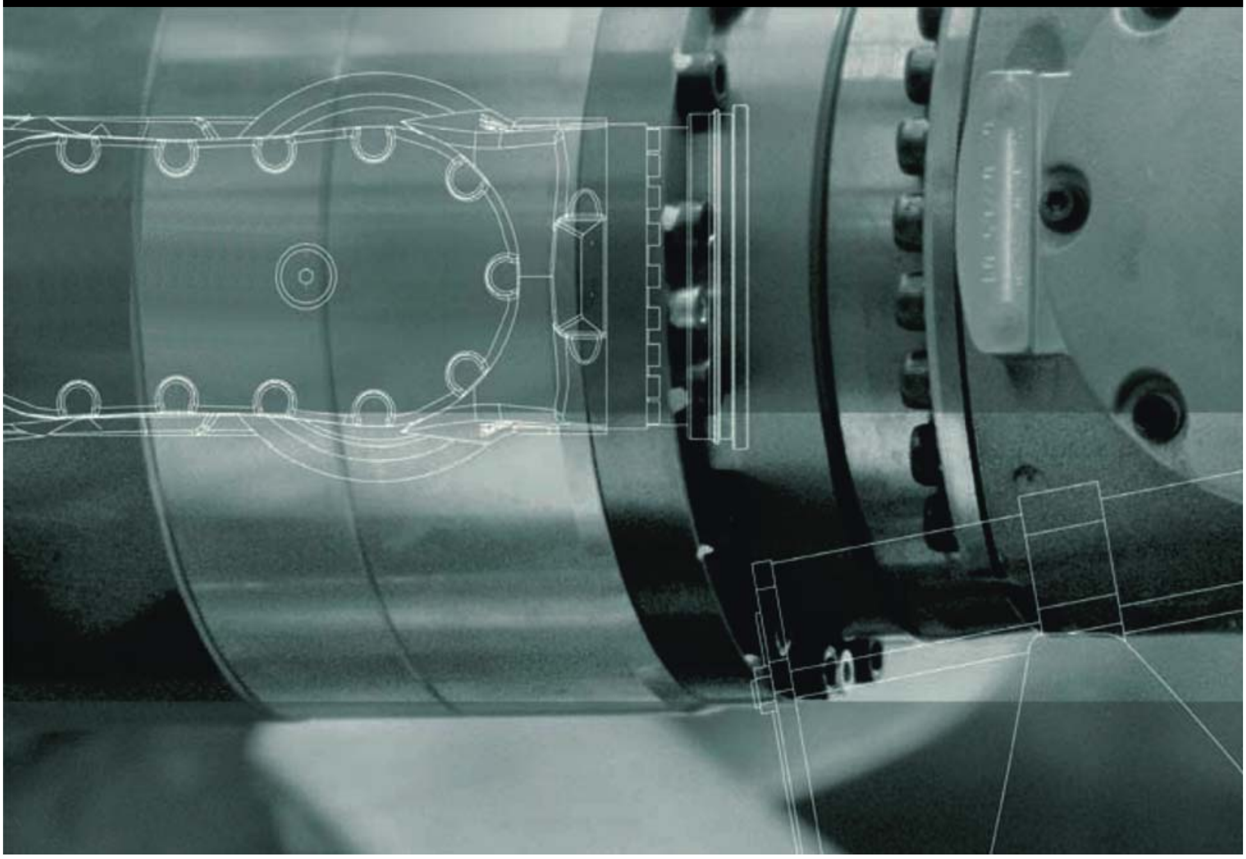
Expert Documentation

KUKA Roboter GmbH

External Axes

For KUKA System Software 5.5



Issued: 09.12.2008

Version: KSS 5.5 Zusatzachsen V1 en

KUKA Roboter GmbH
Zugspitzstraße 140
D-86165 Augsburg
Germany

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, however, and necessary corrections will be incorporated in the subsequent edition.

Subject to technical alterations without an effect on the function.

Translation of the original operating instructions

KIM-PS5-DOC

Publication: Pub KSS 5.5 Zusatzachsen en
Book structure: KSS 5.5 Zusatzachsen V0.50
Label: KSS 5.5 Zusatzachsen V1

Contents

1 Introduction 7

1.1 Target group 7
1.2 Robot system documentation 7
1.3 Terms used 7

2 Fundamentals 9

2.1 Distinction between external axis and kinematic system 9
2.2 Kinematic system types 9
2.3 Types of motion 11
2.4 Master/slave operation 13

3 Hardware 15

3.1 Maximum values for external axis systems 15
3.2 Top-mounted cabinet for external axes 16
3.3 Single Brake Module (SMB2) 17
3.3.1 Machine data 18

4 Safety 19

4.1 General 19
4.1.1 Liability 19
4.1.2 Representation of warnings and notes 19
4.1.3 Designated use of the robot system 20
4.1.4 EC declaration of conformity and declaration of incorporation 20
4.1.5 Description of the robot system 20
4.1.6 Terms used 21
4.2 Personnel 22
4.3 Safety features of the robot system 23
4.3.1 Overview of the safety features 23
4.3.2 ESC safety logic 24
4.3.3 Mode selector switch 24
4.3.4 Stop reactions 26
4.3.5 Workspace, safety zone and danger zone 27
4.3.6 Operator safety 27
4.3.7 EMERGENCY STOP button 28
4.3.8 Enabling switches 28
4.3.9 Connection for external enabling switch 29
4.3.10 Jog mode 29
4.3.11 Mechanical end stops 29
4.3.12 Software limit switches 30
4.3.13 Overview of operating modes and active safety features 30
4.3.14 Mechanical axis range limitation (option) 30
4.3.15 Axis range monitoring (option) 30
4.3.16 Release device (option) 31
4.3.17 KCP coupler (optional) 31
4.3.18 External safeguards 31
4.3.19 Labeling on the robot system 32
4.4 Safety measures 33

4.4.1 General safety measures 33
4.4.2 Transportation 34
4.4.3 Start-up 34
4.4.4 Virus protection and network security 35
4.4.5 Programming 35
4.4.6 Simulation 35
4.4.7 Automatic mode 35
4.4.8 Maintenance and repair 36
4.4.9 Decommissioning, storage and disposal 37
4.5 Applied norms and regulations 37

5 Operation 39

5.1 Coordinate systems 39
5.1.1 Kinematic chain with BASE kinematic system 39
5.1.2 Kinematic chain of a ROBROOT kinematic system 40
5.2 Jogging external axes 41
5.3 Activating mathematical coupling for jogging 43
5.3.1 Manually activating a mathematical coupling 43
5.3.2 Activating a mathematical coupling via a program 43

6 Start-up 45

6.1 Mastering 45
6.1.1 Mastering overview 45
6.1.2 Mastering methods 46
6.1.3 Moving axes to the pre-mastering position 47
6.1.4 Mastering with the EMT 48
6.1.5 Mastering with the dial gauge 48
6.2 Calibration 49
6.2.1 Calibration overview 49
6.2.2 Calibrating the root point 50
6.2.3 Assigning a TOOL coordinate system reference point 51
6.2.4 Moving to the reference point in 4 different positions 51
6.2.5 Entering the root point numerically 52
6.2.6 Offset base calibration 52
6.2.7 Entering the offset base numerically 54
6.2.8 Calibrating an external tool 54
6.2.9 Entering the external tool numerically 56

7 Configuration 57

7.1 Configuring external axis systems 57
7.1.1 Starting up a KUKA linear unit 57
7.1.2 Starting up a KUKA kinematic system 57
7.1.3 Starting up a kinematic system with KUKA MGU 58
7.1.4 Starting up a kinematic system with KUKA motor 58
7.2 Machine data for external axes 59
7.3 Transformation 61
7.3.1 Machine data for configuring the transformation 61
7.3.1.1 $EX_KIN 61
7.3.1.2 $ET1_AX 61

7.3.1.3 $ET1_NAME 62
7.3.1.4 $ET1_TA1KR 62
7.3.1.5 $ET1_TA2A1 63
7.3.1.6 $ET1_TA3A2 63
7.3.1.7 $ET1_TFLA3 63
7.3.2 Transformation of BASE kinematic system 64
7.3.3 Transformation of ROBROOT kinematic system 65
7.4 Optimizing machine data with the oscilloscope 65
7.4.1 Optimization sequence 66
7.4.2 Optimizing controller parameters 66
7.4.2.1 Optimizing $G_VEL_PTP and $G_VEL_CP 66
7.4.2.2 Optimizing $I_VEL_PTP and $I_VEL_CP 68
7.4.2.3 Optimizing $LG_PTP and $LG_CP 70
7.4.3 Optimizing acceleration parameters 72
7.4.3.1 Optimizing $RAISE_TIME 73
7.4.3.2 Optimizing $RED_ACC_EMX 74
7.4.3.3 Optimizing $DECEL_MB 75
7.4.3.4 Configuration examples 75
7.4.4 Checking the optimization results 77
7.4.5 Checking the r.m.s. current over a program cycle 78
7.5 Modifying machine data 78
7.5.1 Modifying individual machine data 79
7.5.2 Loading machine data via a text file 79
7.5.3 Modifying controller parameters 79
7.6 Axis configurator for external axes 80
7.6.1 Configuring machine data for external axes 81
7.6.2 Archiving a configuration 81
7.6.3 Loading a configuration 82
7.6.4 Creating a servo file 82
7.6.5 Loading a servo file 82
7.6.6 “General Data” tab 82
7.6.7 “External Axes” tab 83
7.6.8 “Main Axes” tab 83
7.6.9 “Ext. Kinematics” tab 84
7.6.10 “Servo-File” tab 84

8 System variables 85

8.1 System variables for configuring external axes 85
8.2 Enabling 85
8.2.1 $EXT_AXIS 85
8.2.2 $ASYNC_OPT 85
8.3 Asynchronous, uncoordinated external axes 86
8.3.1 $ZUST_ASYNC 86
8.3.2 $ASYNC_AX... 86
8.4 Asynchronous, coordinated external axes (ASYPTP) 87
8.4.1 $ASYNC_T1_FAST 87
8.4.2 $ASYNC_MODE 88
8.5 Decoupled external axes 89
8.5.1 $ASYNC_EX_AX_DECOUPLE 89

8.6 Permanently asynchronous external axes 90
8.6.1 $EX_AX_ASYNC 90

9 Programming 93

9.1 Programming motions for external axes 93
9.2 Programming synchronous external axes 93
9.2.1 Programming a mathematically coupled motion 94
9.3 Programming asynchronous external axes 94
9.3.1 $ASYNC_AXIS 94
9.3.2 ASYPTP 95
9.3.3 $OV_ASYNC 96
9.3.4 $ASYNC_FLT 97
9.3.5 ASYSTOP 97
9.3.6 ASYCONT 97
9.3.7 ASYCANCEL 98
9.3.8 $ASYNC_STATE 98

10 Example 101

10.1 Transformation for DKP 400 101
10.2 Transformation for KL 1500-2 106

11 KUKA Service 109

11.1 Requesting support 109
11.2 KUKA Customer Support 109

Index 115

1 Introduction

1.1 Target group

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

- Advanced knowledge of the robot controller system
- Advanced KRL programming skills

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 Robot system documentation

The robot system documentation consists of the following parts:

- Operating instructions for the robot
- Operating instructions for the robot controller
- Operating and programming instructions for the KUKA System Software
- Documentation relating to options and accessories

Each of these sets of instructions is a separate document.

1.3 Terms used

Term	Description
DSE	Digital Servo Electronics
KMC KMC kinematic system	KUKA Motion Control KUKA Motion Control can be used to control customer-specific kinematic systems.
KSD	KUKA Servo Drive
KUKA.HMI	Human/Machine Interface KUKA.HMI is the KUKA user interface.
MFC3	Multi-function card
MGU	Motor/gear unit KUKA motor/gear combination for kinematic systems
RDC	Resolver Digital Converter
SBM2	Single Brake Module Individual braking control for asynchronous external axes

2 Fundamentals

2.1 Distinction between external axis and kinematic system

- The robot controller can control up to 6 external axes.
- A kinematic system can consist of up to 3 external axes.
- The robot controller can control up to 6 kinematic systems.

Areas of application

Kinematic systems are used if the robot has to work on a moving workpiece.

- Kinematic systems extend the workspace of the robot, e.g. linear units, Cartesian gantries.
- Kinematic systems improve the accessibility of the workpiece, e.g. two-axis positioner, positioner.

External axes are used if the robot is not working on the workpiece that has to be moved.

- External axis that turns the workpiece ready for a subsequent operation, e.g. loading device
- External axis as drive unit for tools, e.g. electric motor-driven welding gun (KUKA.ServoGun)

2.2 Kinematic system types

Overview

The following kinematic system types are implemented by KUKA:

- External ROBROOT kinematic system
- External BASE kinematic system
- External TOOL kinematic system

ROBROOT kinematic system

ROBROOT kinematic systems move the robot, e.g. the KUKA linear unit.

Fig. 2-1: Robot on linear unit

BASE kinematic system

BASE kinematic systems move the workpiece, e.g. two-axis positioner and positioner.

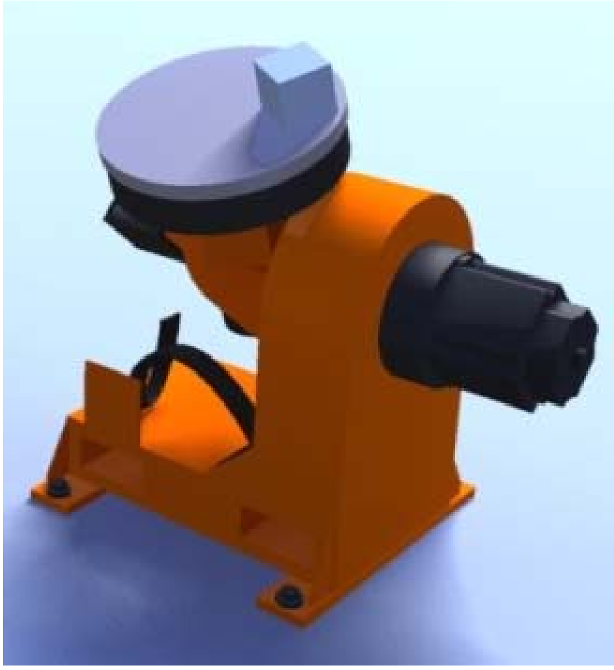


Fig. 2-2: Two-axis positioner

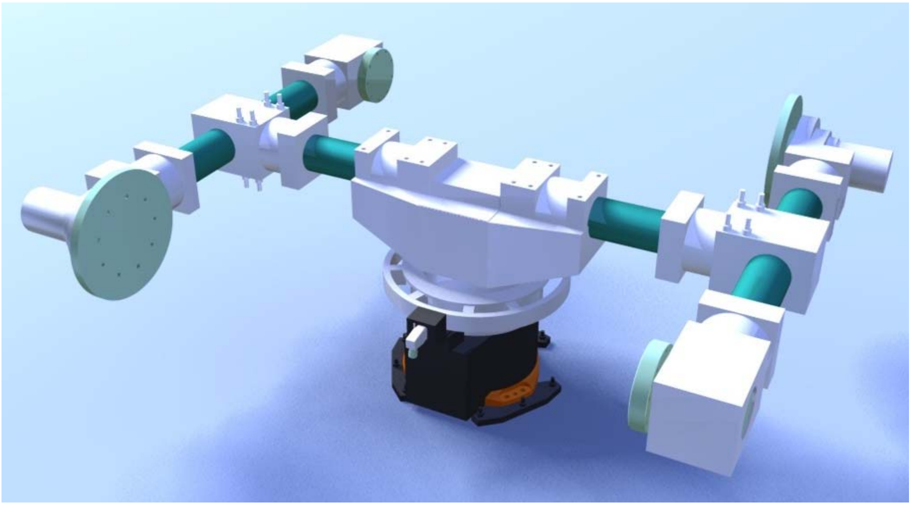


Fig. 2-3: Three-axis positioner

TOOL kinematic system

TOOL kinematic systems move the tool, e.g. the external adhesive nozzle for the application of adhesive to glass.

Fig. 2-4: External adhesive nozzle

2.3 Types of motion

Overview

External axes can be moved synchronously or asynchronously to the robot axes. For asynchronous motions, external axes must be switched to asynchronous mode. There are 3 options available for this:

- ASYPTP in the KRL program
- ASYPTP in the Submit interpreter
  (>>> 9.3.2 "ASYPTP" page 95)
- $EX_AX_ASYNC in the machine data
  (>>> 8.6.1 "$EX_AX_ASYNC" page 90)

Synchronous

In the case of a synchronous motion, all the axes involved (robot axes and external axes) execute a common motion, starting simultaneously and stopping simultaneously. The axis position of the external axes is contained in every taught point (E6POS).

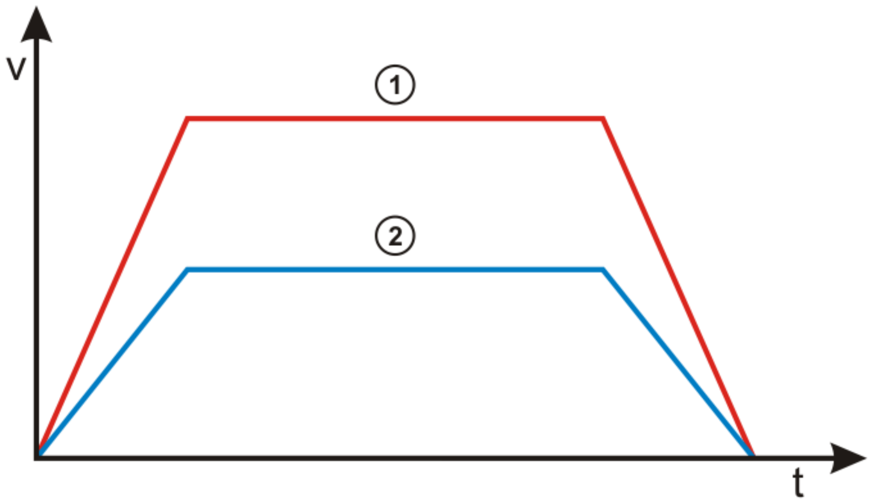


Fig. 2-5: Synchronous robot and external axis motion

1 Robot motion
2 Synchronous motion of an external axis

Synchronous motions	
Mathematically coupled	**Non-coupled**
The robot calculates its motion path in relation to the position of the kinematic system. The kinematic system must be calibrated.	The robot calculates its motion path without taking the position of the external axis into consideration. The external axis need not be calibrated.
Example: ■ Two-axis positioner, positioner ■ KUKA linear unit **Note:** A ROBROOT kinematic system is always mathematically coupled and is not calibrated.	Example: ■ Electric motor-driven welding gun in program mode ■ Turnover positioner

Asynchronous

In the case of an asynchronous motion, the external axes execute a motion that is not synchronized with the robot axes.

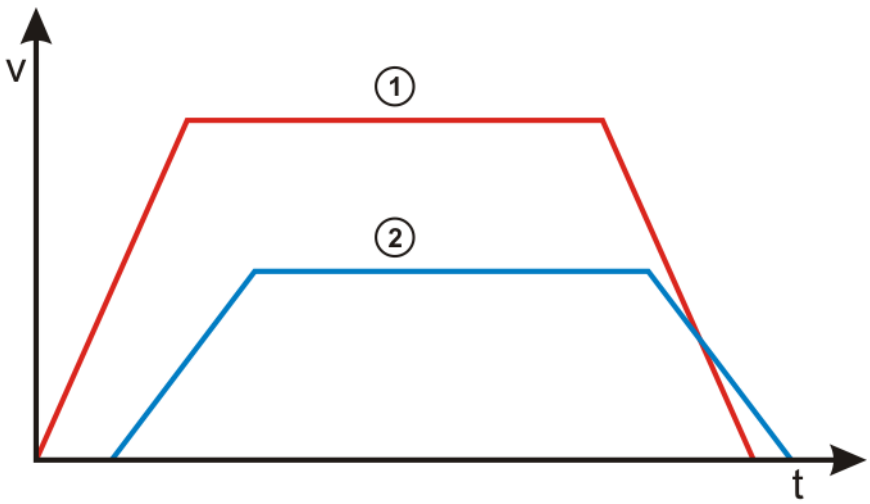


Fig. 2-6: Asynchronous robot and external axis motion

1 Robot motion
2 Asynchronous motion of an external axis

Asynchronous motions	
Coordinated	**Uncoordinated**
The asynchronous external axis is controlled via the KRL program “ASYPTP”.	The asynchronous external axis is controlled via a separate operating panel.
Example: ■ Loading device ■ Electric motor-driven weld gun: operated by means of status keys.	Example: ■ Manual loading area: the operator can move the external axis to a convenient position.

Asynchronous external axes cannot be moved asynchronously of one another. Simultaneous motions are possible, e.g.:
ASYPTP {E1 90, E2 20}
If 2 consecutive ASYPTP statements are programmed, the second motion does not start until the first motion has been completed, e.g.:
ASYPTP {E1 90}
ASYPTP {E2 20}

Example

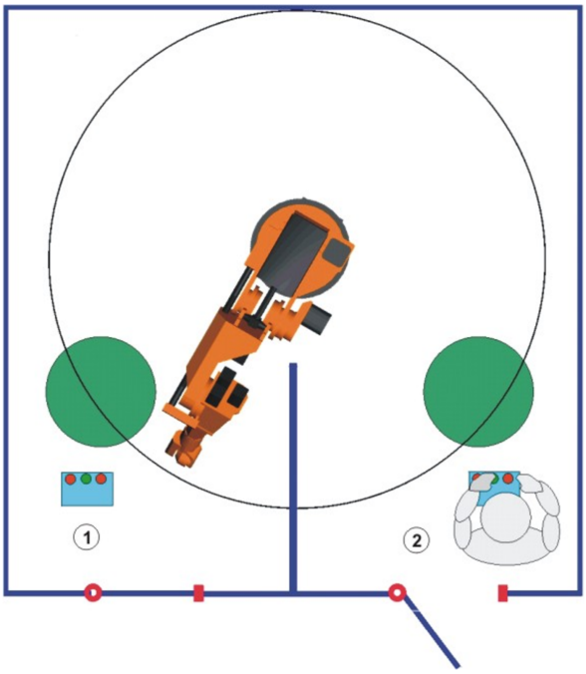


Fig. 2-7: Diagram: system with synchronous and asynchronous external axes

1	Mathematically coupled synchronous motion of robot and kinematic system
2	Uncoordinated asynchronous motion of the external axis

2.4 Master/slave operation

Description

In master/slave operation, external axes can be driven by up to 6 motors. 1 master and up to 5 slaves.

There are 2 types of drive control in master/slave operation:

- Position control
- Torque control

The master motor is always position-controlled. The slave motors are either position-controlled or torque-controlled. Which variant is selected for the slave depends on the required mechanical stiffness between master and slave.

Stiffness	Drive control of master/slave axis
High	Torque-controlled slave Stiff mechanical coupling between master and slave: slave mechanically follows master (torque is transmitted). Example: KUKA linear unit is driven by 2 motors for faster acceleration.

Stiffness	Drive control of master/slave axis
Medium	Selection must be tested.
Low	Position-controlled slave Low-stiffness mechanical coupling between master and slave: slave does not mechanically follow master (position is synchronized). Example: 2-column lifting device (>>> Fig. 2-8)

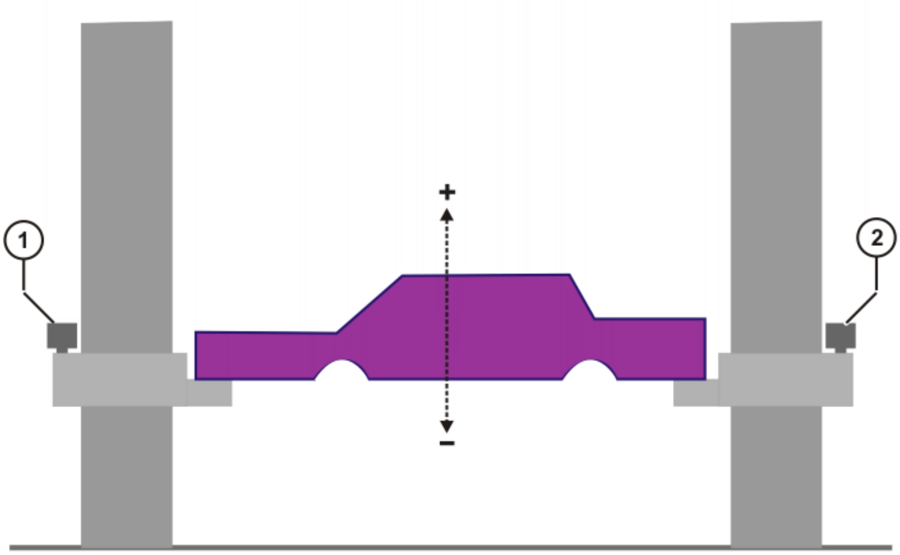


Fig. 2-8: Example: slave position-controlled

1 Master motor
2 Slave motor

3 Hardware

3.1 Maximum values for external axis systems

Robot controller

- The robot controller can control up to 12 axes.
- Up to 8 KSDs can be integrated into the robot controller.
- KSDs 1 … 6 of the robot controller are assigned to the robot.

Further information is contained in the robot controller operating instructions.

External axes

- In the case of a 6-axis robot, a maximum of 2 KSDs (KSD-08/16/32) or one double-sized KSD (KSD-48/64) for external axes can be integrated into the robot controller.
- A top-mounted cabinet is used for 3 or more external axes.
- The robot controller can control up to 6 kinematic systems.
- A kinematic system can consist of up to 3 external axes, i.e. a second robot cannot be operated as an external kinematic system.

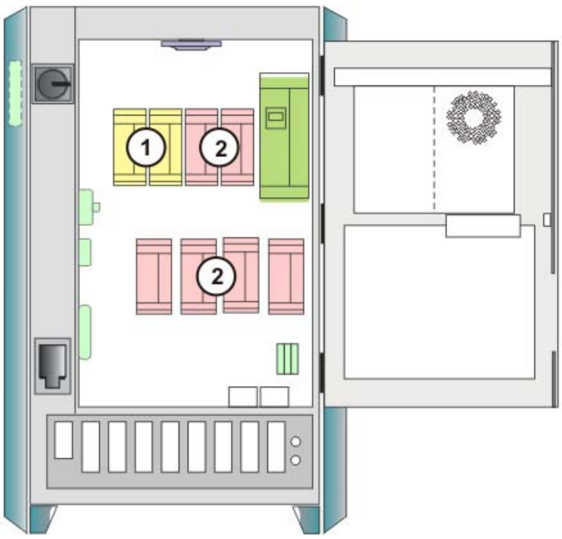


Fig. 3-1: KSDs in the robot controller

1 KSDs for 2 external axes
2 KSDs for 6 robot axes

Master/slave operation

- One axis can be driven by up to 6 motors. (1 master and 1 to 5 slaves)
- The robot controller can control up to 16 motors. (8 channels per DSE; 2 DSEs possible)
- A maximum of 8 master/slave axes can be configured.
- Every motor requires a KSD.
- Master and slave motors must be assigned to the same braking channel.
- Master/slave axes require a DSE-IBS-C33.
- The KSDs of master/slave axes must be connected to the DSE without a gap.
- The master motor and slave motors and their KSDs must all be of the same type.

3.2 Top-mounted cabinet for external axes

Description

- The robot controller can directly supply a maximum of 8 KSDs or 4 double-width KSDs.
- In the case of 3 or more external axes, a second RDC and a second KPS (KPS600) are required. These are located in the top-mounted cabinet.
- In the case of 3 or more external axes, a DSE-IBS-AUX is additionally required in the basic cabinet. The DSE-IBS-AUX (not of the same design as the DSE-IBS) is plugged onto the MFC3 in the basic cabinet.

  The DSE-IBS-AUX is then the second DSE in the system.
- All channels of the first RDC are assigned to the first DSE and all channels of the second RDC are assigned to the second DSE. The second RDC transfers the position signals of the external axes to the DSE-IBS-AUX.

> The system variable $DSECHANNEL, i.e. the assignment of the axes on the DSEs, must be checked when starting up an external axis system and adapted if necessary in the machine data.

- The top-mounted cabinet contains additional control units: a separate ESC system and a separate KPS with its own batteries.

  The batteries ensure the supply of power to the KPS and the ESC system in the event of a power failure.
- Depending on the mounting plate used, a maximum of 5 KSDs or 3 double-width KSDs can be installed in the top-mounted cabinet.
- If a top-mounted cabinet is used, all KSDs for external axes are installed there – if possible – and supplied by the same KPS.
- A single KSD must not be installed in the top-mounted cabinet. If, for example, previously used external axes are removed, leaving just one external axis in use, the KSD must be installed in the basic cabinet.

Overview

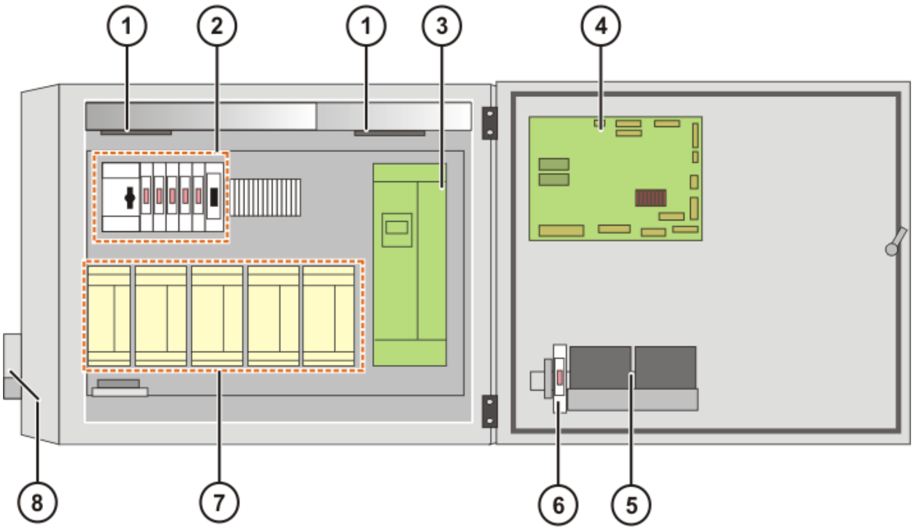


Fig. 3-2: Overview of top-mounted cabinet components

1	Fans	5	Batteries
2	Fuse elements	6	Battery blowout fuse
3	KPS600	7	KSDs (max. 5)
4	CI board	8	Connection panel

> Further information about the top-mounted cabinet with mounting plate can be found in the corresponding operating instructions.

3.3 Single Brake Module (SMB2)

Description

External axes must be switched to asynchronous mode to ensure a safe stop and protection against unexpected restarting. SBM2 is required for this.

Integration of the SBM2 into the KSDs allows the brakes of asynchronous external axes to be controlled independently of the robot and the synchronous external axes.

- All external axes supplied by the same KPS must be equipped and operated with SBM2, even if only one of the external axes actually requires an SBM2.
- If there is only one external axis in the system, no SBM2 is required. A single external axis can always be controlled with a second braking channel of the KPS, i.e. even if it must be moved asynchronously.
- If SBM2 is used, DSE-IBS-C33 and suitable KSDs must also be used.

If existing systems are converted for use with SBM2, KUKA Roboter GmbH must always be consulted in order to ascertain whether or not it is necessary to exchange KSDs.
(>>> 11 "KUKA Service" page 109)

Overview

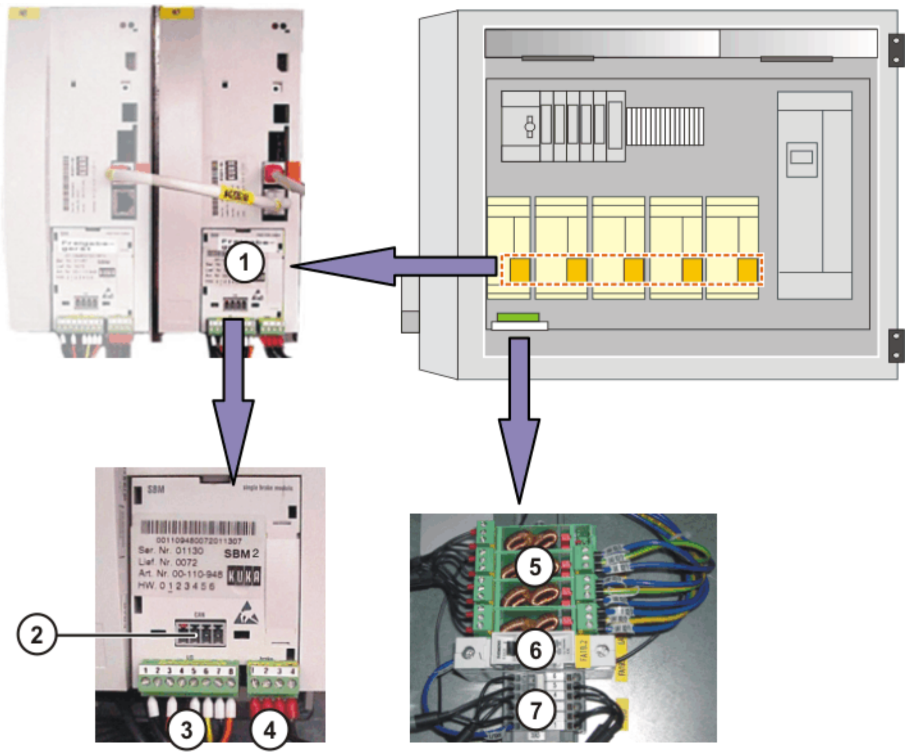


Fig. 3-3: Example: SBM2 components in the top-mounted cabinet

1	Brake module SBM2	5	Interference suppression modules LAx
2	CAN connector X4	6	Miniature circuit-breaker
3	I/O connector X2	7	Terminal block XA5
4	Brake connector X3		

Caution!
Interference suppression modules protect the SBM2 against voltage peaks in the brakes. If interference suppression modules are missing, this can result in malfunction and destruction of the SBM2.

Further information about SBM2 can be found in the operating instructions of the top-mounted cabinet with mounting plate and in the documentation **Single Brake Module** for safety-oriented applications or **Single Brake Module** for non-safety-oriented applications.

3.3.1 Machine data

$PMCHANNEL[ ]

If there is a “1” before the value of a $PMCHANNEL variable, this axis module has an SBM2.

In the case of a robot system with 2 asynchronous external axes and SBM2, for example, the following could apply:

- $PMCHANNEL[1] = 20
- $PMCHANNEL[2] = 20
- $PMCHANNEL[3] = 20
- $PMCHANNEL[4] = 20
- $PMCHANNEL[5] = 20
- $PMCHANNEL[6] = 20
- $PMCHANNEL[7] = 121
- $PMCHANNEL[8] = 121

$BRK_MODE

If SBM2s are used, the following applies:

- $BRK_MODE = 1101

4 Safety

4.1 General

4.1.1 Liability

The device described in these operating instructions is an industrial robot – called “robot system” in the following text – consisting of:

- Robot
- Connecting cables
- Robot controller
- Teach pendant
- Linear unit (optional)
- Positioner (optional)
- Two-axis positioner (optional)
- Top-mounted cabinet (optional)

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

The robot system may only be used in perfect technical condition in accordance with its designated use and only by safety-conscious persons who are fully aware of the risks involved in its operation. Use of the robot system is subject to compliance with these operating instructions and with the declaration of incorporation supplied together with the robot system. Any functional disorders affecting the safety of the system 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 robot system will not cause personal injuries or material damage.

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

4.1.2 Representation of warnings and notes

Safety

Warnings marked with this pictogram are relevant to safety and **must** be observed.

Danger!
This warning means that death, severe physical injury or substantial material damage **will** occur, if no precautions are taken.

Warning!
This warning means that death, severe physical injury or substantial material damage **may** occur, if no precautions are taken.

Caution!
This warning means that minor physical injuries or minor material damage **may** occur, if no precautions are taken.

Notes

Notes marked with this pictogram contain tips to make your work easier or references to further information.

> Tips to make your work easier or references to further information.

> **Specific safety instructions**
> In addition to the Safety chapter, the operating instructions for the robot system and its options contain further safety instructions. These must be observed.

4.1.3 Designated use of the robot system

The robot system is designed exclusively for the specified applications.

> Further information is contained in the technical data of the operating instructions for the robot system and its options.

Using the robot system or its options 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 robot system and its options within the limits of its designated use also involves continuous observance of the operating instructions with particular reference to the maintenance specifications.

Impermissible misuse

Any use or application deviating from the designated use is deemed to be impermissible misuse; examples of such misuse include:

- Transportation of persons and animals
- Use as a climbing aid
- Operation outside the permissible operating parameters
- Use in potentially explosive environments

4.1.4 EC declaration of conformity and declaration of incorporation

Declaration of conformity

The system integrator must issue a declaration of conformity for the overall system in accordance with the Machinery Directive. The declaration of conformity forms the basis for the CE mark for the system. The robot system 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

A declaration of incorporation is provided for the robot system. This declaration of incorporation contains the stipulation that the robot system must not be commissioned until it complies with the provisions of the Machinery Directive.

4.1.5 Description of the robot system

The robot system consists of the following components:

- Robot
- Robot controller
- KCP teach pendant
- Connecting cables

- External axes, e.g. linear unit, two-axis positioner, positioner (optional)
- Top-mounted cabinet (optional)
- Software
- Options, accessories

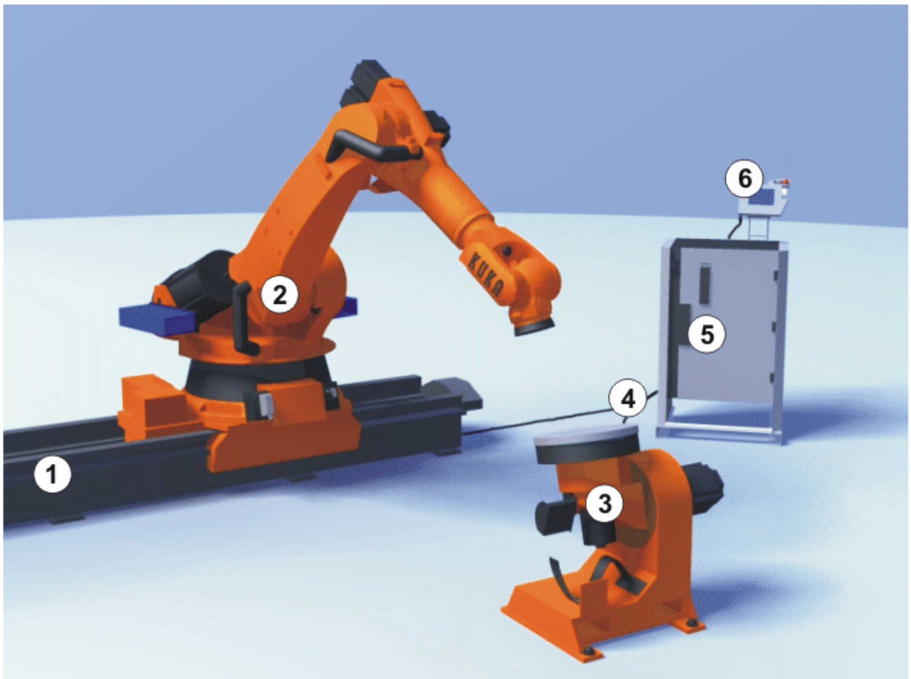


Fig. 4-1: Example of a robot system

1	Linear unit	4	Connecting cables
2	Robot	5	Robot controller
3	Positioner	6	Teach pendant

4.1.6 Terms used

Term	Description
Axis range	Range of an axis, in degrees, within which the robot may move The axis range must be defined for each axis that is to be monitored.
Working envelope	The robot is allowed to move within its workspace. The workspace is derived from the individual axis ranges.
Operator (User)	The user of the robot system can be the management, employer or delegated person responsible for use of the robot system.
Braking distance	The braking distance is the distance covered by the robot and any optional external axes after the stop function has been triggered and before the robot comes to a standstill. The braking distance is part of the danger zone.
Danger zone	The danger zone consists of the workspace and the braking distances.
KCP	The KCP (KUKA Control Panel) teach pendant has all the functions required for operating and programming the robot system.

Term	Description
Robot system	The robot system consists of the robot controller and robot, together with any options (e.g. KUKA linear unit, two-axis positioner, other positioner, top-mounted cabinet).
Safety zone	The safety zone is situated outside the danger zone.
STOP 0 (path-oriented braking)	In the case of a STOP 0, the drives are deactivated immediately and the brakes are applied. The robot and any external axes (optional) perform path-oriented braking.
STOP 1 (path-maintaining braking)	In the case of a STOP 1, the robot and any external axes (optional) perform path-maintaining braking. The drives are deactivated after 1 s and the brakes are applied.
STOP 2 (ramp-down braking)	In the case of a STOP 2, the drives are not deactivated and the brakes are not applied. The robot and any external axes (optional) are braked with a normal braking ramp.
System integrator (plant integrator)	System integrators are people who safely integrate the robot system into a plant and commission it.
T1	Test mode, Manual Reduced Velocity (<= 250 mm/s)
T2	Test mode, Manual High Velocity (> 250 mm/s)
External axis	Motion axis which is not part of the robot but which is controlled using the robot controller, e.g. KUKA linear unit, two-axis positioner, Posiflex

4.2 Personnel

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

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 repeated after particular incidents or technical modifications.

Personnel include the system integrator responsible for integrating the robot system into the production cell, the user, and the operator or programmer of the robot system.

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

User

The user of a robot system is responsible for its use. The user must ensure that it can be operated in complete safety and define all safety measures for personnel.

The user should check at specific intervals selected at his own discretion that the personnel attend to their work in a safety-conscious manner, are fully aware of the risks involved during operation and observe the operating instructions for the robot system.

System integrator

The robot system is safely integrated into a plant by the system integrator.

The system integrator is responsible for the following tasks:

- Installing the robot system
- Connecting the robot system
- Implementing the required facilities
- Issuing the declaration of conformity
- Attaching the CE mark

Operator

The operator must meet the following preconditions:

- The operator must have read and understood the robot system documentation, including the safety chapter.
- The operator must be trained for the work to be carried out.
- Work on the robot system 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 dangers.

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.

Example

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

Tasks	Operator	Programmer	System integrator
Switch robot controller on/off	x	x	x
Start program	x	x	x
Select program	x	x	x
Select operating mode	x	x	x
Calibration (tool, base)		x	x
Master robot		x	x
Configuration		x	x
Programming		x	x
Start-up			x
Maintenance			x
Repair			x
Shutting down			x
Transportation			x

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

4.3 Safety features of the robot system

4.3.1 Overview of the safety features

The following safety features are provided with the robot system:

- Operator safety
- EMERGENCY STOP pushbutton
- Enabling switch
- Mode selector switch
- Jog mode
- Mechanical limit stops
- Software limit switches
- Labeling on the robot system
- Mechanical axis range limitation (optional)
- Axis range monitoring (optional)
- Release device (optional)
- KCP coupler (optional)

The function and triggering of the electronic safety equipment are monitored by the ESC safety logic.

Danger!
In the absence of functional safety equipment, the robot system can cause personal injury or material damage. If safety equipment is dismantled or deactivated, the robot system may not be operated.

4.3.2 ESC safety logic

The ESC (Electronic Safety Circuit) safety logic is a dual-channel computer-aided safety system. It permanently monitors all connected safety-relevant components. In the event of a fault or interruption in the safety circuit, the power supply to the drives is shut off, thus bringing the robot system to a standstill.

Depending on the operating mode of the robot system, the ESC safety logic triggers a different stop reaction.

The ESC safety logic monitors the following inputs:

- Operator safety
- Local EMERGENCY STOP
- External EMERGENCY STOP
- Enabling
- Drives OFF
- Drives ON
- Operating modes
- Qualifying inputs

4.3.3 Mode selector switch

The robot system can be operated in the following modes:

- Manual Reduced Velocity (T1)
- Manual High Velocity (T2)
- Automatic (AUT)
- Automatic External (AUT EXT)

The operating mode is selected using the mode selector switch on the KCP. The switch is activated by means of a key which can be removed. If the key is removed, the switch is locked and the operating mode can no longer be changed.

If the operating mode is changed during operation, the drives are immediately switched off. The robot and any external axes (optional) are stopped with a STOP 0.

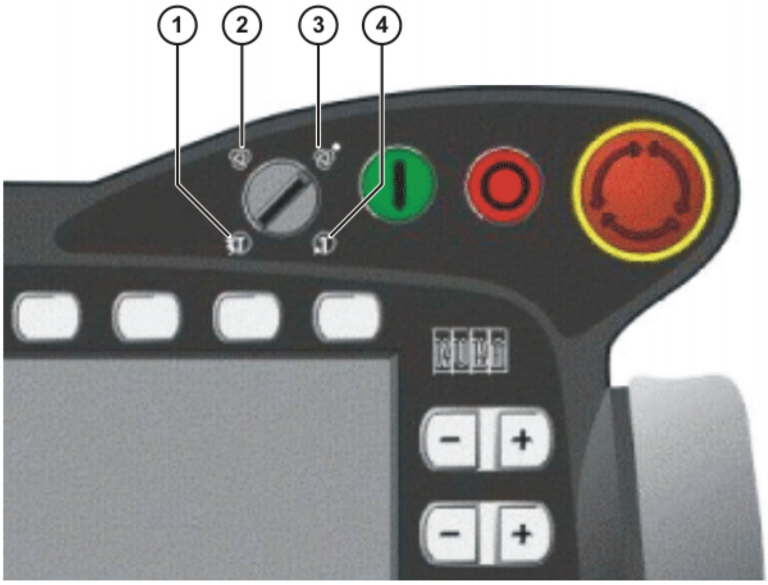


Fig. 4-2: Mode selector switch

1	T2 (Manual High Velocity)
2	AUT (Automatic)
3	AUT EXT (Automatic External)
4	T1 (Manual Reduced Velocity)

Operating mode	Use	Velocities
T1	For test operation	■ Program mode: Programmed velocity, maximum 250 mm/s ■ Jog mode: Jog velocity, maximum 250 mm/s
T2	For test operation	■ Program mode: Programmed velocity ■ Jog mode: Jog velocity, maximum 250 mm/s
AUT	For robot systems without higher-level controllers Only possible with a connected safety circuit	■ Program mode: Programmed velocity ■ Jog mode: not possible
AUT EXT	For robot systems with higher-level controllers, e.g. PLC Only possible with a connected safety circuit	■ Program mode: Programmed velocity ■ Jog mode: not possible

4.3.4 Stop reactions

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

STOP 0, STOP 1 and STOP 2 are the stop definitions according to EN 60204.

Trigger	T1, T2	AUT, AUT EXT
Safety gate opened	-	Path-maintaining braking (STOP 1)
EMERGENCY STOP pressed	Path-oriented braking (STOP 0)	Path-maintaining braking (STOP 1)
Enabling switch released	Path-oriented braking (STOP 0)	-
Start key released	Ramp-down braking (STOP 2)	-
"Drives OFF" key pressed	Path-oriented braking (STOP 0)	
STOP key pressed	Ramp-down braking (STOP 2)	
Operating mode changed	Path-oriented braking (STOP 0)	
Encoder error (DSE-RDC connection broken)	Short-circuit braking (STOP 0)	
Motion enable canceled	Ramp-down braking (STOP 2)	
Robot controller switched off Power failure	Short-circuit braking (STOP 0)	

Stop reaction	Drives	Brakes	Software	Path
Ramp-down braking (STOP 2)	Drives remain on.	Brakes remain open.	Normal ramp which is used for acceleration and deceleration.	The path is maintained exactly.
Path-maintaining braking (STOP 1)	Drives are switched off after 1 second hardware delay.	Brakes are applied after 1 s at latest.	In this time the controller brakes the robot on the path using a steeper stop ramp.	The path is maintained exactly.
Path-oriented braking (STOP 0)	Drives are switched off immediately.	Brakes are applied immediately.	The controller attempts to brake the robot on the path with the remaining energy. If the voltage is not sufficient, the robot leaves the programmed path.	The path is maintained approximately.
Short-circuit braking (STOP 0)	Drives are switched off immediately.	Brakes are applied immediately.	Stop initiated by the drive hardware. Energy present in the intermediate circuit is used for braking.	The path is left.

4.3.5 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. If a safeguard is triggered, the robot and external axes are braked and come to a stop within the workspace or the braking range.

The danger zone consists of the workspace and the braking distances of the robot and external axes (optional). It must be safeguarded by means of protective barriers to prevent danger to persons or the risk of material damage.

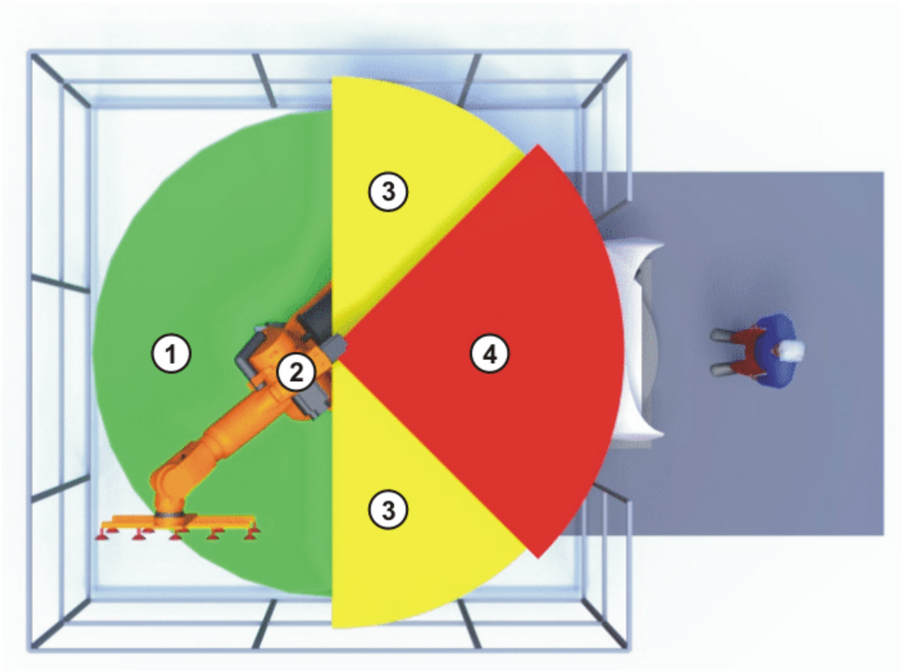


Fig. 4-3: Example of axis range A1

1	Workspace	3	Braking distance
2	Robot	4	Safety zone

4.3.6 Operator safety

The operator safety input is used for interlocking fixed guards. Safety equipment, such as safety gates, can be connected to the dual-channel input. If nothing is connected to this input, operation in Automatic mode is not possible. Operator safety is not active in the test modes T1 (Manual Reduced Velocity) and T2 (Manual High Velocity).

In the event of a loss of signal during Automatic operation (e.g. safety gate is opened), the drives are deactivated after 1 s and the robot and any external axes (optional) are stopped with a STOP 1. When the signal is applied again at the input (e.g. safety gate closed), Automatic operation can be resumed once the corresponding message has been acknowledged.

> The operator safety must be designed in such a way that it is only possible to acknowledge the message from outside.

Operator safety can be connected via the peripheral interface on the robot controller.

4.3.7 EMERGENCY STOP button

The EMERGENCY STOP button for the robot system is located on the KCP. If the EMERGENCY STOP button is pressed in the operating modes T1 (Manual Reduced Velocity) or T2 (Manual High Velocity), the drives are disconnected immediately. The robot and any external axes (optional) are stopped with a STOP 0.

In the Automatic operating modes, the drives are disconnected after 1 s. The robot and any external axes (optional) are stopped with a STOP 1. The EMERGENCY STOP button must be pressed as soon as persons or equipment are endangered. Before operation can be resumed, the EMERGENCY STOP button must be turned to release it and the stop message must be acknowledged.

Fig. 4-4: EMERGENCY STOP button on the KCP

1 EMERGENCY STOP button

4.3.8 Enabling switches

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 T1 (Manual Reduced Velocity) and T2 (Manual High Velocity), the robot can only be moved if one of the enabling switches is held in the central position. If the enabling switch is released or pressed fully down (panic position), the drives are deactivated immediately and the robot stops with a STOP 0.

Fig. 4-5: Enabling switches on the KCP

1 - 3 Enabling switches

4.3.9 Connection for external enabling switch

An external enabling switch is required if there is more than one person in the danger zone of the robot system.

The external enabling switch can be connected via the peripheral interface on the robot controller.

An external enabling switch is not included in the scope of supply of KUKA Roboter GmbH.

4.3.10 Jog mode

In the operating modes T1 (Manual Reduced Velocity) and T2 (Manual High Velocity), the robot 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. If the enabling switch is released or pressed fully down (panic position), the drives are deactivated immediately and the robot and any external axes (optional) stop with a STOP 0. Releasing the Start key causes the robot system to be stopped with a STOP 2.

4.3.11 Mechanical end stops

The axis ranges of main axes A 1 to A 3 and wrist axis A 5 of the robot are limited by means of mechanical limit stops with a buffer.

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

Danger!
If the robot 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 robot system. KUKA Roboter GmbH must be consulted before the robot system is put back into operation (>>> 11 "KUKA Service" page 109). The affected buffer must immediately be replaced with a new one. If a robot (or external axis) collides with a buffer at more than 250 mm/s, the robot (or external axis) must be exchanged or recommissioning must be carried out by the KUKA Roboter GmbH.

4.3.12 Software limit switches

The axis ranges of all robot 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 robot/positioner cannot hit the mechanical limit stops.

The software limit switches are set during commissioning of a robot system.

Further information is contained in the operating and programming instructions.

4.3.13 Overview of operating modes and active safety features

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

Safety features	T1	T2	AUT	AUT EXT
Operator safety	-	-	active	active
EMERGENCY STOP button	active (STOP 0)	active (STOP 0)	active (STOP 1)	active (STOP 1)
Enabling switch	active	active	-	-
Reduced velocity in program mode	active	-	-	-
Jog mode	active	active	-	-
Software limit switches	active	active	active	active

4.3.14 Mechanical axis range limitation (option)

Most robots can be fitted with mechanical axis range limitation in main axes A 1 to A 3. Additional working range limitation can be installed on the positioner axes. The adjustable axis range limitation systems restrict the working range to the required minimum. This increases personal safety and protection of the system.

This option can be retrofitted.

4.3.15 Axis range monitoring (option)

Most robots can be fitted with dual-channel axis range monitoring systems in main axes A 1 to A 3. 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 can be retrofitted.

4.3.16 Release device (option)

Description

The release device can be used to move the robot mechanically 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). After use of the release device, the affected motors must be exchanged.

Caution!
The motors reach temperatures during operation which can cause burns to the skin. Appropriate safety precautions must be taken.

Procedure

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.

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.

Further information is contained in the robot operating instructions.

4.3.17 KCP coupler (optional)

The KCP coupler allows the KCP to be connected and disconnected with the robot controller running.

Warning!
If the KCP is disconnected, the system can no longer be deactivated by means of the EMERGENCY STOP button on the KCP. An external EMERGENCY STOP must be connected to the peripheral interface to prevent personal injury and material damage.

Further information is contained in the robot controller operating instructions.

4.3.18 External safeguards

EMERGENCY STOP

Additional EMERGENCY STOP devices can be connected via the peripheral interface on the robot controller or linked together by means of higher-level controllers (e.g. PLC).

The input/output signals and any necessary external power supplies must ensure a safe state in the case of an EMERGENCY STOP.

Safety fences

Requirements on safety fences are:

- Safety fences must withstand all forces that are likely to occur in the course of operation, whether from inside or outside the enclosure.
- Safety fences must not, themselves, constitute a hazard.
- It is imperative to comply with the minimum clearances from the danger zone.

Further information is contained in the corresponding standards and regulations.

Safety gates

Requirements on safety gates are:

- The number of safety gates in the fencing must be kept to a minimum.
- All safety gates must be safeguarded by means of an operator safety system.
- Automatic mode must be prevented until all safety gates are closed.
- For additional protection in Automatic mode, the safety gate can be mechanically locked by means of a safety system.
- If a safety gate is opened in Automatic mode, it must trigger an EMERGENCY STOP function.
- If the safety gate is closed, the robot cannot be started immediately in Automatic mode. The message on the control panel must be acknowledged.

Further information is contained in the corresponding standards and regulations.

Other safety equipment

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

4.3.19 Labeling on the robot system

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

Labeling on the robot system consists of:

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

Further information can be found in the operating instructions of the robot, linear unit, positioner and robot controller.

4.4 Safety measures

4.4.1 General safety measures

The robot system may only be used in perfect technical condition in accordance with its designated 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 robot system 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 robot or external axes to sag. If work is to be carried out on a switched-off robot system, the robot 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 robot and external axes must be secured by appropriate means.

Danger!
In the absence of functional safety equipment, the robot system can cause personal injury or material damage. If safety equipment is dismantled or deactivated, the robot system may not be operated.

Warning!
The motors reach temperatures during operation which can cause burns to the skin. Contact should be avoided if at all possible. If necessary, appropriate protective equipment must be used.

KCP

If the KCP is not connected, it must be removed from the system, as the EMERGENCY STOP button on the KCP is not functional in such a case.

If there is more than one KCP in operation in the overall system, it must be ensured that the KCPs and EMERGENCY STOP buttons can be unambiguously assigned to the corresponding robot system. There must be no possibility of mixing them up in an emergency situation.

External keyboard, external mouse

An external keyboard and/or external mouse may only be connected during service work (e.g. installation). If a keyboard and/or mouse is connected, the system can no longer be operated safely. If a keyboard and/or mouse is connected, the system must not be operated and there must be no persons within the system.

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 service work is completed.

Faults

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

- 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.

4.4.2 Transportation

Robot

The prescribed transport position of the robot must be observed. Transportation must be carried out in accordance with the robot operating instructions.

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 for the robot controller.

External axis (optional)

The prescribed transport position of the external axis (e.g. KUKA linear unit, two-axis positioner, etc.) must be observed. Transportation must be carried out in accordance with the operating instructions for the external axis.

4.4.3 Start-up

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 robot system. If cables are interchanged, the robot 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 robot, always connect the connecting cables to the robots and their corresponding robot controllers.

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

It must be ensured that no persons or objects are present within the danger zone of the robot during the function test.

The following must be checked during the function test:

- The robot system is installed and connected. There are no foreign bodies or destroyed, loose parts on the robot system.
- All safety devices and protective measures are complete and fully functional.
- All electrical connections are correct.
- The peripheral devices are correctly connected.
- The external environment corresponds to the permissible values indicated in the operating instructions.

Setting

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 robot and the external axes (optional) must be entered during start-up.

Caution!
Incorrect machine data can result in material damage. Check that the correct machine data have been loaded; if not, load the correct machine data.

4.4.4 Virus protection and network security

The user of the robot system is responsible for ensuring that the software is always safeguarded with the latest virus protection. If the robot controller is integrated into a network that is connected to the company network or to the Internet, it is advisable to protect this robot network against external risks by means of a firewall.

For optimal use of our products, we recommend that our customers carry out a regular virus scan. Information about security updates can be found at www.kuka.com.

4.4.5 Programming

The following safety measures must be carried out during programming:

- It must be ensured that no persons are present within the danger zone of the robot system during programming.
- New or modified programs must always be tested first in Manual Reduced Velocity mode (T1).
- If the drives are not required, they must be switched off to prevent the robot or the external axes (optional) from being moved unintentionally.
- The robot, tooling or external axes (optional) must never touch or project beyond the safety fence.
- Components, tooling and other objects must not become jammed due to the motion of the robot system, nor must they lead to short-circuits or be liable to fall off.

The following safety measures must be carried out during programming in the danger zone of the robot system:

- The robot and the external axes (optional) must only be moved at Manual Reduced Velocity (max. 250 mm/s). In this way, persons have enough time to move out of the way of hazardous motions of the robot system or to stop the robot system.
- To prevent other persons from being able to move the robot or external axes (optional), the KCP must be kept within reach of the programmer.
- If two or more persons are working in the system at the same time, they must all use an enabling switch. While the robot or external axes (optional) are being moved, all persons must remain in constant visual contact and have an unrestricted view of the robot system.

4.4.6 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 (SSTEP T1). It may be necessary to modify the program.

4.4.7 Automatic mode

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

- The prescribed safety equipment is present and operational.
- There are no persons in the system.
- The defined working procedures are adhered to.

If the robot or an external axis (optional) comes to a standstill for no apparent reason, the danger zone must not be entered until the EMERGENCY STOP function has been triggered.

4.4.8 Maintenance and repair

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 robot system:

- 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 robot controller and secure it (e.g. with a padlock) to prevent unauthorized persons from switching it 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 equipment is deactivated during maintenance or repair work, it must be reactivated immediately after the work is completed.

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 or isolated 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 the KPS (KUKA Power Supply), the KSDs (KUKA Servo Drives) and the intermediate-circuit connecting cables several minutes after the robot controller has been switched off. To prevent life-threatening injuries, no work may be carried out on the robot system in this time.

Foreign matter, such as swarf, water and dust, must be prevented from entering the robot controller.

Counterbalancing system

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

Hazardous substances

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.

4.4.9 Decommissioning, storage and disposal

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

4.5 Applied norms and regulations

Name	Definition	Edition
2006/95/EC	Low Voltage Directive: Directive of the European Parliament and the Council of 12 December 2006 on the harmonization of the laws of Member States relating to electrical equipment designed for use within certain voltage limits	2006
89/336/EEC	EMC Directive: Council Directive of 3 May 1989 on the approximation of the laws of the Member States relating to electromagnetic compatibility	1993
97/23/EC	Pressure Equipment Directive: Directive of the European Parliament and of the Council of 29 May 1997 on the approximation of the laws of the Member States concerning pressure equipment	1997
98/37/EC	Machinery Directive: Directive of the European Parliament and of the Council of 22 June 1998 on the approximation of the laws of the Member States relating to machinery	1998
EN ISO 13850	Safety of machinery: E-STOP - Principles for design	2007

Name	Definition	Edition
EN ISO 13732-1	Ergonomics of thermal environment: Methods for the assessment of human responses to contact with surfaces - Part 1: Hot surfaces	2006
EN 614-1	Safety of machinery: Ergonomic design principles – Part 1: Terms and general principles	2006
EN 954-1	Safety of machinery: Safety-related parts of control systems - Part 1: General principles for design	1997
EN 55011	Industrial, scientific and medical (ISM) radio-frequency equipment – Radio disturbance characteristics – Limits and methods of measurement	2003
EN 60204-1	Safety of machinery: Electrical equipment of machines - Part 1: General requirements	2007
EN 61000-4-4	Electromagnetic compatibility (EMC): Part 4-4: Testing and measurement techniques; Electrical fast transient/burst immunity test	2005
EN 61000-6-2	Electromagnetic compatibility (EMC): Part 6-2: Generic standards; Immunity for industrial environments	2002
EN 61000-6-4	Electromagnetic compatibility (EMC): Part 6-4: Generic standards; Emission standard for industrial environments	2002
EN 61800-3	Adjustable speed electrical power drive systems: Part 3: EMC product standard including specific test methods	2001
EN ISO 10218-1	Industrial robots: Safety	2006
EN ISO 12100-1	Safety of machinery: Basic concepts, general principles for design - Part 1: Basic terminology, methodology	2004
EN ISO 12100-2	Safety of machinery: Basic concepts, general principles for design - Part 2: Technical principles	2004

5 Operation

5.1 Coordinate systems

5.1.1 Kinematic chain with BASE kinematic system

Overview

The following Cartesian coordinate systems are defined in a robot system with mathematically coupled BASE kinematic system:

- WORLD
- ROBROOT
- ROOT
- FLANGE
- OFFSET
- TOOL

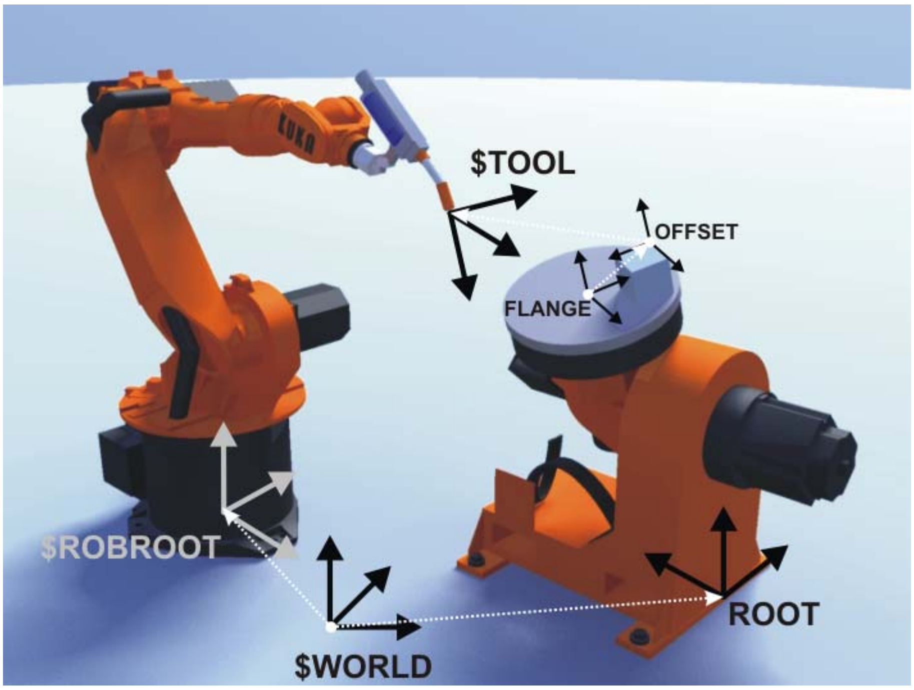


Fig. 5-1: Kinematic chain with DKP 400

Description

WORLD

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

ROOT

The ROOT coordinate system is a Cartesian coordinate system which is located in the root point of the BASE kinematic system. It defines the position of the kinematic system relative to the WORLD coordinate system.

(>>> 6.2.2 "Calibrating the root point" page 50)

FLANGE

The FLANGE coordinate system is a Cartesian coordinate system which is located at the flange center point of the BASE kinematic system.

OFFSET

The OFFSET coordinate system is a Cartesian coordinate system that defines the position of the workpiece on the BASE kinematic system. It is relative to the FLANGE coordinate system.

The OFFSET coordinate system is mobile relative to the WORLD coordinate system, i.e. the position of the workpiece on the kinematic system changes with the motion of the kinematic system.

(>>> 6.2.6 "Offset base calibration" page 52)

> The current position of the OFFSET can be displayed by means of the system variable $BASE_C.

TOOL

The TOOL coordinate system is a Cartesian coordinate system which is located at the tool center point. It is relative to the OFFSET coordinate system.

5.1.2 Kinematic chain of a ROBROOT kinematic system

> A ROBROOT kinematic system is always mathematically coupled. The mathematical coupling cannot be deactivated.

Overview

The following Cartesian coordinate systems are defined for a ROBROOT kinematic system:

- WORLD
- ROBROOT
- ERSYSROOT

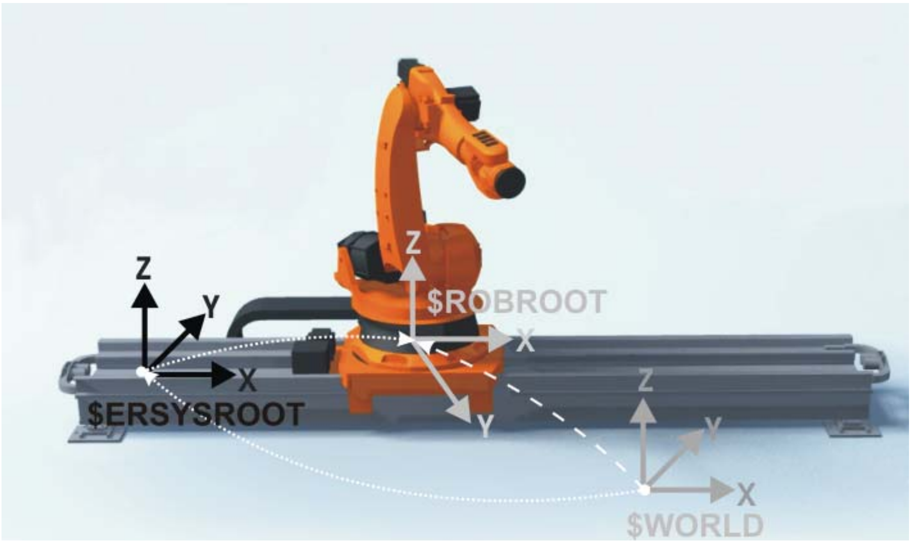


Fig. 5-2: ROBROOT kinematic system – linear unit

Description

WORLD

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

By default, the WORLD coordinate system is located at the root point of the linear unit.

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.

Every time the ROBROOT kinematic system is moved, the position in space of the robot changes. The reference point for calculation of the current position of the robot is thus $ERSYSROOT.

> The current offset between $WORLD and $ROBROOT can be displayed by means of the system variable $ROBROOT_C.

ERSYSROOT

The ERSYSROOT coordinate system is a Cartesian coordinate system which is located at the root point of the linear unit. The root point is situated by default at the zero position of the linear unit and is not dependent on $MAMES.

$ERSYSROOT allows the definition of an offset of the linear unit relative to the WORLD coordinate system.

5.2 Jogging external axes

Precondition

- Operating mode T1 or T2

Procedure

1. Select the jog mode "Jog keys" or "Space Mouse" in the left-hand status key bar:

    or 

2. In the right-hand status key bar, select axis-specific jogging or the coordinate system for Cartesian jogging.
3. Select the axis combination for jogging in the right-hand status key bar.

> The type and number of axis combinations available depend on the system configuration.

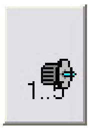
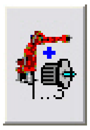
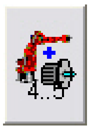
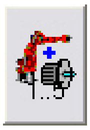
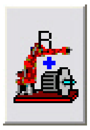
4. Set jog override.
5. Hold down the enabling switch.
6. The jog keys are displayed in the right-hand status key bar.

   Press the Plus or Minus status key to move an axis in the positive or negative direction.

> External axes cannot be moved using the Space Mouse. If “Space Mouse” mode has been selected, the robot can be moved using the Space Mouse and the external axes can be moved using the jog keys.

Description

Depending on the configuration, the following axis combinations for jogging are available in the right-hand status key bar:

Status key	Description
	All robot axes can be jogged using the jog keys or the Space Mouse. The external axes cannot be jogged.
	All configured external axes, e.g. external axes E1 to E5, can be moved using the jog keys.
  	Jog mode “Jog keys” ■ The 3 upper jog keys can be used to jog the robot: ▫ Axis-specific jogging: A1 to A3 ▫ Cartesian jogging: X, Y, Z ■ The 3 upper jog keys can be used to jog the external axes, e.g. external axes E1 to E3. Jog mode “Space Mouse” ■ The robot can be moved using the Space Mouse. ■ All configured external axes, e.g. external axes E1 to E5, can be moved using the jog keys.
	Jog mode “Jog keys” ■ The 3 upper jog keys can be used to jog the robot: ▫ Axis-specific jogging: A1 to A3 ▫ Cartesian jogging: X, Y, Z ■ The 3 lower jog keys can be used to jog the ROBROOT kinematic system, e.g. a linear unit. Jog mode “Space Mouse” ■ The robot can be moved using the Space Mouse. ■ The ROBROOT kinematic system can be moved using the jog keys.
 	Jog mode “Jog keys” ■ The 3 upper jog keys can be used to jog the robot: ▫ Axis-specific jogging: A1 to A3 ▫ Cartesian jogging: X, Y, Z ■ The 3 lower jog keys can be used to jog the axes of a mathematically coupled kinematic system, e.g. kinematic system A. Jog mode “Space Mouse” ■ The robot can be moved using the Space Mouse. ■ The jog keys can be used to jog the axes of a mathematically coupled kinematic system, e.g. kinematic system B. **Note**: A maximum of 6 kinematic systems (A to F) can be mathematically coupled.

5.3 Activating mathematical coupling for jogging

Description

In the case of mathematical coupling, the robot calculates its motion path in relation to the position of the kinematic system. If the kinematic system moves, the robot follows it with the TCP so that the position of the TCP remains constant relative to the moving base of the kinematic system.

The mathematical coupling must be activated for BASE kinematic systems. A ROBROOT kinematic system is automatically mathematically coupled. It cannot be deactivated.

> In the case of BASE kinematic systems, a mathematical coupling is only active in the TOOL or BASE coordinate system.
> In the case of ROBROOT kinematic systems, the mathematical coupling is also active in the WORLD coordinate system.

Overview

There are 2 ways of activating the mathematical coupling for Cartesian jogging:

- Manually via KUKA.HMI
  (>>> 5.3.1 "Manually activating a mathematical coupling" page 43)
- By starting a mathematically coupled motion in a program
  (>>> 5.3.2 "Activating a mathematical coupling via a program" page 43)

5.3.1 Manually activating a mathematical coupling

Description

One tool (TOOL coordinate system) and one offset base (BASE coordinate system of the kinematic system) must be selected for Cartesian jogging.

> The coordinates of an offset base are saved as BASE_DATA[17...22].

Precondition

- Root point of the kinematic system has been calibrated.

Procedure

1. Select the menu sequence **Configure** > **Cur. tool/base**.
2. In the softkey bar, select whether a fixed tool is to be used:
   - **ext. Tool**: The tool is a fixed tool.
   - **Tool**: The tool is mounted on the mounting flange.
3. Enter the number of the desired tool in the box **Tool no.**.
4. Enter the number of the desired offset base in the box **Base No.**.
5. Press **OK**.

5.3.2 Activating a mathematical coupling via a program

Precondition

- Program is selected.
- Operating mode T1 or T2
- Root point of the kinematic system has been calibrated.

Procedure

1. Perform block selection to a motion instruction with coupled BASE of the kinematic system.
2. Recommendation: set program override to 0%.
3. Start the program in order to load the data.

   The mathematical coupling is now active and can be used for the Cartesian jogging.

6 Start-up

6.1 Mastering

6.1.1 Mastering overview

Overview

The mastering of external axes is analogous to that of robot axes. During mastering, the external axes are moved to the mechanical zero position. In this mechanical zero position, the degree or millimeter value stored in $MAMES is accepted for the current axis position.

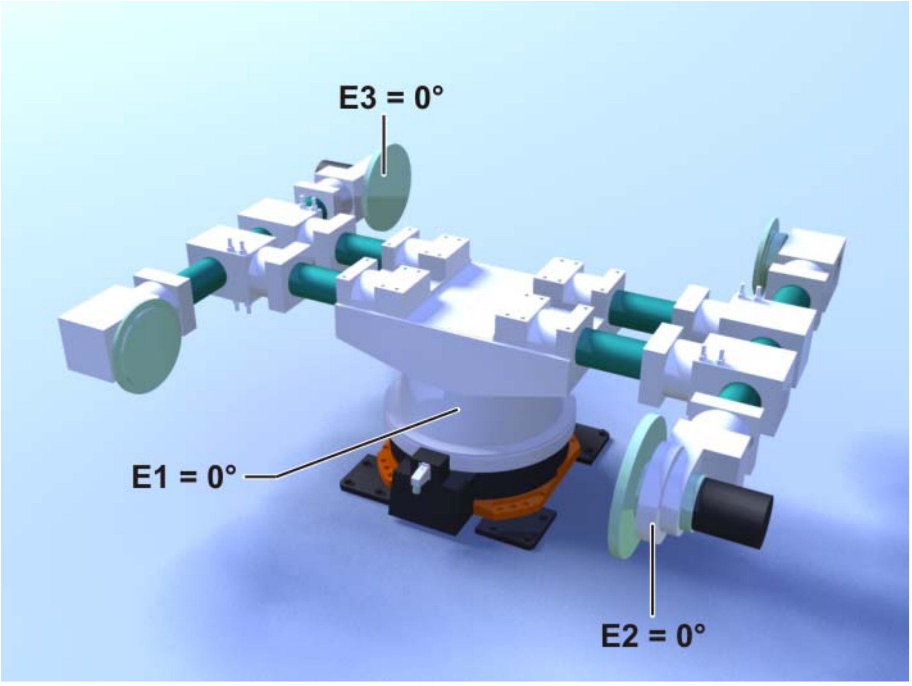


Fig. 6-1: Example of mechanical zero position

Only mastered external axes can move to programmed positions and be moved using Cartesian coordinates.

External axes must be mastered in the following cases:

- During start-up
- After repairs (e.g. after replacement of a motor or RDC)
- If the external axis has been moved without the robot controller (e.g. with the release device)
- After exchanging a gear unit
- After an impact with an end stop at more than 250 mm/s
- After a collision

Before carrying out a new mastering procedure, the old mastering data must first be deleted. Mastering data are deleted by manually unmastering the axes.

Further information is contained in the operating and programming instructions for the KUKA System Software (KSS).

6.1.2 Mastering methods

Overview

There are 2 ways of mastering an external axis or kinematic system:

- With the EMT (electronic mastering tool)
- With the dial gauge

EMT mastering is recommended.

Description of mastering with the EMT

In EMT mastering, the axis is automatically moved by the robot controller to the mechanical zero position. External axes must either always be mastered without a load or always with the same load. It is not possible to save mastering data for different loads.

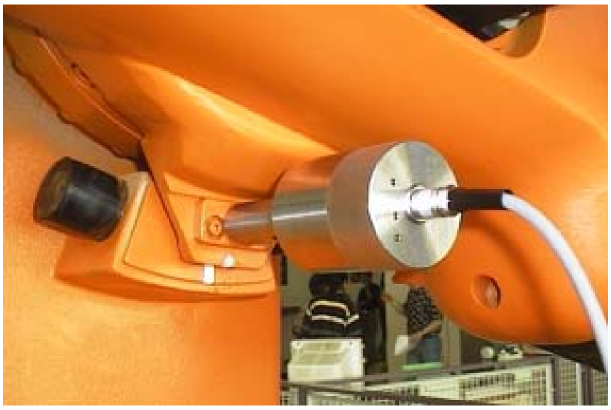


Fig. 6-2: EMT – electronic measuring tool

Description of mastering with the dial gauge

In dial mastering, the axis is moved manually by the user to the mechanical zero position. External axes must either always be mastered without a load or always with the same load. It is not possible to save mastering data for different loads.

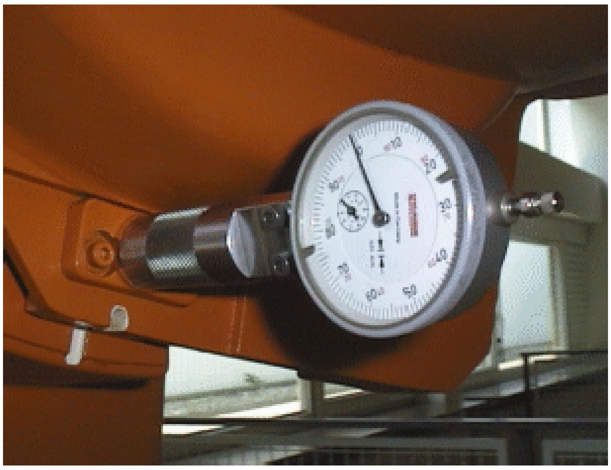


Fig. 6-3: Dial gauge

Further information is contained in the operating and programming instructions for the KUKA System Software (KSS).

6.1.3 Moving axes to the pre-mastering position

Description

The pre-mastering position is a prerequisite for every mastering carried out with the EMT. Each axis is moved so that the mastering marks are aligned or the gauge pin is positioned in front of the reference notch.

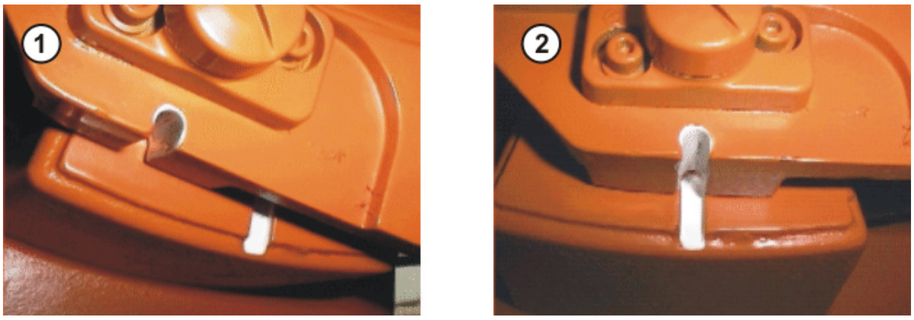


Fig. 6-4: Moving an axis to the pre-mastering position

When mastering with the EMT, an axis is always moved to the mechanical zero position from “+” to “-”.

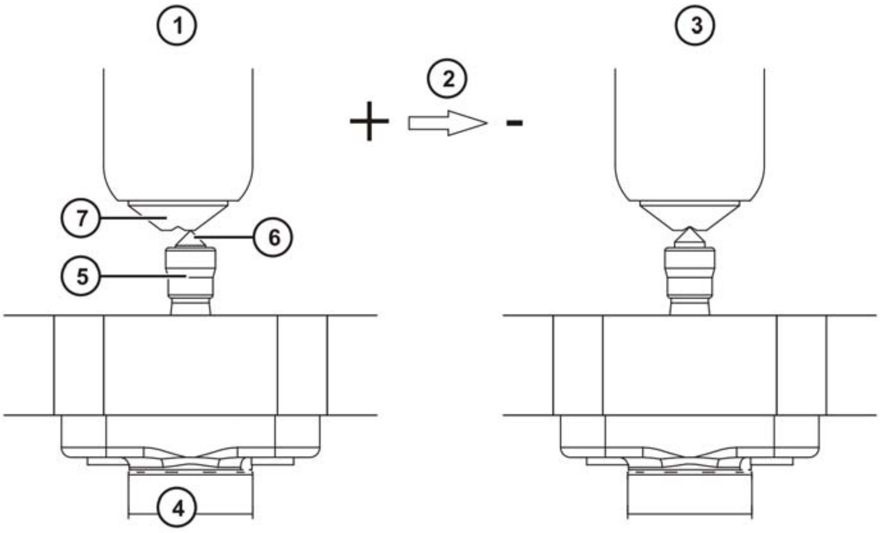


Fig. 6-5: Mastering run

1	Pre-mastering position	5	Gauge cartridge
2	Motion direction of axis	6	Gauge pin
3	Mechanical zero position	7	Reference notch
4	EMT		

Precondition

- Operating mode T1

Procedure

1. Select the jog mode “Jog keys” in the left-hand status key bar:

2. Select external axis jogging in the right-hand status key bar:

3. Select axis-specific jogging in the right-hand status key bar:

4. Hold down the enabling switch.
5. The axes are displayed in the right-hand status key bar. Press the Plus or Minus status key to move an axis in the positive or negative direction.
6. In ascending order, starting with external axis E1, move the axes into the pre-mastering position.

6.1.4 Mastering with the EMT

> If the system contains more than 2 external axes, the signal cable of the EMT must be connected to the second RDC for mastering of the external axes.

Precondition

- All robot axes are mastered.
- All axes are in the pre-mastering position.
- No program is selected.
- Operating mode T1

> Mastering must be carried out either always without a load or always with a load. Otherwise, inaccuracies may result during jogging of the external axis or kinematic system.

Procedure

1. Select the menu **Setup** > **Master** > **EMT** > **With load correction** > **First mastering**.

   An option window is opened. All axes to be mastered are displayed. The axis with the lowest number is highlighted.
2. Remove the protective cap of the gauge cartridge on the axis highlighted in the option window. Screw EMT onto gauge cartridge. Then attach signal cable to EMT and plug into connector X32 on the base frame junction box.

> **Caution!**
> The EMT must always be screwed onto the gauge cartridge **without** the signal cable attached; only then may the signal cable be attached. When removing the EMT, always remove the signal cable from the EMT first, then remove the EMT from the gauge cartridge. Otherwise, the signal cable could be damaged.
> After mastering, remove the signal cable from connection X32. Failure to do so could result in radiation interference or other damage.

3. Press the **Master** softkey.
4. Press an enabling switch and the Start key.

   When the EMT detects the lowest point of the reference notch, the mechanical zero position is reached. The external axis stops automatically. The values are saved. The axis is no longer displayed in the option window.
5. Remove signal cable from EMT. Then remove EMT from the gauge cartridge and replace the protective cap.
6. Repeat steps 2 to 5 for all axes to be mastered.

6.1.5 Mastering with the dial gauge

Precondition

- All robot axes are mastered.
- All axes are in the pre-mastering position.
- Axis-specific jogging with the jog keys is selected.

- No program is selected.
- Operating mode T1

> Mastering must be carried out either always without a load or always with a load. Otherwise, inaccuracies may result during jogging of the external axis or kinematic system.

Procedure

1. Select the menu sequence **Setup** > **Master** > **Dial**.
   An option window is opened. 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 the zero position.
7. Switch to incremental jogging in the right-hand status key bar.
8. Move the axis from “+” to “-” until the zero position is reached.

> If the axis overshoots the zero position, repeat steps 5 to 8.

9. Press the **Master** softkey. The axis that has been mastered is removed from the option 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.

6.2 Calibration

6.2.1 Calibration overview

Calibration of a 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.

Overview

Calibration of a kinematic system consists of 2 steps:

Step	Description
1	Calibrate the root point of the kinematic system. (>>> 6.2.2 "Calibrating the root point" page 50)

Step	Description
2	BASE kinematic system Calibrate the offset base of the kinematic system. (>>> 6.2.6 "Offset base calibration" page 52)
	TOOL kinematic system Calibrate the external tool on the kinematic system. (>>> 6.2.8 "Calibrating an external tool" page 54)

6.2.2 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.

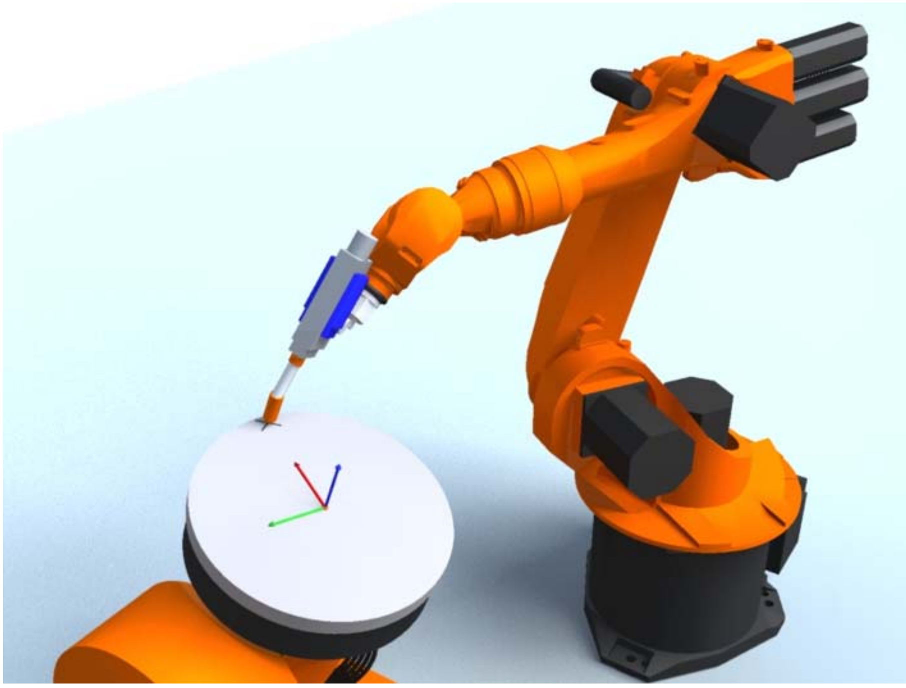


Fig. 6-6: Mark on DKP 400 for root point calibration

Overview

Root point calibration consists of 2 steps:

Step	Description
1	Assign a TOOL coordinate system to the reference point on the kinematic system. (>>> 6.2.3 "Assigning a TOOL coordinate system reference point" page 51)
2	Move to reference point in 4 different positions. (>>> 6.2.4 "Moving to the reference point in 4 different positions" page 51)

If the calibration data are already known, they can be entered directly.
(>>> 6.2.5 "Entering the root point numerically" page 52)

6.2.3 Assigning a TOOL coordinate system reference point

Description

A TOOL coordinate system is assigned to the reference point on the kinematic system and the data for the reference point (TOOL) are entered numerically. The reference point can be freely selected. The TOOL coordinate system is relative to the FLANGE coordinate system of the kinematic system.

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

> Reference point and flange center point must be sufficiently far apart.

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

Procedure

1. Select the menu **Setup** > **Measure** > **Tool** > **Numeric Input**.
2. Assign a number and a name for the reference tool. Confirm with **Continue**.
3. Enter data. Confirm with **Continue**.
4. Press **Save**.

6.2.4 Moving to the reference point in 4 different positions

Description

The TCP of a tool that has already been calibrated is moved to the 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.

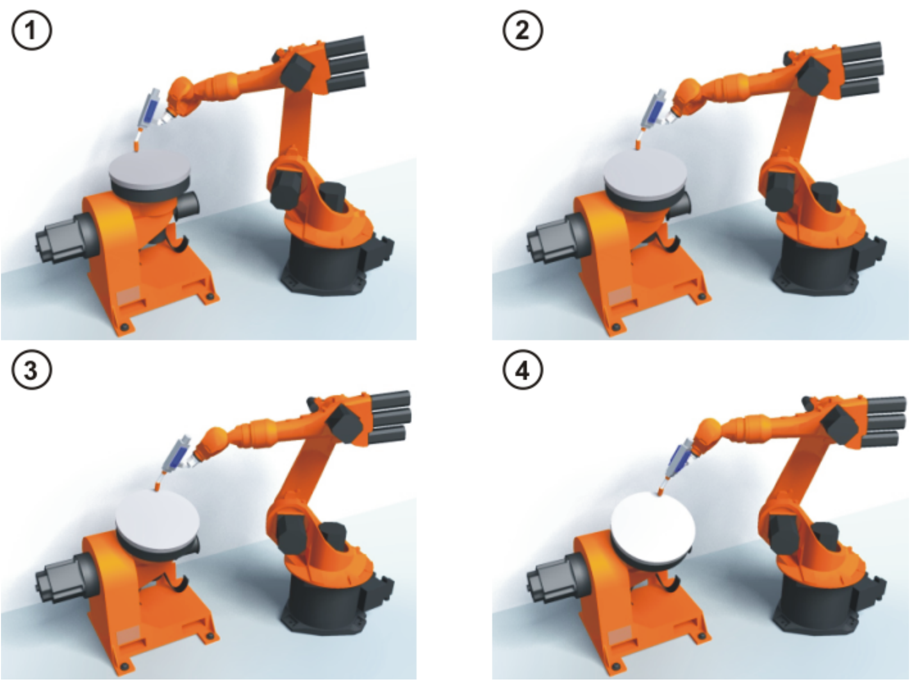


Fig. 6-7: Root point calibration principle

Precondition

- Data have been entered for the reference point (TOOL).
- A previously calibrated tool is mounted on the mounting flange.

- Operating mode T1 or T2

Procedure

1. Select the menu **Setup** > **Measure** > **External kinematic** > **Root point**.
2. Assign a number and a name for the kinematic system. Confirm with **Continue**.
3. Enter the number of the reference tool. Confirm with **Continue**.
4. Enter the number of the mounted tool. Confirm with **Continue**.
5. Move axes of the kinematic system.
6. Move the TCP to the reference point on the kinematic system.
7. Press the **Measure** softkey.
8. Repeat steps 5 to 7 three times.
9. Press **Save**.

6.2.5 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)

Procedure

1. Select the menu **Setup** > **Measure** > **External kinematic** > **Root point (numeric)**.
2. Assign a number and a name for the kinematic system. Confirm with **Continue**.
3. Enter data. Confirm with **Continue**.
4. Press **Save**.

6.2.6 Offset base calibration

Description

The offset base is a moving base that moves in the same way as the kinematic system. It differs from a static base in that its reference point refers to the flange center point of the kinematic system and not the WORLD coordinate system.

During calibration, the TCP of a calibrated tool is moved to the origin and 2 other points of the offset base. These 3 points define the offset base.

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

- The coordinates of the offset base are saved as BASE_DATA[17...22].
- Only one offset base can be calibrated per kinematic system.

If the calibration data are already known, they can be entered directly.
(>>> 6.2.7 "Entering the offset base numerically" page 54)

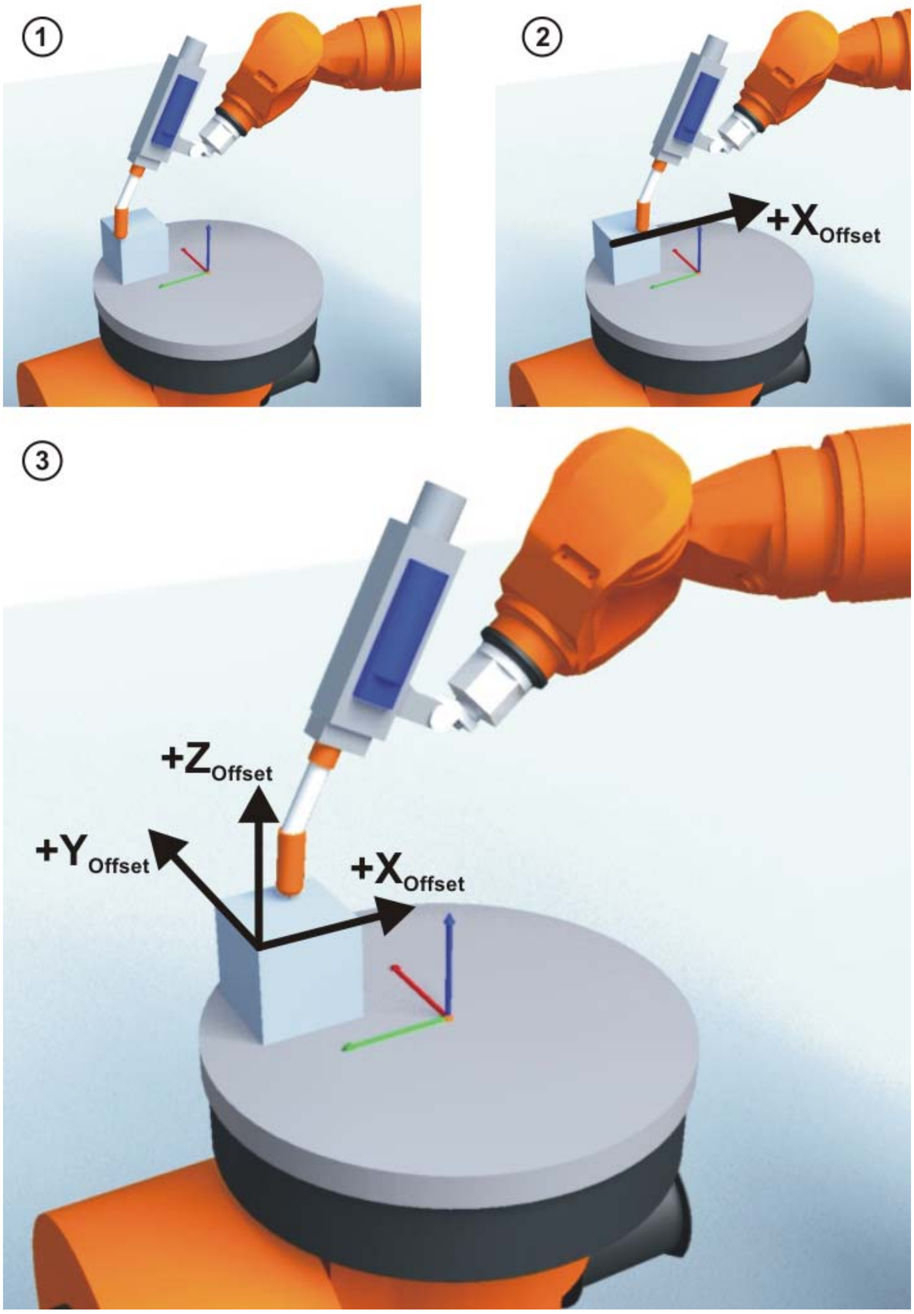


Fig. 6-8: Offset base calibration principle

Precondition

- A previously calibrated tool is mounted on the mounting flange.
- Operating mode T1 or T2

Procedure

1. Select the menu **Setup** > **Measure** > **External kinematic** > **Offset**.
2. Assign a number and a name for the kinematic system. Confirm with **Continue**.
3. Enter the number of the mounted tool. Confirm with **Continue**.
4. Move the TCP to the origin of the offset base. Confirm with **Continue**.
5. Move the TCP to a point on the positive X axis of the offset base. Confirm with **Continue**.
6. Move the TCP to a point in the XY plane with a positive Y value. Confirm with **Continue**.
7. Press **Save**.

6.2.7 Entering the offset base numerically

Precondition

The following numerical values are known, e.g. from CAD data:

- Distance between the origin of the offset base and the origin of the FLANGE coordinate system of the kinematic system (X, Y, Z)
- Rotation of the axes of the offset base relative to the FLANGE coordinate system of the kinematic system (A, B, C)

Procedure

1. Select the menu **Setup** > **Measure** > **External kinematic** > **Offset (numeric)**.
2. Assign a number and a name for the kinematic system. Confirm with **Continue**.
3. Enter data. Confirm with **Continue**.
4. Press **Save**.

6.2.8 Calibrating an external tool

Description

During calibration of the external tool, the user assigns a BASE 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 TCP of the external tool on the kinematic system 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 external 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 external 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.

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

The coordinates of the external tool are saved as BASE_DATA[17...22].

If the calibration data are already known, they can be entered directly.
(>>> 6.2.9 "Entering the external tool numerically" page 56)

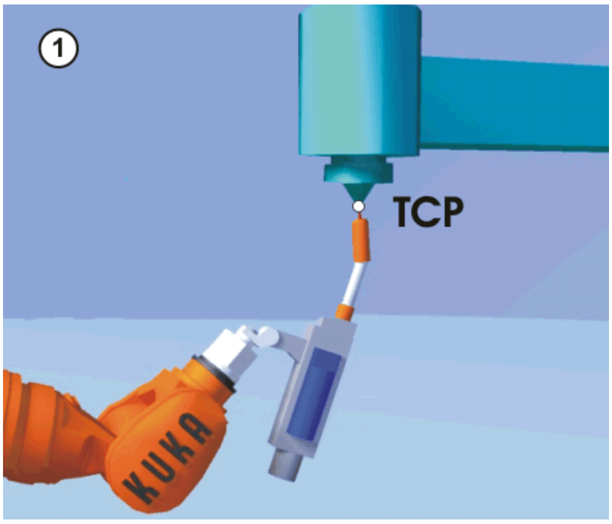


Fig. 6-9: Moving to the external TCP

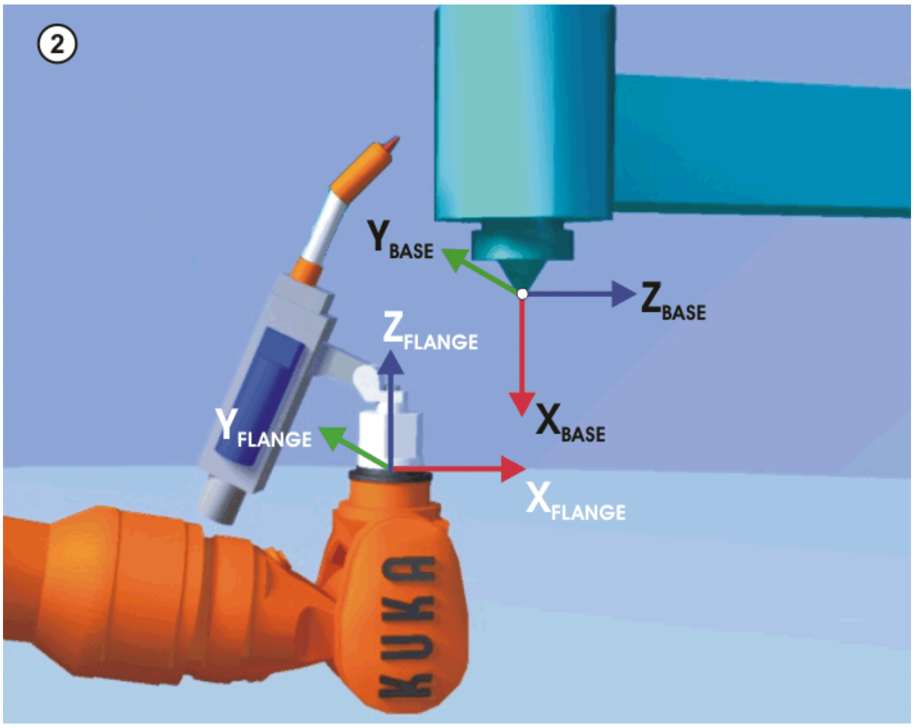


Fig. 6-10: Aligning the coordinate systems parallel to one another

Precondition

- A previously calibrated tool is mounted on the mounting flange.
- Operating mode T1 or T2

Procedure

1. Select the menu **Setup** > **Measure** > **Fixed tool** > **Offset external kinematic**.
2. Assign a number and a name for the kinematic system. Confirm with **Continue**.
3. Enter the number of the calibrated tool. Confirm with **Continue**.
4. Select a variant in the box **5D/6D**. Confirm with **Continue**.
5. Move the TCP of the calibrated tool to the TCP of the external tool. Confirm with **Continue**.
6. If **5D** is selected:

   Align $+X_{BASE}$ parallel to $-Z_{FLANGE}$.

(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:

- $+X_{BASE}$ parallel to $-Z_{FLANGE}$

  (i.e. align the mounting flange perpendicular to the tool direction of the external tool.)
- $+Y_{BASE}$ parallel to $+Y_{FLANGE}$
- $+Z_{BASE}$ parallel to $+X_{FLANGE}$

> This is the default alignment. Depending on customer-specific settings, the axes may be aligned differently.

7. Confirm with **Continue**.
8. Press **Save**.

6.2.9 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)

Procedure

1. Select the menu **Setup** > **Measure** > **Fixed tool** > **Numeric Input**.
2. Assign a number and a name for the external tool. Confirm with **Continue**.
3. Enter data. Confirm with **Continue**.
4. Press **Save**.

7 Configuration

7.1 Configuring external axis systems

Description

KUKA supplies kinematic systems for applications, such as welding, adhesive bonding and laser machining.

The customer can also design his own mechanical systems, however. In this case, either KUKA motors with their own gear units or KUKA motor/gear units – MGUs – must be used.

KUKA MGUs are only to be used for driving rotational positioners, i.e. rotational kinematic systems. Linear units, Cartesian gantries, etc., can only be designed with KUKA motors with their own gear units.

Correct configuration of the system according to the task and correct drive dimensioning in accordance with the load and desired acceleration and velocity are prerequisites for error-free operation.

It is always advisable to discuss the project with KUKA Roboter GmbH to ensure that the correct components are selected and ordered for an external axis system.

Overview

The steps required for starting up and configuring an external axis system vary, depending on the kinematic system used.

Kinematic system used	Description
KUKA linear unit	(>>> 7.1.1 "Starting up a KUKA linear unit" page 57)
KUKA kinematic system	(>>> 7.1.2 "Starting up a KUKA kinematic system" page 57)
Kinematic system with MGU	(>>> 7.1.3 "Starting up a kinematic system with KUKA MGU" page 58)
Kinematic system with KUKA motor	(>>> 7.1.4 "Starting up a kinematic system with KUKA motor" page 58)

7.1.1 Starting up a KUKA linear unit

Installation, mastering, optimization and approval of external kinematic systems for production operation must only be carried out by qualified personnel!

Procedure

1. Master KUKA linear unit.

7.1.2 Starting up a KUKA kinematic system

Installation, mastering, optimization and approval of external kinematic systems for production operation must only be carried out by qualified personnel!

Procedure

1. Check that all hardware components are present and correct, install and connect them.

> The machine data are loaded into the robot controller by KUKA Roboter GmbH during start-up. The machine data can also be found on the CD supplied.

2. Master KUKA kinematic system.
3. If required: optimize controller parameters under maximum load, then record TRACE.
4. If required: optimize acceleration parameters under maximum load, then record TRACE.
5. If required: calibrate KUKA kinematic system.
6. Archive all relevant data, including all trace recordings.

7.1.3 Starting up a kinematic system with KUKA MGU

> Installation, mastering, optimization and approval of external kinematic systems for production operation must only be carried out by qualified personnel!

> Technical data and configuration data for KUKA motor/gear units can be found in the relevant MGU documentation.

Precondition

- The minimum bending radii and specified cable carrier properties for the cables used must be adhered to.

Procedure

1. Check that all hardware components are present and correct, install and connect them.
2. Check the compatibility of the serial numbers and version numbers of the hardware, software and machine data.
3. Define mastering position and axis ranges / software limit switches of the external axes.
4. Create machine data of the kinematic system, including transformation, if desired or necessary.
5. Load the machine data of the kinematic system into the robot controller.
6. Move the axes in axis-specific mode, set motion directions, check gear ratios.
7. Master external axes.
8. Check transformation: move axes in the WORLD coordinate system, check the directions of motion in the WORLD coordinate system.
9. Optimize controller parameters under maximum load, then record TRACE.
10. Optimize acceleration parameters under maximum load, then record TRACE.
11. If required: define reference point and OFFSET base for calibration of the kinematic system.
12. If required: calibrate kinematic system.
13. Archive all relevant data, including all TRACE recordings.

7.1.4 Starting up a kinematic system with KUKA motor

> Installation, mastering, optimization and approval of external kinematic systems for production operation must only be carried out by qualified personnel!

Technical data and configuration data for KUKA motors can be found in the documentation **Motor data**.

Precondition

- The minimum bending radii and specified cable carrier properties for the cables used must be adhered to.
- Dimensioning of the gear unit and drive rating by system builder or KUKA Roboter GmbH.

Procedure

1. Check that all hardware components are present and correct, install and connect them.
2. Check the compatibility of the serial numbers and version numbers of the hardware, software and machine data.
3. Define mastering position and axis ranges / software limit switches of the external axes.
4. Create machine data of the kinematic system, including transformation, if desired or necessary.
5. Load the machine data of the kinematic system into the robot controller.
6. Move the axes in axis-specific mode, set motion directions, check gear ratios.
7. Master external axes.
8. Check transformation: move axes in the WORLD coordinate system, check the directions of motion in the WORLD coordinate system.
9. Optimize controller parameters under maximum load, then record TRACE.
10. Optimize acceleration parameters under maximum load, then record TRACE.
11. If required: define reference point and OFFSET base for calibration of the kinematic system.
12. If required: calibrate kinematic system.
13. Archive all relevant data, including all TRACE recordings.

7.2 Machine data for external axes

The machine data that have to be configured or adapted when external axes are used are grouped together here.

Further information about the machine data is contained in the Expert documentation “Machine Data – KR C2 – For KUKA System Software 5.5”.

Transformation data

(>>> 7.3.1 "Machine data for configuring the transformation" page 61)

Motor-specific machine data

These configuration data can be found in the motor data.

Variable	Description
$VEL_AXIS_MA	Rated motor speed
$SERVOFILE	KSD/motor combination
$CURR_MAX	Maximum KSD current over 2 s
$CURR_LIM	Maximum current setpoint
$KT_MOT	KT factor
$KT0_MOT	KT0 factor
$CURR_MON	Maximum standstill current over 60 s
$RAT_MOT_ENC	Motor/resolver ratio
$RAISE_T_MOT	Motor run-up time

Variable	Description
$BRK_ENERGY_MAX	Maximum switch work per braking operation
$BRK_COOL_OFF_COEFF	Brake cooling factor
$BRK_TORQUE	Dynamic braking torque

Load-specific machine data

Using the oscilloscope function, these configuration data can be optimized according to the maximum load to be moved.

(>>> 7.4 "Optimizing machine data with the oscilloscope" page 65)

Variable	Description
$G_VEL_PTP	Speed controller proportional gain for PTP
$G_VEL_CP	Speed controller proportional gain for CP
$I_VEL_PTP	Integral-action factor speed controller for PTP
$I_VEL_CP	Integral-action factor speed controller for CP
$LG_PTP	Loop gain of position controller for PTP
$LG_CP	Loop gain of position controller for CP
$RAISE_TIME	Axis run-up time
$RED_ACC_EMX	Reduction factor for path-maintaining braking after EMERGENCY STOP
$DECEL_MB	Ramp for path-oriented braking in the case of maximum braking

Additional machine data

Variable	Description
$AXIS_TYPE	Axis type
$MAMES	Mastering position
$RAT_MOT_AX	Motor/axis gear ratio
$DSECHANNEL	Assignment of axes on the DSE
$PMCHANNEL	Assignment of axes on the KPS module
$CURR_COM_EX	Maximum current setpoint for jog mode
$VEL_CPT1_MA	Reduction factor for CP motions in test mode T1
$AXIS_RESO	Positioning resolution of the resolver measurement system
$RED_VEL_AXC	Reduction factor for axial velocity (HOV)
$RED_ACC_AXC	Reduction factor for axial acceleration (HOV)
$VEL_AX_JUS	Velocity for EMT mastering
$L_EMT_MAX	Maximum mastering distance for EMT mastering
$APO_DIS_PTP	Maximum approximation distance
$SEQ_CAL	Mastering sequence
$DIR_CAL	Mastering direction
$BRK_MODE	Brake control mode
$BRK_DEL_EX	Brake delay time
$IN_POS_MA	Positioning window
$SOFTN_END	Negative software limit switch
$SOFTP_END	Positive software limit switch
$TRAFO_AXIS	Number of transformed axes
$AXIS_DIR	Direction of rotation of the axes for the transformation
$INC_EXTAX	Axis-specific increment
$EX_AX_NUM	Number of external axes
$ASR_ERROR	Maximum speed deviation of external position encoder/motor encoder

Variable	Description
$RAT_EXT_ENC	Sensor wheel/sensor ratio
$AX_ENERGY_MAX	Maximum energy of an axis
$AXIS_JERK	Maximum axis jerk

7.3 Transformation

7.3.1 Machine data for configuring the transformation

Overview

Variable	Description
$EX_KIN	(>>> 7.3.1.1 "$EX_KIN" page 61)
$ET*x*_AX	(>>> 7.3.1.2 "$ET1_AX" page 61)
$ET*x*_NAME	(>>> 7.3.1.3 "$ET1_NAME" page 62)
$ET*x*_TA1KR	(>>> 7.3.1.4 "$ET1_TA1KR" page 62)
$ET*x*_TA2A1	(>>> 7.3.1.5 "$ET1_TA2A1" page 63)
$ET*x*_TA3A2	(>>> 7.3.1.6 "$ET1_TA3A2" page 63)
$ET*x*_TFLA3	(>>> 7.3.1.7 "$ET1_TFLA3" page 63)

7.3.1.1 $EX_KIN

Description

Identifier of external transformations

The variable of structure type EX_KIN can be used to assign a kinematic type to external transformations ET1 to ET6.

Syntax

`$EX_KIN={ET1` *Kinematic type ET1* … , `ET6` *Kinematic type ET6*`}`

Explanation of the syntax

Kinematic type	Description
#EASYS to #EFSYS	BASE kinematic system 1 to 6
#ERSYS	ROBROOT kinematic system
#NONE	No external transformation

Example

```
$EX_KIN={ET1 #EASYS,ET2 #EBSYS,ET3 #NONE,ET4 #NONE,ET5 #NONE,ET6
#NONE}
```

External transformations ET1 and ET2 are BASE kinematic systems.

7.3.1.2 $ET1_AX

Description

External axes of the 1st external transformation

The external axes of external transformations ET2 to ET6 are defined analogously with the variables $ET2_AX to $ET6_AX.

The variable of structure type ET_AX defines the external axes that are used by external transformation ET1. This consists of max. 3 transformed axes.

Syntax

`$ET1_AX={TR_A1` *External axis 1* , `TR_A2` *External axis 2* , `TR_A3` *External axis 3*`}`

Explanation of the syntax

The transformed axes can be assigned the following values:

TR_A1...TR_A3	Description
#E1 ... #E6	External axis E1 ... E6
#NONE	No transformed axis

The assignment must begin with transformed axis TR_A1. No gaps are allowed in the aggregate.

Example

```
$ET1_AX={TR_A1 #E2,TR_A2 #NONE,TR_A3 #NONE}
```

The external transformation consists of external axis E2.

7.3.1.3 $ET1_NAME

Description

Name of the 1st external transformation

The names of external transformations ET2 to ET6 are defined analogously with the variables $ET2_NAME to $ET6_NAME.

The variable defines the name of external transformation ET1. The name specified here is displayed in the **Robot** tab accessed via the menu sequence **Help** > **Info**.

Syntax

$ET1_NAME[]="*Name*"

Explanation of the syntax

Element	Description
Name	Data type: CHAR The name can have a maximum length of 20 characters.

Example

```
$ET1_NAME[]="TURNTABLE_1"
```

7.3.1.4 $ET1_TA1KR

Description

Position of the first transformed axis of the external transformation ET1

The variables $ET2_TA1KR to $ET6_TA1KR are available for the external transformations ET2 to ET6.

The variable of structure type FRAME defines the position of the first transformed axis relative to the coordinate system in the root point of the external transformation ET1.

- X, Y, Z: offset of the origin along the axes in [mm]
- A, B, C: rotation of the axis angles in [°]

Example

```
$ET1_TA1KR={X 0.0,Y 280.0,Z 510.0,A 0.0,B 90.0,C 0.0}
```

The origin of the coordinate system is offset, relative to the root point of the external transformation, 280 mm along the Y axis and 510 mm along the Z axis into the joint of the first external axis. Axis angle B is rotated by 90° so that the positive Z direction coincides with the rotational axis of the first external axis.

7.3.1.5 $ET1_TA2A1

Description

Position of the second transformed axis of the external transformation ET1

> The variables $ET2_TA2A1 to $ET6_TA2A1 are available for the external transformations ET2 to ET6.

The variable of structure type FRAME defines the position of the second transformed axis relative to the position of the first transformed axis of the external transformation ET1.

- X, Y, Z: offset of the origin along the axes in [mm]
- A, B, C: rotation of the axis angles in [°]

Example

```
$ET1_TA2A1={X 0.0,Y 0.0,Z 324.0,A 0.0,B -90.0,C 0.0}
```

The origin of the coordinate system is offset, relative to the first transformed axis of the external transformation, 324 mm along the Z axis into the joint of the second external axis. Axis angle B is rotated by 90° so that the positive Z direction coincides with the rotational axis of the second external axis.

7.3.1.6 $ET1_TA3A2

Description

Position of the third transformed axis of the external transformation ET1

> The variables $ET2_TA3A2 to $ET6_TA3A2 are available for the external transformations ET2 to ET6.

The variable of structure type FRAME defines the position of the third transformed axis relative to the position of the second transformed axis of the external transformation ET1.

- X, Y, Z: offset of the origin along the axes in [mm]
- A, B, C: rotation of the axis angles in [°]

Example

```
$ET1_TA3A2={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
```

The external transformation does not use a third external axis.

7.3.1.7 $ET1_TFLA3

Description

Position of the FLANGE coordinate system of the external transformation ET1

> The variables $ET2_TFLA3 to $ET6_TFLA3 are available for the external transformations ET2 to ET6.

The variable of structure type FRAME defines the position of the FLANGE coordinate system relative to the position of the third transformed axis of the external transformation ET1.

- X, Y, Z: offset of the origin along the axes in [mm]
- A, B, C: rotation of the axis angles in [°]

> In the case of ROBROOT kinematic systems, the robot stands on the flange of the kinematic system. In this case, the variable defines the offset and orientation of the robot in the FLANGE coordinate system of the kinematic system.

Example

ROBROOT kinematic system

```
$ET1_TFLA3={X 0.0,Y 0.0,Z 0.0,A 0.0,B -90.0,C 0.0}
```

Axis angle B of the FLANGE coordinate system of the external transformation is rotated by -90°. In this orientation, the robot stands on the flange.

7.3.2 Transformation of BASE kinematic system

Description

The transformation starts at the root point of the kinematic system and ends at the flange of the kinematic system. The transformation from the flange to the reference pin $ETx_TPINFL is no longer taken into consideration. Since software version 5.3, these data must be entered numerically as a tool.

(>>> 6.2.3 "Assigning a TOOL coordinate system reference point" page 51)

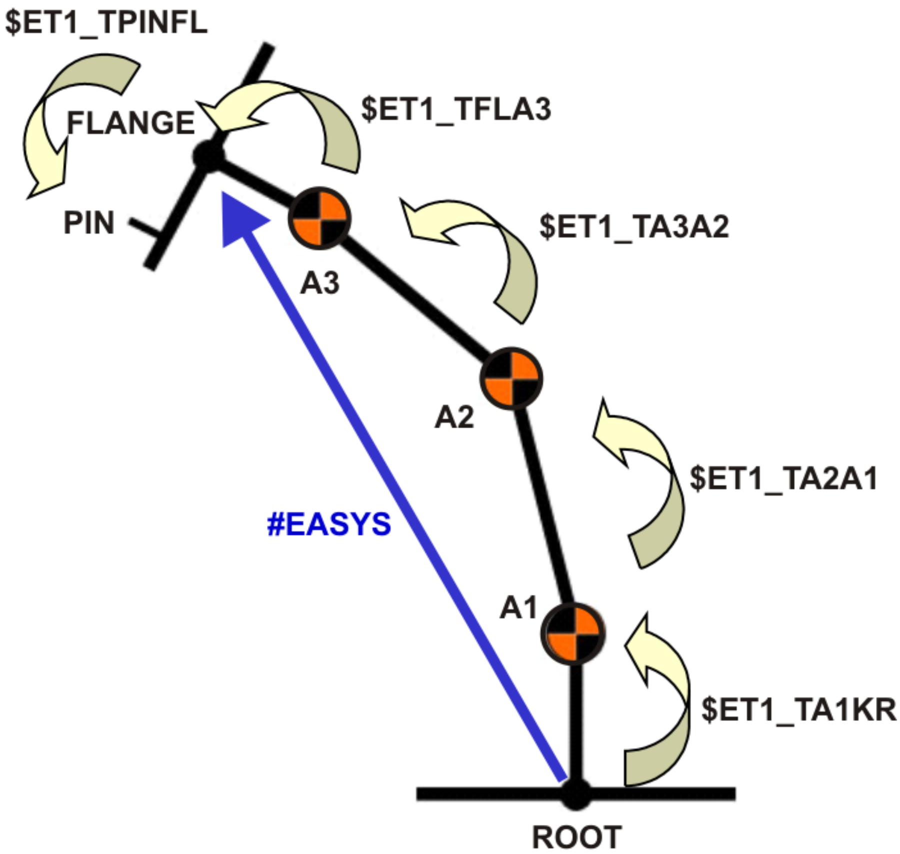


Fig. 7-1: Transformation chain of a BASE kinematic system

Procedure

1. Define the root point of the kinematic system.
2. Define the joints and rotational axes of the kinematic system.
3. Starting at the root point of the kinematic system, offset the coordinate system into the first joint (translation in X, Y and Z direction).
4. There, rotate the coordinate system so that the positive Z axis corresponds to the rotational axis of the first axis (rotation about the angles C, B, A).

Rotation must always be carried out in the sequence C, B, A.

5. If required, offset the coordinate system from the first joint to the second joint and from the second joint to the third joint by means of translation and rotation.

6. If required, offset the coordinate system to the flange center point by means of translation and rotation.

Example (>>> 10.1 "Transformation for DKP 400" page 101)

7.3.3 Transformation of ROBROOT kinematic system

Description

In the case of ROBROOT kinematic systems, the robot stands on the flange of the kinematic system, e.g. KUKA linear unit.

The following rules apply to the transformation of ROBROOT kinematic systems:

- In the case of kinematic systems with one axis, only $ETx_TA1KR is taken into consideration.
- In the case of kinematic systems with 2 axes, $ETx_TA1KR and $ETx_TA2A1 are taken into consideration.
- In the case of kinematic systems with 3 axes, $ETx_TA1KR, $ETx_TA2A1 and $ETx_TA3A2 are taken into consideration.
- $ETx_FLA3 defines the offset and orientation of the robot in the FLANGE coordinate system of the kinematic system and is always taken into consideration.

Procedure

> Here, the transformation is described using the example of a 1-axis ROBROOT kinematic system, i.e. a linear unit.

1. Define the root point of the kinematic system.
2. Starting at the root point of the kinematic system, offset the coordinate system into the flange center point of the kinematic system (translation in X, Y and Z direction).

> The flange is the baseplate on the linear unit.

3. There, rotate the coordinate system so that the positive Z axis corresponds to the direction of travel (rotation about the angles C, B, A).

> Rotation must always be carried out in the sequence C, B, A.

4. Rotate the coordinate system is such a way that the X axis, starting at the connector panel of the robot, points in the positive direction.

Example (>>> 10.2 "Transformation for KL 1500-2" page 106)

7.4 Optimizing machine data with the oscilloscope

Function

The oscilloscope is a function of KUKA.HMI. This function can be used to record, display and analyze different variables with the program running, e.g. actual current, setpoint current, following error, etc.

> Further information about the oscilloscope function is contained in the “Operating and Programming Instructions for System Integrators”.

Overview

The oscilloscope is used to optimize machine data for external axes. There are 2 objectives to be met here:

- Reduction of the cycle time

  For this purpose, the following acceleration parameters are optimized:
  - Acceleration and braking ramp: $RAISE_TIME
  - Ramp for path-oriented braking in the case of maximum braking (STOP 0): $DECEL_MB
  - Ramp for path-maintaining braking after EMERGENCY STOP (STOP 1): $RED_ACC_EMX

  (>>> 7.4.3 "Optimizing acceleration parameters" page 72)
- Increase of path and velocity accuracy

  For this purpose, the following controller parameters are optimized:
  - Proportional component of speed controller: $G_VEL_PTP, $G_VEL_CP
  - Integral component of speed controller: $I_VEL_PTP, $I_VEL_CP
  - Position controller: $LG_PTP, $LG_CP

  (>>> 7.4.2 "Optimizing controller parameters" page 66)

Caution!
Machine data must only be optimized after the kinematic system has warmed up. During operation, gear units and other mechanical components begin to run more smoothly. Optimization with cold drives can result in the kinematic system being over-optimized.

7.4.1 Optimization sequence

The following sequence must be adhered to when optimizing the parameters for external axes by means of the oscilloscope:

Step	Optimization
1	$G_VEL_PTP, $G_VEL_CP
2	$I_VEL_PTP, $I_VEL_CP
3	$LG_PTP, $LG_CP
4	$RAISE_TIME
5	$RED_ACC_EMX
6	$DECEL_MB

7.4.2 Optimizing controller parameters

It is advisable to optimize controller parameters with the maximum permissible load. Optimization with a smaller load may result in reduced cycle times. However, no greater load can then be moved without first carrying out optimization again.

7.4.2.1 Optimizing $G_VEL_PTP and $G_VEL_CP

Description

The proportional component of the speed controller influences the dynamics of the velocity control.

- The higher the proportional component, the greater the reaction of the controller output to a new setpoint value.
- The higher the proportional component, the lower the following error.
- The higher the proportional component, the greater the current pulse height.
- If the control value is set too high, this causes the axis to overshoot and buzz.

- If the control value is set too low, this results in termination of the motion with an error message.

The aim of the optimization is to reduce the following error as far as possible without causing the axis to overshoot or buzz. The optimized value for $G_VEL_PTP and $G_VEL_CP depends on the motor type, the size of the kinematic system and the maximum load to be moved.

Suitable values for most kinematic systems range from 5 to 80. As a general rule, the values must be selected in the lower range for small motors and in the upper range for large motors. It is advisable to commence optimization with a medium start value.

Procedure

1. Set the integral component of the speed controller $I_VEL_PTP to a high value, e.g. 9,999, in order to deactivate its function.
2. Set the proportional component of the speed controller $G_VEL_PTP.
3. Increase or decrease $G_VEL_PTP in increments until dynamic control without current pulses and with a low following error is achieved.

Orientation value for the following error: approx. 1.0 rad

4. Accept the optimized value for $G_VEL_CP.

Soft servo control

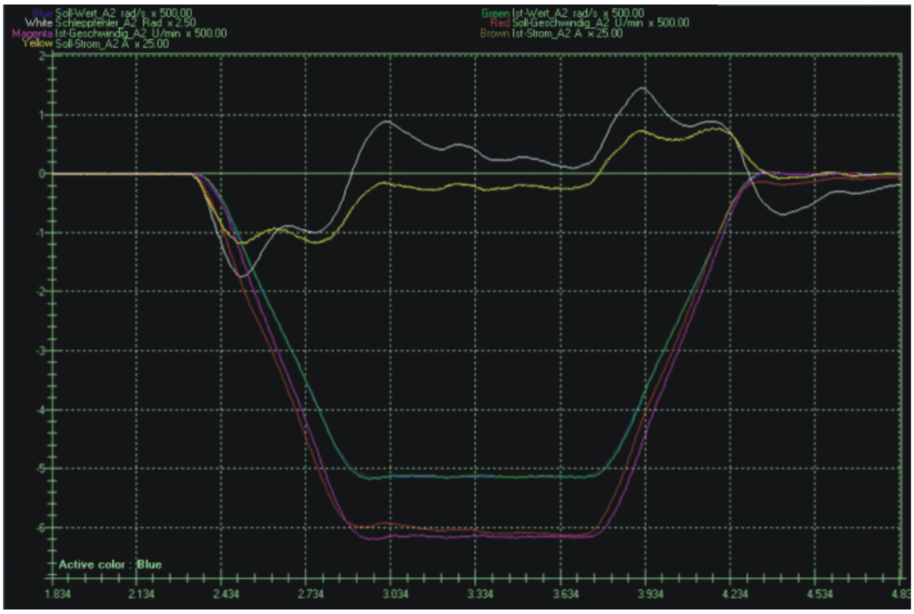


Fig. 7-2: $G_VEL_PTP=30

- Following error: 3.5 rad
- Current pulse height: 0.0 A

The value set for $G_VEL_PTP is too low. The following error is too great.

Hard servo control

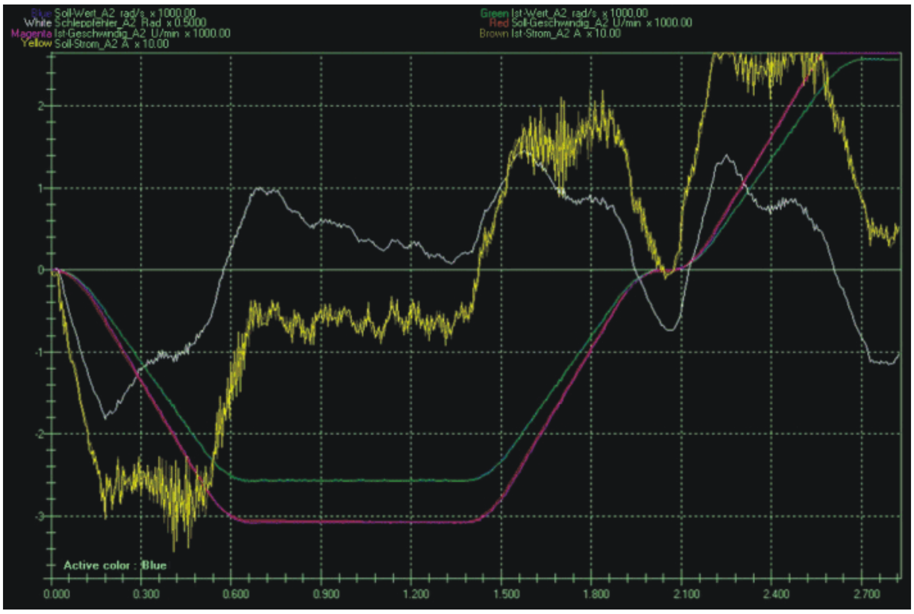


Fig. 7-3: $G_VEL_PTP=120

- Following error: 0.9 rad
- Current pulse height: 10.0 A

The value set for $G_VEL_PTP is too great. The following error is low, but the current pulses are too strong.

Optimized servo control

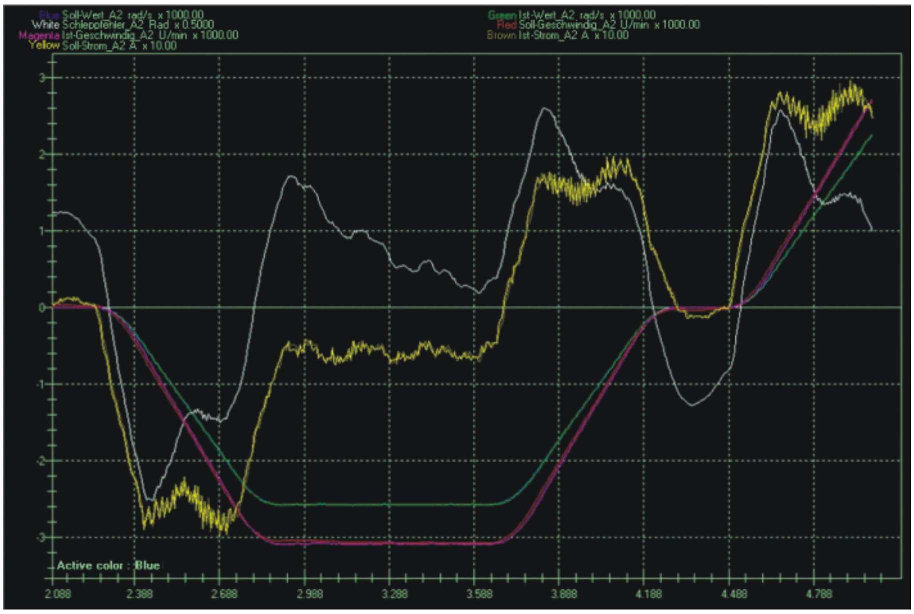


Fig. 7-4: $G_VEL_PTP=75

- Following error: 1.3 rad
- Current pulse height: 4.0 A

7.4.2.2 Optimizing $I_VEL_PTP and $I_VEL_CP

Description

The integral component of the speed controller influences the transient response of the axis to the nominal speed and stabilizes the control loop.

- The lower the integral component, the faster the reaction of the controller output to a new setpoint value.
- The higher the integral component, the greater the following error.

- The integral component has no effect on the current pulse height.
- If the control value is too low, this causes the axis to vibrate.

The aim of the optimization is to reduce the following error as far as possible without causing the axis to vibrate. The optimized value for $I_VEL_PTP and $I_VEL_CP depends on the motor type, the size of the kinematic system and the maximum load to be moved.

> Suitable values for most kinematic systems range from 40 to 800. As a general rule, the values must be selected in the lower range for small motors and in the upper range for large motors. It is advisable to commence optimization with a medium start value.

Procedure

1. Set the integral component of the speed controller $I_VEL_PTP.
2. Increase or decrease $I_VEL_PTP in increments until fast servo control with a low following error is achieved and without the axis vibrating.

> Orientation value for the following error: approx. 1.0 rad

3. Accept the optimized value for $I_VEL_CP.

Slow servo control

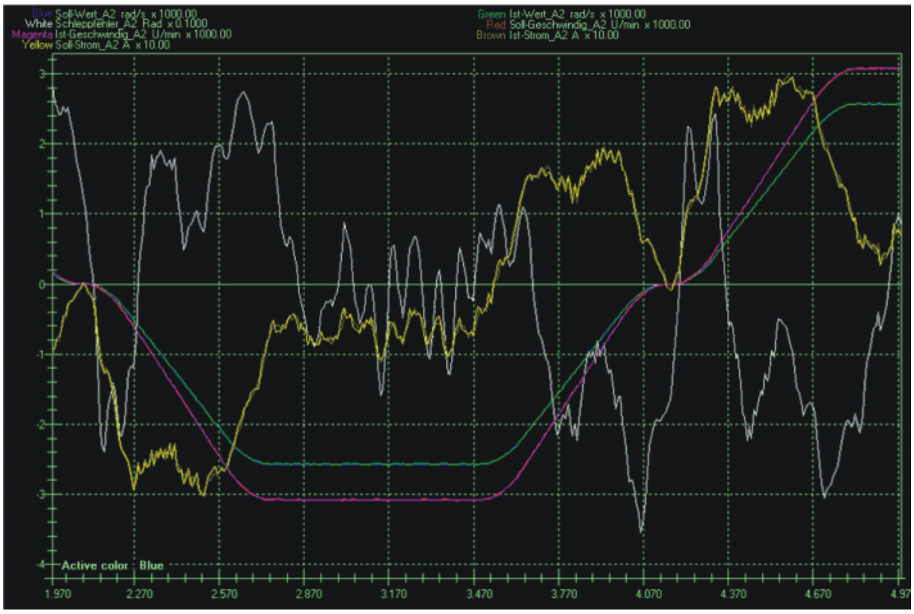


Fig. 7-5: $I_VEL_PTP=999

- Following error: 1.7 rad

The value set for $I_VEL_PTP is too high. The following error is slightly too high and the servo control too slow.

Fast servo control

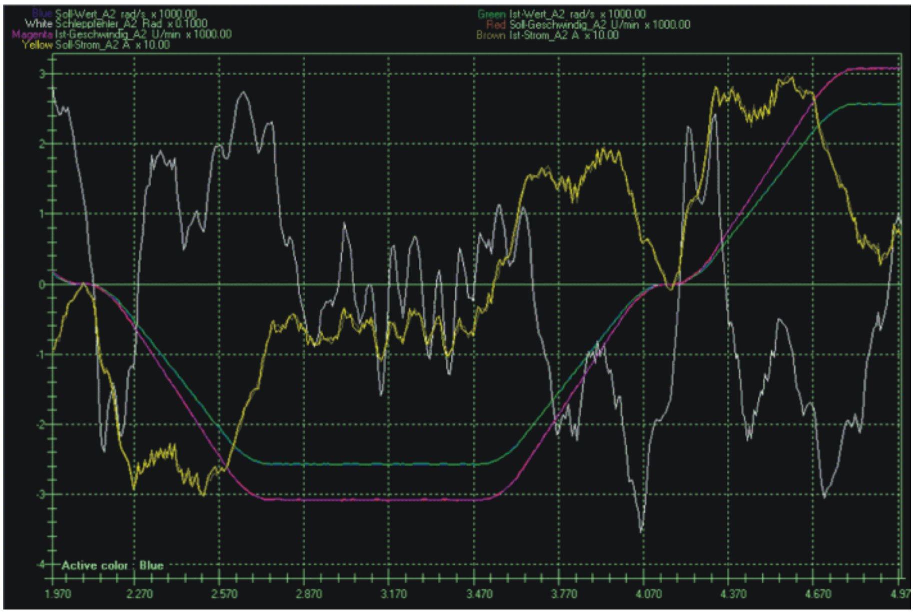


Fig. 7-6: $I_VEL_PTP=20

- Following error: 0.36 rad

The value set for $I_VEL_PTP is too low. The following error is low, but the servo control is too fast. The axis vibrates.

Optimized servo control

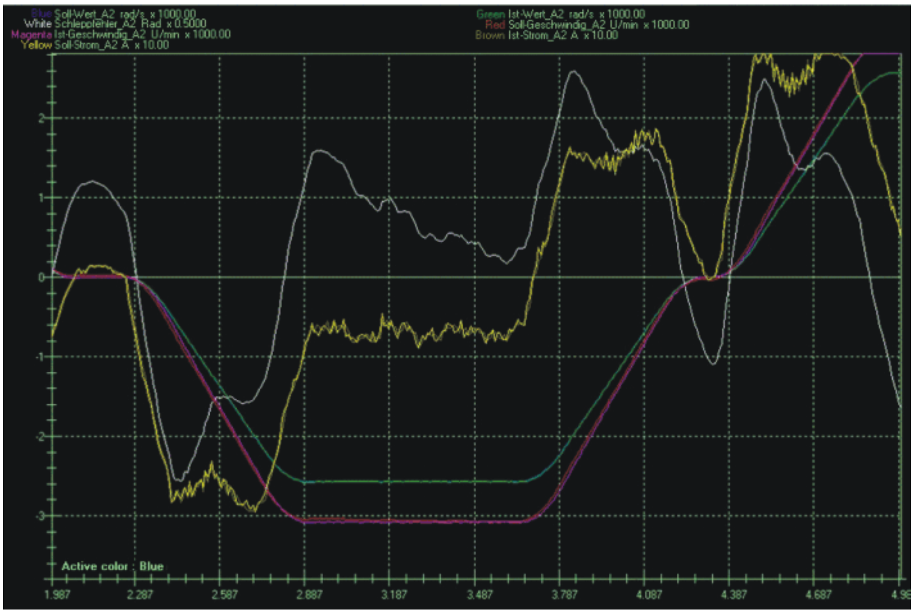


Fig. 7-7: $I_VEL_PTP=400

- Following error: 1.3 rad

7.4.2.3 Optimizing $LG_PTP and $LG_CP

Description

The proportional component of the position controller influences the path-maintaining braking.

- The higher the proportional component, the greater the reaction of the controller output to a new setpoint value.
- The higher the proportional component, the lower the following error.
- The higher the proportional component, the greater the current pulse height.

- If the control value is set too high, this causes the axis to overshoot and buzz.

The aim of the optimization is to reduce the following error as far as possible without causing the axis to overshoot or buzz. The optimized value for $LG_PTP and $LG_CP depends on the motor type, the size of the kinematic system and the maximum load to be moved.

> Suitable values for most kinematic systems range from 0.2 to 0.8. As a general rule, the values must be selected in the upper range for small motors and in the lower range for large motors. It is advisable to commence optimization with a medium start value.

Procedure

1. Set the proportional component of the position controller $LG_PTP.
2. Increase or decrease $LG_PTP in increments until dynamic control without current pulses and with a low following error is achieved.

> Orientation value for the following error: approx. 1.0 rad

3. Accept the optimized value for $LG_CP.

Soft servo control

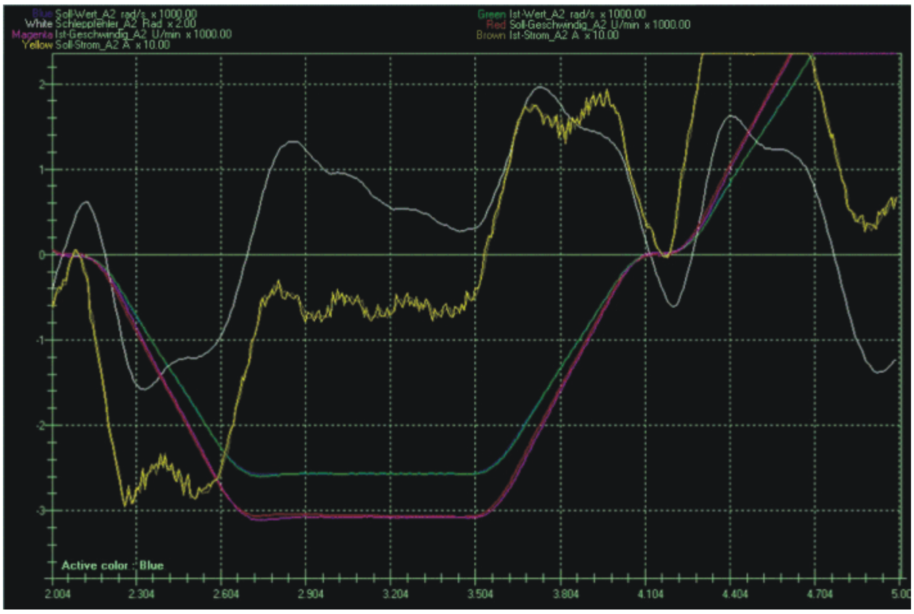


Fig. 7-8: $LG_PTP=0.10

- Following error: 4.0 rad
- Current pulse height: 2.0 A

The value set for $LG_PTP is too low. The following error is too great.

Hard servo control

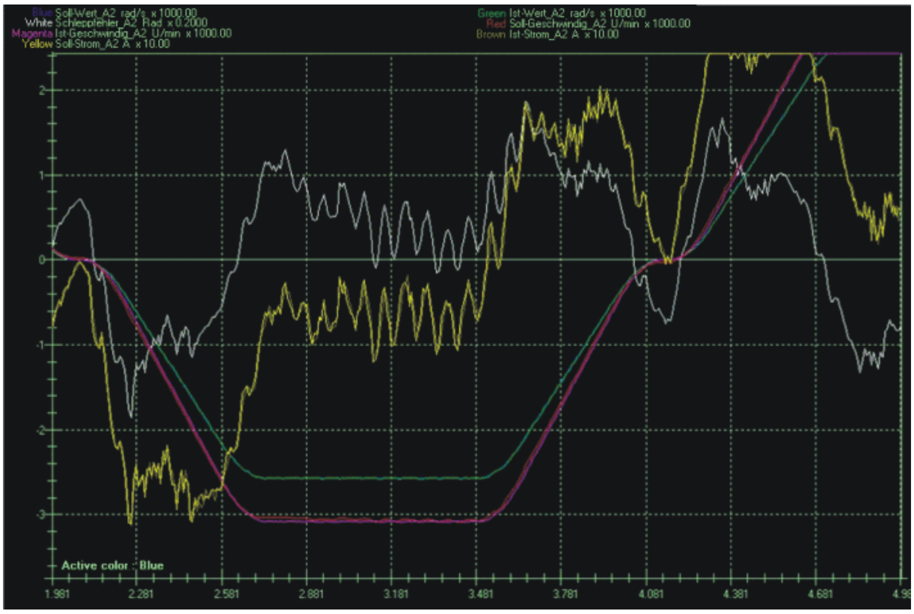


Fig. 7-9: $LG_PTP=1.4

- Following error: 0.9 rad
- Current pulse height: 10.0 A

The value set for $LG_PTP is too great. The following error is low, but the current pulses are too strong.

Optimized servo control

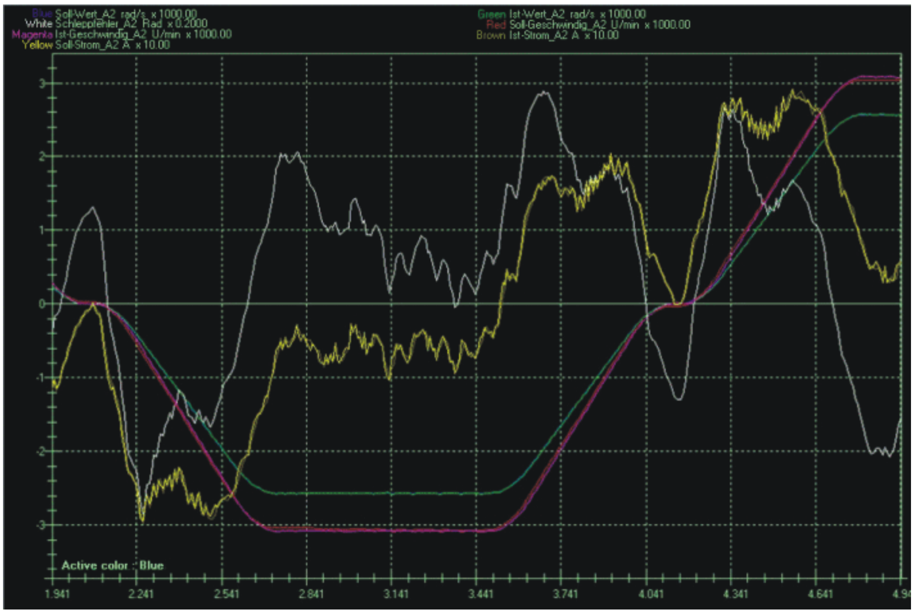


Fig. 7-10: $LG_PTP=0.8

- Following error: 1.3 rad
- Current pulse height: 4.0 A

7.4.3 Optimizing acceleration parameters

It is advisable to optimize acceleration parameters with the maximum permissible load. Optimization with a smaller load may result in reduced cycle times. However, no greater load can then be moved without first carrying out optimization again.

7.4.3.1 Optimizing $RAISE_TIME

Description

$RAISE_TIME defines the time in which an axis is accelerated to rated speed. The aim of the optimization is to move the axes as fast as possible without exceeding the maximum permissible current.

- Too high a value leads to slow accelerations and unnecessarily long cycle times.
- Too low a value leads to fast accelerations so that the axis goes into current limitation. This results in overshoot and following errors.

The required system-specific or customer-specific acceleration and deceleration times must be checked for feasibility. If no value is specified, it is advisable to commence optimization with a start value of 500 ms. This is a feasible value for most kinematic systems.

For optimization, $RAISE_TIME must be reduced in increments in the machine data. During testing of the axis motion, the current must not exceed 90% of the maximum value.

> In the case of very large kinematic systems, a start value of 500 ms may be too low. In this case, the value for the optimization must be increased in increments. Suitable values for most kinematic systems range from 150 to 1,000 ms.

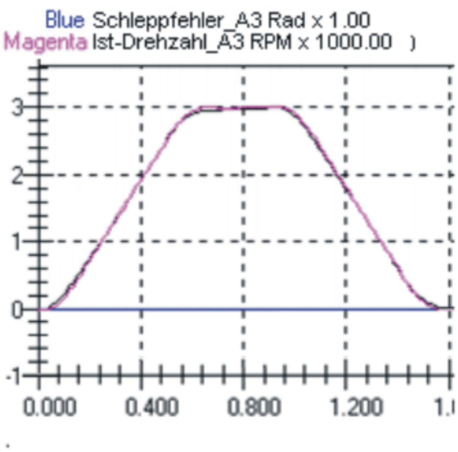


Fig. 7-11: $RAISE_TIME=500

Example

Data for $RAISE_TIME for a 10-axis robot system

```
REAL $RAISE_TIME[12]

$RAISE_TIME[1]=350.0
$RAISE_TIME[2]=750.0
$RAISE_TIME[3]=300.0
$RAISE_TIME[4]=250.0
$RAISE_TIME[5]=180.0
$RAISE_TIME[6]=240.0

$RAISE_TIME[7]=400.0
$RAISE_TIME[8]=150.0
$RAISE_TIME[9]=250.0
$RAISE_TIME[10]=200.0
$RAISE_TIME[11]=0.0
$RAISE_TIME[12]=0.0
```

7.4.3.2 Optimizing $RED_ACC_EMX

Description

$RED_ACC_EMX is used to define a braking ramp for the path-maintaining EMERGENCY STOP. $RED_ACC_EMX is specified as a percentage and refers to $RAISE_TIME, e.g. a value of 200% means that the EMERGENCY STOP braking ramp is twice as steep as the acceleration ramp.

The aim of the optimization is to brake the axes as quickly as possible in the event of an EMERGENCY STOP, without exceeding the maximum permissible current.

- If the braking ramp is too shallow, path-maintaining braking is ensured, but the braking distance is too long for an EMERGENCY STOP.
- If the braking ramp is too steep, the axis goes into current limitation and path-maintaining braking is lost, i.e. the programmed path is left in the case of an EMERGENCY STOP.

The required system-specific or customer-specific deceleration times must be checked for feasibility. If no value is specified, it is advisable to commence optimization with a start value of 100%.

For optimization, $RED_ACC_EMX must be increased in increments in the machine data. When an EMERGENCY STOP button is pressed, the current must not exceed 90% of the maximum value.

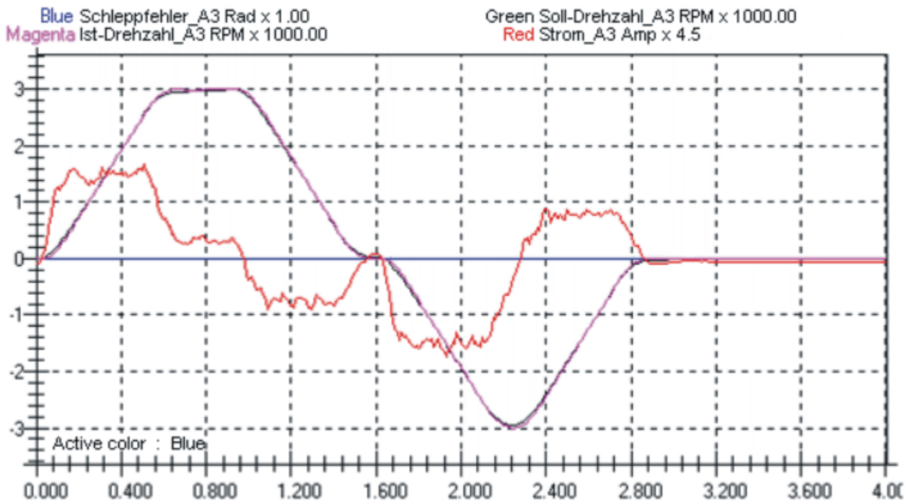


Fig. 7-12: $RED_ACC_EMX=100

Example

Data for $RED_ACC_EMX for a 10-axis robot system

```
INT $RED_ACC_EMX[12]

$RED_ACC_EMX[1]=190
$RED_ACC_EMX[2]=300
$RED_ACC_EMX[3]=300
$RED_ACC_EMX[4]=250
$RED_ACC_EMX[5]=250
$RED_ACC_EMX[6]=250

$RED_ACC_EMX[7]=300
$RED_ACC_EMX[8]=1000
$RED_ACC_EMX[9]=300
$RED_ACC_EMX[10]=150
$RED_ACC_EMX[11]=100
$RED_ACC_EMX[12]=100
```

7.4.3.3 Optimizing $DECEL_MB

Description

$DECEL_MB is used to define a braking ramp for path-oriented maximum braking. The axes are stopped in the time defined in $DECEL_MB, with the axis speed being reduced from maximum to zero.

In the case of maximum braking, the current actual speed value is taken as the speed setpoint and linearly reduced to zero using the set ramp. The ramp prevents the speed setpoint from falling too quickly and causing the current controller to go into limitation, which in turn would prevent the robot from being braked in a controlled manner.

The ramp is calculated for each axis using the optimized values for $RAISE_TIME and $RED_ACC_EMX:

$DECEL_MB = $RAISE_TIME * 100% / $RED_ACC_EMX

Following optimization, the 3 parameters are independent of one another. $DECEL_MB must be at least 180 ms, even if the calculation gives a smaller value.

Example

Data for $DECEL_MB for a 10-axis robot system

```
REAL $DECEL_MB[12]

$DECEL_MB[1]=211.0
$DECEL_MB[2]=267.0
$DECEL_MB[3]=180.0
$DECEL_MB[4]=200.0
$DECEL_MB[5]=200.0
$DECEL_MB[6]=200.0

$DECEL_MB[7]=500.0
$DECEL_MB[8]=200.0
$DECEL_MB[9]=200.0
$DECEL_MB[10]=200.0
$DECEL_MB[11]=0.0
$DECEL_MB[12]=0.0
```

7.4.3.4 Configuration examples

Non-optimized

The axes of most kinematic systems can follow the programming without any problem using the non-optimized start values, but they are moved too slowly.

Parameters:

- $RAISE_TIME=500
- $RED_ACC_EMX=100
- $DECEL_MB=500

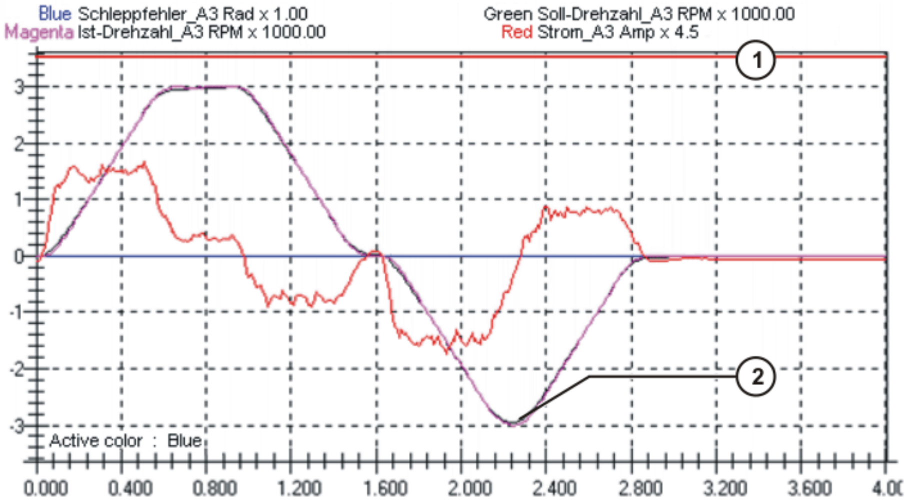


Fig. 7-13: Non-optimized basic setting

1 Current limitation: 16 A
2 E-STOP

Only part of the torque is used (current approx. 8 A) to accelerate the axis to the rated speed. In the event of an EMERGENCY STOP, the axis does not brake with the maximum possible torque. The braking distance is long.

Over-optimized

With over-optimized values, the axes move at maximum velocity, but can no longer follow the programming. During acceleration or braking, the axes leave the programmed path and the setpoint speeds of the motors exceed the values actually reached. In the oscilloscope trace, the axis overshoot and the following errors become visible.

Parameters:

- $RAISE_TIME=100
- $RED_ACC_EMX=300
- $DECEL_MB=180 (minimum permissible value)

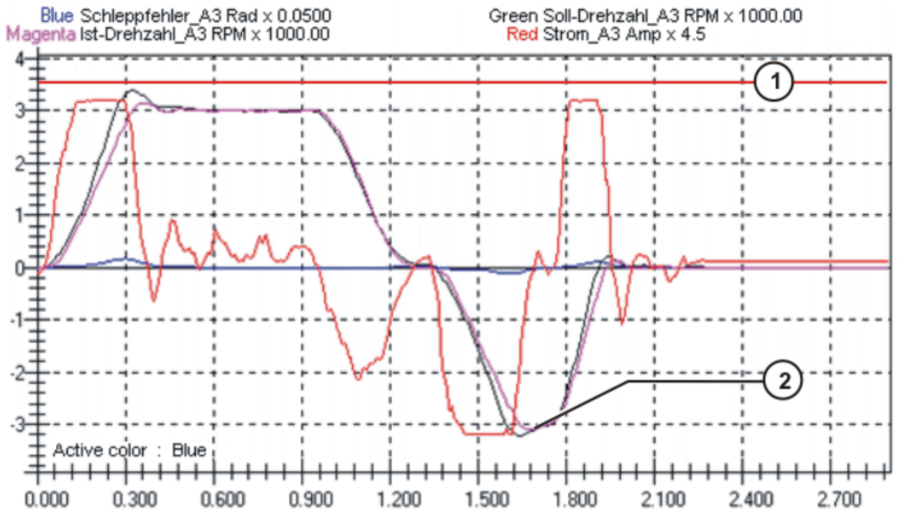


Fig. 7-14: Over-optimized setting

1 Current limitation: 16 A
2 E-STOP

The axis attempts to follow the setpoint speed. The actual speed deviates from the setpoint speed because of current limitation; the following error is large.

Optimized

With optimized values, the axes are accelerated and braked with their maximum values, without leaving the programmed path.

Parameters:

- $RAISE_TIME=250
- $RED_ACC_EMX=250
- $DECEL_MB=180 (minimum permissible value)

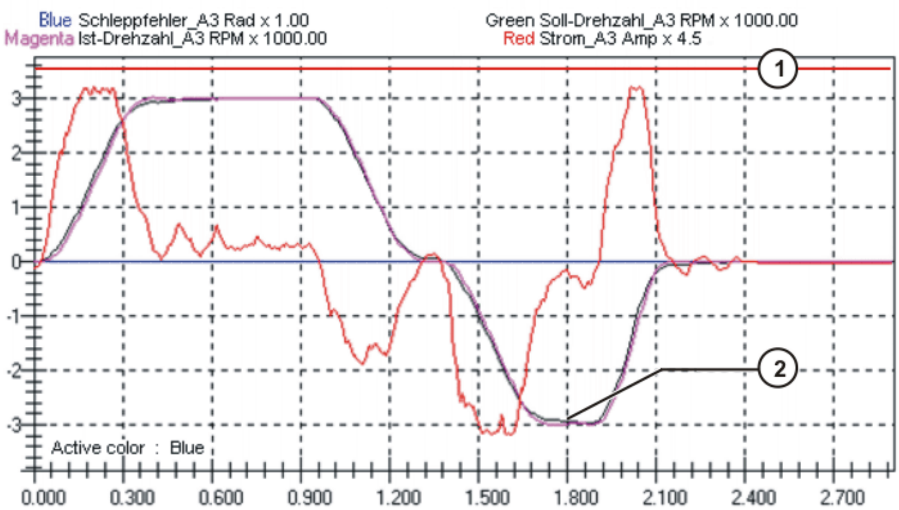


Fig. 7-15: Optimized setting

1 Current limitation: 16 A
2 E-STOP

The maximum torque is used (current approx. 14 A) to accelerate the axis to the rated speed. In the event of an EMERGENCY STOP, the axis brakes with the maximum possible torque. The actual speed is virtually identical to the setpoint speed; the following error is virtually zero.

7.4.4 Checking the optimization results

The optimization results must be checked with the aid of the oscilloscope function.

The following tests must be carried out:

- Motion program in Automatic mode
- EMERGENCY STOP in Automatic mode
- Path-oriented braking in T2 mode after releasing the enabling switch

The following phenomena must not occur:

- Strong current pulses, i.e. axis hums during motion or vibrates.
- Axis goes into current limitation.
- Permissible following error is exceeded, i.e. motion is aborted with an error message.

7.4.5 Checking the r.m.s. current over a program cycle

Description

$CURR_MON defines the maximum permissible standstill current over 60 s. The variable defines the limit for the I^2t monitoring at 55 °C (rise in temperature of cables, amplifiers and motors).

If, during program execution, the value of $CURR_MON is exceeded over 60 s, the message *"i*i-t monitoring, current limit of the motor cable Ex after 60 s exceeded 100%"* is displayed in the message window. In this case, the r.m.s. current of external axis Ex must be recorded with the oscilloscope over the entire program cycle, including the wait times.

The r.m.s. value must be lower than $CURR_MON. If $CURR_MON is exceeded, the power consumption of the axis must be reduced.

> The value for $CURR_MON in the machine data must not be modified. This value is dependent on the motor type and must be taken from the motor data.

Solution strategies:

- Insert wait time into program.
- Reduce program override and subsequently increase it again.

> In the case of continuous-path applications with defined CP velocity, the program override must not be reduced, e.g. welding or bonding.

- Make acceleration and braking ramps less steep. Disadvantage: ramps are valid for all loads and programs (longer cycle times).
- Reduce mass or mass inertia of the load, e.g. by drilling holes or using a different material.
- Distribute load over a number of different axes.

7.5 Modifying machine data

> Machine data for external axis systems may only be configured by specially trained personnel.

> **Warning!**
> Incomplete or incorrect machine data may result in unpredictable machine motions. This can cause danger to life and limb. The system must always be checked and tested after machine data have been modified.

Overview

There are 3 ways of modifying machine data:

- Modify individual machine data directly in $MACHINE.DAT.

  (>>> 7.5.1 "Modifying individual machine data" page 79)
- Create machine data with the text editor and load them into the robot controller.

  (>>> 7.5.2 "Loading machine data via a text file" page 79)
- Create machine data with the axis configurator and load them into the robot controller.

  (>>> 7.6.1 "Configuring machine data for external axes" page 81)

7.5.1 Modifying individual machine data

No additional lines can be inserted into $MACHINE.DAT. Only modifications to existing lines are permissible.

Precondition

- Expert user group

Procedure

1. Open $MACHINE.DAT.
2. Go to the desired variables and modify the values.
3. Press the **Close** softkey. Respond to the request for confirmation asking whether the changes should be saved by pressing the **Yes** softkey.
4. Initialize modified machine data. To do so, reinitialize the user interface or reboot the robot controller with a cold restart.

7.5.2 Loading machine data via a text file

No additional lines can be inserted into $MACHINE.DAT. Only modifications to existing lines are permissible.

Precondition

- Expert user group

Procedure

1. Create machine data with a text editor, e.g. NOTEPAD.
2. Save machine data as a TXT file in directory C:\ in Windows.
3. Open the file in the KUKA.HMI.
4. Copy machine data and close the TXT file.
5. Open $MACHINE.DAT.
6. Activate the DEF line. To do so, select the menu sequence **Configure** > **Tools** > **Editor** > **Def-line**.
7. Create a blank line immediately before the ENDDAT line.

Additional blank lines before the ENDDAT line overwrite previous lines.

8. Position the cursor in the blank line and insert machine data. To do so, select the menu sequence **Edit** > **Paste**.
9. Press the **Close** softkey. Respond to the request for confirmation asking whether the changes should be saved by pressing the **Yes** softkey.
10. Initialize modified machine data. To do so, reinitialize the user interface or reboot the robot controller with a cold restart.

7.5.3 Modifying controller parameters

There are 2 other options available for modifying controller parameters, e.g. $G_VEL_PTP:

- Modify parameters by means of the variable correction function.
- Modify parameters by means of KRL instructions in the program.

In the case of master/slave configurations, only the controller parameters for master drives can be modified in this way.

7.6 Axis configurator for external axes

Overview

The axis configurator is part of KUKA.HMI and has the following functions:

- Parameterization of external axes
- Parameterization of main axes
- Editing of transformation data
- Creation of servo files

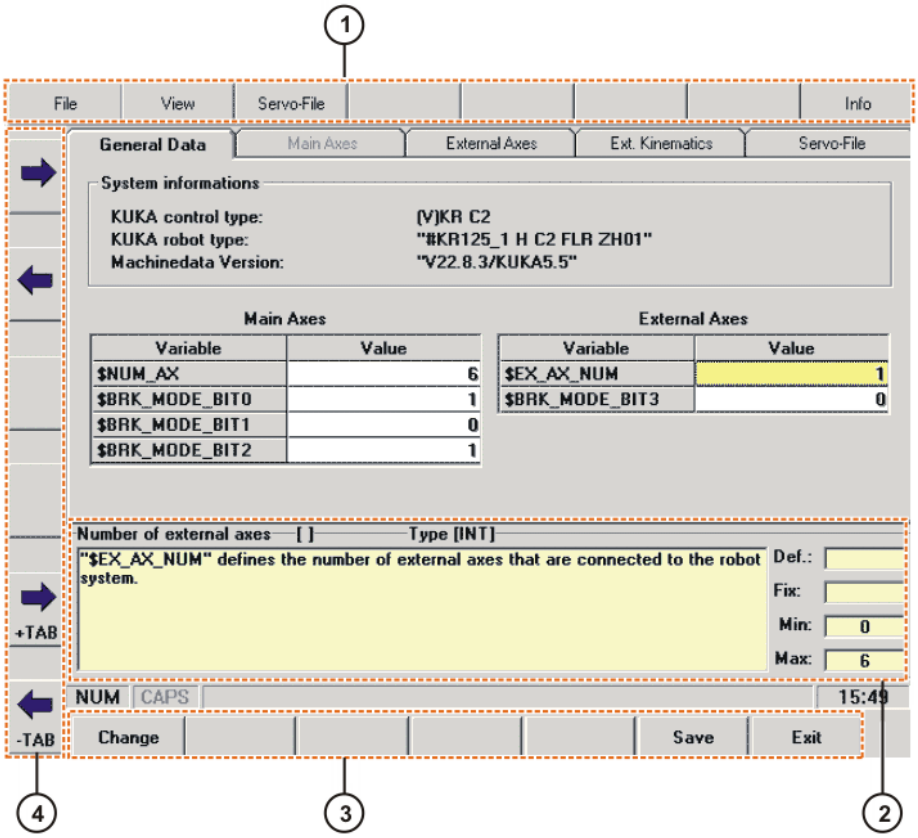


Fig. 7-16: Axis configurator, user interface

1	Menu	3	Softkeys
2	Information window	4	Status keys

Information window

This window contains information about the selected variable, e.g.:

- Brief definition
- Data type and unit
- **Def.** box: default value
- **Fix** box: fixed value
- **Min** box: minimum value
- **Max** box: maximum value

Status keys

Status key	Description
	Jump to the next tab
	Jump to the previous tab

Status key	Description
+TAB	Jump to the next table (in the tab)
-TAB	Jump to the previous table (in the tab)

Softkeys

Softkey	Description
Change	The selected variable can be modified. To select a variable, move to the relevant box using the arrow keys. Then enter a new value and accept it with the Enter key.
Create	Creates a servo file. This softkey is only available in the **Servo file** tab.

7.6.1 Configuring machine data for external axes

Machine data for external axis systems may only be configured by specially trained personnel.

Warning!
Incomplete or incorrect machine data may result in unpredictable machine motions. This can cause danger to life and limb. The system must always be checked and tested after machine data have been modified.

Precondition

- User group “Expert”
- Operating mode T1 or T2.
- No program is selected.

Procedure

1. Select the menu sequence **Setup** > **Service** > **Axisconfigurator**.

   The current machine data are loaded into the configurator. Depending on the specific machine data set, individual tabs of the axis configurator may be deactivated.
2. Go to the desired tabs and modify the machine data.
3. Press the **Save** softkey. Respond to the request for confirmation asking whether the changes should be saved with **Yes**.

   The existing data are overwritten.
4. Confirm the message about initializing the modifications with **OK**.
5. Exit the axis configurator by pressing the **End** softkey. Confirm the request for confirmation with **Yes**.
6. Initialize modified machine data. To do so, reinitialize the user interface or reboot the robot controller with a cold restart.

7.6.2 Archiving a configuration

Precondition

- User group “Expert”
- Operating mode T1 or T2.
- No program is selected.

Procedure

1. Select the menu sequence **Setup** > **Service** > **Axisconfigurator**.

2. In the axis configurator, select the menu sequence **File** > **Archive ext. axes data**.
3. Specify a path and name for the archive file and confirm with **Save**.

7.6.3 Loading a configuration

Precondition

- User group “Expert”
- Operating mode T1 or T2.
- No program is selected.

Procedure

1. Select the menu sequence **Setup** > **Service** > **Axisconfigurator**.
2. In the axis configurator, select the menu sequence **File** > **Load ext. axes data**.
3. Navigate to the archive file, select the archive file and load with **Open**.

7.6.4 Creating a servo file

Precondition

- User group “Expert”
- Operating mode T1 or T2.
- No program is selected.

Procedure

1. Select the menu sequence **Setup** > **Service** > **Axisconfigurator**.
2. Go to the **Servo-File** tab and enter all servo file parameters.
3. Press the **Create** softkey.

   The servo file data are created automatically.
4. To create additional servo files, select the menu sequence **Servo-File** > **Create a new Servo-File** in the axis configurator.

7.6.5 Loading a servo file

Precondition

- User group “Expert”
- Operating mode T1 or T2.
- No program is selected.

Procedure

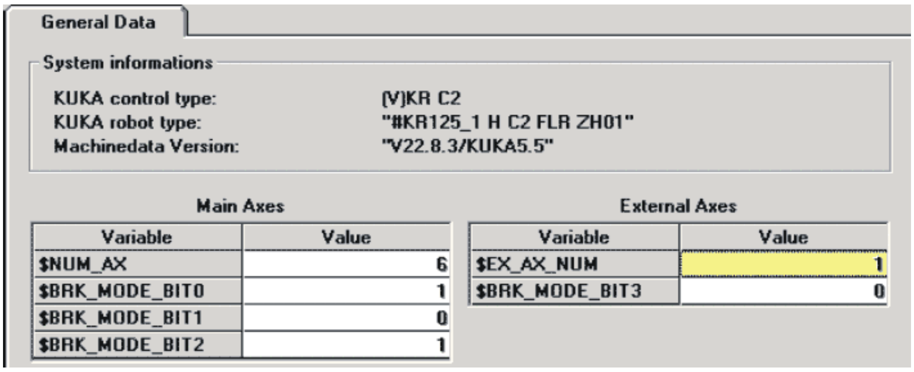
1. Select the menu sequence **Setup** > **Service** > **Axisconfigurator**.
2. In the axis configurator, select the menu sequence **Servo-File** > **Open Servo-File**.
3. Select the servo file in the directory KRC:\R1\MADA or navigate to the directory with the servo file, select the servo file and load it with **Open**.

> It is advisable not to modify KUKA servo files. These files already contain optimized values for a motor/KSD combination.

7.6.6 “General Data” tab

The following machine data can be configured here:

- Number of external axes or axes in the kinematic system
- Brake control mode


General Data

System informations

KUKA control type:	(V)KR C2
KUKA robot type:	"#KR125_1 H C2 FLR ZH01"
Machinedata Version:	"V22.8.3/KUKA5.5"

Main Axes

Variable	Value
$NUM_AX	6
$BRK_MODE_BIT0	1
$BRK_MODE_BIT1	0
$BRK_MODE_BIT2	1

External Axes

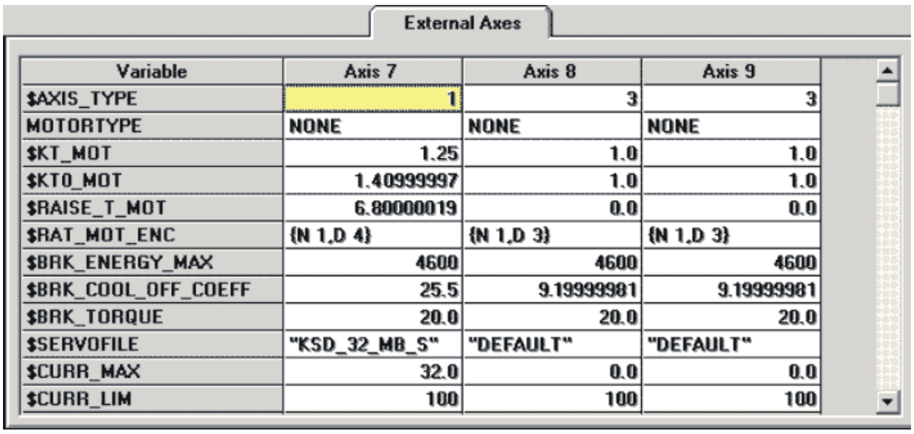
Variable	Value
$EX_AX_NUM	1
$BRK_MODE_BIT3	0

Fig. 7-17: “General Data” tab

Group	Description
System information	System information from $MACHINE.DAT: ■ KUKA robot type: $TRAFONAME ■ Machine data version: $V_R1MADA
Main axes	Axis data ■ Robot axes: values cannot be modified. ■ Axes of a KMC kinematic system: values can be modified.
External axes	External axis data Values can be modified.

7.6.7 “External Axes” tab

The machine data for the external axes of a KUKA kinematic system can be configured here.


External Axes

Variable	Axis 7	Axis 8	Axis 9
$AXIS_TYPE	1	3	3
MOTORTYPE	NONE	NONE	NONE
$KT_MOT	1.25	1.0	1.0
$KT0_MOT	1.40999997	1.0	1.0
$RAISE_T_MOT	6.80000019	0.0	0.0
$RAT_MOT_ENC	{N 1,D 4}	{N 1,D 3}	{N 1,D 3}
$BRK_ENERGY_MAX	4600	4600	4600
$BRK_COOL_OFF_COEFF	25.5	9.19999981	9.19999981
$BRK_TORQUE	20.0	20.0	20.0
$SERVOFILE	"KSD_32_MB_S"	"DEFAULT"	"DEFAULT"
$CURR_MAX	32.0	0.0	0.0
$CURR_LIM	100	100	100

Fig. 7-18: “External Axes” tab

7.6.8 “Main Axes” tab

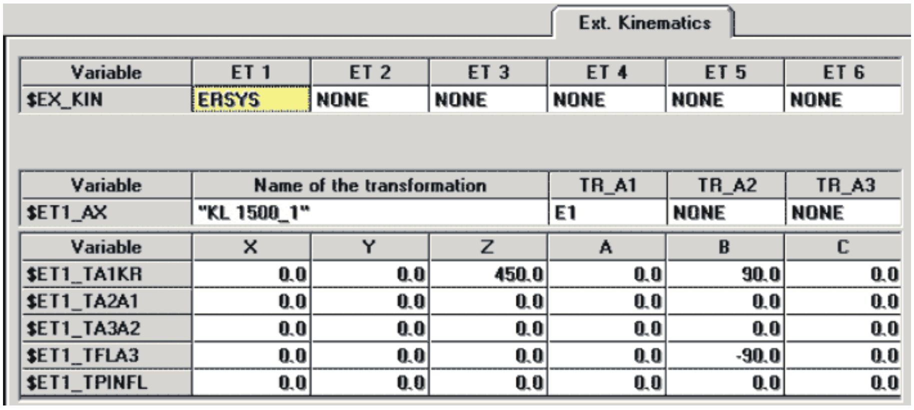
The machine data for the axes of a KMC kinematic system can be configured here.

This tab is enabled if $TRAFONAME is the name of a KMC kinematic system:

- The string “#KR” must not be contained.
- The string “ C2 ” or “ C3E ” must be contained. The spaces in the string must be observed.

7.6.9 “Ext. Kinematics” tab

The machine data for the transformation of a kinematic system can be configured here.



Ext. Kinematics

Variable	ET 1	ET 2	ET 3	ET 4	ET 5	ET 6
$EX_KIN	ERSYS	NONE	NONE	NONE	NONE	NONE

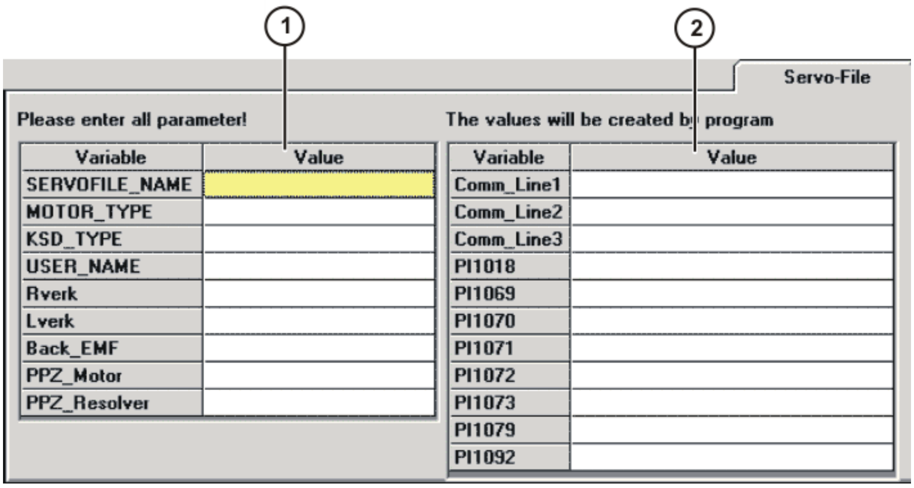
Variable	Name of the transformation	TR_A1	TR_A2	TR_A3
$ET1_AX	"KL 1500_1"	E1	NONE	NONE

Variable	X	Y	Z	A	B	C
$ET1_TA1KR	0.0	0.0	450.0	0.0	90.0	0.0
$ET1_TA2A1	0.0	0.0	0.0	0.0	0.0	0.0
$ET1_TA3A2	0.0	0.0	0.0	0.0	0.0	0.0
$ET1_TFLA3	0.0	0.0	0.0	0.0	-90.0	0.0
$ET1_TPINFL	0.0	0.0	0.0	0.0	0.0	0.0

Fig. 7-19: “External Axes” tab

7.6.10 “Servo-File” tab

Servo files can be created here.



Variable	Value
SERVOFILE_NAME	
MOTOR_TYPE	
KSD_TYPE	
USER_NAME	
Rverk	
Lverk	
Back_EMF	
PPZ_Motor	
PPZ_Resolver	

Variable	Value
Comm_Line1	
Comm_Line2	
Comm_Line3	
PI1018	
PI1069	
PI1070	
PI1071	
PI1072	
PI1073	
PI1079	
PI1092	

Fig. 7-20: “Servo-File” tab

1 Servo file parameters
2 Servo file data created from the parameters

8 System variables

8.1 System variables for configuring external axes

Enabling	
$ASYNC_OPT	(>>> 8.2.2 "$ASYNC_OPT" page 85)
$EXT_AXIS	(>>> 8.2.1 "$EXT_AXIS" page 85)

Asynchronous, uncoordinated external axes	
$ASYNC_AX...	(>>> 8.3.2 "$ASYNC_AX…" page 86)
$ZUST_ASYNC	(>>> 8.3.1 "$ZUST_ASYNC" page 86)

Asynchronous, uncoordinated external axes (ASYPTP)	
$ASYNC_MODE	(>>> 8.4.2 "$ASYNC_MODE" page 88)
$ASYNC_T1_FAST	(>>> 8.4.1 "$ASYNC_T1_FAST" page 87)

Permanently asynchronous external axes	
$EX_AX_ASYNC	(>>> 8.6.1 "$EX_AX_ASYNC" page 90)

Decoupled external axes	
$ASYNC_EX_AX_DECOUPLE	(>>> 8.5.1 "$ASYNC_EX_AX_DECOUPLE" page 89)

8.2 Enabling

8.2.1 $EXT_AXIS

Description

Enabling of external axes

External axes must be enabled by means of the variable in the directory KRC:\STEU\MADA\$OPTION.DAT.

Syntax

`$EXT_AXIS=`*State*

Explanation of the syntax

Element	Description
State	Data type: BOOL ■ **TRUE**: External axes can be configured. (Default) ■ **FALSE**: External axes cannot be configured.

8.2.2 $ASYNC_OPT

Description

Enabling of asynchronous external axes

Asynchronous external axes must be enabled by means of the variable in the directory KRC:\STEU\MADA\$OPTION.DAT.

Syntax

`$ASYNC_OPT=`*State*

Explanation of the syntax

Element	Description
State	Data type: BOOL ■ **TRUE**: External axes can be switched to asynchronous mode. ■ **FALSE**: External axes cannot be switched to asynchronous mode. (Default)

8.3 Asynchronous, uncoordinated external axes

8.3.1 $ZUST_ASYNC

Description

Input for enabling switch

A separate enabling switch must be pressed for asynchronous, uncoordinated motions. Releasing the enabling switch terminates the motion.

A digital input must be assigned to this enabling switch by means of the variable in the directory KRC:\STEU\MADA\$MACHINE.DAT with a signal declaration.

> There is only one input for all asynchronous external axes.

Syntax

`$ZUST_ASYNC $IN[`*Input number* `]`

Explanation of the syntax

Element	Description
Input number	Data type: INT ■ **1 ... 4 096**

Example

```
SIGNAL $ZUST_ASYNC $IN[105]
```

The enabling switch is connected to input 105.

8.3.2 $ASYNC_AX...

Description

Motion direction of the asynchronous, uncoordinated external axes

Asynchronous, uncoordinated external axes must be assigned one digital input for the positive motion direction and one digital input for the negative motion direction by means of the variable in the directory KRC:\STEU\MADA\ $MACHINE.DAT.

Syntax

`$ASYNC_AX`*Axis number*`_P$IN[`*Input number*`]`

`$ASYNC_AX`*Axis number*`_M$IN[`*Input number*`]`

Explanation of the syntax

Element	Description
Axis number	Data type: INT ■ **1 … 6**: external axis E1 … E6
Input number	Data type: INT ■ **1 ... 4 096**
`_P $IN`	Input for positive motion direction
`_M $IN`	Input for negative motion direction

Example

```
SIGNAL $ASYNC_AX1_P $IN[100]
...
SIGNAL $ASYNC_AX1_M $IN[101]
```

External axis E1 is moved asynchronously in the positive direction by means of input 100 and in the negative direction by means of input 101.

8.4 Asynchronous, coordinated external axes (ASYPTP)

8.4.1 $ASYNC_T1_FAST

Description

Velocity reduction in test mode T1

The velocity reduction for ASYPTP motions can be deactivated using the variable in the directory KRC:\R1\MADA\$MACHINE.DAT. If velocity reduction is deactivated, ASYPTP motions can be executed in T1 mode at the programmed velocity.

Caution!
Velocity reduction may only be deactivated for external axes in applications that are not safety-relevant.

Syntax

`$ASYNC_T1_FAST=`*n-bit value*

Explanation of the syntax

Element	Description
n-bit value	The value specifies the external axes for which the velocity reduction is deactivated: ■ **Bit n = 0**: velocity reduction is activated. ■ **Bit n = 1**: velocity reduction is deactivated. **Note:** Bits may only be set for external axes that have been configured via $EX_AX_NUM. If no external axes have been configured ($EX_AX_NUM=0), the value is not checked. Bit value: ■ LSB: external axis E1 ■ MSB: external axis E6

Bit n	5	4	3	2	1	0
Axis	E6	E5	E4	E3	E2	E1

Example 1

The following ASYPTP motion is programmed in the KRL program:

```
ASYPTP={E1 20.0}
```

Asynchronous external axis E1 can be moved to position 20° at the programmed velocity in T1 mode. The following applies in this case:

```
$ASYNC_T1_FAST='B0001'
```

Example 2

The following ASYPTP motion is programmed in the KRL program:

```
ASYPTP={E1 20.0, E2 50.0}
```

Asynchronous external axes E1 and E2 can be moved to positions 20° and 50° at the programmed velocity in T1 mode. The following applies in this case:

```
$ASYNC_T1_FAST='B0011'
```

Only if velocity reduction is deactivated for all axes involved in an ASYPTP motion is the motion executed at the programmed velocity in T1 mode.

```
$ASYNC_T1_FAST='B0001'
```

Asynchronous external axes E1 and E2 are moved to positions 20° and 50° at the programmed velocity in T1 mode, because velocity reduction for external axis E2 is not deactivated.

8.4.2 $ASYNC_MODE

Description

Modes for asynchronous external axes

4 different modes for asynchronous external axes can be set using the variable in the directory KRC:\STEU\MADA\$CUSTOM.DAT.

The mode cannot be changed while the robot system is in operation.

Syntax

`$ASYNC_MODE=`*4-bit value*

Explanation of the syntax

Element	Description
4-bit value	Only bit 0 and bit 1 are used. ■ **Bit 0**: ASYPTP response in the Submit interpreter ■ **Bit 1**: ASYPTP response during block selection

Bit	Description
0	ASYPTP response in the Submit interpreter ■ **Bit 0 = 0**: Default mode ASYPTP is possible in the Submit interpreter, irrespective of the status of the robot interpreter. The return position of the asynchronous motions is saved, i.e. repositioning is not carried out in the Submit interpreter following asynchronous motions. In this mode, all external axes involved in an ASYPTP motion must be switched to asynchronous mode. ■ **Bit 0 = 1**: Mode 1 ASYPTP is only possible in the Submit interpreter if the robot interpreter is not active ($PRO_STATE <> #P_ACTIVE). The return position of the asynchronous motions is not saved, i.e. repositioning is carried out in the Submit interpreter following asynchronous motions. In this mode, the external axes involved in an ASYPTP motion do not have to be switched to asynchronous mode. In this mode, it is possible to execute individual motion sequences manually in PLC programs, e.g. manual welding with an electric motor-driven welding gun. The gun welds when the operator presses an assigned status key and is repositioned when the program is started.
1	ASYPTP response during block selection The response configured here also applies in the case of implicit block selection, e.g. for backward motion, reteaching a point, deleting a point or executing a program in the program run modes MSTEP and ISTEP. ■ **Bit 1 = 0**: Default mode In the case of a block selection, the system variable $ASYNC_AXIS is set to the value of $EX_AX_ASYNC. ■ **Bit 1 = 1**: Mode 2 In the case of a block selection, the system variable $ASYNC_AXIS is not changed.

Example 1

Default mode

```
$ASYNC_MODE='B0000'
```

Example 2

Default mode in the Submit interpreter and mode 2 for block selection

```
$ASYNC_MODE='B0010'
```

8.5 Decoupled external axes

8.5.1 $ASYNC_EX_AX_DECOUPLE

Description

Decoupling of external axes

The variable can be used in the directory KRC:\R1\MADA\$MACHINE.DAT for electrical and mechanical decoupling of external axes while the robot system is in operation, e.g. interchangeable servo guns (KUKA.ServoGun). The position and DSE data are saved for subsequent recoupling.

When the variable $ASYNC_EX_AX_DECOUPLE is defined, an advance run stop is triggered if the value changes. The new value is not saved until all synchronous and asynchronous motions have been completed and all axes are in position.

Precondition

- External axes to be decoupled must not be coupled to other axes, either mathematically or mechanically, nor may they be part of an external kinematic system.
- The system variable $ASYNC_OPT in the directory KRC:\STEU\MADA\$OPTION.DAT must be set to TRUE.
- The external axis mode must be set in the directory KRC:\R1\MADA\$MACHINE.DAT ($BRK_MODE, Bit3=1).

Characteristics

- Decoupled external axes can no longer be moved by the robot controller. All monitoring functions are deactivated.
- Decoupled external axes are automatically switched to asynchronous mode by setting the analog bit in the system variable $ASYNC_AXIS.

Decoupled external axes cannot be switched back to synchronous mode in the KRL program by means of the system variable $ASYNC_AXIS.

- The mastering of decoupled external axes is deleted and the system variable $AXIS_JUS is set to FALSE.
- Decoupled external axes can be mastered in the KRL program by means of the system variable $AXIS_ACT.

Syntax

`$ASYNC_EX_AX_DECOUPLE=`*n-bit value*

Explanation of the syntax

Element	Description
n-bit value	The value specifies which external axes are decoupled: ■ **Bit n = 0**: external axis is coupled. ■ **Bit n = 1**: external axis is decoupled. Bit value: ■ LSB: external axis E1 ■ MSB: external axis E6

Bit n	5	4	3	2	1	0
Axis	E6	E5	E4	E3	E2	E1

Example

```
$ASYNC_EX_AX_DECOUPLE='B0100'
```

External axis E3 is decoupled.

8.6 Permanently asynchronous external axes

8.6.1 $EX_AX_ASYNC

Description

Asynchronous external axes

External axes can be switched permanently to asynchronous mode and removed from the display using the variable in the directory KRC:\R1\MADA\$MACHINE.DAT.

The axis position of external axes is contained in every taught point (E6POS). Asynchronous external axes are not moved when a taught point is addressed, e.g. PTP XP1.

If external axes are switched to asynchronous mode using this variable, they cannot be switched back to synchronous mode by means of KRL statements.

Syntax

`$EX_AX_ASYNC=`*n-bit value*

Explanation of the syntax

Element	Description
n-bit value	The value specifies which external axes are switched to asynchronous mode: - **Bit n = 0**: external axis can be moved as synchronous or as asynchronous axis. - **Bit n = 1**: external axis can only be moved as an asynchronous axis. Bit value: - LSB: external axis E1 - MSB: external axis E6

Bit n	5	4	3	2	1	0
Axis	E6	E5	E4	E3	E2	E1

Example

3-axis positioner

```
$EX_AX_ASYNC='B0001'
...
PTP XP1
```

External axis E1 can only be moved asynchronously with ASYPTP. External axes E2 and E3 can be moved synchronously or asynchronously. If point P1 is addressed with a PTP motion, external axis E1 is not moved.

9 Programming

9.1 Programming motions for external axes

Description

It is possible to switch between synchronous and asynchronous motions of the external axes within a program. Whether an external axis is programmed as synchronous or asynchronous depends on the specific task.

Example

Welding system with one robot and 2 two-axis positioners

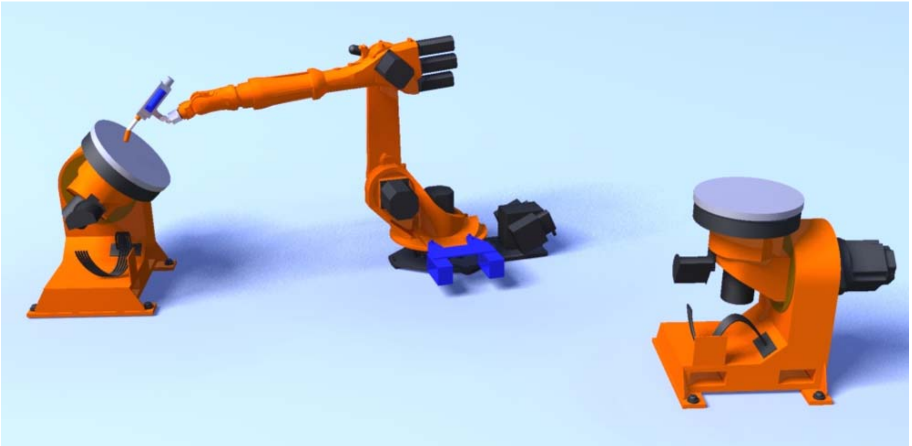


Fig. 9-1: Example: welding system

A robot welds workpieces that are loaded manually into 2 different two-axis positioners. One two-axis positioner moves the workpiece to a convenient welding position during the welding operation. While the robot is welding, the other two-axis positioner is moved to a convenient position so that the operator can exchange the workpiece.

The motion of the two-axis positioner at which welding is being carried out must be coupled to that of the robot, both synchronously and mathematically:

- The position of the two-axis positioner must be adapted to every welding step.
- The relative positions of the workpiece and robot must be defined at all times to exclude the possibility of collisions.
- The weld seam is to be programmed with weld velocity.

The two-axis positioner for the workpiece change must be moved asynchronously, independently of the robot:

- The two-axis positioner must be moved manually by means of a separate operating panel.
- The asynchronous motions must be enabled by the KRL program and are not coordinated with it.

  As long as the robot is working at the two-axis positioner, the positioner cannot be moved manually. Only when the robot has finished working and moved to the other two-axis positioner are the uncoordinated motions enabled by the KRL program.

9.2 Programming synchronous external axes

Description

The axis position of external axes is contained in every taught point (E6POS).

If positions are taught relative to a static base, the robot and external axis motions are synchronized, but without being mathematically coupled. The robot calculates its motion path independently of the position of the external axes.

If positions are taught relative to a moving base, the robot and external axis motions are synchronized and mathematically coupled. The robot calculates its motion path in relation to the position of the kinematic system.

9.2.1 Programming a mathematically coupled motion

Precondition

- Program is selected.
- Operating mode T1 or T2
- Root point of the kinematic system has been calibrated.

Procedure

1. Program the motion with an inline form.
2. To activate mathematical coupling, select the offset base in the option window **Frames** as the base to which the robot motion is relative.

The coordinates of an offset base are saved as BASE_DATA[17...22].

9.3 Programming asynchronous external axes

Overview

The following system variables and KRL instructions are available for programming asynchronous external axes:

System variable	Description
$ASYNC_AXIS	(>>> 9.3.1 "$ASYNC_AXIS" page 94)
$ASYNC_FLT	(>>> 9.3.4 "$ASYNC_FLT" page 97)
$ASYNC_STATE	(>>> 9.3.8 "$ASYNC_STATE" page 98)
$OV_ASYNC	(>>> 9.3.3 "$OV_ASYNC" page 96)

KRL instruction	Description
ASYCANCEL	(>>> 9.3.7 "ASYCANCEL" page 98)
ASYCONT	(>>> 9.3.6 "ASYCONT" page 97)
ASYPTP	(>>> 9.3.2 "ASYPTP" page 95)
ASYSTOP	(>>> 9.3.5 "ASYSTOP" page 97)

9.3.1 $ASYNC_AXIS

Description

The variable can be used to switch external axes to asynchronous or synchronous mode in the KRL program. Mechanically coupled external axes must always be switched to asynchronous mode together.

When the variable $ASYNC_AXIS is defined, an advance run stop is triggered if the value changes. The new value is not saved until all synchronous and asynchronous motions have been completed and all axes are in position.

This variable must not be used in the Submit interpreter or in an interrupt program.

Axes of a ROBROOT kinematic system and axes of a mathematically coupled BASE kinematic system cannot be switched to asynchronous mode.

Syntax

`$ASYNC_AXIS=`*n-bit value*

Explanation of the syntax

Element	Description
n-bit value	The value specifies which external axes are switched to synchronous or asynchronous mode: ■ **Bit n = 0**: external axis is switched to synchronous mode. Precondition: □ External axis is not permanently switched to asynchronous mode. ($EX_AX_ASYNC) ■ **Bit n = 1**: external axis is switched to asynchronous mode. Precondition: □ Asynchronous external axes are enabled. ($ASYNC_OPT=TRUE) □ Mathematical coupling is canceled. **Note**: Following a reset, asynchronous external axes are automatically switched back to synchronous mode. Bit value: ■ LSB: external axis E1 ■ MSB: external axis E6

Bit n	5	4	3	2	1	0
Axis	E6	E5	E4	E3	E2	E1

Example

```
PTP P10 VEL = 100% PDAT50 Tool[1]:Pen Base[17]:DKP400
PTP P11 VEL = 100% PDAT5 Tool[1]:Pen Base[0]
$ASYNC_AXIS = 'B0100'
```

The mathematical coupling is canceled by programming a motion block with a static base. External axis E3 is switched to asynchronous mode.

9.3.2 ASYPTP

Description

Asynchronous coordinated motions

The KRL instruction can be used to program coordinated motions of asynchronous external axes.

The instruction is executed in the advance run. There may be a maximum of 3 ASYPTP motions in the motion buffer. The first ASYPTP motion must be completed when the fourth ASYPTP motion is planned. If necessary, successive ASYPTP motions must therefore be synchronized.

There are 2 ways of synchronizing ASYPTP motions:

- by assigning to $ASYNC_AXIS with modification of the value of this variable
- by checking the state with $ASYNC_STATE

ASYPTP motions cannot be approximated.

ASYPTP may be used in the Submit interpreter or in an interrupt program. The response of ASYPTP in the Submit interpreter is configured with the system variable $ASYNC_MODE.

Syntax

`ASYPTP`{*Target position*}

OR

`ASYPTP X`*Target variable*

Explanation of the syntax

Element	Description
Target position	The asynchronous external axes specified in the target position are moved by means of axis-specific jogging.
Target variable	Data type: E6POS, E6AXIS The target variable contains a pre-taught position. The robot controller only access the position data of the asynchronous external axes.

Example 1

```
ASYPTP {E1 10.0, E3 20.0}
```

External axis E1 is moved to position 10.0° and external axis E3 is moved to position 20.0°.

Example 2

```
ASYPTP XP1
```

Asynchronous external axes are moved to the position saved in variable P1.

Example 3

```
DEF Program()
PTP HOME  Vel= 100 % DEFAULT
...
PTP P10
TRIGGER WHEN DISTANCE = 1 DELAY= -50 DO Async() PRIO = -1
PTP P11
...
PTP HOME  Vel= 100 % DEFAULT
END

DEF Async()
ASYPTP {E1 45.0}
END
```

50 s before point P11 is reached, external axis E1 is moved to position 45.0°.

9.3.3 $OV_ASYNC

Description

The velocity of asynchronous axes is not influenced by the program override (POV). The override for asynchronous coordinated motions must be set with $OV_ASYNC in the KRL program. The 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.
Exception: Velocity reduction is deactivated. ($ASYNC_T1_FAST=TRUE)

Syntax

`$OV_ASYNC=`*Override*

Explanation of the syntax

Element	Description
Override	Data type: INT; unit: % ■ **0 … 100**

Example

```
$OV_ASYNC=20
```

ASYPTP motions are carried out with 20% of the programmed velocity.

9.3.4 $ASYNC_FLT

Description

The variable sets the filter for asynchronous coordinated motions in the KRL program. This filter can be used to smooth ASYPTP motions.

Syntax

`$ASYNC_FLT=`*Filter value*

Explanation of the syntax

Element	Description
Filter value	Data type: INT; unit: ms ■ **0 ... 16** * interpolation cycle The value must be an integer multiple of the interpolation cycle (12 ms). Default: $DEF_FLT_PTP

Example

```
$ASYNC_FLT = 96
```

Filter value = 6 * interpolation cycle

9.3.5 ASYSTOP

Description

The KRL instruction can be used to stop asynchronous coordinated motions. The ASYCONT instruction can be used to resume the motions.

Syntax

`ASYSTOP=`*Axis number*

Explanation of the syntax

Element	Description
Axis number	Data type: INT ■ **0**: all asynchronous motions are stopped. ■ **1 … $EX_AX_NUM**: the number of the asynchronous external axis whose motion is to be stopped.

Example

```
ASYPTP {E2 40, E3 40}
WAIT SEC 2
ASYSTOP 2
```

An ASYPTP motion of external axes E2 and E3 is started. The ASYSTOP instruction only adresses external axis E2. Since external axis E3 is involved in the asynchronous motion, both external axes are stopped after a wait time of 2 s.

9.3.6 ASYCONT

Description

The KRL instruction can be used to resume asynchronous coordinated motions that have been stopped with ASYSTOP.

Syntax

`ASYCONT=`*Axis number*

Explanation of the syntax

Element	Description
Axis number	Data type: INT ■ **0**: all asynchronous motions are resumed. ■ **1 … $EX_AX_NUM**: the number of the asynchronous external axis whose motion is to be resumed.

Example

```
ASYPTP {E2 40, E3 40}
WAIT SEC 2
ASYSTOP 2
WAIT SEC 2
ASYCONT 0
```

An ASYPTP motion of external axes E2 and E3 is started. The ASYSTOP instruction only adresses external axis E2. Since external axis E3 is involved in the asynchronous motion, both external axes are stopped after a wait time of 2 s. The ASYPTP motion is resumed after a wait time of 2 s.

9.3.7 ASYCANCEL

Description

The KRL instruction can be used to cancel and delete asynchronous coordinated motions. Deleted motions cannot be resumed with ASYCONT.

Syntax

`ASYCANCEL=`*Axis_Number*

Explanation of the syntax

Element	Description
Axis_Number	Data type: INT ■ **0**: all asynchronous motions are canceled and deleted. The asynchronous motions of individual external axes cannot be canceled and deleted.

Example

```
ASYPTP {E2 40, E3 40}
WAIT SEC 2
ASYCANCEL 0
```

An ASYPTP motion of external axes E2 and E3 is started. The ASYPTP motion is canceled and deleted after a wait time of 2 s.

9.3.8 $ASYNC_STATE

Description

The variable can be used to poll the current state of asynchronous coordinated motions in the KRL program.

Syntax

`$ASYNC_STATE=`*State*

Explanation of the syntax

Element	Description
State	Data type: ENUM ■ **#BUSY**: asynchronous motions are active. ■ **#CANCELLED**: there are no active or stopped asynchronous motions. The last asynchronous motion was canceled with ASYCANCEL. ■ **#IDLE**: there are no active or stopped asynchronous motions. The last asynchronous motion was completed and not canceled with ASYCANCEL. ■ **#PEND**: asynchronous motions were stopped with ASYSTOP.

Example

```
ASYPTP {E2 45}
WHILE $ASYNC_STATE == #BUSY
$OUT[10] = TRUE
ENDWHILE
$OUT[10] = FALSE
```

An ASYPTP motion of external axis E2 is started. An output is set during the motion, e.g. to activate a warning lamp. The output is reset when the ASYPTP motion is completed.

10 Example

10.1 Transformation for DKP 400

In addition to the standard transformation, a simplified transformation is described here.

Standard transformation

1. The transformation starts at the root point of the kinematic system. The position of the root point is to be selected such that all required dimensions can be read from the technical drawing.

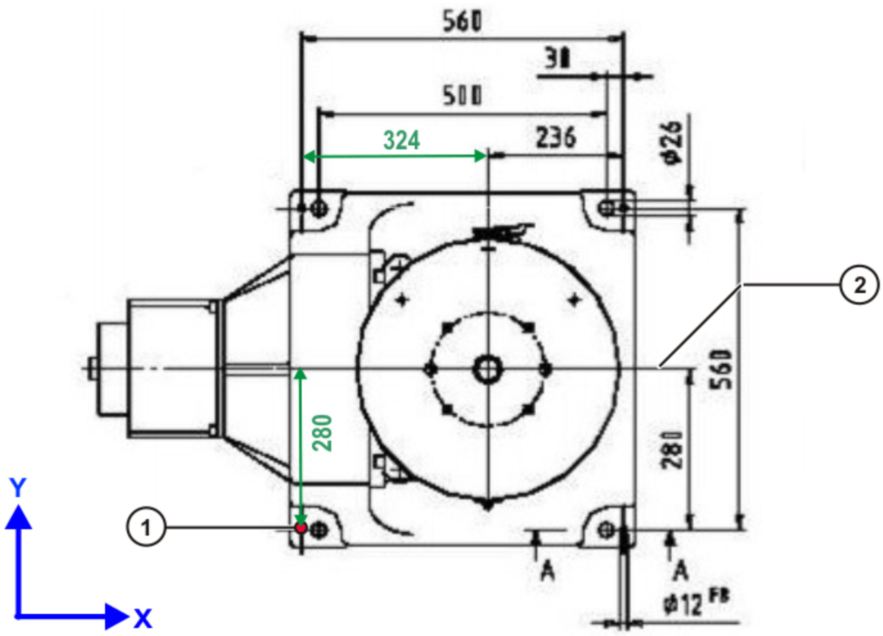


Fig. 10-1: Top view of DKP 400

1	Root point
2	Axis of symmetry

The root point is situated on the floor. The alignment of the coordinate system axes at the root point can be freely defined.

2. The joints and the rotational axes are defined.
3. Starting at the root point, transformations follow the structural design of the kinematic system from one joint to the next, up to the flange.

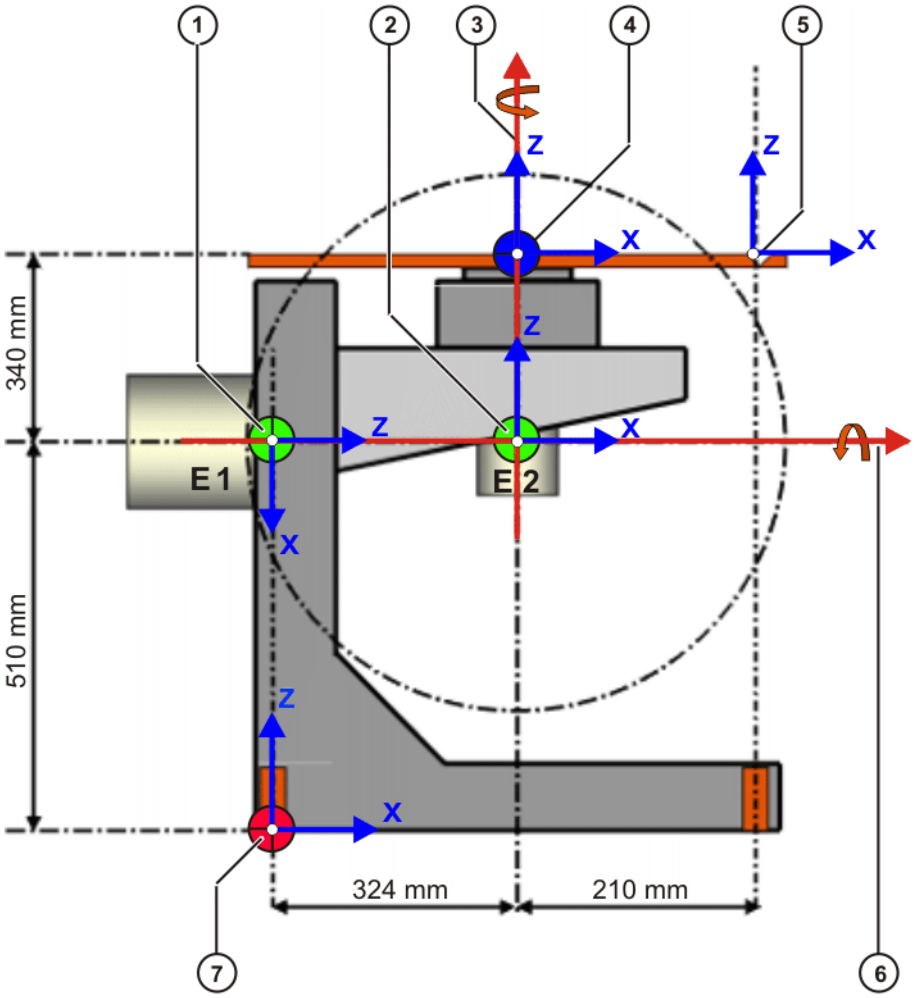


Fig. 10-2: Standard transformation for DKP 400

1	Joint E1	4	Flange center point
2	Joint E2	5	Reference pin
3	Rotational axis E2	6	Rotational axis E1
7	Root point		

```
$ET1_TA1KR={X 0.0,Y 280.0,Z 510.0,A 0.0,B 90.0,C 0.0}
$ET1_TA2A1={X 0.0,Y 0.0,Z 324.0,A 0.0,B -90.0,C 0.0}
$ET1_TA3A2={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
$ET1_TFLA3={X 0.0,Y 0.0,Z 340.0,A 0.0,B 0.0,C 0.0}
$ET1_TPINFL={X 210.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
```

Line	Description
1	1. Translation of 280 mm in the Y direction: from the root point to the axis of symmetry 2. Translation of 510 mm in the Z direction to joint E1 3. Rotation of 90° about angle B so that the positive Z direction coincides with rotational axis E1.
2	1. Translation of 324 mm in the Z direction: from joint E1 to joint E2 2. Rotation of -90° about angle B so that the positive Z direction coincides with rotational axis E2.
3	Since the DKP 400 has no third axis, no transformation is carried out here.

Line	Description
4	Translation of 340 mm in the Z direction: from joint E2 to the flange center point
5	Translation of 210 mm in the X direction: from the flange center point to the reference pin These data are no longer relevant for the transformation. They must be entered numerically as a tool. (>>> 6.2.3 "Assigning a TOOL coordinate system reference point" page 51)

Simplified transformation

The simplification is achieved by selecting a particularly suitable root point. Visible symmetries can be used here. The root point can be situated in any position, e.g. in the first joint.

These measures make it possible to reduce the reading of dimensions from the technical drawing. Fewer translations are required.

The rotations must always be carried out.

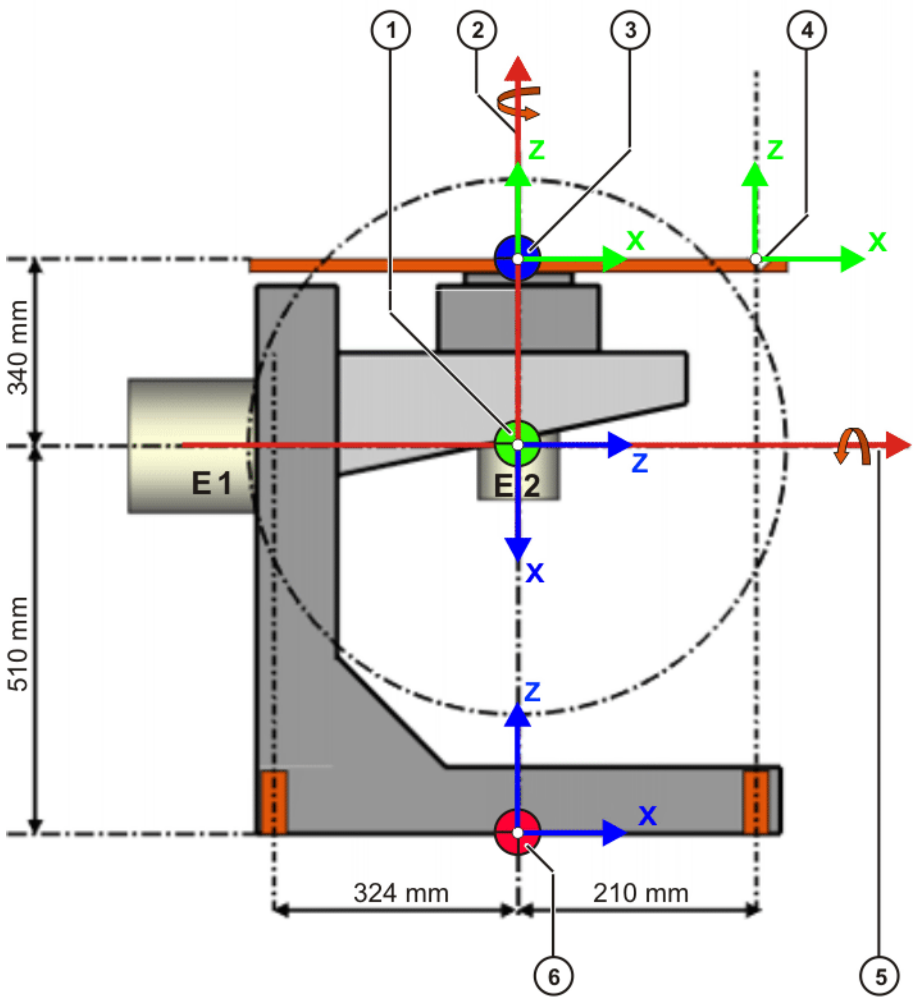


Fig. 10-3: Simplified transformation for DKP 400

1	Joint E1/E2	4	Reference pin
2	Rotational axis E2	5	Rotational axis E1
3	Flange center point	6	Root point

The root point is situated on the floor, on rotational axis E2. Joints E1 and E2 are located at the point of intersection of rotational axes E1 and E2.

```
1  $ET1_TA1KR={X 0.0,Y 0.0,Z 510.0,A 0.0,B 90.0,C 0.0}
2  $ET1_TA2A1={X -340.0,Y 0.0,Z 0.0,A 0.0,B -90.0,C 0.0}
3  $ET1_TA3A2={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
4  $ET1_TFLA3={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
5  $ET1_TPINFL={X 210.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
```

Line	Description
1	1. Translation of 510 mm in the Z direction: from the root point to joint E1/E2 2. Rotation of 90° about angle B so that the positive Z direction coincides with rotational axis E1.
2	1. Translation of -340 mm in the X direction: from joint E1/E2 to the flange center point 2. Rotation of -90° about angle B so that the positive Z direction coincides with rotational axis E2.
3	Since the DKP 400 has no third axis, no transformation is carried out here.
5	Translation of 210 mm in the X direction: from the flange center point to the reference pin These data are no longer relevant for the transformation. They must be entered numerically as a tool. (>>> 6.2.3 "Assigning a TOOL coordinate system reference point" page 51)

Machine data

```
$AXIS_TYPE[7]=3
$AXIS_TYPE[8]=3
$MAMES[7]=0.0
$MAMES[8]=0.0
$RAT_MOT_AX[7]={N -185,D 1}
$RAT_MOT_AX[8]={N -1185,D 10}
$RAT_MOT_ENC[7]={N 1,D 4}
$RAT_MOT_ENC[8]={N 1,D 4}
$DSECHANNEL[7]=7
$DSECHANNEL[8]=8
$PMCHANNEL[7]=121
$PMCHANNEL[8]=121
$SERVOFILE7[]="KSD_32_MG1_S7_0"
$SERVOFILE8[]="KSD_16_MH_S7"
$CURR_MAX[7]=32.0
$CURR_MAX[8]=16.0
$CURR_LIM[7]=100
$CURR_LIM[8]=100
$CURR_MON[7]=15.0
$CURR_MON[8]=9.2
$CURR_COM_EX[1]=100.0
$CURR_COM_EX[2]=100.0
$KT_MOT[7]=0.9743
$KT_MOT[8]=1.162
$KT0_MOT[7]=1.2
$KT0_MOT[8]=1.2
$RAISE_TIME[7]=600.0
$RAISE_TIME[8]=400.0
$RAISE_T_MOT[7]=6.80000019
$RAISE_T_MOT[8]=5.0
$VEL_AXIS_MA[7]=3000.0
$VEL_AXIS_MA[8]=2250.0
$AXIS_RESO[7]=4096
$AXIS_RESO[8]=4096
$RED_VEL_AXC[7]=6
$RED_VEL_AXC[8]=6
$RED_ACC_AXC[7]=10
$RED_ACC_AXC[8]=10
$RED_ACC_EMX[7]=400
$RED_ACC_EMX[8]=300
$VEL_AX_JUS[7]=0.129999995
$VEL_AX_JUS[8]=0.109999999
$L_EMT_MAX[7]=2.4000001
$L_EMT_MAX[8]=2.24000001
$LG_PTP[7]=0.5
$LG_PTP[8]=0.3
$LG_CP[7]=0.25
$LG_CP[8]=0.3
$DECEL_MB[7]=180.0
$DECEL_MB[8]=180.0
$G_VEL_PTP[7]=65.0
$G_VEL_PTP[8]=65.0
$G_VEL_CP[7]=80.0
$G_VEL_CP[8]=50.0
$I_VEL_PTP[7]=500.0
$I_VEL_PTP[8]=100.0
$I_VEL_CP[7]=500.0
$I_VEL_CP[8]=100.0
$APO_DIS_PTP[7]=90.0
$APO_DIS_PTP[8]=90.0
INT $BRK_MODE='B0101'
INT $BRK_DEL_EX=200
$IN_POS_MA[7]=0.100000001
$IN_POS_MA[8]=0.100000001
$SOFTN_END[7]=-90.0
$SOFTN_END[8]=-190.0
$SOFTP_END[7]=10.0
$SOFTP_END[8]=190.0
$AXIS_DIR[7]=1
$AXIS_DIR[8]=1
$INC_EXTAX[1]=0.00499999989
$INC_EXTAX[2]=0.00499999989
```

```
INT $EX_AX_NUM=2
DECL EX_KIN $EX_KIN={ET1 #EASYS,ET2 #NONE,ET3 #NONE,ET4 #NONE,ET5
#NONE,ET6 #NONE}
DECL ET_AX $ET1_AX={TR_A1 #E1,TR_A2 #E2,TR_A3 #NONE}
CHAR $ET1_NAME[20]
$ET1_NAME[]="DKP_400"
$ET1_TA1KR={X 0.0,Y 0.0,Z 510.0,A 0.0,B 90.0,C 0.0}
$ET1_TA2A1={X -340.0,Y 0.0,Z 0.0,A 0.0,B -90.0,C 0.0}
$ET1_TA3A2={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
$ET1_TFLA3={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
$AX_ENERGY_MAX[7]=700
$AX_ENERGY_MAX[8]=500
$BRK_ENERGY_MAX[7]=6855
$BRK_ENERGY_MAX[8]=1908
$BRK_COOL_OFF_COEFF[7]=38.0
$BRK_COOL_OFF_COEFF[8]=30.8
$BRK_TORQUE[7]=21.0
$BRK_TORQUE[8]=22.0
```

Further information about the machine data is contained in the Expert documentation “Machine Data – KR C2 – For KUKA System Software 5.5”.

10.2 Transformation for KL 1500-2

Description

The transformation starts at the root point of the linear unit. The position of the root point must be selected such that the required dimension can be read from the technical drawing, e.g. the height from the floor of the linear unit to the baseplate on which the robot is mounted.

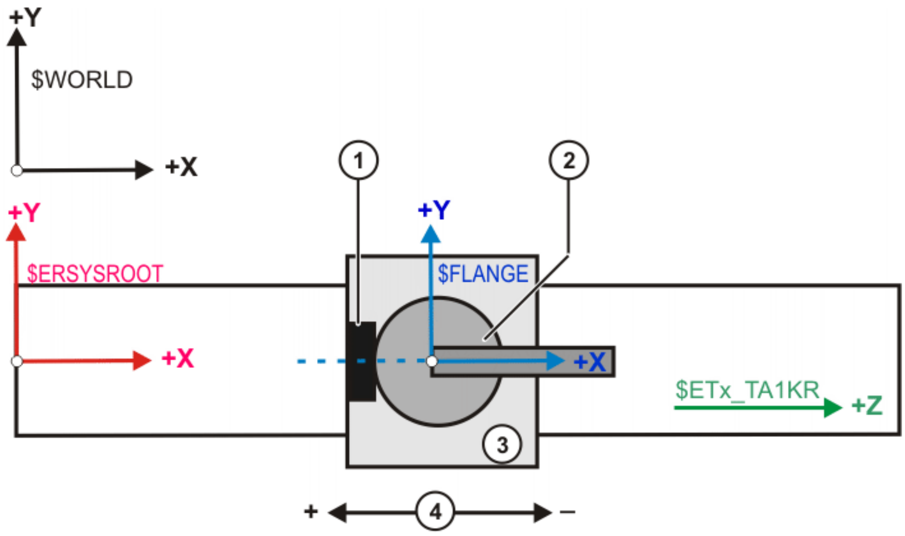


Fig. 10-4: Transformation for linear unit – top view

1	Connector panel	3	Carriage
2	Robot	4	Motion direction

The root point is situated on the floor of the linear unit, directly beneath the flange center point (not visible because of the top view). In the KL 1500-2, the height from the floor of the linear unit to the baseplate (flange) is 450 mm.

```
$ET1_TA1KR={X 0.0,Y 0.0,Z 450.0,A 0.0,B 90.0,C 0.0}
$ET1_TA2A1={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
$ET1_TA3A2={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
$ET1_TFLA3={X 0.0,Y 0.0,Z 0.0,A 0.0,B -90.0,C 0.0}
```

Line	Description
1	1. Translation of 450 mm in the Z direction: from the root point to the flange center point of the linear unit 2. Rotation of 90° about angle B so that the positive Z axis points in the direction of motion
2 … 3	Since the linear unit only has one axis, no transformation is carried out here.
4	Rotation of -90° about angle B so that the X axis, starting at the connector panel, points in the positive direction: orientation of the robot at the flange of the linear unit.

Machine data

```
$AXIS_TYPE[7]=1
$MAMES[7]=0.0
$RAT_MOT_AX[7]={N -346,D 10}
$RAT_MOT_ENC[7]={N 1,D 4}
$DSECHANNEL[7]=7
$PMCHANNEL[7]=21
$SERVOFILE7[]="KSD_32_MB_S"
$CURR_MAX[7]=32.0
$CURR_LIM[7]=100
$CURR_MON[7]=12.8
$CURR_COM_EX[1]=100
$KT_MOT[7]=1.25
$KT0_MOT[7]=1.4
$RAISE_TIME[7]=900.0
$RAISE_T_MOT[7]=6.8
$VEL_AXIS_MA[7]=3000.0
$AXIS_RESO[7]=4096
$RED_VEL_AXC[7]=5
$RED_ACC_AXC[7]=10
$RED_ACC_EMX[7]=150
$VEL_AX_JUS[7]=0.9648
$L_EMT_MAX[7]=9.6
$LG_PTP[7]=0.5
$LG_CP[7]=0.6
$DECEL_MB[7]=600.0
$G_VEL_PTP[7]=75.0
$G_VEL_CP[7]=80.0
$I_VEL_PTP[7]=200.0
$I_VEL_CP[7]=200.0
$APO_DIS_PTP[7]=500.0
INT $BRK_MODE='B0101'
INT $BRK_DEL_EX=200
$IN_POS_MA[7]=1.5
$SOFTN_END[7]=10000.0
$SOFTP_END[7]=10000.0
$AXIS_DIR[7]=1
$INC_EXTAX[1]=0.006
INT $EX_AX_NUM=1
DECL EX_KIN $EX_KIN={ET1 #ERSYS,ET2 #NONE,ET3 #NONE,ET4 #NONE,ET5
#NONE,ET6 #NONE}
DECL ET_AX $ET1_AX={TR_A1 #E1,TR_A2 #NONE,TR_A3 #NONE}
CHAR $ET1_NAME[20]
$ET1_NAME[]="KL1500K_1"
FRAME $ET1_TA1KR={X 0.0,Y 0.0,Z 450.0,A 0.0,B 90.0,C 0.0}
FRAME $ET1_TA2A1={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
FRAME $ET1_TA3A2={X 0.0,Y 0.0,Z 0.0,A 0.0,B 0.0,C 0.0}
FRAME $ET1_TFLA3={X 0.0,Y 0.0,Z 0.0,A 0.0,B -90.0,C 0.0}
$AX_ENERGY_MAX[7]=3079
$BRK_ENERGY_MAX[7]=4600
$BRK_COOL_COEFF[7]=25.5
$BRK_TORQUE[7]=20.0
$AXIS_JERK[7]=11.64
```

Further information about the machine data is contained in the Expert documentation “Machine Data – KR C2 – For KUKA System Software 5.5”.

11 KUKA Service

11.1 Requesting support

Introduction

The KUKA Robot Group documentation offers information on operation and provides assistance with troubleshooting. For further assistance, please contact your local KUKA subsidiary.

Faults leading to production downtime should be reported to the local KUKA subsidiary within one hour of their occurrence.

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

11.2 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

Marand Precision Engineering Pty. Ltd. (Agency)
153 Keys Road
Moorabbin
Victoria 31 89
Australia
Tel. +61 3 8552-0600
Fax +61 3 8552-0605
robotics@marand.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 Flexible Manufacturing Equipment (Shanghai) Co., Ltd.
Shanghai Qingpu Industrial Zone
No. 502 Tianying Rd.
201712 Shanghai
P.R. China
Tel. +86 21 5922-8652
Fax +86 21 5922-8538
Franz.Poeckl@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, Private Limited
621 Galleria Towers
DLF Phase IV
122 002 Gurgaon
Haryana
India
Tel. +91 124 4148574
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.
Ogikubo TM Building 7F
5-26-13 Ogikubo, Suginami-ku
Tokio
167-0051
Japan
Tel. +81 353 47-9831
Fax +81 353 47-9835
kuka.co.jp

Korea

KUKA Robot Automation Korea Co. Ltd.
4 Ba 806 Sihwa Ind. Complex
Sung-Gok Dong, Ansan City
Kyunggi Do
425-110
Korea
Tel. +82 31 496-9937 or -9938
Fax +82 31 496-9939
info@kukakorea.com

Malaysia

KUKA Robot Automation Sdn Bhd
South East Asia Regional Office
No. 24, Jalan TPP 1/10
Taman Industri Puchong
47100 Puchong
Selangor
Malaysia
Tel. +60 3 8061-0613 or -0614
Fax +60 3 8061-7386
info@kuka.com.my

Mexico

KUKA de Mexico S. de R.L. de C.V.
Rio San Joaquin #339, Local 5
Colonia Pensil Sur
C.P. 11490 Mexico D.F.
Mexico
Tel. +52 55 5203-8407
Fax +52 55 5203-8148
info@kuka.com.mx

Norway

KUKA Sveiseanlegg + Roboter
Bryggeveien 9
2821 Gjövik
Norway
Tel. +47 61 133422
Fax +47 61 186200
geir.ulsrud@kuka.no

Austria

KUKA Roboter Austria GmbH
Vertriebsbüro Österreich
Regensburger Strasse 9/1
4020 Linz
Austria
Tel. +43 732 784752
Fax +43 732 793880
office@kuka-roboter.at
www.kuka-roboter.at

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
Riedstr. 7
8953 Dietikon
Switzerland
Tel. +41 44 74490-90
Fax +41 44 74490-91
info@kuka-roboter.ch
www.kuka-roboter.ch

Spain

KUKA Sistemas de Automatización S.A.
Pol. Industrial
Torrent de la Pastera
Carrer del Bages s/n
08800 Vilanova i la Geltrú (Barcelona)
Spain
Tel. +34 93 814-2353
Fax +34 93 814-2950
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.
136, Section 2, Huanjung E. Road
Jungli City, Taoyuan
Taiwan 320
Tel. +886 3 4371902
Fax +886 3 2830023
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

Hungary KUKA Robotics Hungaria Kft.
Fö 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
$ASYNC_AX... 86
$ASYNC_AXIS 89, 94
$ASYNC_EX_AX_DECOUPLE 89
$ASYNC_FLT 97
$ASYNC_MODE 88
$ASYNC_OPT 85, 95
$ASYNC_STATE 98
$ASYNC_T1_FAST 87
$BASE_C 40
$BRK_MODE 18
$CURR_MON 78
$DECEL_MB 75
$ET1_AX 61
$ET1_NAME 62
$ET1_TA1KR 62
$ET1_TA2A1 63
$ET1_TA3A2 63
$ET1_TFLA3 63
$EX_AX_ASYNC 89, 90, 95
$EX_KIN 61
$EXT_AXIS 85
$G_VEL_CP 66
$G_VEL_PTP 66
$I_VEL_CP 68
$I_VEL_PTP 68
$LG_CP 70
$LG_PTP 70
$OV_ASYNC 96
$RAISE_TIME 73
$RED_ACC_EMX 74
$ROBROOT_C 41
$ZUST_ASYNC 86

Numbers
2006/95/EC 37
89/336/EEC 37
97/23/EC 37
98/37/EC 37

A
Accessories 21
Applied norms and regulations 37
ASYCANCEL 98
ASYCONT 97
Asynchronous 12
ASYPTP 85, 95
ASYSTOP 97
AUT 24, 25
AUT EXT 24, 25
Automatic 24, 25
Automatic External 24, 25
Automatic mode 35
Axis configurator 80
Axis range 21
Axis range limitation 30
Axis range monitoring 30

B
BASE kinematic system 9, 39
Brake defect 33
Braking distance 21, 27
Braking, path-maintaining 22, 26
Braking, path-oriented 22, 26

C
Calibration 49
Calibration, offset base 52
Calibration, root point, kinematic system 50
Calibration, TOOL kinematic system 54
CE mark 20
Cleaning work 36
Configuration 57
Configuration, external axis system 57
Connecting cables 20
Connection for external enabling switch 29
Controller parameters, modifying 79
Coordinate systems 39
Coordinated 12
Counterbalancing system 36
Cur. tool/base (menu item) 43

D
Danger zone 21
Declaration of conformity 20
Declaration of incorporation 19, 20
Decommissioning 37
Designated use 20
Dial gauge 46, 48
Disposal 37
Documentation, robot system 7
Drives OFF 24
Drives ON 24
DSE 7

E
EC declaration of conformity 20
Electronic mastering tool 46, 48
EMC Directive 20, 37
EMERGENCY STOP 26, 31
EMERGENCY STOP button 28, 30
EMERGENCY STOP function 36
EMT 46, 48
EN 55011 38
EN 60204-1 38
EN 61000-4-4 38
EN 61000-6-2 38
EN 61000-6-4 38
EN 614-1 38
EN 61800-3 38
EN 954-1 38
EN ISO 10218-1 38
EN ISO 12100-1 38
EN ISO 12100-2 38
EN ISO 13732-1 38
EN ISO 13850 37

Enabling 24
Enabling switch 30
Enabling switches 28, 29
ERSYSROOT coordinate system 40
ESC 24
Example 101
External axes 21
External axis 22
External EMERGENCY STOP 24
External safeguards 31

F
Faults 33
Firewall 35
FLANGE coordinate system 39
Function test 34
Fundamentals 9

G
Gauge pin 47
General safety measures 33
Guard interlock 27

H
Hazardous substances 37

I
Introduction 7
ISTEP 89

J
Jog mode 29, 30
Jogging 41

K
KCP 20, 21, 33
KCP coupler 31
Keyboard, external 33
Kinematic system type 9
KMC 7
KMC kinematic system 7
KSD 7
KUKA Customer Support 109
KUKA.HMI 7

L
Labeling 32
Liability 19
Linear unit 21
Local EMERGENCY STOP 24
Low Voltage Directive 20, 37

M
Machine data 18
Machine data, configuring 81
Machine data, modifying 78
Machine data, optimizing 65
Machinery Directive 20, 37
Maintenance 36
Manual High Velocity 24, 25
Manual Reduced Velocity 24, 25
Master/slave operation 13, 15
Mastering 45
Mastering methods 46
Mathematical coupling, activating 43
Mechanical axis range limitation 30
Mechanical end stops 29
Mechanical zero position 45
MFC3 7
MGU 7
Mode selector switch 24
Modifying controller parameters 79
Modifying machine data 78
Mouse, external 33
MSTEP 89

N
Network security 35
Numeric entry, external tool 56
Numeric entry, offset base 54
Numeric entry, root point, kinematic system 52

O
Offset base calibration 52
OFFSET coordinate system 39
Operating modes 24, 30
Operation 39
Operator 21, 23
Operator safety 24, 27, 30
Optimization 65
Options 21
Oscilloscope 65
Overload 33
Overview of the safety features 23

P
Panic position 28, 29
Path-maintaining 22, 26
Path-oriented 26
path-oriented 22
Personnel 22
Plant integrator 22
Positioner 21
Pre-mastering position 47
Pressure Equipment Directive 36, 37
Preventive maintenance work 36
Programming 35, 93

R
r.m.s. current, checking 78
Ramp-down braking 22, 26
RDC 7
Reduced velocity, program mode 30
Reference notch 47
Release device 31
Repair 36
Robot 20, 27
Robot controller 20, 35
Robot system 19, 20, 22
ROBROOT coordinate system 39, 40
ROBROOT kinematic system 9, 40
ROOT coordinate system 39

Root point, kinematic system 50

S
Safety 19
Safety features 30
Safety fences 32
Safety gates 32
Safety instructions 19
Safety logic 24
Safety zone 22, 27
Safety, general 19
SBM2 7, 17
Service, KUKA Roboter 109
Servo file, creating 82
Servo file, loading 82
Setting 34
Short-circuit braking 26
Simulation 35
Softkeys 81
Software 21
Software limit switches 30
Start-up 34, 45
Status keys 80
STOP 0 22, 26
STOP 1 22, 26
STOP 2 22, 26
Stop reactions 26
Storage 37
Support request 109
Synchronous 11
System integrator 20, 22
System variables 85

T
T1 22, 24, 25
T2 22, 24, 25
Teach pendant 20
Terms used 7, 21
TOOL coordinate system 39, 51
TOOL kinematic system 10
Tool, external 54
Top-mounted cabinet 16, 21
Training program 7, 23
Transformation, BASE kinematic system 64
Transformation, ROBROOT kinematic system 65
Transport position 34
Transportation 34
Two-axis positioner 21

U
Uncoordinated 12
Use, contrary to designated use 19
Use, improper 19
User 21, 22

V
Virus protection 35

W
Warnings 19
Working envelope 21
Working range limitation 30
Workspace 27
WORLD coordinate system 39, 40