using the keypad.
   - The display jumps to the input/output with this number.
The following buttons are available:

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
</table>
| Voltage | A voltage can be entered for the selected output.  
  -10 ... 10 V  
  This button is not available for inputs. |
| Name   | The name of the selected input or output can be changed. |

### Fig. 4-24: Analog inputs

1. Input/output number
2. Input/output voltage  
  - -10 ... 10 V
3. Name of the input/output

### Fig. 4-25: Analog outputs

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Input/output number</td>
</tr>
</tbody>
</table>
| 2    | Input/output voltage  
  - -10 ... 10 V |
| 3    | Name of the input/output |
4.16.4 Displaying inputs/outputs for Automatic External

Procedure
- In the main menu, select **Display > Inputs/outputs > Automatic External**.

Description

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Number</td>
</tr>
<tr>
<td>2</td>
<td>Status</td>
</tr>
<tr>
<td></td>
<td>- Gray: inactive (FALSE)</td>
</tr>
<tr>
<td></td>
<td>- Red: active (TRUE)</td>
</tr>
<tr>
<td>3</td>
<td>Long text name of the input/output</td>
</tr>
<tr>
<td>4</td>
<td>Type</td>
</tr>
<tr>
<td></td>
<td>- Green: input/output</td>
</tr>
<tr>
<td></td>
<td>- Yellow: variable or system variable ($...)</td>
</tr>
<tr>
<td>5</td>
<td>Name of the signal or variable</td>
</tr>
<tr>
<td>6</td>
<td>Input/output number or channel number</td>
</tr>
</tbody>
</table>

Columns 4, 5 and 6 are only displayed if **Details** has been pressed.

Fig. 4-26: Automatic External inputs (detail view)

Fig. 4-27: Automatic External outputs (detail view)
The following buttons are available:

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Config.</td>
<td>Switches to the configuration of the Automatic External interface.</td>
</tr>
<tr>
<td>Inputs/Outputs</td>
<td>Toggles between the windows for inputs and outputs.</td>
</tr>
<tr>
<td>Details/Normal</td>
<td>Toggles between the Details and Normal views.</td>
</tr>
</tbody>
</table>

### 4.16.5 Displaying calibration data

**Procedure**

1. In the main menu, select Start-up > Calibrate > Calibration points and the desired menu item:
   - Tool type
   - Base type
   - External axis
2. Enter the number of the tool, base or external kinematic system. The calibration method and the calibration data are displayed.

### 4.16.6 Displaying information about the robot and robot controller

**Procedure**

- Select Help > Info in the main menu.

**Description**

The information is required, for example, when requesting help from KUKA Customer Support.

The tabs contain the following information:

<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
</tr>
</thead>
</table>
| Info  | - Robot controller type  
        - Robot controller version  
        - User interface version  
        - Kernel system version   |
| Robot | - Robot name  
        - Robot type and configuration  
        - Service life  
        The operating hours meter is running as long as the drives are switched on. Alternatively, the operating hours can also be displayed via the variable $ROB_RUNTIME.  
        - Number of axes  
        - List of external axes  
        - Machine data version   |
| System| - Control PC name  
        - Operating system versions  
        - Storage capacities   |
| Options| Additionally installed options and technology packages |
| Comments| Additional comments |
| Modules| Names and versions of important system files  
        The Save button exports the contents of the Modules tab to the file C:\KRC\ROBOTER\LOG\OCXVER.TXT. |
4.16.7 Displaying/editing robot data

Procedure

- Select **Start-up > Robot data** in the main menu.

Description

![Robot data window](image)

**Fig. 4-28: Robot data window**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Serial number</td>
</tr>
<tr>
<td>2</td>
<td>Operating hours. The operating hours meter is running as long as the drives are switched on. Alternatively, the operating hours can also be displayed via the variable $ROBRUNTIME.</td>
</tr>
<tr>
<td>3</td>
<td>Machine data name</td>
</tr>
<tr>
<td>4</td>
<td>Robot name. The robot name can be changed.</td>
</tr>
</tbody>
</table>
| 5    | Robot controller data can be archived to a network path.  
(>>> 6.8.3 "Archiving on the network" Page 123)  
The archive path is defined here. |
| 6    | This box is only displayed if the check box **Incorporate robot name into archive name** is not activated.  
A name for the archive file can be defined here. |
| 7    | **Check box active:** The robot name is used as the name for the archive file. If no robot name is defined, the name *archive* is used.  
**Check box not active:** A separate name can be defined for the archive file. |

The buttons are not available in the user group “User”.
5 Start-up and recommissioning

5.1 Start-up wizard

Description
Start-up can be carried out using the Start-up wizard. This guides the user through the basic start-up steps.

Procedure
- Select Start-up > Start-up wizard in the main menu.

5.2 Checking the machine data

Description
The correct machine data must be loaded. This must be checked by comparing the loaded machine data with the machine data on the rating plate.

If machine data are reloaded, the version of the machine data must correspond exactly to the KSS version. This is ensured if the machine data supplied together with the KSS release are used.

DANGER
The robot must not be moved if incorrect machine data are loaded. Death, severe physical injuries or considerable damage to property may otherwise result. The correct machine data must be loaded.

Fig. 5-1: Rating plate

Procedure
1. In the main menu, select Start-up > Robot data.
   The Robot Data window is opened.
2. Compare the following entries:
   - In the Robot data window: the entry in the Machine data box
   - On the rating plate on the base of the robot: the entry in the line $TRAFTONAME()=“# .....”

The file path of the machine data on the CD is specified on the rating plate in the line ...\MADA1.
5.3 Jogging the robot without a higher-level safety controller

Description
To jog the robot without a higher-level safety controller, Start-up mode must first be activated. The robot can then be jogged in T1 mode.

If there is a connection to a higher-level safety controller, Start-up mode cannot be activated. If the robot is in Start-up mode and a connection to a higher-level safety controller is established, Start-up mode is automatically deactivated.

In Start-up mode, the system switches to the following simulated input image:

- The safety gate is open.
- The external EMERGENCY STOP is not active.
- Safety stop 2 is not requested.

Precondition
- No connection to a higher-level safety controller
- Operating mode T1

Procedure
- In the main menu, select Start-up > Service > Start-up mode.

5.4 Checking the activation of the positionally accurate robot model

Description
If a positionally accurate robot is used, it must be checked that the positionally accurate robot model is activated.

In the case of positionally accurate robots, position deviations resulting from workpiece tolerances and elastic effects of the individual robots are compensated for. The positionally accurate robot positions the programmed TCP anywhere in the Cartesian workspace within the tolerance limits. The model parameters of the positionally accurate robot are determined at a calibration station and permanently saved on the robot (RDC).

Functions
A positionally accurate robot has the following functions:

- Increased positioning accuracy, approximately by the factor 10
- Increased path accuracy

A precondition for the increased positioning and path accuracy is the correct input of the load data into the robot controller.

- Simplified transfer of programs if the robot is exchanged (no reteaching)
- Simplified transfer of programs after offline programming with WorkVisual (no reteaching)

**Procedure**

1. In the main menu, select **Help > Info**.
2. Check on the **Robot** tab that the positionally accurate robot model is activated. (= specification **Positionally accurate robot**).

**5.5 Mastering**

**Overview**

Every robot must be mastered. Only if the robot has been mastered can it move to programmed positions and be moved using Cartesian coordinates. During mastering, the mechanical position and the electronic position of the robot are aligned. For this purpose, the robot is moved to a defined mechanical position, the mastering position. The encoder value for each axis is then saved.

The mastering position is similar, but not identical, for all robots. The exact positions may even vary between individual robots of a single robot type.

![Fig. 5-2: Mastering position – approximate position](image)

A robot must be mastered in the following cases:

<table>
<thead>
<tr>
<th>Case</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>During commissioning</td>
<td>- - -</td>
</tr>
<tr>
<td>After maintenance work during which the robot loses its mastering, e.g. exchange of motor or RDC</td>
<td>&gt;&gt;&gt; 5.5.6 &quot;Reference mastering&quot; Page 80</td>
</tr>
<tr>
<td>When the robot has been moved without the robot controller (e.g. with the release device)</td>
<td>- - -</td>
</tr>
</tbody>
</table>
5.5.1 Mastering methods

Overview
A robot can be mastered in the following ways:

- With the EMD (Electronic Mastering Device)
  (>>> 5.5.3 "Mastering with the EMD" Page 73)
- With the dial gauge
  (>>> 5.5.4 "Mastering with the dial gauge" Page 78)

The axes must be moved to the pre-mastering position before every mastering operation.

<table>
<thead>
<tr>
<th>Case</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>After exchanging a gear unit</td>
<td>Before carrying out a new mastering procedure, the old mastering data must first be deleted! Mastering data are deleted by manually unmast-</td>
</tr>
<tr>
<td>After an impact with an end stop at</td>
<td>ering the axes. (&gt;&gt;&gt; 5.5.7 &quot;Manually unmastering axes&quot; Page 81)</td>
</tr>
<tr>
<td>more than 250 mm/s</td>
<td></td>
</tr>
<tr>
<td>After a collision</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 5-3: Moving an axis to the pre-mastering position

The mastering marks are situated in the following positions on the robot:
### Precondition

- The jog mode “Jog keys” is active.
- Operating mode T1

### Procedure

1. Select **Axes** as the coordinate system for the jog keys.
2. Hold down the enabling switch.
   - Axes A1 to A6 are displayed next to the jog keys.
3. Press the Plus or Minus jog key to move an axis in the positive or negative direction.
4. Move each axis, starting from A1 and working upwards, so that the mastering marks line up.

### 5.5.3 Mastering with the EMD

#### Overview

In EMD mastering, the axis is automatically moved by the robot controller to the mastering position. Mastering is carried out first without and then with a load. It is possible to save mastering data for different loads.
EMD mastering consists of the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | First mastering  
  (>>> 5.5.3.1 "First mastering with the EMD" Page 74)  
  First mastering is carried out without a load. |
| 2    | Teach offset  
  (>>> 5.5.3.2 "Teach offset" Page 76)  
  "Teach offset" is carried out with a load. The difference from the first mastering is saved. |
| 3    | If required: Master load with offset  
  (>>> 5.5.3.3 "Master load with offset" Page 77)  
  "Load mastering with offset" is carried out with a load for which an offset has already been taught.  
  Area of application:  
  ■ Checking first mastering  
  ■ Restoring first mastering if it has been lost (e.g. following exchange of motor or collision). Since an offset that has been taught is retained, even if mastering is lost, the robot controller can calculate the first mastering. |

**5.5.3.1 First mastering with the EMD**

**Precondition**  
- There is no load on the robot; i.e. there is no tool, workpiece or supplementary load mounted.  
- All axes are in the pre-mastering position.  
- No program is selected.  
- Operating mode T1

**Procedure**

The EMD must always be screwed onto the gauge cartridge without the signal cable attached. Only then may the signal cable be attached to the EMD. Otherwise, the signal cable could be damaged. Similarly, when removing the EMD, the signal cable must always be removed from the EMD first. Only then may the EMD be removed from the gauge cartridge. After mastering, remove the signal cable from connection X32. Failure to do so may result in interference signals or damage.

1. Select Start-up > Master > EMD > With load correction > First mastering in the main menu.  
   A window opens. All axes to be mastered are displayed. The axis with the lowest number is highlighted.  
2. Remove the cover from connection X32.
3. Connect the signal cable to X32.

4. Remove the protective cap of the gauge cartridge on the axis highlighted in the window. (Turned around, the EMD can be used as a screwdriver.)

5. Screw the EMD onto the gauge cartridge.
6. Attach the signal cable to the EMD, aligning the red dot on the connector with the groove in the EMD.

7. Press Master.

8. Press an enabling switch and the Start key.
   
   When the EMD has passed through the reference notch, the mastering position is calculated. The robot stops automatically. The values are saved.
   
   The axis is no longer displayed in the window.

9. Remove the signal cable from the EMD. Then remove the EMD from the gauge cartridge and replace the protective cap.

10. Repeat steps 4 to 9 for all axes to be mastered.

11. Close the window.

12. Remove signal cable from connection X32.

5.5.3.2 Teach offset

Description

“Teach offset” is carried out with a load. The difference from the first mastering is saved.

If the robot is operated with different loads, “Teach offset” must be carried out for every load. In the case of grippers used for picking up heavy workpieces, “Teach offset” must be carried out for the gripper both with and without the workpiece.

Fig. 5-8: Screwing EMD onto gauge cartridge

Fig. 5-9: Attaching signal cable to EMD
Start-up and recommissioning

Precondition
- Same ambient conditions (temperature, etc.) as for first mastering.
- The load is mounted on the robot.
- All axes are in the pre-mastering position.
- No program is selected.
- Operating mode T1

Procedure

1. Select **Start-up > Master > EMD > With load correction > Teach offset** in the main menu.
2. Enter tool number. Confirm with **Tool OK**.
   - A window opens. All axes for which the tool has not yet been taught are displayed. The axis with the lowest number is highlighted.
3. Remove the cover from connection X32 and connect the signal cable.
4. Remove the protective cap of the gauge cartridge on the axis highlighted in the window. (Turned around, the EMD can be used as a screwdriver.)
5. Screw the EMD onto the gauge cartridge.
6. Attach the signal cable to the EMD, aligning the red dot on the connector with the groove in the EMD.
7. Press **Learn**.
8. Press an enabling switch and the Start key.
   - When the EMD has passed through the reference notch, the mastering position is calculated. The robot stops automatically. A window opens. The deviation of this axis from the first mastering is indicated in degrees and increments.
9. Click **OK** to confirm. The axis is no longer displayed in the window.
10. Remove the signal cable from the EMD. Then remove the EMD from the gauge cartridge and replace the protective cap.
11. Repeat steps 4 to 10 for all axes to be mastered.
12. Close the window.
13. Remove signal cable from connection X32.

5.5.3.3 Master load with offset

Description
Area of application:
- Checking first mastering
- Restoring first mastering if it has been lost (e.g. following exchange of motor or collision). Since an offset that has been taught is retained, even if mastering is lost, the robot controller can calculate the first mastering.

An axis can only be checked if all axes with lower numbers have been mastered.

Precondition
- Same ambient conditions (temperature, etc.) as for first mastering.
- A load for which “Teach offset” has been carried out is mounted on the robot.

NOTICE: The EMD must always be screwed onto the gauge cartridge without the signal cable attached. Only then may the signal cable be attached to the EMD. Otherwise, the signal cable could be damaged.

Similarly, when removing the EMD, the signal cable must always be removed from the EMD first. Only then may the EMD be removed from the gauge cartridge.

After mastering, remove the signal cable from connection X32. Failure to do so may result in interference signals or damage.

- The EMD must always be screwed onto the gauge cartridge without the signal cable attached. Only then may
  the signal cable be attached to the EMD. Otherwise, the signal cable could be damaged.

Similarly, when removing the EMD, the signal cable must always be removed from the EMD first. Only then may the EMD be removed from the gauge cartridge.

After mastering, remove the signal cable from connection X32. Failure to do so may result in interference signals or damage.
- All axes are in the pre-mastering position.
- No program is selected.
- Operating mode T1

**Procedure**

**NOTICE** The EMD must always be screwed onto the gauge cartridge without the signal cable attached. Only then may the signal cable be attached to the EMD. Otherwise, the signal cable could be damaged.

Similarly, when removing the EMD, the signal cable must always be removed from the EMD first. Only then may the EMD be removed from the gauge cartridge.

After mastering, remove the signal cable from connection X32. Failure to do so may result in interference signals or damage.

1. In the main menu, select **Start-up > Master > EMD > With load correction > Master load > With offset**.
2. Enter tool number. Confirm with **Tool OK**.
   
   A window opens. All axes for which an offset has been taught with this tool are displayed. The axis with the lowest number is highlighted.

3. Remove the cover from connection X32 and connect the signal cable.

4. Remove the protective cap of the gauge cartridge on the axis highlighted in the window. (Turned around, the EMD can be used as a screwdriver.)

5. Screw the EMD onto the gauge cartridge.

6. Attach the signal cable to the EMD, aligning the red dot on the connector with the groove in the EMD.

7. Press **Check**.

8. Hold down an enabling switch and press the Start key.
   
   When the EMD has passed through the reference notch, the mastering position is calculated. The robot stops automatically. The difference from “Teach offset” is displayed.

9. If required, press **Save** to save the values. The old mastering values are deleted.
   
   To restore a lost first mastering, always save the values.

10. Remove the signal cable from the EMD. Then remove the EMD from the gauge cartridge and replace the protective cap.

11. Repeat steps 4 to 10 for all axes to be mastered.

12. Close the window.

13. Remove signal cable from connection X32.

### 5.5.4 Mastering with the dial gauge

**Description** In dial mastering, the axis is moved manually by the user to the mastering position. Mastering is always carried out with a load. It is not possible to save mastering data for different loads.
Precondition
- The load is mounted on the robot.
- All axes are in the pre-mastering position.
- The jog mode “Jog keys” is active and the coordinate system Axis has been selected.
- No program is selected.
- Operating mode T1

Procedure
1. In the main menu, select Start-up > Master > Dial.
   A window opens. All axes that have not been mastered are displayed. The axis that must be mastered first is selected.
2. Remove the protective cap from the gauge cartridge on this axis and mount the dial gauge on the gauge cartridge.
   Using the Allen key, loosen the screws on the neck of the dial gauge. Turn the dial so that it can be viewed easily. Push the pin of the dial gauge in as far as the stop.
   Using the Allen key, tighten the screws on the neck of the dial gauge.
3. Reduce jog override to 1%.
4. Jog axis from “+” to “-”. At the lowest position of the reference notch, recognizable by the change in direction of the pointer, set the dial gauge to 0.
   If the axis inadvertently overshoots the lowest position, jog the axis backwards and forwards until the lowest position is reached. It is immaterial whether the axis is moved from “+” to “-” or from “-” to “+”.
5. Move the axis back to the pre-mastering position.
6. Move the axis from “+” to “-” until the pointer is about 5-10 scale divisions before zero.
7. Switch to incremental jogging.
8. Move the axis from “+” to “-” until zero is reached.
   If the axis overshoots zero, repeat steps 5 to 8.
9. Press Master. The axis that has been mastered is removed from the window.
10. Remove the dial gauge from the gauge cartridge and replace the protective cap.
11. Switch back from incremental jogging to the normal jog mode.
12. Repeat steps 2 to 11 for all axes to be mastered.
13. Close the window.
5.5.5 Mastering external axes

Description
- KUKA external axes can be mastered using either the EMD or the dial gauge.
- Non-KUKA external axes can be mastered using the dial gauge. If mastering with the EMD is desired, the external axis must be fitted with gauge cartridges.

Procedure
- The procedure for mastering external axes is the same as that for mastering robot axes. Alongside the robot axes, the configured external axes now also appear in the axis selection window.

Fig. 5-11: Selection list of axes to be mastered

5.5.6 Reference mastering

Description
Reference mastering is suitable if maintenance work is due on a correctly mastered robot and it is to be expected that the robot will lose its mastering. Examples:
- Exchange of RDC
- Exchange of motor

The robot is moved to the $MAMES position before the maintenance work is commenced. Afterwards, the axis values of this system variable are reassigned to the robot by means of reference mastering. The state of the robot is then the same as before the loss of mastering. Taught offsets are retained. No EMD or dial gauge is required.

In the case of reference mastering, it is irrelevant whether or not there is a load mounted on the robot. Reference mastering can also be used for external axes.
Preparation

- Move the robot to the $MAMES position before commencing the maintenance work. To do so, program a point PTP $MAMES and move the robot to it. This is only possible in the user group "Expert"!

**WARNING**
The robot must not move to the default HOME position instead of to $MAMES. $MAMES may be, but is not always, identical to the default HOME position. Only in the $MAMES position will the robot be correctly mastered by means of reference mastering. If the robot is reference mastered at any position other than $MAMES, this may result in physical injury and material damage.

Precondition

- No program is selected.
- The position of the robot was not changed during the maintenance work.
- If the RDC has been exchanged: the robot data have been transferred from the hard drive to the RDC (this can only be done in the user group "Expert"!)

Procedure

1. In the main menu, select **Start-up > Master > Reference**.
   The option window Reference mastering is opened. All axes that have not been mastered are displayed. The axis that must be mastered first is selected.
2. Press **Master**. The selected axis is mastered and removed from the option window.
3. Repeat step 2 for all axes to be mastered.

5.5.7 Manually unmastering axes

Description

The mastering values of the individual axes can be deleted. The axes do not move during unmastering.

**NOTICE**
The software limit switches of an unmastered robot are deactivated. The robot can hit the end stop buffers, thus damaging the robot and making it necessary to exchange the buffers. An unmastered robot must not be jogged, if at all avoidable. If it must be jogged, the jog override must be reduced as far as possible.

Precondition

- No program is selected.

Procedure

1. In the main menu, select **Start-up > Master > Unmaster**. A window opens.
2. Select the axis to be unmastered.
3. Press **Unmaster**. The mastering data of the axis are deleted.
4. Repeat steps 2 and 3 for all axes to be unmastered.
5. Close the window.
5.6 Calibration

5.6.1 Tool calibration

Description

During tool calibration, the user assigns a Cartesian coordinate system (TOOL coordinate system) to the tool mounted on the mounting flange.

The TOOL coordinate system has its origin at a user-defined point. This is called the TCP (Tool Center Point). The TCP is generally situated at the working point of the tool.

Advantages of the tool calibration:

- The tool can be moved in a straight line in the tool direction.
- The tool can be rotated about the TCP without changing the position of the TCP.
- In program mode: The programmed velocity is maintained at the TCP along the path.

A maximum of 16 TOOL coordinate systems can be saved. Variable: TOOL_DATA[1…16].

The following data are saved:

- X, Y, Z:
  Origin of the TOOL coordinate system relative to the FLANGE coordinate system
- A, B, C:
  Orientation of the TOOL coordinate system relative to the FLANGE coordinate system

Overview

Tool calibration consists of 2 steps:

Fig. 5-12: TCP calibration principle
If the calibration data are already known, they can be entered directly. (>>> 5.6.1.5 "Numeric input" Page 88)

### 5.6.1.1 TCP calibration: XYZ 4-point method

The XYZ 4-point method cannot be used for palletizing robots.

**Description**

The TCP of the tool to be calibrated is moved to a reference point from 4 different directions. The reference point can be freely selected. The robot controller calculates the TCP from the different flange positions.

The 4 flange positions at the reference point must be sufficiently different from one another.
Precondition
- The tool to be calibrated is mounted on the mounting flange.
- Operating mode T1

Procedure
1. In the main menu, select Start-up > Calibrate > Tool > XYZ 4-point.
2. Assign a number and a name for the tool to be calibrated. Confirm with Next.
4. Move the TCP to the reference point from a different direction. Press Calibrate. Confirm with Next.
5. Repeat step 4 twice.
6. Either press Save. The data are saved and the window is closed.
   Or press Load data. The data are saved and a window is opened in which the payload data can be entered.
   (>>> 5.7.3 "Entering payload data" Page 105)
   Or press ABC 2-point or ABC World. The data are saved and a window is opened in which the orientation of the TOOL coordinate system can be defined.
   (>>> 5.6.1.4 "Defining the orientation: ABC 2-point method" Page 86)
   (>>> 5.6.1.3 "Defining the orientation: ABC World method" Page 85)

5.6.1.2 TCP calibration: XYZ Reference method

Description
In the case of the XYZ Reference method, a new tool is calibrated with a tool that has already been calibrated. The robot controller compares the flange positions and calculates the TCP of the new tool.
Precondition
- A previously calibrated tool is mounted on the mounting flange.
- Operating mode T1

Preparation
Calculate the TCP data of the calibrated tool:
1. In the main menu, select Start-up > Calibrate > Tool > XYZ Reference.
2. Enter the number of the calibrated tool.
3. Note the X, Y and Z values.
4. Close the window.

Procedure
1. In the main menu, select Start-up > Calibrate > Tool > XYZ Reference.
2. Assign a number and a name for the new tool. Confirm with Next.
3. Enter the TCP data of the calibrated tool. Confirm with Next.
5. Move the tool away and remove it. Mount the new tool.
7. Either press Save. The data are saved and the window is closed.
   Or press Load data. The data are saved and a window is opened in which the payload data can be entered.
   (>>> 5.7.3 "Entering payload data" Page 105)
   Or press ABC 2-point or ABC World. The data are saved and a window is opened in which the orientation of the TOOL coordinate system can be defined.
   (>>> 5.6.1.4 "Defining the orientation: ABC 2-point method" Page 88)
   (>>> 5.6.1.3 "Defining the orientation: ABC World method" Page 85)

5.6.1.3 Defining the orientation: ABC World method

Description
The axes of the TOOL coordinate system are aligned parallel to the axes of the WORLD coordinate system. This communicates the orientation of the TOOL coordinate system to the robot controller.

There are 2 variants of this method:
- 5D: Only the tool direction is communicated to the robot controller. By default, the tool direction is the X axis. The directions of the other axes are defined by the system and cannot be detected easily by the user.
  Area of application: e.g. MIG/MAG welding, laser cutting or waterjet cutting
- 6D: The directions of all 3 axes are communicated to the robot controller.
  Area of application: e.g. for weld guns, grippers or adhesive nozzles

Precondition
- The tool to be calibrated is mounted on the mounting flange.
- The TCP of the tool has already been measured.
- Operating mode T1

The following procedure applies if the tool direction is the default tool direction (= X axis). If the tool direction has been changed to Y or Z, the procedure must also be changed accordingly.

**Procedure**

1. In the main menu, select **Start-up > Calibrate > Tool > ABC World**.
2. Enter the number of the tool. Confirm with **Next**.
3. Select a variant in the box **5D/6D**. Confirm with **Next**.
4. If **5D** is selected:
   - Align +X<sub>TOOL</sub> parallel to -Z<sub>WORLD</sub>. (+X<sub>TOOL</sub> = tool direction)
   - If **6D** is selected:
     - Align the axes of the TOOL coordinate system as follows.
       - +X<sub>TOOL</sub> parallel to -Z<sub>WORLD</sub>. (+X<sub>TOOL</sub> = tool direction)
       - +Y<sub>TOOL</sub> parallel to +Y<sub>WORLD</sub>
       - +Z<sub>TOOL</sub> parallel to +X<sub>WORLD</sub>
5. Press **Calibrate**. Confirm with **Next**.
6. Either press **Save**. The data are saved and the window is closed.
   - Or press **Load data**. The data are saved and a window is opened in which the payload data can be entered.

(>>> 5.7.3 "Entering payload data" Page 105)

### 5.6.1.4 Defining the orientation: ABC 2-point method

**Description**

The axes of the TOOL coordinate system are communicated to the robot controller by moving to a point on the X axis and a point in the XY plane.

This method is used if it is necessary to define the axis directions with particular precision.
Precondition

- The tool to be calibrated is mounted on the mounting flange.
- The TCP of the tool has already been measured.
- Operating mode T1

The following procedure applies if the tool direction is the default tool direction (= X axis). If the tool direction has been changed to Y or Z, the procedure must also be changed accordingly.

Procedure

1. In the main menu, select **Start-up > Calibrate > Tool > ABC 2-point**.
2. Enter the number of the mounted tool. Confirm with **Next**.
3. Move the TCP to any reference point. Press **Calibrate**. Confirm with **Next**.
4. Move the tool so that the reference point on the X axis has a negative X value (i.e. move against the tool direction). Press **Calibrate**. Confirm with **Next**.
5. Move the tool so that the reference point in the XY plane has a negative Y value. Press **Calibrate**. Confirm with **Next**.
6. Either press **Save**. The data are saved and the window is closed.
   
   Or press **Load data**. The data are saved and a window is opened in which the payload data can be entered.

(>>>> 5.6.1.4 "Defining the orientation: ABC 2-point method" Page 86)
5.6.1.5 Numeric input

**Description**

The tool data can be entered manually.

Possible sources of data:
- CAD
- Externally calibrated tool
- Tool manufacturer specifications

In the case of palletizing robots with 4 axes, e.g. KR 180 PA, the tool data must be entered numerically. The XYZ and ABC methods cannot be used as reorientation of these robots is highly restricted.

**Precondition**

- The following values are known:
  - X, Y and Z relative to the FLANGE coordinate system
  - A, B and C relative to the FLANGE coordinate system
  - Operating mode T1

**Procedure**

1. In the main menu, select **Start-up > Calibrate > Tool > Numeric input**.
2. Assign a number and a name for the tool to be calibrated. Confirm with **Next**.
3. Enter data. Confirm with **Next**.
4. Either press **Save**. The data are saved and the window is closed.
   Or press **Load data**. The data are saved and a window is opened in which the payload data can be entered.
   (>>> 5.7.3 "Entering payload data" Page 105)

5.6.2 Base calibration

**Description**

During base calibration, the user assigns a Cartesian coordinate system (BASE coordinate system) to a work surface or the workpiece. The BASE coordinate system has its origin at a user-defined point.

Advantages of base calibration:
- The TCP can be jogged along the edges of the work surface or workpiece.
- Points can be taught relative to the base. If it is necessary to offset the base, e.g. because the work surface has been offset, the points move with it and do not need to be retaught.

A maximum of 32 BASE coordinate systems can be saved. Variable: BASE_DATA[1...32].

**Overview**

There are 2 ways of calibrating a base:
- 3-point method (>>> 5.6.2.1 "3-point method" Page 89)
- Indirect method (>>> 5.6.2.2 "Indirect method" Page 90)

If the calibration data are already known, they can be entered directly.
(>>> 5.6.2.3 "Numeric input" Page 90)
5.6.2.1 3-point method

**Description**
The robot moves to the origin and 2 further points of the new base. These 3 points define the new base.

**Precondition**
- A previously calibrated tool is mounted on the mounting flange.
- Operating mode T1

**Procedure**
1. In the main menu, select **Start-up > Calibrate > Base > ABC 3-point**.
2. Assign a number and a name for the base. Confirm with **Next**.
3. Enter the number of the mounted tool. Confirm with **Next**.
4. Move the TCP to the origin of the new base. Press **Calibrate**. Confirm with **Next**.
5. Move the TCP to a point on the positive X axis of the new base. Press **Calibrate**. Confirm with **Next**.
6. Move the TCP to a point in the XY plane with a positive Y value. Press **Calibrate**. Confirm with **Next**.
7. Press **Save**.
5.6.2.2 Indirect method

Description

The indirect method is used if it is not possible to move to the origin of the base, e.g. because it is inside a workpiece or outside the workspace of the robot.

The TCP is moved to 4 points in the base, the coordinates of which must be known. The robot controller calculates the base from these points.

Fig. 5-17: Indirect method

Precondition

- A calibrated tool is mounted on the mounting flange.
- The coordinates of 4 points in the new base are known, e.g. from CAD data. The 4 points are accessible to the TCP.
- Operating mode T1

Procedure

1. In the main menu, select Start-up > Calibrate > Base > Indirect.
2. Assign a number and a name for the base. Confirm with Next.
3. Enter the number of the mounted tool. Confirm with Next.
4. Enter the coordinates of a known point in the new base and move the TCP to this point. Press Calibrate. Confirm with Next.
5. Repeat step 4 three times.
6. Press Save.

5.6.2.3 Numeric input

Precondition

- The following numerical values are known, e.g. from CAD data:
  - Distance between the origin of the base and the origin of the WORLD coordinate system
  - Rotation of the base axes relative to the WORLD coordinate system
- Operating mode T1

Procedure

1. In the main menu, select Start-up > Calibrate > Base > Numeric input.
2. Assign a number and a name for the base. Confirm with Next.
3. Enter data. Confirm with Next.
4. Press Save.

### 5.6.3 Fixed tool calibration

#### Overview

Calibration of a fixed tool consists of 2 steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | **Calibration of the TCP of the fixed tool**<br>The TCP of a fixed tool is called an external TCP. If the calibration data are already known, they can be entered directly.  
(>>> 5.6.3.1 "Calibrating an external TCP" Page 91)<br>(>>> 5.6.3.2 "Entering the external TCP numerically" Page 93) |
| 2    | **Calibration of the workpiece**<br>The following methods are available:  
- Direct method  
(>>> 5.6.3.3 "Workpiece calibration: direct method" Page 93)<br>  
- Indirect method  
(>>> 5.6.3.4 "Workpiece calibration: indirect method" Page 94) |

The robot controller saves the external TCP as the BASE coordinate system and the workpiece as the TOOL coordinate system. A maximum of 32 BASE coordinate systems and 16 TOOL coordinate systems can be saved.

#### 5.6.3.1 Calibrating an external TCP

##### Description

First of all, the TCP of the fixed tool is communicated to the robot controller. This is done by moving a calibrated tool to it.

Then, the orientation of the coordinate system of the fixed tool is communicated to the robot controller. For this purpose, the coordinate system of the calibrated tool is aligned parallel to the new coordinate system. There are 2 variants:

- **5D**: Only the tool direction of the fixed tool is communicated to the robot controller. By default, the tool direction is the X axis. The orientation of the other axes is defined by the system and cannot be detected easily by the user.
- **6D**: The orientation of all 3 axes is communicated to the robot controller.
Precondition

- A previously calibrated tool is mounted on the mounting flange.
- Operating mode T1

The following procedure applies if the tool direction is the default tool direction (= X axis). If the tool direction has been changed to Y or Z, the procedure must also be changed accordingly.

Procedure

1. In the main menu, select Start-up > Calibrate > Fixed tool > Tool.
2. Assign a number and a name for the fixed tool. Confirm with Next.
3. Enter the number of the calibrated tool. Confirm with Next.
4. Select a variant in the box 5D/6D. Confirm with Next.
5. Move the TCP of the calibrated tool to the TCP of the fixed tool. Press Calibrate. Confirm with Next.
6. If 5D is selected:
Align \( X_{BASE} \parallel -Z_{FLANGE} \).
(i.e. align the mounting flange perpendicular to the tool direction of the fixed tool.)

If 6D is selected:
Align the mounting flange so that its axes are parallel to the axes of the fixed tool:
- \( X_{BASE} \parallel -Z_{FLANGE} \)
  (i.e. align the mounting flange perpendicular to the tool direction.)
- \( Y_{BASE} \parallel Y_{FLANGE} \)
- \( Z_{BASE} \parallel X_{FLANGE} \)

8. Press Save.

5.6.3.2 Entering the external TCP numerically

**Precondition**
- The following numerical values are known, e.g. from CAD data:
  - Distance between the TCP of the fixed tool and the origin of the WORLD coordinate system (X, Y, Z)
  - Rotation of the axes of the fixed tool relative to the WORLD coordinate system (A, B, C)
  - Operating mode T1

**Procedure**
1. In the main menu, select Start-up > Calibrate > Fixed tool > Numeric input.
2. Assign a number and a name for the fixed tool. Confirm with Next.
3. Enter data. Confirm with Next.
4. Press Save.

5.6.3.3 Workpiece calibration: direct method

**Description**
The origin and 2 further points of the workpiece are communicated to the robot controller. These 3 points uniquely define the workpiece.

Fig. 5-20
Precondition
- The workpiece is mounted on the mounting flange.
- A previously calibrated fixed tool is mounted.
- Operating mode T1

Procedure
1. In the main menu, select **Start-up** > **Calibrate** > **Fixed tool** > **Workpiece** > **Direct calibration**.
2. Assign a number and a name for the workpiece. Confirm with **Next**.
3. Enter the number of the fixed tool. Confirm with **Next**.
4. Move the origin of the workpiece coordinate system to the TCP of the fixed tool. Press **Calibrate**. Confirm with **Next**.
5. Move a point on the positive X axis of the workpiece coordinate system to the TCP of the fixed tool. Press **Calibrate**. Confirm with **Next**.
6. Move a point with a positive Y value in the XY plane of the workpiece coordinate system to the TCP of the fixed tool. Press **Calibrate**. Confirm with **Next**.
7. Either press **Save**. The data are saved and the window is closed.
   Or press **Load data**. The data are saved and a window is opened in which the payload data can be entered.

   (>>> 5.7.3 "Entering payload data" Page 105)

5.6.3.4 Workpiece calibration: indirect method

Description
The robot controller calculates the workpiece on the basis of 4 points whose coordinates must be known. The robot does not move to the origin of the workpiece.
Precondition

- A previously calibrated fixed tool is mounted.
- The workpiece to be calibrated is mounted on the mounting flange.
- The coordinates of 4 points of the new workpiece are known, e.g. from CAD data. The 4 points are accessible to the TCP.
- Operating mode T1

Procedure

1. In the main menu, select Start-up > Calibrate > Fixed tool > Workpiece > Indirect calibration.
2. Assign a number and a name for the workpiece. Confirm with Next.
3. Enter the number of the fixed tool. Confirm with Next.
4. Enter the coordinates of a known point on the workpiece and move this point to the TCP of the fixed tool. Press Calibrate. Confirm with Next.
5. Repeat step 4 three times.
6. Either press Save. The data are saved and the window is closed.

Or press Load data. The data are saved and a window is opened in which the payload data can be entered.

(>>> 5.7.3 "Entering payload data" Page 105)

5.6.4 Renaming the tool/base

Precondition

- Operating mode T1

Procedure

1. In the main menu, select Start-up > Calibrate > Tool or Base > Change name.
2. Select the tool or base and press Name.
3. Enter the new name and confirm with Save.

5.6.5 Linear unit

The KUKA linear unit is a self-contained, one-axis linear unit mounted on the floor or ceiling. It is used for linear traversing of the robot and is controlled by the robot controller as an external axis.
The linear unit is a ROBROOT kinematic system. When the linear unit is moved, the position of the robot in the WORLD coordinate system changes. The current position of the robot in the WORLD coordinate system is defined by the vector $ROBROOT_C$.

$ROBROOT_C$ consists of:

- $ERSYSROOT$ (static component)
  
  Root point of the linear unit relative to $WORLD$. The root point is situated by default at the zero position of the linear unit and is not dependent on $MAMES$.

- $ERSYS$ (dynamic component)
  
  Current position of the robot on the linear unit relative to the $ERSYSROOT$.

### 5.6.5.1 Checking whether the linear unit needs to be calibrated

**Description**

The robot is standing on the flange of the linear unit. Ideally, the ROBROOT coordinate system of the robot should be identical to the FLANGE coordinate system of the linear unit. In reality, there are often slight discrepancies which mean that positions cannot be moved to correctly. Calibration allows mathematical correction of these discrepancies. (Rotations about the direction of motion of the linear unit cannot be corrected. They do not, however, cause errors when moving to positions.)

If there are no discrepancies, the linear unit does not need to be calibrated. The following procedure can be used to determine whether calibration is required.

**Precondition**

- The machine data of the linear unit have been configured and loaded into the robot controller.
- A previously calibrated tool is mounted on the mounting flange.
- No program is open or selected.
- Operating mode T1

**Procedure**

1. Align the TCP against a freely selected point and observe it.
2. Execute a Cartesian (not axis-specific!) motion with the linear unit.
   
   - If the TCP stops: the linear unit does not require calibration.
   - If the TCP moves: the linear unit does require calibration.

   (>>> 5.6.5.2 "Calibrating the linear unit" Page 97)

---

![Fig. 5-23: ROBROOT kinematic system – linear unit](image)
If the calibration data are already known (e.g. from CAD), they can be entered directly. (>>> 5.6.5.3 “Entering the linear unit numerically” Page 98)

5.6.5.2 Calibrating the linear unit

Description
During calibration, the TCP of a tool that has already been calibrated is moved to a reference point 3 times.

- The reference point can be freely selected.
- The position of the robot on the linear unit from which the reference point is approached must be different all 3 times. The 3 positions must be far enough apart.

The correction values determined by the calibration are factored into the system variable $ETx_TFLA3.

Precondition
- The machine data of the linear unit have been configured and loaded into the robot controller.
- A previously calibrated tool is mounted on the mounting flange.
- No program is open or selected.
- Operating mode T1

Procedure
1. In the main menu, select Start-up > Calibrate > External kinematic system > Linear unit.
   The robot controller detects the linear unit automatically and displays the following data:
   - Ext. kinematic system no.: number of the external kinematic system (1 … 6) ($EX_KIN)
   - Axis: number of the external axis (1 … 6) ($ETx_AX)
   - Name of the external kinematic system ($ETx_NAME)
   (If the robot controller is unable to determine these values, e.g. because the linear unit has not yet been configured, calibration cannot be continued.)
2. Move the linear unit with the jog key “+”.
3. Specify whether the linear unit is moving to “+” or “-”. Confirm with Next.
4. Move the TCP to the reference point.
5. Press Calibrate.
6. Repeat steps 4 and 5 twice, but move the linear unit first each time in order to address the reference point from different positions.
7. Press Save. The calibration data are saved.
8. The system asks whether the positions that have already been taught are to be corrected.
   - If no positions have been taught prior to the calibration, it makes no difference whether the question is answered with Yes or No.
   - If positions have been taught prior to the calibration:
     Answering Yes will cause positions with base 0 to be corrected automatically. Other positions will not be corrected!
     Answering No will cause no positions to be corrected.

**NOTICE**
After calibration of a linear unit, the following safety measures must be carried out:

1. Check the software limit switches of the linear unit and adapt them if required.
2. Test programs in T1.
   Damage to property may otherwise result.
5.6.5.3 Entering the linear unit numerically

Precondition
- The machine data of the linear unit have been configured and loaded into the robot controller.
- No program is open or selected.
- The following numerical values are known, e.g. from CAD data:
  - Distance between the robot base flange and the origin of the ERSYS-ROOT coordinate system (X, Y, Z)
  - Orientation of the robot base flange relative to the ERSYSROOT coordinate system (A, B, C)
- Operating mode T1

Procedure
1. In the main menu, select Start-up > Calibrate > External kinematic system > Linear unit (numeric).
   The robot controller detects the linear unit automatically and displays the following data:
   - **Ext. kinematic system no.**: number of the external kinematic system (1 … 6)
   - **Axis**: number of the external axis (1 … 6)
   - **Name of the kinematic system**
     (If the robot controller is unable to determine these values, e.g. because the linear unit has not yet been configured, calibration cannot be continued.)
2. Move the linear unit with the jog key “+”.
3. Specify whether the linear unit is moving to “+” or “-”. Confirm with Next.
4. Enter data. Confirm with Next.
5. Press Save. The calibration data are saved.
6. The system asks whether the positions that have already been taught are to be corrected.
   - If no positions have been taught prior to the calibration, it makes no difference whether the question is answered with Yes or No.
   - If positions have been taught prior to the calibration:
     Answering Yes will cause positions with base 0 to be corrected automatically. Other positions will not be corrected!
     Answering No will cause no positions to be corrected.

5.6.6 Calibrating an external kinematic system

Description
Calibration of the external kinematic system is necessary to enable the motion of the axes of the kinematic system to be synchronized and mathematically coupled with the robot axes. An external kinematic system can be a turn-tilt table or positioner, for example.

After calibration of a linear unit, the following safety measures must be carried out:
1. Check the software limit switches of the linear unit and adapt them if required.
2. Test programs in T1.
Damage to property may otherwise result.

For linear units, the type of calibration described here must not be used. A separate type of calibration must be used for linear units.

(>>> 5.6.5 "Linear unit" Page 95)
Overview

Calibration of an external kinematic system consists of 2 steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | Calibrate the root point of the external kinematic system.  

(>>> 5.6.6.1 "Calibrating the root point" Page 99)  
If the calibration data are already known, they can be entered directly.  

(>>> 5.6.6.2 "Entering the root point numerically" Page 100) |
| 2    | If there is a workpiece on the external kinematic system: calibrate the base of the workpiece.  

(>>> 5.6.6.3 "Workpiece base calibration" Page 101)  
If the calibration data are already known, they can be entered directly.  

(>>> 5.6.6.4 "Entering the workpiece base numerically" Page 103)  
If there is a tool mounted on the external kinematic system: calibrate the external tool.  

(>>> 5.6.6.5 "Calibrating an external tool" Page 103)  
If the calibration data are already known, they can be entered directly.  

(>>> 5.6.6.6 "Entering the external tool numerically" Page 104) |

5.6.6.1 Calibrating the root point

Description

In order to be able to move the robot with a mathematical coupling to a kinematic system, the robot must know the precise location of the kinematic system. This location is determined by means of root point calibration.

The TCP of a tool that has already been calibrated is moved to a reference point on the kinematic system 4 times. The position of the reference point must be different each time. This is achieved by moving the axes of the kinematic system. The robot controller uses the different positions of the reference point to calculate the root point of the kinematic system.

In the case of external kinematic systems from KUKA, the reference point is configured in the system variable $ETx_TPINF in the machine data. This contains the position of the reference point relative to the FLANGE coordinate system of the kinematic system. (x = number of the kinematic system.) The reference point is also marked on the kinematic system. During calibration, this reference point must be addressed.

In the case of non-KUKA external kinematic systems, the reference point must be configured in the machine data.

The robot controller saves the coordinates of the root point as the BASE coordinate system.
Precondition

- The machine data of the kinematic system have been configured and loaded into the robot controller.
- The number of the external kinematic system is known.
- A previously calibrated tool is mounted on the mounting flange.
- If $ET_x_{TPINFL}$ is to be modified: user group “Expert”
- Operating mode T1

Procedure

1. In the main menu, select **Start-up > Calibrate > External kinematic system > Root point**.
2. Select the number of the BASE coordinate system the root point is to be saved as. Confirm with **Next**.
3. Enter the number of the external kinematic system.
4. Assign a name for the external kinematic system. Confirm with **Next**.
5. Enter the number of the reference tool. Confirm with **Next**.
6. The value of $ET_x_{TPINFL}$ is displayed.
   - If the value is not correct: the value can be modified here in the user group “Expert”.
   - If the value is correct: confirm with **Next**.
7. Move the TCP to the reference point.
8. Press **Calibrate**. Confirm with **Next**.
9. Repeat steps 7 and 8 three times. Each time, move the kinematic system first so that the reference point is approached from different positions.
10. Press **Save**.

### 5.6.6.2 Entering the root point numerically

**Precondition**

- The following numerical values are known, e.g. from CAD data:
  - Distance between the origin of the ROOT coordinate system and the origin of the WORLD coordinate system (X, Y, Z)
  - Orientation of the ROOT coordinate system relative to the WORLD coordinate system (A, B, C)
The number of the external kinematic system is known.

Operating mode T1

Procedure

1. In the main menu, select **Start-up > Calibrate > External kinematic system > Root point (numeric)**.
2. Select the number of the BASE coordinate system the root point is to be saved as. Confirm with **Next**.
3. Enter the number of the external kinematic system.
4. Assign a name for the external kinematic system. Confirm with **Next**.
   (The name is automatically also assigned to the BASE coordinate system.)
5. Enter the data of the ROOT coordinate system. Confirm with **Next**.
6. Press **Save**.

5.6.6.3 Workpiece base calibration

Description

During this calibration, the user assigns a BASE coordinate system to a workpiece located on the kinematic system. This BASE coordinate system is relative to the FLANGE coordinate system of the kinematic system. The base is thus a moving base that moves in the same way as the kinematic system.

It is not strictly necessary to calibrate a base. If none is calibrated, the FLANGE coordinate system of the kinematic system is taken as the base.

During calibration, the TCP of a calibrated tool is moved to the origin and 2 other points of the desired base. These 3 points define the base. Only one base can be calibrated per kinematic system.
Precondition

- The machine data of the kinematic system have been configured and loaded into the robot controller.
- A previously calibrated tool is mounted on the mounting flange.
- The root point of the external kinematic system has been calibrated.
- The number of the external kinematic system is known.
- Operating mode T1

Procedure

1. In the main menu, select **Start-up > Calibrate > External kinematic system > Offset**.
2. Enter the number of the BASE coordinate system the root point was saved as. The name of the BASE coordinate system is displayed. Confirm with **Next**.
3. Enter the number of the external kinematic system. The name of the external kinematic system is displayed. Confirm with **Next**.

---

**Fig. 5-25: Base calibration principle**
4. Enter the number of the reference tool. Confirm with **Next**.
5. Move the TCP to the origin of the workpiece base. Press **Calibrate** and confirm with **Next**.
6. Move the TCP to a point on the positive X axis of the workpiece base. Press **Calibrate** and confirm with **Next**.
7. Move the TCP to a point in the XY plane with a positive Y value. Press **Calibrate** and confirm with **Next**.
8. Press **Save**.

### 5.6.6.4 Entering the workpiece base numerically

**Precondition**
- The following numerical values are known, e.g. from CAD data:
  - Distance between the origin of the workpiece base and the origin of the FLANGE coordinate system of the kinematic system (X, Y, Z)
  - Rotation of the axes of the workpiece base relative to the FLANGE coordinate system of the kinematic system (A, B, C)
- The root point of the external kinematic system has been calibrated.
- The number of the external kinematic system is known.
- Operating mode T1

**Procedure**
1. In the main menu, select **Start-up > Calibrate > External kinematic system > Offset (numeric)**.
2. Enter the number of the BASE coordinate system the root point was saved as. The name of the BASE coordinate system is displayed. Confirm with **Next**.
3. Enter the number of the external kinematic system. The name of the external kinematic system is displayed. Confirm with **Next**.
4. Enter data. Confirm with **Next**.
5. Press **Save**.

### 5.6.6.5 Calibrating an external tool

**Description**
During calibration of the external tool, the user assigns a coordinate system to the tool mounted on the kinematic system. This coordinate system has its origin in the TCP of the external tool and is relative to the FLANGE coordinate system of the kinematic system.

First of all, the user communicates to the robot controller the TCP of the tool mounted on the kinematic system. This is done by moving a calibrated tool to the TCP.

Then, the orientation of the coordinate system of the tool is communicated to the robot controller. For this purpose, the user aligns the coordinate system of the calibrated tool parallel to the new coordinate system. There are 2 variants:

- **5D**: The user communicates the tool direction to the robot controller. By default, the tool direction is the X axis. The orientation of the other axes is defined by the system and cannot be influenced by the user. The system always defines the orientation of the other axes in the same way. If the tool subsequently has to be calibrated again, e.g. after a crash, it is therefore sufficient to define the tool direction again. Rotation about the tool direction need not be taken into consideration.
- **6D**: The user communicates the direction of all 3 axes to the robot controller.
The robot controller saves the coordinates of the external tool as the BASE coordinate system.

**Precondition**
- The machine data of the kinematic system have been configured and loaded into the robot controller.
- A previously calibrated tool is mounted on the mounting flange.
- The root point of the external kinematic system has been calibrated.
- The number of the external kinematic system is known.
- Operating mode T1

**Procedure**
1. In the main menu, select **Start-up > Calibrate > Fixed tool > External kinematic offset**.
2. Enter the number of the BASE coordinate system the root point was saved as. The name of the BASE coordinate system is displayed. Confirm with **Next**.
3. Enter the number of the external kinematic system. The name of the external kinematic system is displayed. Confirm with **Next**.
4. Enter the number of the reference tool. Confirm with **Next**.
5. Select a variant in the box **5D/6D**. Confirm with **Next**.
6. Move the TCP of the calibrated tool to the TCP of the external tool. Press **Calibrate** and confirm with **Next**.
7. If **5D** is selected:
   Align \( +X_{\text{BASE}} \) parallel to \( -Z_{\text{FLANGE}} \).
   (i.e. align the mounting flange perpendicular to the tool direction of the external tool.)
   If **6D** is selected:
   Align the mounting flange so that its axes are parallel to the axes of the external tool:
   - \( +X_{\text{BASE}} \) parallel to \( -Z_{\text{FLANGE}} \)
     (i.e. align the mounting flange perpendicular to the tool direction of the external tool.)
   - \( +Y_{\text{BASE}} \) parallel to \( +Y_{\text{FLANGE}} \)
   - \( +Z_{\text{BASE}} \) parallel to \( +X_{\text{FLANGE}} \)
8. Press **Calibrate** and confirm with **Next**.
9. Press **Save**.

**5.6.6.6 Entering the external tool numerically**

**Precondition**
- The following numerical values are known, e.g. from CAD data:
  - Distance between the TCP of the external tool and the origin of the FLANGE coordinate system of the kinematic system \((X, Y, Z)\)
  - Rotation of the axes of the external tool relative to the FLANGE coordinate system of the kinematic system \((A, B, C)\)
■ Operating mode T1

Procedure
1. In the main menu, select **Start-up > Calibrate > Fixed tool > Numeric input**.
2. Assign a number and a name for the external tool. Confirm with **Next**.
3. Enter data. Confirm with **Next**.
4. Press **Save**.

5.7 Load data

The load data are factored into the calculation of the paths and accelerations and help to optimize the cycle times. The load data must be entered in the robot controller.

Sources
Load data can be obtained from the following sources:
- Software option KUKA.LoadDataDetermination (only for payloads on the flange)
- Manufacturer information
- Manual calculation
- CAD programs

5.7.1 Checking loads with KUKA.Load

All load data (payload and supplementary loads) must be checked with the KUKA.Load software. Exception: If the payload is checked with KUKA.LoadDataDetermination, it is not necessary to check it with KUKA.Load.

A sign-off sheet can be generated for the loads with KUKA.Load. KUKA.Load can be downloaded free of charge, complete with the documentation, from the KUKA website www.kuka.com.

More information is contained in the **KUKA.Load** documentation.

5.7.2 Calculating payloads with KUKA.LoadDataDetermination

Description
KUKA.LoadDataDetermination can be used to calculate payloads exactly and transfer them to the robot controller.

Procedure
- In the main menu, select **Start-up > Service > Load data determination**.

More information is contained in the **KUKA.LoadDataDetermination** documentation.

5.7.3 Entering payload data

Description
The payload data must be entered in the robot controller and assigned to the correct tool.

Exception: If the payload data have already been transferred to the robot controller by KUKA.LoadDataDetermination, no manual entry is required.

Precondition
- The payload data have been checked with KUKA.Load or KUKA.LoadDataDetermination and the robot is suitable for these payloads.

Procedure
1. In the main menu, select **Setup > Measure > Tool > Payload data**.
2. Enter the number of the tool in the box **Tool no.** Confirm with **Continue**.

3. Enter the payload data:
   - Box **M**: Mass
   - Boxes **X, Y, Z**: Position of the center of gravity relative to the flange
   - Boxes **A, B, C**: Orientation of the principal inertia axes relative to the flange
   - Boxes **JX, JY, JZ**: Mass moments of inertia
     (JX is the inertia about the X axis of the coordinate system that is rotated relative to the flange by A, B and C. JY and JZ are the analogous inertia about the Y and Z axes.)

4. Confirm with **Continue**.

5. Press **Save**.

### 5.7.4 Entering supplementary load data

**Description**
The supplementary load data must be entered in the robot controller.

**Reference systems of the X, Y and Z values for each supplementary load:**

<table>
<thead>
<tr>
<th>Load</th>
<th>Reference system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supplementary load A1</td>
<td>ROBROOT coordinate system A1 = 0°</td>
</tr>
<tr>
<td>Supplementary load A2</td>
<td>ROBROOT coordinate system A2 = -90°</td>
</tr>
<tr>
<td>Supplementary load A3</td>
<td>FLANGE coordinate system A4 = 0°, A5 = 0°, A6 = 0°</td>
</tr>
</tbody>
</table>

**Precondition**
- The supplementary loads have been verified with KUKA.Load and are suitable for this robot type.

**Procedure**
1. In the main menu, select **Setup > Measure > Supplementary load data**.
2. Enter the number of the axis on which the supplementary load is to be mounted. Confirm with **Continue**.
3. Enter the load data. Confirm with **Continue**.
4. Press **Save**.
6 Program management

6.1 Navigator file manager

Overview

**Fig. 6-1: Navigator**

1. Header
2. Directory structure
3. File list
4. Status bar

Description

In the Navigator, the user manages programs and system-specific files.

Header

- **Left-hand area:** the selected filter is displayed.

  (>>> 6.1.1 "Selecting filters" Page 108)

- **Right-hand area:** the directory or drive selected in the directory structure is displayed.

Directory structure

Overview of directories and drives. Exactly which directories and drives are displayed depends on the user group and configuration.

File list

The contents of the directory or drive selected in the directory structure are displayed. The manner in which programs are displayed depends on the selected filter.

The file list has the following columns:
### Status bar

The status bar can display the following information:
- Selected objects
- Action in progress
- User dialogs
- User entry prompts
- Requests for confirmation

### 6.1.1 Selecting filters

**Description**
This function is not available in the user group “User”.

The filter defines how programs are displayed in the file list. The following filters are available:

- **Detail**
  Programs are displayed as SRC and DAT files. (Default setting)

- **Modules**
  Programs are displayed as modules.

**Precondition**
- Expert user group

**Procedure**
1. Select the menu sequence **Edit > Filter**.
2. Select the desired filter in the left-hand section of the Navigator.
3. Click **OK** to confirm.

### 6.1.2 Creating a new folder

**Precondition**
- The Navigator is displayed.

**Procedure**
1. In the directory structure, select the folder in which the new folder is to be created, e.g. the folder **R1**.
   Not all folders allow the creation of new folders within them. In the user groups “Operator” and “User”, new folders can only be created in the folder **R1**.
2. Press **New**.
3. Enter a name for the folder and confirm it with **OK**.
6.1.3 Creating a new program

Precondition
- The Navigator is displayed.

Procedure
1. In the directory structure, select the folder in which the program is to be created, e.g. the folder Program. (Not all folders allow the creation of programs within them.)
3. Only in the user group “Expert”:
   - The Template selection window is opened. Select the desired template and confirm with OK.
4. Enter a name for the program and confirm it with OK.

   It is not possible to select a template in the user group “User”. By default, a program of type “Module” is created.

6.1.4 Renaming a file

Precondition
- The Navigator is displayed.

Procedure
1. In the directory structure, select the folder in which the file is located.
2. Select the file in the file list.
3. Select Edit > Rename.
4. Overwrite the file name with the new name and confirm with OK.

6.2 Selecting or opening a program

Overview
A program can be selected or opened. Instead of the Navigator, an editor is then displayed with the program.

(>>> 6.2.1 "Selecting and deselecting a program" Page 110)
(>>> 6.2.2 "Opening a program" Page 111)

It is possible to toggle backwards and forwards between the program display and the Navigator.

(>>> 6.2.3 "Toggling between the Navigator and the program" Page 111)

Differences
Program is selected:
- The block pointer is displayed.
- The program can be started.
- The program can be edited to a certain extent.
  Selected programs are particularly suitable for editing in the user group “User”.
  Example: KRL instructions covering several lines (e.g. LOOP ... END- LOOP) are not permissible.
- When the program is deselected, modifications are accepted without a request for confirmation. If impermissible modifications are programmed, an error message is displayed.

Program is opened:
- The program cannot be started.
- The program can be edited.
  Opened programs are particularly suitable for editing in the user group “Expert”.

It is not possible to select a template in the user group “User”. By default, a program of type “Module” is created.
A request for confirmation is generated when the program is closed. Modifications can be accepted or rejected.

6.2.1 Selecting and deselecting a program

- If a selected program is edited in the user group “Expert”, the cursor must then be removed from the edited line and positioned in any other line! Only in this way is it certain that the editing will be applied when the program is deselected again.

**Precondition**
- T1, T2 or AUT mode

**Procedure**
1. Select the program in the Navigator and press Select.
   - The program is displayed in the editor. It is irrelevant whether a module, an SRC file or a DAT file is selected. It is always the SRC file that is displayed in the editor.
2. Start or edit the program.
3. Deselect the program again:
   - Select Edit > Cancel program.
   - Or: In the status bar, touch the Robot interpreter status indicator. A window opens. Select Cancel program.

- When the program is deselected, modifications are accepted without a request for confirmation!

If the program is running, it must be stopped before it can be deselected.

**Description**
- If a program is selected, this is indicated by the Robot interpreter status indicator.

(>>> 6.5.6 “Robot interpreter status indicator” Page 116)

![Fig. 6-2: Program is selected](image)
6.2.2 Opening a program

Precondition
- T1, T2 or AUT mode

A program can be opened in AUT EXT mode, but not edited.

Procedure
1. Select the program in the Navigator and press Open. The program is displayed in the editor.
   - If a module has been selected, the SRC file is displayed in the editor. If an SRC file or DAT file has been selected, the corresponding file is displayed in the editor.
2. Edit the program.
3. Close the program.
4. To accept the changes, answer the request for confirmation with Yes.

Description

Fig. 6-3: Program is open

6.2.3 Toggling between the Navigator and the program

Description
- If a program is selected or open, it is possible to display the Navigator again without having to deselect or close the program. The user can then return to the program.
6.3 Structure of a KRL program

The first motion instruction in a KRL program must define an unambiguous starting position. The HOME position, which is stored by default in the robot controller, ensures that this is the case.

If the first motion instruction is not the default HOME position, or if this position has been changed, one of the following statements must be used:

- Complete PTP instruction of type POS or E6POS
- Complete PTP instruction of type AXIS or E6AXIS

The following table describes the program:

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DEF my_program( )</td>
</tr>
<tr>
<td>2</td>
<td>INI</td>
</tr>
<tr>
<td>3</td>
<td>PTP HOME Vel= 100 % DEFAULT</td>
</tr>
<tr>
<td>4</td>
<td>LIN point_5 CONT Vel= 2 m/s CPDAT1 Tool[3] Base[4]</td>
</tr>
<tr>
<td>8</td>
<td>PTP point_1 CONT Vel= 100 % PDAT1 Tool[3] Base[4]</td>
</tr>
<tr>
<td>14</td>
<td>PTP point_1 CONT Vel= 100 % PDAT1 Tool[3] Base[4]</td>
</tr>
<tr>
<td>20</td>
<td>PTP HOME Vel= 100 % DEFAULT</td>
</tr>
<tr>
<td>22</td>
<td>END</td>
</tr>
</tbody>
</table>

The DEF line indicates the name of the program. If the program is a function, the DEF line begins with "DEFFCT" and contains additional information. The DEF line can be displayed or hidden.

The INI line contains initializations for internal variables and parameters.

The HOME position.

LIN motion.

PTP motion.

The END line is the last line in any program. If the program is a function, the wording of the END line is "ENDFCT". The END line must not be deleted!
“Complete” means that all components of the end point must be specified.

**WARNING** If the HOME position is modified, this affects all programs in which it is used. Physical injuries or damage to property may result.

In programs that are used exclusively as subprograms, different statements can be used as the first motion instruction.

### 6.3.1 HOME position

The HOME position is not program-specific. It is generally used as the first and last position in the program as it is uniquely defined and uncritical.

The HOME position is stored by default with the following values in the robot controller:

<table>
<thead>
<tr>
<th>Axis</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
<th>A6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td>0°</td>
<td>-90°</td>
<td>+90°</td>
<td>0°</td>
<td>0°</td>
<td>0°</td>
</tr>
</tbody>
</table>

Additional HOME positions can be taught. A HOME position must meet the following conditions:

- Good starting position for program execution
- Good standstill position. For example, the stationary robot must not be an obstacle.

**WARNING** If the HOME position is modified, this affects all programs in which it is used. Physical injuries or damage to property may result.

### 6.4 Displaying/hiding program sections

#### 6.4.1 Displaying/hiding the DEF line

**Description**

By default, the DEF line is hidden. Declarations can only be made in a program if the DEF line is visible.

The DEF line is displayed and hidden separately for opened and selected programs. If detail view (ASCII mode) is activated, the DEF line is visible and does not need to be activated separately.

**Precondition**

- User group “Expert”
- Program is selected or open.

**Procedure**

- Select the menu sequence **Edit > View > DEF line**.
  - Check mark activated in menu: DEF line is displayed.
  - Check mark not activated in menu: DEF line is hidden.

#### 6.4.2 Activating detail view

**Description**

Detail view (ASCII mode) is deactivated by default to keep the program transparent. If detail view is activated, hidden program lines, such as the FOLD and ENDFOLD lines and the DEF line, are displayed.

Detail view is activated and deactivated separately for opened and selected programs.

**Precondition**

- Expert user group
6.4.3 Activating/deactivating the line break function

**Description**
If a line is wider than the program window, the line is broken by default. The part of the line after the break has no line number and is marked with a black, L-shaped arrow. The line break function can be deactivated.

Precondition
- User group “Expert”
- Program is selected or open.

**Procedure**

1. Select the menu sequence Edit > View > Line break.
   - Check mark activated in menu: line break function is activated.
   - Check mark not activated in menu: line break function is deactivated.

6.5 Starting a program

6.5.1 Selecting the program run mode

**Procedure**

1. Touch the Program run mode status indicator. The Program run mode window is opened.
2. Select the desired program run mode.

   (>>> 6.5.2 “Program run modes” Page 114)

   The window closes and the selected program run mode is applied.

6.5.2 Program run modes

<table>
<thead>
<tr>
<th>Program run mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Go #GO</td>
<td>The program is executed through to the end without stopping.</td>
</tr>
</tbody>
</table>
6.5.3 Advance run

The advance run is the **maximum** number of motion blocks that the robot controller calculates and plans in advance during program execution. The **actual** number is dependent on the capacity of the computer. The default value is 3. The advance run refers to the current position of the block pointer. The advance run is required, for example, in order to be able to calculate approximate positioning motions.

Certain statements trigger an advance run stop. These include statements that influence the periphery, e.g. OUT statements.

6.5.4 Setting the program override (POV)

**Description**

Program override is the velocity of the robot during program execution. The program override is specified as a percentage of the programmed velocity.

In T1 mode, the maximum velocity is 250 mm/s, irrespective of the value that is set.

**Procedure**

1. Touch the **POV/HOV** status indicator. The **Overrides** window is opened.
2. Set the desired program override. It can be set using either the plus/minus keys or by means of the slide controller.
   - Plus/minus keys: The value can be set to 100%, 75%, 50%, 30%, 10%, 3%, 1%
   - Slide controller: The override can be adjusted in 1% steps.
3. Touch the **POV/HOV** status indicator again. (Or touch the area outside the window.)
   The window closes and the selected override value is applied.

The **Jog options** window can be opened via **Options** in the **Overrides** window.

**Alternative procedure**

Alternatively, the override can be set using the plus/minus key on the right-hand side of the KCP.

The value can be set to 100%, 75%, 50%, 30%, 10%, 3%, 1%.

6.5.5 Switching drives on/off

The status of the drives is indicated in the status bar. The drives can also be switched on or off here.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Color</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Green Icon]</td>
<td>Green</td>
<td>Drives ready.</td>
</tr>
<tr>
<td>![Red Icon]</td>
<td>Red</td>
<td>Drives not ready.</td>
</tr>
</tbody>
</table>
6.5.6 Robot interpreter status indicator

<table>
<thead>
<tr>
<th>Icon</th>
<th>Color</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Gray Icon" /></td>
<td>Gray</td>
<td>No program is selected.</td>
</tr>
<tr>
<td><img src="image" alt="Yellow Icon" /></td>
<td>Yellow</td>
<td>The block pointer is situated on the first line of the selected program.</td>
</tr>
<tr>
<td><img src="image" alt="Green Icon" /></td>
<td>Green</td>
<td>The program is selected and is being executed.</td>
</tr>
<tr>
<td><img src="image" alt="Red Icon" /></td>
<td>Red</td>
<td>The selected and started program has been stopped.</td>
</tr>
<tr>
<td><img src="image" alt="Black Icon" /></td>
<td>Black</td>
<td>The block pointer is situated at the end of the selected program.</td>
</tr>
</tbody>
</table>

6.5.7 Starting a program forwards (manual)

**Precondition**
- Program is selected.
- Operating mode T1 or T2.

**Procedure**
1. Select the program run mode.
2. Hold the enabling switch down and wait until the status bar indicates “Drives ready”.
3. Carry out a BCO run: Press Start key and hold it down until the message “Programmed path reached (BCO)” is displayed in the message window. The robot stops.

![Green Icon](image)

Fig. 6-5

3. Carry out a BCO run: Press Start key and hold it down until the message “Programmed path reached (BCO)” is displayed in the message window. The robot stops.

**WARNING** A BCO run is always executed as a PTP motion from the actual position to the target position. Observe the motion to avoid collisions. The velocity is automatically reduced during the BCO run.

4. Press Start key and hold it down.
   - The program is executed with or without stops, depending on the program run mode.

To stop a program that has been started manually, release the Start key.

6.5.8 Starting a program forwards (automatic)

**Precondition**
- A program is selected.
- Operating mode Automatic (not Automatic External)

**Procedure**
1. Select the program run mode **Go**.
2. Switch on the drives.
3. Carry out a BCO run:
Press Start key and hold it down until the message “Programmed path reached (BCO)” is displayed in the message window. The robot stops.

**WARNING** A BCO run is always executed as a PTP motion from the actual position to the target position. Observe the motion to avoid collisions. The velocity is automatically reduced during the BCO run.

4. Press the Start key. The program is executed.

To stop a program that has been started in Automatic mode, press the STOP key.

### 6.5.9 Carrying out a block selection

**Description**
A program can be started at any point by means of a block selection.

**Precondition**
- Program is selected.
- Operating mode T1 or T2.

**Procedure**
1. Select the program run mode.
2. Select the motion block at which the program is to be started.
3. Press **Block selection**. The block pointer indicates the motion block.
4. Hold the enabling switch down and wait until the status bar indicates “Drives ready”:

   ![Status Bar](image)

5. Carry out a BCO run: Press Start key and hold it down until the message “Programmed path reached (BCO)” is displayed in the message window. The robot stops.

**WARNING** A BCO run is always executed as a PTP motion from the actual position to the target position. Observe the motion to avoid collisions. The velocity is automatically reduced during the BCO run.

6. The program can now be started manually or automatically. It is not necessary to carry out a BCO run again.

### 6.5.10 Starting a program backwards

**Description**
In the case of backward motion, the robot stops at every point. Approximate positioning is not possible.

**Precondition**
- Program is selected.
- Operating mode T1 or T2.

**Procedure**
1. Hold the enabling switch down and wait until the status bar indicates “Drives ready”:

   ![Status Bar](image)

2. Carry out a BCO run: Press Start key and hold it down until the message “Programmed path reached (BCO)” is displayed in the message window. The robot stops.

**WARNING** A BCO run is always executed as a PTP motion from the actual position to the target position. Observe the motion to avoid collisions. The velocity is automatically reduced during the BCO run.
3. Press Start backwards key.
4. Press Start backwards key again for each motion block.

6.5.11 Resetting a program

Description
In order to restart an interrupted program from the beginning, it must be reset. This returns the program to the initial state.

Precondition
- Program is selected.

Procedure
- Select the menu sequence Edit > Reset program.

Alternative procedure
- In the status bar, touch the Robot interpreter status indicator. A window opens.
  Select Reset program.

6.5.12 Starting Automatic External mode

Precondition
- Operating mode T1 or T2
- Inputs/outputs for Automatic External and the program CELL.SRC are configured.

Procedure
1. Select the program CELL.SRC in the Navigator. (This program is located in the folder “R1”.)
2. Set program override to 100%. (This is the recommended setting. A different value can be set if required.)
3. Carry out a BCO run:
   Hold down the enabling switch. Then press the Start key and hold it down until the message “Programmed path reached (BCO)” is displayed in the message window.

WARNING There is no BCO run in Automatic External mode. This means that the robot moves to the first programmed position after the start at the programmed (not reduced) velocity and does not stop there.

4. Select “Automatic External” mode.
5. Start the program from a higher-level controller (PLC).

To stop a program that has been started in Automatic mode, press the STOP key.

6.6 Editing a program

Overview
- A running program cannot be edited.
- Programs cannot be edited in AUT EXT mode.

If a selected program is edited in the user group “Expert”, the cursor must then be removed from the edited line and positioned in any other line!

Only in this way is it certain that the editing will be applied when the program is deselected again.
<table>
<thead>
<tr>
<th>Action</th>
<th>Possible in user group ...?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert comment or stamp</td>
<td><strong>User</strong>: Yes</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
<tr>
<td>Delete lines</td>
<td><strong>User</strong>: Yes</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
<tr>
<td>Create folds</td>
<td><strong>User</strong>: No</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
<tr>
<td>Copy</td>
<td><strong>User</strong>: No</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
<tr>
<td>Paste</td>
<td><strong>User</strong>: No</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
<tr>
<td>Insert blank lines (press the Enter key)</td>
<td><strong>User</strong>: No</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
<tr>
<td>Cut</td>
<td><strong>User</strong>: No</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
<tr>
<td>Find</td>
<td><strong>User</strong>: Yes</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
<tr>
<td>Replace</td>
<td><strong>User</strong>: No</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes (program is open, not selected)</td>
</tr>
<tr>
<td>Programming with inline forms</td>
<td><strong>User</strong>: Yes</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
<tr>
<td>KRL programming</td>
<td><strong>User</strong>: Possible to a certain extent. KRL</td>
</tr>
<tr>
<td></td>
<td>instructions covering several lines (e.g.</td>
</tr>
<tr>
<td></td>
<td>LOOP … ENDLOOP) are not permissible.</td>
</tr>
<tr>
<td></td>
<td><strong>Expert</strong>: Yes</td>
</tr>
</tbody>
</table>
6.6.1  Inserting a comment or stamp

Precondition
- Program is selected or open.
- T1, T2 or AUT mode

Procedure
1. Select the line after which the comment or stamp is to be inserted.
2. Select the menu sequence **Commands > Comment > Normal** or **Stamp**.
3. Enter the desired data. If a comment or stamp has already been entered previously, the inline form still contains the same entries.
   - In the case of a comment, the box can be cleared using **New text** ready for entry of a new text.
   - In the case of a stamp, the system time can also be updated using **New time** and the **NAME** box can be cleared using **New name**.
4. Save with **Cmd Ok**.

Description

**Comment**

![Fig. 6-6: Inline form “Comment”](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Any text</td>
</tr>
</tbody>
</table>

**Stamp**

A stamp is a comment that is extended to include the system date and time and the user ID.

![Fig. 6-7: Inline form “Stamp”](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>System date (cannot be edited)</td>
</tr>
<tr>
<td>2</td>
<td>System time</td>
</tr>
<tr>
<td>3</td>
<td>Name or ID of the user</td>
</tr>
<tr>
<td>4</td>
<td>Any text</td>
</tr>
</tbody>
</table>

6.6.2  Deleting program lines

Precondition
- Program is selected or open.
- T1, T2 or AUT mode

Procedure
1. Select the line to be deleted. (The line need not have a colored background. It is sufficient for the cursor to be in the line.)
   - If several consecutive lines are to be deleted: drag a finger or stylus across the desired area. (The area must now have a colored background.)
2. Select the menu sequence **Edit > Delete**.
3. Confirm the request for confirmation with **Yes**.

**Lines cannot be restored once they have been deleted!**
6.6.3 Additional editing functions

The following additional program editing functions can be called using **Edit**:

**Copy**
Precondition:
- Program is selected or open.
- Expert user group
- T1, T2 or AUT mode

**Paste**
Precondition:
- Program is selected or open.
- Expert user group
- T1, T2 or AUT mode

**Cut**
Precondition:
- Program is selected or open.
- Expert user group
- T1, T2 or AUT mode

**Find**
Precondition:
- Program is selected or open.

**Replace**
Precondition:
- Program has been opened.
- Expert user group

6.7 Printing a program

**Procedure**
1. Select the program in the Navigator. Multiple program selection is also possible.
2. Select the menu sequence **Edit > Print**.

6.8 Archiving and restoring data

6.8.1 Archiving overview

**Target locations**
Archiving can be performed to the following target destinations:
- USB stick in KCP or robot controller
- Network
Menu items

The following menu items are available:

<table>
<thead>
<tr>
<th>Menu item</th>
<th>Archives the directories/files</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>KRC:\</td>
</tr>
<tr>
<td>Applications</td>
<td>KRC:\R1\Program Files</td>
</tr>
<tr>
<td>Configuration</td>
<td>KRC:\R1\Mada\</td>
</tr>
<tr>
<td>Log data</td>
<td>\Roboter\log\</td>
</tr>
<tr>
<td>KrcDiag</td>
<td>If it is necessary for an error to be analyzed by KUKA Roboter GmbH, this menu item can be used to compress the data for sending to KUKA.</td>
</tr>
</tbody>
</table>

In addition to the menu sequence File > Archive, there are other methods available for compressing these data.

(>>> 6.8.6 "Compressing data for error analysis at KUKA* Page 124)

If archiving is carried out using the menu item All and there is an existing archive present, this will be overwritten.

If archiving is carried out using a menu item other than All or KrcDiag and an archive is already available, the robot controller compares its robot name with that in the archive. If the names are different, a request for confirmation is generated.

If archiving is carried out repeatedly via KrcDiag, a maximum of 10 archives can be created. Further archives will overwrite the oldest existing archive.

The logbook can also be activated. (>>> 6.8.4 "Archiving the logbook* Page 124)
6.8.2 Archiving to a USB stick

**Description**
This procedure generates a ZIP file on the stick. By default, this file has the same name as the robot. A different name can be defined for the file, however, under Robot data.

(>>> 4.16.7 "Displaying/editing robot data" Page 68)

The archive is displayed in the ARCHIVE:\ directory in the Navigator. Archiving is also carried out automatically to D:\ as well as to the stick. The file INTERN.ZIP is generated here.

Special case KrcDiag:
This menu item generates the folder KRCDiag on the stick. This contains a ZIP file. The ZIP file is also automatically archived in C:\KUKA\KRCDiag.

**Precondition**
- A KUKA.USB data stick is connected.
  The stick can be connected to the KCP or robot controller.

**Procedure**
1. In the main menu, select File > Archive > USB (KCP) or USB (cabinet) and then the desired menu item.
2. Confirm the request for confirmation with Yes. The archive is created.
   Once the archiving is completed, this is indicated in the message window.
   Special case KrcDiag: If archiving is carried out using this menu item, a separate window indicates when archiving has been completed. The window is then automatically hidden again.
3. The stick can be removed when the LED on the stick is no longer lit.

6.8.3 Archiving on the network

**Description**
This procedure generates a ZIP file on the network path. By default, this file has the same name as the robot. A different name can be defined for the file, however, under Robot data.

The network path to which archiving is to be carried out must be configured under Robot data.

(>>> 4.16.7 "Displaying/editing robot data" Page 68)

The archive is displayed in the ARCHIVE:\ directory in the Navigator. Archiving is also carried out automatically to D:\ as well as to the network path. The file INTERN.ZIP is generated here.

Special case KrcDiag:
This menu item generates the folder KRCDiag on the network path. This contains a ZIP file. The ZIP file is also automatically archived in C:\KUKA\KRCDiag.

**Precondition**
- The network path to which the data are to be archived is configured.

**Procedure**
1. In the main menu, select File > Archive > Network and then the desired menu item.
2. Confirm the request for confirmation with Yes. The archive is created.
   Once the archiving is completed, this is indicated in the message window.
   Special case KrcDiag: If archiving is carried out using this menu item, a separate window indicates when archiving has been completed. The window is then automatically hidden again.

Only the KUKA.USB data stick may be used. Data may be lost or modified if any other USB stick is used.
6.8.4 Archiving the logbook

Description
The file "Logbuch.txt" is generated as an archive in the directory C:\KRC\ROBOTER\LOG.

Procedure
- In the main menu, select File > Archive > Logbook.
  The archive is created. Once the archiving is completed, this is indicated in the message window.

6.8.5 Restoring data

Description

![WARNING]
Only KSS 8.2 archives may be loaded into KSS 8.2. If other archives are loaded, the following may occur:
- Error messages
- Robot controller is not operable.
- Personal injury and damage to property.

The following menu items are available for restoring data:
- All
- Applications
- Configuration

If the archived files are not the same version as the files present in the system, an error message is generated during restoration.
Similarly, if the version of the archived technology packages does not match the installed version, an error message is generated.

Precondition
- If data are to be restored from the USB stick: A KUKA.USB data stick with the archive is connected.
  The stick can be connected to the KCP or robot controller.

Procedure
1. In the main menu, select File > Restore and then the desired subitems.
2. Confirm the request for confirmation with Yes. Archived files are restored to the robot controller. A message indicates completion of the restoration process.
3. When data have been restored from the USB stick: remove the stick when the LED on the stick is no longer lit.
4. Reboot the robot controller.

6.8.6 Compressing data for error analysis at KUKA

Description
If it is necessary for an error to be analyzed by KUKA Roboter GmbH, this procedure can be used to compress the data for sending to KUKA. The procedure generates a ZIP file in the directory C:\KUKA\KRCDiag. This contains the data required by KUKA Roboter GmbH to analyze an error (including information about system resources, screenshots, and much more.)

Procedure via “Diagnosis”
- In the main menu, select Diagnosis > KrcDiag.
  The data are compressed. Progress is displayed in a window. Once the operation has been completed, this is also indicated in the window. The window is then automatically hidden again.

Procedure via smartPAD
This procedure uses keys on the smartPAD instead of menu items. It can thus also be used if the smartHMI is not available, due to Windows problems for example.

Precondition:
- The smartPAD is connected to the robot controller.
- The robot controller is switched on.

![Information icon]

The keys must be pressed within 2 seconds. Whether or not the main menu and keypad are displayed in the smartHMI is irrelevant.

1. Press the “Main menu” key and hold it down.
2. Press the keypad key twice.
3. Release the “Main menu” key.

The data are compressed. Progress is displayed in a window. Once the operation has been completed, this is also indicated in the window. The window is then automatically hidden again.

### Procedure via “Archive”

Alternatively, the data can also be compressed via **File > Archive > [...].** In this way, the data can be stored on a USB stick or network path.

(>>> 6.8 "Archiving and restoring data" Page 121)
7 Basic principles of motion programming

7.1 Overview of motion types

The following motion types can be programmed:

- Point-to-point motions (PTP)  
  (>>> 7.2 "Motion type PTP" Page 127)
- Linear motions (LIN)  
  (>>> 7.3 "Motion type LIN" Page 127)
- Circular motions (CIRC)  
  (>>> 7.4 "Motion type CIRC" Page 128)
- Spline motions  
  (>>> 7.7 "Motion type "Spline"" Page 131)

LIN, CIRC and spline motions are also known as CP ("Continuous Path") motions.

The start point of a motion is always the end point of the previous motion.

7.2 Motion type PTP

The robot guides the TCP along the fastest path to the end point. The fastest path is generally not the shortest path and is thus not a straight line. As the motions of the robot axes are rotational, curved paths can be executed faster than straight paths.

The exact path of the motion cannot be predicted.

![Fig. 7-1: PTP motion](image_url)

7.3 Motion type LIN

The robot guides the TCP at a defined velocity along a straight path to the end point.
7.4 Motion type CIRC

The robot guides the TCP at a defined velocity along a circular path to the end point. The circular path is defined by a start point, auxiliary point and end point.

7.5 Approximate positioning

Approximate positioning means that the motion does not stop exactly at the programmed point. Approximate positioning is an option that can be selected during motion programming.
Approximate positioning is not possible if the motion instruction is followed by an instruction that triggers an advance run stop.

**PTP motion**

The TCP leaves the path that would lead directly to the end point and moves along a faster path. During programming of the motion, the maximum distance from the end point at which the TCP may deviate from its original path is defined.

The path of an approximated PTP motion cannot be predicted. It is also not possible to predict which side of the approximated point the path will run.

![Fig. 7-4: PTP motion, P2 is approximated](image)

**LIN motion**

The TCP leaves the path that would lead directly to the end point and moves along a shorter path. During programming of the motion, the maximum distance from the end point at which the TCP may deviate from its original path is defined.

The path in the approximate positioning range is **not** an arc.

![Fig. 7-5: LIN motion, P2 is approximated](image)

**CIRC motion**

The TCP leaves the path that would lead directly to the end point and moves along a shorter path. During programming of the motion, the maximum distance from the end point at which the TCP may deviate from its original path is defined.

The motion always stops exactly at the auxiliary point.

The path in the approximate positioning range is **not** an arc.
7.6 Orientation control LIN, CIRC

**Description**

The orientation of the TCP can be different at the start point and end point of a motion. There are several different types of transition from the start orientation to the end orientation. A type must be selected when a CP motion is programmed.

The orientation control for LIN and CIRC motions is defined as follows:

- In the option window **Motion parameter**
  
  (>>> 8.2.9 "Option window “Motion parameters” (LIN, CIRC)" Page 152)

<table>
<thead>
<tr>
<th>LIN motion</th>
<th>Orientation control</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIN motion</td>
<td>Constant orientation</td>
<td>The orientation of the TCP remains constant during the motion. The programmed orientation is disregarded for the end point and that of the start point is retained.</td>
</tr>
</tbody>
</table>
| LIN motion | Standard | The orientation of the TCP changes continuously during the motion.  
**Note:** If, with **Standard**, the robot passes through a wrist axis singularity, use **Wrist PTP** instead. |
| LIN motion | Wrist PTP | The orientation of the TCP changes continuously during the motion. This is done by linear transformation (axis-specific motion) of the wrist axis angles.  
**Note:** Use **Wrist PTP** if, with **Standard**, the robot passes through a wrist axis singularity. The orientation of the TCP changes continuously during the motion, but not uniformly. **Wrist PTP** is thus not suitable if a specific orientation must be maintained exactly, e.g. in the case of laser welding. |

If a wrist axis singularity occurs with **Standard** and the desired orientation cannot be maintained exactly enough with **Wrist PTP**, the following remedy is recommended:

Re-teach start and/or end point. Select orientations that prevent a wrist axis singularity from occurring and allow the path to be executed with **Standard**.
7 Basic principles of motion programming

CIRC motion

The same orientation control options are available for selection for CIRC motions as for LIN motions.

During CIRC motions, the robot controller only takes the programmed orientation of the end point into consideration. The programmed orientation of the auxiliary point is disregarded.

7.7 Motion type “Spline”

“Spline” is a Cartesian motion type that is suitable for particularly complex, curved paths. Such paths can generally also be generated using approximated LIN and CIRC motions, but Spline nonetheless has advantages.

Disadvantages of approximated LIN and CIRC motions:

- The path is defined by means of approximated points that are not located on the path. The approximate positioning ranges are difficult to predict. Generating the desired path is complicated and time-consuming.
- In many cases, the velocity may be reduced in a manner that is difficult to predict, e.g. in the approximate positioning ranges and near points that are situated close together.
- The path changes if approximate positioning is not possible, e.g. for time reasons.
- The path changes in accordance with the override setting, velocity or acceleration.
Advantages of Spline:

- The path is defined by means of points that are located on the path. The desired path can be generated easily.
- The programmed velocity is maintained. There are few cases in which the velocity is reduced.
  
  (>>> 7.7.1 "Velocity profile for spline motions" Page 133)
- The path always remains the same, irrespective of the override setting, velocity or acceleration.
- Circles and tight radii are executed with great precision.

A spline motion can consist of several individual motions: spline segments. These are taught separately. The segments are grouped together to form the overall motion in a so-called spline block. A spline block is planned and executed by the robot controller as a single motion block.

Furthermore, individual SLIN and SCIRC motions are possible (without spline block).

Fig. 7-9: Curved path with LIN

Fig. 7-10: Curved path with spline block
Further characteristics of all spline motions:

- If all points are situated on a plane, then the path is also situated in this plane.
- If all points are situated on a straight line, then the path is also a straight line.

### 7.7.1 Velocity profile for spline motions

The path always remains the same, irrespective of the override setting, velocity or acceleration. Only dynamic effects can cause deviations at different velocities.

The programmed acceleration is valid not only for the direction along the path, but also perpendicular to the path. The same applies to the jerk limitation. Effects include the following:

- In the case of circles, the centrifugal acceleration is taken into consideration. The velocity that can be achieved thus also depends on the programmed acceleration and the radius of the circle.
- In the case of curves, the maximum permissible velocity is derived from the radius of the curve, the acceleration and the jerk limitation.

#### Reduction of the velocity

In the case of spline motions, the velocity may, under certain circumstances, fall below the programmed velocity. This occurs particularly in the case of:

- Tight corners
- Major reorientation
- Large motions of the external axes

*If the points are close together, the velocity is not reduced.*

#### Reduction of the velocity to 0

This is the case for:

- Successive points with the same Cartesian coordinates.
- Successive SLIN and/or SCIRC segments. Cause: inconstant velocity direction.

In the case of SLIN-SCIRC transitions, the velocity is also reduced to 0 if the straight line is a tangent of the circle, as the circle, unlike the straight line, is curved.

---

**Fig. 7-11: Exact positioning at P2**

![Diagram showing exact positioning at P2](image-url)
Fig. 7-12: Exact positioning at P2

Exceptions:
- In the case of successive SLIN segments that result in a straight line and in which the orientations change uniformly, the velocity is not reduced.

Fig. 7-13: P2 is executed without exact positioning.
- In the case of a SCIRC-SCIRC transition, the velocity is not reduced if both circles have the same center point and the same radius and if the orientations change uniformly. (This is difficult to teach, so calculate and program points.)

Circles with the same center point and the same radius are sometimes programmed to obtain circles ≥ 360°. A simpler method is to program a circular angle.

7.7.2 Block selection with spline motions

Spline block
A spline block is planned and executed by the robot controller as a single motion block. Block selection to the spline segments is nonetheless possible. The BCO run is executed as a LIN motion. This is indicated by means of a message that must be acknowledged.

If the second segment in the spline block is an SPL segment, a modified path is executed in the following cases:
- Block selection to the first segment in the spline block
- Block selection to the spline block
- Block selection to a line before the spline block if this does not contain a motion instruction and if there is no motion instruction before the spline block

If the Start key is pressed after the BCO run, the modified path is indicated by means of a message that must be acknowledged.

Example:

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PTP P0</td>
</tr>
<tr>
<td>2</td>
<td>SPLINE</td>
</tr>
<tr>
<td>3</td>
<td>SPL P1</td>
</tr>
<tr>
<td>4</td>
<td>SPL P2</td>
</tr>
<tr>
<td>5</td>
<td>SPL P3</td>
</tr>
<tr>
<td>6</td>
<td>SPL P4</td>
</tr>
<tr>
<td>7</td>
<td>SCIRC P5, P6</td>
</tr>
<tr>
<td>8</td>
<td>SPL P7</td>
</tr>
<tr>
<td>9</td>
<td>SLIN P8</td>
</tr>
<tr>
<td>10</td>
<td>ENDSPLINE</td>
</tr>
</tbody>
</table>

Line 2: Start of the spline block
7 Basic principles of motion programming

In the case of block selection to a SCIRC instruction for which a circular angle has been programmed, the motion is executed to the end point including the circular angle, provided that the robot controller knows the start point. If this is not possible, the motion is executed to the programmed end point. In this case, a message is generated, indicating that the circular angle is not being taken into consideration.

<table>
<thead>
<tr>
<th>Position/type of SCIRC instruction</th>
<th>End point for block selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCIRC segment is 1st segment in the spline block</td>
<td>Circular angle is not taken into consideration</td>
</tr>
<tr>
<td>Other SCIRC segments in the spline block</td>
<td>Circular angle is taken into consideration</td>
</tr>
<tr>
<td>Individual SCIRC motions</td>
<td>Circular angle is not taken into consideration</td>
</tr>
</tbody>
</table>

Fig. 7-14: Example: modified path in the case of block selection to P1

SCIRC

In the case of block selection to a SCIRC instruction for which a circular angle has been programmed, the motion is executed to the end point including the circular angle, provided that the robot controller knows the start point. If this is not possible, the motion is executed to the programmed end point. In this case, a message is generated, indicating that the circular angle is not being taken into consideration.

7.7.3 Modifications to spline blocks

**Description**

- Modification of the position of the point:
  
  If a point within a spline block is offset, the path is modified, at most, in the 2 segments before this point and the 2 segments after it.
  
  Small point offsets generally result in small modifications to the path. If, however, very long segments are followed by very short segments or vice versa, small modifications can have a very great effect, as the tangents and curves can change very greatly in such cases.

- Modification of the segment type:
  
  If an SPL segment is changed into an SLIN segment or vice versa, the path changes in the previous segment and the next segment.
Example 1

P3 is offset. This causes the path to change in segments P1 - P2, P2 - P3 and P3 - P4. Segment P4 - P5 is not changed in this case, as it belongs to an SCIRC and a circular path is thus defined.

Fig. 7-15: Original path

P3 is offset. This causes the path to change in segments P1 - P2, P2 - P3 and P3 - P4. Segment P4 - P5 is not changed in this case, as it belongs to an SCIRC and a circular path is thus defined.

Fig. 7-16: Point has been offset

In the original path, the segment type of P2 - P3 is changed from SPL to SLIN. The path changes in segments P1 - P2, P2 - P3 and P3 - P4.
Example 2

P3 is offset. This causes the path to change in all the segments illustrated. Since P2 - P3 and P3 - P4 are very short segments and P1 - P2 and P4 - P5 are long segments, the slight offset causes the path to change greatly.

Remedy:
- Distribute the points more evenly
- Program straight lines (except very short ones) as SLIN segments

7.7.4 Approximate positioning of spline motions

Approximate positioning between spline motions (individual SLIN and SCIRC motions and spline blocks) is possible.
Approximate positioning between spline motions and LIN, CIRC or PTP is not possible.

**Approximation not possible due to time or advance run stop:**

If approximation is not possible for reasons of time or due to an advance run stop, the robot waits at the start of the approximate positioning arc.

- In the case of time reasons: the robot moves again as soon as it has been possible to plan the next block.
- In the case of an advance run stop: the end of the current block is reached at the start of the approximate positioning arc. This means that the advance run stop is canceled and the robot controller can plan the next block. Robot motion is resumed.

In both cases, the robot now moves along the approximate positioning arc. Approximate positioning is thus technically possible; it is merely delayed.

This response differs from that for LIN, CIRC or PTP motions. If approximate positioning is not possible for the reasons specified, the motion is executed to the end point with exact positioning.

**No approximate positioning in MSTEP and ISTEP:**

In the program run modes MSTEP and ISTEP, the robot stops exactly at the end point, even in the case of approximated motions.

In the case of approximate positioning from one spline block to another spline block, the result of this exact positioning is that the path is different in the last segment of the first block and in the first segment of the second block from the path in program run mode GO.

In all other segments of both spline blocks, the path is identical in MSTEP, ISTEP and GO.

### 7.7.5 Replacing an approximated motion with a spline block

**Description**

In order to replace conventional approximated motions with spline blocks, the program must be modified as follows:

- Replace LIN - LIN with SLIN - SPL - SLIN.
- Replace LIN - CIRC with SLIN - SPL - SCIRC.

Recommendation: Allow the SPL to project a certain way into the original circle. The SCIRC thus starts later than the original CIRC.

In approximated motions, the corner point is programmed. In the spline block, the points at the start and end of the approximation are programmed instead.

The approximate positioning arc of the approximated motions varies according to the override. For this reason, when reproducing an approximated motion, it must be ensured that it is executed with the desired override.

The following approximated motion is to be reproduced:

```
LIN P1 C_DIS
LIN P2
```

Spline motion:

```
SPLINE
SLIN P1A
SPL P1B
SLIN P2
ENDSPLINE
```

P1A = start of approximation, P1B = end of approximation
Ways of determining P1A and P1B:

- Execute the approximated path and save the positions at the desired point by means of Trigger.
- Calculate the points in the program with KRL.
- The start of the approximation can be determined from the approximate positioning criterion. Example: If C_DIS is specified as the approximate positioning criterion, the distance from the start of the approximation to the corner point corresponds to the value of $APO.CDIS.

The end of the approximation is dependent on the programmed velocity. The SPL path does not correspond exactly to the approximate positioning arc, even if P1A and P1B are exactly at the start/end of the approximation. In order to recreate the exact approximate positioning arc, additional points must be inserted into the spline. Generally, one point is sufficient.

**Example**

The following approximated motion is to be reproduced:

```
$APO.CDIS=20
$VEL.CP=0.5
LIN (Z 10) C_DIS
LIN (Y 60)
```

Spline motion:

```
SPLINE WITH $VEL.CP=0.5
SLIN (Z 30)
SPL (Y 30, Z 10)
SLIN (Y 60)
ENDSPLINE
```

The start of the approximate positioning arc has been calculated from the approximate positioning criterion.
The SPL path does not yet correspond exactly to the approximate positioning arc. For this reason, an additional SPL segment is inserted into the spline.

```
SPLINE WITH $VEL_CP=0.5
SLIN {Z 30}
SPL {Y 15, Z 15}
SPL {Y 30, Z 10}
SLIN {Y 60}
ENDSPLINE
```

Fig. 7-21: Example: Approximated motion - spline motion 1

With the additional point, the path now corresponds to the approximate positioning arc.

Fig. 7-22: Example: Approximated motion - spline motion 2
7.7.5.1 SLIN-SPL-SLIN transition

In the case of a SLIN-SPL-SLIN segment sequence, it is usually desirable for the SPL segment to be located within the smaller angle between the two straight lines. Depending on the start and end point of the SPL segment, the path may also move outside this area.

![Diagram of SLIN-SPL-SLIN transition]

Fig. 7-23: SLIN-SPL-SLIN

The path remains inside if the following conditions are met:

- The extensions of the two SLIN segments intersect.
- \( \frac{2}{3} \leq \frac{a}{b} \leq \frac{3}{2} \)
  - \( a \) = distance from start point of the SPL segment to intersection of the SLIN segments
  - \( b \) = distance from intersection of the SLIN segments to end point of the SPL segment

7.8 Orientation control SPLINE

Description

The orientation of the TCP can be different at the start point and end point of a motion. When a CP motion is programmed, it is necessary to select how to deal with the different orientations.

The orientation control for SLIN and SCIRC motions is defined as follows:

- In the option window Motion parameter
  (\( \gg \gg \) 8.3.2.2 “Option window “Motion parameters” (SLIN)” Page 155)
  (\( \gg \gg \) 8.3.3.2 “Option window “Motion parameters” (SCIRC)” Page 157)
  (\( \gg \gg \) 8.3.4.3 “Option window “Motion parameters” (spline block)” Page 160)
  (\( \gg \gg \) 8.3.4.8 “Option window “Motion parameters” (spline segment)” Page 163)
Ignore Orientation

The option Ignore Orientation is used if no specific orientation is required at a point. If this option is selected, the taught or programmed orientation of the point is ignored. Instead, the robot controller calculates the optimal orientation for this point on the basis of the orientations of the surrounding points.

Characteristics of Ignore Orientation:

- In the program run modes MSTEP and ISTEP, the robot stops with the orientations calculated by the robot controller.
- In the case of a block selection to a point with Ignore Orientation, the robot adopts the orientation calculated by the robot controller.
Ignore **Orientation** is not allowed for the following segments:

- The first segment in a spline block
- The last segment in a spline block
- SCIRC segments with Circle orientation control = path-related
- Segments followed by a SCIRC segment with Circle orientation control = path-related
- Segments followed by a segment with Orientation control = Constant orientation
- In the case of successive segments with identical Cartesian end points, Ignore Orientation is not allowed for the first and last segments.

**SCIRC**

The same orientation control options are available for selection for SCIRC motions as for SLIN motions. It is also possible to define for SCIRC motions whether the orientation control is to be space-related or path-related.

<table>
<thead>
<tr>
<th>Orientation control</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>base-related</td>
<td>Base-related orientation control during the circular motion</td>
</tr>
<tr>
<td>path-oriented</td>
<td>Path-related orientation control during the circular motion</td>
</tr>
</tbody>
</table>

(>>> 7.8.1 "Combinations of “Orientation control” and “Circle orientation control”" Page 144)

The option **Path-related** is not allowed for the following motions:

- SCIRC segments with the orientation control type Ignore Orientation
- SCIRC motions preceded by a spline segment with the orientation control type Ignore Orientation

**Orientation of the auxiliary point:**

During SCIRC motions with the orientation control type **Standard**, the robot controller takes the programmed orientation of the auxiliary point into consideration, but only to a certain extent.

The transition from the start orientation to the end orientation passes through the programmed orientation of the auxiliary point, i.e. the orientation of the auxiliary point is accepted at some point during the transition, but not necessarily at the auxiliary point.
7.8.1 Combinations of “Orientation control” and “Circle orientation control”

Fig. 7-26: Constant orientation control + path-related

Fig. 7-27: Standard + path-related

Fig. 7-28: Constant orientation control + base-related
7.9 Singularities

KUKA robots with 6 degrees of freedom have 3 different singularity positions.

- Overhead singularity
- Extended position singularity
- Wrist axis singularity

A singularity position is characterized by the fact that unambiguous reverse transformation (conversion of Cartesian coordinates to axis-specific values) is not possible, even though Status and Turn are specified. In this case, or if very slight Cartesian changes cause very large changes to the axis angles, one speaks of singularity positions.

**Overhead**

In the overhead singularity, the wrist root point (intersection of axes A4, A5 and A6) is located vertically above axis 1.

The position of axis A1 cannot be determined unambiguously by means of reverse transformation and can thus take any value.

If the end point of a PTP motion is situated in this overhead singularity position, the robot controller may react as follows by means of the system variable $SINGUL_POS[1]$

- 0: The angle for axis A1 is defined as 0 degrees (default setting).
- 1: The angle for axis A1 remains the same from the start point to the end point.

**Extended position**

In the extended position singularity, the wrist root point (intersection of axes A4, A5 and A6) is located in the extension of axes A2 and A3 of the robot.

The robot is at the limit of its work envelope.

Although reverse transformation does provide unambiguous axis angles, low Cartesian velocities result in high axis velocities for axes A2 and A3.

If the end point of a PTP motion is situated in this extended position singularity, the robot controller may react as follows by means of the system variable $SINGUL_POS[2]$

- 0: The angle for axis A2 is defined as 0 degrees (default setting).
- 1: The angle for axis A2 remains the same from the start point to the end point.
Wrist axes

In the wrist axis singularity position, the axes A4 and A6 are parallel to one another and axis A5 is within the range ±0.01812°.

The position of the two axes cannot be determined unambiguously by reverse transformation. There is an infinite number of possible axis positions for axes A4 and A6 with identical axis angle sums.

If the end point of a PTP motion is situated in this wrist axis singularity, the robot controller may react as follows by means of the system variable $SINGUL_POS[3]:

- 0: The angle for axis A4 is defined as 0 degrees (default setting).
- 1: The angle for axis A4 remains the same from the start point to the end point.

In the case of SCARA robots, only the extended position singularity can arise. In this case, the robot starts to move extremely fast.
8 Programming for user group “User” (inline forms)

Inline forms are available in the KSS for frequently used instructions. They simplify programming.

Instructions can also be programmed without inline forms. The KRL programming language (KUKA Robot Language) is used for this. Information is contained in the Operating and Programming Instructions for System Integrators.

8.1 Names in inline forms

Names for data sets can be entered in inline forms. These include, for example, point names, names for motion data sets, etc.

The following restrictions apply to names:

- Maximum length 23 characters
- No special characters are permissible, with the exception of $.
- The first character must not be a number.

The restrictions do not apply to output names.

Other restrictions may apply in the case of inline forms in technology packages.

8.2 Programming PTP, LIN and CIRC motions

8.2.1 Programming a PTP motion

When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Precondition

- A program is selected.
- Operating mode T1

In the case of programs with the following axis motions or positions, the film of lubricant on the gear units of the axes may break down:

- Motions <3°
- Oscillating motions
- Areas of gear units permanently facing upwards

It must be ensured that the gear units have a sufficient supply of oil. For this, in the case of oscillating motions or short motions (<3°), programming must be carried out in such a way that the affected axes regularly move more than 40° (e.g. once per cycle).

In the case of areas of gear units permanently facing upwards, sufficient oil supply must be achieved by programming re-orientations of the in-line wrist. In this way, the oil can reach all areas of the gear units by means of gravity.

Required frequency of re-orientations:

- With low loads (gear unit temperature <+35 °C): daily
- With medium loads (gear unit temperature +35 °C to 55 °C): hourly
- With heavy loads (gear unit temperature >+55 °C): every 10 minutes

Failure to observe this precaution may result in damage to the gear units.

Failure to observe this precaution may result in damage to the gear units.

When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Precondition

- A program is selected.
- Operating mode T1
Procedure

1. Move the TCP to the position that is to be taught as the end point.
2. Position the cursor in the line after which the motion instruction is to be inserted.
3. Select the menu sequence **Commands > Motion > PTP**.
4. Set the parameters in the inline form.
   
   [>>> 8.2.2 "Inline form “PTP”" Page 148]
5. Save instruction with **Cmd Ok**.

### 8.2.2 Inline form “PTP”

**Fig. 8-1: Inline form for PTP motions**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Motion type PTP</td>
</tr>
</tbody>
</table>
| 2    | Name of the end point  
The system automatically generates a name. The name can be overwritten.  
(>>> 8.1 "Names in inline forms" Page 147)  
Touch the arrow to edit the point data. The corresponding option window is opened.  
(>>> 8.2.7 "Option window “Frames”” Page 151) |
| 3    | CONT: end point is approximated.  
[Empty box]: the motion stops exactly at the end point. |
| 4    | Velocity  
1 … 100% |
| 5    | Name for the motion data set  
The system automatically generates a name. The name can be overwritten.  
Touch the arrow to edit the point data. The corresponding option window is opened.  
(>>> 8.2.8 "Option window “Motion parameter” (PTP)” Page 151) |

### 8.2.3 Programming a LIN motion

**NOTICE** When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Precondition

- A program is selected.
- Operating mode T1

Procedure

1. Move the TCP to the position that is to be taught as the end point.
2. Position the cursor in the line after which the motion instruction is to be inserted.
3. Select the menu sequence **Commands > Motion > LIN**.
4. Set the parameters in the inline form.
   (>>> 8.2.4 "Inline form “LIN”" Page 149)

5. Save instruction with **Cmd Ok**.

### 8.2.4 Inline form “LIN”

![Fig. 8-2: Inline form for LIN motions]

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Motion type <strong>LIN</strong></td>
</tr>
</tbody>
</table>
| 2    | Name of the end point  
The system automatically generates a name. The name can be overwritten.  
(>>> 8.1 "Names in inline forms" Page 147)  
Touch the arrow to edit the point data. The corresponding option window is opened.  
(>>> 8.2.7 "Option window “Frames”" Page 151) |
| 3    |  
- **CONT**: end point is approximated.  
- **[Empty box]**: the motion stops exactly at the end point. |
| 4    | Velocity  
- 0.001 … 2 m/s |
| 5    | Name for the motion data set  
The system automatically generates a name. The name can be overwritten.  
Touch the arrow to edit the point data. The corresponding option window is opened.  
(>>> 8.2.9 "Option window “Motion parameters” (LIN, CIRC)” Page 152) |

### 8.2.5 Programming a CIRC motion

---

**NOTICE** When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

**Precondition**
- A program is selected.
- Operating mode T1

**Procedure**
1. Move the TCP to the position that is to be taught as the auxiliary point.
2. Position the cursor in the line after which the motion instruction is to be inserted.
3. Select the menu sequence **Commands > Motion > CIRC**.
4. Set the parameters in the inline form.  
   (>>> 8.2.6 "Inline form “CIRC”" Page 150)
5. Press **Teach Aux**.
6. Move the TCP to the position that is to be taught as the end point.
7. Save instruction with **Cmd Ok**.

### 8.2.6 Inline form “CIRC”

![Inline form for CIRC motions](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Motion type <strong>CIRC</strong></td>
</tr>
</tbody>
</table>
| 2    | Name of the auxiliary point  
The system automatically generates a name. The name can be overwritten.  
(>>> 8.1 "Names in inline forms" Page 147) |
| 3    | Name of the end point  
The system automatically generates a name. The name can be overwritten.  
Touch the arrow to edit the point data. The corresponding option window is opened.  
(>>> 8.2.7 "Option window “Frames”" Page 151) |
| 4    | - **CONT**: end point is approximated.  
- **[Empty box]**: the motion stops exactly at the end point. |
| 5    | Velocity  
- 0.001 ... 2 m/s |
| 6    | Name for the motion data set  
The system automatically generates a name. The name can be overwritten.  
Touch the arrow to edit the point data. The corresponding option window is opened.  
(>>> 8.2.9 "Option window “Motion parameters” (LIN, CIRC)” Page 152) |
8.2.7 Option window “Frames”

This option window is called from the following inline form:

- PTP

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | Tool selection.  
  If True in the box External TCP: workpiece selection.  
  Range of values: [1] … [16] |
| 2    | Base selection.  
  If True in the box External TCP: fixed tool selection.  
  Range of values: [1] … [32] |
| 3    | Interpolation mode  
  - False: The tool is mounted on the mounting flange.  
  - True: The tool is a fixed tool. |
| 4    |  
  - True: For this motion, the robot controller calculates the axis torques. These are required for collision detection.  
  - False: For this motion, the robot controller does not calculate the axis torques. Collision detection is thus not possible for this motion. |

8.2.8 Option window “Motion parameter” (PTP)

This option window is called from the following inline form:

- PTP

Fig. 8-4: Option window “Frames”

Fig. 8-5: Option window “Motion parameter” (PTP)
### 8.2.9 Option window “Motion parameters” (LIN, CIRC)

This option window is called from the following inline forms:

- LIN, CIRC

---

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | Acceleration  
Refers to the maximum value specified in the machine data. The maximum value depends on the robot type and the selected operating mode.  
- 1 … 100% |
| 2    | Furthest distance before the end point at which approximate positioning can begin  
The maximum permissible value is half the distance between the start point and the end point. If a higher value is entered, this is ignored and the maximum value is used.  
This box is only displayed if CONT was selected in the inline form.  
- 1 … 100% |
| 3    | Orientation control selection.  
- Standard  
- Wrist PTP  
- Constant orientation control  

(>>> 7.6 "Orientation control LIN, CIRC" Page 130)
8.3 Spline motions

8.3.1 Programming tips for spline motions

- A spline block should cover only 1 process (e.g. 1 adhesive seam). More than one process in a spline block leads to a loss of structural clarity within the program and makes changes more difficult.

- Use SLIN and SCIRC segments in cases where the workpiece necessitates straight lines and arcs. (Exception: use SPL segments for very short straight lines.) Otherwise, use SPL segments, particularly if the points are close together.

- Procedure for defining the path:
  a. First teach or calculate a few characteristic points. Example: points at which the curve changes direction.
  b. Test the path. At points where the accuracy is still insufficient, add more SPL points.

- Avoid successive SLIN and/or SCIRC segments, as this often reduces the velocity to 0.
  Program SPL segments between SLIN and SCIRC segments. The length of the SPL segments must be at least > 0.5 mm. Depending on the actual path, significantly larger SPL segments may be required.

- Avoid successive points with identical Cartesian coordinates, as this reduces the velocity to 0.

- The parameters (tool, base, velocity, etc.) assigned to the spline block have the same effect as assignments before the spline block. The assignment to the spline block has the advantage, however, that the correct parameters are read in the case of a block selection.

- Use the option **Ignore Orientation** if no specific orientation is required at a point. The robot controller calculates the optimal orientation for this point on the basis of the orientations of the surrounding points. This way, even large changes in orientation between two points are optimally distributed over the points in between.

- Jerk limitation can be programmed. The jerk is the change in acceleration.
  Procedure:
  a. Use the default values initially.
  b. If vibrations occur at tight corners: reduce values.
     If the velocity drops or the desired velocity cannot be reached: increase values or increase acceleration.

- If the robot executes points on a work surface, a collision with the work surface is possible when the first point is addressed.

![Fig. 8-7: Collision with work surface](image)

In order to avoid a collision, observe the recommendations for the SLIN-SPL-SLIN transition.

(>>> 7.7.5.1 "SLIN-SPL-SLIN transition" Page 141)
8.3.2 Programming a SLIN motion (individual motion)

When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Precondition
- A program is selected.
- Operating mode T1

Procedure
1. Move the TCP to the end point.
2. Position the cursor in the line after which the motion is to be inserted. (But not within a spline block. This opens another inline form.)
   (>>> 8.3.4.6 "Inline form for spline segment" Page 162)
3. Select Commands > Motion > SLIN.
4. Set the parameters in the inline form.
   (>>> 8.3.2.1 "Inline form “SLIN”" Page 154)
5. Press Cmd OK.

8.3.2.1 Inline form “SLIN”

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Motion type SLIN</td>
</tr>
</tbody>
</table>
| 2    | Point name for end point. The system automatically generates a name. The name can be overwritten.  
   (>>> 8.1 "Names in inline forms" Page 147) 
   Touch the arrow to edit the point data. The corresponding option window is opened.  
   (>>> 8.2.7 "Option window “Frames”" Page 151) |
| 3    | CONT: end point is approximated.  
   [Empty box]: the motion stops exactly at the end point. |
### 8.3.2.2 Option window “Motion parameters” (SLIN)

#### Fig. 8-10: Option window “Motion parameters” (SLIN)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Path acceleration. The value refers to the maximum value specified in the machine data.</td>
</tr>
<tr>
<td></td>
<td>1 ... 100%</td>
</tr>
<tr>
<td>2</td>
<td>Jerk limitation. The jerk is the change in acceleration. The value refers to the maximum value specified in the machine data.</td>
</tr>
<tr>
<td></td>
<td>1 ... 100%</td>
</tr>
<tr>
<td>3</td>
<td>This box is only displayed if CONT was selected in the inline form. Furthest distance before the end point at which approximate positioning can begin. The maximum permissible value is half the distance between the start point and the end point. If a higher value is entered, this is ignored and the maximum value is used.</td>
</tr>
<tr>
<td>4</td>
<td>Axis velocity. The value refers to the maximum value specified in the machine data.</td>
</tr>
<tr>
<td></td>
<td>1 ... 100%</td>
</tr>
<tr>
<td>5</td>
<td>Axis acceleration. The value refers to the maximum value specified in the machine data.</td>
</tr>
<tr>
<td></td>
<td>1 ... 100%</td>
</tr>
<tr>
<td>6</td>
<td>Orientation control selection.</td>
</tr>
</tbody>
</table>
8.3.3 Programming a SCIRC motion (individual motion)

**NOTICE** When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

**Precondition**
- A program is selected.
- Operating mode T1

**Procedure**
1. Move the TCP to the auxiliary point.
2. Position the cursor in the line after which the motion is to be inserted. (But not within a spline block. This opens another inline form.)
   (>>> 8.3.4.6 "Inline form for spline segment" Page 162)
3. Select the menu sequence **Commands > Motion > SCIRC**.
4. Set the parameters in the inline form.
   (>>> 8.3.3.1 "Inline form “SCIRC”" Page 156)
5. Press **Teach Aux**.
6. Move the TCP to the end point.
7. Press **Cmd OK**.

8.3.3.1 Inline form “SCIRC”

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Motion type <strong>SCIRC</strong></td>
</tr>
</tbody>
</table>
| 2    | Point names for auxiliary and end point. The system automatically assigns names. The names can be overwritten.  
   (>>> 8.1 "Names in inline forms" Page 147)  
   Touch the arrow to edit the point data. The corresponding option window is opened.  
   (>>> 8.2.7 "Option window “Frames”" Page 151) |
| 3    | **CONT**: end point is approximated.  
   **[Empty box]**: the motion stops exactly at the end point. |
| 4    | Velocity  
   **0.001 ... 2 m/s** |
8 Programming for user group “User” (inline forms)

8.3.3.2 Option window “Motion parameters” (SCIRC)

5 Name for the motion data set. The system automatically generates a name. The name can be overwritten.

   Touch the arrow to edit the point data. The corresponding option window is opened.

   ![Fig. 8-12: Option window “Motion parameters” (SCIRC)](image)

   Table 8-12: Option window “Motion parameters” (SCIRC)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Name for the motion data set. The system automatically generates a name. The name can be overwritten. Touch the arrow to edit the point data. The corresponding option window is opened.</td>
</tr>
</tbody>
</table>

6 Specifies the overall angle of the circular motion. This makes it possible to extend the motion beyond the programmed end point or to shorten it. The actual end point thus no longer corresponds to the programmed end point.

- Positive circular angle: the circular path is executed in the direction Start point › Auxiliary point › End point.
- Negative circular angle: the circular path is executed in the direction Start point › End point › Auxiliary point.
- -9,999° … +9,999°

If a circular angle less than -400° or greater than +400° is entered, a request for confirmation is generated when the inline form is saved asking whether entry is to be confirmed or rejected.

8.3.3.2 Option window “Motion parameters” (SCIRC)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Path acceleration. The value refers to the maximum value specified in the machine data.</td>
</tr>
<tr>
<td></td>
<td>1 … 100%</td>
</tr>
<tr>
<td>2</td>
<td>Jerk limitation. The jerk is the change in acceleration. The value refers to the maximum value specified in the machine data.</td>
</tr>
<tr>
<td></td>
<td>1 … 100%</td>
</tr>
</tbody>
</table>
A spline block can be used to group together several SPL, SLIN and/or SCIRC segments to an overall motion. A spline block that contains no segments is not a motion statement.

A spline block may contain the following:

- Spline segments (only limited by the memory capacity.)
- PATH trigger
- Comments and blank lines
- Inline commands from technology packages that support the spline functionality

A spline block must not include any other instructions, e.g. variable assignments or logic statements. A spline block does not trigger an advance run stop.

### Precondition

- A program is selected.
- Operating mode T1

### Procedure

1. Position the cursor in the line after which the spline block is to be inserted.
2. Select the menu sequence **Commands > Motion > SPLINE block**.
3. Set the parameters in the inline form.
   - (>>> 8.3.4.1 "Inline form for spline block" Page 159)
4. Press **Cmd OK**.
5. Press **Open/close fold**. Spline segments and other lines can now be inserted into the spline block.
   - (>>> 8.3.4.4 "Programming an SPL or SLIN segment" Page 161)
   - (>>> 8.3.4.5 "Programming an SCIRC segment" Page 161)
   - (>>> 8.3.4.9 "Programming triggers in the spline block" Page 164)
8.3.4.1 Inline form for spline block

![Fig. 8-13: Inline form for spline block](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | Name of the spline block. The system automatically generates a name. The name can be overwritten.  
       (>>> 8.1 "Names in inline forms" Page 147)  
       Position the cursor in this box to edit the motion data. The corresponding option window is opened.  
       (>>> 8.3.4.2 "Option window “Frames” (spline block)” Page 159) |
| 2    | CONT: end point is approximated.  
       [Empty box]: the motion stops exactly at the end point. |
| 3    | The velocity is valid by default for the entire spline block. It can also be defined separately for individual segments.  
       0.001 … 2 m/s |
| 4    | Name for the motion data set. The system automatically generates a name. The name can be overwritten.  
       Position the cursor in this box to edit the motion data. The corresponding option window is opened.  
       (>>> 8.3.4.3 "Option window “Motion parameters” (spline block)” Page 160)  
       The motion data are valid by default for the entire spline block. They can also be defined separately for individual segments. |

8.3.4.2 Option window “Frames” (spline block)

![Fig. 8-14: Option window “Frames” (spline block)](image)
### Item | Description
--- | ---
1 | Tool selection.  
   If True in the box **External TCP**: workpiece selection.  
   ■ [1] … [16]
2 | Base selection.  
   If True in the box **External TCP**: fixed tool selection.  
3 | Interpolation mode  
   ■ False: The tool is mounted on the mounting flange.  
   ■ True: The tool is a fixed tool.

#### 8.3.4.3 Option window “Motion parameters” (spline block)

![Option window “Motion parameters” (spline block)](image)

- **Path acceleration.** The value refers to the maximum value specified in the machine data.  
  ■ 1 … 100%
- **Jerk limitation.** The jerk is the change in acceleration.  
  The value refers to the maximum value specified in the machine data.  
  ■ 1 … 100%
- This box is only displayed if **CONT** was selected in the inline form.  
  Furthest distance before the end point at which approximate positioning can begin.  
  The maximum distance is that of the last segment in the spline. If there is only one segment present, the maximum distance is half the segment length. If a higher value is entered, this is ignored and the maximum value is used.
- **Axis velocity.** The value refers to the maximum value specified in the machine data.  
  ■ 1 … 100%
### Programming an SPL or SLIN segment

**Precondition**
- A program is selected.
- Operating mode T1
- The spline block fold is open.

**Procedure**
1. Move the TCP to the end point.
2. Position the cursor in the line after which the segment is to be inserted in the spline block.
3. Select the menu sequence **Commands > Motion > SPL** or **SLIN**.
4. Set the parameters in the inline form.
   
   (>>> 8.3.4.6 “Inline form for spline segment” Page 162)
5. Press **Cmd OK**.

### Programming an SCIRC segment

**Precondition**
- A program is selected.
- Operating mode T1
- The spline block fold is open.

**Procedure**
1. Move the TCP to the auxiliary point.
2. Position the cursor in the line after which the segment is to be inserted in the spline block.
3. Select the menu sequence **Commands > Motion > SCIRC**.
4. Set the parameters in the inline form.
   
   (>>> 8.3.4.6 “Inline form for spline segment” Page 162)
5. Press **Teach Aux**.
6. Move the TCP to the end point.
7. Press **Cmd OK**.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Axis acceleration. The value refers to the maximum value specified in the machine data.</td>
</tr>
<tr>
<td></td>
<td>1 ... 100%</td>
</tr>
<tr>
<td>6</td>
<td>Orientation control selection.</td>
</tr>
<tr>
<td>7</td>
<td>Orientation control reference system selection. This parameter only affects SCIRC segments (if present) in the spline block.</td>
</tr>
</tbody>
</table>
8.3.4.6 Inline form for spline segment

The boxes in the inline form can be displayed or hidden one by one with Toggle parameters.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Motion type</td>
</tr>
<tr>
<td></td>
<td>▪ SPL</td>
</tr>
<tr>
<td></td>
<td>▪ SLIN</td>
</tr>
<tr>
<td></td>
<td>▪ SCIRC</td>
</tr>
<tr>
<td>2</td>
<td>Point name for end point. Only for SCIRC: point names for auxiliary point and end point. The system automatically generates a name. The name can be overwritten.</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 8.1 &quot;Names in inline forms&quot; Page 147)</td>
</tr>
<tr>
<td></td>
<td>Touch the arrow to edit the point data. The corresponding option window is opened.</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 8.3.4.7 &quot;Option window “Frames” (spline segment)” Page 163)</td>
</tr>
<tr>
<td>3</td>
<td>Velocity</td>
</tr>
<tr>
<td></td>
<td>This only refers to the segment to which it belongs. It has no effect on subsequent segments.</td>
</tr>
<tr>
<td></td>
<td>▪ 0.001 … 2 m/s</td>
</tr>
<tr>
<td>4</td>
<td>Name for the motion data set. The system automatically generates a name. The name can be overwritten.</td>
</tr>
<tr>
<td></td>
<td>Touch the arrow to edit the point data. The corresponding option window is opened.</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 8.3.4.8 &quot;Option window “Motion parameters” (spline segment)” Page 163)</td>
</tr>
<tr>
<td></td>
<td>The motion data only refer to the segment to which they belong. They have no effect on subsequent segments.</td>
</tr>
<tr>
<td>5</td>
<td>Only available if the motion type SCIRC has been selected.</td>
</tr>
<tr>
<td></td>
<td>Specifies the overall angle of the circular motion. This makes it possible to extend the motion beyond the programmed end point or to shorten it. The actual end point thus no longer corresponds to the programmed end point.</td>
</tr>
<tr>
<td></td>
<td>▪ Positive circular angle: the circular path is executed in the direction Start point › Auxiliary point › End point.</td>
</tr>
<tr>
<td></td>
<td>▪ Negative circular angle: the circular path is executed in the direction Start point › End point › Auxiliary point.</td>
</tr>
<tr>
<td></td>
<td>▪ - 9,999° … + 9,999°</td>
</tr>
<tr>
<td></td>
<td>If a circular angle less than -400° or greater than +400° is entered, a request for confirmation is generated when the inline form is saved asking whether entry is to be confirmed or rejected.</td>
</tr>
</tbody>
</table>
8.3.4.7 Option window “Frames” (spline segment)

Fig. 8-17: Option window “Frames” (spline segment)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | **True:** For this motion, the robot controller calculates the axis torques. These are required for collision detection.  

**False:** For this motion, the robot controller does not calculate the axis torques. Collision detection is thus not possible for this motion. |

8.3.4.8 Option window “Motion parameters” (spline segment)

Fig. 8-18: Option window “Motion parameters” (spline segment)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | Path acceleration. The value refers to the maximum value specified in the machine data.  

**1 … 100%** |
| 2    | Jerk limitation. The jerk is the change in acceleration.  

The value refers to the maximum value specified in the machine data.  

**1 … 100%** |
| 3    | Axis velocity. The value refers to the maximum value specified in the machine data.  

**1 … 100%** |
8.3.4.9 Programming triggers in the spline block

**Precondition**
- A program is selected.
- Operating mode T1
- The spline block fold is open.

**Procedure**
1. Position the cursor in the line after which the trigger is to be inserted in the spline block.
2. Select the menu sequence **Commands > Logic > Spline trigger**.
3. The inline form **Set output** is displayed by default. A different inline form can be displayed by pressing the **Toggle type** button.
4. Set the parameters in the inline form.
5. Press **Cmd OK**.

**Description**
The specific inline form that is displayed depends on which type has been selected using **Toggle type**.

<table>
<thead>
<tr>
<th>Inline form type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Set output</strong></td>
<td>The trigger sets an output.</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 8.3.4.10 &quot;Inline form for spline trigger type &quot;Set output&quot;&quot; Page 165)</td>
</tr>
<tr>
<td><strong>Set pulse output</strong></td>
<td>The trigger sets a pulse of a defined length.</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 8.3.4.11 &quot;Inline form for spline trigger type &quot;Set pulse output&quot;&quot; Page 166)</td>
</tr>
<tr>
<td><strong>Trigger assignment</strong></td>
<td>The trigger assigns a value to a variable. Only available in the user group “Expert”.</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 8.3.4.12 &quot;Inline form for spline trigger type &quot;Trigger assignment&quot;&quot; Page 167)</td>
</tr>
<tr>
<td><strong>Trigger function call</strong></td>
<td>The trigger calls a subprogram. Only available in the user group “Expert”.</td>
</tr>
<tr>
<td></td>
<td>(&gt;&gt;&gt; 8.3.4.13 &quot;Inline form for spline trigger type &quot;Trigger function call&quot;&quot; Page 167)</td>
</tr>
</tbody>
</table>

Further information on triggers, on offsetting the switching point and on the offset limits can be found in the Operating and Programming Instructions for System Integrators.
8.3.4.10 Inline form for spline trigger type “Set output”

![Inline form for spline trigger type Set output](image)

Fig. 8-19: Inline form for spline trigger type Set output

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | If the statement is to be shifted in space, the desired distance from the start or end point must be specified here. If no shift in space is desired, enter the value 0.  
  - Positive value: shifts the statement towards the end of the motion.  
  - Negative value: shifts the statement towards the start of the motion.  
  Only for the user group “Expert”: **Toggle path** makes it possible to enter a variable, constant or function in this box. The functions are subject to constraints.  
  (>>> 8.3.4.14 "Limits for functions in the spline trigger" Page 168) |
| 2    | **Toggle OnStart** can be used to set or cancel the parameter ONSTART.  
  - Without ONSTART: the PATH value refers to the end point.  
  - With ONSTART: the PATH value refers to the start point. |
| 3    | If the statement is to be shifted in time (relative to the value in item 1), the desired duration must be specified here. If no shift in time is desired, enter the value 0.  
  - Positive value: shifts the statement towards the end of the motion. Maximum: 1,000 ms  
  - Negative value: shifts the statement towards the start of the motion.  
  Only for the user group “Expert”: **Toggle Delay** makes it possible to enter a variable, constant or function in this box. The functions are subject to constraints.  
  (>>> 8.3.4.14 "Limits for functions in the spline trigger" Page 168) |
| 4    | Output number  
  - 1 … 4096 |
| 5    | State to which the output is switched  
  - TRUE  
  - FALSE |
### 8.3.4.11 Inline form for spline trigger type “Set pulse output”

![Fig. 8-20: Inline form for spline trigger type Set pulse output](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | If the statement is to be shifted in space, the desired distance from the start or end point must be specified here. If no shift in space is desired, enter the value 0.  
  - Positive value: shifts the statement towards the end of the motion.  
  - Negative value: shifts the statement towards the start of the motion.  
  Only for the user group “Expert”: **Toggle path** makes it possible to enter a variable, constant or function in this box. The functions are subject to constraints.  
  (>>> 8.3.4.14 ”Limits for functions in the spline trigger” Page 168) |
| 2    | **Toggle OnStart** can be used to set or cancel the parameter ONSTART.  
  - Without ONSTART: the PATH value refers to the end point.  
  - With ONSTART: the PATH value refers to the start point. |
| 3    | If the statement is to be shifted in time (relative to the value in item 1), the desired duration must be specified here. If no shift in time is desired, enter the value 0.  
  - Positive value: shifts the statement towards the end of the motion. Maximum: 1,000 ms  
  - Negative value: shifts the statement towards the start of the motion.  
  Only for the user group “Expert”: **Toggle Delay** makes it possible to enter a variable, constant or function in this box. The functions are subject to constraints.  
  (>>> 8.3.4.14 ”Limits for functions in the spline trigger” Page 168) |
| 4    | Output number  
  - 1 ... 4096 |
| 5    | State to which the output is switched  
  - **TRUE**: “High” level  
  - **FALSE**: “Low” level |
| 6    | Length of the pulse  
  - 0.10 ... 3.00 s |
8.3.4.12 Inline form for spline trigger type “Trigger assignment”

Fig. 8-21: Inline form for spline trigger type Trigger assignment

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | If the statement is to be shifted in space, the desired distance from the start or end point must be specified here. If no shift in space is desired, enter the value 0.  
  - Positive value: shifts the statement towards the end of the motion.  
  - Negative value: shifts the statement towards the start of the motion.  
  Only for the user group “Expert”: Toggle path makes it possible to enter a variable, constant or function in this box. The functions are subject to constraints.  
  (>>> 8.3.4.14 "Limits for functions in the spline trigger" Page 168) |
| 2    | Toggle OnStart can be used to set or cancel the parameter ONSTART.  
  - Without ONSTART: the PATH value refers to the end point.  
  - With ONSTART: the PATH value refers to the start point. |
| 3    | If the statement is to be shifted in time (relative to the value in item 1), the desired duration must be specified here. If no shift in time is desired, enter the value 0.  
  - Positive value: shifts the statement towards the end of the motion. Maximum: 1,000 ms  
  - Negative value: shifts the statement towards the start of the motion.  
  Only for the user group “Expert”: Toggle Delay makes it possible to enter a variable, constant or function in this box. The functions are subject to constraints.  
  (>>> 8.3.4.14 "Limits for functions in the spline trigger" Page 168) |
| 4    | Variable to which a value is to be assigned  
  **Note:** Runtime variables cannot be used. |
| 5    | Value to be assigned to the variable |

8.3.4.13 Inline form for spline trigger type “Trigger function call”

Fig. 8-22: Inline form for spline trigger type Trigger function call
The values for PATH and DELAY can be assigned using functions. The following constraints apply to these functions:

- The KRL program containing the function must have the attribute **Hidden**.
- The function must be globally valid.
- The functions may only contain the following statements or elements:
  - Value assignments
  - IF statements
  - Comments
  - Blank lines
  - RETURN
  - Read system variable

### Item 3

**Description**

If the statement is to be shifted in time (relative to the value in item 1), the desired duration must be specified here. If no shift in time is desired, enter the value 0.

- Positive value: shifts the statement towards the end of the motion. Maximum: 1,000 ms
- Negative value: shifts the statement towards the start of the motion.

Only for the user group “Expert”: **Toggle Delay** makes it possible to enter a variable, constant or function in this box. The functions are subject to constraints.

(>>> 8.3.4.14 “Limits for functions in the spline trigger” Page 168)
8 Programming for user group "User" (inline forms)

8.3.5 Copying spline inline forms

Overview
The following copying operations can be carried out:
- Copy an individual motion into a spline block
- Copy a spline block
- Copy a spline segment to another spline block
- Copy a spline segment out of a spline block

Precondition
- Expert user group
- Program is selected or open.
- T1, T2 or AUT mode

Copy
Copy an individual motion into a spline block:
The following individual motions can be copied and pasted into a spline block:
- SLIN
- SCIRC
- LIN
- CIRC

Precondition:
- The following frame data (= data in the option window Frames) of the individual motion and the block are identical: Tool, Base and Interpolation mode

Copy spline block:
A spline block can be copied and pasted at a different point in the program. Only the empty block is pasted. It is not possible to copy a block and its contents at the same time.

The contents must be copied and pasted separately.

Copy a spline segment to another spline block:
One or more spline segments can be copied and pasted to another block.

Precondition:
- The following frame data (= data in the option window Frames) of the spline blocks are identical: Tool, Base and Interpolation mode

Copy a spline segment out of a spline block:
One or more spline segments can be copied and pasted outside a spline block. The motion types change as follows:

<table>
<thead>
<tr>
<th>Spline segment...</th>
<th>... becomes an individual motion</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLIN</td>
<td>SLIN</td>
</tr>
<tr>
<td>SCIRC</td>
<td>SCIRC</td>
</tr>
<tr>
<td>SPL</td>
<td>PTP</td>
</tr>
</tbody>
</table>

- For individual SLIN, SCIRC motions: the frame and motion data from the segment are applied if available; otherwise, the data from the spline block are applied.
- For individual PTP motion: the position and frame data from the SPL are applied to the PTP. Motion data are not applied.
8.3.6 Converting spline inline forms from 8.1

**Description**
More parameters can be set in spline inline forms in KSS 8.2 than in KSS 8.1. This allows more detailed determination of the motion characteristics.

Programs with inline forms from 8.1 can be used in 8.2. For this, values must be assigned to the new parameters. This is done by opening the inline form and closing it again. Default values are automatically assigned to all new parameters.

**Precondition**
- A program is selected.
- Operating mode T1

**Procedure**
1. Position the cursor in the line with the inline form.
2. Press **Change**. The inline form is opened. The default values are automatically set for all new parameters.
3. If required, modify the values.
4. Press **Cmd OK**.
5. Repeat steps 1 to 4 for all spline inline forms in the program.

8.4 Modifying motion parameters

**Precondition**
- A program is selected.
- Operating mode T1

**Procedure**
1. Position the cursor in the line containing the instruction that is to be changed.
2. Press **Change**. The inline form for this instruction is opened.
3. Modify parameters.
4. Save changes by pressing **Cmd Ok**.

8.5 Modifying the coordinates of a taught point

**Description**
The coordinates of a taught point can be modified. This is done by moving to the new position and overwriting the old point with the new position.

**Precondition**
- A program is selected.
- Operating mode T1

**Procedure**
1. Move the TCP to the desired position.
2. Position the cursor in the line containing the motion instruction that is to be changed.
3. Press **Change**. The inline form for this instruction is opened.
4. For PTP and LIN motions: Press **Touch Up** to accept the current position of the TCP as the new end point.
   For CIRC motions:
   - Press **Teach Aux** to accept the current position of the TCP as the new auxiliary point.
   - Or press **Teach End** to accept the current position of the TCP as the new end point.
5. Confirm the request for confirmation with **Yes**.
6. Save change by pressing **Cmd Ok**.
8.6 Programming logic instructions

8.6.1 Inputs/outputs

Digital inputs/outputs
The robot controller can manage up to 4096 digital inputs and 4096 digital outputs. The configuration is customer-specific.

Analog inputs/outputs
The robot controller can manage 32 analog inputs and 32 analog outputs. The configuration is customer-specific.

Permissible range of values for inputs/outputs: -1.0 to +1.0. This corresponds to a voltage range from -10 V to +10 V. If the value is exceeded, the input/output takes the maximum value and a message is displayed until the value is back in the permissible range.

The inputs/outputs are managed via the following system variables:

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
</table>

8.6.2 Setting a digital output - OUT

Precondition
- A program is selected.
- Operating mode T1

Procedure
1. Position the cursor in the line after which the logic instruction is to be inserted.
2. Select the menu sequence **Commands** > **Logic** > **OUT** > **OUT**.
3. Set the parameters in the inline form.
   (>>>> 8.6.3 "Inline form ‘OUT’" Page 171)
4. Save instruction with **Cmd Ok**.

8.6.3 Inline form “OUT”

The instruction sets a digital output.

![Fig. 8-23: Inline form “OUT”](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output number</td>
</tr>
<tr>
<td>1</td>
<td>1 … 4096</td>
</tr>
<tr>
<td>2</td>
<td>If a name exists for the output, this name is displayed. Only for the user group &quot;Expert&quot;: A name can be entered by pressing <strong>Long text</strong>. The name is freely selectable.</td>
</tr>
</tbody>
</table>
8.6.4 Setting a pulse output - PULSE

Precondition
- A program is selected.
- Operating mode T1

Procedure
1. Position the cursor in the line after which the logic instruction is to be inserted.
2. Select the menu sequence Commands > Logic > OUT > PULSE.
3. Set the parameters in the inline form.
   (>>> 8.6.5 "Inline form “PULSE”” Page 172)
4. Save instruction with Cmd Ok.

8.6.5 Inline form “PULSE”

The instruction sets a pulse of a defined length.

![Fig. 8-24: Inline form “PULSE”](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output number</td>
</tr>
<tr>
<td>2</td>
<td>If a name exists for the output, this name is displayed. Only for the user group “Expert”: A name can be entered by pressing Long text. The name is freely selectable.</td>
</tr>
<tr>
<td>3</td>
<td>State to which the output is switched</td>
</tr>
<tr>
<td>4</td>
<td>CONT: Execution in the advance run</td>
</tr>
<tr>
<td>5</td>
<td>Length of the pulse</td>
</tr>
</tbody>
</table>

8.6.6 Setting an analog output - ANOUT

Precondition
- A program is selected.
- Operating mode T1

Procedure
1. Position the cursor in the line after which the instruction is to be inserted.
2. Select Commands > Analog output > Static or Dynamic.
3. Set the parameters in the inline form.
   (>>> 8.6.7 "Inline form “ANOUT” (static)" Page 173)
   (>>> 8.6.8 "Inline form “ANOUT” (dynamic)" Page 173)
4. Save instruction with Cmd Ok.

8.6.7 Inline form “ANOUT” (static)

This instruction sets a static analog output.

A maximum of 8 analog outputs (static and dynamic together) can be used at any one time. ANOUT triggers an advance run stop.

The voltage is set to a fixed level by means of a factor. The actual voltage level depends on the analog module used. For example, a 10 V module with a factor of 0.5 provides a voltage of 5 V.

![Fig. 8-25: Inline form “ANOUT” (static)](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Number of the analog output</td>
</tr>
<tr>
<td></td>
<td>CHANNEL_1 … CHANNEL_32</td>
</tr>
<tr>
<td>2</td>
<td>Factor for the voltage</td>
</tr>
<tr>
<td></td>
<td>0 … 1 (intervals: 0.01)</td>
</tr>
</tbody>
</table>

8.6.8 Inline form “ANOUT” (dynamic)

This instruction activates or deactivates a dynamic analog output.

A maximum of 4 dynamic analog outputs can be activated at any one time. ANOUT triggers an advance run stop.

The voltage is determined by a factor. The actual voltage level depends on the following values:

- Velocity or function generator
  For example, a velocity of 1 m/s with a factor of 0.5 results in a voltage of 5 V.
- Offset
  For example, an offset of +0.15 for a voltage of 0.5 V results in a voltage of 6.5 V.

![Fig. 8-26: Inline form “ANOUT” (dynamic)](image)
8.6.9 Programming a wait time - WAIT

Precondition
- A program is selected.
- Operating mode T1

Procedure
1. Position the cursor in the line after which the logic instruction is to be inserted.
2. Select the menu sequence Commands > Logic > WAIT.
3. Set the parameters in the inline form.
   (>>> 8.6.10 "Inline form "WAIT"" Page 174)
4. Save instruction with Cmd Ok.

8.6.10 Inline form “WAIT”

WAIT can be used to program a wait time. The robot motion is stopped for a programmed time. WAIT always triggers an advance run stop.

![WAIT time](image)

Fig. 8-27: Inline form “WAIT”

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Wait time</td>
</tr>
<tr>
<td></td>
<td>≥ 0 s</td>
</tr>
</tbody>
</table>

8.6.11 Programming a signal-dependent wait function - WAITFOR

Precondition
- A program is selected.
- Operating mode T1

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Wait time</td>
</tr>
<tr>
<td></td>
<td>≥ 0 s</td>
</tr>
</tbody>
</table>
Procedure

1. Position the cursor in the line after which the logic instruction is to be inserted.
2. Select the menu sequence Commands > Logic > WAITFOR.
3. Set the parameters in the inline form.
(>>> 8.6.12 "Inline form “WAITFOR”" Page 175)
4. Save instruction with Cmd Ok.

8.6.12 Inline form “WAITFOR”

The instruction sets a signal-dependent wait function.

If required, several signals (maximum 12) can be linked. If a logic operation is added, boxes are displayed in the inline form for the additional signals and links.

Fig. 8-28: Inline form “WAITFOR”

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Add external logic operation. The operator is situated between the bracketed expressions.</td>
</tr>
<tr>
<td></td>
<td>- AND</td>
</tr>
<tr>
<td></td>
<td>- OR</td>
</tr>
<tr>
<td></td>
<td>- EXOR</td>
</tr>
<tr>
<td>Add NOT.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- NOT</td>
</tr>
<tr>
<td></td>
<td>- [Empty box]</td>
</tr>
<tr>
<td>Enter the desired operator by means of the corresponding button.</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Add internal logic operation. The operator is situated inside a bracketed expression.</td>
</tr>
<tr>
<td></td>
<td>- AND</td>
</tr>
<tr>
<td></td>
<td>- OR</td>
</tr>
<tr>
<td></td>
<td>- EXOR</td>
</tr>
<tr>
<td>Add NOT.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- NOT</td>
</tr>
<tr>
<td></td>
<td>- [Empty box]</td>
</tr>
<tr>
<td>Enter the desired operator by means of the corresponding button.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Signal for which the system is waiting</td>
</tr>
<tr>
<td></td>
<td>- IN</td>
</tr>
<tr>
<td></td>
<td>- OUT</td>
</tr>
<tr>
<td></td>
<td>- CYCFLAG</td>
</tr>
<tr>
<td></td>
<td>- TIMER</td>
</tr>
<tr>
<td></td>
<td>- FLAG</td>
</tr>
<tr>
<td>4</td>
<td>Number of the signal</td>
</tr>
<tr>
<td></td>
<td>- 1 … 4096</td>
</tr>
</tbody>
</table>
8.6.13 Switching on the path - SYN OUT

Precondition

- A program is selected.
- Operating mode T1

Procedure

1. Position the cursor in the line after which the logic instruction is to be inserted.
2. Select the menu sequence **Commands > Logic > OUT > SYN OUT**.
3. Set the parameters in the inline form.
   (>>> 8.6.14 "Inline form “SYN OUT”, option “START/END”" Page 176)
   (>>> 8.6.15 "Inline form “SYN OUT”, option “PATH”" Page 179)
4. Save instruction with **Cmd Ok**.

8.6.14 Inline form “SYN OUT”, option “START/END”

A switching action can be triggered relative to the start or end point of a motion block. The switching action can be delayed or brought forward. The motion block can be a LIN, CIRC or PTP motion.

Possible applications include:

- Closing or opening the weld gun during spot welding
- Switching the welding current on/off during arc welding
- Starting or stopping the flow of adhesive in bonding or sealing applications.

![Fig. 8-29: Inline form “SYN OUT”, option “START/END”](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output number</td>
</tr>
<tr>
<td></td>
<td>1 … 4096</td>
</tr>
<tr>
<td>2</td>
<td>If a name exists for the output, this name is displayed. Only for the user group “Expert”: A name can be entered by pressing <strong>Long text</strong>. The name is freely selectable.</td>
</tr>
<tr>
<td>3</td>
<td>State to which the output is switched</td>
</tr>
<tr>
<td></td>
<td>TRUE</td>
</tr>
<tr>
<td></td>
<td>FALSE</td>
</tr>
</tbody>
</table>
### Example 1

Start point and end point are exact positioning points.

```
LIN P1 VEL=0.3m/s CPDAT1
LIN P2 VEL=0.3m/s CPDAT2
SYN OUT 1 '' State= TRUE at START Delay=20ms
SYN OUT 2 '' State= TRUE at END Delay=-20ms
LIN P3 VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
```

![Fig. 8-30](image)

OUT 1 and OUT 2 specify approximate positions at which switching is to occur. The dotted lines indicate the switching limits.

Switching limits:

- **START**: The switching point can be delayed, at most, as far as exact positioning point P3 (+ ms).
- **END**: The switching point can be brought forward, at most, as far as exact positioning point P2 (- ms).

If greater values are specified for the delay, the controller automatically switches at the switching limit.

### Example 2

Start point is exact positioning point, end point is approximated.

```
LIN P1 VEL=0.3m/s CPDAT1
LIN P2 VEL=0.3m/s CPDAT2
SYN OUT 1 '' State= TRUE at START Delay=20ms
SYN OUT 2 '' State= TRUE at END Delay=-20ms
LIN P3 CONT VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
```
OUT 1 and OUT 2 specify approximate positions at which switching is to occur. The dotted lines indicate the switching limits. M = middle of the approximate positioning range.

Switching limits:

- **START:** The switching point can be delayed, at most, as far as the start of the approximate positioning range of P3 (+ ms).
- **END:** The switching point can be brought forward, at most, as far as the start of the approximate positioning range of P3 (-).
  
  The switching point can be delayed, at most, as far as the end of the approximate positioning range of P3 (+).

If greater values are specified for the delay, the controller automatically switches at the switching limit.

**Example 3**

Start point and end point are approximated

```
LIN P1 VEL=0.3m/s CPDAT1
LIN P2 CONT VEL=0.3m/s CPDAT2
SYN OUT 1 '' State= TRUE at START Delay=20ms
SYN OUT 2 '' State= TRUE at END Delay=-20ms
LIN P3 CONT VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
```

OUT 1 and OUT 2 specify approximate positions at which switching is to occur. The dotted lines indicate the switching limits. M = middle of the approximate positioning range.
Switching limits:

- **START**: The switching point can be situated, at the earliest, at the end of the approximate positioning range of P2.
  The switching point can be delayed, at most, as far as the start of the approximate positioning range of P3 (+ ms).
- **END**: The switching point can be brought forward, at most, as far as the start of the approximate positioning range of P3 (-).
  The switching point can be delayed, at most, as far as the end of the approximate positioning range of P3 (+).

If greater values are specified for the delay, the controller automatically switches at the switching limit.

### 8.6.15 Inline form “SYN OUT”, option “PATH”

A switching action can be triggered relative to the end point of a motion block. The switching action can be shifted in space and delayed or brought forward. The motion block can be a LIN or CIRC motion. It must not be a PTP motion.

![Fig. 8-33: Inline form “SYN OUT”, option “PATH”](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output number</td>
</tr>
<tr>
<td></td>
<td>1 … 4096</td>
</tr>
<tr>
<td>2</td>
<td>If a name exists for the output, this name is displayed. Only for the user group “Expert”: A name can be entered by pressing Long text. The name is freely selectable.</td>
</tr>
<tr>
<td>3</td>
<td>State to which the output is switched</td>
</tr>
<tr>
<td></td>
<td>TRUE</td>
</tr>
<tr>
<td></td>
<td>FALSE</td>
</tr>
<tr>
<td>4</td>
<td>Point at which switching is carried out</td>
</tr>
<tr>
<td></td>
<td><strong>PATH</strong>: Switching is carried out at the end point of the motion block.</td>
</tr>
<tr>
<td></td>
<td><strong>START</strong>: (&gt;&gt;&gt; 8.6.14 &quot;Inline form “SYN OUT”, option “START/END” Page 176)</td>
</tr>
<tr>
<td></td>
<td><strong>END</strong>: (&gt;&gt;&gt; 8.6.14 &quot;Inline form “SYN OUT”, option “START/END” Page 176)</td>
</tr>
<tr>
<td>5</td>
<td>Distance from the switching point to the end point</td>
</tr>
<tr>
<td></td>
<td>-2,000 … +2,000 mm</td>
</tr>
<tr>
<td></td>
<td>This box is only displayed if <strong>PATH</strong> has been selected.</td>
</tr>
<tr>
<td>6</td>
<td>Switching action delay</td>
</tr>
<tr>
<td></td>
<td>-1,000 … +1,000 ms</td>
</tr>
</tbody>
</table>

**Note**: The time specification is absolute. The switching point varies according to the velocity of the robot.

**Example 1**

Start point is exact positioning point, end point is approximated.
OUT 1 specifies the approximate position at which switching is to occur. The dotted lines indicate the switching limits. M = middle of the approximate positioning range.

Switching limits:
- The switching point can be brought forward, at most, as far as exact positioning point P1.
- The switching point can be delayed, at most, as far as the next exact positioning point P4. If P3 was an exact positioning point, the switching point could be delayed, at most, as far as P3.

If greater values are specified for the shift in space or time, the controller automatically switches at the switching limit.

**Example 2**

Start point and end point are approximated
OUT 1 specifies the approximate position at which switching is to occur. The dotted lines indicate the switching limits. M = middle of the approximate positioning range.

Switching limits:

- The switching point can be brought forward, at most, as far as the start of the approximate positioning range of P1.
- The switching point can be delayed, at most, as far as the next exact positioning point P4. If P3 was an exact positioning point, the switching point could be delayed, at most, as far as P3.

If greater values are specified for the shift in space or time, the controller automatically switches at the switching limit.

8.6.16 Setting a pulse on the path - SYN PULSE

Precondition

- A program is selected.
- Operating mode T1

Procedure

1. Position the cursor in the line after which the logic instruction is to be inserted.
2. Select the menu sequence Commands > Logic > OUT > SYN PULSE.
3. Set the parameters in the inline form.
   (>>> 8.6.17 "Inline form “SYN PULSE”" Page 181)
4. Save instruction with Cmd Ok.

8.6.17 Inline form “SYN PULSE”

A pulse can be triggered relative to the start or end point of a motion block. The pulse can be delayed or brought forward and shifted in space.

Fig. 8-36: Inline form “SYN PULSE”

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output number</td>
</tr>
<tr>
<td></td>
<td>1 … 4096</td>
</tr>
<tr>
<td>2</td>
<td>If a name exists for the output, this name is displayed. Only for the user group &quot;Expert&quot;: A name can be entered by pressing Long text. The name is freely selectable.</td>
</tr>
<tr>
<td>3</td>
<td>State to which the output is switched</td>
</tr>
<tr>
<td></td>
<td>TRUE</td>
</tr>
<tr>
<td></td>
<td>FALSE</td>
</tr>
<tr>
<td>4</td>
<td>Duration of the pulse</td>
</tr>
<tr>
<td></td>
<td>0.1 … 3 s</td>
</tr>
</tbody>
</table>
8.6.18 Modifying a logic instruction

Precondition
- A program is selected.
- Operating mode T1

Procedure
1. Position the cursor in the line containing the instruction that is to be changed.
2. Press the Change softkey. The inline form for this instruction is opened.
3. Modify parameters.
4. Save changes by pressing the Cmd Ok softkey.

### Table

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| 5    | - **START**: The pulse is triggered at the start point of the motion block.  
- **END**: The pulse is triggered at the end point of the motion block.  
See SYN OUT for examples and switching limits.  
(>>> 8.6.14 "Inline form “SYN OUT”, option “START/END”" Page 176)  
- **PATH**: The pulse is triggered at the end point of the motion block.  
See SYN OUT for examples and switching limits.  
(>>> 8.6.15 "Inline form “SYN OUT”, option “PATH” Page 179) |
| 6    | Distance from the switching point to the end point  
- **-2,000 … +2,000 mm**  
This box is only displayed if **PATH** has been selected. |
| 7    | Switching action delay  
- **-1,000 … +1,000 ms**  
**Note**: The time specification is absolute. The switching point varies according to the velocity of the robot. |
## 9 Messages

### 9.1 Automatic External error messages

<table>
<thead>
<tr>
<th>No.</th>
<th>Message text</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>P00:1</td>
<td>PGNO_TYPE incorrect value permissible values (1,2,3)</td>
<td>The data type for the program number was entered incorrectly.</td>
</tr>
<tr>
<td>P00:2</td>
<td>PGNO_LENGTH incorrect value</td>
<td>The selected program number length in bits was too high.</td>
</tr>
<tr>
<td></td>
<td>Range of values $1 \leq \text{PGNO}_\text{LENGTH} \leq 16$</td>
<td></td>
</tr>
<tr>
<td>P00:3</td>
<td>PGNO_LENGTH incorrect value permissible values (4,8,12,16)</td>
<td>If BCD format was selected for reading the program number, a corresponding number of bits must also be set.</td>
</tr>
<tr>
<td>P00:4</td>
<td>PGNO_FBIT incorrect value not in the $\text{IN}$ range</td>
<td>The value “0” or a non-existent input was specified for the first bit of the program number.</td>
</tr>
<tr>
<td>P00:7</td>
<td>PGNO_REQ incorrect value not in the $\text{OUT}$ range</td>
<td>The value “0” or a non-existent output was specified for the output via which the program number is to be requested.</td>
</tr>
<tr>
<td>P00:10</td>
<td>Transmission error incorrect parity</td>
<td>Discrepancy detected when checking parity. A transmission error must have occurred.</td>
</tr>
<tr>
<td>P00:11</td>
<td>Transmission error incorrect program number</td>
<td>The higher-level controller has transferred a program number for which there is no CASE branch in the file CELL.SRC.</td>
</tr>
<tr>
<td>P00:12</td>
<td>Transmission error incorrect BCD encoding</td>
<td>The attempt to read the program number in BCD format led to an invalid result.</td>
</tr>
<tr>
<td>P00:13</td>
<td>Incorrect operating mode</td>
<td>The I/O interface output has not been activated, i.e. the system variable $\text{I_O_ACTCONF}$ currently has the value FALSE. This can have the following causes:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ The mode selector switch is not in the “Automatic External” position.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ The signal $\text{I_O_ACT}$ currently has the value FALSE.</td>
</tr>
<tr>
<td>P00:14</td>
<td>Move to Home position in operating mode T1</td>
<td>The robot has not reached the HOME position.</td>
</tr>
<tr>
<td>P00:15</td>
<td>Incorrect program number</td>
<td>More than one input set with “1 of n”.</td>
</tr>
</tbody>
</table>
10 KUKA Service

10.1 Requesting support

Introduction
The KUKA Roboter GmbH documentation offers information on operation and provides assistance with troubleshooting. For further assistance, please contact your local KUKA subsidiary.

Information
The following information is required for processing a support request:

- Model and serial number of the robot
- Model and serial number of the controller
- Model and serial number of the linear unit (if applicable)
- Version of the KUKA System Software
- Optional software or modifications
- Archive of the software
- Application used
- Any external axes used
- Description of the problem, duration and frequency of the fault

10.2 KUKA Customer Support

Availability
KUKA Customer Support is available in many countries. Please do not hesitate to contact us if you have any questions.

Argentina
Ruben Costantini S.A. (Agency)
Luis Angel Huergo 13 20
Parque Industrial
2400 San Francisco (CBA)
Argentina
Tel. +54 3564 421033
Fax +54 3564 428877
ventas@costantini-sa.com

Australia
Headland Machinery Pty. Ltd.
Victoria (Head Office & Showroom)
95 Highbury Road
Burwood
Victoria 31 25
Australia
Tel. +61 3 9244-3500
Fax +61 3 9244-3501
vic@headland.com.au
www.headland.com.au
Belgium
KUKA Automatisering + Robots N.V.
Centrum Zuid 1031
3530 Houthalen
Belgium
Tel. +32 11 516160
Fax +32 11 526794
info@kuka.be
www.kuka.be

Brazil
KUKA Roboter do Brasil Ltda.
Avenida Franz Liszt, 80
Parque Novo Mundo
Jd. Guançã
CEP 02151 900 São Paulo
SP Brazil
Tel. +55 11 69844900
Fax +55 11 62017883
info@kuka-roboter.com.br

Chile
Robotec S.A. (Agency)
Santiago de Chile
Chile
Tel. +56 2 331-5951
Fax +56 2 331-5952
robotec@robotec.cl
www.robotec.cl

China
KUKA Automation Equipment (Shanghai) Co., Ltd.
Songjiang Industrial Zone
No. 388 Minshen Road
201612 Shanghai
China
Tel. +86 21 6787-1808
Fax +86 21 6787-1805
info@kuka-sha.com.cn
www.kuka.cn

Germany
KUKA Roboter GmbH
Zugspitzstr. 140
86165 Augsburg
Germany
Tel. +49 821 797-4000
Fax +49 821 797-1616
info@kuka-roboter.de
www.kuka-roboter.de
France
KUKA Automatisme + Robotique SAS
Techvallée
6, Avenue du Parc
91140 Villebon S/Yvette
France
Tel. +33 1 6931660-0
Fax +33 1 6931660-1
commercial@kuka.fr
www.kuka.fr

India
KUKA Robotics India Pvt. Ltd.
Office Number-7, German Centre,
Level 12, Building No. - 9B
DLF Cyber City Phase III
122 002 Gurgaon
Haryana
India
Tel. +91 124 4635774
Fax +91 124 4635773
info@kuka.in
www.kuka.in

Italy
KUKA Roboter Italia S.p.A.
Via Pavia 9/a - int.6
10098 Rivoli (TO)
Italy
Tel. +39 011 959-5013
Fax +39 011 959-5141
kuka@kuka.it
www.kuka.it

Japan
KUKA Robotics Japan K.K.
Daiba Garden City Building 1F
2-3-5 Daiba, Minato-ku
Tokyo
135-0091
Japan
Tel. +81 3 6380-7311
Fax +81 3 6380-7312
info@kuka.co.jp

Korea
KUKA Robotics Korea Co. Ltd.
RIT Center 306, Gyeonggi Technopark
1271-11 Sa 3-dong, Sangnok-gu
Ansan City, Gyeonggi Do
426-901
Korea
Tel. +82 31 501-1451
Fax +82 31 501-1461
info@kukakorea.com
<table>
<thead>
<tr>
<th>Country</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Malaysia</td>
<td>KUKA Robot Automation Sdn Bhd</td>
</tr>
<tr>
<td></td>
<td>South East Asia Regional Office</td>
</tr>
<tr>
<td></td>
<td>No. 24, Jalan TPP 1/10</td>
</tr>
<tr>
<td></td>
<td>Taman Industri Puchong</td>
</tr>
<tr>
<td></td>
<td>47100 Puchong</td>
</tr>
<tr>
<td></td>
<td>Selangor</td>
</tr>
<tr>
<td></td>
<td>Malaysia</td>
</tr>
<tr>
<td></td>
<td>Tel. +60 3 8061-0613 or -0614</td>
</tr>
<tr>
<td></td>
<td>Fax +60 3 8061-7386</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:info@kuka.com.my">info@kuka.com.my</a></td>
</tr>
<tr>
<td>Mexico</td>
<td>KUKA de Mexico S. de R.L. de C.V.</td>
</tr>
<tr>
<td></td>
<td>Rio San Joaquin #339, Local 5</td>
</tr>
<tr>
<td></td>
<td>Colonia Pensil Sur</td>
</tr>
<tr>
<td></td>
<td>C.P. 11490 Mexico D.F.</td>
</tr>
<tr>
<td></td>
<td>Mexico</td>
</tr>
<tr>
<td></td>
<td>Tel. +52 55 5203-8407</td>
</tr>
<tr>
<td></td>
<td>Fax +52 55 5203-8148</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:info@kuka.com.mx">info@kuka.com.mx</a></td>
</tr>
<tr>
<td>Norway</td>
<td>KUKA Sveisætning + Roboter</td>
</tr>
<tr>
<td></td>
<td>Bryggeveien 9</td>
</tr>
<tr>
<td></td>
<td>2821 Gjøvik</td>
</tr>
<tr>
<td></td>
<td>Norway</td>
</tr>
<tr>
<td></td>
<td>Tel. +47 61 133422</td>
</tr>
<tr>
<td></td>
<td>Fax +47 61 186200</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:geir.ulsrud@kuka.no">geir.ulsrud@kuka.no</a></td>
</tr>
<tr>
<td>Austria</td>
<td>KUKA Roboter Austria GmbH</td>
</tr>
<tr>
<td></td>
<td>Vertriebsbüro Österreich</td>
</tr>
<tr>
<td></td>
<td>Regensburger Strasse 9/1</td>
</tr>
<tr>
<td></td>
<td>4020 Linz</td>
</tr>
<tr>
<td></td>
<td>Austria</td>
</tr>
<tr>
<td></td>
<td>Tel. +43 732 784752</td>
</tr>
<tr>
<td></td>
<td>Fax +43 732 793880</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:office@kuka-roboter.at">office@kuka-roboter.at</a></td>
</tr>
<tr>
<td></td>
<td><a href="http://www.kuka-roboter.at">www.kuka-roboter.at</a></td>
</tr>
<tr>
<td>Poland</td>
<td>KUKA Roboter Austria GmbH</td>
</tr>
<tr>
<td></td>
<td>Spółka z ograniczoną odpowiedzialności</td>
</tr>
<tr>
<td></td>
<td>Oddział w Polsce</td>
</tr>
<tr>
<td></td>
<td>Ul. Porcelanowa 10</td>
</tr>
<tr>
<td></td>
<td>40-246 Katowice</td>
</tr>
<tr>
<td></td>
<td>Poland</td>
</tr>
<tr>
<td></td>
<td>Tel. +48 327 30 32 13 or -14</td>
</tr>
<tr>
<td></td>
<td>Fax +48 327 30 32 26</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:ServicePL@kuka-roboter.de">ServicePL@kuka-roboter.de</a></td>
</tr>
</tbody>
</table>
Portugal
KUKA Sistemas de Automatización S.A.
Rua do Alto da Guerra n° 50
Armazém 04
2910 011 Setúbal
Portugal
Tel. +351 265 729780
Fax +351 265 729782
kuka@mail.telepac.pt

Russia
OOO KUKA Robotics Rus
Webnaja ul. 8A
107143 Moskau
Russia
Tel. +7 495 781-31-20
Fax +7 495 781-31-19
kuka-robotics.ru

Sweden
KUKA Svetsanläggningar + Robotar AB
A. Odhners gata 15
421 30 Västra Frölunda
Sweden
Tel. +46 31 7266-200
Fax +46 31 7266-201
info@kuka.se

Switzerland
KUKA Roboter Schweiz AG
Industriestr. 9
5432 Neuenhof
Switzerland
Tel. +41 44 74490-90
Fax +41 44 74490-91
info@kuka-roboter.ch
www.kuka-roboter.ch

Spain
KUKA Robots IBÉRICA, S.A.
Pol. Industrial
Torrent de la Pastera
Carrer del Bages s/n
08800 Vilanova i la Geltrú (Barcelona)
Spain
Tel. +34 93 8142-353
Fax +34 93 8142-950
Comercial@kuka-e.com
www.kuka-e.com
South Africa  
Jendamark Automation LTD (Agency)  
76a York Road  
North End  
6000 Port Elizabeth  
South Africa  
Tel. +27 41 391 4700  
Fax +27 41 373 3869  
www.jendamark.co.za

Taiwan  
KUKA Robot Automation Taiwan Co., Ltd.  
No. 249 Pujong Road  
Jungli City, Taoyuan County 320  
Taiwan, R. O. C.  
Tel. +886 3 4331988  
Fax +886 3 4331948  
info@kuka.com.tw  
www.kuka.com.tw

Thailand  
KUKA Robot Automation (M)SdnBhd  
Thailand Office  
c/o Maccall System Co. Ltd.  
49/9-10 Soi Kingkaew 30 Kingkaew Road  
Tt. Rachatheva, A. Bangpli  
Samutprakarn  
10540 Thailand  
Tel. +66 2 7502737  
Fax +66 2 6612355  
atika@ji-net.com  
www.kuka-roboter.de

Czech Republic  
KUKA Roboter Austria GmbH  
Organisation Tschechien und Slowakei  
Sezemická 2757/2  
193 00 Praha  
Horní Počernice  
Czech Republic  
Tel. +420 22 62 12 27 2  
Fax +420 22 62 12 27 0  
support@kuka.cz

Hungary  
KUKA Robotics Hungaria Kft.  
Fő út 140  
2335 Taksony  
Hungary  
Tel. +36 24 501609  
Fax +36 24 477031  
info@kuka-robotics.hu
USA  
KUKA Robotics Corp.  
22500 Key Drive  
Clinton Township  
48036  
Michigan  
USA  
Tel. +1 866 8735852  
Fax +1 586 5692087  
info@kukarobotics.com  
www.kukarobotics.com

UK  
KUKA Automation + Robotics  
Hereward Rise  
Halesowen  
B62 8AN  
UK  
Tel. +44 121 585-0800  
Fax +44 121 585-0900  
sales@kuka.co.uk
Index

Symbols
#BSTEP 114
#ISTEP 114
#MSTEP 114
$ANIN 171
$ANOUT 171
$IN 171
$OUT 171
$ROBRUNTIME 67, 68

Numbers
2004/108/EC 35
2006/42/EC 35
3-point method 89
89/336/EEC 35
95/16/EC 35
97/23/EC 35

A
ABC 2-point method 86
ABC World method 85
Accessories 11, 13
Actual position 62
Administrator 48
Advance run 115
ANOUT 172
Applied norms and regulations 35
Approximate positioning 128, 152
Archiving overview 121
Archiving, logbook 124
Archiving, network 123
Archiving, to USB stick 123
Automatic External error messages 183
Automatic mode 31
Auxiliary point 128
Axis range 15
Axis range limitation 24
Axis range monitoring 24

B
Backward motion 117
Base calibration 88
BASE coordinate system 50, 88
Block pointer 111
Block selection 117, 134
Brake defect 26
Braking distance 15

C
Calibrating an external kinematic system 98
Calibration 82
Calibration points (menu item) 67
Calibration, base 88
Calibration, external TCP 91
Calibration, fixed tool 91
Calibration, linear unit 97
Calibration, root point, kinematic system 99
Calibration, tool 82
Calibration, TOOL kinematic system 103
Calibration, workpiece 91
CE mark 14
CELL.SRC 118
CIRC motion 149
CIRC, motion type 128
Circular angle 157, 162
Cleaning work 32
Cold start 47
Comment 120
Connecting cables 11, 13
Connection manager 38
Continuous Path 127
Coordinate system for jog keys 42
Coordinate system for Space Mouse 41
Coordinate systems 49
Coordinate systems, angles 51
Coordinate systems, orientation 51
Copy 121
Counterbalancing system 33
CP motions 127
Creating a new folder 108
Creating a new program 109
Cut 121

D
Danger zone 15
Data, restoring 124
Declaration of conformity 14
Declaration of incorporation 13, 14
Decommissioning 33
DEF line (menu item) 113
DEF line, displaying/hiding 113
Deleting mastering 81
Deselecting a program 110
Detail view (ASCII) (menu item) 114
Detail view, activating 113
Dial gauge 78
Directory structure 107
Disabling the robot controller 48
Displaying, robot controller information 67
Displaying, robot information 67
Disposal 33
Documentation, industrial robot 9

E
EC declaration of conformity 14
Edit (button) 42
Editing a program 118
Editor 109
Electronic Mastering Device 73
EMC Directive 14, 35
EMD 73
EMERGENCY STOP 38
EMERGENCY STOP button 21, 29
EMERGENCY STOP device 21, 22, 26
EMERGENCY STOP, external 22, 29
EMERGENCY STOP, local 29
N
Name, archive 68
Name, control PC 67
Name, robot 67, 68
Navigator 107
Numeric entry, external TCP 93
Numeric entry, external tool 104
Numeric entry, linear unit 98
Numeric entry, root point, kinematic system 100
Numeric input, base 90
Numeric input, tool 88

O
Offset 74, 76, 173
Opening a program 109
Operating hours 68
Operating hours meter 68
Operating mode, changing 49
Operation 37
Operator 15, 17, 48
Operator safety 19, 20, 26
Options 11, 13
Orientation control 152
Orientation control (spline) 158, 161, 164
Orientation control, LIN, CIRC 130
Orientation control, SPLINE 141
OUT 171
Output, analog 172
Output, digital 171
Overload 26
Override 55, 115
Override (menu item) 62
Overview of the industrial robot 11

P
Palletizing robots 83, 88
Panic position 22
Paste 121
Payload data 105
Payload data (menu item) 105
Performance Level 19
Personnel 16
Plant integrator 16
Point-to-point 127
Positionally accurate robot, checking activation 70
Positioner 13, 98
POV 115
Pre-mastering position 72
Pressure Equipment Directive 33, 35
Preventive maintenance work 32
Printing a program 121
Product description 11
Program lines, deleting 120
Program management 107
Program override 115
Program run mode, selecting 114
Program run modes 114
Program, closing 111
Program, creating 109
Program, deselecting 110
Program, opening 109
Program, selecting 109
Program, starting 116
Programmer 48
Programming, inline forms 147
Programming, User 147
Protective equipment 23
PTP motion 147
PTP, motion type 127
PULSE 172
Pulse 172
Pulse, path-related 181

R
Rating plate 69
RDC, exchange 80
Reaction distance 15
Recommissioning 28, 69
Reference mastering 80
Release device 24
Renaming a file 109
Renaming the base 95
Renaming the tool 95
Repair 32
Replace 121
Resetting a program 118
Robot controller 11, 13
Robot data (menu item) 68
ROBROOT coordinate system 49

S
Safe operational stop 15, 23
Safeguards, external 25
Safety 13
Safety controller 19
Safety functions 26
Safety functions, overview 19
Safety instructions 9
Safety STOP 0 15
Safety STOP 1 15
Safety STOP 2 15
Safety STOP 0 15
Safety STOP 1 15
Safety STOP 2 15
Safety stop, external 23
Safety zone 15, 17, 18
Safety, general 13
SCIRC motion, programming 156
SCIRC segment, programming 161
Selecting a program 109
Selecting the base 56
Selecting the tool 56
Serial number 68
Service life 67
Service, KUKA Roboter 185
Shutdown (menu item) 45
Simulation 31
Single point of control 33
Singularities 145
SLIN motion, programming 154
SLIN segment, programming 161
smartHMI 12, 41
smartPAD 15, 37
Software 11, 13
Software components 11
Software limit switches 23, 26, 81
Space Mouse 38, 51, 56, 58, 59
Special characters 147
SPL segment, programming 161
Spline block, programming 158
Spline motion, orientation control 158, 161, 164
Spline, motion type 131
SPOC 33
Stamp 120
Start backwards key 38
Start key 38, 39
Start-up 28, 69
Start-up mode 30
Start-up wizard 69
Starting a program, automatic 116
Starting a program, backwards 117
Starting a program, manual 116
Starting Automatic External mode 118
Starting the KSS 44
Status bar 41, 42, 107
STOP 0 14, 16
STOP 1 14, 16
STOP 2 14, 16
Stop category 0 16
Stop category 1 16
Stop category 2 16
STOP key 38
Stop reactions 18
Stopping a program 116, 117, 118
Stopping distance 15, 18
Storage 33
Storage capacities 67
Submit interpreter 43
Submit interpreter, status indicator 43
Supplementary load data (menu item) 106
Support request 185
Switching action, path-related 176
Switching on the robot controller 44
SYN OUT 176
SYN PULSE 181
System integrator 14, 16
Touch screen 37, 44
Trademarks 9
Training 9
Transport position 28
Transportation 28
TRIGGER, for spline 164
Turn-tilt table 13, 98
Type, robot 67
Type, robot controller 67
Unmastering 81
USB connection 39
Use, contrary to intended use 13
Use, improper 13
User 15, 16
User group, changing 48
User group, default 48
User interface 41
Velocity 55, 115
Velocity monitoring 23
Version, kernel system 67
Version, operating system 67
Version, robot controller 67
Version, user interface 67
Voltage 65, 171, 173, 174
WAIT 174
Wait function, signal-dependent 174
Wait time 174
WAITFOR 174
Warnings 9
Working range limitation 24
Workpiece base calibration 101
Workpiece base, numeric entry 103
Workspace 15, 17, 18
Workspace monitoring, bypassing 61
WORLD coordinate system 49
XYZ 4-point method 83
XYZ Reference method 84

T
T1 16
T2 16
Target group 9
TCP 82
TCP, external 91
Teach pendant 11, 13
Teaching 170
Technology keys 38
Technology packages 12, 67, 147
Terms used, safety 14
Tool calibration 82
Tool Center Point 82
TOOL coordinate system 50, 82
Tool, external 103