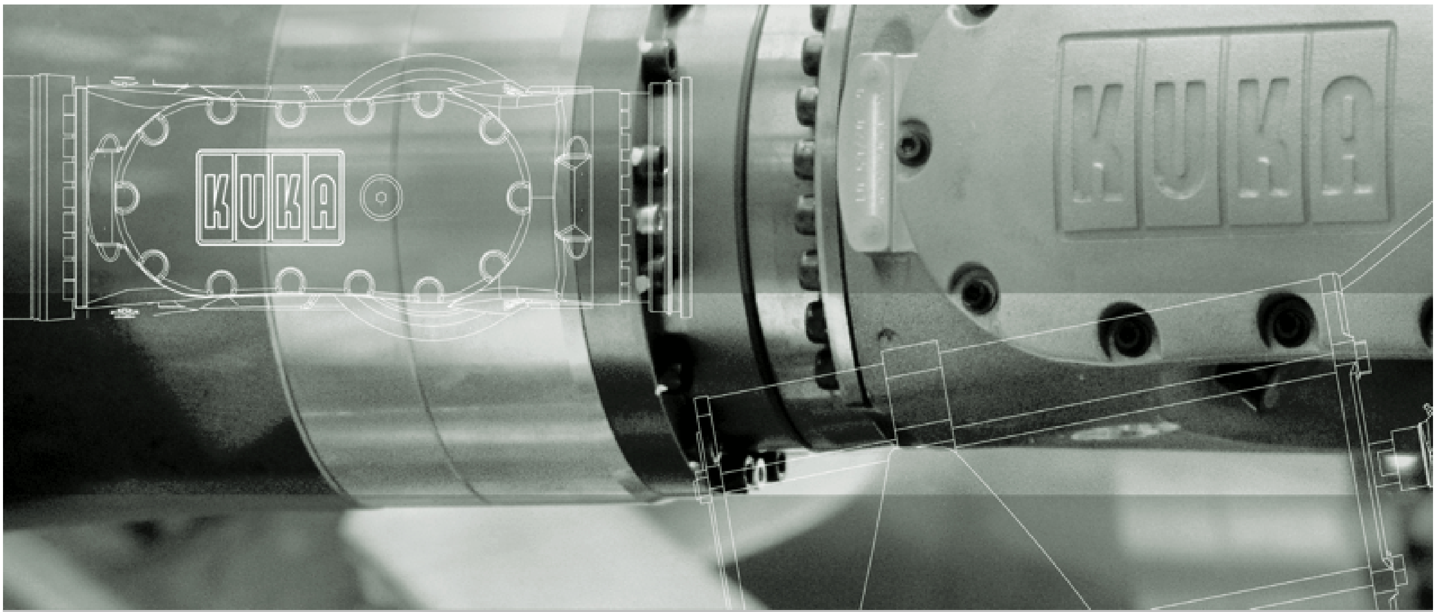


KUKA Robot Group

KUKA System Software

KUKA System Software 5.5

Operating and Programming Instructions for End Users

Issued: 15.04.2008 Version: KSS 5.5 END V1 en

KUKA Roboter GmbH
Zugspitzstraße 140
D-86165 Augsburg
Germany

KIM-PS5-DOC

Publication: Pub KSS 5.5 END en

Book structure: KSS 5.5 END V1.9

Label: KSS 5.5 END V1

Contents

1 Introduction

1.1 Target group

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

- Basic knowledge of the robot system

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 Trademarks

Windows is a trademark of Microsoft Corporation.

2 Product description

2.1 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

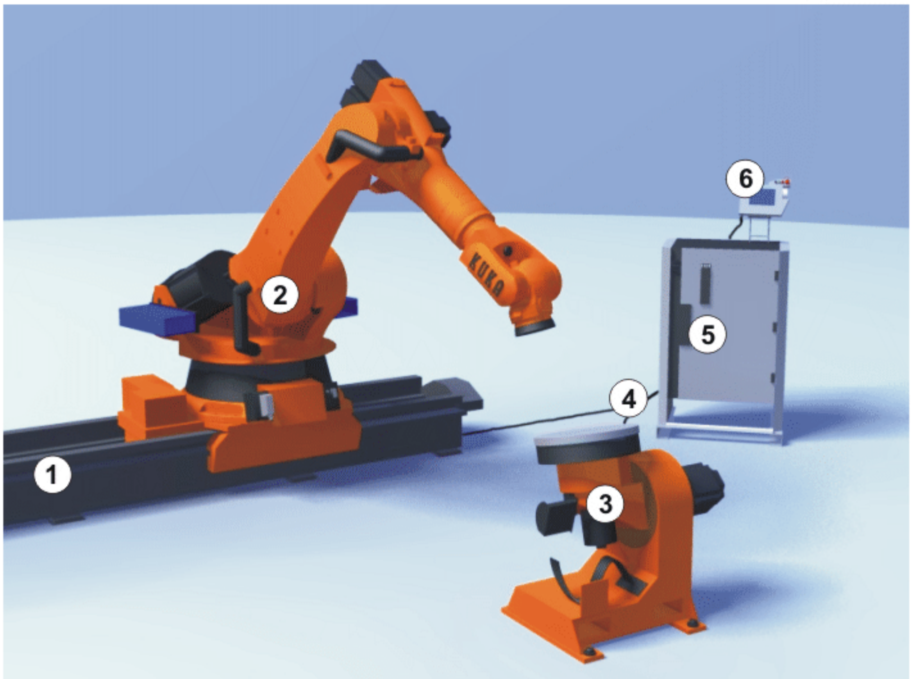


Fig. 2-1: Example of a robot system

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

2.2 Overview of the software components

Overview

The following software components are used:

- KUKA System Software 5.5
- Windows XP embedded 2.x incl. Service Pack 2

It is not possible to upgrade from Windows Service Pack 1 to Windows Service Pack 2.

2.3 Overview of KUKA System Software (KSS)

Description

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

- Path planning
- I/O management
- Data and file management
- etc.

Additional technology packages, containing application-specific instructions and configurations, can be installed.

KUKA.HMI

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

Features:

- User management
- Program editor
- KRL (KUKA Robot Language)
- Inline forms for programming
- Message display
- Configuration window
- Online help
- etc.

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

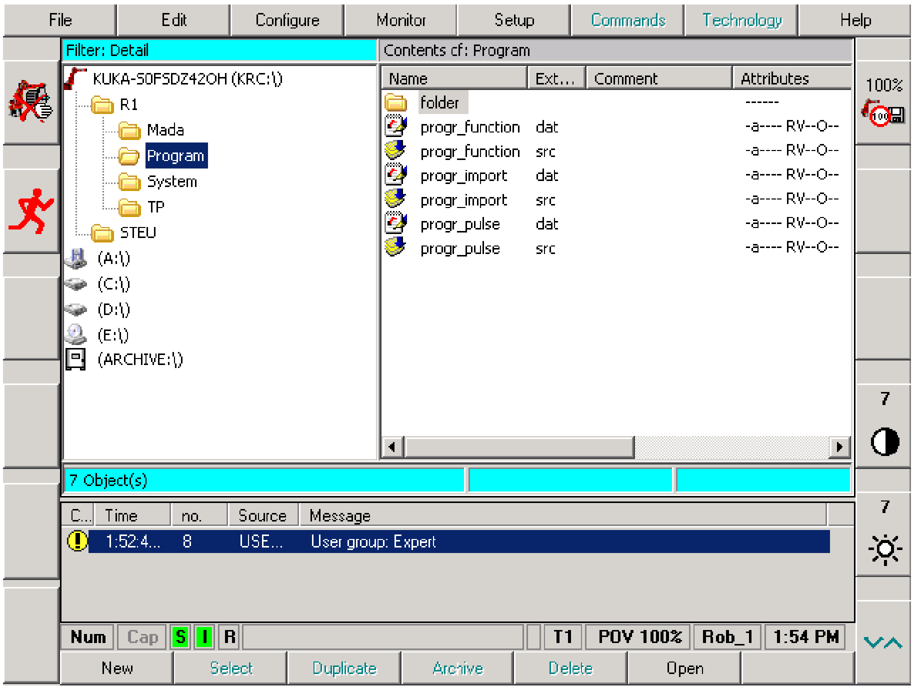


Fig. 2-2: KUKA.HMI user interface

3 Safety

3.1 General

3.1.1 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 robot system must be rectified immediately.

Safety information

Safety information cannot be held against the KUKA Robot Group. 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 the KUKA Robot Group. Additional components (tools, software, etc.), not supplied by KUKA Robot Group, 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.

3.1.2 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.

3.1.3 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

3.1.4 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.

3.1.5 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

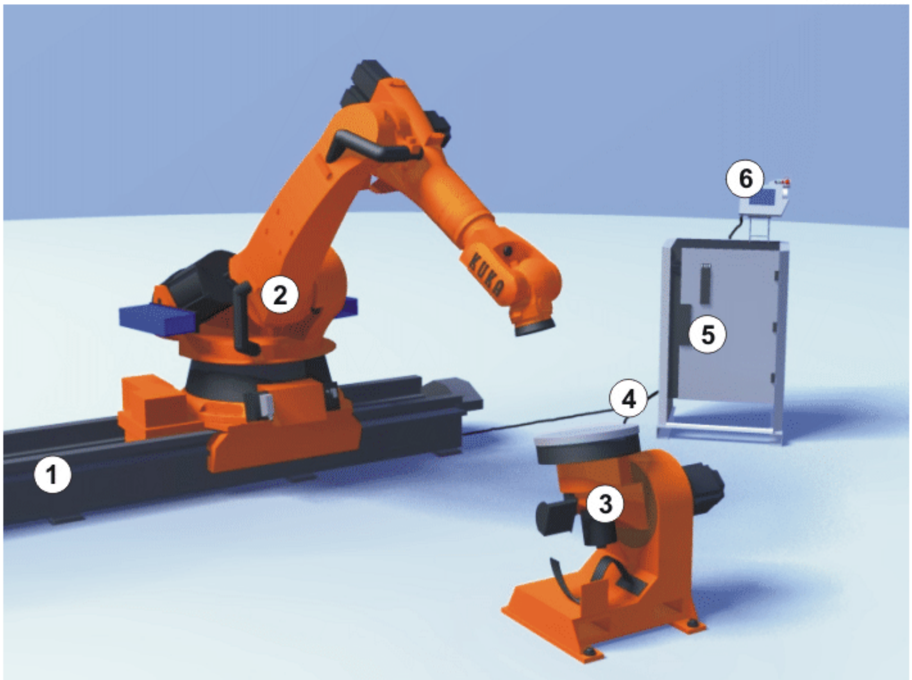


Fig. 3-1: Example of a robot system

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

3.1.6 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 work-space. 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 (System 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

3.2 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.

3.3 Safety features of the robot system

3.3.1 Overview of the safety features

The following safety features are provided with the robot system:

- Operator safety
- EMERGENCY STOP pushbutton
- Enabling switches
- 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.

3.3.2 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

3.3.3 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.

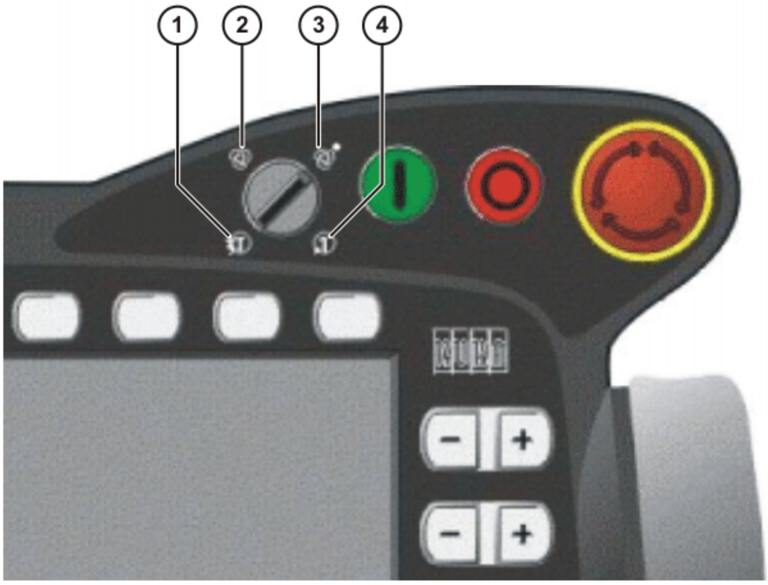


Fig. 3-2: Mode selector switch

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

Operating mode	Use	Velocities
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

3.3.4 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.

3.3.5 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.

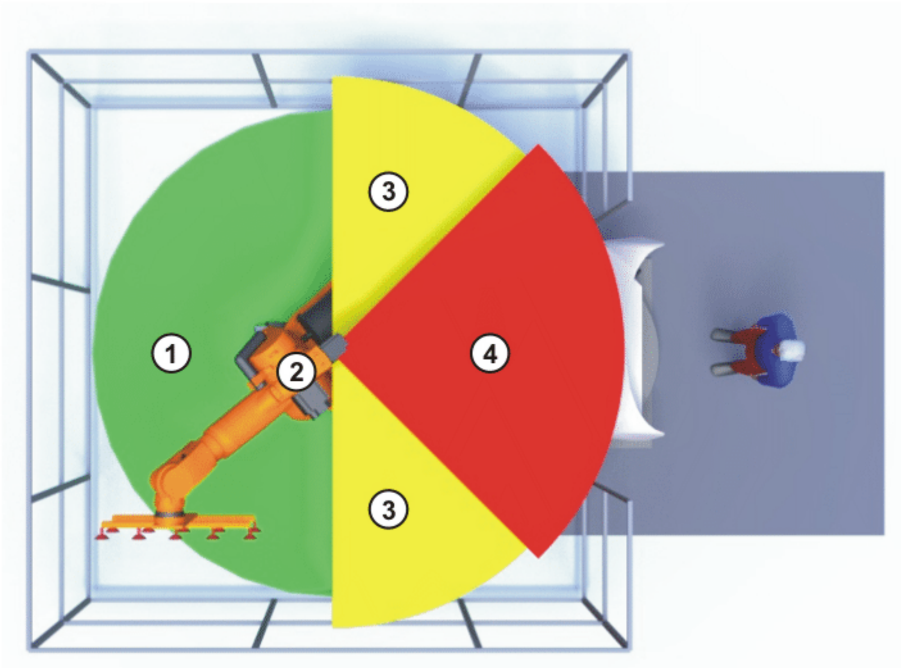


Fig. 3-3: Example of axis range A1

1	Workspace	3	Braking distance
2	Robot	4	Safety zone

3.3.6 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.

3.3.7 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. 3-4: EMERGENCY STOP button on the KCP

1 EMERGENCY STOP button

3.3.8 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. 3-5: Enabling switches on the KCP

1 - 3 Enabling switches

3.3.9 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 the KUKA Robot Group.

3.3.10 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.

3.3.11 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. The KUKA Robot Group must be consulted before the robot system is put back into operation (>>> 10 "KUKA Service" page 157). 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 Robot Group.

3.3.12 Software limit switches

The axis ranges of all robot 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 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.

3.3.13 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 switches	active	active	-	-
Reduced velocity in program mode	active	-	-	-
Jog mode	active	active	-	-
Software limit switches	active	active	active	active

3.3.14 Mechanical axis range limitation (option)

Most robots can be fitted with mechanical axis range limitation in main axes A 1 to A 3. 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.

3.3.15 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 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.

3.3.16 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.

3.3.17 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.

3.3.18 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.

3.3.19 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.

3.4 Safety measures

3.4.1 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.
- Keep a record of the faults.
- Eliminate the fault and carry out a function test.

3.4.2 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.

3.4.3 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.

3.4.4 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.

3.4.5 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.

3.4.6 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.

3.4.7 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.

3.4.8 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 the KUKA Robot Group 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 up to 5 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.

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.

Inspection intervals and inspection personnel:

Category	Inspection before commissioning*	Internal inspection (≤ 3 years)	Strength test (≤ 10 years)
II	Approved inspection agency	Competent person	Competent person
III	Approved inspection agency	Approved inspection agency	Approved inspection agency

*Inspection by KUKA Robot Group

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.

3.4.9 Decommissioning, storage and disposal

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

3.5 Applied norms and regulations

Name	Definition	Edition
73/23/EEC	Low Voltage Directive: Council Directive of 19 February 1973 on the harmonization of the laws of Member States relating to electrical equipment designed for use within certain voltage limits	1993
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 418	Safety of machinery: EMERGENCY STOP equipment, functional aspects; principles for design	1993
EN 563	Safety of machinery: Temperatures of touchable surfaces - Ergonomics data to establish temperature limit values for hot surfaces	2000
EN 614-1	Safety of machinery: Ergonomic design principles – Part 1: Terms and general principles	1995
EN 775	Industrial robots: Safety	
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	1998
EN 61000-4-4	Electromagnetic compatibility (EMC): Part 4-4: Testing and measurement techniques - Electrical fast transient/burst immunity test	2002

Name	Definition	Edition
EN 61000-4-5	Electromagnetic compatibility (EMC): Part 4-5: Testing and measurement techniques; Surge immunity test	2001
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

4 Operation

4.1 KCP teach pendant

4.1.1 Front view

Function

The KCP (KUKA Control Panel) is the teach pendant for the robot system. The KCP has all the functions required for operating and programming the robot system.

Overview

Fig. 4-1: Front view of KCP

1	Mode selector switch	10	Numeric keypad
2	Drives ON	11	Softkeys
3	Drives OFF / SSB GUI	12	Start backwards key
4	EMERGENCY STOP button	13	Start key
5	Space Mouse	14	STOP key
6	Right-hand status keys	15	Window selection key
7	Enter key	16	ESC key
8	Arrow keys	17	Left-hand status keys
9	Keypad	18	Menu keys

Description

Element	Description
Mode selector switch	(>>> 4.12 "Operating modes" page 50)
Drives ON	Switches the robot drives on.
Drives OFF **SSB GUI**	Switches the robot drives off. Only with Shared Pendant (KCP for KUKA.RoboTeam): SSB GUI calls the user interface of the Safety Selection Board
EMERGENCY STOP pushbutton	Stops the robot in hazardous situations. The EMERGENCY STOP button locks itself in place when it is pressed.
Space Mouse	Jogs the robot.

Element	Description
Right-hand status keys	(>>> 4.2.1 "Status keys, menu keys, softkeys" page 41)
Enter key	The Enter key is used to close an active window or inline form. Changes are saved.
Arrow keys	The arrow keys are used to jump from element to element in the user interface. **Note:** If an element cannot be accessed using the arrow keys, use the TAB key instead.
Keypad	(>>> 4.1.2 "Keypad" page 36)
Numeric keypad	(>>> 4.1.3 "Numeric keypad" page 37)
Softkeys	(>>> 4.2.1 "Status keys, menu keys, softkeys" page 41)
Start backwards key	The Start backwards key is used to start a program backwards. The program is executed step by step.
Start key	The Start key is used to start a program.
STOP key	The STOP key is used to stop a program that is running.
Window selection key	The window selection key is used to toggle between the main, option and message windows. The selected window is indicated by a blue background.
ESC key	The ESC key is used to abort an action on the user interface.
Left-hand status keys	(>>> 4.2.1 "Status keys, menu keys, softkeys" page 41)
Menu keys	(>>> 4.2.1 "Status keys, menu keys, softkeys" page 41)

4.1.2 Keypad

Fig. 4-2: Keypad

Key	Description
NUM	NUM is used to toggle between the numeric function and the control function of the numeric keypad. The status bar indicates which of the functions is active (>>> 4.2.4 "Status bar" page 44).
ALT	ALT is used in keyboard shortcuts. The key remains activated for one keystroke. In other words, it does not need to be held down.

Key	Description
SHIFT	SHIFT is used to switch between upper-case and lower-case letters. The key remains activated for one keystroke. In other words, it does not need to be held down to type one upper-case letter. To type several upper-case characters, the SHIFT key must be held down. SYM+SHIFT switches to permanent upper-case typing. The status bar indicates whether upper-case or lower-case typing is active (>>> 4.2.4 "Status bar" page 44).
SYM	SYM must be pressed to enter the secondary characters assigned to the letter keys, e.g. the "#" character on the "E" key. The key remains activated for one keystroke. In other words, it does not need to be held down.

4.1.3 Numeric keypad

Fig. 4-3: Numeric keypad

The NUM key in the keypad is used to toggle between the numeric function and the control function of the numeric keypad. The status bar indicates which of the functions is active. (>>> 4.2.4 "Status bar" page 44)

Key	Control function
INS (0)	Switches between insert and overwrite mode.
DEL (.)	Deletes the character to the right of the cursor.
<-	Deletes the character to the left of the cursor.
END (1)	Positions the cursor to the end of the line in which it is currently situated.
CTRL (2)	Used in keyboard shortcuts.
PG DN (3)	Scrolls one screen towards the end of the file.
(4)	----
UNDO (5)	Undoes the last input. (This function is not currently supported.)
TAB (6)	Positions the focus or the cursor on the next user interface element. **Note:** If an element cannot be accessed using the TAB key, use the arrow keys instead.
HOME (7)	Positions the cursor to the start of the line in which it is currently situated.
LDEL (8)	Deletes the line in which the cursor is positioned.
PG UP (9)	Scrolls one screen towards the start of the file.

4.1.4 Rear view

Overview

Fig. 4-4: Rear view of KCP

1	Rating plate	4	Enabling switch
2	Start key	5	Enabling switch
3	Enabling switch		

Description

Element	Description
Rating plate	KCP rating plate
Start key	The Start key is used to start a program.
Enabling switch	The enabling switch has 3 positions: ■ Not pressed ■ Center position ■ Panic position The enabling switch must be held in the **center position** in operating modes T1 and T2 in order to be able to jog the robot. In the operating modes Automatic and Automatic External, the enabling switch has no function.

4.1.5 KCP coupler

The KCP coupler is an optional component of the KR C2 edition2005 robot controller. The KCP coupler allows the KCP to be connected and disconnected with the robot controller running.

If the robot is controlled via a PLC and the KCP is not used, it may be useful to uncouple the KCP. This is particularly the case in systems with large numbers of robots: The unused KCPs can be removed from the system, thereby improving the transparency of the system.

4.1.5.1 Planning the KCP coupler option

Visualization

If the robot controller is operated with a detachable KCP, the following system variables must be visualized:

- $Mode_T1 (T1 mode)
- $Mode_T2 (T2 mode)
- $Mode_Ext (External mode)
- $Mode_Aut (Automatic mode)
- $Notaus (Emergency Stop)
- $Pro_Act (program active)

The display can be configured using I/Os or a PLC. The system variables can be configured in the file: STEU/$MACHINE.DAT.

Warning!
If the KCP is disconnected, the system can no longer be deactivated by means of the E-STOP button on the KCP. An external E-STOP must be connected to interface X11 to prevent personal injury and material damage.

4.1.5.2 Display and operator control elements of the KCP coupler (optional)

Overview

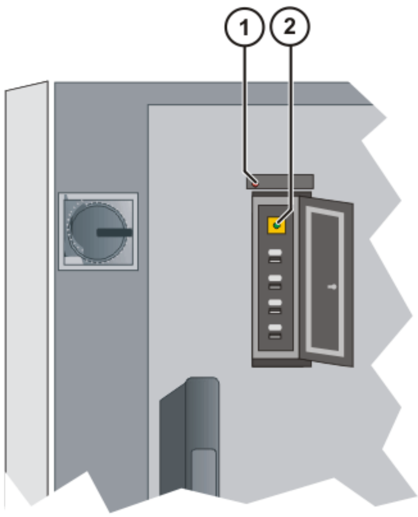


Fig. 4-5: KCP coupler LEDs and request button

1	Fault LED (red), KCP coupler
2	Request button with request LED (green)

4.1.5.3 Uncoupling the KCP

Procedure

1. Press the request button for at least 1 s.
   The green request LED flashes.
   The KCP is switched off (display goes dark).

Caution!
The KCP must not be disconnected without pressing the request button. If the KCP is disconnected without the request button being pressed, an EMERGENCY STOP is triggered.

2. Disconnect the KCP within 60 s.

Caution!
The KCP with EMERGENCY STOP is deactivated for the request time of 60 s. The EMERGENCY STOP on the KCP is not activated during this time.

3. The KCP must be removed from the system.

Caution!
The KCP must be removed from the system if it is not connected. The EMERGENCY STOP is not operational in this case.

4.1.5.4 Coupling the KCP

Preconditions

- The KCP variant to be coupled must be the same as that which was uncoupled.

Procedure

1. Set the operating mode on the KCP to the same operating mode as on the robot controller (the operating mode display is application-specific (>>> 4.1.5.1 "Planning the KCP coupler option" page 39)).

If the KCP is connected with the wrong operating mode selected, the robot controller switches to the operating mode set on the KCP.

2. Couple the KCP to the robot controller.

   The request LED flashes quickly.

   Once coupling has been completed, the request LED lights up and the KCP display shows the user interface. The robot controller can once again be operated via the KCP.

4.2 KUKA.HMI user interface

4.2.1 Status keys, menu keys, softkeys

Overview

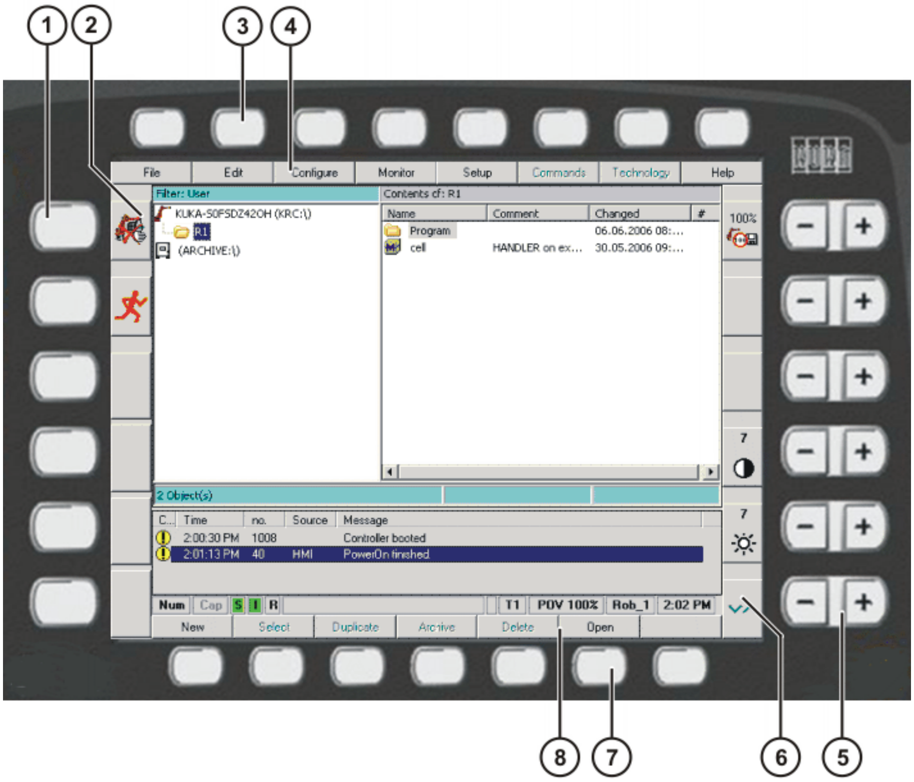


Fig. 4-6: Status keys, menu keys and softkeys in the user interface

1	Left-hand status keys	5	Right-hand status keys
2	Left-hand status keys (icons)	6	Right-hand status keys (icons)
3	Menu keys	7	Softkeys
4	Menu keys (icons)	8	Softkeys (icons)

Description

Element	Description
Status keys	The status keys are used primarily for controlling the robot and setting values. Example: selecting the robot jog mode. The icons change dynamically.
Menu keys	The menu keys are used to open the menus.
Softkeys	The icons change dynamically and always refer to the active window.

4.2.2 Windows in the user interface

Overview

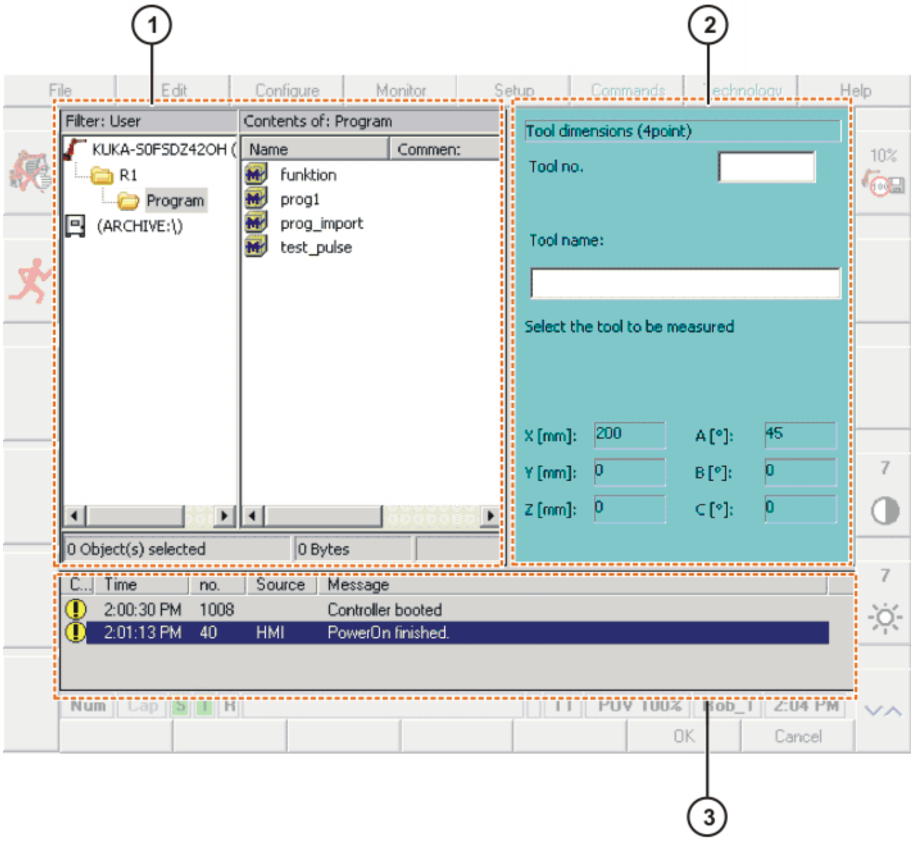


Fig. 4-7: Windows in the user interface

1	Main window	3	Message window
2	Option window		

Description

A maximum of 3 windows can be displayed at a time. The window selection key is used to toggle between the windows. The selected window is indicated by a blue background.

Window	Description
Main window	The main window displays either the Navigator or the selected or opened program.
Option window	Option windows are associated with individual functions or work sequences. They are not permanently visible in the user interface. It is not possible to have more than one option window open at any one time.
Message window	The message window displays error messages, system messages and dialog messages. The message window is not shown if there are no messages present, e.g. if all messages have been acknowledged.

4.2.3 Elements in the user interface

Input box

A value or a text can be entered.

Fig. 4-8: Example of an input box

List box

A parameter can be selected from a list.

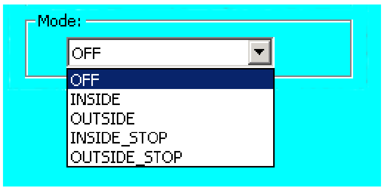


Fig. 4-9: Example of an opened list box

Check box

One or more options can be selected.

Fig. 4-10: Example of a check box

Option box

One option can be selected.

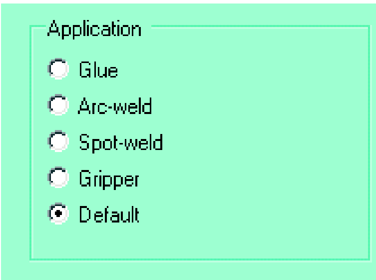


Fig. 4-11: Example of an option box

Slider

A value on a scale can be set.

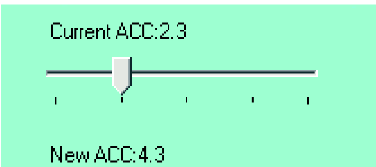


Fig. 4-12: Example of a slider

Group

Boxes can be arranged in groups. A group is indicated by a frame. The name of the group is generally indicated in the top left-hand corner of the frame.

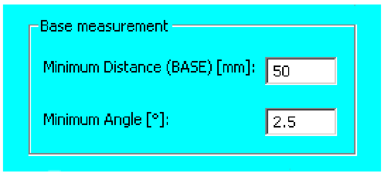


Fig. 4-13: Example of a group

4.2.4 Status bar

Overview

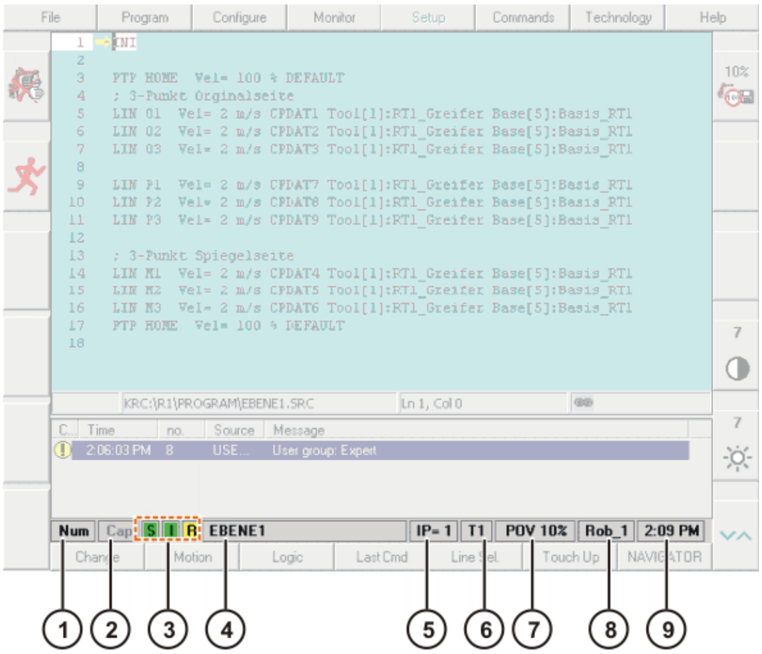


Fig. 4-14: Status bar in the user interface

1 Status of the numeric keypad
2 Upper-case/lower-case status
3 **S**: Status of the Submit interpreter

  I/O: Status of the drives

  R: Status of the program
4 Name of the selected program
5 Number of the current block
6 Current operating mode
7 Current override setting
8 Robot name
9 System time

Description

Icon	Description
Num	The numeric function of the numeric keypad is active.
Num	The control function of the numeric keypad is active.

Icon	Description
Cap	Upper-case characters are active.
Cap	Lower-case characters are active.

Icon	Color	Description
S	Gray	Submit interpreter is deselected.
S	Red	Submit interpreter has been stopped.
S	Green	Submit interpreter is running.
I	Green	Drives ready.
O	Red	Drives not ready.
R	Gray	No program is selected.
R	Yellow	The block pointer is situated on the first line of the selected program.
R	Green	The program is selected and is being executed.
R	Red	The selected and started program has been stopped.
R	Black	The block pointer is situated on the last line of the selected program.

Information about the Submit interpreter is contained in the Expert documentation "Submit-Interpreter".

4.2.5 Calling online help

Description

Help texts are available for the following user interface elements:

- Messages
- Inline forms
- Error display
- Logbook entries

Procedure

1. Select, or position the cursor in, the element for which a help text is to be displayed.
2. Select the menu sequence **Help** > **Online help**.

   The help text for the element is displayed.

Alternative procedure

- Select the menu sequence **Help** > **Online help - Contents/Index**.

  You can search for a help text in the **Contents** and **Index** tabs.

4.2.6 Setting the brightness and contrast of the user interface

Precondition

- The following status key must be displayed for the jog mode:

Procedure

- Set the brightness using the following status key:

- Set the contrast using the following status key:

4.3 Switching on the robot controller and starting the KSS

Procedure

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

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

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

4.4 Restarting the KSS

Precondition

- User group "Expert".
- Operating mode T1 or T2.

Procedure

1. Select the menu sequence **File** > **Shut down KRC**.
2. Select the start type.
3. Press the **Shut Down** softkey. Confirm the request for confirmation with **Yes**.
   The KSS is shut down and then automatically restarts immediately with the selected start type.

Caution!
If the KSS has been restarted with the command **Shut down KRC**, the main switch on the robot controller must not be actuated during the rebooting sequence. System files may otherwise be destroyed.

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

Description

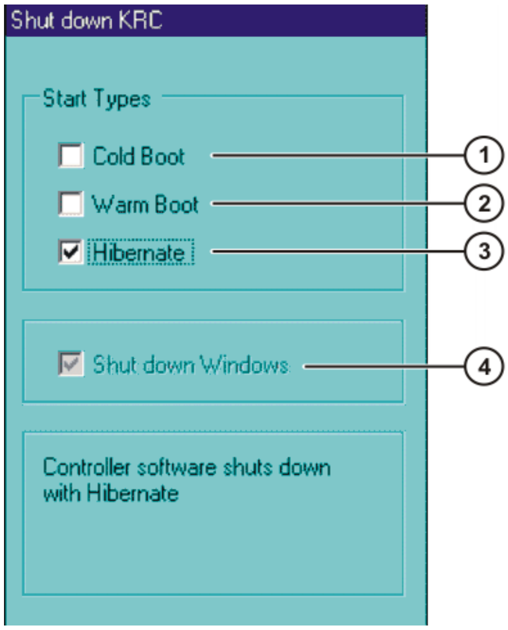


Fig. 4-15: Option window "Shut down KRC"

Item	Description
1	The next start is a cold start.
2	The next start is a warm start.
3	The next start is a start after "Hibernate".
4	Windows is rebooted. This option is only available in the user group "Expert". **Shut down Windows** is automatically active if **Hibernate** has been selected. In this case, Windows is also restarted with Hibernate.

The following softkeys are available:

Softkey	Description
Shut Down	KSS is rebooted. If the option **Shut down Windows** is active, then Windows is also restarted.
Cancel	Closes the window. The settings are not saved.

4.5 Defining the start type for KSS

This function defines how the KSS starts after a power failure. A power failure and start are generally triggered by switching the main switch on the robot controller off and on.

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

Procedure

1. Select the menu sequence **Configure** > **On/Off Options** > **Start Types**.
2. Select the start type.
3. Press the **OK** softkey.

Description

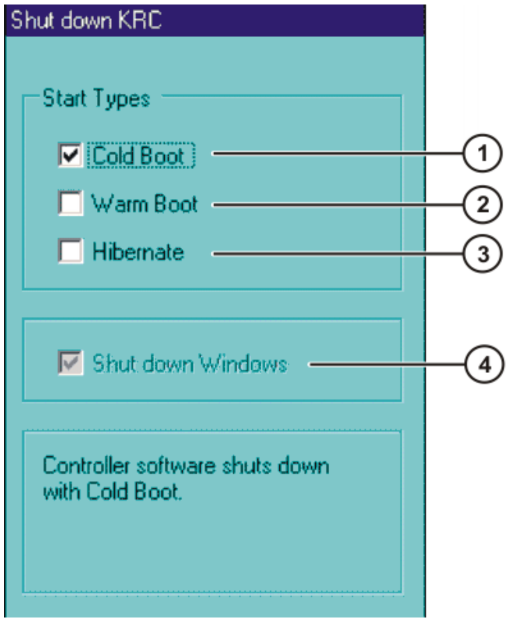


Fig. 4-16: Option window "Start types"

Item	Description
1	Cold start
2	Warm start
3	Hibernate
4	Windows is rebooted. This option cannot be modified in the option window **Start Types**. If **Hibernate** has been selected, Windows is also restarted with Hibernate.

The following softkeys are available:

Softkey	Description
OK	Saves the settings and closes the window.
Cancel	Closes the window. The settings are not saved.

4.6 Start types

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

4.7 Switching the robot controller off

Procedure

- Turn the main switch on the robot controller to OFF.

  The robot controller automatically backs up data.

Caution!
If the KSS has been restarted with the command **Shut down KRC**, the main switch on the robot controller must not be actuated during the rebooting sequence. System files may otherwise be destroyed.

4.8 Setting the user interface language

Procedure

1. Select the menu sequence **Configure** > **Tools** > **Language**.
2. Select the desired language. Confirm with **OK**.

4.9 Changing user group

Procedure

1. Select the menu sequence **Configure** > **User group**. The current user group is displayed.
2. Press the **Default** softkey to switch to the default user group.

   Press the **Log On...** softkey to switch to a different user group. Select the desired user group and confirm with the **Log On...** softkey.

   If prompted: Enter password and confirm with the **Log On** softkey.

Description

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

- **Operator**

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

  User group for the operator
- **Expert**

  User group for the programmer. In this user group it is possible to switch to the Windows interface.

  This user group is protected by means of a password.
- **Administrator**

  The range of functions is the same as that for the user group “Expert”. It is additionally possible, in this user group, to integrate plug-ins into the robot controller.

  This user group is protected by means of a password.

The default password is “kuka”.

By default, the user groups “Operator” and “User” are defined for the same target group. Depending on the customer-specific settings, the range of functions available in the user groups may deviate from the standard and additional user groups may also exist.

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

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

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

4.10 Disabling the robot controller

Description

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

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

Precondition

- The default user group is not selected.

Procedure

1. Select the menu sequence **Configure** > **User group**.
2. Press the **Lock** softkey. The robot controller is then disabled for all actions except logging on. The current user group is displayed.
3. Log back on:
   - Log on as the default user: Press the **Default** softkey.
   - Log on as a different user: Press the **Log On...** softkey. Select the desired user group and confirm with the **Log On...** softkey.

     If prompted: Enter password and confirm with the **Log On** softkey.

If you log onto the same user group as before, all the windows and programs of the previous user remain open. No data are lost.
If you log onto a different user group, the windows and programs of the previous user may be closed. Data can be lost!

4.11 Switching to the operating system interface

Precondition

- User group “Expert”
- The NUM function of the numeric keypad is deactivated.

Procedure

Switch to a different application

1. Press the ALT key and hold it down.
2. Press the TAB key. A window opens, displaying all active applications.
3. Press TAB repeatedly until the desired application is selected. Release both keys. The application is displayed.
4. Pressing ALT + ESC returns to the previous application.

Open the Start menu of the operating system.

1. CTRL + ESC. The Start menu is opened.
2. Using the arrow keys, select the desired menu item and press the Enter key.

4.12 Operating modes

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.

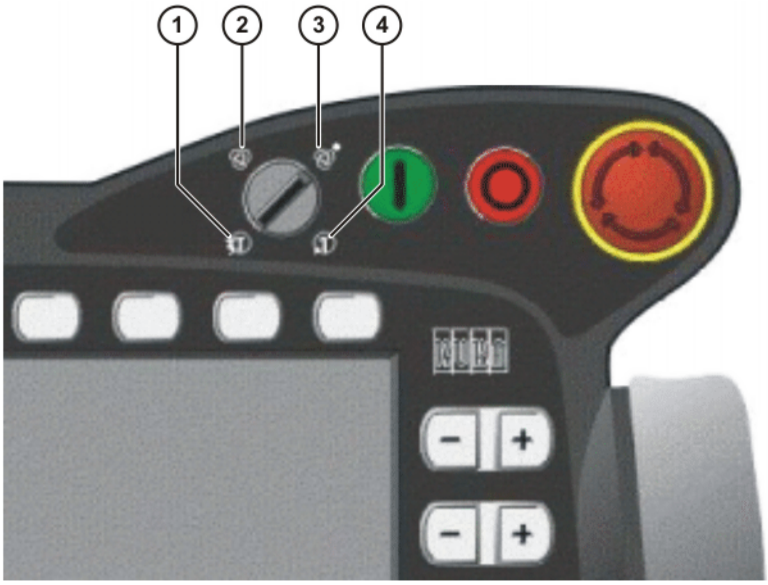


Fig. 4-17: Mode selector switch

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

4.13 Coordinate systems

Overview

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

- WORLD
- ROBROOT
- BASE
- TOOL

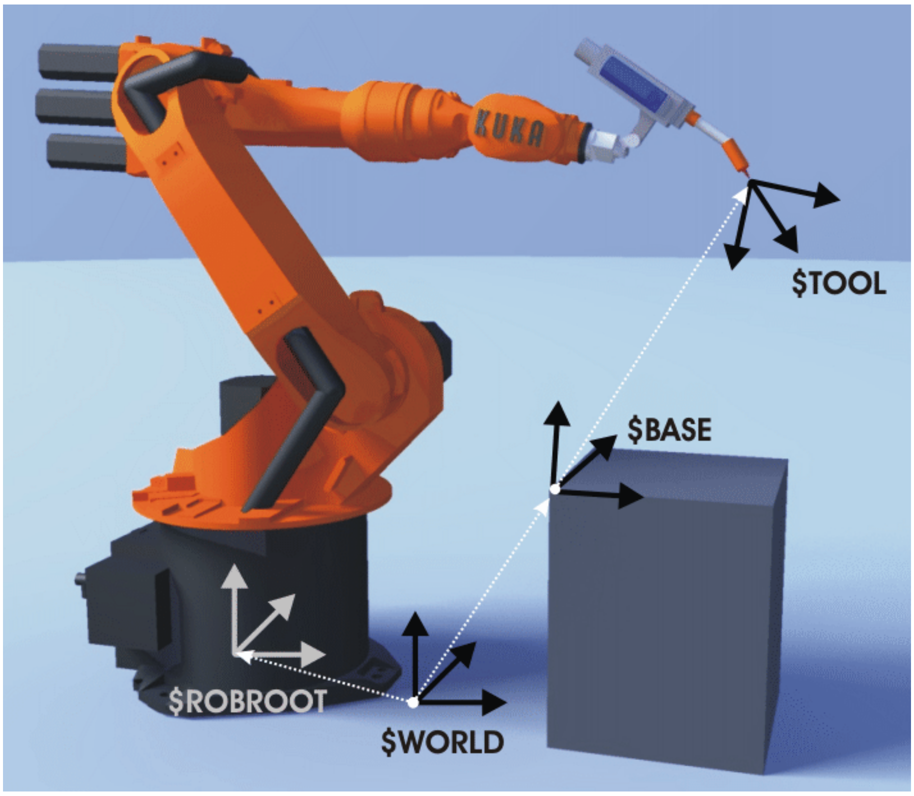


Fig. 4-18: Overview of coordinate systems

Description

WORLD

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

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

ROBROOT

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

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

BASE

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

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

(>>> 5.3.3 "Base calibration" page 90)

TOOL

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

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

(>>> 5.3.1 "Tool calibration" page 80)

4.14 Jogging the robot

Description

There are 2 ways of jogging the robot:

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

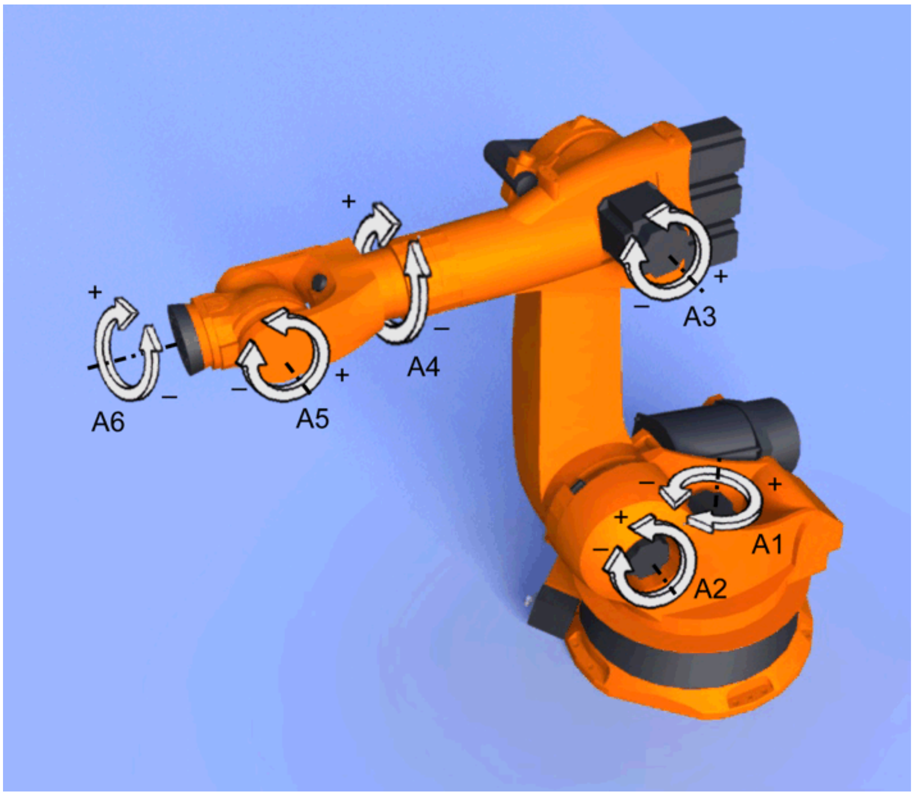


Fig. 4-19: Axis-specific jogging

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

- Jog keys
- Space Mouse

Overview

	Cartesian jogging	Axis-specific jogging
Jog keys	(>>> 4.14.4 "Cartesian jogging with the jog keys" page 55)	(>>> 4.14.3 "Axis-specific jogging with the jog keys" page 54)
Space Mouse	(>>> 4.14.7 "Cartesian jogging with the Space Mouse" page 58)	Axis-specific jogging with the Space Mouse is possible, but is not described here.

4.14.1 Setting the jog override (HOV)

Description

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

Preparation

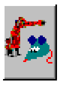
- Define the jog override intervals:

  Select the menu sequence **Configure** > **Jogging** > **Jog OV Steps**.

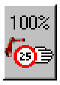
Active	Meaning
No	The override can be adjusted in 1% steps.
Yes	Intervals: 100%, 75%, 50%, 30%, 10%, 3%, 1%

Procedure

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

    or 

2. Increase or reduce the override in the right-hand status key bar. The status key always indicates the current override as a percentage.

4.14.2 Selecting the tool and base

Description

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

Procedure

1. Select the menu sequence **Configure** > **Set 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 base in the box **Base No.**.
5. Press **OK**.

4.14.3 Axis-specific jogging with the jog keys

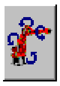
Precondition

- Operating mode T1 or T2.

Procedure

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

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



3. Set jog override.
4. Hold down the enabling switch.
5. Axes A1 to A6 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.

4.14.4 Cartesian jogging with the jog keys

Precondition

- Tool and base have been selected.
  (>>> 4.14.2 "Selecting the tool and base" page 54)
- Operating mode T1 or T2.

Procedure

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

2. Select the coordinate system in the right-hand status key bar.
3. Set jog override.
4. Hold down the enabling switch.
5. The following status keys are displayed in the right-hand status key bar:

   X, **Y**, **Z**: for the linear motions along the axes of the selected coordinate system

   A, **B**, **C**: for the rotational motions about the axes of the selected coordinate system

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

> The position of the robot during jogging can be displayed: select the menu sequence **Monitor** > **Rob. Position**.

4.14.5 Configuring the Space Mouse

Procedure

1. Select the menu sequence **Configure** > **Jogging** > **Mouse configuration**.
2. **Axis selection**: Select whether the TCP is to be moved using translational motions, rotational motions, or both. The following softkeys are available:

   6D; **XYZ**; **ABC**
3. **Dominant mode**: Activate or deactivate. The following softkeys are available:

   Dominant; **Not dom.**
4. The softkey **Close** saves the current settings and closes the window.

Axis selection description

Softkey	Description
XYZ	The robot can only be moved by pulling or pushing the Space Mouse. The following motions are possible with Cartesian jogging: ■ Translational motions in the X, Y and Z directions
ABC	The robot can only be moved by rotating or tilting the Space Mouse. The following motions are possible with Cartesian jogging: ■ Rotational motions about the X, Y and Z axes
6D	The robot can be moved by pulling, pushing, rotating or tilting the Space Mouse. The following motions are possible with Cartesian jogging: ■ Translational motions in the X, Y and Z directions ■ Rotational motions about the X, Y and Z axes

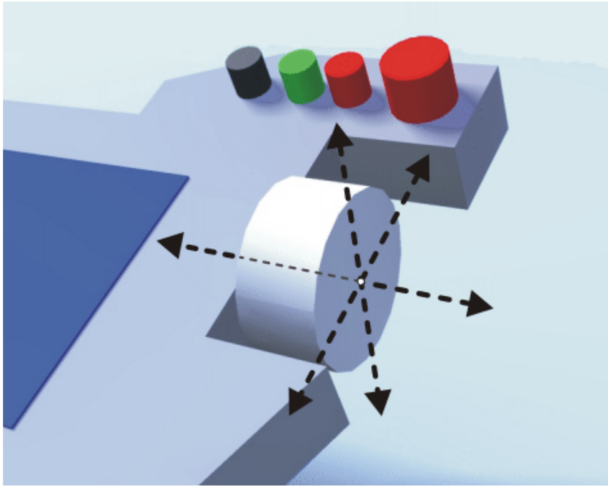


Fig. 4-20: Pushing and pulling the Space Mouse

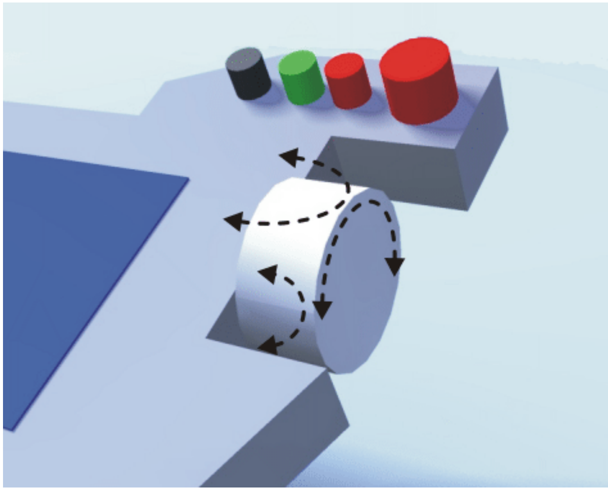


Fig. 4-21: Rotating and tilting the Space Mouse

Description of dominant mode

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

Softkey	Description
Dominant	Activates the dominant mode. Only the coordinate axis with the greatest deflection of the Space Mouse is moved.
Not dom.	Deactivates the dominant mode. Depending on the axis selection, either 3 or 6 axes can be moved simultaneously.

4.14.6 Defining the alignment of the Space Mouse

Description

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

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

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

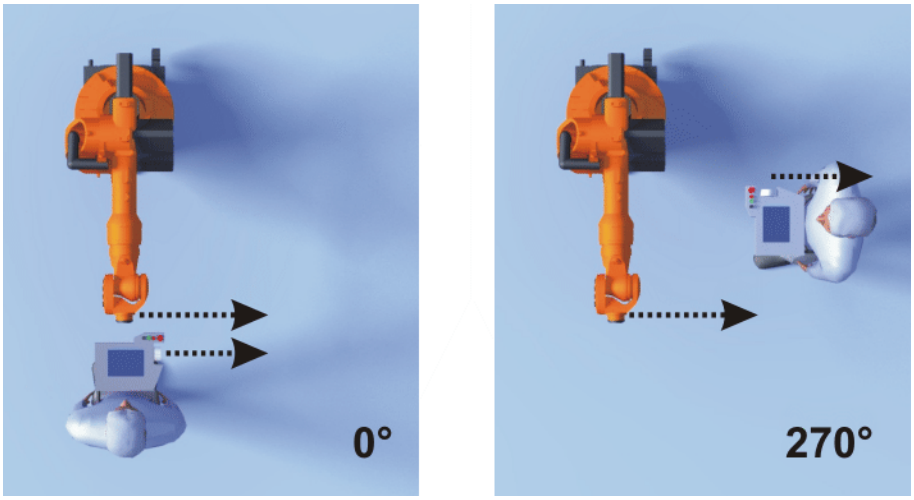


Fig. 4-22: Space Mouse: 0° and 270°

Precondition

- Operating mode T1 or T2.

Procedure

1. Select the menu sequence **Configure** > **Jogging** > **Mouse position**.
2. The alignment of the Space Mouse can be modified using the “**+**” or “**-**” softkey.

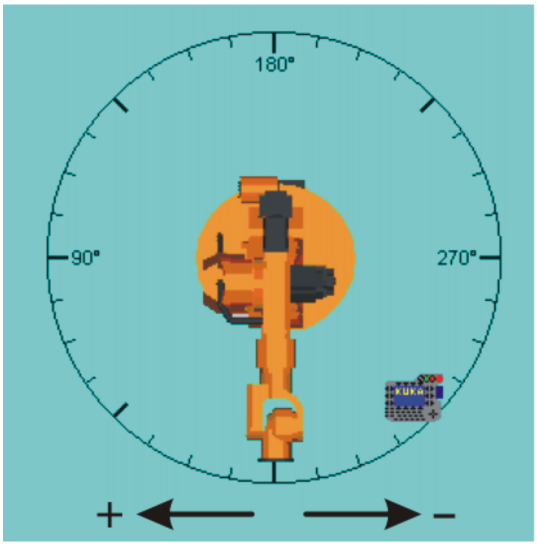


Fig. 4-23: Option window for aligning the Space Mouse

3. The softkey **Close** saves the current settings and closes the window.

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

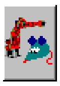
4.14.7 Cartesian jogging with the Space Mouse

Precondition

- Tool and base have been selected.
  (>>> 4.14.2 "Selecting the tool and base" page 54)
- The Space Mouse is configured.
  (>>> 4.14.5 "Configuring the Space Mouse" page 55)
- The alignment of the Space Mouse has been defined.
  (>>> 4.14.6 "Defining the alignment of the Space Mouse" page 57)
- Operating mode T1 or T2.

Procedure

1. Select the following jog mode in the left-hand status key bar:

2. Select the coordinate system in the right-hand status key bar.
3. Set jog override.
4. Hold down the enabling switch.
5. Move the robot in the desired direction using the Space Mouse.

> The position of the robot during jogging can be displayed: select the menu sequence **Monitor** > **Rob. Position**.

4.14.8 Incremental jogging

Description

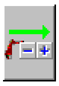
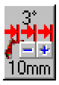
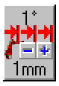
Incremental jogging makes it possible to move the robot a defined distance, e.g. 10 mm or 3°. The robot then stops by itself.

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

Area of application:

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

The following status keys are available in the right-hand status key bar for incremental jogging:

Status key	Description
	Incremental jogging switched off
	Increment = 100 mm or 10°
	Increment = 10 mm or 3°
	Increment = 1 mm or 1°
	Increment = 0.1 mm or 0.005°

Increments in mm:

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

Increments in degrees:

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

Procedure

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

2. Set the size of the increment in the right-hand status key bar.
3. Jog the robot using the jog keys. Jogging can be Cartesian or axis-specific.

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

   (>>> 4.14.3 "Axis-specific jogging with the jog keys" page 54)
   (>>> 4.14.4 "Cartesian jogging with the jog keys" page 55)

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

4.15 Bypassing workspace monitoring

Description

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

There are 2 types of workspace:

- The workspace is an exclusion zone.

  The robot may only move outside the workspace.

- Only the workspace is a permitted zone.
  The robot may not move outside the workspace.

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

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

Precondition

- User group “Expert”
- Operating mode T1

Procedure

1. Select the menu sequence **Configure** > **Tools** > **Monitoring working envelope** > **Override**.
2. Move the robot manually out of the prohibited workspace.
   Once the robot has left the prohibited workspace, the workspace monitoring is automatically active again.

4.16 Monitor functions

4.16.1 Overview of the monitor functions

Topic	Monitor functions
Displaying the current robot position	(>>> 4.16.2 "Displaying the actual position" page 60)
Displaying inputs/outputs	(>>> 4.16.4 "Displaying analog inputs/outputs" page 62) (>>> 4.16.3 "Displaying digital inputs/outputs" page 61) (>>> 4.16.5 "Displaying inputs/outputs for Automatic External" page 63)
Displaying information about the robot system	(>>> 4.16.6 "Displaying information about the robot system" page 65) (>>> 4.16.7 "Displaying robot data" page 65)
Displaying information about the hardware	(>>> 4.16.8 "Displaying hardware information" page 66)

4.16.2 Displaying the actual position

Procedure

- Select the menu sequence **Monitor** > **Rob. Position** > **Cartesian** or **Axis specific**.

Description

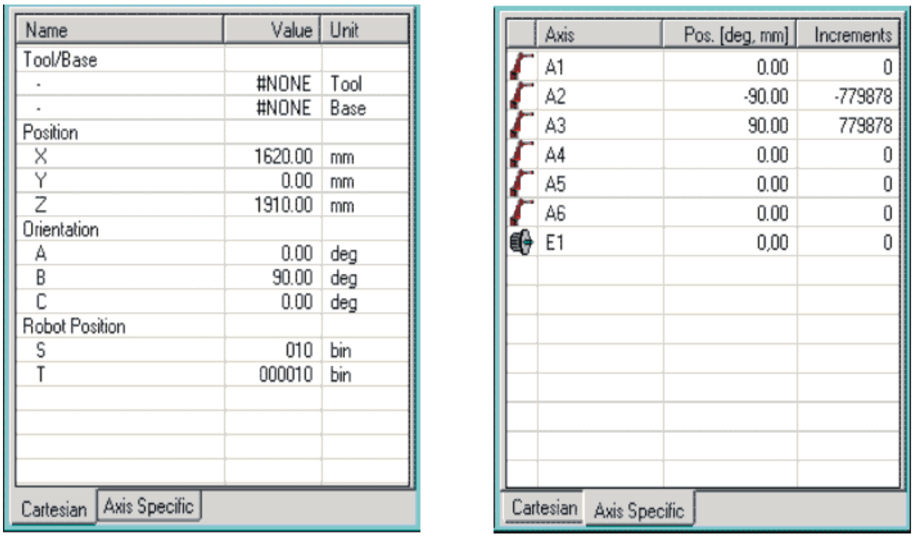


Fig. 4-24: Actual position: Cartesian and axis-specific

Cartesian

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

Axis-specific

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

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

4.16.3 Displaying digital inputs/outputs

Procedure

- Select the menu sequence **Monitor** > **I/O** > **Digital Outputs** or **Digital Inputs**.

Description

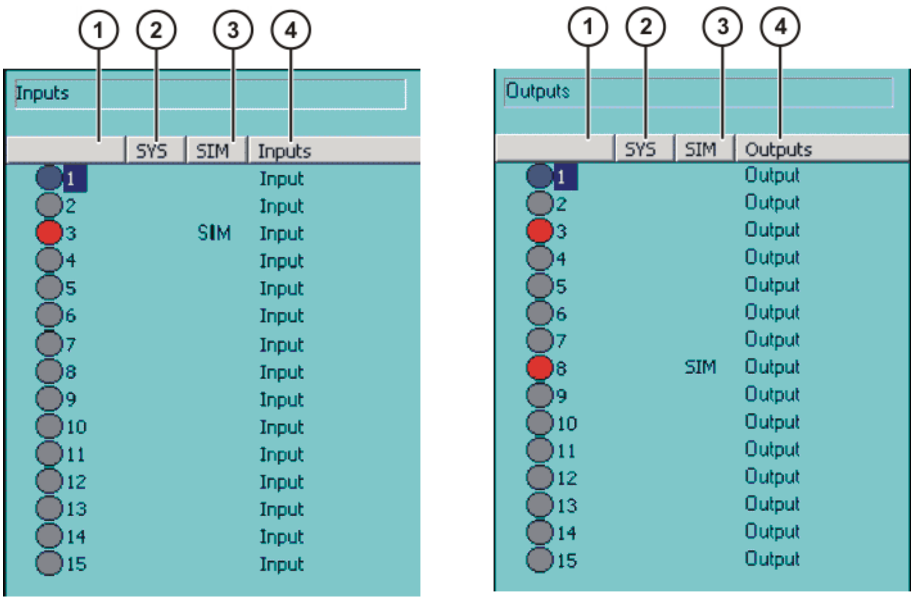


Fig. 4-25: Digital inputs/outputs

Col-umn	Description
1	Input/output number. The icon is red if the input or output is set.
2	SYS entry: input/output whose value is saved in a system variable.
3	SIM entry: simulated input/output. This column is only displayed if I/O simulation is activated.
4	Name of the input/output

The following softkeys are available:

Softkey	Description
Value	Toggles the selected input/output between TRUE and FALSE. Precondition: The enabling switch is pressed. This softkey is not available in AUT mode.
Name	The name of the selected input or output can be changed.

An input/output can be selected by entering its number via the numeric keypad. In this way, inputs/outputs that are not visible in the option window can be displayed. Preconditions:

- The option window is active.
- The “NUM” function is active in the status bar.

4.16.4 Displaying analog inputs/outputs

Procedure

- Select the menu sequence **Monitor** > **I/O** > **Analog I/O**.

Description

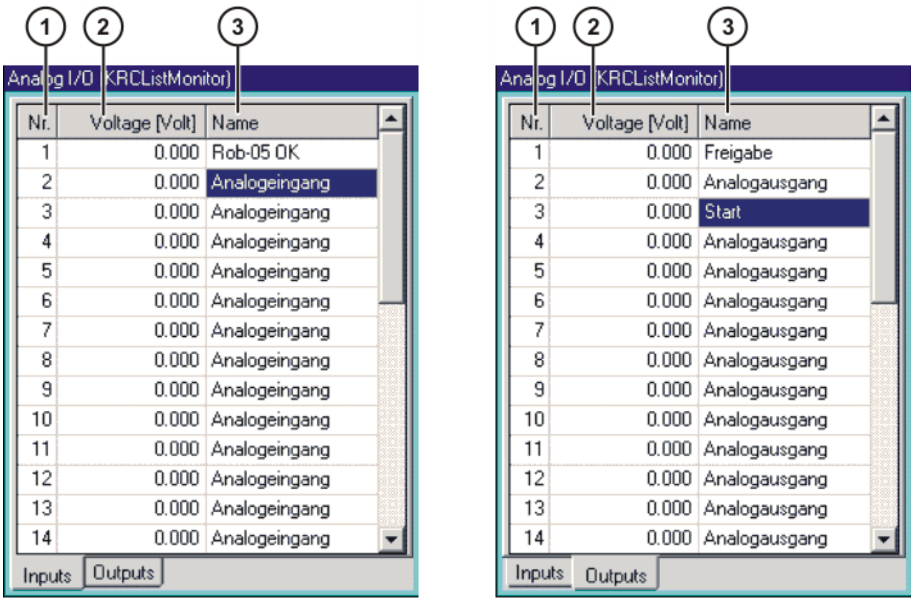


Fig. 4-26: Analog inputs/outputs

Col-umn	Description
1	Input/output number

Column	Description
2	Input/output voltage Range of values: -10 to +10 volts
3	Name of the input/output

The following softkeys are available:

Softkey	Description
Tab +	Toggles between the **Inputs** and **Outputs** tabs.
Voltage	A voltage can be entered for the selected output. This softkey is not available for inputs.
Name	The name of the selected input or output can be changed.

An input/output can be selected by entering its number via the numeric keypad. In this way, inputs/outputs that are not visible in the option window can be displayed. Preconditions:

- The option window is active.
- The “NUM” function is active in the status bar.

4.16.5 Displaying inputs/outputs for Automatic External

Procedure

- Select the menu sequence **Monitor** > **I/O** > **Automatic External**.

Description

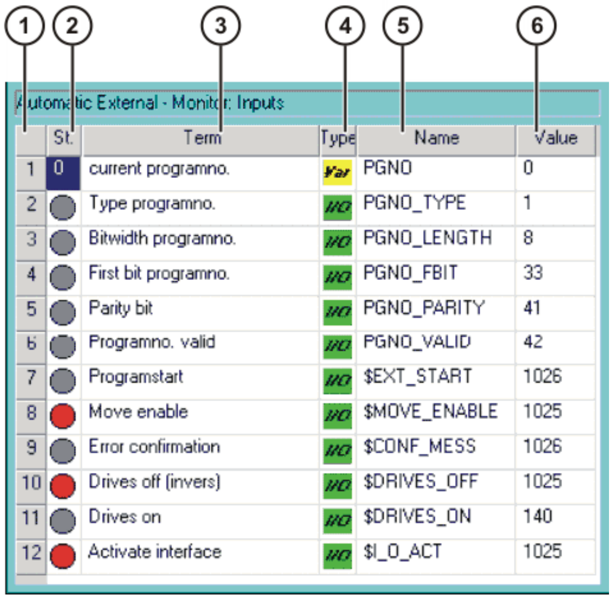


Fig. 4-27: Automatic External inputs (detail view)

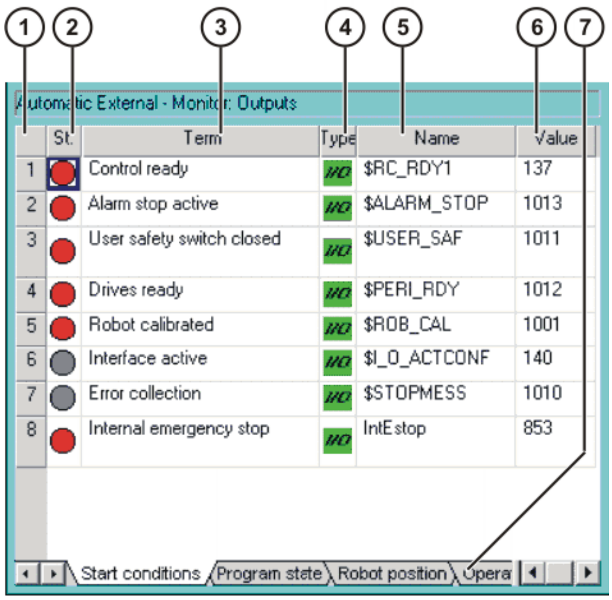


Fig. 4-28: Automatic External outputs (detail view)

Column	Description
1	Number
2	State ■ Gray: inactive (FALSE) ■ Red: active (TRUE)
3	Long text name of the input/output
4	Type ■ Green: Input/output ■ Yellow: Variable or system variable ($...)
5	Name of the signal or variable
6	Input/output number or channel number
7	The outputs are thematically assigned to the following tabs: ■ Start conditions ■ Program status ■ Robot position ■ Operating mode

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

The following softkeys are available:

Softkey	Description
Configure	Switches to the configuration of the Automatic External interface.
Inputs/outputs	Toggles between the windows for inputs and outputs.
Details/Normal	Toggles between the **Details** and **Normal** views.
Tab -/Tab +	Toggles between the tabs. This softkey is only available for outputs.

4.16.6 Displaying information about the robot system

Procedure

- Select the menu sequence **Help** > **Info**.

Description

Information about the robot system is required, for example, when requesting help from KUKA Customer Support.

The tabs contain the following information:

Tab	Description
Info	■ Robot controller type ■ Robot controller version ■ User interface version ■ Kernel system version
Robot	■ Robot name ■ Robot type and configuration ■ Operating hours The operating hours meter is running as long as the drives are switched on. Alternatively, the operating hours can also be displayed via the variable $ROB-RUNTIME. ■ Number of axes ■ List of external axes ■ Machine data version
System	■ Control PC name ■ Operating system versions and BIOS version ■ Storage capacities
Options	Additionally installed options and technology packages
Comments	Additional comments
Modules	Names and versions of important system files The **Save** softkey exports the contents of the **Modules** tab to the file C:\ KRC\ ROBOTER\ LOG\ OCXVER.TXT.
Virus scanner	Names and versions of installed virus scanner files The **Export** softkey exports the contents of the **Virus Scanner** tab to the file C:\KRC\ROBOTER\LOG\VIRUS-INFO.XML.

4.16.7 Displaying robot data

Procedure

- Select the menu sequence **Setup** > **Robot data**.

Description

Data on RDC and hard drive are identical

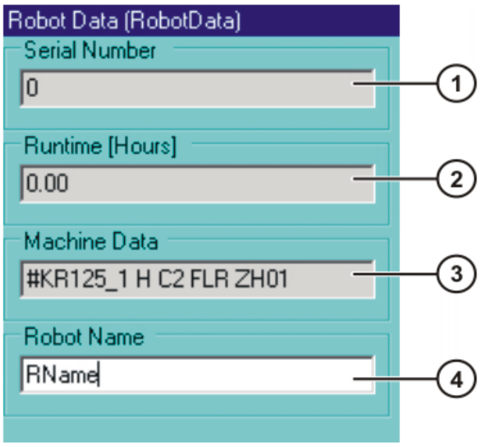


Fig. 4-29: Robot data – data on RDC and hard drive are identical

Item	Description
1	Serial number. The serial number can be modified in the user group “Expert”.
2	Operating hours. The operating hours meter is running as long as the drives are switched on. Alternatively, the operating hours can also be displayed via the variable $ROBRUNTIME.
3	Machine data name
4	Robot name. The robot name can be changed.

Data on RDC and hard drive are not identical

If the RDC or the hard drive has been exchanged, the data on the RDC and on the hard drive are no longer identical. This is indicated by a message. (This is also the case, for example, if not just the hard drive, but the entire robot controller has been exchanged.)

The **Robot data** window indicates the different data.

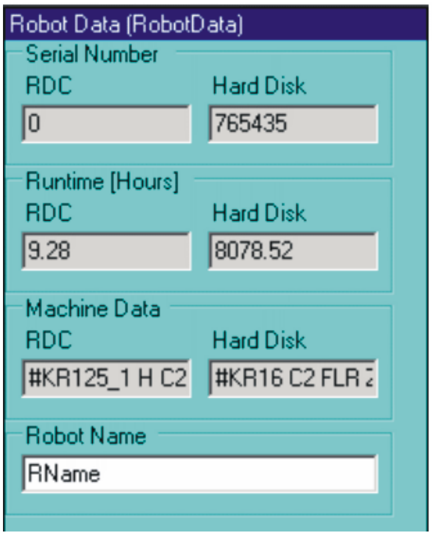


Fig. 4-30: Robot data – data on RDC and hard drive are not identical

4.16.8 Displaying hardware information

Procedure

1. Select the menu sequence **Monitor** > **Hardware Info**.

2. If required, open up the tree structure in the left-hand section of the window and select the desired hardware component.

   Information about the selected component is displayed in the right-hand section of the window.

Description

The following softkeys are available:

Softkey	Description
Load config	Loads the last saved configuration.
Refresh	Refreshes the display.
Export	Exports the hardware information as an XML file.

5 Start-up

5.1 Checking the machine data

Description

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

If machine data are reloaded, the version of the machine data must correspond exactly to the KSS version. This is ensured if only the machine data from the CD with the KSS, that is also installed, are used.

Warning!
The robot must not be moved if incorrect machine data are loaded. Personal injuries or damage to property may result in this case.

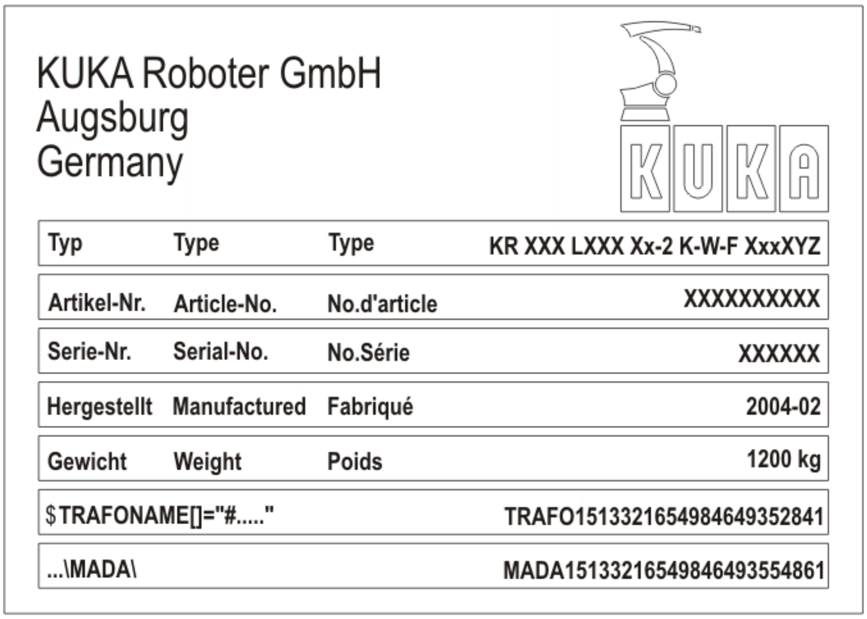


Fig. 5-1: Rating plate

Procedure

1. Select the menu sequence **Setup** > **Robot data**.
   The **Robot data** window is opened.
2. Compare the following entries:
   - In the **Robot data** window: the entry in the **Machine data** box
   - On the rating plate on the base of the robot: the entry in the line **$TRAFONAME()="# ..... "**

The file path of the machine data on the CD is specified on the rating plate in the line **...\MADA**.

5.2 Mastering

Overview

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

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

Fig. 5-2: Mastering position – approximate position

A robot must be mastered in the following cases:

Case	Comments
During commissioning	- - -
After maintenance work during which the robot loses its mastering, e.g. exchange of motor or RDC	(>>> 5.2.6 "Reference mastering" page 78)
When the robot has been moved without the robot controller (e.g. with the release device)	- - -
After exchanging a gear unit	Before carrying out a new mastering procedure, the old mastering data must first be deleted! Mastering data are deleted by manually unmastering the axes. (>>> 5.2.8 "Manually unmastering axes" page 79)
After an impact with an end stop at more than 250 mm/s	
After a collision	

The mastering is automatically backed up in the following cases:

- the brakes of all axes are applied.
- The KRL program is stopped or terminated.

The saved data are checked. If they do not match the last backup, the flash of the RDC is defective. In this case, an error message is displayed.

5.2.1 Mastering methods

Overview

A robot can be mastered in the following ways:

- With the EMT (electronic measuring tool)
  (>>> 5.2.3 "Mastering with the EMT" page 72)
- With the dial gauge
  (>>> 5.2.4 "Mastering with the dial gauge" page 76)

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

EMT mastering is recommended.

Another method is “reference mastering”. It is only used for mastering the robot after certain maintenance tasks.

(>>> 5.2.6 "Reference mastering" page 78)

5.2.2 Moving axes to the pre-mastering position

Description

Each axis is moved so that the mastering marks line up. The pre-mastering position is a prerequisite for every mastering.

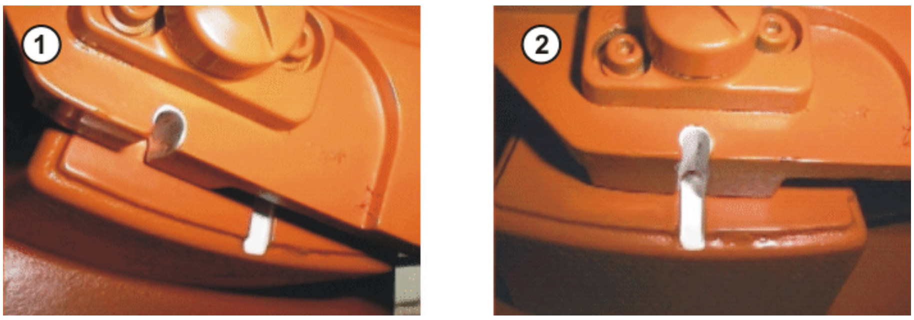


Fig. 5-3: Moving an axis to the pre-mastering position

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

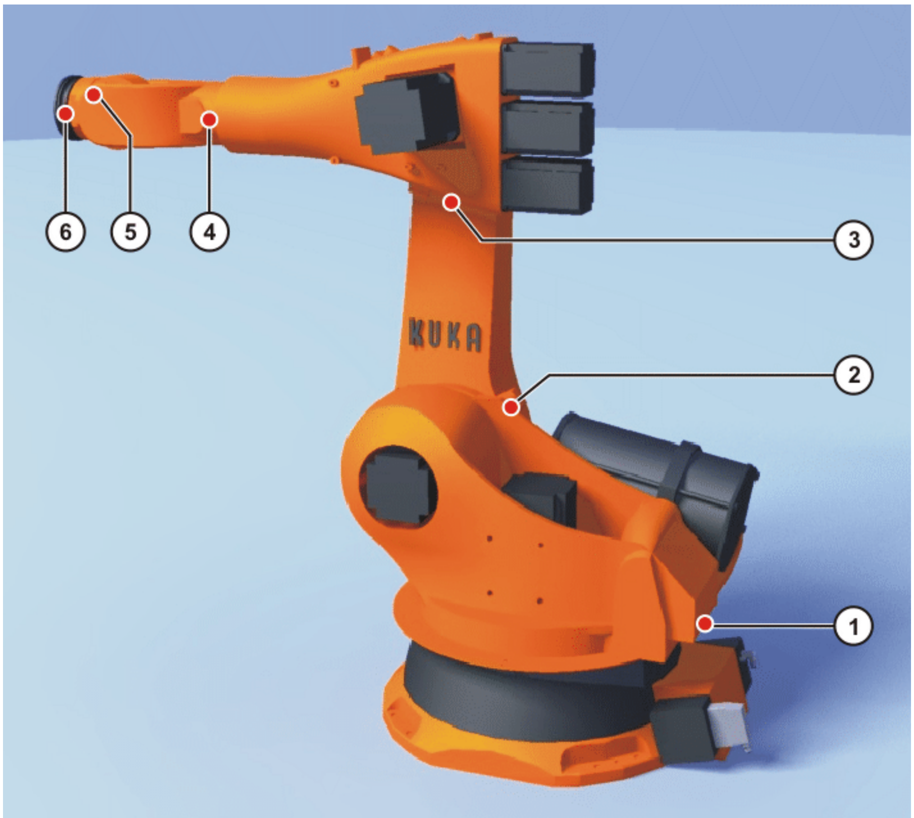


Fig. 5-4: Mastering marks on the robot

> Depending on the specific robot model, the positions of the mastering marks may deviate slightly from those illustrated.

Precondition

- Operating mode T1

Procedure

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

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

3. Hold down the enabling switch.
4. Axes 1 to 6 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.
5. Move each axis, starting from axis 1 and working upwards, so that the mastering marks line up.

> If A4 and A6 are moved to the pre-mastering position, ensure that the energy supply system – if present – is in its correct position and not rotated through 360°.

5.2.3 Mastering with the EMT

Overview

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

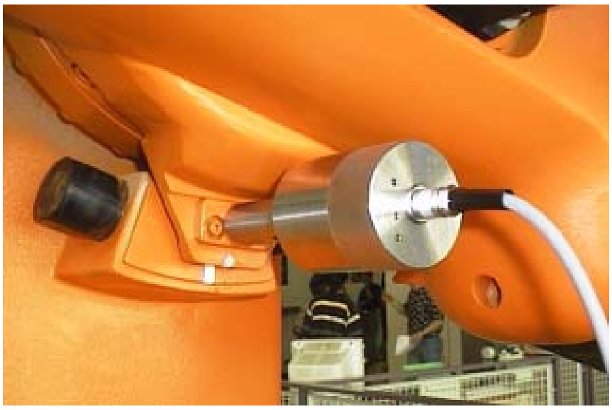


Fig. 5-5: Electronic measuring tool

EMT mastering consists of the following steps:

Step	Description
1.	**First mastering** (>>> 5.2.3.1 "First mastering with the EMT" page 73) First mastering is carried out without a load.
2.	**Teach offset** (>>> 5.2.3.2 "Teach offset" page 74) “Teach offset” is carried out with a load. The difference from the first mastering is saved.
3.	If required: **Master load with offset** (>>> 5.2.3.3 "Master load with offset" page 75) "Load mastering with offset" is carried out with a load for which an offset has already been taught. Area of application: ■ Checking first mastering ■ Restoring first mastering if it has been lost (e.g. following exchange of motor or collision). Since an offset that has been taught is retained, even if mastering is lost, the robot controller can calculate the first mastering.

5.2.3.1 First mastering with the EMT

Precondition

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

Procedure

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.

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.
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 mastering position is reached. The robot 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.
7. Remove signal cable from connection X32.
8. Exit the option window by means of the softkey **Close**.

5.2.3.2 Teach offset

Description

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

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

Precondition

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

Procedure

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.

1. Select the menu **Setup** > **Master** > **EMT** > **With load correction** > **Teach offset**.

2. Enter tool number. Confirm with **Tool OK**.

   An option window is opened. All axes for which the tool has not yet been taught are displayed. The axis with the lowest number is highlighted.
3. 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.
4. Press the softkey **Teach**.
5. Press an enabling switch and the Start key.

   When the EMT detects the lowest point of the reference notch, the mastering position is reached. The robot stops automatically. An option window is opened. The deviation of this axis from the first mastering is indicated in degrees and increments.
6. Confirm with **OK**. The axis is no longer displayed in the option window.
7. Remove signal cable from EMT. Then remove EMT from the gauge cartridge and replace the protective cap.
8. Repeat steps 3 to 7 for all axes to be mastered.
9. Remove signal cable from connection X32.
10. Exit the option window by means of the softkey **Close**.

5.2.3.3 Master load with offset

Description

Area of application:

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

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

Precondition

- Same ambient conditions (temperature, etc.) as for first mastering.
- A load for which “Teach offset” has been carried out is mounted on the robot.
- All axes are in the pre-mastering position.
- No program is selected.
- Operating mode T1

Procedure

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.

1. Select the menu **Setup** > **Master** > **EMT** > **With load correction** > **Master load** > **With offset**.
2. Enter tool number. Confirm with **Tool OK**.

   An option window is opened. All axes for which an offset has been taught with this tool are displayed. The axis with the lowest number is highlighted.

3. Remove the protective cap of the gauge cartridge on the axis highlighted in the option window. Mount EMT on gauge cartridge. Then attach signal cable to EMT and plug into connector X32 on the base frame junction box.
4. Press the softkey **Check**.
5. Hold down an enabling switch and press the Start key.

   When the EMT detects the lowest point of the reference notch, the mastering position is reached. The robot stops automatically. The difference from "Teach offset" is displayed.
6. If required, press **Save** to save the values. The old mastering values are deleted.

   To restore a lost first mastering, always save the values.

Axes A4, A5 and A6 are mechanically coupled. This means:
If the values for A4 are deleted, the values for A5 and A6 are also deleted.
If the values for A5 are deleted, the values for A6 are also deleted.

7. Remove signal cable from EMT. Then remove EMT from the gauge cartridge and replace the protective cap.
8. Repeat steps 3 to 7 for all axes to be mastered.
9. Remove signal cable from connection X32.
10. Exit the option window by means of the softkey **Close**.

5.2.4 Mastering with the dial gauge

Description

In dial mastering, the axis is moved manually by the user to the mastering position. Mastering is always carried out with a load. It is not possible to save mastering data for different loads.

Fig. 5-6: Dial gauge

Precondition

- The load is mounted on the robot.
- All axes are in the pre-mastering position.
- Axis-specific jogging with the jog keys is selected.

  (>>> 4.14.3 "Axis-specific jogging with the jog keys" page 54)
- No program is selected.
- Operating mode T1

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 zero.
7. Switch to incremental jogging in the right-hand status key bar.
8. Move the axis from “+” to “-” until zero is reached.

> If the axis overshoots zero, 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.
13. Exit the option window by means of the softkey **Close**.

5.2.5 Mastering external axes

Description

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

Procedure

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

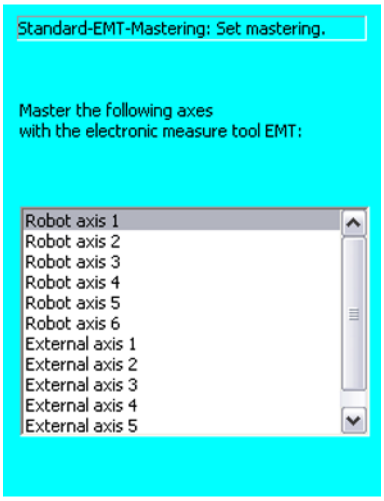


Fig. 5-7: Selection list of axes to be mastered

Mastering in the case of robot systems with more than 2 external axes: if the system contains more than 8 axes, it may be necessary to connect the signal cable of the EMT to the second RDC.

5.2.6 Reference mastering

Description

Reference mastering cannot be used when commissioning the robot.

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

- Exchange of RDC
- Exchange of motor

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

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

Preparation

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

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

Precondition

- No program is selected.

- The position of the robot was not changed during the maintenance work.
- If the RDC has been exchanged: the robot data have been transferred from the hard drive to the RDC (this can only be done in the user group “Expert”!)

Procedure

1. Select the menu sequence **Setup** > **Master** > **Reference**.

   The option window **Reference-Mastering** is opened. All axes that have not been mastered are displayed. The axis that must be mastered first is selected.
2. Press the **Master** softkey. The selected axis is mastered and removed from the option window.
3. Repeat step 2 for all axes to be mastered.
4. Exit the option window **Reference-Mastering** by pressing the **Close** softkey.

5.2.7 Saving the mastering

Procedure

- Select the menu sequence **Setup** > **Master** > **Save current data**.

Description

Save current data saves all mastering data to the hard drive.

In earlier KSS versions, this manual backup prevented loss of the mastering data if the robot controller could not be shut down properly, e.g. due to a defective battery.

The mastering is now backed up automatically. (>>> 5.2 "Mastering" page 69) Manual backup is no longer necessary. The command is still available in the user interface, however.

5.2.8 Manually unmastering axes

Description

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

Axes A4, A5 and A6 are mechanically coupled. This means:
If the values for A4 are deleted, the values for A5 and A6 are also deleted.
If the values for A5 are deleted, the values for A6 are also deleted.

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

Precondition

- No program is selected.

Procedure

1. Select the menu sequence **Setup** > **Unmaster**. An option window is opened.
2. Select the axis to be unmastered.
3. Press the **Unmaster** softkey. The mastering data of the axis are deleted.
4. Repeat steps 2 and 3 for all axes to be unmastered.
5. Exit the option window by means of the softkey **Close**.

5.3 Calibration

5.3.1 Tool calibration

Description

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

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

In the case of a fixed tool, the type of calibration described here must not be used. A separate type of calibration must be used for fixed tools.
(>>> 5.3.2 "Fixed tool calibration" page 86)

Advantages of the tool calibration:

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

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

The following data are saved:

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

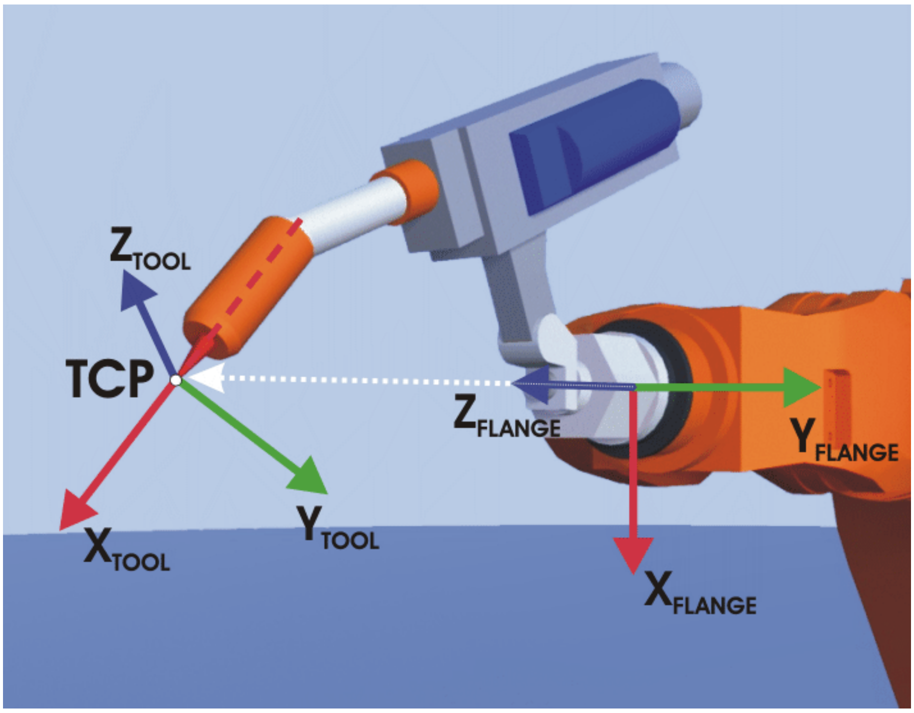


Fig. 5-8: TCP calibration principle

Overview

Tool calibration consists of 2 steps:

Step	Description
1	**Definition of the origin of the TOOL coordinate system** The following methods are available: ■ XYZ 4-Point (>>> 5.3.1.1 "TCP calibration: XYZ 4-Point method" page 81) ■ XYZ Reference (>>> 5.3.1.2 "TCP calibration: XYZ Reference method" page 82)
2	**Definition of the orientation of the TOOL coordinate system** The following methods are available: ■ ABC World (>>> 5.3.1.3 "Defining the orientation: ABC World method" page 83) ■ ABC 2-Point (>>> 5.3.1.4 "Defining the orientation: ABC 2-Point method" page 84)

If the calibration data are already known, they can be entered directly.
(>>> 5.3.1.5 "Numeric input" page 85)

5.3.1.1 TCP calibration: XYZ 4-Point method

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

Description

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

The 4 flange positions at the reference point must be sufficiently different from one another.

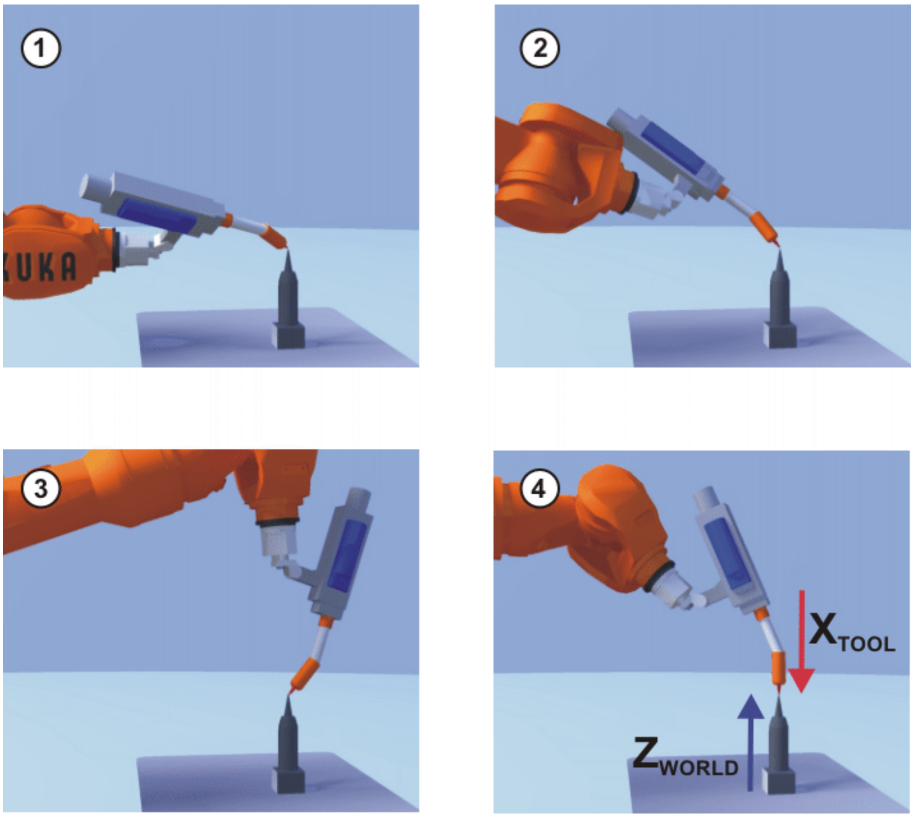


Fig. 5-9: XYZ 4-Point method

Precondition

- The tool to be calibrated is mounted on the mounting flange.
- Operating mode T1 or T2.

Procedure

1. Select the menu **Setup** > **Measure** > **Tool** > **XYZ 4-Point**.
2. Assign a number and a name for the tool to be calibrated. Confirm with **Continue**.
3. Move the TCP to a reference point. Confirm with **Continue**.
4. Move the TCP to the reference point from a different direction. Confirm with **Continue**.
5. Repeat step 4 twice.
6. Press **Save**.

5.3.1.2 TCP calibration: XYZ Reference method

Description

In the case of the XYZ Reference method, a new tool is calibrated with a tool that has already been calibrated. The robot controller compares the flange positions and calculates the TCP of the new tool.

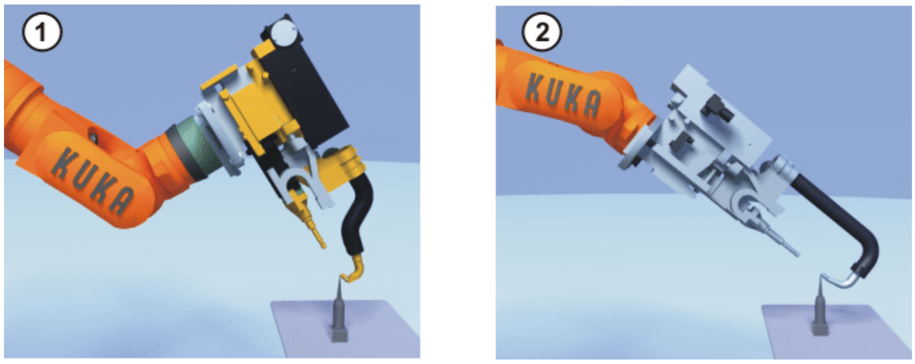


Fig. 5-10: XYZ Reference method

Precondition

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

Preparation

Calculate the TCP data of the calibrated tool:

1. Select the menu **Setup** > **Measure** > **Tool** > **XYZ Reference**.
2. Enter the number of the calibrated tool.
3. Note the X, Y and Z values.
4. Close the window by pressing **Cancel**.

Procedure

1. Select the menu **Setup** > **Measure** > **Tool** > **XYZ Reference**.
2. Assign a number and a name for the new tool. Confirm with **Continue**.
3. Enter the TCP data of the calibrated tool. Confirm with **Continue**.
4. Move the TCP to a reference point. Confirm with **Continue**.
5. Move the tool away and remove it. Mount the new tool.
6. Move the TCP of the new tool to the reference point. Confirm with **Continue**.
7. Press **Save**.

5.3.1.3 Defining the orientation: ABC World method

Description

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

There are 2 variants of this method:

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

  Area of application: e.g. MIG/MAG welding, laser cutting or waterjet cutting
- **6D**: The directions of all 3 axes are communicated to the robot controller.

  Area of application: e.g. for weld guns, grippers or adhesive nozzles

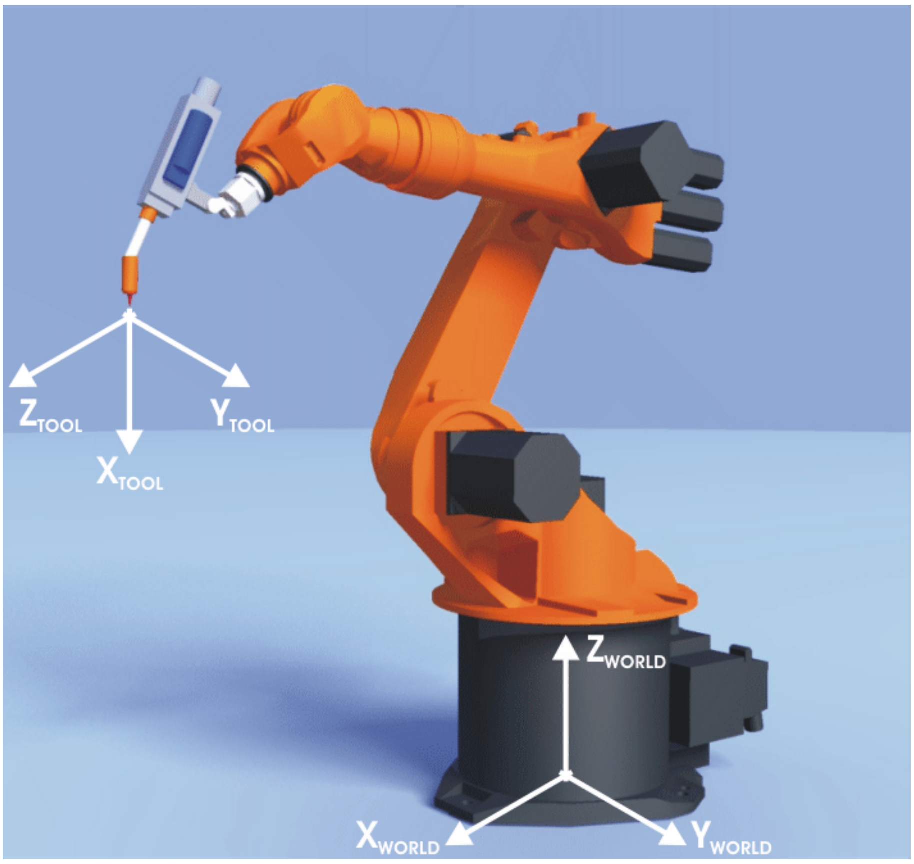


Fig. 5-11: ABC World method

Precondition

- The tool to be calibrated is mounted on the mounting flange.
- The TCP of the tool has already been measured.
- Operating mode T1 or T2

Procedure

1. Select the menu **Setup** > **Measure** > **Tool** > **ABC World**.
2. Enter the number of the tool. Confirm with **Continue**.
3. Select a variant in the box **5D/6D**. Confirm with **Continue**.
4. If **5D** is selected:

   Align $+X_{TOOL}$ parallel to $-Z_{WORLD}$. ($+X_{TOOL}$ = tool direction)

   If **6D** is selected:

   Align the axes of the TOOL coordinate system as follows.

   - $+X_{TOOL}$ parallel to $-Z_{WORLD}$. ($+X_{TOOL}$ = tool direction)
   - $+Y_{TOOL}$ parallel to $+Y_{WORLD}$
   - $+Z_{TOOL}$ parallel to $+X_{WORLD}$

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

5. Confirm with **Continue**.
6. Press **Save**.

5.3.1.4 Defining the orientation: ABC 2-Point method

Description

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

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

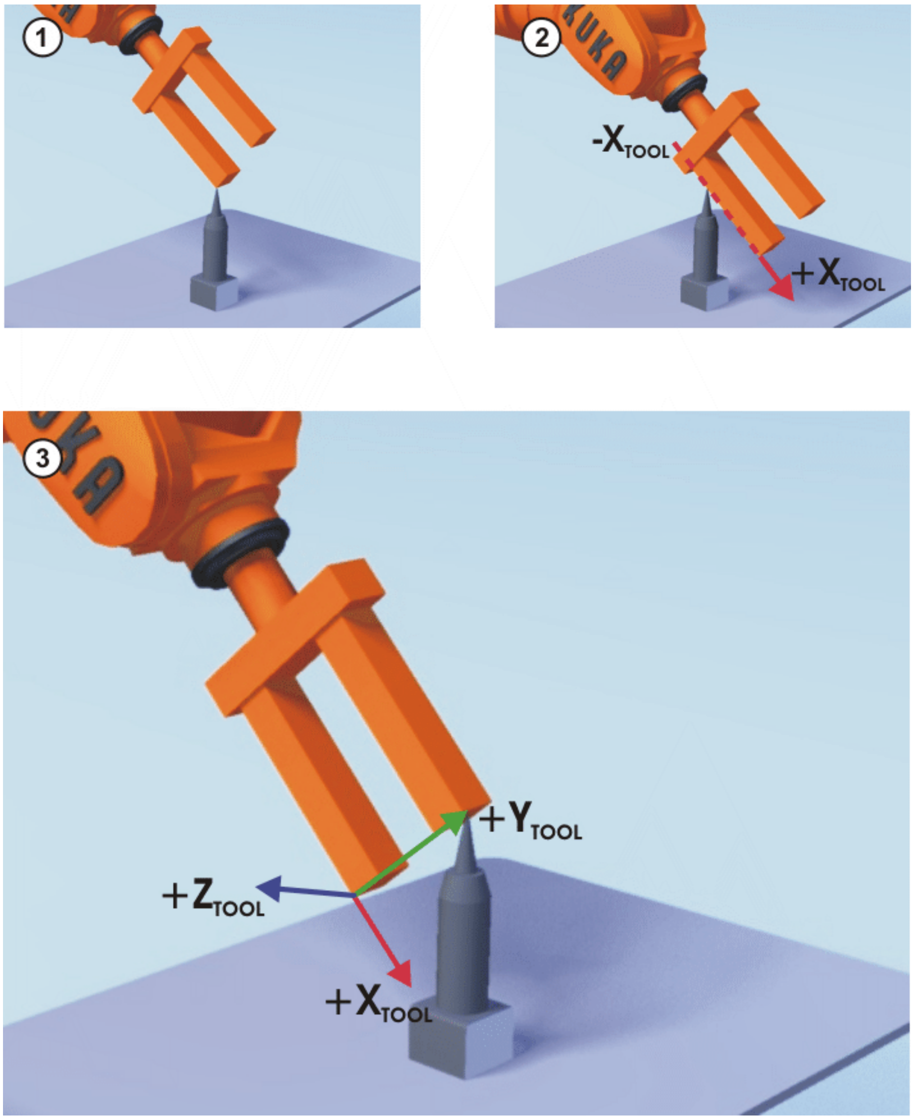


Fig. 5-12: ABC 2-Point method

Precondition

- The tool to be calibrated is mounted on the mounting flange.
- The TCP of the tool has already been measured.
- Operating mode T1 or T2

Procedure

1. Select the menu **Setup** > **Measure** > **Tool** > **ABC 2-Point**.
2. Enter the number of the mounted tool. Confirm with **Continue**.
3. Move the TCP to any reference point. Confirm with **Continue**.
4. By default, the tool direction is the X axis. Move the tool so that the reference point on the X axis has a negative X value (i.e. move against the tool direction). Confirm with **Continue**.
5. Move the tool so that the reference point in the XY plane has a negative Y value. Confirm with **Continue**.
6. Press **Save**.

5.3.1.5 Numeric input

Description

The tool data can be entered manually.

Possible sources of data:

- CAD
- Externally calibrated tool
- Tool manufacturer specifications

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

Precondition

The following values are known:

- X, Y and Z relative to the FLANGE coordinate system
- A, B and C relative to the FLANGE coordinate system

Procedure

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

5.3.2 Fixed tool calibration

Overview

Calibration of a fixed tool consists of 2 steps:

Step	Description
1	**Calibration of the TCP of the fixed tool** The TCP of a fixed tool is called an external TCP. If the calibration data are already known, they can be entered directly. (>>> 5.3.2.1 "Calibration of an external TCP" page 86) (>>> 5.3.2.2 "Entering the external TCP numerically" page 88)
2	**Calibration of the workpiece** The following methods are available: ■ Direct method (>>> 5.3.2.3 "Workpiece calibration: direct method" page 88) ■ Indirect method (>>> 5.3.2.4 "Workpiece calibration: indirect method" page 89)

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

5.3.2.1 Calibration of an external TCP

Description

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

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

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

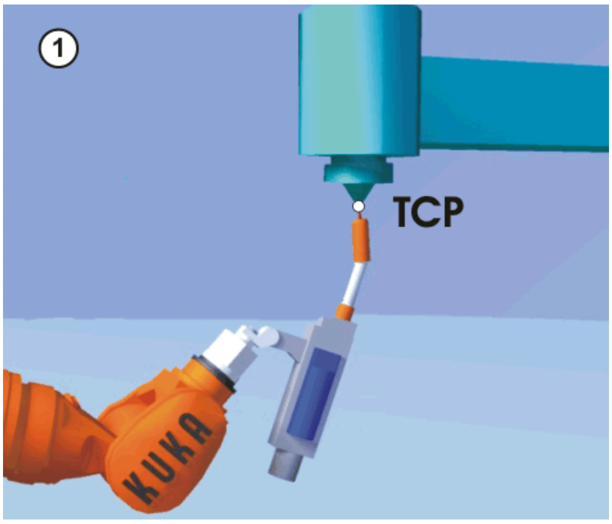


Fig. 5-13: Moving to the external TCP

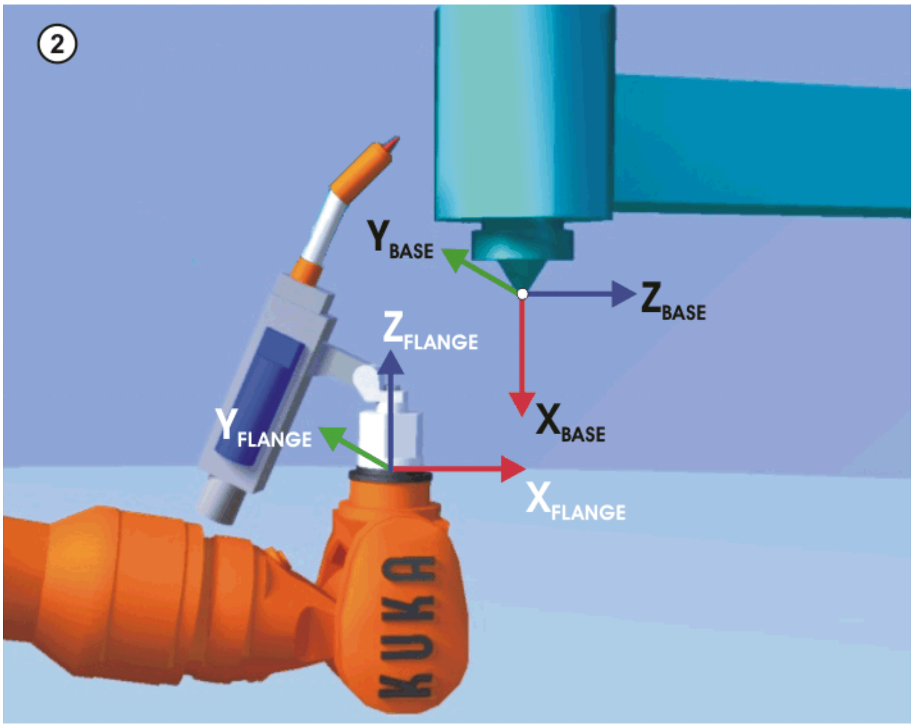


Fig. 5-14: Aligning the coordinate systems parallel to one another

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** > **Tool**.
2. Assign a number and a name for the fixed tool. 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 fixed 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.)

   If **6D** is selected:

   Align the mounting flange so that its axes are parallel to the axes of the fixed tool:

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

     (i.e. align the mounting flange perpendicular to the tool direction.)
   - $+Y_{BASE}$ 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**.

5.3.2.2 Entering the external TCP numerically

Precondition

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

- Distance between the TCP of the fixed tool and the origin of the WORLD coordinate system (X, Y, Z)
- Rotation of the axes of the fixed tool relative to the WORLD coordinate system (A, B, C)

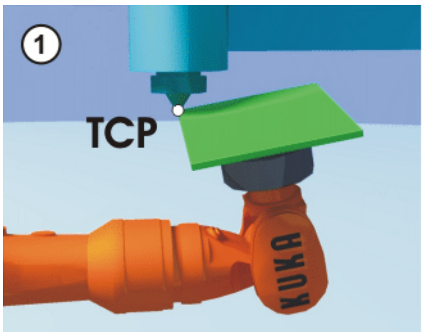
Procedure

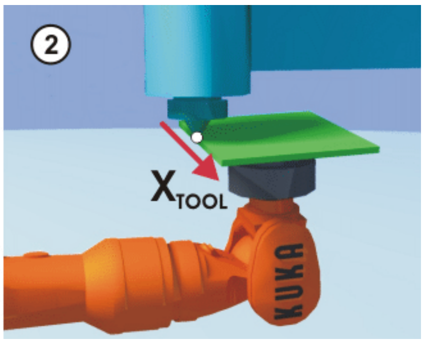
1. Select the menu **Setup** > **Measure** > **Fixed tool** > **Numeric Input**.
2. Assign a number and a name for the fixed tool. Confirm with **Continue**.
3. Enter data. Confirm with **Continue**.
4. Press **Save**.

5.3.2.3 Workpiece calibration: direct method

Description

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





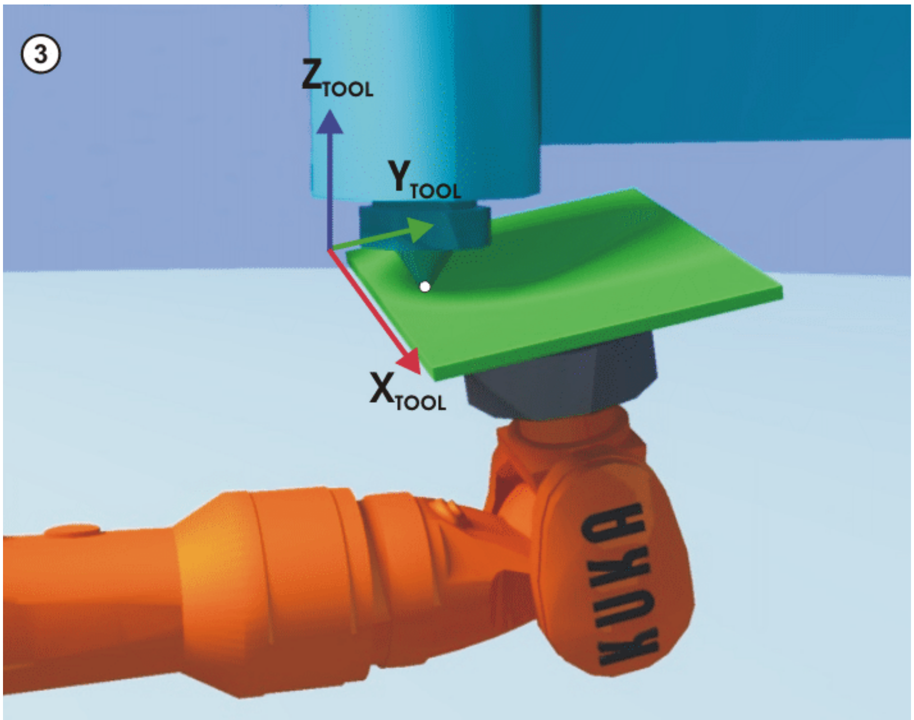


Fig. 5-15: Workpiece calibration: direct method

Precondition

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

Procedure

1. Select the menu **Setup** > **Measure** > **Fixed tool** > **Workpiece** > **Direct measuring**.
2. Assign a number and a name for the workpiece. Confirm with **Continue**.
3. Enter the number of the fixed tool. Confirm with **Continue**.
4. Move the origin of the workpiece coordinate system to the TCP of the fixed tool.

   Confirm with **Continue**.
5. Move a point on the positive X axis of the workpiece coordinate system to the TCP of the fixed tool.

   Confirm with **Continue**.
6. Move a point with a positive Y value in the XY plane of the workpiece co-ordinate system to the TCP of the fixed tool.

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

5.3.2.4 Workpiece calibration: indirect method

Description

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

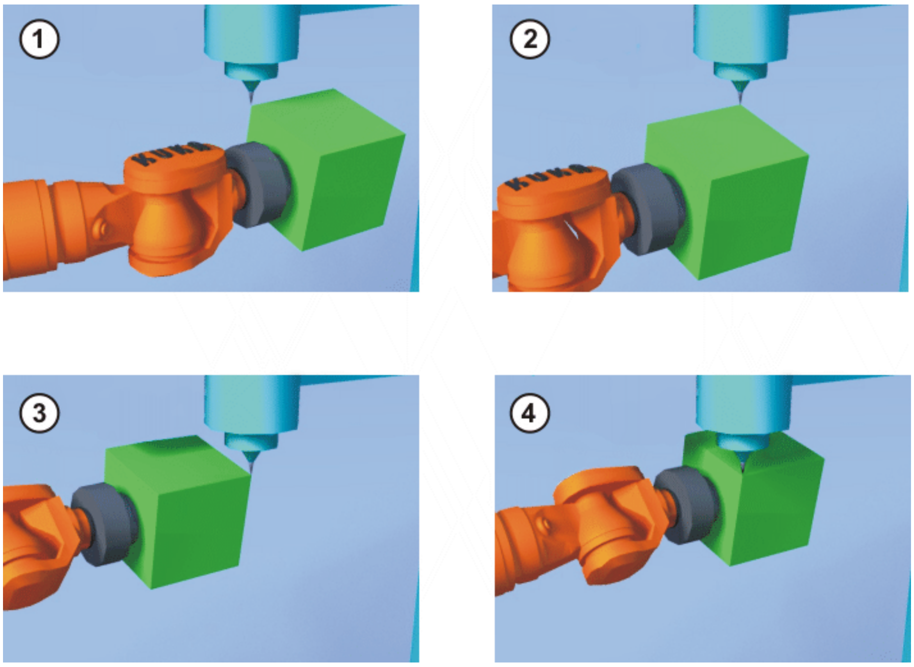


Fig. 5-16: Workpiece calibration: indirect method

Precondition

- A previously calibrated fixed tool is mounted.
- The workpiece to be calibrated is mounted on the mounting flange.
- The coordinates of 4 points of the new workpiece are known, e.g. from CAD data. The 4 points are accessible to the TCP.
- Operating mode T1 or T2.

Procedure

1. Select the menu **Setup** > **Measure** > **Fixed tool** > **Workpiece** > **Indirect measuring**.
2. Assign a number and a name for the workpiece. Confirm with **Continue**.
3. Enter the number of the fixed tool. Confirm with **Continue**.
4. Enter the coordinates of a known point on the workpiece and move this point to the TCP of the fixed tool. Confirm with **Continue**.
5. Repeat step 4 three times.
6. Press **Save**.

5.3.3 Base calibration

Description

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

> If the workpiece is mounted on the mounting flange, the type of calibration described here must not be used. A separate type of calibration must be used for workpieces mounted on the mounting flange. (>>> 5.3.2 "Fixed tool calibration" page 86)

Advantages of base calibration:

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

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

Overview

There are 2 ways of calibrating a base:

- 3-point method (>>> 5.3.3.1 "3-point method" page 91)
- Indirect method (>>> 5.3.3.2 "Indirect method" page 92)

If the calibration data are already known, they can be entered directly. (>>> 5.3.3.3 "Numeric input" page 93)

5.3.3.1 3-point method

Description

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

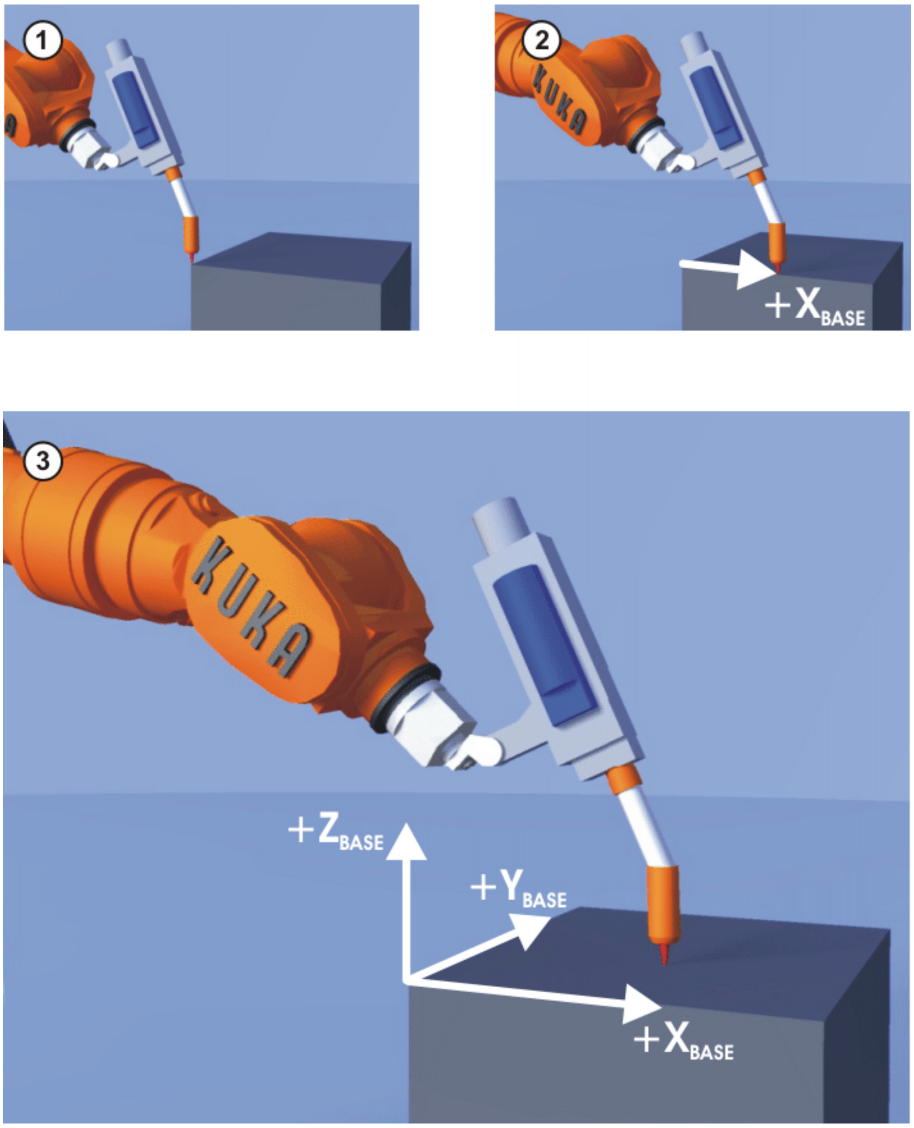


Fig. 5-17: 3-point method

Precondition

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

Procedure

1. Select the menu **Setup** > **Measure** > **Base** > **ABC 3-Point**.
2. Assign a number and a name for the base. Confirm with **Continue**.

3. Enter the number of the mounted tool. Confirm with **Continue**.
4. Move the TCP to the origin of the new base. Confirm with **Continue**.
5. Move the TCP to a point on the positive X axis of the new 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**.

5.3.3.2 Indirect method

Description

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

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

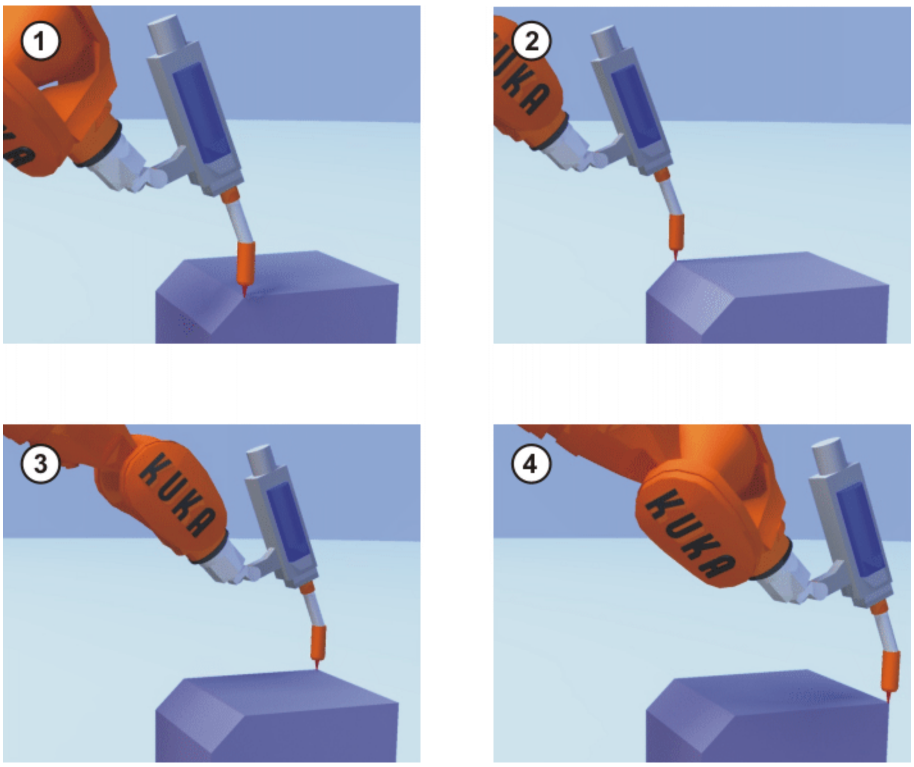


Fig. 5-18: Indirect method

Precondition

- A calibrated tool is mounted on the mounting flange.
- The coordinates of 4 points in the new base are known, e.g. from CAD data. The 4 points are accessible to the TCP.
- Operating mode T1 or T2.

Procedure

1. Select the menu **Setup** > **Measure** > **Base** > **Indirect**.
2. Assign a number and a name for the base. Confirm with **Continue**.
3. Enter the number of the mounted tool. Confirm with **Continue**.
4. Enter the coordinates of a known point in the new base and move the TCP to this point. Confirm with **Continue**.
5. Repeat step 4 three times.
6. Press **Save**.

5.3.3.3 Numeric input

Precondition

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

- Distance between the origin of the base and the origin of the WORLD coordinate system
- Rotation of the base axes relative to the WORLD coordinate system

Procedure

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

5.4 Load data

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

Warning!
If a robot is operated with incorrect load data or an unsuitable load, this can result in danger to life and limb and/or substantial material damage to the robot system.

Sources

Load data can be obtained from the following sources:

- Software option KUKA.LoadDetect (only for payloads)
- Manufacturer information
- Manual calculation
- CAD programs

5.4.1 Checking loads with KUKA.Load

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

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

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

5.4.2 Determining payloads with KUKA.Load Detect

KUKA.Load Detect can be used to calculate payloads exactly and transfer them to the robot controller. KUKA.Load Detect can only be used for payloads over 20% of the rated payload.

Functional principle: the payload is mounted on the robot. The mass, center of gravity and the mass inertia at the center of gravity are determined exactly by means of pendulum motions.

A payload can also be checked with KUKA.Load Detect, in a similar way to with KUKA.Load. If a sign-off sheet is to be created for the load, however, the load must be checked with KUKA.Load.

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

5.4.3 Entering payload data

Description

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

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

Precondition

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

Procedure

1. Select the menu **Setup** > **Measure** > **Tool** > **Payload data**.
2. Enter the number of the tool in the box **Tool no.**. Confirm with **Continue** softkey.
3. Enter the payload data:
   - In the box **M**, enter the mass.
   - In the boxes **X**, **Y**, **Z**, enter the values for the center of gravity.
   - In the boxes **A**, **B**, **C**, always enter the value *0.000*.
   - In the boxes **JX**, **JY**, **JZ**, enter the mass moments of inertia.
4. Confirm with **Continue** softkey.
5. Press **Save**.

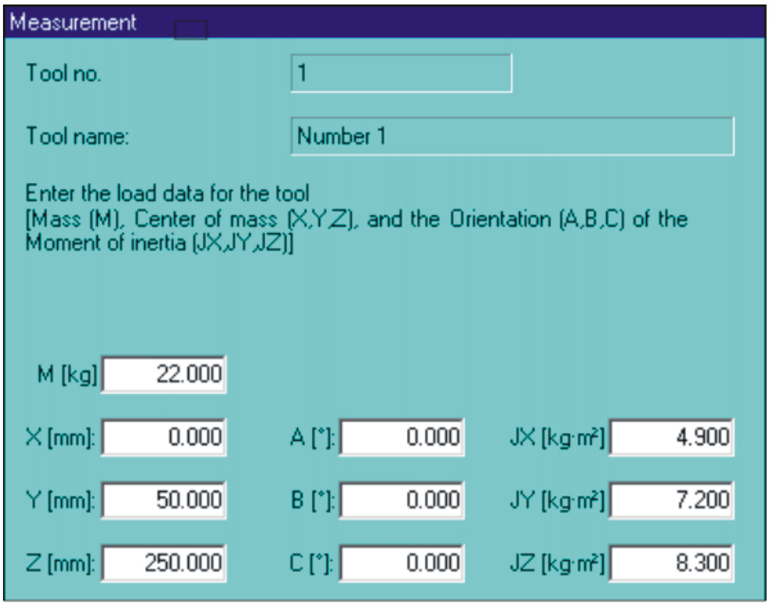


Fig. 5-19: Entering payload data

5.4.4 Entering supplementary load data

Description

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

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

Load	Reference system
Supplementary load A1	ROBROOT coordinate system A1 = 0°
Supplementary load A2	ROBROOT coordinate system A2 = -90°
Supplementary load A3	FLANGE coordinate system A4 = 0°, A5 = 0°, A6 = 0°

Precondition

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

Procedure

1. Select the menu sequence **Setup** > **Measure** > **Supplementary load data**.
2. Enter the number of the axis on which the supplementary load is to be mounted. Confirm with **Continue** softkey.
3. Enter the load data. Confirm with **Continue** softkey.
4. Press **Save**.

6 Program management

6.1 Navigator file manager

Overview

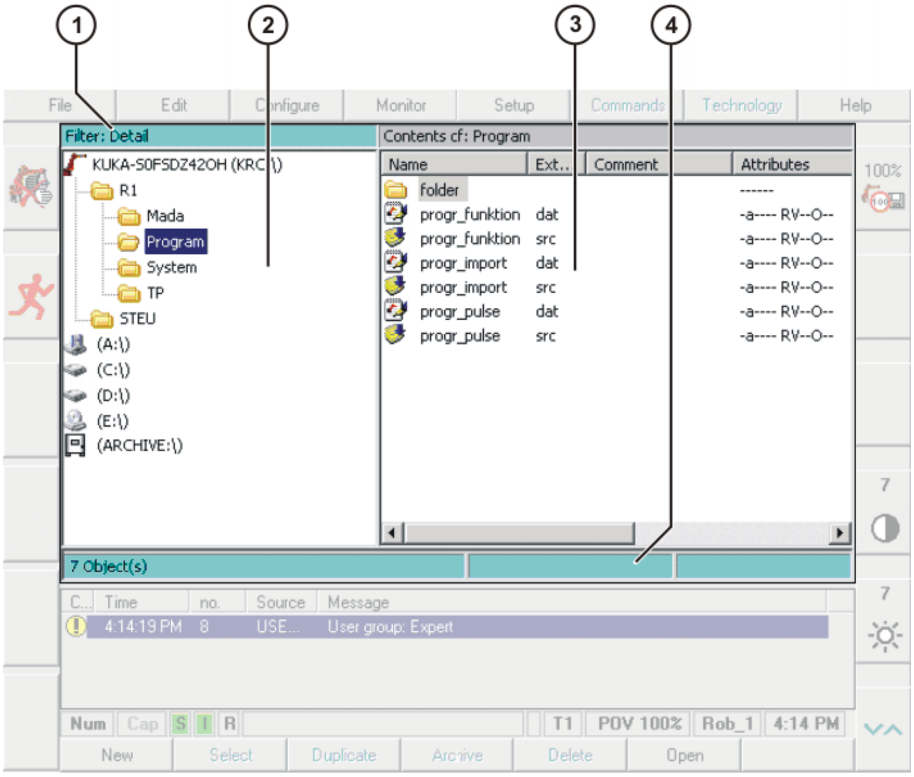


Fig. 6-1: Navigator

1 Header
2 Directory structure
3 File list
4 Status bar

Description

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

Header

- Left-hand area: the selected filter is displayed.
  (>>> 6.1.1 "Selecting filters" page 98)
- Right-hand area: the directory or drive selected in the directory structure is displayed.

Directory structure

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

File list

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

The file list has the following columns:

Column	Description
Name	Directory or file name
Extension	File extension This column is not displayed in the user group “User”.

Column	Description
Comment	Comment
Attributes	Attributes of the operating system and kernel system This column is not displayed in the user group “User”.
Size	File size in kilobytes This column is not displayed in the user group “User”.
#	Number of changes made to the file
Modified	Date and time of the last change
Created	Date and time of file creation This column is not displayed in the user group “User”.

The user can scroll left and right in the file list using the keyboard shortcuts SHIFT+RIGHT ARROW or SHIFT+LEFT ARROW.

Pop-up menus are available for the objects in the file list. Calling the pop-up menu: select object(s) and press the RIGHT ARROW key.

Status bar

The status bar can display the following information:

- Selected objects
- Action in progress
- User dialogs
- User entry prompts
- Requests for confirmation

6.1.1 Selecting filters

Description

This function is not available in the user group "User".

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

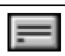
- **Detail**
  Programs are displayed as SRC and DAT files. (Default setting)
- **Modules**
  Programs are displayed as modules.

Procedure

1. Select the menu sequence **File** > **Filter**.
2. Select the desired filter in the left-hand section of the Navigator.
3. Confirm with **OK**.

6.1.2 Icons in the Navigator

Drives:

Icon	Description	Default path
	Robot	KRC:\
	Floppy disk	A:\
	Hard disk	e.g. "KUKADISK (C:\)" or "KUKADATA (D:\)"

Icon	Description	Default path
	CD-ROM	E:\
	Network drive	F:\, G:\, …
	Backup drive	Archive:\

Directories and files:

Icon	Description
	Directory
	Open directory
	Archive in ZIP format
	The contents of a directory are being read.
	Module
	Module containing errors
	SRC file
	SRC file containing errors
	DAT file
	DAT file containing errors
	ASCII file. Can be read using any editor.
	Binary file. Cannot be read in the text editor.

6.1.3 Creating a new folder

Precondition

- The Navigator is displayed.

Procedure

1. In the directory structure, use the UP and DOWN arrow keys to select the folder in which the new folder is to be created.

   Closed folders can be opened by pressing the Enter key.
2. Press the **NEW** softkey.
3. Enter a name for the folder and press **OK**.

6.1.4 Creating a new program

Precondition

- The Navigator is displayed.

Procedure

1. In the directory structure, use the UP and DOWN arrow keys to select the folder in which the program is to be created.

Closed folders can be opened by pressing the Enter key.

2. Move to the file list by pressing the RIGHT arrow button.
3. Press the **New** softkey.

   The **Template selection** window is opened.
4. Select the desired template and press **OK**.
5. Enter a name for the program and press the softkey **OK**.

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

6.1.5 Renaming a file

Precondition

- The Navigator is displayed.

Procedure

1. In the directory structure, use the UP and DOWN arrow keys to select the folder in which the file is located.

   Closed folders can be opened by pressing the Enter key.
2. Move to the file list by pressing the RIGHT arrow button. Select the desired file.
3. Select the menu sequence **File** > **Rename**.
4. Overwrite the file name with a new name and press **OK**.

6.1.6 Toggling between the Navigator and the program

Description

If a program is selected, it is possible to toggle to the Navigator without having to deselect the program. The user can then return to the program.

Procedure

- Toggling from the program to the Navigator: press the **NAVIGATOR** softkey.
- Toggling from the program to the Navigator: press the **PROGRAM** softkey.

6.2 Selecting and deselecting a program

Description

Before a program can be started or edited, it must first be selected. Following program execution or editing, it must be deselected again.

Procedure

1. Select the program in the Navigator.

   If the program is displayed as a SRC file and a DAT file, the SRC file or the DAT file can be selected.
2. Press the **Select** softkey.
3. Execute or edit the program.

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

4. Select the menu sequence **Program** > **Cancel program**.

6.3 Structure of a KRL program

```
1  DEF my_program( )
2  INI
3
4  PTP HOME  Vel= 100 % DEFAULT
   ...
8  LIN point_5 CONT Vel= 2 m/s CPDAT1 Tool[3] Base[4]
   ...
14 PTP point_1 CONT Vel= 100 % PDAT1 Tool[3] Base[4]
   ...
20 PTP HOME  Vel= 100 % DEFAULT
21
22 END
```

Line	Description
1	The DEF line indicates the name of the program. If the program is a function, the DEF line begins with "DEFFCT" and contains additional information. The DEF line can be displayed or hidden. Select the menu sequence **Configure** > **Tools** > **Editor** > **Def-line**. This function is not available in the user group "User".
2	The INI line contains initializations for internal variables and parameters.
4	HOME position (>>> 6.3.1 "HOME position" page 101)
8	LIN motion (>>> 8.2.3 "Programming a LIN motion" page 130)
14	PTP motion (>>> 8.2.1 "Programming a PTP motion" page 129)
20	HOME position
22	The END line is the last line in any program. If the program is a function, the wording of the END line is "ENDFCT". The END line must not be deleted!

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

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

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

“Complete” means that all components of the end point must be specified.

Warning!
If a HOME position is modified, this affects all programs in which it is used. Physical injuries or damage to property may result.

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

6.3.1 HOME position

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

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

Axis	A1	A2	A3	A4	A5	A6
Item	0°	- 90°	+ 90°	0°	0°	0°

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

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

Warning!
If a HOME position is modified, this affects all programs in which it is used. Physical injuries or damage to property may result.

6.4 Displaying/hiding program sections

6.4.1 Displaying/hiding the DEF line

Description

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

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

Precondition

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

Procedure

- Select the menu sequence **Configure** > **Tools** > **Editor** > **Def-line**.
  Check mark activated in menu: DEF line is displayed.
  Check mark not activated in menu: DEF line is hidden.

6.4.2 Activating detail view (ASCII mode)

Description

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

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

Precondition

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

Procedure

- Select the menu sequence **Configure** > **Tools** > **Editor** > **ASCII Mode**.
  Check mark activated in menu: ASCII mode is activated.
  Check mark not activated in menu: ASCII mode is deactivated.

6.4.3 Activating/deactivating the line break function

Description

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

```
25   INTERRUPT DECL 15 WHEN
   ↳ $MEAS_PULSE[TOUCH_I[TOUCH_ACTIVE].IN_NR] DO H70 (6,CD0 )
```

Fig. 6-2: Line break

The line break function is activated and deactivated separately for opened and selected programs.

Precondition

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

Procedure

- Select the menu sequence **Configure** > **Tools** > **Editor** > **Linebreak**.
  Check mark activated in menu: line break function is activated.
  Check mark not activated in menu: line break function is deactivated.

6.5 Starting a program

6.5.1 Program run modes

The program run mode is selected in the left-hand status key bar.

Status key	Program run mode	Description
	GO	The program is executed through to the end without stopping.
	MSTEP (Motion Step)	The program is executed with a stop after each motion block. The Start key must be pressed again for each motion block.
	ISTEP (Incremental Step)	The program is executed with a stop after each program line. Program lines that cannot be seen and blank lines are also taken into consideration. The Start key must be pressed again for each line. ISTEP is only available to the user group "Expert".
	Backward motion	This program run mode is automatically selected if the Start backwards key is pressed.

In MSTEP and ISTEP modes, the program is executed without an advance run.

6.5.2 Advance run

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

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

6.5.3 Icons in the program

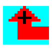
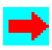
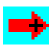
Line break

If a line is wider than the program window, the line is broken by default. The part of the line after the break has no line number and is marked with a black, L-shaped arrow. The line break function can be deactivated.
(>>> 6.4.3 "Activating/deactivating the line break function" page 102)

Fig. 6-3: Line break

Block pointer

During program execution, the block pointer indicates which motion block is currently being executed.

Icon	Description
	L-shaped arrow (yellow): The motion block is being executed in the forwards direction.
	L-shaped arrow (yellow) with plus sign: The motion block is being executed in the forwards direction. This block pointer is not displayed in the user group "User".
	Normal arrow (yellow): The robot has completed the motion block in the forwards direction
	Normal arrow (yellow) with plus sign: The robot has completed the motion block in the forwards direction This block pointer is not displayed in the user group "User".
	L-shaped arrow (red): The motion block is being executed in the backwards direction.
	L-shaped arrow (red) with plus sign: The motion block is being executed in the backwards direction. This block pointer is not displayed in the user group "User".
	Normal arrow (red): The robot has completed the motion block in the backwards direction
	Normal arrow (red) with plus sign: The robot has completed the motion block in the backwards direction This block pointer is not displayed in the user group "User".

Icon	Description
	The block pointer is located higher up in the program.
	The block pointer is located lower down in the program.

6.5.4 Setting the program override (POV)

Description

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

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

Preparation

- Define the program override intervals:

  Select the menu sequence **Configure** > **Jogging** > **Program OV Steps**.

Active	Meaning
No	The override can be adjusted in 1% steps.
Yes	Intervals: 100%, 75%, 50%, 30%, 10%, 3%, 1%, 0%

Procedure

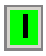
- Increase or reduce the override in the right-hand status key bar. The status key indicates the current override as a percentage.

6.5.5 Starting a program forwards (manual)

Precondition

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

Procedure

1. Select the program run mode.
2. Hold the enabling switch down and wait until the status bar indicates  (i.e. drives ready).
3. Carry out a BCO run:

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

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

4. Press Start key and hold it down.

   The program is executed with or without stops, depending on the program run mode.

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

6.5.6 Starting a program forwards (automatic)

Precondition

- Program is selected.
- Operating mode Automatic (**not** Automatic External)

Procedure

1. Select the program run mode GO in the left-hand status key bar:

2. Press **Drives ON**.
3. Carry out a BCO run:

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

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

4. Press the Start key. Program is executed.

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

6.5.7 Carrying out a block selection

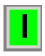
Description

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

Precondition

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

Procedure

1. Select the program run mode.
2. Position the cursor in the line containing the motion block at which the program is to be started.
3. Press the **Line Sel.** softkey. The yellow block pointer indicates the motion block.
4. Hold the enabling switch down and wait until the status bar indicates  (i.e. drives ready).
5. Carry out a BCO run: Press Start key and hold it down until the message “*Programmed path reached (BCO)*” is displayed in the message window. The robot stops.

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

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

6.5.8 Starting a program backwards

Description

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

Exactly how the controller responds during backward motion depends on the configuration.

Precondition

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

Procedure

1. Hold the enabling switch down and wait until the status bar indicates  (i.e. drives ready).
2. Carry out a BCO run:

   Press Start key and hold it down until the message "*Programmed path reached (BCO)*" is displayed in the message window. The robot stops.

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

3. Press Start backwards key. The program run mode "Backward motion" is automatically selected:

4. Press Start backwards key again for each motion block.

6.5.9 Resetting a program

Description

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

Precondition

- Program is selected.

Procedure

- Select the menu sequence **Program** > **Reset program**.

6.5.10 Starting Automatic External mode

Precondition

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

Procedure

1. Select the program CELL.SRC in the Navigator. (This program is located in the folder "R1".)
2. Set program override to 100%. (This is the recommended setting. A different value can be set if required.)
3. Carry out BCO run:

   Hold down the enabling switch. Then press the Start key and hold it down until the message "Programmed path reached (BCO)" is displayed in the message window.

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

4. Turn the mode selector switch to "Automatic External".

5. Start the program from a higher-level controller (PLC).

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

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

6.6 Editing a program

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

6.6.1 Inserting a comment or stamp

Precondition

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

Procedure

1. Position the cursor in the line **after** which the comment or stamp is to be inserted.
2. Select the menu sequence **Commands** > **Comment** > **Normal** or **Stamp**.
3. In the case of Stamp: update the system time by pressing the **New time** softkey.
4. Enter text.
5. Save by pressing the **Cmd Ok** softkey.

Comment description

Fig. 6-4: Inline form "Comment"

Item	Description
1	Any text

Stamp description

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

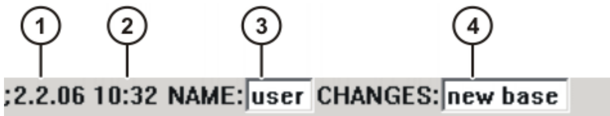


Fig. 6-5: Inline form "Stamp"

Item	Description
1	Current system date (cannot be edited)
2	Current system time
3	Name or ID of the user
4	Any text

6.6.2 Deleting program lines

Precondition

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

Procedure

1. Position the cursor in the line to be deleted.

   If several consecutive lines are to be deleted:

   Position the cursor in the first line. Then press SHIFT + DOWN ARROW until all the lines are selected.
2. Select the menu sequence **Program** > **Delete**.
3. Confirm the request for confirmation with **Yes**.

> If a program line containing a motion instruction is deleted, the point name and coordinates remain saved in the DAT file. The point can be used in other motion instructions and does not need to be taught again.

> Lines cannot be restored once they have been deleted!

6.6.3 Additional editing functions

The following additional program editing functions can be found in the **Program** menu:

- **Copy**
- **Paste**
- **Cut**
- **Find**
- **Replace**

6.7 Printing a program

Procedure

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

6.8 Archiving

6.8.1 Destination for archiving

Overview

Depending on the specific robot controller, various destinations are available for archiving in KSS 5.5.

Robot controller	Destination
KR C2	■ Floppy disk (default configuration) ■ Network path
KR C2 edition2005	■ Floppy disk (default configuration) Floppy disk drive can optionally be installed. ■ Writable USB media ■ Network path

Caution!
The only USB stick that may be used is the KUKA USB stick. Data may be lost or modified if any other USB stick is used.

6.8.2 Formatting the floppy disk

Procedure

1. Insert floppy disk into the disk drive.
2. Select the menu sequence **File** > **Format floppy disk**.
3. Confirm the request for confirmation with **Yes**.

   A message indicates completion of the formatting process.

Caution!
Do not remove the floppy disk from the drive until the LED on the drive is no longer lit. Otherwise the disk drive and/or the floppy disk could suffer damage.

6.8.3 Archiving

Precondition

For archiving to **floppy**:

- Disk has been formatted (if required).
- Disk is in disk drive.

More than one floppy disk will be required if large quantities of data are involved!

For archiving to a writable **USB medium**:

- The path has been configured in the KRC Configurator.
- The USB medium is connected.

For archiving to a **network path**:

- The path has been configured in the KRC Configurator.

Procedure

1. Select the menu sequence **File** > **Archive** and the desired menu item.
2. Confirm the request for confirmation with **Yes**.
3. Only in the case of archiving to a floppy disk: a message indicates when another disk is required. Insert new disk.
4. A message indicates completion of the archiving process. The file ARCHIVE.ZIP is generated by default.

Caution!
Only in the case of archiving to a USB medium: the medium must not be removed until the LED on the USB medium is no longer lit. Otherwise, the medium could be damaged.

6.8.4 Menu item “Archive”

Description

The following menu items are available for archiving. Exactly which files are archived depends on the configuration in the KRC Configurator.

Menu item	Description
All	The data that are required to restore an existing system are archived.
Applications	All user-defined KRL modules and their corresponding system files are archived.
Machine data	The machine data are archived.

Menu item	Description
Configuration > Drivers	The I/O drivers are archived. Not available in the user group “User”.
Configuration > I/O Longtexts	The long text names of the inputs/outputs are archived. Not available in the user group “User”.
Configuration > KUKA Tech-Pack	The configuration of the installed technology packages is archived. Not available in the user group “User”.
Log Data	The log files are archived.
Current selection	The files selected in the Navigator are archived.

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

6.8.5 Restoring data

Overview

Caution!

Only KSS 5.5 archives may be loaded into KSS 5.5. If other archives are loaded, the following may occur:

- Error messages
- Robot controller is not operable.
- Personal injury and damage to property.

The robot controller cannot detect whether multiple floppy disks were used for archiving. For this reason, the user is not prompted, after the first floppy disk, to insert additional disks, but must do so without prompting. The order in which the disks are inserted is irrelevant.

If the archive files are not the same version as the system files, an error message is generated. Similarly, if the version of the archived technology packages does not match the installed version, an error message is generated.

6.8.5.1 Restoring data via the menu

Description

With the exception of **Log Data**, the menu items available for restoring data are the same as those available for archiving.

When restoring data, the robot controller accesses the path configured for archiving. For example, if the default configuration (floppy disk path) has been left, data will also be restored from the floppy disk.

Precondition

If data were archived to **floppy**:

- Disk is in disk drive.

If data were archived to **USB medium**:

- The USB medium is connected.
- The path has been configured in the KRC Configurator.

Procedure

1. Select the menu sequence **File** > **Restore** and the desired menu item.
2. Confirm the request for confirmation with **Yes**.

   A message indicates completion of the restoration process.
3. Only if restoring data from floppy: if data have been archived on more than one disk, insert the next floppy disk and repeat steps 1 and 2.
4. Only if restoring data from floppy: remove floppy from floppy drive.

Caution!
Only in the case of restoring data from a USB medium: the medium must not be removed until the LED on the USB medium is no longer lit. Otherwise, the medium could be damaged.

5. Reboot the robot controller.

6.8.5.2 Restoring data via softkey

Description

When restoring data using the softkey, a different path can be selected from that to which the data were archived.

Precondition

- The path from which data are to be restored must be configured in the KRC Configurator.

If data were archived to **floppy**:

- Disk is in disk drive.

If data were archived to **USB medium**:

- The USB medium is connected.

Procedure

1. Select the drive **(ARCHIVE:\)** in the Navigator. The directories belonging to **(ARCHIVE:\)** are displayed.
2. Select the desired object in a directory.
3. Press the **Restore** softkey.
4. Confirm the request for confirmation with **Yes**. A message indicates completion of the restoration process.
5. Only if restoring data from floppy: if data have been archived on more than one disk, insert the next floppy disk and repeat steps 1 to 4.
6. Only if restoring data from floppy: remove floppy from floppy drive.

Caution!
Only in the case of restoring data from a USB medium: the medium must not be removed until the LED on the USB medium is no longer lit. Otherwise, the medium could be damaged.

7. Reboot the robot controller.

The **Restr. All** softkey ignores which objects are selected in the Navigator and accesses the path that is configured for archiving.

7 Basic principles of motion programming

7.1 Motion types

Overview

The following motion types can be programmed:

- Point-to-point motions (PTP)
  (>>> 7.1.1 "Motion type PTP" page 113)
- Linear motions (LIN)
  (>>> 7.1.2 "Motion type LIN" page 114)
- Circular motions (CIRC)
  (>>> 7.1.3 "Motion type CIRC" page 114)
- Spline motions
  (>>> 7.1.4 "Motion type SPLINE" page 115)

LIN, CIRC and spline motions are also known as CP (“Continuous Path”) motions.

The start point of a motion is always the end point of the previous motion.

7.1.1 Motion type PTP

The robot guides the TCP along the fastest path to the end point. The fastest path is generally not the shortest path and is thus not a straight line. As the motions of the robot axes are rotational, curved paths can be executed faster than straight paths.

The exact path of the motion cannot be predicted.

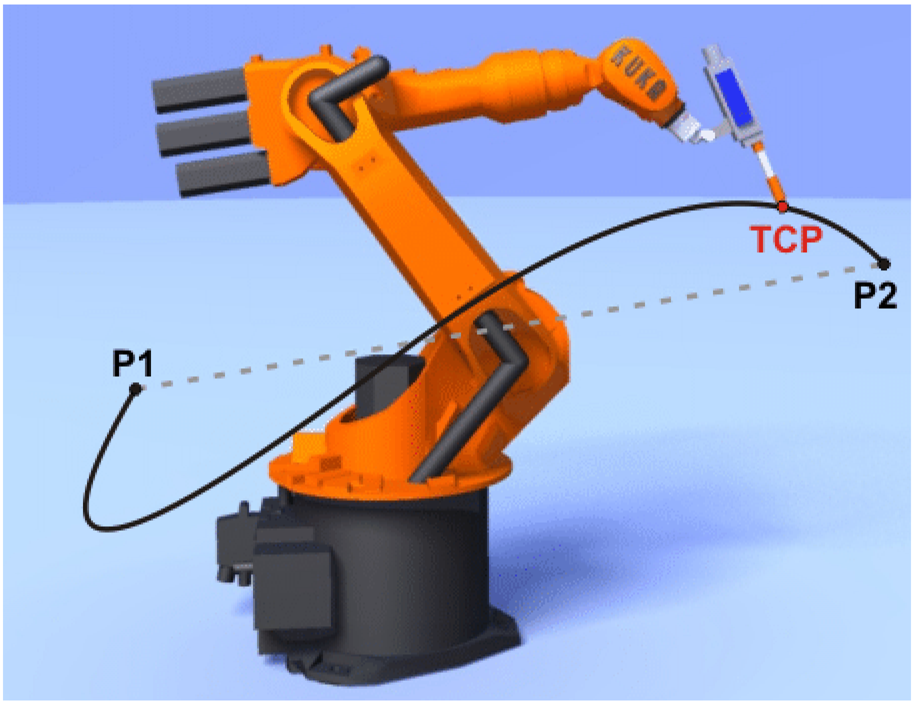


Fig. 7-1: PTP motion

7.1.2 Motion type LIN

The robot guides the TCP at a defined velocity along a straight path to the end point.

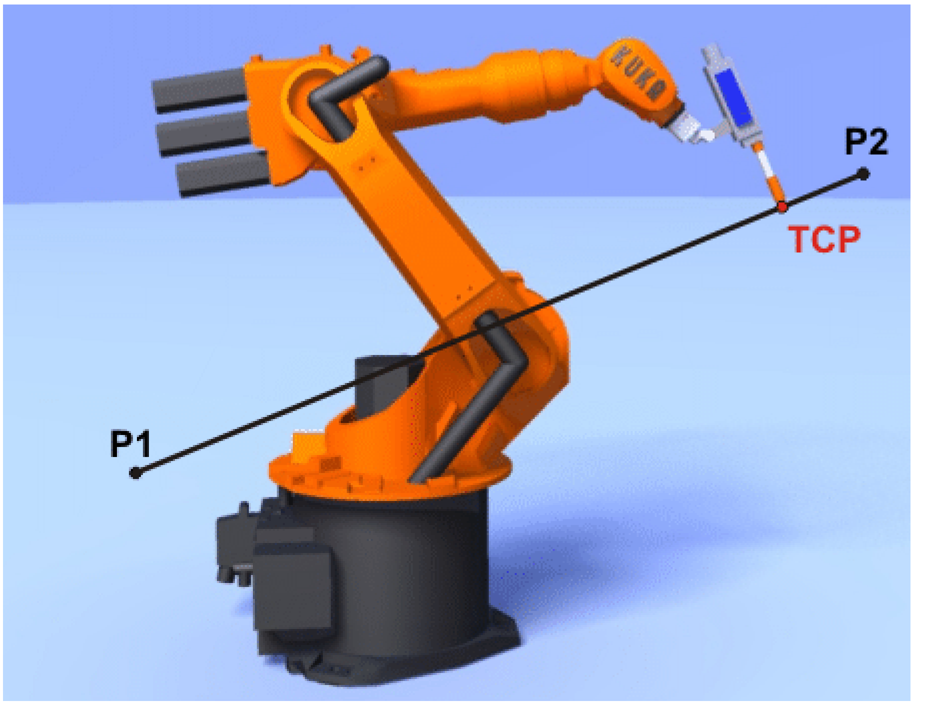


Fig. 7-2: LIN motion

7.1.3 Motion type CIRC

The robot guides the TCP at a defined velocity along a circular path to the end point. The circular path is defined by a start point, auxiliary point and end point.

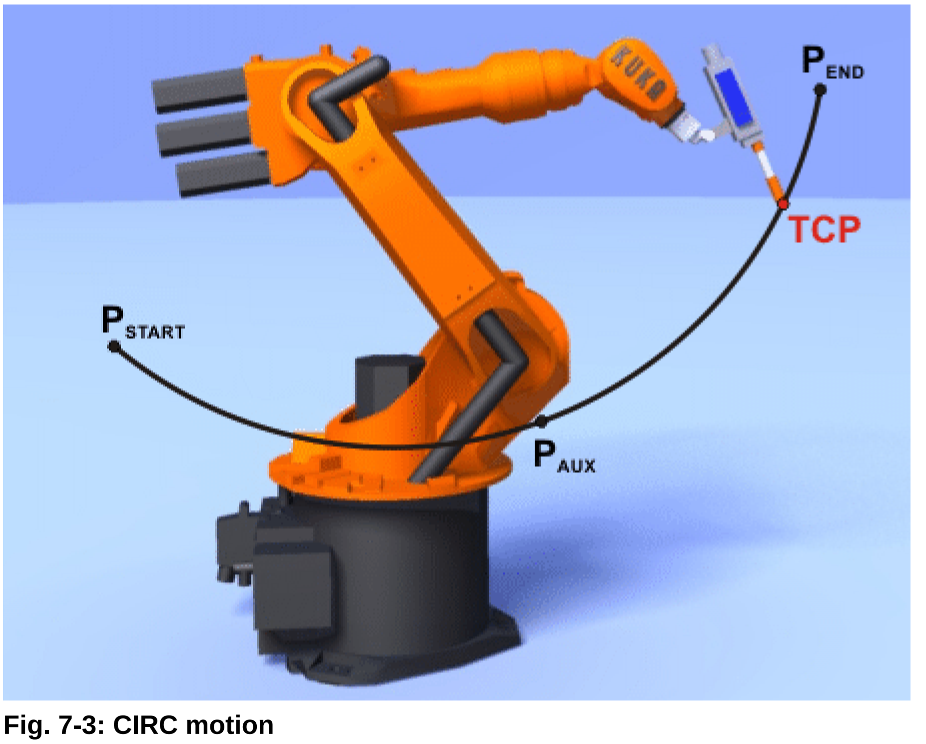


Fig. 7-3: CIRC motion

7.1.4 Motion type SPLINE

SPLINE is a Cartesian motion type that is suitable for particularly complex, curved paths.

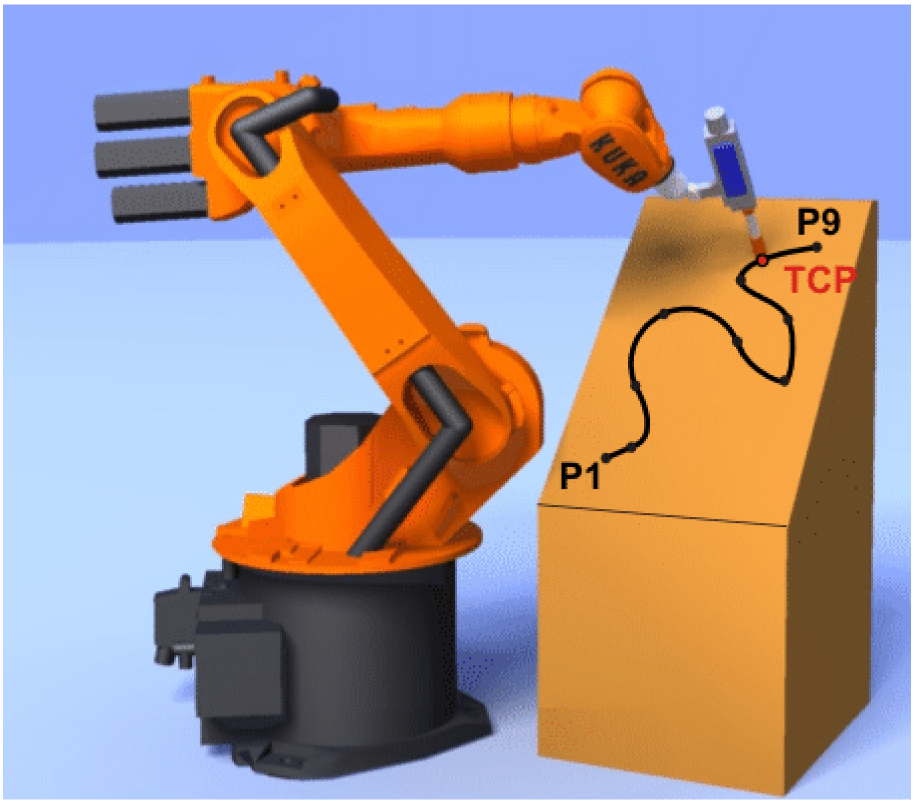


Fig. 7-4: Spline motions

Characteristics:

- The path always remains the same, irrespective of the override setting, velocity or acceleration. The same path is also executed in the program run mode MSTEP.
- Circles and tight radii are executed with great precision.
- A spline motion consists of several individual motions: spline segments. These are taught separately. The segments are grouped together to form the overall motion in a so-called spline block.

  A spline block is planned and executed by the robot controller as a single motion block.
- Tool, base, interpolation mode, load data: these parameters are programmed for the entire spline motion.
- Velocity, acceleration, orientation control, jerk limitation: these parameters are programmed for the entire spline motion. They can also be programmed separately for individual segments.
- All points are passed through (no approximate positioning)
- If all points are situated on a plane, then the path is also situated in this plane.
- If all points are situated on a straight line, then the path is also a straight line.

7.1.4.1 Velocity profile for spline motions

The path always remains the same, irrespective of the override setting, velocity or acceleration. Only dynamic effects can cause deviations at different velocities.

The programmed acceleration is valid not only for the direction along the path, but also perpendicular to the path. The same applies to the jerk limitation. Effects include the following:

- In the case of circles, the centrifugal acceleration is taken into consideration. The velocity that can be achieved thus also depends on the programmed acceleration and the radius of the circle.
- At corners, the maximum permissible velocity is derived from the radius of the curve, the acceleration and the jerk limitation.

Reduction of the velocity

In the case of spline motions, the velocity may, under certain circumstances, fall below the programmed velocity. This occurs particularly in the case of:

- Tight corners
- Major reorientation
- Large motions of the external axes

If the points are close together, the velocity is **not** reduced.

Exact positioning

The following cases trigger exact positioning:

- Successive points with the same Cartesian coordinates
- Successive SLIN and/or SCIRC segments. Cause: inconstant velocity direction.

  In the case of SLIN-SCIRC transitions, exact positioning is also carried out if the straight line is a tangent of the circle.

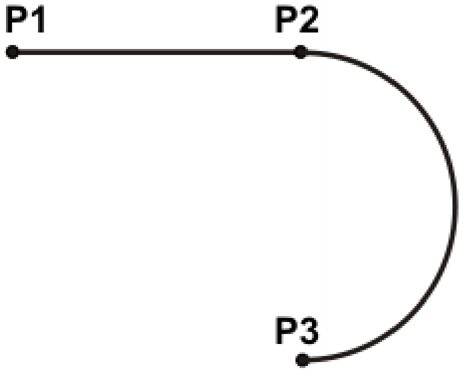


Fig. 7-5: Exact positioning at P2

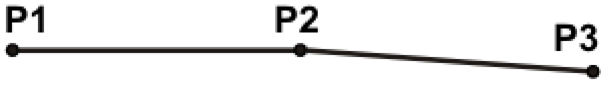


Fig. 7-6: Exact positioning at P2

Exceptions:

- If successive SLIN segments result in a straight line, the straight line is executed without exact positioning.

Fig. 7-7: P2 is executed without exact positioning.

- A SCIRC-SCIRC transition does not trigger exact positioning if both circles have the same center point and the same radius. (This is difficult to teach, so work with calculated points.) This means that circles ≥ 360° are possible.

7.1.4.2 Block selection with spline motions

Description

A spline block is planned and executed by the robot controller as a single motion block. Block selection to the spline segments is nonetheless possible. The BCO run is executed as a LIN motion. This is indicated by means of a message that must be acknowledged.

If the second segment in the spline block is an SPL segment, a modified path is executed in the following cases:

- Block selection to the first segment in the spline block
- Block selection to the spline block
- Block selection to a line before the spline block if this does not contain a motion instruction and if there is no motion instruction before the spline block

If the Start key is pressed after the BCO run, the modified path is indicated by means of a message that must be acknowledged.

Example

```
1  PTP P0
2  SPLINE
3   SPL P1
4   SPL P2
5   SPL P3
6   SPL P4
7   SCIRC P5, P6
8   SPL P7
9   SLIN P8
10 ENDSPLINE
```

Line	Description
2	Start of the spline block
3 … 9	Spline segments
10	End of the spline block

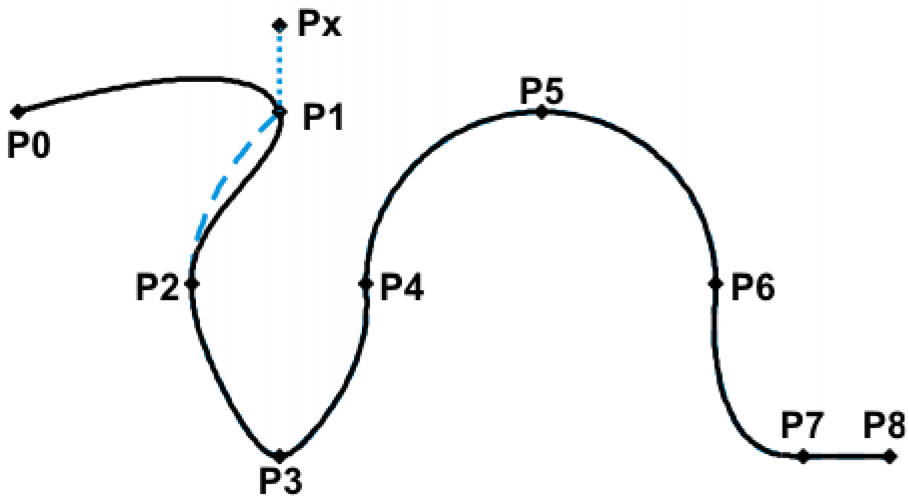


Fig. 7-8: Example: modified path in the case of block selection to P1

7.1.4.3 Modifications to spline motions

Description

- Modification of the position of the point:

  If a point within a spline block is offset, the path is modified, at most, in the 2 segments before this point and the 2 segments after it.

  Small point offsets generally result in small modifications to the path. If, however, very long segments are followed by very short segments or vice versa, small modifications can have a very great effect, as the tangents and curves can change very greatly in such cases.
- Modification of the segment type:

  If an SPL segment is changed into an SLIN segment or vice versa, the path changes in the previous segment and the next segment.

Example 1

```
PTP P0
SPLINE
 SPL P1
 SPL P2
 SPL P3
 SPL P4
 SCIRC P5, P6
 SPL P7
 SLIN P8
ENDSPLINE
```

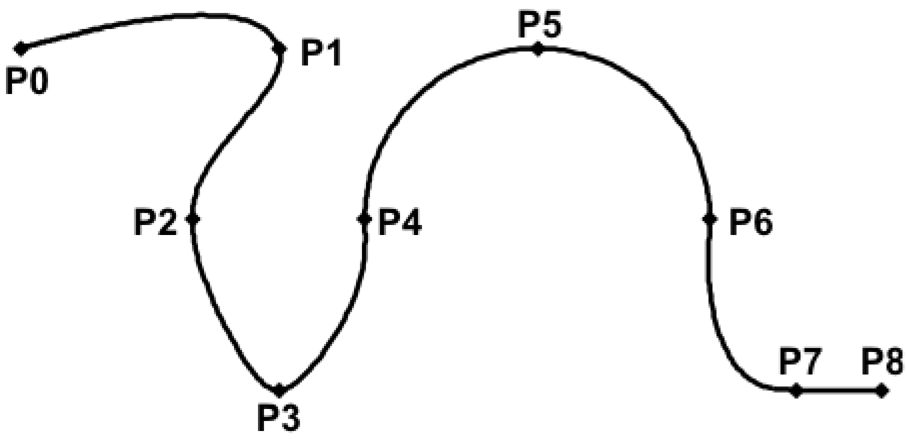


Fig. 7-9: Example 1: original path

P3 is offset. This causes the path to change in segments P1 - P2, P2 - P3 and P3 - P4. Segment P4 - P5 is not changed in this case, as it belongs to an SCIRC and a circular path is thus defined.

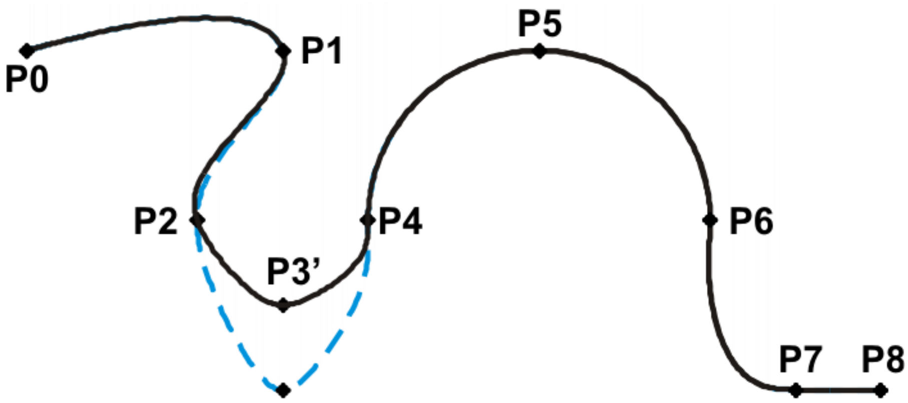


Fig. 7-10: Example 1: point offset

In the original path, the segment type of P2 - P3 is changed from SPL to SLIN. The path changes in segments P1 - P2, P2 - P3 and P3 - P4.

```
PTP P0
SPLINE
 SPL P1
 SPL P2
 SLIN P3
 SPL P4
 SCIRC P5, P6
 SPL P7
 SLIN P8
ENDSPLINE
```

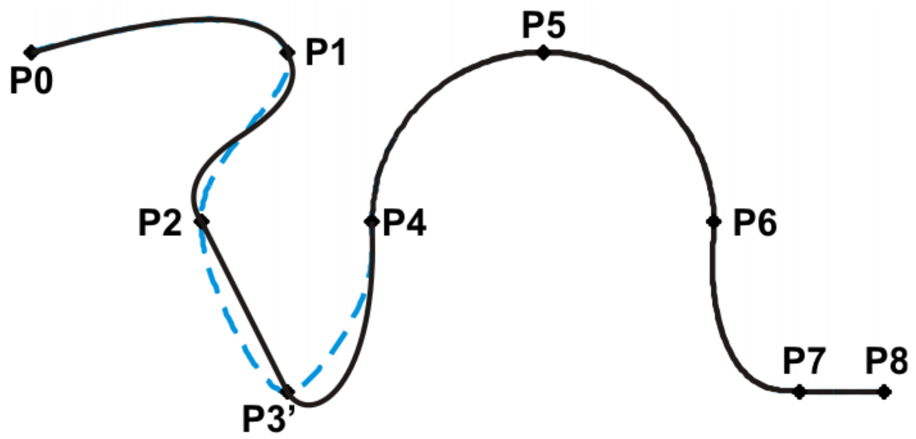


Fig. 7-11: Example 1: segment type changed

Example 2

```
...
SPLINE
 SPL {X 100, Y 0, ...}
 SPL {X 102, Y 0}
 SPL {X 104, Y 0}
 SPL {X 204, Y 0}
ENDSPLINE
```

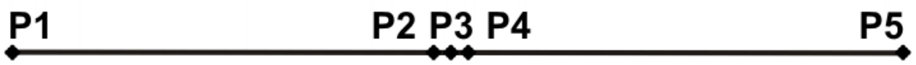


Fig. 7-12: Example 2: original path

P3 is offset. This causes the path to change in all the segments illustrated. Since P2 - P3 and P3 - P4 are very short segments and P1 - P2 and P4 - P5 are long segments, the slight offset causes the path to change greatly.

```
...
SPLINE
 SPL {X 100, Y 0, ...}
 SPL {X 102, Y 1}
 SPL {X 104, Y 0}
 SPL {X 204, Y 0}
ENDSPLINE
```

Fig. 7-13: Example 2: point offset

Remedy:

- Distribute the points more evenly
- Program straight lines (except very short ones) as SLIN segments

7.2 Approximate positioning

Approximate positioning means that the motion does not stop exactly at the programmed point. Approximate positioning is an option that can be selected during motion programming.

> Approximate positioning is not possible if the motion instruction is followed by an instruction that triggers an advance run stop.

PTP motion

The TCP leaves the path that would lead directly to the end point and moves along a faster path. During programming of the motion, the maximum distance from the end point at which the TCP may deviate from its original path is defined.

The path of an approximated PTP motion cannot be predicted. It is also not possible to predict which side of the approximated point the path will run.

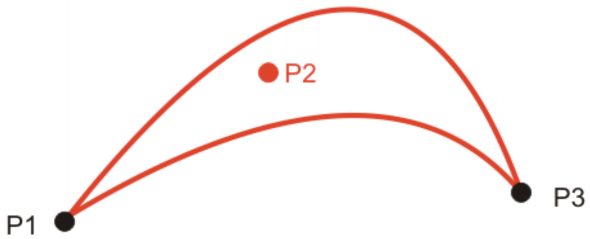


Fig. 7-14: PTP motion, P2 is approximated

LIN motion

The TCP leaves the path that would lead directly to the end point and moves along a shorter path. During programming of the motion, the maximum distance from the end point at which the TCP may deviate from its original path is defined.

The path in the approximate positioning range is **not** an arc.

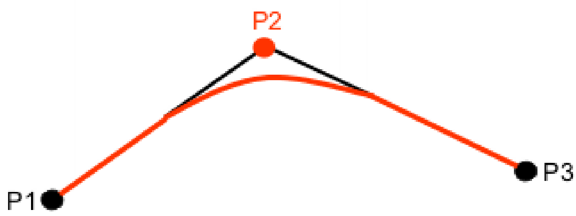


Fig. 7-15: LIN motion, P2 is approximated

CIRC motion

The TCP leaves the path that would lead directly to the end point and moves along a shorter path. During programming of the motion, the maximum distance from the end point at which the TCP may deviate from its original path is defined.

The motion always stops exactly at the auxiliary point.

The path in the approximate positioning range is **not** an arc.

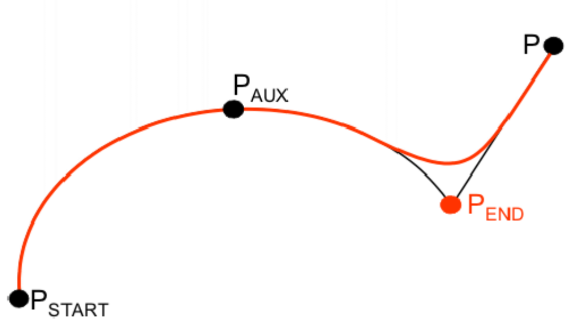


Fig. 7-16: CIRC motion, P_{END} is approximated

7.3 Orientation control

The orientation of the TCP can be different at the start point and end point of a motion. There are several different types of transition from the start orientation to the end orientation. A type must be selected when a CP motion is programmed.

7.3.1 Orientation control LIN, CIRC

The orientation control for LIN and CIRC motions is defined as follows:

- In the option window **Motion parameter** (CP motion), **Orientation control** box

  (>>> 8.2.9 "Option window "Motion parameter" (CP motion)" page 134)

LIN motion

Orientation control	Description
Constant orientation	The orientation of the TCP remains constant during the motion. The programmed orientation is disregarded for the end point and that of the start point is retained.
Standard	The orientation of the TCP changes continuously during the motion. **Note:** If, with **Standard**, the robot passes through a wrist axis singularity, use **Wrist PTP** instead.
Wrist PTP	The orientation of the TCP changes continuously during the motion. This is done by linear transformation (axis-specific motion) of the wrist axis angles. **Note:** Use **Wrist PTP** if, with **Standard**, the robot passes through a wrist axis singularity. The orientation of the TCP changes continuously during the motion, but not uniformly. **Wrist PTP** is thus not suitable if a specific orientation must be maintained exactly, e.g. in the case of laser welding.

If a wrist axis singularity occurs with **Standard** and the desired orientation cannot be maintained exactly enough with **Wrist PTP**, the following remedy is recommended:
Re-teach start and/or end point. Select orientations that prevent a wrist axis singularity from occurring and allow the path to be executed with **Standard**.

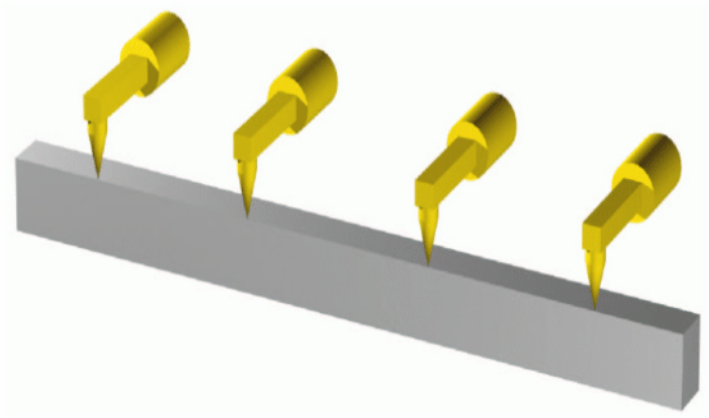


Fig. 7-17: Orientation control - Constant

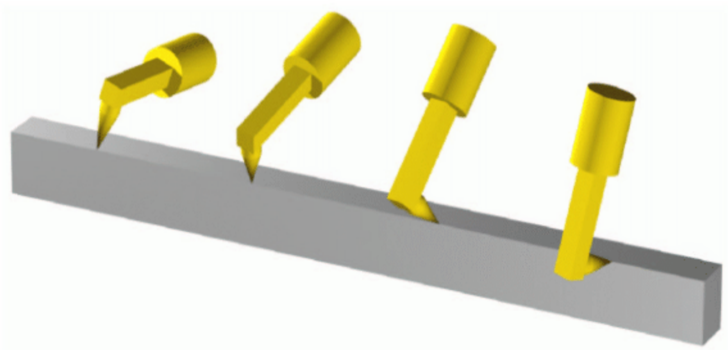


Fig. 7-18: Standard or Wrist PTP

CIRC motion

The same orientation control options are available for selection for CIRC motions as for LIN motions.

During CIRC motions, the robot controller only takes the programmed orientation of the end point into consideration. The programmed orientation of the auxiliary point is disregarded.

7.3.2 Orientation control SPLINE

The orientation control for SLIN and SCIRC motions is defined as follows:

- In the option window **Motion parameter** (spline motion), **Orientation control** box

  (>>> 8.2.10.4 "Option window “Motion parameter” (spline motion)" page 138)

The orientation control can be defined in the spline block or in the individual segment. Settings in the segment overwrite the setting in the spline block.

SLIN segment

Orientation control	Description
Constant orientation	The orientation of the TCP remains constant during the motion. The programmed orientation is disregarded for the end point and that of the start point is retained.
Standard	The orientation of the TCP changes continuously during the motion.
Ignore Orientation	This option is only available for individual spline segments, not for the spline block. It is used if no specific orientation is required at a point. (>>> "Ignore Orientation" page 124)

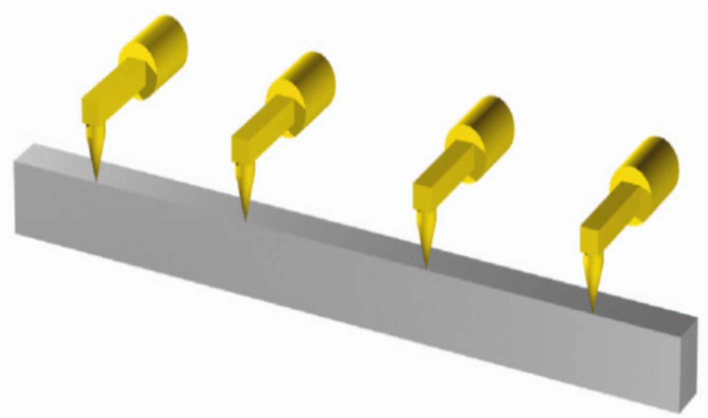


Fig. 7-19: Orientation control - Constant

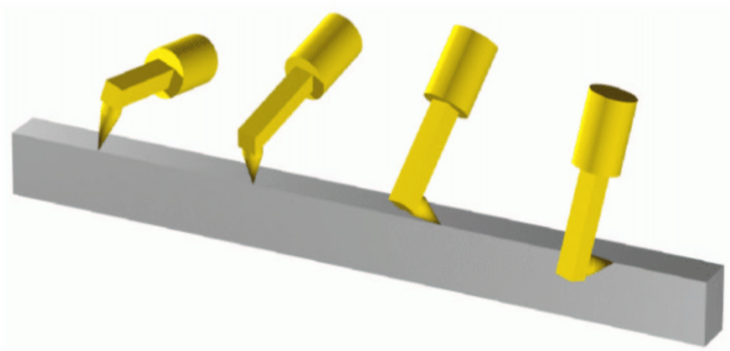


Fig. 7-20: Standard

Ignore Orientation

The option **Ignore Orientation** is used if no specific orientation is required at a point. If this option is selected, the taught or programmed orientation of the point is ignored. Instead, the robot controller calculates the optimal orientation for this point on the basis of the orientations of the surrounding points.

Characteristics of **Ignore Orientation**:

- In the program run modes MSTEP and ISTEP, the robot stops with the orientations calculated by the robot controller.
- In the case of a block selection to a point with **Ignore Orientation**, the robot adopts the orientation calculated by the robot controller.

Ignore Orientation is not allowed for the following segments:

- The first segment in a spline block
- The last segment in a spline block
- SCIRC segments with **Circle orientation control** = **path-related**
- Segments followed by a SCIRC segment with **Circle orientation control** = **path-related**
- Segments followed by a segment with **Orientation control** = **Constant orientation**
- In the case of successive segments with identical Cartesian end points, **Ignore Orientation** is not allowed for the first and last segments.

SCIRC segment

During SCIRC motions, the robot controller takes the programmed orientation of the auxiliary point into consideration.

The same orientation control options are available for selection for SCIRC motions as for SLIN motions. It is also possible to define for SCIRC motions whether the orientation control is to be space-related or path-related. This can be defined as follows:

- In the option window **Motion parameter** (spline motion), **Circle orientation control** box
  (>>> 8.2.10.4 "Option window “Motion parameter” (spline motion)" page 138)

Circle orientation control	Description
Base-related	Base-related orientation control during the circular motion
Path-related	Path-related orientation control during the circular motion

Possible combinations:

- Constant orientation, path-related

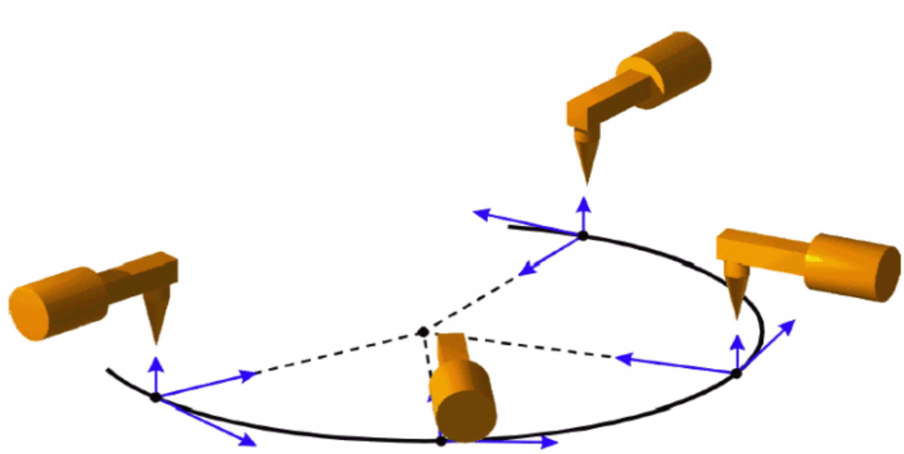


Fig. 7-21: Constant orientation, path-related

- Variable orientation, path-related

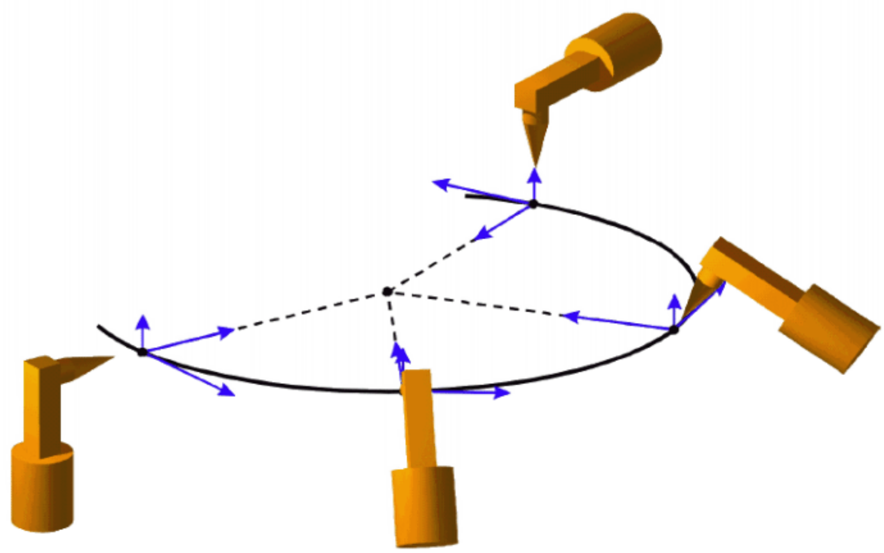


Fig. 7-22: Variable orientation, path-related

- Constant orientation, base-related

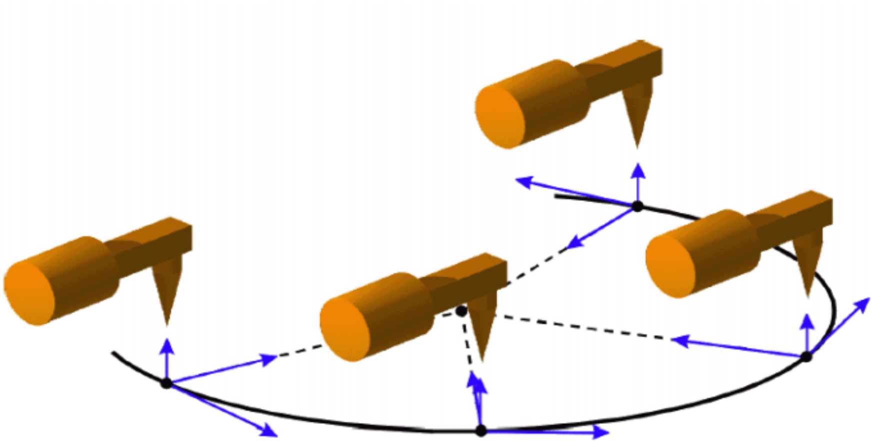


Fig. 7-23: Constant orientation, base-related

- Variable orientation, base-related

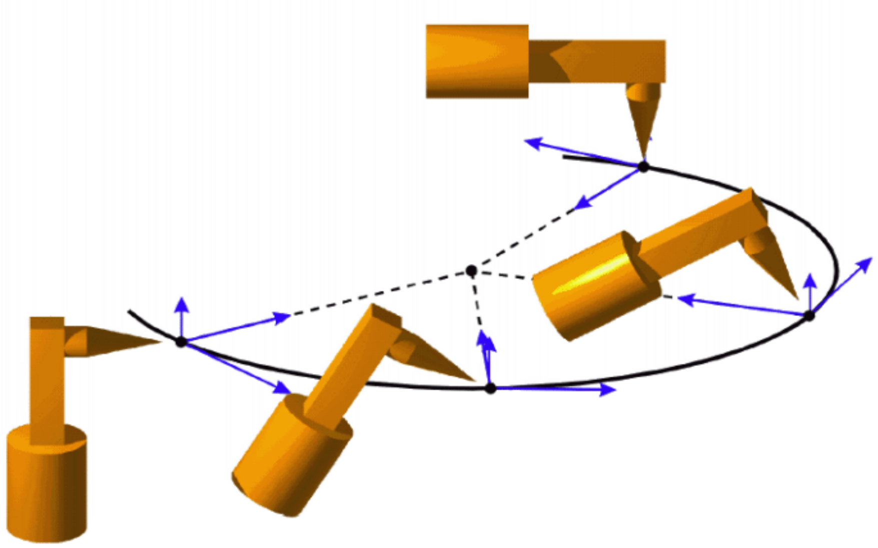


Fig. 7-24: Variable orientation, base-related

7.4 Singularities

KUKA robots with 6 degrees of freedom have 3 different singularity positions.

- Overhead singularity
- Extended position singularity
- Wrist axis singularity

A singularity position is characterized by the fact that unambiguous reverse transformation (conversion of Cartesian coordinates to axis-specific values) is not possible, even though Status and Turn are specified. In this case, or if very slight Cartesian changes cause very large changes to the axis angles, one speaks of singularity positions.

Overhead

In the overhead singularity, the wrist root point (intersection of axes A4, A5 and A6) is located vertically above axis 1.

The position of axis A1 cannot be determined unambiguously by means of reverse transformation and can thus take any value.

If the end point of a PTP motion is situated in this overhead singularity position, the robot controller may react as follows by means of the system variable $SINGUL_POS[1]:

- 0: The angle for axis A1 is defined as 0 degrees (recommended default setting).
- 1: The angle for axis A1 remains the same from the start point to the end point.

Extended position

In the extended position singularity, the wrist root point (intersection of axes A4, A5 and A6) is located in the extension of axes A2 and A3 of the robot.

The robot is at the limit of its work envelope.

Although reverse transformation does provide unambiguous axis angles, low Cartesian velocities result in high axis velocities for axes A2 and A3.

If the end point of a PTP motion is situated in this extended position singularity, the robot controller may react as follows by means of the system variable $SINGUL_POS[2]:

- 0: The angle for axis A2 is defined as 0 degrees (recommended default setting).
- 1: The angle for axis A2 remains the same from the start point to the end point.

Wrist axes

In the wrist axis singularity position, the axes A4 and A6 are parallel to one another and axis A5 is within the range ±0.01812°.

The position of the two axes cannot be determined unambiguously by reverse transformation. There is an infinite number of possible axis positions for axes A4 and A6 with identical axis angle sums.

If the end point of a PTP motion is situated in this wrist axis singularity, the robot controller may react as follows by means of the system variable $SINGUL_POS[3]:

- 0: The angle for axis A4 is defined as 0 degrees (recommended default setting).
- 1: The angle for axis A4 remains the same from the start point to the end point.

> In the case of SCARA robots, only the extended position singularity can arise. In this case, the robot starts to move extremely fast.

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

Inline forms are available in the KSS for frequently used instructions. They simplify programming.

> Instructions can also be programmed without inline forms. Information is contained in the description of the KRL syntax.

8.1 Names in inline forms

Names for data sets can be entered in inline forms. These include, for example, point names, names for motion data sets, etc.

The following restrictions apply to names:

- Maximum length 23 characters
- No special characters are permissible, with the exception of $.
- The first character must not be a number.

The restrictions do **not** apply to output names.

> Other restrictions may apply in the case of inline forms in technology packages.

8.2 Programming motions (with inline forms)

8.2.1 Programming a PTP motion

Description

Programming a PTP motion involves the following steps:

- Saving the coordinates of the end point.
- Setting various parameters (e.g. velocity).

The process of saving the point coordinates is called "teaching".

The start point of a motion is always the end point of the previous motion.

> **Caution!**
> When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Precondition

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

Procedure

1. Move the TCP to the position that is to be taught as the end point.
2. Position the cursor in the line **after** which the motion instruction is to be inserted.
3. Select the menu sequence **Commands** > **Motion** > **PTP**.
4. Set the parameters in the inline form.
   (>>> 8.2.2 "Inline form for PTP motions" page 130)
5. Save the instruction by pressing the **Cmd Ok** softkey.

8.2.2 Inline form for PTP motions

Fig. 8-1: Inline form for PTP motions

Item	Description	Range of values
1	Motion type	PTP, LIN, CIRC
2	Name of the end point The system automatically generates a name. The name can be overwritten. Position the cursor in this box to edit the point data. The corresponding option window is opened. (>>> 8.2.7 "Option window “Frames”" page 132)	(>>> 8.1 "Names in inline forms" page 129)
3	■ CONT: end point is approximated ■ [blank]: the motion stops exactly at the end point	CONT, [blank]
4	Velocity	1% … 100%
5	Name for the motion data set The system automatically generates a name. The name can be overwritten. Position the cursor in this box to edit the motion data. The corresponding option window is opened. (>>> 8.2.8 "Option window “Motion parameter” (PTP motion)" page 133)	(>>> 8.1 "Names in inline forms" page 129)

8.2.3 Programming a LIN motion

Description

Programming a LIN motion involves the following steps:

- Saving the coordinates of the end point.
- Setting various parameters (e.g. velocity).

The process of saving the point coordinates is called "teaching".

The start point of a motion is always the end point of the previous motion.

Caution!
When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Precondition

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

Procedure

1. Move the TCP to the position that is to be taught as the end point.
2. Position the cursor in the line **after** which the motion instruction is to be inserted.
3. Select the menu sequence **Commands** > **Motion** > **LIN**.

4. Set the parameters in the inline form.
   (>>> 8.2.4 "Inline form for LIN motions" page 131)
5. Save the instruction by pressing the **Cmd Ok** softkey.

8.2.4 Inline form for LIN motions

Fig. 8-2: Inline form for LIN motions

Item	Description	Range of values
1	Type of motion	PTP, LIN, CIRC
2	name of the end point The system automatically generates a name. The name can be overwritten. Position the cursor in this box to edit the point data. The corresponding option window is opened. (>>> 8.2.7 "Option window “Frames”" page 132)	(>>> 8.1 "Names in inline forms" page 129)
3	■ CONT: end point is approximated ■ [blank]: the motion stops exactly at the end point	CONT, [blank]
4	Velocity	0.001 … 2 m/s
5	Name for the motion data set The system automatically generates a name. The name can be overwritten. Position the cursor in this box to edit the motion data. The corresponding option window is opened. (>>> 8.2.9 "Option window "Motion parameter" (CP motion)" page 134)	(>>> 8.1 "Names in inline forms" page 129)

8.2.5 Programming a CIRC motion

Description

Programming a CIRC motion involves the following steps:

- Saving the coordinates of the auxiliary point.
- Saving the coordinates of the end point.
- Setting various parameters (e.g. velocity).

The process of saving the point coordinates is called "teaching".

The start point of a motion is always the end point of the previous motion.

Precondition

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

Caution!
When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Procedure

1. Move the TCP to the position that is to be taught as the auxiliary point.
2. Position the cursor in the line **after** which the motion instruction is to be inserted.
3. Select the menu sequence **Commands** > **Motion** > **CIRC**.
4. Set the parameters in the inline form.

   (>>> 8.2.6 "Inline form for CIRC motions" page 132)
5. Press the **Teach Aux** softkey.
6. Move the TCP to the position that is to be taught as the end point.
7. Save the instruction by pressing the **Cmd Ok** softkey.

8.2.6 Inline form for CIRC motions

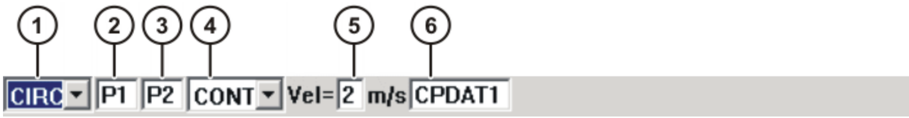


Fig. 8-3: Inline form for CIRC motions

Item	Description	Range of values
1	Type of motion	PTP, LIN, CIRC
2	Name of the auxiliary point The system automatically generates a name. The name can be overwritten.	(>>> 8.1 "Names in inline forms" page 129)
3	name of the end point The system automatically generates a name. The name can be overwritten. Position the cursor in this box to edit the point data. The corresponding option window is opened. (>>> 8.2.7 "Option window “Frames”" page 132)	(>>> 8.1 "Names in inline forms" page 129)
4	■ CONT: end point is approximated ■ [blank]: the motion stops exactly at the end point	CONT, [blank]
5	Velocity	0.001 … 2 m/s
6	Name for the motion data set The system automatically generates a name. The name can be overwritten. Position the cursor in this box to edit the motion data. The corresponding option window is opened. (>>> 8.2.9 "Option window "Motion parameter" (CP motion)" page 134)	(>>> 8.1 "Names in inline forms" page 129)

8.2.7 Option window “Frames”

This option window is called from the following inline forms:

- PTP (>>> 8.2.2 "Inline form for PTP motions" page 130)
- LIN (>>> 8.2.4 "Inline form for LIN motions" page 131)

- CIRC (>>> 8.2.6 "Inline form for CIRC motions" page 132)
- SPL, SLIN, SCIRC (>>> 8.2.10.8 "Inline form "Spline Segment"" page 141)

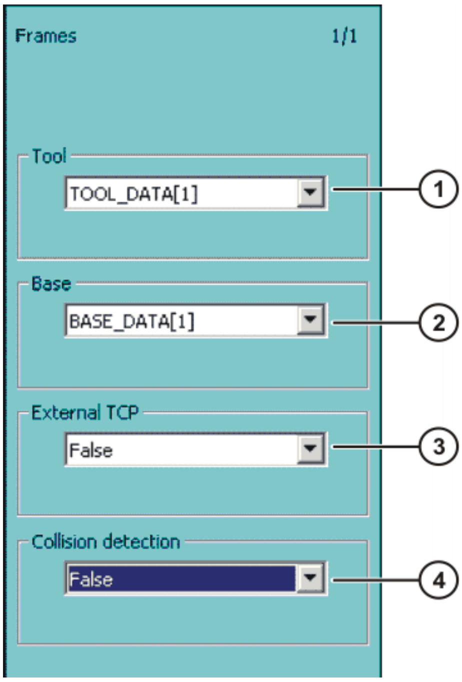


Fig. 8-4: Option window "Frames"

Item	Description	Range of values
1	Tool selection. If **True** in the box **External TCP**: workpiece selection.	[1] … [16]
2	Base selection. If **True** in the box **External TCP**: fixed tool selection.	[1] … [32]
3	Interpolation mode ▪ Tool on mounting flange: **False** ▪ Fixed tool: **True**	True, False
4	▪ **True**: For this motion, the robot controller calculates the axis torques. These are required for collision detection. ▪ **False**: For this motion, the robot controller does not calculate the axis torques. Collision detection is thus not possible for this motion.	True, False

8.2.8 Option window "Motion parameter" (PTP motion)

This option window is called from the following inline form:

- PTP (>>> 8.2.2 "Inline form for PTP motions" page 130)

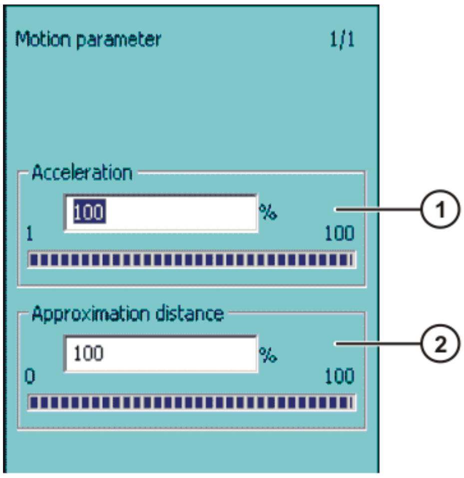


Fig. 8-5: Option window "Motion parameter" (PTP motion)

Item	Description	Range of values
1	Acceleration Refers to the maximum value specified in the machine data. The maximum value depends on the robot type and the selected operating mode.	1% … 100%
2	This box is only displayed if **CONT** has been selected in the inline form. Furthest distance before the end point at which approximate positioning can begin. Maximum distance 100%: half the distance between the start point and the end point relative to the contour of the PTP motion without approximate positioning The unit for this box can also be mm. This depends on the configuration. Distance in mm: The maximum permissible value is half the distance between the start point and the end point. If a higher value is entered, this is ignored and the maximum value is used.	0% … 100% or 0 mm … 300 mm

8.2.9 Option window "Motion parameter" (CP motion)

This option window is called from the following inline forms:

- LIN (>>> 8.2.4 "Inline form for LIN motions" page 131)
- CIRC (>>> 8.2.6 "Inline form for CIRC motions" page 132)

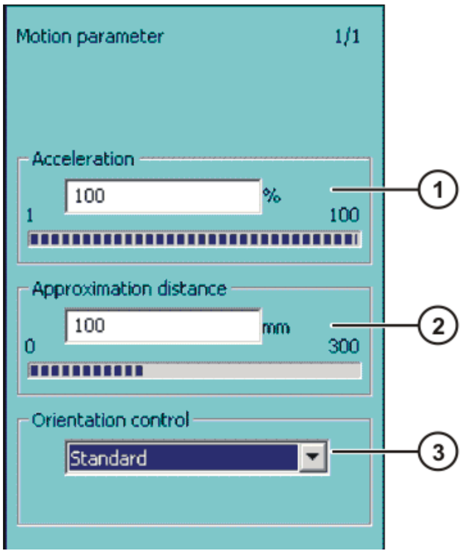


Fig. 8-6: Option window “Motion parameter” (CP motion)

Item	Description	Range of values
1	Acceleration Refers to the maximum value specified in the machine data. The maximum value depends on the robot type and the selected operating mode.	1% … 100%
2	Furthest distance before the end point at which approximate positioning can begin. The maximum permissible value is half the distance between the start point and the end point. If a higher value is entered, this is ignored and the maximum value is used. This box is only displayed if **CONT** has been selected in the inline form.	0 mm … 300 mm
3	Orientation control selection	■ Standard ■ Wrist PTP ■ Orientation control - Constant

8.2.10 Programming a spline motion

Overview

Step	Description
1	Program the spline block. (>>> 8.2.10.2 "Programming a spline block" page 137)

Step	Description
2	Program the spline segments. (>>> 8.2.10.5 "Programming an SPL segment" page 139) (>>> 8.2.10.6 "Programming an SLIN segment" page 140) (>>> 8.2.10.7 "Programming an SCIRC segment" page 140)
3	If required, and only in the user group “Expert”: Program PATH trigger.

As for all motion instructions, the start point of a spline motion is the end point of the previous motion.

8.2.10.1 Programming tips for spline motions

- A spline block should cover one process (e.g. an adhesive seam). More than one process in a spline block leads to a loss of structural clarity within the program and makes changes more difficult.
- Use SLIN and SCIRC segments in cases where the workpiece necessitates straight lines and arcs. (Exception: use SPL segments for very short straight lines.) Otherwise, use SPL segments, particularly if the points are close together.
- Procedure for defining the path:
  a. First teach or calculate a few characteristic points. Example: points at which the curve changes direction.
  b. Test the path. At points where the accuracy is still insufficient, add more SPL points.
- Avoid successive SLIN and/or SCIRC segments, as these trigger exact positioning. Exact positioning can be avoided as follows:
  - Program SPL segments between SLIN and SCIRC segments. The length of the SPL segments must be greater than 0.5 mm.
  - Replace an SLIN segment with several SPL segments. If SPL points are situated on a straight line, then the path will also be a straight line.
- Avoid successive points with identical Cartesian coordinates, as these trigger exact positioning.
- The parameters (tool, base, velocity, etc.) assigned to the spline block have the same effect as assignments before the spline block. The assignment to the spline block has the advantage, however, that the correct parameters are read in the case of a block selection.
- Use the option **Ignore Orientation** if no specific orientation is required at a point. The robot controller calculates the optimal orientation for this point on the basis of the orientations of the surrounding points. This way, even large changes in orientation between two points are optimally distributed over the points in between.
- Jerk limitation can be programmed. The jerk is the change in acceleration.

  Procedure:

  a. Use the default values initially.
  b. If vibrations occur at tight corners: reduce values.

     If the velocity drops or the desired velocity cannot be reached: increase values or increase acceleration.
- If the robot executes points on a work surface, a collision with the work surface is possible when the first point is addressed.

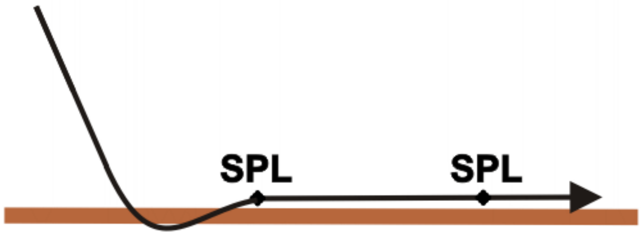


Fig. 8-7: Collision with work surface

A collision can be avoided as follows:

- Program the first segment on the work surface as SLIN.
- Additionally, if required: Insert an SLIN segment before the first point. The following precondition must be met: 2/3 ≤ a/b ≤ 3/2

  a = distance from start point of the SPL segment to intersection of the SLIN segments

  b = distance from intersection of the SLIN segments to end point of the SPL segment

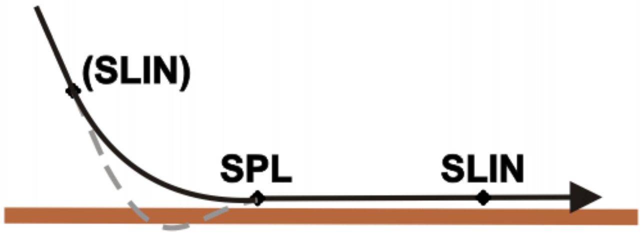


Fig. 8-8: Avoiding a collision with the work surface

8.2.10.2 Programming a spline block

Precondition

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

Procedure

1. Position the cursor in the line **after** which the spline motion is to be inserted.
2. Select the menu sequence **Commands** > **Motion** > **SPLINE Block**.
3. Set the parameters in the inline form.

   (>>> 8.2.10.3 "Inline form for spline block" page 138)
4. Press the **Cmd OK** softkey.
5. Press the **Fold open/cls** softkey. Lines can now be inserted into the spline block.

Description

A spline block may contain the following:

- Spline segments

  (The only limit here is the memory capacity. As a rule, at least 1,000 segments are possible.)
- PATH trigger
- Comments
- Blank lines
- Inline commands from technology packages that support the spline function

A spline block must not include any other instructions, e.g. variable assignments or logic statements.

A spline block does not trigger an advance run stop.

8.2.10.3 Inline form for spline block

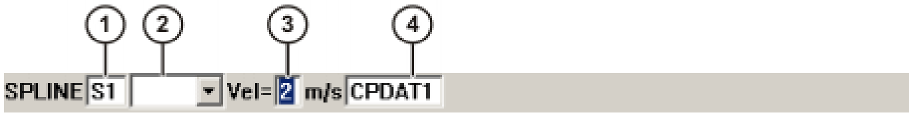


Fig. 8-9: Inline form for spline block

Item	Description	Range of values
1	Name of the spline motion. The system automatically generates a name. The name can be overwritten. Position the cursor in this box to edit the motion data. The corresponding option window is opened. (>>> 8.2.7 "Option window “Frames”" page 132)	(>>> 8.1 "Names in inline forms" page 129)
2	■ **CONT**: the end point (= last point in the spline block) is approximated. **Note:** This function is not currently supported. If **CONT** is selected, the inline form cannot be closed and an error message is generated. ■ [blank]: the motion stops exactly at the end point	CONT, [blank]
3	The velocity is valid by default for the entire spline motion. It can also be defined separately for individual segments.	0.001 … 2 m/s
4	Name for the motion data set. The system automatically generates a name. The name can be overwritten. Position the cursor in this box to edit the motion data. The corresponding option window is opened. (>>> 8.2.10.4 "Option window “Motion parameter” (spline motion)" page 138) The motion data are valid by default for the entire spline motion. They can also be defined separately for individual segments.	(>>> 8.1 "Names in inline forms" page 129)

8.2.10.4 Option window “Motion parameter” (spline motion)

This option window is called from the following inline forms:

- SPLINE (>>> 8.2.10.3 "Inline form for spline block" page 138)
- SPL/SLIN/SCIRC (>>> 8.2.10.8 "Inline form “Spline Segment”" page 141)

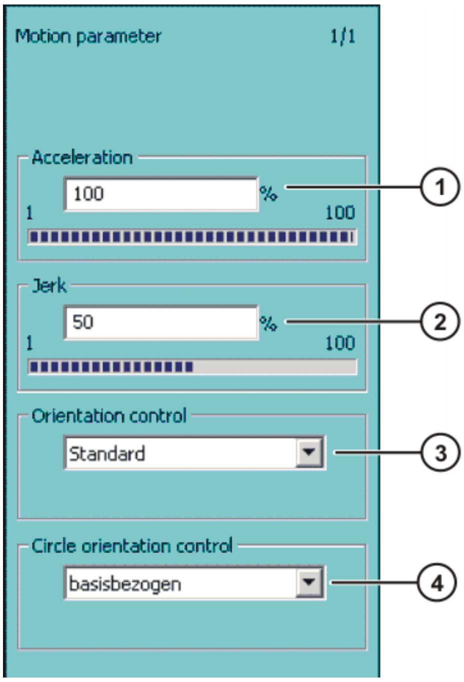


Fig. 8-10: Option window "Motion parameter" (spline motion)

Item	Description	Range of values
1	Acceleration. The value refers to the maximum value specified in the machine data.	1% … 100%
2	Jerk limitation. The jerk is the change in acceleration. The value refers to the maximum value specified in the machine data.	1% … 100% Default value: 50%
3	Orientation control selection	(>>> 7.3 "Orientation control" page 122)
4	Only for spline block and SCIRC segments: Circle orientation control selection	■ Base-related ■ Path-related

8.2.10.5 Programming an SPL segment

Caution!
When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Description

The robot guides the TCP to the end point. The robot controller selects a suitable path.

Precondition

- Program is selected.
- Operating mode T1 or T2.
- The spline block fold is open.

Procedure

1. Move the TCP to the end point.
2. Position the cursor in the line **after** which the segment is to be inserted in the spline block.
3. Select the menu sequence **Commands** > **Motion** > **SPL**.

4. Set the parameters in the inline form.
   (>>> 8.2.10.8 "Inline form “Spline Segment”" page 141)
5. Press the **Cmd OK** softkey.

8.2.10.6 Programming an SLIN segment

Caution!
When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Description

The robot guides the TCP along the shortest path to the end point. The shortest path is always a straight line.

Precondition

- Program is selected.
- Operating mode T1 or T2.
- The spline block fold is open.

Procedure

1. Move the TCP to the end point.
2. Position the cursor in the line **after** which the segment is to be inserted in the spline block.
3. Select the menu sequence **Commands** > **Motion** > **SLIN**.
4. Set the parameters in the inline form.
   (>>> 8.2.10.8 "Inline form “Spline Segment”" page 141)
5. Press the **Cmd OK** softkey.

8.2.10.7 Programming an SCIRC segment

Caution!
When programming motions, it must be ensured that the energy supply system is not wound up or damaged during program execution.

Description

The robot guides the TCP along a circular path to the end point. The circular path is defined by a start point, auxiliary point and end point.

Precondition

- Program is selected.
- Operating mode T1 or T2.
- The spline block fold is open.

Procedure

1. Move the TCP to the auxiliary point.
2. Position the cursor in the line **after** which the segment is to be inserted in the spline block.
3. Select the menu sequence **Commands** > **Motion** > **SCIRC**.
4. Set the parameters in the inline form.
   (>>> 8.2.10.8 "Inline form “Spline Segment”" page 141)
5. Press the **Teach Aux** softkey.
6. Move the TCP to the end point.
7. Press the **Cmd OK** softkey.

8.2.10.8 Inline form “Spline Segment”

Fig. 8-11: Inline form “Spline Segment”

The boxes in the inline form can be displayed or hidden one by one using the **Toggle Param** softkey.

Item	Description	Range of values
1	Select the motion type for the spline segment using the **Toggle Cmd** softkey.	SPL; SLIN; SCIRC
2	Point name for end point. Only for SCIRC: point names for auxiliary point and end point. The system automatically generates a name. The name can be overwritten. Position the cursor in the box of the end point to edit the point data. The corresponding option window is opened. (>>> 8.2.10.9 "Inline form “Frames” (spline segment)" page 141)	(>>> 8.1 "Names in inline forms" page 129)
3	Velocity This only refers to the segment to which it belongs. It has no effect on subsequent segments.	0.001 … 2 m/s
4	Name for the motion data set. The system automatically generates a name. The name can be overwritten. Position the cursor in this box to edit the motion data. The corresponding option window is opened. (>>> 8.2.10.4 "Option window “Motion parameter” (spline motion)" page 138) The motion data only refer to the segment to which they belong. They have no effect on subsequent segments.	(>>> 8.1 "Names in inline forms" page 129)

8.2.10.9 Inline form “Frames” (spline segment)

This option window is called from the following inline forms:

- Spline segment (>>> 8.2.10.8 "Inline form “Spline Segment”" page 141)

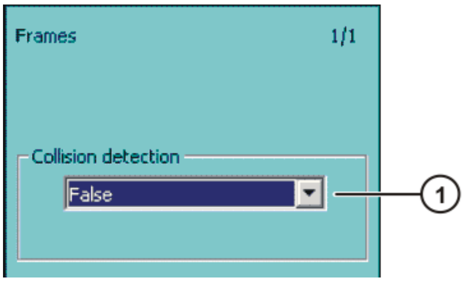


Fig. 8-12: Inline form "Frames" (spline segment)

Item	Description	Range of values
1	■ **True**: For this motion, the robot controller calculates the axis torques. These are required for collision detection. ■ **False**: For this motion, the robot controller does not calculate the axis torques. Collision detection is thus not possible for this motion.	True, False

8.2.11 Modifying motion parameters

Precondition

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

Procedure

1. Position the cursor in the line containing the instruction that is to be changed.
2. Press the **Change** softkey. The inline form for this instruction is opened.
3. Modify parameters.
4. Save changes by pressing the **Cmd Ok** softkey.

8.2.12 Reteaching a point

Description

The coordinates of a taught point can be modified. This is done by moving to the new position and overwriting the old point with the new position.

Precondition

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

Procedure

1. Move the TCP to the desired position.
2. Position the cursor in the line containing the motion instruction that is to be changed.
3. Press the **Change** softkey. The inline form for this instruction is opened.
4. For PTP and LIN motions: Press the **Touch Up** softkey to accept the current position of the TCP as the new end point.

   For CIRC motions:

   - Press the **Teach Aux** softkey to accept the current position of the TCP as the new auxiliary point.
   - Press the **Teach End** softkey to accept the current position of the TCP as the new end point.
5. Confirm the request for confirmation with **Yes**.
6. Save change by pressing the **Cmd Ok** softkey.

8.3 Programming logic instructions

8.3.1 Inputs/outputs

Digital inputs/outputs

The robot controller can manage up to 4096 digital inputs and 4096 digital outputs. The inputs/outputs are implemented in the control PC by means of optional field bus cards. The configuration is customer-specific.

Analog inputs/outputs

The robot controller can manage 32 analog inputs and 32 analog outputs. The inputs/outputs are implemented in the control PC by means of KUKA field bus cards. The configuration is customer-specific.

Permissible range of values for inputs/outputs: -1.0 to +1.0. This corresponds to a voltage range from -10 V to +10 V. If the value is exceeded, the input/output takes the maximum value and a message is displayed until the value is back in the permissible range.

The inputs/outputs are managed via the following system variables:

	Inputs	Outputs
Digital	$IN[1] … $IN[4096]	$OUT[1] … $OUT[4096]
Analog	$ANIN[1] … $ANIN[32]	$ANOUT[1] … $ANOUT[32]

8.3.2 Setting a digital output - OUT

Precondition

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

Procedure

1. Position the cursor in the line **after** which the logic instruction is to be inserted.
2. Select the menu sequence **Commands** > **Logic** > **OUT** > **OUT**.
3. Set the parameters in the inline form.
   (>>> 8.3.3 "Inline form "OUT"" page 143)
4. Save the instruction by pressing the **Cmd Ok** softkey.

8.3.3 Inline form "OUT"

The instruction sets a digital output.

Fig. 8-13: Inline form "OUT"

Item	Description	Range of values
1	Output number	1 … 4096
2	If a name exists for the output, this name is displayed. Only for the user group "Expert": A name can be entered by pressing the **Longtext** softkey.	Freely selectable
3	State to which the output is switched	TRUE, FALSE
4	■ CONT: Execution in the advance run ■ [blank]: Execution with advance run stop	CONT, [blank]

8.3.4 Setting a pulse output - PULSE

Precondition

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

Procedure

1. Position the cursor in the line **after** which the logic instruction is to be inserted.
2. Select the menu sequence **Commands** > **Logic** > **OUT** > **PULSE**.
3. Set the parameters in the inline form.

   (>>> 8.3.5 "Inline form "PULSE"" page 144)
4. Save the instruction by pressing the **Cmd Ok** softkey.

8.3.5 Inline form "PULSE"

The instruction sets a pulse of a defined length.

Fig. 8-14: Inline form "PULSE"

Item	Description	Range of values
1	Output number	1 … 4096
2	If a name exists for the output, this name is displayed. Only for the user group "Expert": A name can be entered by pressing the **Longtext** softkey.	Freely selectable
3	State to which the output is switched ■ TRUE: "High" level ■ FALSE: "Low" level	TRUE, FALSE
4	■ CONT: Execution in the advance run ■ [blank]: Execution with advance run stop	CONT, [blank]
5	Length of the pulse	0.1 … 3

8.3.6 Setting an analog output - ANOUT

Precondition

- Program is selected.

- Operating mode T1 or T2.

Procedure

1. Position the cursor in the line **after** which the instruction is to be inserted.
2. Select the menu sequence **Commands** > **Analog output** > **Static** or **Dynamic**.
3. Set the parameters in the inline form.

   (>>> 8.3.7 "Inline form "ANOUT" (static)" page 145)

   (>>> 8.3.8 "Inline form "ANOUT" (dynamic)" page 145)
4. Save the instruction by pressing the **Cmd Ok** softkey.

8.3.7 Inline form "ANOUT" (static)

This instruction sets a static analog output.

A maximum of 8 analog outputs (static and dynamic together) can be used at any one time. ANOUT triggers an advance run stop.

The voltage is set to a fixed level by means of a factor. The actual voltage level depends on the analog module used. For example, a 10 V module with a factor of 0.5 provides a voltage of 5 V.

Fig. 8-15: Inline form "ANOUT" (static)

Item	Description	Range of values
1	Analog output number	CHANNEL_1 … CHANNEL_32
2	Factor for the voltage	0 … 1 Intervals: 0.01

8.3.8 Inline form "ANOUT" (dynamic)

This instruction activates or deactivates a dynamic analog output.

A maximum of 4 dynamic analog outputs can be activated at any one time. ANOUT triggers an advance run stop.

The voltage is determined by a factor. The actual voltage level depends on the following values:

- Velocity or function generator

  For example, a velocity of 1 m/s with a factor of 0.5 results in a voltage of 5 V.
- Offset

  For example, an offset of +0.15 for a voltage of 0.5 V results in a voltage of 6.5 V.

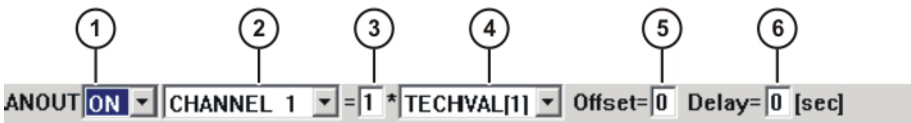


Fig. 8-16: Inline form "ANOUT" (dynamic)

Item	Description	Range of values
1	Activation or deactivation of the analog output	ON, OFF
2	Analog output number	CHANNEL_1 … CHANNEL_32
3	Factor for the voltage	0 … 10 Intervals: 0.01
4	■ VEL_ACT: The voltage is dependent on the velocity. ■ TECHVAL[x]: The voltage is controlled by a function generator.	VEL_ACT, TECHVAL[1] … TECHVAL[6]
5	Value by which the voltage is increased or decreased	-1 … +1 Intervals: 0.01
6	Time by which the output signal is delayed (+) or brought forward (-)	-0.2 … +0.5 s

8.3.9 Programming a wait time - WAIT

Precondition

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

Procedure

1. Position the cursor in the line **after** which the logic instruction is to be inserted.
2. Select the menu sequence **Commands** > **Logic** > **WAIT**.
3. Set the parameters in the inline form.
   (>>> 8.3.10 "Inline form "WAIT"" page 146)
4. Save the instruction by pressing the **Cmd Ok** softkey.

8.3.10 Inline form "WAIT"

WAIT can be used to program a wait time. The robot motion is stopped for a programmed time. WAIT always triggers an advance run stop.

Fig. 8-17: Inline form "WAIT"

Item	Description	Range of values
1	Wait time	≥ 0 s

8.3.11 Programming a signal-dependent wait function - WAITFOR

Precondition

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

Procedure

1. Position the cursor in the line **after** which the logic instruction is to be inserted.
2. Select the menu sequence **Commands** > **Logic** > **WAITFOR**.
3. Set the parameters in the inline form.

(>>> 8.3.12 "Inline form "WAITFOR"" page 147)

4. Save the instruction by pressing the **Cmd Ok** softkey.

8.3.12 Inline form "WAITFOR"

The instruction sets a signal-dependent wait function.

If required, several signals (maximum 12) can be linked. If a logic operation is added, boxes are displayed in the inline form for the additional signals and links.

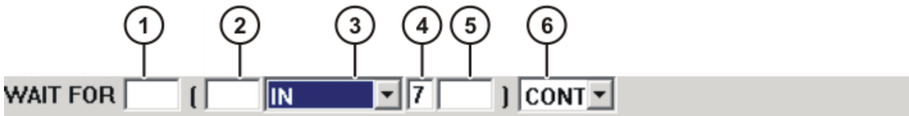


Fig. 8-18: Inline form "WAITFOR"

Item	Description	Range of values
1	■ Add external logic operation. The operator is situated between the bracketed expressions. ■ Add NOT. Enter the desired operator or NOT by means of the softkey.	AND, OR, EXOR, [blank] NOT, [blank]
2	■ Add internal logic operation. The operator is situated inside a bracketed expression. ■ Add NOT. Enter the desired operator or NOT by means of the softkey.	AND, OR, EXOR, [blank] NOT, [blank]
3	Signal for which the system is waiting	IN, OUT, CYCFLAG, TIMER, FLAG
4	Signal number	1 … 4096
5	If a name exists for the signal, this name is displayed. Only for the user group "Expert": A name can be entered by pressing the **Longtext** softkey.	Freely selectable
6	■ CONT: Execution in the advance run ■ [blank]: Execution with advance run stop	CONT, [blank]

8.3.13 Switching on the path - SYN OUT

Precondition

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

Procedure

1. Position the cursor in the line **after** which the logic instruction is to be inserted.
2. Select the menu sequence **Commands** > **Logic** > **OUT** > **SYN OUT**.
3. Set the parameters in the inline form.

(>>> 8.3.14 "Inline form SYN OUT, option START/END" page 148)

(>>> 8.3.15 "Inline form SYN OUT, option PATH" page 150)

4. Save the instruction by pressing the **Cmd Ok** softkey.

8.3.14 Inline form SYN OUT, option START/END

A switching action can be triggered relative to the start or end point of a motion block. The switching action can be delayed or brought forward. The motion block can be a LIN, CIRC or PTP motion.

Possible applications include:

- Closing or opening the weld gun during spot welding
- Switching the welding current on/off during arc welding
- Starting or stopping the flow of adhesive in bonding or sealing applications.

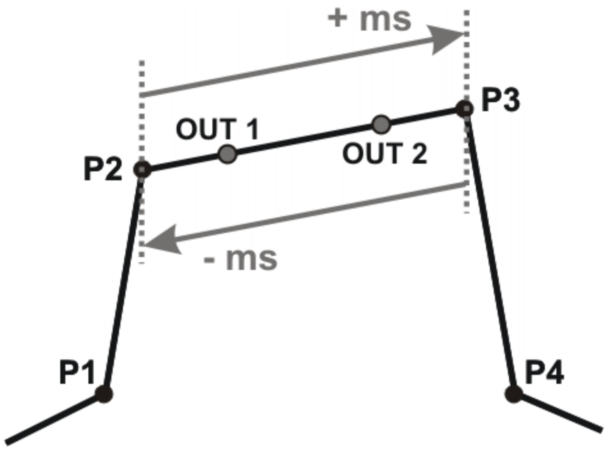
Fig. 8-19: Inline form SYN OUT, option START/END

Item	Description	Range of values
1	Output number	1 … 4096
2	If a name exists for the output, this name is displayed. Only for the user group "Expert": A name can be entered by pressing the **Longtext** softkey.	Freely selectable
3	State to which the output is switched	TRUE, FALSE
4	Point at which switching is carried out ■ **START**: Switching is carried out at the start point of the motion block. ■ **END**: Switching is carried out at the end point of the motion block.	START, END Option PATH: (>>> 8.3.15 "Inline form SYN OUT, option PATH" page 150)
5	Switching action delay **Note:** The time specification is absolute. The switching point thus varies according to the velocity of the robot.	-1,000 … +1,000 ms

Example 1

Start point and end point are exact positioning points.

```
LIN P1 VEL=0.3m/s CPDAT1
LIN P2 VEL=0.3m/s CPDAT2
SYN OUT 1 '' State= TRUE at START Delay=20ms
SYN OUT 2 '' State= TRUE at END Delay=-20ms
LIN P3 VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
```



OUT 1 and OUT 2 specify approximate positions at which switching is to occur. The dotted lines indicate the switching limits.

Switching limits:

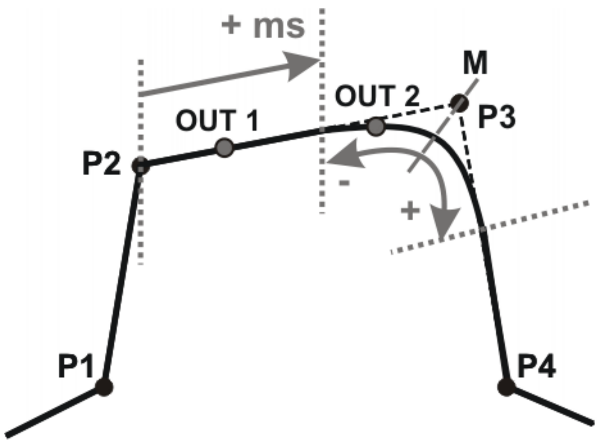
- START: The switching point can be delayed, at most, as far as exact positioning point P3 (+ ms).
- END: The switching point can be brought forward, at most, as far as exact positioning point P2 (- ms).

If greater values are specified for the delay, the controller automatically switches at the switching limit.

Example 2

Start point is exact positioning point, end point is approximated.

```
LIN P1 VEL=0.3m/s CPDAT1
LIN P2 VEL=0.3m/s CPDAT2
SYN OUT 1 '' State= TRUE at START Delay=20ms
SYN OUT 2 '' State= TRUE at END Delay=-20ms
LIN P3 CONT VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
```



OUT 1 and OUT 2 specify approximate positions at which switching is to occur. The dotted lines indicate the switching limits. M = middle of the approximate positioning range.

Switching limits:

- START: The switching point can be delayed, at most, as far as the start of the approximate positioning range of P3 (+ ms).

- END: The switching point can be brought forward, at most, as far as the start of the approximate positioning range of P3 (-).

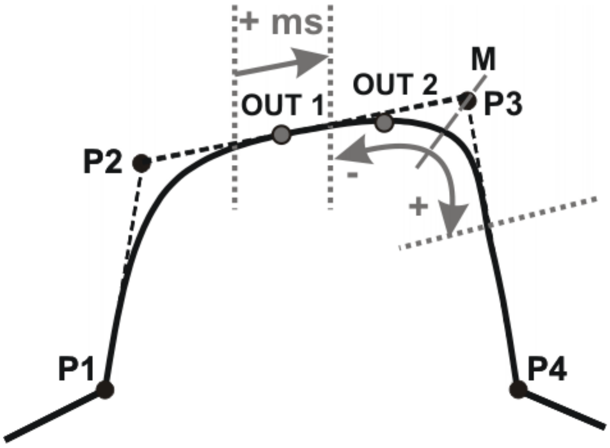
  The switching point can be delayed, at most, as far as the end of the approximate positioning range of P3 (+).

If greater values are specified for the delay, the controller automatically switches at the switching limit.

Example 3

Start point and end point are approximated.

```
LIN P1 VEL=0.3m/s CPDAT1
LIN P2 CONT VEL=0.3m/s CPDAT2
SYN OUT 1 '' State= TRUE at START Delay=20ms
SYN OUT 2 '' State= TRUE at END Delay=-20ms
LIN P3 CONT VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
```



OUT 1 and OUT 2 specify approximate positions at which switching is to occur. The dotted lines indicate the switching limits. M = middle of the approximate positioning range.

Switching limits:

- START: The switching point can be situated, at the earliest, at the end of the approximate positioning range of P2.

  The switching point can be delayed, at most, as far as the start of the approximate positioning range of P3 (+ ms).
- END: The switching point can be brought forward, at most, as far as the start of the approximate positioning range of P3 (-).

  The switching point can be delayed, at most, as far as the end of the approximate positioning range of P3 (+).

If greater values are specified for the delay, the controller automatically switches at the switching limit.

8.3.15 Inline form SYN OUT, option PATH

A switching action can be triggered relative to the end point of a motion block. The switching action can be shifted in space and delayed or brought forward. The motion block can be a LIN or CIRC motion. It must not be a PTP motion.

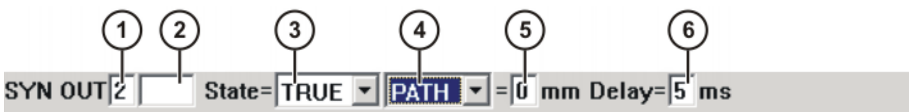


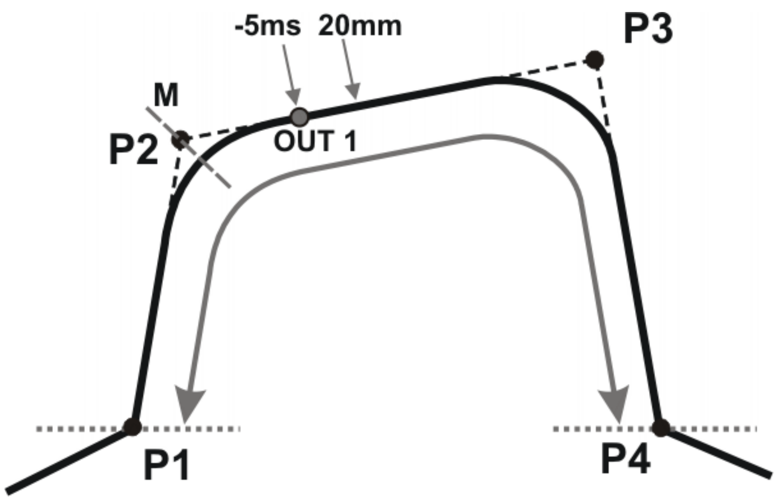
Fig. 8-20: Inline form SYN OUT, option PATH

Item	Description	Range of values
1	Output number	1 … 4096
2	If a name exists for the output, this name is displayed. Only for the user group "Expert": A name can be entered by pressing the **Longtext** softkey.	Freely selectable
3	State to which the output is switched	TRUE, FALSE
4	- - -	PATH Option START or END: (>>> 8.3.14 "Inline form SYN OUT, option START/ END" page 148)
5	Distance from the switching point to the end point This box is only displayed if **PATH** has been selected under item 4.	-2,000 … +2,000 mm
6	Switching action delay **Note:** The time specification is absolute. The switching point thus varies according to the velocity of the robot.	-1,000 … +1,000 ms

Example 1

Start point is exact positioning point, end point is approximated.

```
LIN P1 VEL=0.3m/s CPDAT1
SYN OUT 1 '' State= TRUE at START PATH=20mm Delay=-5ms
LIN P2 CONT VEL=0.3m/s CPDAT2
LIN P3 CONT VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
```



OUT 1 specifies the approximate position at which switching is to occur. The dotted lines indicate the switching limits. M = middle of the approximate positioning range.

Switching limits:

- The switching point can be brought forward, at most, as far as exact positioning point P1.

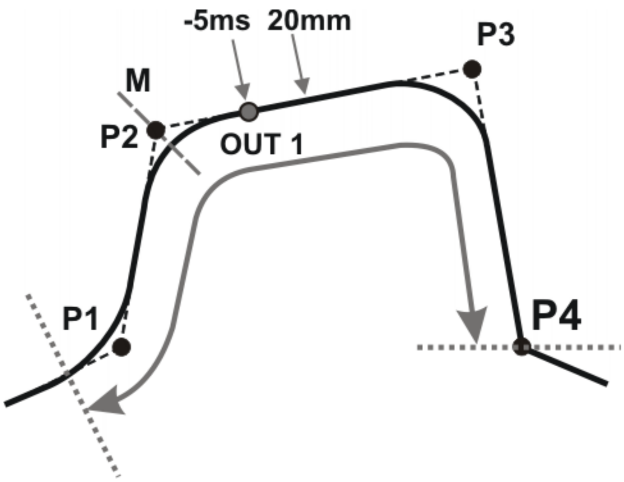
- The switching point can be delayed, at most, as far as the next exact positioning point P4. If P3 was an exact positioning point, the switching point could be delayed, at most, as far as P3.

If greater values are specified for the shift in space or time, the controller automatically switches at the switching limit.

Example 2

Start point and end point are approximated.

```
LIN P1 CONT VEL=0.3m/s CPDAT1
SYN OUT 1 '' State= TRUE at START PATH=20mm Delay=-5ms
LIN P2 CONT VEL=0.3m/s CPDAT2
LIN P3 CONT VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
```



OUT 1 specifies the approximate position at which switching is to occur. The dotted lines indicate the switching limits. M = middle of the approximate positioning range.

Switching limits:

- The switching point can be brought forward, at most, as far as the start of the approximate positioning range of P1.
- The switching point can be delayed, at most, as far as the next exact positioning point P4. If P3 was an exact positioning point, the switching point could be delayed, at most, as far as P3.

If greater values are specified for the shift in space or time, the controller automatically switches at the switching limit.

8.3.16 Setting a pulse on the path - SYN PULSE

Precondition

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

Procedure

1. Position the cursor in the line **after** which the logic instruction is to be inserted.
2. Select the menu sequence **Commands** > **Logic** > **OUT** > **SYN PULSE**.
3. Set the parameters in the inline form.

   (>>> 8.3.17 "Inline form "SYN PULSE"" page 153)
4. Save the instruction by pressing the **Cmd Ok** softkey.

8.3.17 Inline form "SYN PULSE"

A pulse can be triggered relative to the start or end point of a motion block. The pulse can be delayed or brought forward and shifted in space.

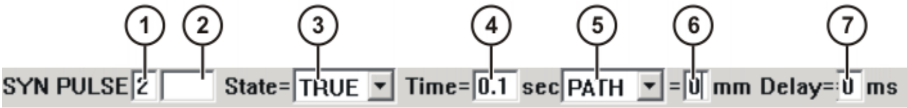


Fig. 8-21: Inline form "SYN PULSE"

Item	Description	Range of values
1	Output number	1 … 4096
2	If a name exists for the output, this name is displayed. Only for the user group "Expert": A name can be entered by pressing the **Longtext** softkey.	Freely selectable
3	State to which the output is switched	TRUE, FALSE
4	Duration of the pulse	0.1 … 3 s
5	■ **START**: The pulse is triggered at the start point of the motion block. ■ **END**: The pulse is triggered at the end point of the motion block. See SYN OUT for examples and switching limits. (>>> 8.3.14 "Inline form SYN OUT, option START/END" page 148) ■ **PATH**: The pulse is triggered at the end point of the motion block. See SYN OUT for examples and switching limits. (>>> 8.3.15 "Inline form SYN OUT, option PATH" page 150)	START, END, PATH
5	Distance from the switching point to the end point This box is only displayed if **PATH** has been selected under item 4.	-2,000 … +2,000 mm
6	Pulse delay. **Note:** The time specification is absolute. The switching point thus varies according to the velocity of the robot.	-1,000 … +1,000 ms

8.3.18 Modifying a logic instruction

Precondition

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

Procedure

1. Position the cursor in the line containing the instruction that is to be changed.
2. Press the **Change** softkey. The inline form for this instruction is opened.
3. Modify parameters.
4. Save changes by pressing the **Cmd Ok** softkey.

9 Messages

9.1 System messages

Details about the system messages can be found in the online help.

(>>> 4.2.5 "Calling online help" page 45)

9.2 Automatic External error messages

No.	Message text	Cause
P00:1	PGNO_TYPE incorrect value permissible values (1,2,3)	The data type for the program number was entered incorrectly.
P00:2	PGNO_LENGTH incorrect value Range of values 1 ≤ PGNO_LENGTH ≤ 16	The selected program number length in bits was too high.
P00:3	PGNO_LENGTH incorrect value permissible values (4,8,12,16)	If BCD format was selected for reading the program number, a corresponding number of bits must also be set.
P00:4	PGNO_FBIT incorrect value not in the $IN range	The value “0” or a non-existent input was specified for the first bit of the program number.
P00:7	PGNO_REQ incorrect value not in the $OUT range	The value “0” or a non-existent output was specified for the output via which the program number is to be requested.
P00:10	Transmission error incorrect parity	Discrepancy detected when checking parity. A transmission error must have occurred.
P00:11	Transmission error incorrect program number	A program number was sent by the host computer for which no branch for execution has (yet) been created in the CELL.SRC control structure.
P00:12	Transmission error incorrect BCD encoding	The attempt to read the program number in BCD format led to an invalid result.
P00:13	Incorrect operating mode	The I/O interface output has not been activated, i.e. the system variable $I_O_ACTCONF currently has the value FALSE. This can have the following causes: ■ The mode selector switch is not in the “Automatic External” position. ■ The signal $I_O_ACT currently has the value FALSE.
P00:14	Move to Home position in operating mode T1	The robot has not reached the HOME position.
P00:15	Incorrect program number	More than one input set with “1 of n”.

10 KUKA Service

10.1 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 are to 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

10.2 KUKA Customer Support

Availability

KUKA Customer Support is available in many countries. Please do not hesitate to contact us if you have any questions.

Argentina

Ruben Costantini S.A. (Agency)
Luis Angel Huergo 13 20
Parque Industrial
2400 San Francisco (CBA)
Argentina
Tel. +54 3564 421033
Fax +54 3564 428877
ventas@costantini-sa.com

Australia

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

Austria	KUKA Roboter 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
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

France

KUKA Automatisme + Robotique SAS
Techvallée
6 Avenue du Parc
91140 Villebon s/Yvette
France
Tel. +33 1 6931-6600
Fax +33 1 6931-6601
commercial@kuka.fr
www.kuka.fr

Germany

KUKA Roboter GmbH
Blücherstr. 144
86165 Augsburg
Germany
Tel. +49 821 797-4000
Fax +49 821 797-1616
info@kuka-roboter.de
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

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

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

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

KUKA-VAZ Engineering
Jushnoje Chaussee, 36 VAZ, PTO
445633 Togliatti
Russia
Tel. +7 8482 391249 or 370564
Fax +7 8482 736730
Y.Klychkov@VAZ.RU

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

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

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

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

UK

KUKA Automation + Robotics
Hereward Rise
Halesowen
B62 8AN
UK
Tel. +44 121 585-0800
Fax +44 121 585-0900
sales@kuka.co.uk

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

Index