SIEMENS

SINUMERIK

SINUMERIK ONE / 840D sl
SINUMERIK Run MyCC /ROBX
(ROBot transformation eX)

Function Manual

Valid for

SINUMERIK Run MyCC /ROBX Version 06.05.00

12/2023
A5E53376759B AA

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER
indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠ WARNING
indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠ CAUTION
indicates that minor personal injury can result if proper precautions are not taken.

NOTICE
indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens Aktiengesellschaft. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens Aktiengesellschaft
Smart Infrastructure
Electrical Products
Postfach 10 09 53
93009 Regensburg
GERMANY

A5E53376759B AA
Ⓟ 02/2024 Subject to change

Table of contents

1 Change log

FB version	ROBX version	Description
V5.5	05.04.03	Error correction for PR 790862: The RPY angle in the BCS is calculated according to the "shortest path" strategy. As a consequence, 360 degree rotations of the wrist axes are avoided for REPOS.
V5.6	05.05.00	Function expansion: Monitoring for self collisions for robots with parallelogram structure ($MC_ROBX_SPECIAL_KIN = 2) Error correction: Check whether sufficient memory is available for CC block elements. (new alarms 75322 and 75334)
V6.0	06.00.00	Expansion of the description of $MC_ROBX_TOOL_DIR. Expansion of the CC kinematics to include one inclined wrist axis
V6.1	06.01.00	Description of a 6-axis articulated arm robot with angled inclined wrist, Check whether all axes are referenced when activating the transformation, Option bit for RODI no longer interrogated. The option bit for ROBX must always be set.
V6.2	06.02.00	Debugs: • Retract after initiating parallelogram angle monitoring • After initiating the parallelogram angle monitoring, it is not permissible that setpoint step monitoring is active.
V6.3	06.03.00	Expansions: • Integration of an internal interface to compile cycle CCROCO for transferring internal parameters • Expansion of the documentation to include Chapter "Singular positions and their handling"
V6.4	06.04.00	Expansions: • Integration of a new coordinate system between the world coordinate system (WO) and the coordinate system at the robot base (TROWO).
V6.5	06.05.00	Expansions: • Integration of 3 additional redundant axes in the flange coordinate system, which rotate and tilt the tool. • Improved behavior when triggering the software limit switch monitoring near a singularity. • Error correction: – Braking response for Cartesian manual traversing.

Important product information 2

2.1 Compatibility

Information about the compatibility of interface versions of compile cycles and CNC software versions is provided on the Internet at address (https://support.industry.siemens.com/cs/ww/en/view/109812591).

2.2 Technical documentation

Technical documentation for your product is included in the scope of delivery and is available through the Internet. The following table is intended to provide you with orientation and navigation support:

Topic	Document(s)	Source
Description of the specific commissioning and functionality of a loadable Siemens compile cycle	Specific compile cycle function description	Included in the scope of delivery
General information regarding installing, updating and uninstalling loadable Siemens compile cycles	Technologies Function Manual	Internet → Address (https://support.industry.siemens.com/cs/ww/en/view/109812591)
Technical documentation on SINUMERIK ONE	All manuals	Internet → Address (https://support.industry.siemens.com/cs/ww/en/view/109768483)
Technical documentation for SINUMERIK 840D sl	All manuals	Internet → Address (https://support.industry.siemens.com/cs/ww/en/view/109766213)

You can access the individual document via the specified internet addresses and a document overview. You can display the document or download it in the PDF and HTML5 format.

Note

Hyperlinks to third-party websites

Documents can include hyperlinks to third-party websites. Siemens is not responsible for and shall not be liable for these websites and their content. Siemens has no control over the information that appears on these websites and is not responsible for the content and information provided there. The user bears the full risk when visiting these websites.

2.3 Compliance with the General Data Protection Regulation

Siemens observes standard data protection principles, in particular the data minimization rules (privacy by design).

For this product, this means:

The product does not process or store any personal data, only technical function data (e.g. time stamps). If the user links this data with other data (e.g. shift plans) or if he/she stores person-related data on the same data medium (e.g. hard disk), thus personalizing this data, he/she must ensure compliance with the applicable data protection stipulations.

2.4 Using OpenSSL

This product can contain the following software:

- Software developed by the OpenSSL project for use in the OpenSSL toolkit
- Cryptographic software created by Eric Young.
- Software developed by Eric Young

You can find more information on the internet:

- OpenSSL (https://www.openssl.org)
- Cryptsoft (https://www.cryptsoft.com)

3 Fundamental safety instructions

3.1 General safety instructions

WARNING

Danger to life if the safety instructions and residual risks are not observed

If the safety instructions and residual risks in the associated hardware documentation are not observed, accidents involving severe injuries or death can occur.

- Observe the safety instructions given in the hardware documentation.
- Consider the residual risks for the risk evaluation.

WARNING

Malfunctions of the machine as a result of incorrect or changed parameter settings

As a result of incorrect or changed parameterization, machines can malfunction, which in turn can lead to injuries or death.

- Protect the parameterization against unauthorized access.
- Handle possible malfunctions by taking suitable measures, e.g. emergency stop or emergency off.

3.2 Warranty and liability for application examples

Application examples are not binding and do not claim to be complete regarding configuration, equipment, or any eventuality which may arise. Application examples do not represent customer-specific solutions, but merely serve to provide assistance with typical tasks.

As the user you yourself are responsible for ensuring that the products described are operated correctly. Application examples do not relieve you of your responsibility for safe handling when using, installing, operating and maintaining the equipment.

3.3 Cybersecurity information

Siemens provides products and solutions with industrial cybersecurity functions that support the secure operation of plants, systems, machines and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial cybersecurity concept. Siemens' products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For additional information on industrial cybersecurity measures that may be implemented, please visit
https://www.siemens.com/cybersecurity-industry.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customer's exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Cybersecurity RSS Feed under
https://new.siemens.com/cert.

Further information is provided on the Internet:

Industrial Security Configuration Manual (https://support.industry.siemens.com/cs/ww/en/view/108862708)

WARNING

Unsafe operating states resulting from software manipulation

Software manipulations, e.g. viruses, Trojans, or worms, can cause unsafe operating states in your system that may lead to death, serious injury, and property damage.

- Keep the software up to date.
- Incorporate the automation and drive components into a state-of-the-art, integrated industrial cybersecurity concept for the installation or machine.
- Make sure that you include all installed products in the integrated industrial cybersecurity concept.
- Protect files stored on exchangeable storage media from malicious software by with suitable protection measures, e.g. virus scanners.
- Carefully check all cybersecurity-related settings once commissioning has been completed.

Function description

4

4.1 Function

Main axes

Transformation ROBX facilitates a large number of robot kinematics through machine data parameterization alone. Currently, kinematics with at least 4 up to a maximum of 6 axes (main axes), which are incorporated in the transformation, can be emulated. As a consequence, up to 6 degrees of freedom in space, a maximum of 3 degrees of freedom for the offset (translation) and 3 degrees of freedom for rotation (orientation). For a 6-axis robot, this means that a workpiece can be oriented and rotated as required at each accessible point of the machining space.

Basic axes

The first 3 axes included in the transformation are generally referred to as the "basic axes". They must always be mutually parallel or perpendicular to one another.

Linear additional axes

With up to 3 additional linear axes, which are included in the transformation, the robot can be moved relative to the workpiece.

Rotary additional axes

With up to 3 additional rotary axes, which are also included in the transformation, the workpiece can be rotated in space.

Workpiece programming

The workpiece is always programmed in the right-angled workpiece coordinate system (WCS). Programmed or set frames rotate and/or shift the WCS to the basic coordinate system (BCS). The kinematic transformation converts this information into motion instructions for the real machine axes. To do this, the kinematic transformation requires the appropriate information about the design (kinematics) of the machine, which are stored in machine data.

4.2 Coordinate systems: Translation and rotation using frames

In order to make a clear distinction from the term frame as it is used in the NC language, the following description explains the meaning of the term frame with regard to the ROBX transformation.

Frame

One coordinate system can be transitioned into another coordinate system using a frame. A distinction is made between translation and rotation. While translation causes only an offset, the rotation rotates the coordinate system with regard to the reference system. Coordinate axes X, Y and Z are used as identifier for the direction of a translation. The coordinate axes are defined so that an orthogonal right-handed coordinate system is obtained.

When transforming a coordinate system into another coordinate system, the translation is always executed before the rotation.

Translation

Translations are always specified relating to the direction of the coordinate axes of the initial coordinate system. The coordinate axes are assigned to the indices of the relevant machine data as follows:

- In the direction of the **X** axis: ..._POS[**0**]
- In the direction of the **Y** axis: ..._POS[**1**]
- In the direction of the **Z** axis: ..._POS[**2**]

Rotation

Rotations are specified using the RPY notation (RPY: roll, pitch, yaw). Rotation angles are specified using RPY angles A, B and C. Angles A and C are only defined in the interval [-180°; +180°] and angle B, only in interval [-90°; +90°].

The positive direction of rotation is defined by the right hand rule, i.e. if the thumb on the right hand is pointing in the direction of the axis of rotation, then the fingers point in the positive direction of rotation.

Rotation angles A, B, C

The effect of RPY angles A, B and C is defined as follows:

- Angle **A**: 1st rotation around the Z axis of the initial system
- Angle **B**: 2nd rotation about the rotated Y axis
- Angle **C**: 3rd rotation around the X axis that has been rotated twice

RPY angles A, B and C are assigned to the indices of the relevant machine data as follows:

- Angle **A**: ..._RPY[**0**]
- Angle **B**: ..._RPY[**1**]
- Angle **C**: ..._RPY[**2**]

Example

Rotating the initial coordinate system X1, Y1, Z1 through RPY angles A = -90, B = 90 and C = -90 to generate the target coordinate system X4, Y4, Z4:

1. Rotating the initial coordinate system X1, Y1, Z1 through angle A around the Z1 axis ⇒ coordinate system X2, Y2, Z2
2. Rotating the coordinate system X2, Y2, Z2 through angle B around the Y2 axis ⇒ coordinate system X3, Y3, Z3
3. Rotating the coordinate system X3, Y3, Z3 through angle C around the X3 axis ⇒ coordinate system X4, Y4, Z4

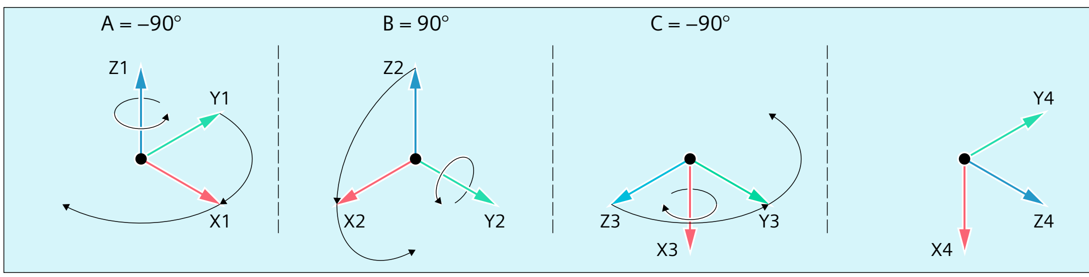


Figure 4-1 Example of rotation through the RPY angles

4.3 Joint definition

A joint is an axis from which the kinematics is established. The axis can either be a translatory or a rotary axis.

The following diagram shows the symbols for the translatory and rotary axes corresponding to their specific motion and/or rotational direction is well as for the flange to mount the tool and the tool itself:

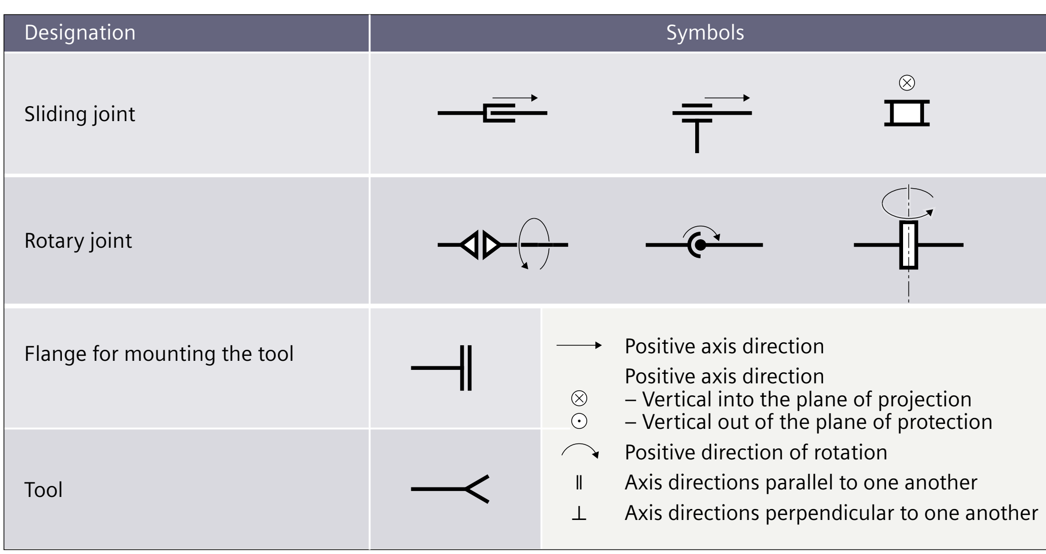


Figure 4-2 Joint identifying letters

4.4 Preconditions

Referencing status of the axes involved in the ROBX transformation

All axes involved in the ROBX transformation must already have been referenced **before** selecting the transformation. If the transformation is selected without all of the axes involved in the transformation having been referenced, the following alarm is displayed with the first traversing motion after selection:

75335 "ROBX: Transformation prevents movements with unreferenced axes"

Commissioning 5

5.1 General machine data

5.1.1 Number of block elements for compile cycles

The machine data is used to define how many block elements are reserved for compile cycles.

For the ROBX transformation, the currently parameterized number must be increased by one block element.

MD28090 $MC_MM_NUM_CC_BLOCK_ELEMENTS[<index>] = <current number **+ 1**>

5.1.2 Memory size for compile cycles

The machine data defines how much memory is reserved for compile cycle block elements.

The currently parameterized memory must be increased by 100 kB for the ROBX transformation.

MD28100 $MC_MM_NUM_CC_BLOCK_USER_MEM[<index>] = <current size + 100>

5.1.3 First transformation in the channel

The machine data sets which transformation is available as first transformation in the channel.

A value of 4100 should be entered for the ROBX transformation.

MD42100 $MC_TRAFO_TYPE_1 = **4100**

5.1.4 Axis assignment of the transformation

The machine data defines which axes at the input of the transformation are formed from which channel axes.

For the ROBX transformation, depending on how many axes are involved in the transformation, the corresponding channel axis numbers must be entered.

MD24110 $MC_TRAFO_AXES_IN_1[<index>] = <channel axis number>

<index>	**Meaning:**
	Basic axes (see Chapter Arrangement of the basic axes: (Page 27)**)**
0	1st basic axis
1	2nd basic axis

<index>	Meaning:
2	3rd basic axis
	Wrist axes (see Chapter Wrist axis identifier (Page 30)**)**
3	1st wrist axis
4	2nd wrist axis
5	3rd wrist axis
	Translatory additional axes in the basic coordinate system (BCS)
6	1st translatory additional axis (BCS)
7	2nd translatory additional axis (BCS)
8	3rd translatory additional axis (BCS)
	Rotary additional axes in the workpiece coordinate system (WCS)
9	1st rotary additional axis (WCS)
10	2nd rotary additional axis (WCS)
11	3rd rotary additional axis (WCS)
	Rotary additional axes in the flange coordinate system (WCS)
12	1st rotary additional axis (FL)
13	2nd rotary additional axis (FL)
14	3rd rotary additional axis (FL)

5.1.5 Assignment of the geometry axes to channel axes for transformation 1

The machine data defines how many translatory degrees of freedom are available for the transformation, and which geometry axes are formed from which channel axes for transformation 1.

Normally, the 3 geometry axes correspond to Cartesian axis directions X, Y and Z. In this case, the first 3 channel axis numbers should be taken from MD24110 $MC_TRAFO_AXES_IN_1 (Page 17).

- MD24120 $MC_TRAFO_GEO_AX_ASSIGN_TAB_1[0] = 1
- MD24120 $MC_TRAFO_GEO_AX_ASSIGN_TAB_1[1] = 2
- MD24120 $MC_TRAFO_GEO_AX_ASSIGN_TAB_1[2] = 3

5.2 Transformation-specific machine data

5.2.1 Kinematic category

With the machine data, the kinematic class active in the channel for the ROBX transformation is set.

MD62900 $MC_ROBX_KINCLASS = <kinematic class>

The standard kinematic class encompasses all currently available robot kinematics.

The special kinematics class is intended for expansions.

5.2.2 Axis type

For the 6 main axes of the ROBX transformation, the machine data specifies the particular axis type; e.g. linear or rotary axis or triangular/trapezoidal spindle:

MD62901 $MC_ROBX_AXES_TYPE[<main axis index>] = <axis type>

5.2.3 Type of special kinematics

The machine data specifies the kinematic type for special kinematics:

MD62902 $MC_ROBX_SPECIAL_KIN = <kinematic type>

Standard kinematics

The default value of the machine data corresponds to the standard kinematics with:

MD62900 $MC_ROBX_KINCLASS = 1

Special kinematics

Special kinematics is for example, kinematics where, when axis 2 moves, the angle of axis 3 remains constant in space.

For example "Comau NJ220 2.7", a robot with parallelogram structure to the drive of axis 3.

5.2.4 Number of transformed axes

The machine data specifies the number of main axes that are incorporated in the ROBX transformation.

Currently, only kinematics with 4 - 6 main axes are supported.

MD62905 $MC_ROBX_NUM_AXES = <value>

5.3 Machine data to parameterize the geometry

5.3.1 Modular principle

The machine geometry is described in the machine data in a modular fashion. The machine geometry is configured step-by-step from its base point to the tooltip using ROBX-specific frames (see ChapterCoordinate systems: Translation and rotation using frames (Page 13)) so that a closed, kinematic chain is obtained.

As can be seen from the diagram below, the ROBX transformation is then used to kinematically convert the operating point of the tool in the tool coordinate system (TCS), which is specified with respect to the basic coordinate system (BCS), into axis values for machine axes A1, A2, ... A6. The basic coordinate system (BCS) is identical to the world coordinate system (WO).

The tool operating point in the tool coordinate system (TCS) is specified in the part program as usual referred to the workpiece to be machined in the workpiece coordinate system (WCS). In addition, the workpiece coordinate system (WCS) can be offset with respect to the basic coordinate system (BCS) using the standard "programmable frames".

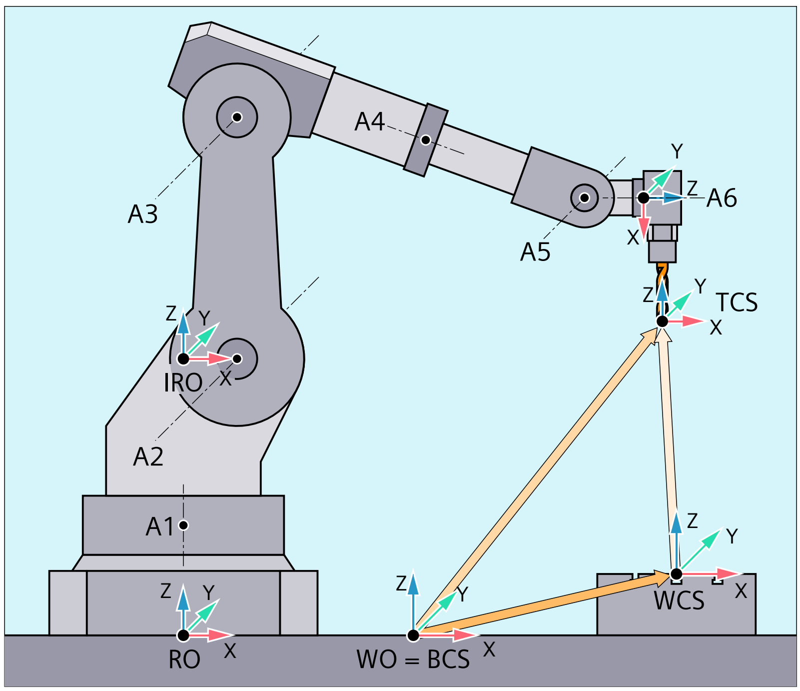


Figure 5-1 Closed kinematic loop illustrated using a robot as example

5.3.2 ROBX-specific frame: World coordinate system for the machine base point

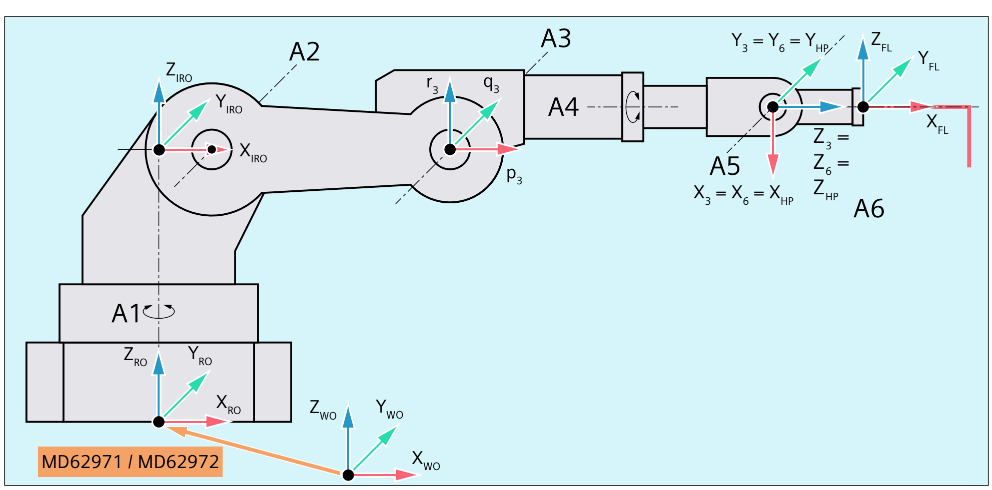


Figure 5-2 ROBX-specific frame: World coordinate system for the machine base point

The ROBX-specific frame, which connects the basic coordinate system (BCS) or the world coordinate system (WO) with the base point of the machine (RO), is parameterized with the two machine data.

Translatory component

The translatory component in the X, Y or Z direction is parameterized using the machine data:

MD62971 $MC_ROBX_TROWO_POS[<index>] = <value>

<index>	Meaning:
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

Rotation component

The rotation component around the X, Y or Z coordinate is parameterized using the machine data:

MD62972 $MC_ROBX_TROWO_RPY[<index>] = <value>

<index>	Meaning:
0	Rotation around the Z coordinate
1	Rotation around the Y coordinate
2	Rotation around the X coordinate

See also

Coordinate systems: Translation and rotation using frames (Page 13)

5.3.3 ROBX-specific frame: Machine base point for the first internal coordinate system

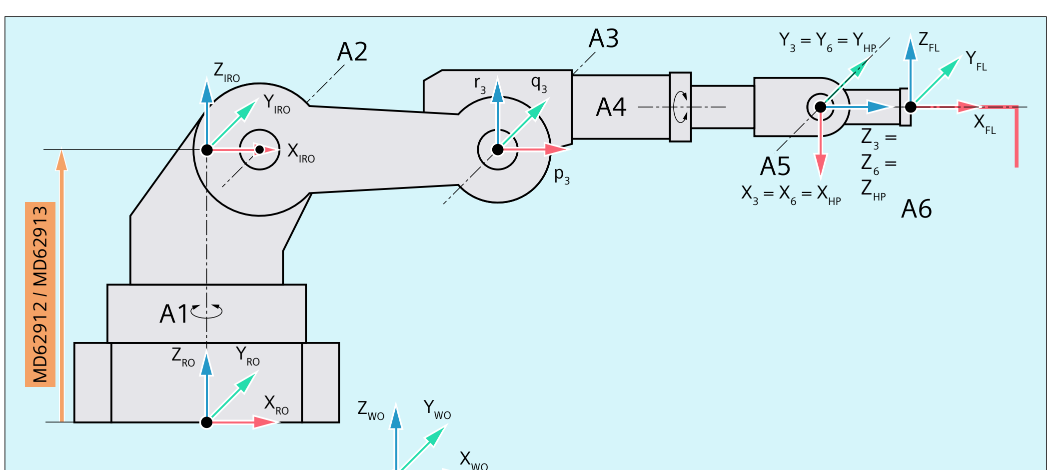


Figure 5-3 ROBX-specific frame: Machine base point for the first internal coordinate system

The ROBX-specific frame, which connects the base point coordinate system of the machine (RO) with the first internal coordinate system (IRO) defined by the transformation, is parameterized using the two machine data.

Translatory component

The translatory component in the X, Y or Z direction is parameterized using the machine data:

MD62912 $MC_ROBX_TIRORO_POS[<index>] = <value>

<index>	Meaning:
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

Rotation component

The rotation component around the X, Y or Z coordinate is parameterized using the machine data:

MD62913 $MC_ROBX_TIRORO_RPY[<index>] = <value>

<index>	Meaning:
0	Rotation around the Z coordinate
1	Rotation around the Y coordinate
2	Rotation around the X coordinate

More information

The frame defined using the machine data above, connects the base point coordinate system (RO) with the first internal coordinate system (IRO) defined by the transformation. This internal coordinate system (IRO) is specified for each basic axis type via the ROBX transformation and is plotted in the kinematic diagrams for the basic axis arrangements. It is specified with respect to the mathematical zero point of axis A1 and does **not** twist or shift with axis A1.

The base point coordinate system (RO) is located in the Cartesian zero point of the machine and can be specified in relation to a world coordinate system (WO) (see section ROBX-specific frame: World coordinate system for the machine base point (Page 20)). The world coordinate system corresponds to the basic coordinate system (BCS). If no FRAMES are programmed, then the basic coordinate system corresponds to the workpiece coordinate system (WCS).

5-axis kinematics

For 5-axis kinematics, the frame between the world coordinate system (WO) and the first internal coordinate system (IRO) defined by the transformation, is not subject to any restrictions.

4-axis kinematics

The following restrictions apply to 4-axis kinematics:

- The first rotary axis must always be in parallel/antiparallel to one of the coordinate axes of the world coordinate system (WO).
- No further restrictions apply to type SS basic axes.

- In the case of type CC, CS or SC basic axes, no further restrictions apply provided that the 4th axis is parallel to the last rotary basic axis.
- For all other basic axes and basic axes of type CC, CS or SC, if the 4th axis is perpendicular to the last rotary basic axis, the Z axis of world coordinate system (WO) must be parallel to the Z axis of the first internal coordinate system (IRO) defined by the transformation.

5.3.4 ROBX-specific frame: Last basic axis coordinate system for the wrist coordinate system

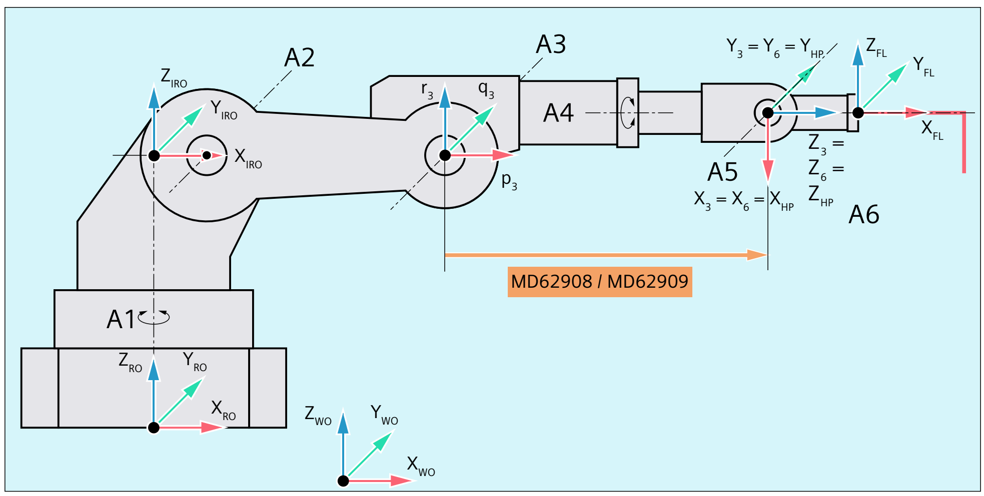


Figure 5-4 ROBX-specific frame: Last basic axis coordinate system for the wrist coordinate system

The ROBX-specific frame, which connects the coordinate system of the last basic axis A3 with the coordinate system of wrist axis A5, is parameterized using the two machine data.

Translatory component

The translatory component in the X, Y or Z direction is parameterized using the machine data:

MD62908 $MC_ROBX_TX3P3_POS[<index>] = <value>

<index>	Meaning:
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

Rotation component

The rotation component around the X, Y or Z coordinate is parameterized using the machine data:

MD62909 $MC_ROBX_TX3P3_RPY[<index>] = <value>

<index>	Meaning:
0	Rotation around the Z coordinate
1	Rotation around the Y coordinate
2	Rotation around the X coordinate

More information

The frame defined with the above machine data connects the coordinate system of the last basic axis A3 (p_3, q_3, r_3) with the coordinate system of wrist axis A5 ($X_3 = X_6 = X_{HP}$, $Y_3 = Y_6 = Y_{HP}$, $Z_3 = Z_6 = Z_{HP}$).

The coordinate axis Z must always lie in the direction of the 4th axis.

Corresponding to the number of axes to be incorporated in the transformation, this frame is subject to certain restrictions relating to the wrist and basic axes:

5-axis kinematics

For 5-axis kinematics, this frame can be freely selected in the following cases:

- If the basic axes are type SS.
- If the basic axes are of type CC, CS, or SC, then there must either be a central wrist axis or the 4th axis must be in parallel to the last rotary basic axis.
- If the basic axes are type NR or RR, then either a central wrist axis (ZEH) must be available or the 4th axis must be in parallel to the last basic rotary axis and an X flange must intersect with the 5th axis.
- A central wrist axis must be available if the basic axes are of type NN.

4-axis kinematics

For 4-axis kinematics, coordinate axis Z_3 must be parallel/antiparallel or perpendicular to the last basic axis.

5.3.5 ROBX-specific frame: Last wrist coordinate system for the flange coordinate system

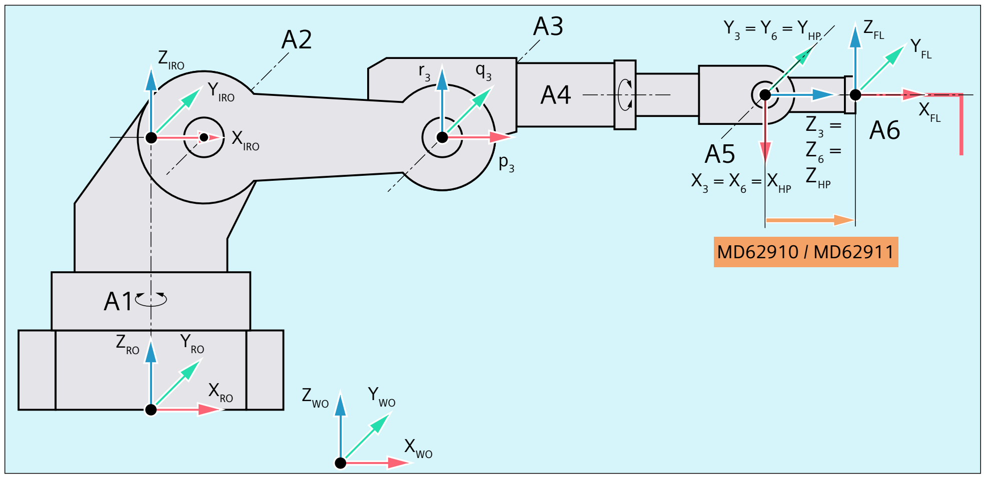


Figure 5-5 ROBX-specific frame: Last wrist coordinate system for the flange coordinate system

The ROBX-specific frame, which connects the coordinate system of the last wrist axis with the coordinate system of the flange (FL), is parameterized using the two machine data.

Translatory component

The translatory component in the X, Y or Z direction is parameterized using the machine data:

MD62910 $MC_ROBX_TFLWP_POS[<index>] = <value>

<index>	Meaning:
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

Rotation component

The rotation component around the X, Y or Z coordinate is parameterized using the machine data:

MD62911 $MC_ROBX_TFLWP_RPY[<index>] = <value>

<index>	Meaning:
0	Rotation around the Z coordinate
1	Rotation around the Y coordinate
2	Rotation around the X coordinate

More information

5-axis kinematics

For 5-axis kinematics, the X axis of the flange must intersect with the 5th axis. It is not permissible that it is in parallel with this.

5.3.6 ROBX-specific frame: Flange coordinate system for the coordinate system of the tool carrier

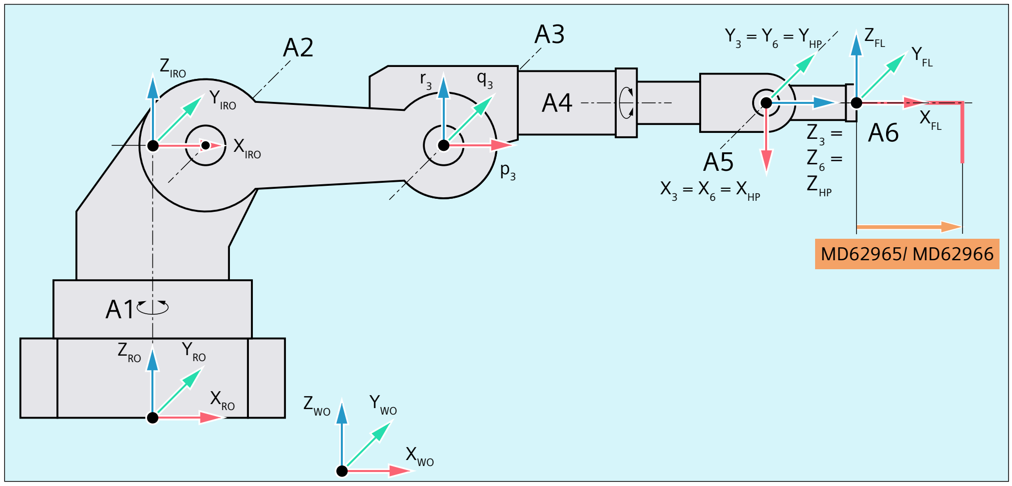


Figure 5-6 ROBX-specific frame: Flange coordinate system for the tool carrier coordinate system

The ROBX-specific frame, which connects the coordinate system of the flange (FL) with the coordinate system of the tool carrier (TC), is parameterized using the two machine data.

Translatory component

The translatory component in the X, Y or Z direction is parameterized using the machine data:

MD62965 $MC_ROBX_TTCFL_POS[<Index>] = <value>

<index>	Meaning:
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

Rotation component

The rotation component around the X, Y or Z coordinate is parameterized using the machine data:

MD62966 $MC_ROBX_TTCFL_POS[<Index>] = <value>

<index>	Meaning:
0	Rotation around the Z coordinate
1	Rotation around the Y coordinate
2	Rotation around the X coordinate

5.3.7 Arrangement of the basic axes:

The principle arrangement of the basic axes is specified using the machine data. The first 3 axes of the machine included in the transformation are generally referred to as the "basic axes". Within the framework of the ROBX transformation, basic axes must always be either parallel with one another or perpendicular to one another.

The following diagram shows the respective identifier (SS, CC, CS, ...), for the arrangement of the basic axes:

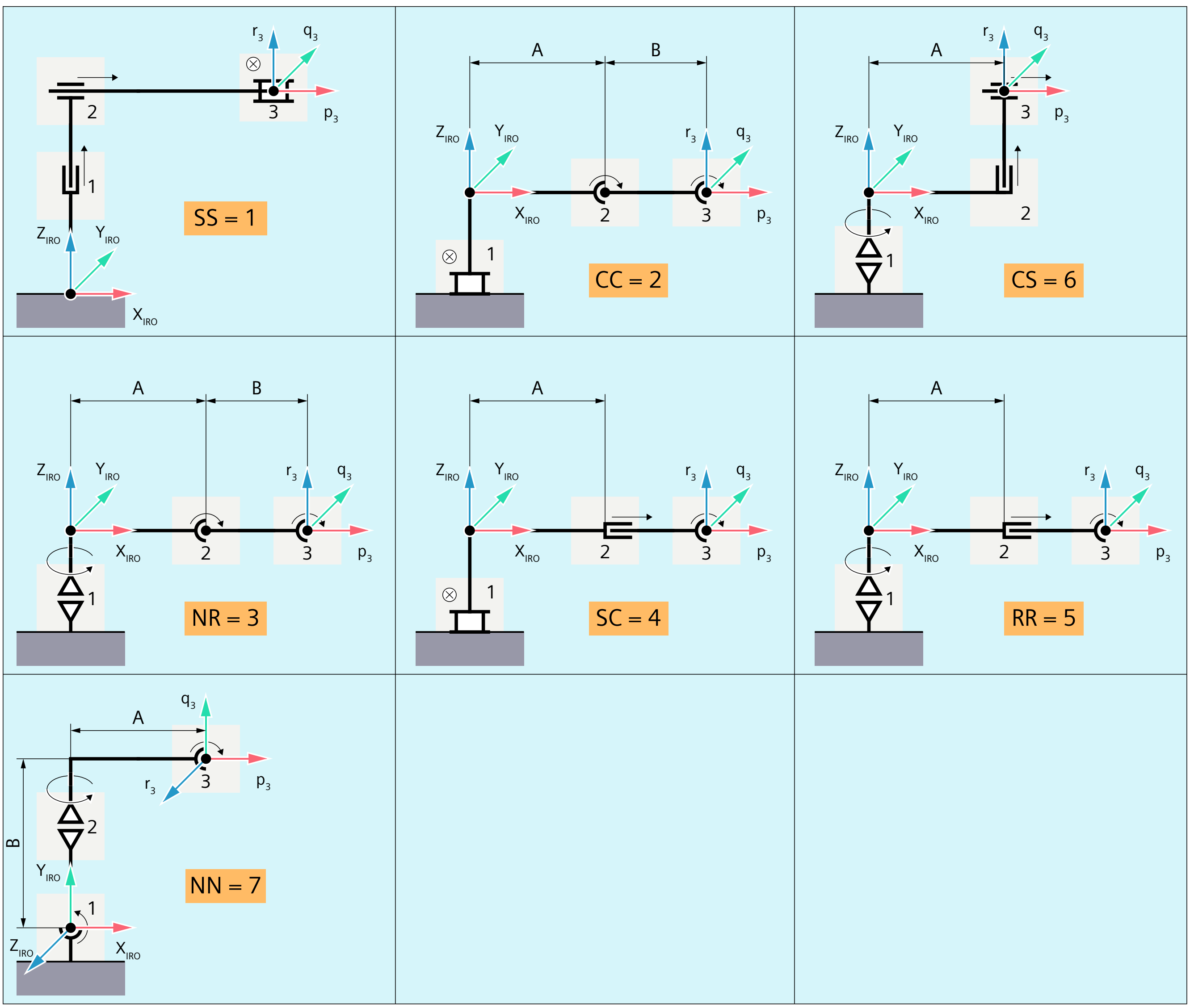


Figure 5-7 Basic arrangement of the basic axes

Machine data

The arrangement of the basic axes is entered in the machine data corresponding to the identifier:

MD62903 $MC_ROBX_MAIN_AXES = <value>

<value>	Identifier	Meaning
1	SS	Gantry (3 linear axes, at right angles)
2	CC	Scara (1 linear axis, 2 rotary axes (in parallel))
3	NR	Articulated arm (3 rotary axes (2 axes in parallel))
4	SC	Scara (2 linear axes, 1 rotary axis (axis of rotation))
5	RR	Articulated arm (1 linear axis, 2 rotary axes (perpendicular))
6	CS	Scara (2 linear axes, 1 rotary axis (axis of rotation))
7	NN	Articulated arm (3 rotary axes)

5.3.8 Basic axis lengths A and B

Lengths A and B of the two basic axes are specified using the machine data. The length data for no (SS), one (CS, SC, RR) or both (CC, NR, NN) basic axes must be entered corresponding to the arrangement of the basic axes (Page 27).

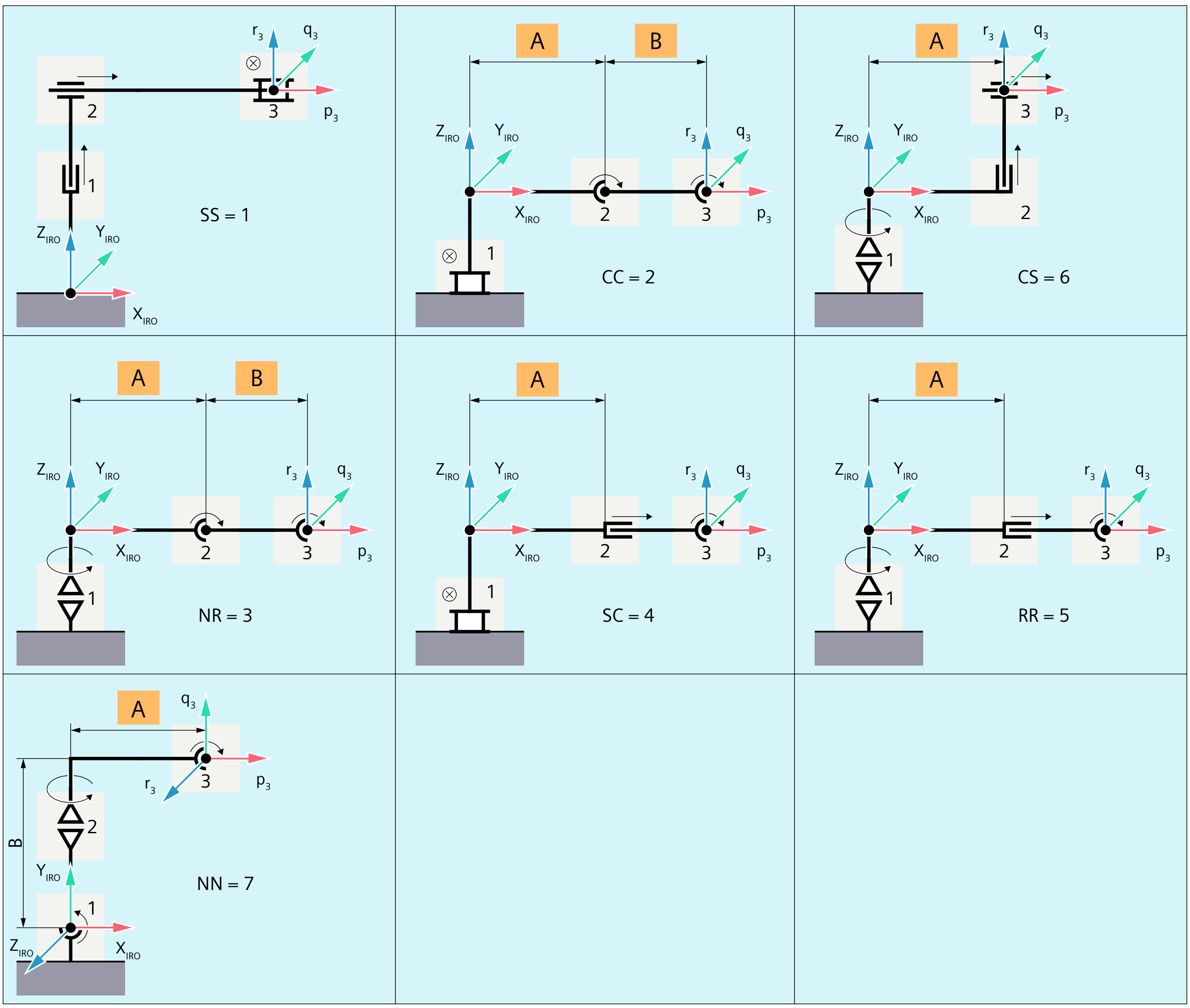


Figure 5-8 Basic axis config_axes_A_B

Machine data

MD62907 $MC_ROBX_MAIN_LENGTH_AB[<index>] = <length>

<index>	Meaning
0	Length of basic axis A
1	Length of basic axis B

5.3.9 Arrangement of the 4th axis

Using the machine data, for kinematics with more than 3 axes, it is specified as to whether the 4th axis is mounted parallel or not parallel to the last rotary basic axis:

MD62906 $MC_ROBX_A4PAR = <value>

<value>	Meaning
0	**Not** parallel
1	Parallel

5.3.10 Wrist axis identifier

The machine data is used to specify which wrist axis type is available. Axes 4 - 6 are generally designated as wrist.

MD62904 $MC_ROBX_WRIST_AXES = <value>

<value>	Meaning
1	There are no wrist axes
2	Central wrist axis
3	Inclined wrist axis
4	---
5	Angled wrist axis
6	Inclined wrist axis

Note

Wrist axis types

Currently, the ROBX transformation can only process wrists with rotary axes.

Central and inclined wrist axis

For wrists with less than 3 axes, the identifier for **central wrist** or **inclined wrist axis** must be identified in the machine data:

5.3.11 Wrist axes: Central wrist axis

A central wrist is characterized by the fact that all axes of the wrist are perpendicular to one another in pairs and all axes of the wrist intersect at one point.

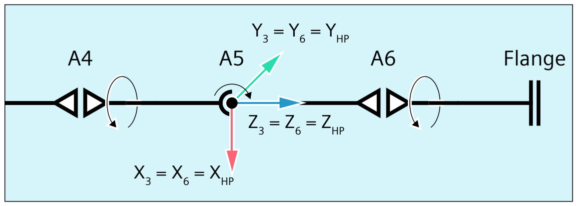


Figure 5-9 Central wrist axis

Machine data

The machine data for a central wrist are as follows:

Number: MD	Identifier: $MC_	Value	Comment
62904	ROBX_WRIST_AXES	2	Central wrist axis
62914	ROBX_DHPAR4_5A	[0.0, 0.0]	a: Offset X coordinate
62915	ROBX_DHPAR4_5D	[0.0, 0.0]	d: Offset Z coordinate
62916	ROBX_DHPAR4_5ALPHA	[-90.0, 90.0]	α: Rotation around the X coordinate

5.3.12 Wrist axes: Inclined wrist axis

An inclined wrist axis is characterized by the fact that, just the same as a central wrist, all axes of the wrist intersect at one point; however, the axes are arranged at various angles with respect to another.

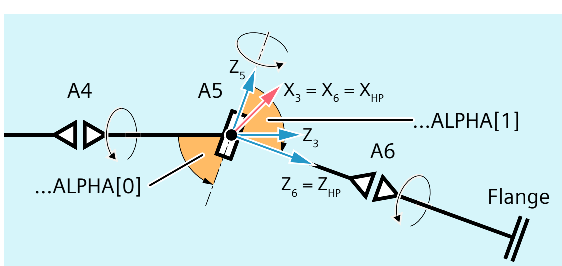


Figure 5-10 Inclined wrist axis

Machine data

The machine data for an inclined wrist axis are as follows:

Number: MD	Identifier: $MC_	Value	Comment
62904	ROBX_WRIST_AXES	3	Inclined wrist axis
62914	ROBX_DHPAR4_5A	[0.0, 0.0]	a: Offset X coordinate
62915	ROBX_DHPAR4_5D	[0.0, 0.0]	d: Offset Z coordinate
62916	ROBX_DHPAR4_5ALPHA	[$\alpha_{4\to5}$, $\alpha_{5\to6}$]	α: Rotation around the X coordinate

5.3.13 Wrist axes: Angled wrist axis

An angled wrist axis is characterized by the fact that the axes of the wrist are perpendicular to each other, but usually do not intersect.

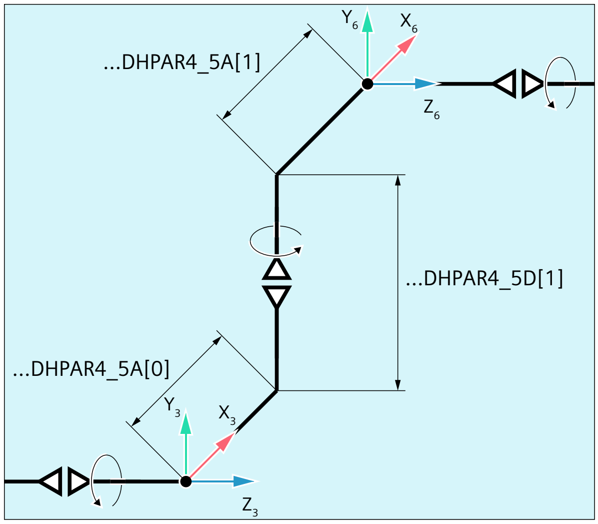


Figure 5-11 Angled wrist axis

Machine data

The machine data for an angled wrist axis are as follows:

Number: MD	Identifier: $MC_	Value	Comment
62904	ROBX_WRIST_AXES	5	Angled wrist axis
62914	ROBX_DHPAR4_5A	[$a_{4\rightarrow5}$, $a_{5\rightarrow6}$]	a: Offset X coordinate
62915	ROBX_DHPAR4_5D	[0.0, $d_{5\rightarrow6}$]	d: Offset Z coordinate
62916	ROBX_DHPAR4_5ALPHA	[-90, 90]	α: Rotation around the X coordinate

5.3.14 Wrist axes: Inclined wrist axis

An inclined wrist axis is characterized by the fact that the axes of the wrist neither intersect nor are they perpendicular with one another.

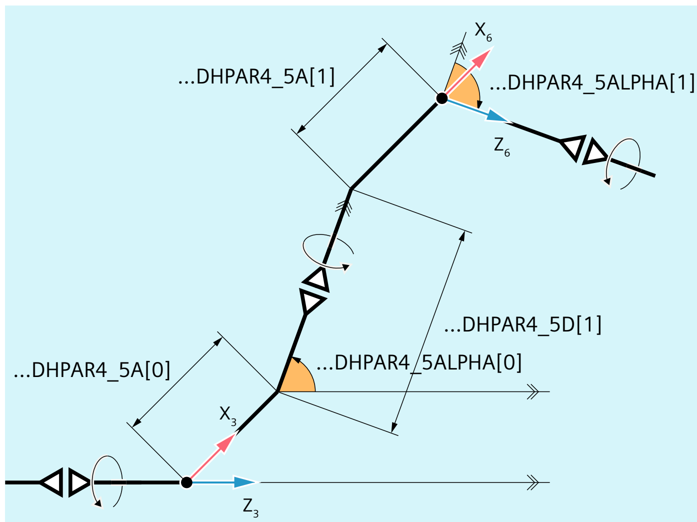


Figure 5-12 Inclined wrist axis

Machine data

The machine data for an inclined wrist axis are as follows:

Number: MD	Identifier: $MC_	Value	Comment
62904	ROBX_WRIST_AXES	6	Inclined wrist axis
62914	ROBX_DHPAR4_5A	[$a_{4\rightarrow5}$, $a_{5\rightarrow6}$]	a: Offset X coordinate
62915	ROBX_DHPAR4_5D	[0.0, $d_{5\rightarrow6}$]	d: Offset Z coordinate
62916	ROBX_DHPAR4_5ALPHA	[$\alpha_{4\rightarrow5}$, $\alpha_{5\rightarrow6}$]	α: Rotation around the X coordinate

5.3.15 Changing the axis sequence

As a result of the parameterized basic axis arrangement (Page 27) (SS, CC, CS, ...), the ROBX transformation assumes a defined arrangement of the axes at the machine. If the arrangement of the axes at the machine does not match the parameterized basic axis arrangement, for certain kinematics, the axis sequence can be interchanged so that internally the axis arrangement corresponds to the parameterized basic axis arrangement and the kinematic response at the machine remains the same:

MD62620 $MC_TRAFO6_AXIS_SEQ[<index>] = <axis number>

Parameter	Value	Meaning
<index>	0 ... 5	The new axis number 1 ... 6 of the ROBX transformation
<axis number>	1 ... 6	The axis numbers parameterized in MD24110 $MC_TRAFO_AXES_IN_1 (Page 17)

Basic axis kinematics that can be changed

For the following basic axis kinematics, the axes can be interchanged without the response at the machine changing:

Basic axis kinematics	Options for changing axis sequence
SS, CC	Any
SC	1 and 2
CS	2 and 3

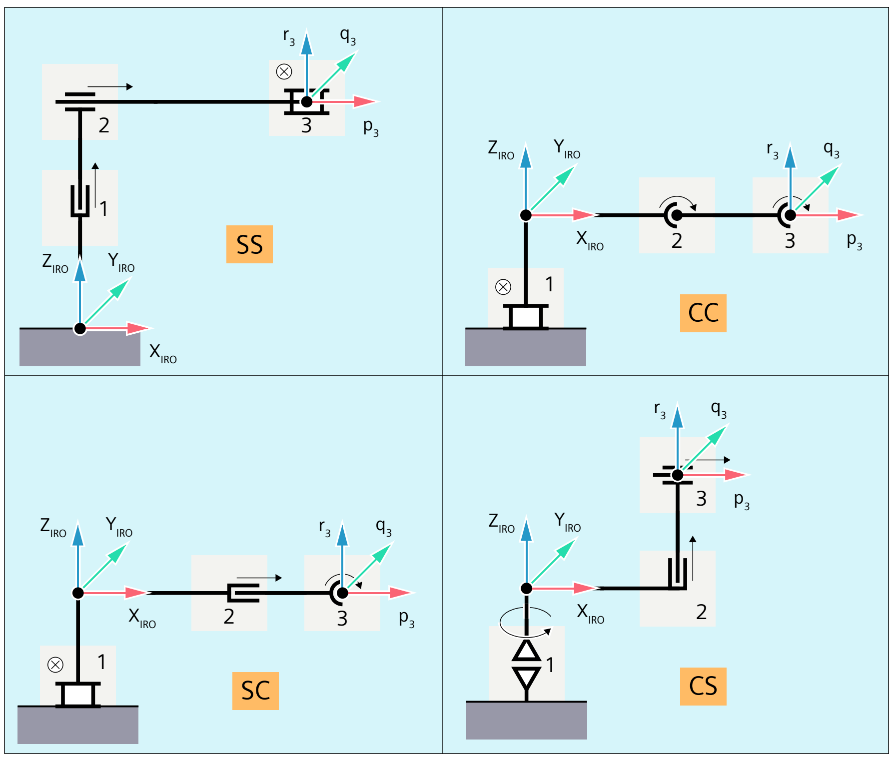


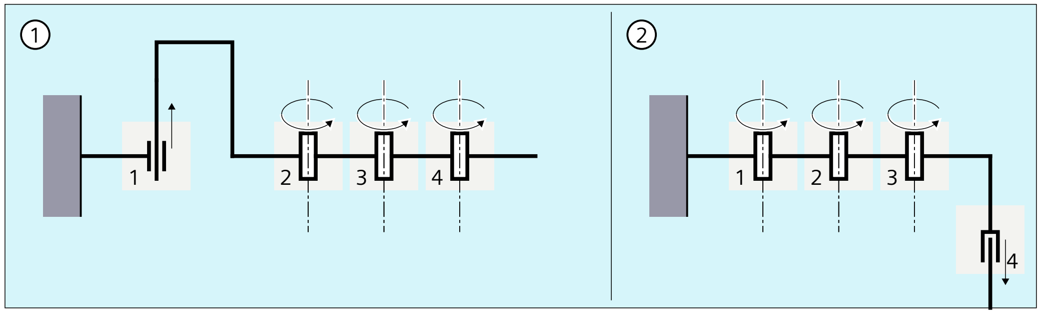
Figure 5-13 Basic axis kinematics

Examples

Example 1

In the following diagram, under ① the machine kinematics in an CC arrangement are shown with the 1st wrist axis parallel to the last rotary basic axis. As a consequence, it is therefore not necessary to change the axis arrangement.

Under ②, the machine kinematics does not correspond to the CC kinematics parameterized in the machine data (Page 27).



1 Machine kinematics equal to the CC kinematics
2 Machine kinematics not equal to the CC kinematics

Figure 5-14 CC kinematics: Rearrangement, example 1

In case ①, the axis arrangement in machine data MD62620 does not have to be changed.

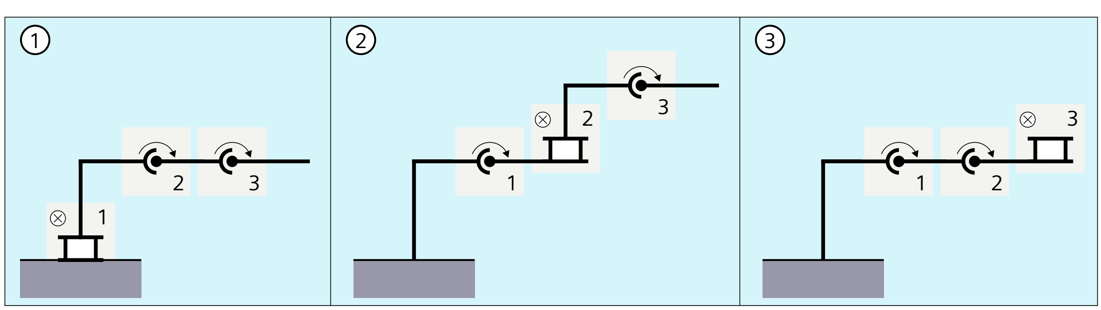
In case ②, the axis arrangement must be changed in the machine data to align the machine kinematics to the parameterized CC kinematics:

Machine data	Arrangement ①	Arrangement ②
MD62620 $MC_TRAFO6_AXIS_SEQ[0] =	1	4
MD62620 $MC_TRAFO6_AXIS_SEQ[1] =	2	1
MD62620 $MC_TRAFO6_AXIS_SEQ[2] =	3	2
MD62620 $MC_TRAFO6_AXIS_SEQ[23] =	4	3

However, by changing the axis arrangement, the same machine motion is obtained, as it is of no significance whether the translatory axis is axis 1 or axis 4.

Example 2

The machine kinematics in a CC arrangement are shown in the following diagram, under ①. In cases ② and ③, the machine kinematics do not correspond to the CC kinematics parameterized (Page 27) in the machine data. As a consequence, in cases ② and ③, the axis arrangement must be adapted using machine data MD62620.



1 Machine kinematics equal to the CC kinematics
2 Machine kinematics not equal to the CC kinematics
3 Machine kinematics not equal to the CC kinematics

Figure 5-15 CC kinematics: Rearrangement, example 2

Machine data	Arrangement ①	Arrangement ②	Arrangement ③
MD62620 $MC_TRAFO6_AXIS_SEQ[0] =	1	2	3
MD62620 $MC_TRAFO6_AXIS_SEQ[1] =	2	1	1
MD62620 $MC_TRAFO6_AXIS_SEQ[2] =	3	3	2

5.3.16 Changing the traversing direction

Internally, for each of the axes involved in the transformation, the ROBX transformation assumes a certain traversing direction. If this mathematical traversing direction does not match the physical traversing direction at the machine itself, then this can be aligned using the machine data:

MD62918 $MC_ROBX_AXES_DIR[<index>] = <value>

Parameter	Value	Meaning
<index>	0 ... 5	Axis 1 ... 6 of the ROBX transformation
<value>	+1	Mathematical traversing direction kept
	-1	Inverse mathematical traversing direction

5.3.17 Adapting the zero points of the axes

Internally, for each of the axes involved in the transformation, the ROBX transformation assumes a certain zero position. If this mathematical traversing direction does not match the physical zero position at the machine itself, an appropriate offset value can be parameterized using the machine data.

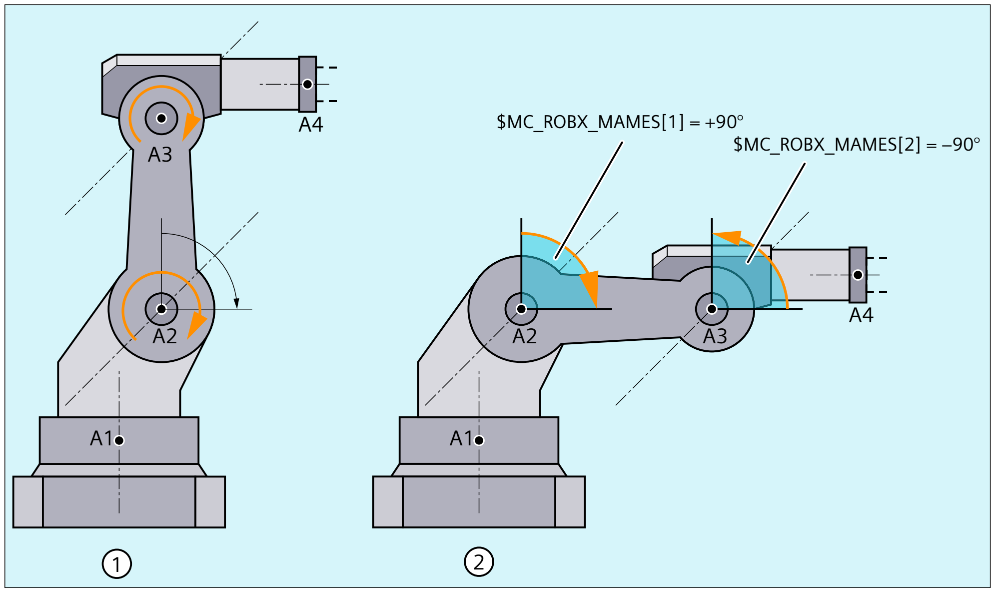
The offset value is obtained from the difference between the physical and the mathematical zero positions, referred to the mathematically positive direction of rotation of the axis.

MD62917 $MC_ROBX_MAMES[<index>] = <offset value>

Parameter	Value	Meaning
<index>	0 ... 5	Axis 1 ... 6 of the ROBX transformation

Example

The following diagram shows articulated arm kinematics in the physical zero position and then in the mathematical zero position of axes A2 and A3.



① Physical zero position and positive direction of rotation of axes A2 and A3
② Mathematical zero position and zero offset of axes A2 and A3

Figure 5-16 Physical and mathematical zero position

To transition a physical zero position into the mathematical zero position, the following values should be entered into the machine data assuming an 90° offset:

- MD62917 $MC_ROBX_MAMES[1] = +90 ; A2 is the 2nd axis of the transformation
- MD62917 $MC_ROBX_MAMES[2] = -90 ; A3 is the 3rd axis of the transformation

5.3.18 Procedure when an override controller is active

In order to avoid unnecessarily limiting velocities and acceleration rates in conjunction with ROBX, using the following machine data, substitute limit values for velocity and acceleration components for directions that have a very small component with respect to the main direction of motion should be entered.

Override controller

For ROBX, the override controller is always active if movements are interpolated whose dynamic response limits cannot be secured or not completely secured using the look-ahead function of the control system. The override controller is then active in the following cases:

- When ROBX transformation is active in the JOG mode
- For motion override in realtime (e.g. $AA_OFF[X])
- For velocity limitations initiated by a Safety Integrated function (SG level active)

Velocity limit value

Using the machine data, substitute limit values can be specified for velocities for translatory motion in the direction of the X, Y and Z axes of the basic coordinate system (BCS) when traversing with the override controller active:

MD62929 $MC_ROBX_VELCP[<index>] = <value>

<index>	Meaning
0	Velocity limit value in the X direction of the basic coordinate system (BCS)
1	Velocity limit value in the Y direction of the basic coordinate system (BCS)
2	Velocity limit value in the Z direction of the basic coordinate system (BCS)

Acceleration limit value

Using the machine data, a substitute limit value can be specified for acceleration for translatory motion in the direction of the X, Y and Z axes of the basic coordinate system (BCS) when traversing with the override controller active:

MD62930 $MC_ROBX_ACCCP[<index>] = <value>

<index>	Meaning
0	Acceleration limit value in the X direction of the basic coordinate system (BCS)
1	Acceleration limit value in the Y direction of the basic coordinate system (BCS)
2	Acceleration limit value in the Z direction of the basic coordinate system (BCS)

Angular velocity limit value

Using the machine data, a substitute limit value can be specified for the angular velocity for translatory motion in the direction of the X, Y and Z axes of the basic coordinate system (BCS) when traversing with the override controller active:

MD62931 $MC_ROBX_VELORI[<index>] = <value>

<index>	Meaning
0	Angular velocity limit value in the basic coordinate system (BCS); A angle
1	Angular velocity limit value in the basic coordinate system (BCS); B angle
2	Angular velocity limit value in the basic coordinate system (BCS); C angle

Angular acceleration limit value

Using the machine data, a substitute limit value can be specified for the angular acceleration for translatory motion in the direction of the X, Y and Z axes of the basic coordinate system (BCS) when traversing with the override controller active:

MD62932 $MC_ROBX_ACCORI[<index>] = <value>

<index>	Meaning
0	Angular acceleration limit value in the basic coordinate system (BCS); A angle
1	Angular acceleration limit value in the basic coordinate system (BCS); B angle
2	Angular acceleration limit value in the basic coordinate system (BCS); C angle

Reduction factor

Using the machine data, a reduction factor can be specified as reserve for the maximum velocity. This ensures that if the override controller increases the velocity, the maximum velocity is not exceeded. The value should be seen as a factor that is referred to the maximum velocity.

MD62934 $MC_ROBX_DYN_LIM_REDUCE = <value>

Time constant PT1 filter

Using the machine data, the time constant of the PT1 filter of the override controller can be set so that the controller does not oscillate:

MD62935 $MC_ROBX_VEL_FILTER_TIME = <value>

5.3.19 Special functions

Activation

Various special functions of the ROBX transformation can be activated using the machine data:

MD62953 $MC_ROBX_SPECIAL_FEATURE_MASK.Bit <n> = <value>

Bit	Value	Meaning
0	0	For 4-axis SCARA kinematics, the **C angle** is controlled in the basic coordinate system (BCS) in the range [-180°, 180°].
	1	For 4-axis SCARA kinematics, the **C angle** is controlled in the basic coordinate system (BCS) in absolute terms. See below, paragraph "Absolute C angle"

Bit 0: C angle in the basic coordinate system (BCS)

Absolute C angle

For 4-axis SCARA kinematics, normally the **C angle** is defined in the basic coordinate system (BCS) only in the range [-180° < C ≤ 180°]. This means that the C angle in the display jumps to -179° at the transition from 180° to 181°. When the bit is set, a jump as a result of the C angle being controlled in absolute terms is avoided.

For the following 4-axis kinematics, the C angle is controlled in absolute terms:

- CC kinematics
- SC kinematics
- CS kinematics

This is on the condition that the 4th axis is arranged parallel to the last rotary basic axis. See MD62906 $MC_ROBX_A4PAR (Page 30).

Angle C is then calculated from the sum of the rotary axis angles. The rotational components parameterized in the following machine data are taken into consideration when doing this:

- MD62917 $MC_ROBX_MAMES (Page 36)
- MD62918 $MC_ROBX_AXES_DIR (Page 36)

5.3.20 Offset for calculating the absolute C angle

If additional rotations are obtained as a result of the rotation components parameterized in MD62913 $MC_ROBX_TIRORO_RPY (Page 21) and MD62911 $MC_ROBX_TFLWP_RPY (Page 25), then these must also be compensated. For 4-axis SCARA kinematics, the angle, which is displayed as "virtual orientation angle" C at the mathematical zero position of the axes, must be entered in the machine data with inverted sign:

MD62954 $MC_ROBX_C_ANGLE_OFFSET = <value>

Requirement

MD62953 $MC_ROBX_SPECIAL_FEATURE_MASK.bit 0 (Page 39) == TRUE

5.3.21 Limit angle for collision detection

For robots with a parallelogram structure to the drive of axis 3 (MD62902 $MC_ROBX_SPECIAL_KIN (Page 19) = 2), for example, Comau NJ220 2.7 (see the following diagram), as a result of the inherent design, for certain axis positions a collision can occur with the robot housing.

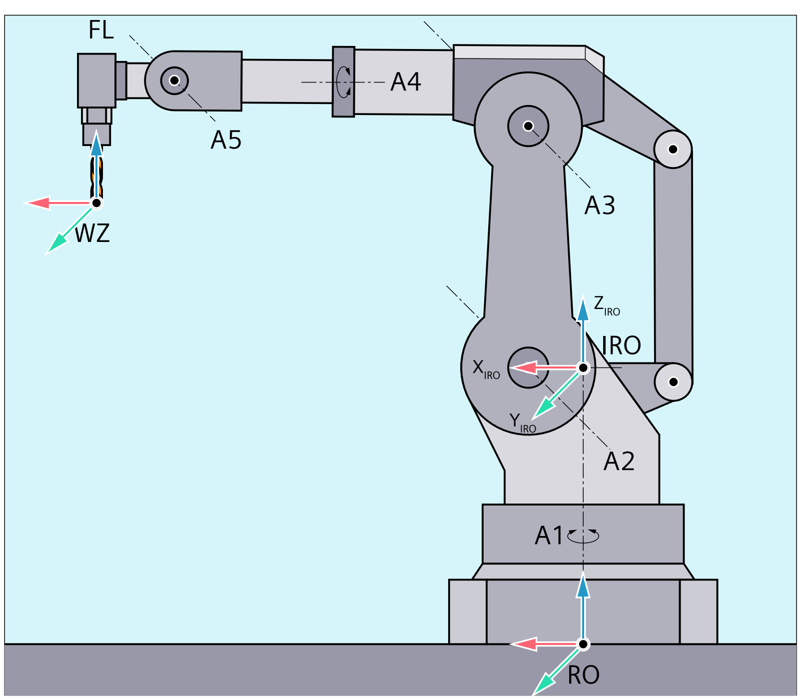


Figure 5-17 Schematic design: Comau NJ220 2.7

These types of axis positions can be emulated using the following formula:

$q_{min} < q_{3DH} - q_{2DH} < q_{max}$

with:

- $q_{2DH} = -A2_{MCS} + 90°$; $A2_{MCS}$: Position of the 2nd axis in the machine coordinate system
- $q_{3DH} = A3_{MCS} + 270°$; $A3_{MCS}$: Position of the 3rd axis in the machine coordinate system

To avoid these types of collisions, the lower and upper limit angle can be specified using the machine data:

MD62964 $MC_ROBX_SW_LIMITS_PARALLEL[<index>] = <value>

<index>	Meaning
0	Lower limit angle q_{min}
1	Upper limit angle q_{min}

Check in preprocessing

In the **AUTOMATIC** and **MDA** operating modes and when the ROBX transformation is **active**, any potential violation of the limit angle is checked in **preprocessing** and, if necessary, alarm 75328 is output.

Check in the main run

In the **JOG** mode and for **all movements** when the ROBX transformation is **inactive**, any potential violation of the limit angle is checked in the **main run** and, if necessary, alarm 75329 is output.

Retracting

Retracting to the permissible range is permissible both in Cartesian terms as well as for specific axes.

Programming

6

6.1 Tool programming

6.1.1 Tool orientation based on the OEM cutting edge data

Tool length compensations are programmed in standard tool parameters $TC_DPn.

Tool orientation is programmed in OEM tool parameters $TC_DP**C**n.

Tool length compensation (default)

Tool**length**compensations are programmed in standard tool parameters:

Standard tool parameters	Meaning
$TC_DP1[1,1]	(Tool type)
$TC_DP3[1,1]	(Z) Length (for G17)
$TC_DP4[1,1]	(Y) Length (for G17)
$TC_DP5[1,1]	(X) Length (for G17)

Tool length compensations act dependent on the active machining plane. Tool lengths are taken into consideration in the directions of the flange coordinate system (see the following diagram).

Tool orientation (OEM)

Tool**orientations** are programmed in OEM tool parameters $TC_DPCn, with n = 1...10.

Machine data can be used to parameterize from which OEM tool parameter $TC_DPC(n) the parameters for tool orientation start. On this topic, see the following Section "Machine data" > "Start parameters of the tool orientation".

Values $TC_DPCn, $TC_DPCn+1 and $TC_DPCn+2 correspond to the 3 RPY angles A, B and C, with which the tool can be rotated with respect to the flange coordinate system. The offset for the tool lengths ($TC_DPn) is first realized and then the rotations ($TC_DPCn) around the tool tip (TCP) (see the following diagram).

OEM tool parameters	Meaning
$TC_DPC1[1,1]	...
...	...
$TC_DPC(n)[1,1]	1st angle A (rotation around Z)
$TC_DPC(n+1)[1,1]	2nd angle B (rotation around Y)
$TC_DPC(n+2)[1,1]	3rd angle C (rotation around X)
...	...
$TC_DPC10[1,1]	...

Example for tool orientation

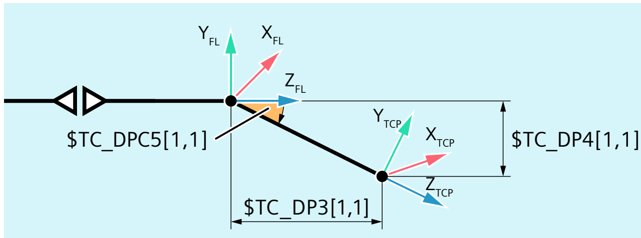


Figure 6-1 Tool orientation

Machine data

Before being used, OEM tool parameters must first be set up using the subsequent machine data They then can be used in precisely in the same way as standard tool parameters. OEM tool parameters are **displayed** in the standard tool table.

Additional memory for user data

Additional memory for user data can be requested using the machine data:

- MD18080 $MN_MM_TOOL_MANAGEMENT_MASK, bit 2 = 1

Number of OEM tool parameters per tool cutting edge

The number of parameters for each tool cutting edge are parameterized for the tool orientation using the machine data:

- MD18096 $MN_MM_NUM_CC_TOA_PARAM = 3

Start parameters for the tool orientation

In the machine data, the number **n** of start parameter $TC_DPC**n** is specified from which the ROBX transformation uses the 3 parameters for tool orientation:

- MD62936 $MC_ROBX_CC_TOA_START_NUM = <start number>

Tool orientation for active tool management

For active tool management and when tool orientation is programmed using tool parameter $TC_DPCx, tool number Tx of the ROBX transformation used must be communicated using procedure CC_SET_TOOL_NUM( ):

Program code	**Comment**
`N10 DEF INT TOOL_NUM`	`; Create INT variable`
`...`	
`N100 T1`	`; Activate tool No.`
`N110 GETSELT( TOOL_NUM )`	`; Read active Tool No.`
`N120` **`CC_SET_TOOL_NUM( TOOL_NUM )`**	**`; Active tool No. at ROBX transf.`**
`N130 D1`	`; Activate 1st tool cutting edge`
`N140 M6`	

6.1.2 Tool orientation based on system variables

Instead of using OEM tool parameters (Page 43), the tool orientation can also be programmed via channel-specific system variables $TC_DPVn and $TC_DPVNn.

References

Detailed information about programming the tool orientation using system variables $TC_DPVn and $TC_DPVNn is provided in:

"Tools" Function Manual

Orientation vector

The orientation vector is programmed using system variables $TC_DPVn, with n = 3 ... 5:

System variable	Meaning
$TC_DPV[1,1]	Predefined orientation vector (0)
$TC_DPV3[1,1]	L1 component of the orientation vector (Z for G17)
$TC_DPV4[1,1]	L2 component of the orientation vector (Y for G17)
$TC_DPV5[1,1]	L3 component of the orientation vector (X for G17)

Normal vector

The orientation normal vector is programmed using system variables $TC_DPV**N**n, with n = 3 ... 5:

System variable	Meaning
$TC_DPVN3[1,1]	L1 component of the normal vector (Z for G17)
$TC_DPVN4[1,1]	L2 component of the normal vector (Y for G17)
$TC_DPVN5[1,1]	L3 component of the normal vector (X for G17)

Machine data

System variable for normal vector

Using the machine data, in addition to the standard system variables $TC_DPVn, the system variables for the normal vector of the tool orientation are activated:

- MD18114 $MN_MM_ENABLE_TOOL_ORIENT = **3**

Start parameters for the tool orientation

In order that for the tool orientation, the ROBX transformation uses system variables $TC_DPVn and $TC_DPVNn instead of OEM tool parameters (Page 43), in the machine data, the start number for the tool parameters must be set to a value of -1.

- MD 62936 $MC_ROBX_CC_TOA_START_NUM = **-1**

6.1.3 Orientation programming for 5/6-axis kinematics

Using orientation vector

For 5-axis kinematics, when programming the tool orientation using the orientation vector, it is assumed that the orientation vector corresponds to the X component of the tool.

Using orientation angle

When programming the tool orientation using the orientation angle (RPY angle according to robotics definition), the X component of the tool is used as the starting point for rotations. For this purpose, the vector in the X tool direction is initially rotated through angle A around the Z axis and then through angle B around the rotated Y axis.

The rotation by the angle C is not possible for 5-axis kinematics because of the restricted degrees of freedom for the orientation.

Adapting the tool orientation

Using the machine data, the flange coordinate system can be set on the user side so that for a 5 or 6-axis kinematics, the Z coordinate points in the tool direction:

MD62948 $MC_ROBX_TFL_EXT_RPY[<index>] = <angle>

<index>	Meaning
0	Angle of rotation around RPY angle A
1	Angle of rotation around RPY angle B
2	Angle of rotation around RPY angle C

Definition of the tool direction

The machine data is used to set as to whether the tool direction is according to the robotic or NC convention and whether tool lengths are negated.

If the tool lengths are negated, then the same response is obtained as for a 5-axis machine tool, where the tool length is considered to be positive if the tool vector points into the flange.

MD62949 $MC_ROBX_TOOL_DIR = <value>

<value>	Meaning
0	**Robotic** convention • Tool direction is X • Rotation sequence is Z, Y • Tool lengths are **not** negated
+1	**NC** convention • Tool direction is Z • Rotation sequence is X, Y • Tool lengths are negated
-1	**Robotic** convention • Tool direction is X • Rotation sequence is Z, Y • Tool lengths are negated

Rotation sequence

The machine data influences the rotation sequence of the virtual orientation axes. When using the **NC**convention, the vector is first rotated in the Z tool direction through angle A around the X axis and then through angle B around the rotated Y axis.

Rotation through angle C depends on the kinematics:

- 5-axis kinematics: Rotation through angle C is **not** possible as the degree of freedom is not available.
- 6-axis kinematics: Rotation is through angle C around the Z axis that has been rotated twice.

Restrictions

The ROBX transformation does not support all rotation sequences that can be set using G group 50: "Orientation programming". Only the following are supported:

- Rotation sequence Z Y ’X“ corresponding to `ORIRPY`
- Rotation sequence X Y’ Z“ corresponding to `ORIRPY2`

Overview

<value> = 0 or -1 **(robotic convention)**	**<value> = 1** **(NC convention)**
Angle A = ROT(Z)	Angle A = ROT(X)
Angle B = ROT(Y’)	Angle B = ROT(Y’)
Angle C = ROT(X“)	Angle C = ROT(Z“)
G group 50 = ORIRPY or ORIVIRTx with:	G group 50 = ORIRPY2 or ORIVIRTx with:
$MC_ORIAX_TURN_TAB[0] = 3	$MC_ORIAX_TURN_TAB[0] = 1
$MC_ORIAX_TURN_TAB[1] = 2	$MC_ORIAX_TURN_TAB[1] = 2
$MC_ORIAX_TURN_TAB[2] = 1	$MC_ORIAX_TURN_TAB[2] = 3

Examples: Parameterizing a 6-axis robot 7

7.1 Kinematics type NR:

The following chapter describes the parameterization of a 6-axis articulated arm robot, type NR, shown as example in the diagram.

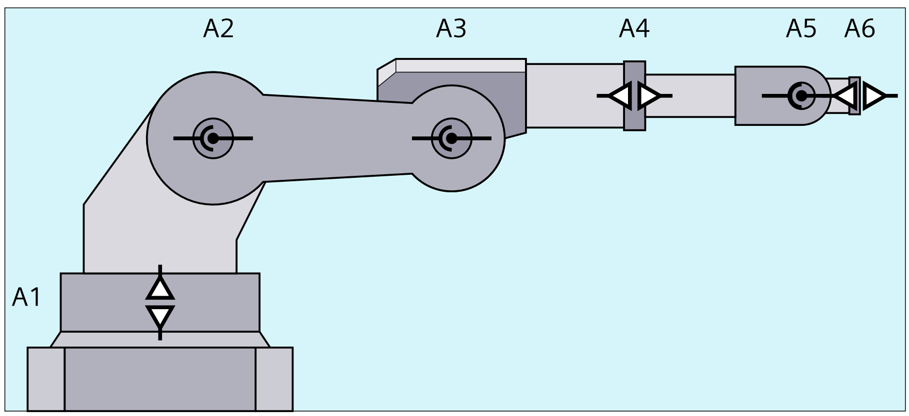


Figure 7-1 Typical 6-axis articulated arm robot

Kinematics

Corresponding to type NR, the robot is based on 6 rotary axes, which are arranged as shown in the following diagram:

1. The first rotary axis is an axis of rotation.
2. The second and third rotary axes are swiveling axes, which move perpendicular to the first rotary axis.
3. The robot wrist comprises 3 additional rotary axes, which all intersect at one point, the wrist point (WP).
   For the special case where these 3 rotary axes do not intersect at one point, see Chapter "Geometry parameters: Inclined wrist axis (Page 54)"

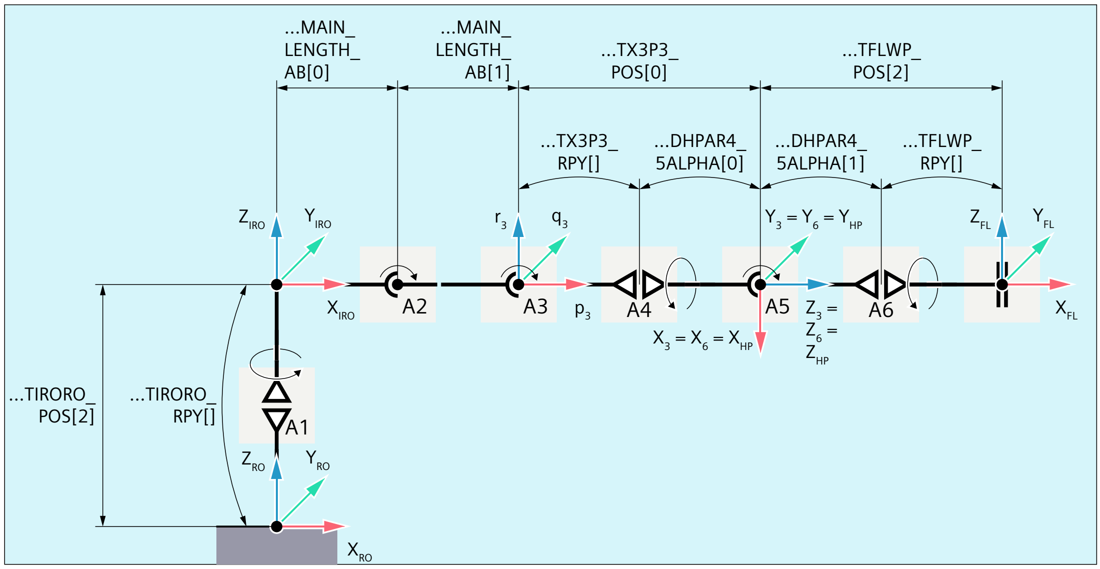


Figure 7-2 NR kinematics

7.1.1 Machine data of the standard system

Number: MD	Identifier: $MC_	Parameterization
24100	TRAFO_TYPE	A value of **4100** should be entered as transformation type for the ROBX transformation.
24110	TRAFO_AXES_IN_1	The **channel axis numbers** of **robot axes A1** to **A6** should be entered.
24120	TRAFO_GEOAX_ ASSIGN_TAB_1	The **channel axis numbers** of **geometry axes A1** to **A3** should be entered.
24585	TRAFO5_ORIAX_ ASSIGN_TAB_1	The **channel axis numbers** of **orientation axes A4** to **A6** should be entered.
24130	TRAFO_INCLUDES_ TOOL_1	As the tool must be taken into account within the ROBX transformation, for tool handling when transformation 1 is active, a **value = 1** should be entered.

7.1.2 General machine data of the ROBX transformation

Number: MD	Identifier: $MC_	Parameterization
62903	ROBX_MAIN_AXES	The machine data specifies the basic axes type. For robots, a **value = 3** should be entered.
62904	ROBX_WRIST_AXES	The machine data specifies the wrist axes type. Only the following types are permitted for a 6-axis articulated arm robot: • Central wrist: **Value = 2** • Inclined wrist axis: **Value = 3** The 3 wrist axes must intersect at one point.
62905	ROBX_NUM_AXES	The number of axes incorporated in the transformation must be entered in the machine data. For the 6-axis robot here, a **value = 6** should be entered.
62906	ROBX_A4PAR	The arrangement (perpendicular: 1 or parallel: 0) of the 4th to the 3rd rotary axis should be specified in the machine data. For the robot example shown above, the 4th axis is arranged perpendicular to the 3rd axis, which is why a **value = 0** should be entered.
62901	ROBX_AXES_TYPE	The appropriate axis type should be entered for each of the 6 axes in the machine data. For the robot, for all axes A1 to A6, a **value = 3** (rotary axis) should be entered.
62920	ROBX_AXIS_SEQ	Using the machine data, the sequence of the axes with respect to the external view and the sequence, which is internally defined in the transformation, can be interchanged. The internal sequence for robots is defined so that axes are numbered in an ascending order from the robot base point up to the flange. As a consequence, values = {**1, 2, 3, 4, 5, 6**} should be entered.
62917	ROBX_MAMES	The ROBX transformation specifies the mathematical zero point of the axes (see Adapting the zero points of the axes (Page 36)). However, the mathematical zero point does not always correspond to the mechanical zero point (calibration point) of axes. In order for the zero points to match up with one another, the difference between the mathematical zero point and the mechanical zero point for each axis must be entered in the following machine data: Starting from the mechanical zero point, the deviation should be entered with reference to the mathematically positive direction of rotation of the axis.
62918	ROBX_AXES_DIR	For each axis A1 to A6, the ROBX transformation assumes a certain direction of rotation. If this direction of rotation does not match the direction of rotation of the axis at the machine, then the direction of rotation must be inverted using the machine data. Positive directions of rotation are defined according to the "Right-hand rule". For the 1st rotary axis, the Z-IRO component specifies the positive direction of rotation, for the 2nd and 3rd rotary axes the Y-IRO component specifies the positive direction of rotation (see Adapting the zero points of the axes (Page 36)). For wrist axes A4 to A6, the Z component of the specific wrist axis coordinate system specifies the positive direction of rotation. **No** direction of rotation reversal required: **Value = 1** Direction of rotation reversal required: **Value = -1**

Overview

Number: MD	Identifier: $MC_	Value
62900	ROBX_KINCLASS	1
62903	ROBX_MAIN_AXES	3
62904	ROBX_WRIST_AXES	2 or 3
62905	ROBX_NUM_AXES	6
62906	ROBX_A4PAR	0
62901	ROBX_AXES_TYPE	[3, 3, 3, 3, 3, 3]
62920	ROBX_AXIS_SEQ	[1, 2, 3, 4, 5, 6]
62917	ROBX_MAMES	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
62918	ROBX_AXES_DIR	[1, 1, 1, 1, 1, 1]

See also

Wrist axes: Central wrist axis (Page 31)

Wrist axes: Inclined wrist axis (Page 31)

7.1.3 Geometry parameters: Basic axes (A1-A3)

<table>
<tr><th>Number: MD</th><th>Identifier: $MC_</th><th>Parameterization</th></tr>
<tr><td>62912</td><td>ROBX_TIRORO_POS[0...2]</td><td rowspan="2">Normally, the basic coordinate system (BCS) corresponds to the coordinate system at the base point of the machine (RO).
The ROBX transformation specifies the internal robot coordinate system (IRO), which is located at the point of intersection of the extension of the 1st rotary axis (A1) and the line that runs at the height of axis A2 (see Kinematics type NR: (Page 49)).
The Z-IRO coordinate runs in the direction of the 1st rotary axis.
The mathematically positive direction of rotation of the 2nd rotary axis is specified by the Y-IRO coordinate.
Using the machine data, the IRO coordinate system can be offset and rotated as required with respect to the base point coordinate system (RO). All 6 degrees of freedom are possible.</td></tr>
<tr><td>62913</td><td>ROBX_TIRORO_RPY[0...2]</td></tr>
<tr><td rowspan="2">62907</td><td>ROBX_MAIN_LENGTH_AB[0]</td><td>The horizontal distance between the 2nd rotary axis (A2) and the point of rotation of the 1st rotary axis (A1) can be parameterized using the machine data (see Kinematics type NR: (Page 49)).
Axis A2 must run perpendicular to axis A1.
Only one 1-dimensional offset in the X-IRO direction is possible between the 1st and the 2nd rotary axes.</td></tr>
<tr><td>ROBX_MAIN_LENGTH_AB[1]</td><td>The distance between the 2nd rotary axis (A2) and the 3rd rotary axis (A3) can be parameterized using the machine data (see Kinematics type NR: (Page 49)).
Axis A3 must run parallel to axis A2.
Only one offset in the direction of the X-IRO coordinate is permitted between axes A2 and A3.</td></tr>
</table>

7.1.4 Geometry parameters: Robot wrist (A3-A4)

Number: MD	Identifier: $MC_	Parameterization
62908	ROBX_TX3P3_**POS** **[0...2]**	The position and direction of axis A4 with respect to axis A3 is parameterized using the machine data (see Kinematics type NR: (Page 49)). All 6 degrees of freedom are possible.
62909	ROBX_TX3P3_**RPY** **[0...2]**	

7.1.5 Geometry parameters: Wrist axes (A4-A6)

Number: MD	Identifier: $MC_	Parameterization
62914	ROBX_DHPAR4_5A **[0...1]**	The position of the coordinate systems in the wrist with respect to one another is defined using the machine data. The machine data correspond to Denavit-Hartenberg parameters a, d and α for offsets and rotations of the coordinate system between axes A4→A5 and A5→A6. For a 6-axis articulated arm robot, the machine data should be set as follows for the wrist types: • Wrist axes: Central wrist axis (Page 31) • Wrist axes: Inclined wrist axis (Page 31) Wrists with an offset between the wrist axes are not permitted.
62915	ROBX_DHPAR4_5D **[0..1]**	
62916	ROBX_DHPAR4_ 5ALPHA**[0...1]**	

7.1.6 Geometry parameters: Flange

Number: MD	Identifier: $MC_	Parameterization
62910	ROBX_TFLWP_POS **[0...2]**	The flange coordinate system (FL) with respect to the wrist point coordinate system (WP) can be parameterized using the machine data. For a central wrist or for an inclined wrist axis, the wrist point coordinate system (WP) lies at the point of intersection of the 3 wrist axes. The wrist point coordinate system (WP) corresponds to the coordinate system, which is assigned to the 6th axis. The offsets and rotations of the flange are specified referred to this coordinate system.
62911	ROBX_TFLWP_RPY **[0...2]**	
62965	ROBX_TTCFL_POS **[0...2]**	Using the machine data, an additional offset and rotation of the physical flange coordinate system can be parameterized with respect to the above parameterized flange coordinate system (FL) of the robot. These machine data act the same as a tool carrier.
62966	ROBX_TTCFL_RPY **[0...2]**	

7.1.7 Geometry parameters: Inclined wrist axis

For the articulated arm robot described as example in Chapter Kinematics type NR: (Page 49), an offset can be parameterized in the robot wrist so that the 3 wrist axes do not intersect at one point (see the following diagram). To do this, an inclined wrist axis (Page 33) should be parameterized for the wrist. However, in the case shown, there is no offset in the direction of the X coordinate (parameter a). This is the reason that Denavit-Hartenberg parameter a must be set to a value of 0 for an offset in the direction of the Z coordinate:

- MD62915 $MC_ROBX_DHPAR4_5A[**0**] = 0
- MD62915 $MC_ROBX_DHPAR4_5A[**1**] = 0

All other geometry parameters should be set as described in the previous chapters Machine data of the standard system (Page 50).

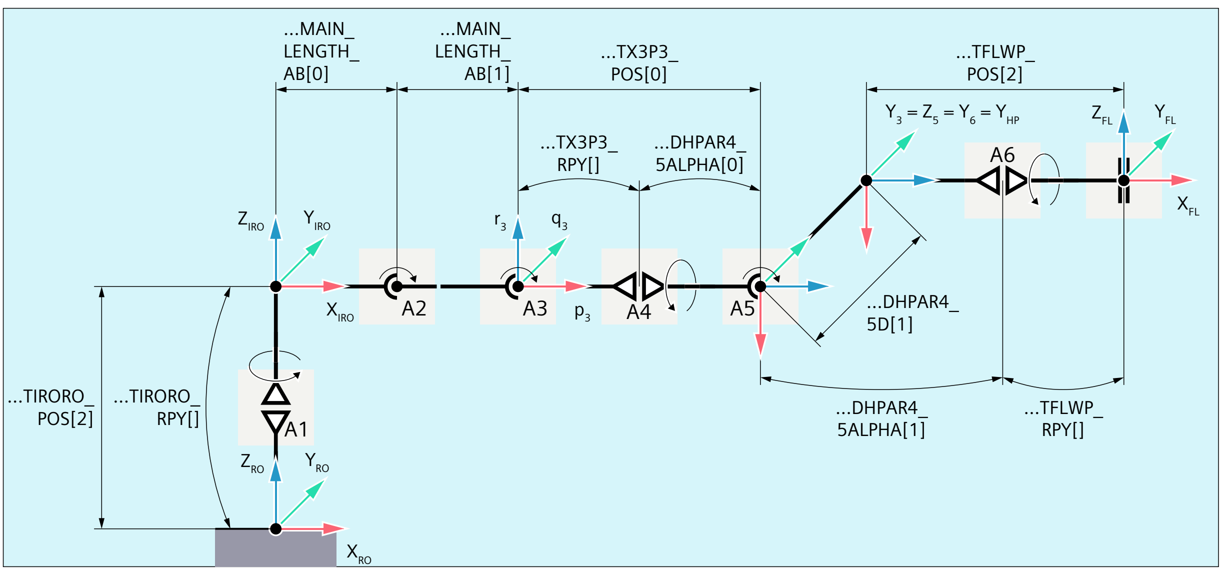


Figure 7-3 NR kinematics with inclined wrist axis

Defining computational overhead

For an inclined wrist axis, the reverse transformation can only be solved using an iterative calculation technique, and no longer analytically. The computational performance released for this purpose is defined using the following machine data. The specified values represent a proven initial basis:

Number: MD	Identifier: $MC_	Parameterization
62967	ROBX_ITER_ MAXCOUNT	Maximum number of iteration runs for a reverse transformation: **Value = 8**
62968	ROBX_ITERPRECISION	Termination barrier for internal target function: **Value = 1.0**$^{-4}$
62968	ROBX_ITERLIMIT	Limit angle to identify the singular position: **Value = 3**
62970	ROBX_ITERLIMIT2	Help angle to retract from the singular position: **Value = 0**

7.2 SC kinematics type

The following chapter describes the parameterization of a 6-axis articulated arm robot, type SC, shown as example in the diagram.

Kinematics

Corresponding to type SC, the robot is based on 2 linear axes and 4 rotary axes. Its arrangement is shown in the following diagram.

1. Axes A1 and A2 are linear axes.
2. Axis A3 is a swiveling axes that is arranged in parallel to one of the linear axes.
3. Axes A4 - A6 are the rotary axes of the robot wrist. These do not have to intersect at one point and they also do not have to be perpendicular to one another in pairs.

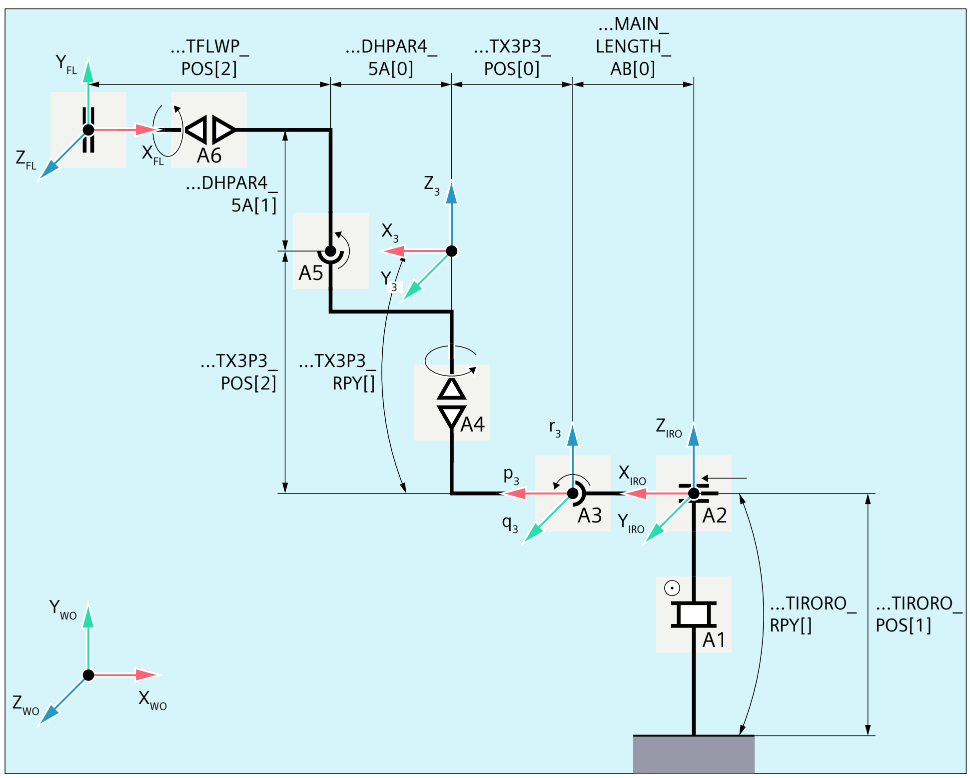


Figure 7-4 SC kinematics

Denavit-Hartenberg parameters

The following diagram shows the Denavit-Hartenberg parameters of wrist axes A4 - A6 in the mathematical initial position.

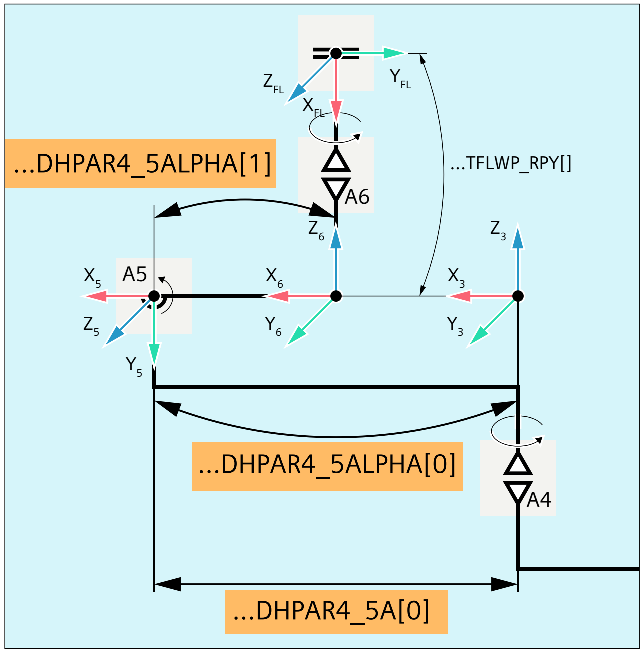


Figure 7-5 Denavit-Hartenberg parameters

7.2.1 Machine data of the standard system

See Chapter: Machine data of the standard system (Page 50)

7.2.2 General machine data of the ROBX transformation

Number: MD	Identifier: $MC_	Parameterization
62903	ROBX_MAIN_AXES	The machine data specifies the basic axes type. For type SC robots, a **value = 4** should be entered.
62904	ROBX_WRIST_AXES	The machine data specifies the wrist axes type. In this case, a distinction is made between the following cases: • The 3 wrist axes intersect at one point. – For a **central wrist**, a **value = 2** should be entered – For an **inclined wrist axis**, a **value = 3** should be entered • The 3 wrist axes do **not** intersect at a point If there is an offset between axes A4→A5 or A5→A6, which means that at least one of the Denavit-Hartenberg parameters a for an offset in the direction of the X coordinate (MD62914 $MC_ROBX_DHPAR4_5A[0...1]) is not equal to 0, then an **inclined wrist axis** is involved: **Value = 6**
62905	ROBX_NUM_AXES	The number of axes incorporated in the transformation must be entered in the machine data. For the 6-axis robot here, a **value = 6** should be entered.

Number: MD	Identifier: $MC_	Parameterization
62906	ROBX_A4PAR	The arrangement (perpendicular: 1 or parallel: 0) of the 4th to the 3rd rotary axis should be specified in the machine data. This is because for this particular robot the 4th axis is in parallel to the 3rd axis, which is why a **value = 0** should be entered.
62901	ROBX_AXES_TYPE	The appropriate axis type should be entered for each of the 6 axes in the machine data. The following values should be set for this type SC robot: • Axes A1 to A2 (index 0...1), a **value = 1** (linear axis) should be entered. • Axes A3 to A6 (index 2...5), a **value = 3** (rotary axis) should be entered.
62920	ROBX_AXIS_SEQ	Using the machine data, the sequence of the axes with respect to the external view and the sequence, which is internally defined in the transformation, can be interchanged. The internal sequence for robots is defined so that axes are numbered in an ascending order from the robot base point up to the flange. As a consequence, values = {**1, 2, 3, 4, 5, 6**} should be entered.
62917	ROBX_MAMES	The ROBX transformation specifies the mathematical zero point of the axes (see Adapting the zero points of the axes (Page 36) for axes A1 to A4 and Fig. 17 for axis A5). However, the mathematical zero point does not always correspond to the mechanical zero point (calibration point) of axes. In order for the zero points to match up with one another, the difference between the mathematical zero point and the mechanical zero point for each axis must be entered in the following machine data: Starting from the mechanical zero point, the deviation should be entered with reference to the mathematically positive direction of rotation of the axis.
62918	ROBX_AXES_DIR	For each axis A1 to A6, the ROBX transformation assumes a certain direction of rotation. If this direction of rotation does not match the direction of rotation of the axis at the machine, then the direction of rotation must be inverted using the machine data. Positive directions of rotation are defined according to the "Right-hand rule". For the 1st rotary axis, the Y-IRO component specifies the positive direction of rotation (see Adapting the zero points of the axes (Page 36)). For wrist axes A4 to A6, the Z component of the specific wrist axis coordinate system specifies the positive direction of rotation. **No** direction of rotation reversal required: **Value = 1** Direction of rotation reversal required: **Value = -1**

Overview

Number: MD	Identifier: $MC_	Value
62900	ROBX_KINCLASS	1
62903	ROBX_MAIN_AXES	4
62904	ROBX_WRIST_AXES	2 or 3 or 6
62905	ROBX_NUM_AXES	6
62906	ROBX_A4PAR	0
62901	ROBX_AXES_TYPE	[1, 1, 3, 3, 3.,3]
62920	ROBX_AXIS_SEQ	[1, 2, 3, 4, 5, 6]
62917	ROBX_MAMES	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
62918	ROBX_AXES_DIR	[1, 1, 1, 1, 1, 1]

See also

Wrist axes: Central wrist axis (Page 31)

Wrist axes: Inclined wrist axis (Page 31)

Wrist axes: Inclined wrist axis (Page 33)

7.2.3 Geometry parameters: Basic axes (A1-A3)

Number: MD	Identifier: $MC_	Parameterization
62912	ROBX_TIRORO_**POS** **[0...2]**	Normally, the basic coordinate system (BCS) corresponds to the coordinate system at the base point of the machine (RO). The transformation defines the internal robot coordinate system (IRO), and for the plane spanning axes A2 and A3, can be freely selected (see SC kinematics type (Page 55)). The Z-IRO coordinate is perpendicular to this plane. The Y-IRO coordinate lies in parallel to the first rotary axis. Using the machine data, the internal robot coordinate system (IRO) can be offset and rotated as required with respect to the base point coordinate system (RO). All 6 degrees of freedom are possible.
62913	ROBX_TIRORO_**RPY** **[0...2]**	
62907	ROBX_MAIN_LENGTH_AB[**0**]	The distance between the 1st rotary axis (A3) and the origin of the internal robot coordinate system (IRO) is defined using the machine data (see SC kinematics type (Page 55)).
	ROBX_MAIN_LENGTH_AB[**1**]	The machine data is **not required** for type SC robots.

7.2.4 Geometry parameters: Robot wrist (A3-A4)

Number: MD	Identifier: $MC_	Parameterization
62908	ROBX_TX3P3_**POS** **[0...2]**	The position and direction of axis A4 with respect to axis A3 is parameterized using the machine data (see SC kinematics type (Page 55)). All 6 degrees of freedom are possible.
62909	ROBX_TX3P3_**RPY** **[0...2]**	

7.2.5 Geometry parameters: Wrist axes (A4-A6)

Number: MD	Identifier: $MC_	Parameterization
62914	ROBX_DHPAR4_5A **[0...1]**	The position of the coordinate systems in the wrist with respect to one another is defined using the machine data (see SC kinematics type (Page 55)). The machine data correspond to Denavit-Hartenberg parameters a, d and α for offsets and rotations of the coordinate system between axes A4→A5 and A5→A6. For type SC robots, the machine data should be set as follows for the wrist types: • Wrist axes: Central wrist axis (Page 31) • Wrist axes: Inclined wrist axis (Page 31) • Wrist axes: Inclined wrist axis (Page 33) **Note**: For an inclined wrist axis, offset in the direction of the Z coordinate is not permitted or Denavit-Hartenberg parameter d must be 0. – MD62915 $MC_ROBX_DHPAR4_**5D[0]** = 0 – MD62915 $MC_ROBX_DHPAR4_**5D[1]** = 0
62915	ROBX_DHPAR4_5D **[0..1]**	
62916	ROBX_DHPAR4_ 5ALPHA**[0...1]**	

7.2.6 Geometry parameters: Flange

Number: MD	Identifier: $MC_	Parameterization
62910	ROBX_TFLWP_POS **[0...2]**	The flange coordinate system (FL) with respect to the wrist point coordinate system (WP) can be parameterized using the machine data. For a central wrist or for an inclined wrist axis, the wrist point coordinate system (WP) lies at the point of intersection of the 3 wrist axes. The wrist point coordinate system (WP) corresponds to the coordinate system, which is assigned to the 6th axis. The offsets and rotations of the flange are specified referred to this coordinate system.
62911	ROBX_TFLWP_RPY **[0...2]**	
62965	ROBX_TTCFL_POS **[0...2]**	Using the machine data, an additional offset and rotation of the physical flange coordinate system can be parameterized with respect to the above parameterized flange coordinate system (FL) of the robot. These machine data act the same as a tool carrier.
62966	ROBX_TTCFL_RPY **[0...2]**	

7.2.7 Computational performance for inclined wrist axis

For an inclined wrist axis, the reverse transformation can only be solved using an iterative calculation technique, and no longer analytically. The computational performance released for this purpose is defined using the following machine data. The specified values represent a proven initial basis:

Number: MD	Identifier: $MC_	Parameterization
62967	ROBX_ITER_ MAXCOUNT	Maximum number of iteration runs for a reverse transformation: **Value = 8**
62968	ROBX_ITERPRECISION	Termination barrier for internal target function: **Value = 1.0**$^{-15}$

Number: MD	Identifier: $MC_	Parameterization
62968	ROBX_ITERLIMIT	Limit angle to identify the singular position: **Value = 3**
62970	ROBX_ITERLIMIT2	Help angle to retract from the singular position: **Value = 0**

7.3 Kinematic type CC

The following chapter describes the parameterization of a 6-axis articulated arm robot, type CC, shown as example in the diagram.

Kinematics

Corresponding to type CC, the robot is based on 1 linear axis and 5 rotary axes, which are arranged as shown in the following diagram:

1. Axis A1 is a linear axis.
2. Axes A2 and A3 are swiveling axes that are parallel to one another, and which are arranged in parallel to linear axis A1.
3. Axes A4 - A6 are the rotary axes of the robot wrist. These do not have to intersect at one point.

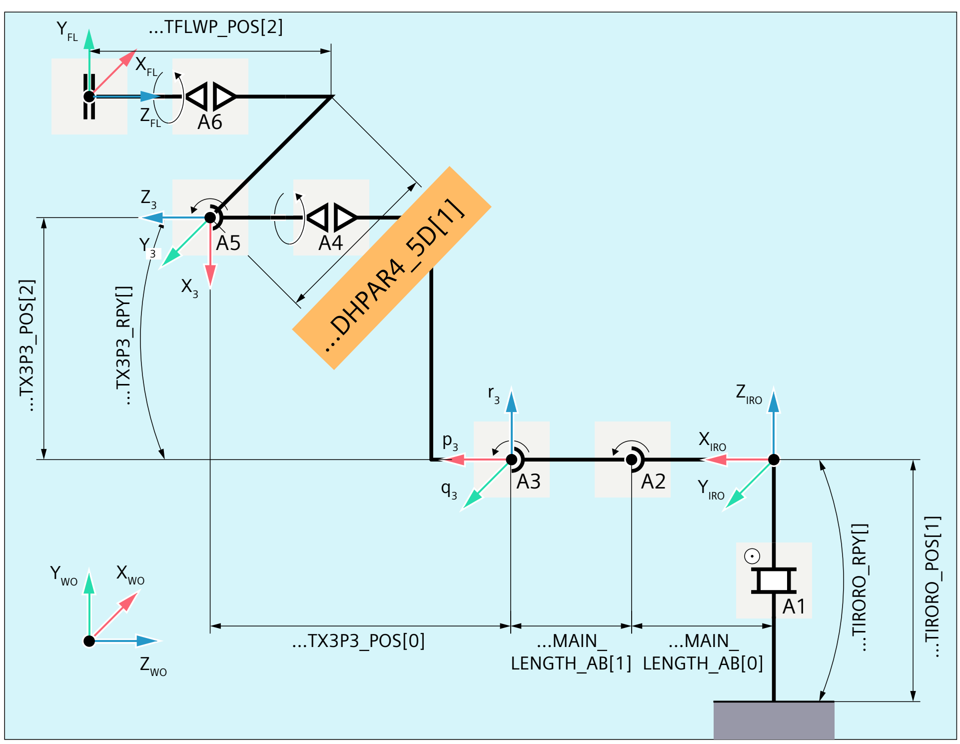


Figure 7-6 CC kinematics

Denavit-Hartenberg parameters for an inclined wrist axis

An inclined wrist axis (type 6) is involved if the axes of the robot wrist do not intersect at one point.

In contrast to a general inclined wrist axis, the Denavit-Hartenberg parameters must be set as follows:

- Parameter a (offset along the X coordinate)
  - A4→A5 = 0
  - A5→A6 = 0
- Parameter α (rotation around the X coordinate)
  - A4→A5 = -90
  - A5→A6 = 90
- Parameter d (offset along the Z coordinate)
  - A4→A5 = 0
  - A5→A6 = offset d (see the diagram above)

7.3.1 Machine data of the standard system

See Chapter: Machine data of the standard system (Page 50)

7.3.2 General machine data of the ROBX transformation

Overview

Number: MD	Identifier: $MC_	Value
62903	ROBX_MAIN_AXES	2
62904	ROBX_WRIST_AXES	2 or 3 or 6
62905	ROBX_NUM_AXES	6
62906	ROBX_A4PAR	0
62901	ROBX_AXES_TYPE	[1, 3, 3, 3, 3.,3]
62920	ROBX_AXIS_SEQ	[1, 2, 3, 4, 5, 6]
62917	ROBX_MAMES	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
62918	ROBX_AXES_DIR	[1, 1, 1, 1, 1, 1]

See also

Adapting the zero points of the axes (Page 36)

7.3.3 Geometry parameters: Basic axes (A1-A3)

Number: MD	Identifier: $MC_	Parameterization
62912	ROBX_TIRORO_**POS** **[0...2]**	Normally, the basic coordinate system (BCS) corresponds to the coordinate system at the base point of the machine (RO). The transformation defines the internal robot coordinate system (IRO), and for the plane spanning axes A2 and A3, can be freely selected (see Kinematic type CC (Page 60)). The Z-IRO coordinate is perpendicular to this plane. The Y-IRO coordinate lies in parallel to the first rotary axis. Using the machine data, the internal robot coordinate system (IRO) can be offset and rotated as required with respect to the base point coordinate system (RO). All 6 degrees of freedom are possible.
62913	ROBX_TIRORO_**RPY** **[0...2]**	
62907	ROBX_MAIN_LENGTH_AB[**0**]	The distance between the 1st rotary axis (A2) and the origin of the internal robot coordinate system (IRO) is defined using the machine data (see Kinematic type CC (Page 60)).
	ROBX_MAIN_LENGTH_AB[**1**]	The distance between the 1st rotary axis (A2) and the 2nd rotary axis (A3) can be defined using the machine data (see Kinematic type CC (Page 60)). Axis A3 must run parallel to axis A2. Only one offset in the X-IRO direction is permitted between axes A2 and A3.

7.3.4 Geometry parameters: Robot wrist (A3-A4)

Number: MD	Identifier: $MC_	Parameterization
62908	ROBX_TX3P3_**POS** **[0...2]**	The position and direction of axis A4 with respect to axis A3 is parameterized using the machine data (see Kinematic type CC (Page 60)). All 6 degrees of freedom are possible.
62909	ROBX_TX3P3_**RPY** **[0...2]**	

7.3.5 Geometry parameters: Wrist axes (A4-A6)

<table>
<tr><th>Number: MD</th><th>Identifier: $MC_</th><th>Parameterization</th></tr>
<tr><td>62914</td><td>ROBX_DHPAR4_5A
[0...1]</td><td rowspan="3">The position of the coordinate systems in the wrist with respect to one another is defined using the machine data (see Kinematic type CC (Page 60)). The machine data correspond to Denavit-Hartenberg parameters a, d and α for offsets and rotations of the coordinate system between axes A4→A5 and A5→A6.
For type CC robots, the machine data should be set as follows for the wrist types:
• Wrist axes: Central wrist axis (Page 31)
• Wrist axes: Inclined wrist axis (Page 31)
• Wrist axes: Inclined wrist axis (Page 33)
Note: If, for an inclined wrist axis, there is an offset between axes A5 and A6 in the direction of the Z coordinate (parameter d), then the Denavit-Hartenberg parameters should be set as follows:
– MD62914 $MC_ROBX_DHPAR4_5A[0] = 0
– MD62914 $MC_ROBX_DHPAR4_5A[1] = 0
– MD62915 $MC_ROBX_DHPAR4_5D[0] = 0
– MD62915 $MC_ROBX_DHPAR4_5D[1] = <offset A5→A6>
– MD62916 $MC_ROBX_DHPAR4_5ALPHA[0] = -90
– MD62916 $MC_ROBX_DHPAR4_5ALPHA[1] = 90</td></tr>
<tr><td>62915</td><td>ROBX_DHPAR4_5D
[0..1]</td></tr>
<tr><td>62916</td><td>ROBX_DHPAR4_
5ALPHA[0...1]</td></tr>
</table>

7.3.6 Geometry parameters: Flange

<table>
<tr><th>Number: MD</th><th>Identifier: $MC_</th><th>Parameterization</th></tr>
<tr><td>62910</td><td>ROBX_TFLWP_POS
[0...2]</td><td rowspan="2">The flange coordinate system (FL) with respect to the wrist point coordinate system (WP) can be parameterized using the machine data.
For a central wrist or for an inclined wrist axis, the wrist point coordinate system (WP) lies at the point of intersection of the 3 wrist axes.
The wrist point coordinate system (WP) corresponds to the coordinate system, which is assigned to the 6th axis. The offsets and rotations of the flange are specified referred to this coordinate system.</td></tr>
<tr><td>62911</td><td>ROBX_TFLWP_RPY
[0...2]</td></tr>
<tr><td>62965</td><td>ROBX_TTCFL_POS
[0...2]</td><td rowspan="2">Using the machine data, an additional offset and rotation of the physical flange coordinate system can be parameterized with respect to the above parameterized flange coordinate system (FL) of the robot. These machine data act the same as a tool carrier.</td></tr>
<tr><td>62966</td><td>ROBX_TTCFL_RPY
[0...2]</td></tr>
</table>

7.3.7 Computational performance for inclined wrist axis

For an inclined wrist axis, the reverse transformation can only be solved using an iterative calculation technique, and no longer analytically. The computational performance released for this purpose is defined using the following machine data. The specified values represent a proven initial basis:

Number: MD	Identifier: $MC_	Parameterization
62967	ROBX_ITER_ MAXCOUNT	Maximum number of iteration runs for a reverse transformation: **Value = 8**
62968	ROBX_ITERPRECISION	Termination barrier for internal target function: **Value = 1.0**$^{-6}$
62968	ROBX_ITERLIMIT	Limit angle to identify the singular position: **Value = 3**
62970	ROBX_ITERLIMIT2	Help angle to retract from the singular position: **Value = 0**

Additional axes 8

8.1 Additional axes in the basic coordinate system (BCS)

8.1.1 Function description

Using up to 3 linear additional axes, the complete machine can be traversed into the basic coordinate position (BCS). The basic coordinate system (BCS) is identical to the world coordinate system (WO). The traversing directions of the additional axes should be specified referred to the world coordinate system (WO).

The following diagram shows as example a robot with 6 rotary axes A1 to A6 and a linear axis EX1 as an additional axis. Additional axis EX1 moves the complete machine in the direction of the X coordinate of the basic (BCS) or world coordinate system (WO).

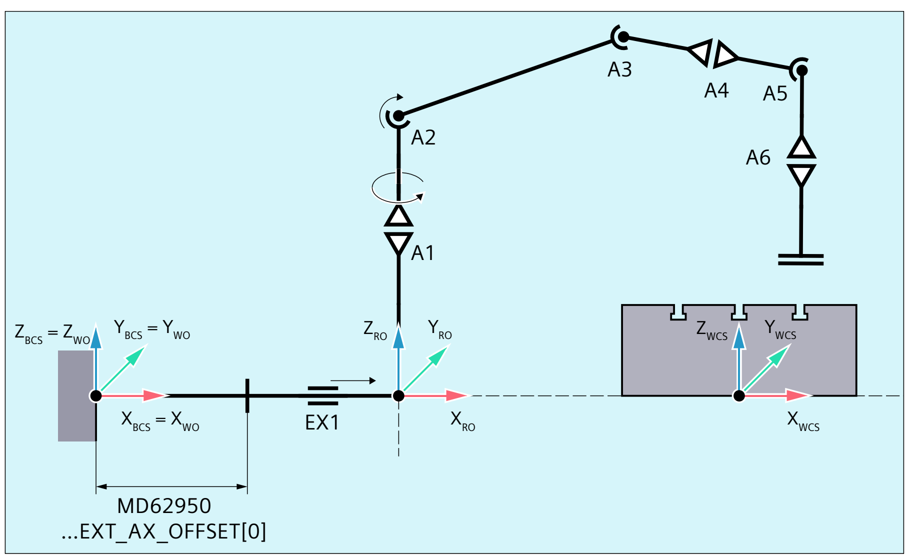


Figure 8-1 Kinematics, additional axes in the BCS

Theoretically, redundancy at the machine is obtained as a result of the additional axes. However, the redundancy is avoided as the additional axes must be explicitly programmed as additional input axes of the ROBX transformation in addition to Cartesian position and orientation of the robot.

8.1.2 Commissioning

8.1.2.1 Machine data of the standard system

The linear additional axes are parameterized as input axes of the ROBX transformation in MD24110 $MC_TRAFO_AXES_IN_1 in indices 6 to 8.

See Chapter Axis assignment of the transformation (Page 17).

8.1.2.2 Machine data of the ROBX transformation

Direction vectors of the linear additional axes

The direction vectors for the linear additional axes are parameterized in the basic coordinate system (BCS), which corresponds to the world coordinate system (WO), using the machine data:

- MD62937 $MC_ROBX_EXT_AXIS_VECTOR_1[<index>] = <value>

<index>	Meaning: Direction vector 1st linear additional axis (BCS)
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

- MD62938 $MC_ROBX_EXT_AXIS_VECTOR_2[<index>] = <value>

<index>	Meaning: Direction vector 2nd linear additional axis (BCS)
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

- MD62939 $MC_ROBX_EXT_AXIS_VECTOR_3[<index>] = <value>

<index>	Meaning: Direction vector 3rd linear additional axis (BCS)
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

Work offset of the linear additional axes

The work offsets of the linear additional axes are parameterized in the basic coordinate system (BCS), which corresponds to the world coordinate system (WO), using the machine data, (see Function description (Page 65)). Starting from the mechanical zero point, the deviation should be entered specified to the mathematically positive traversing direction of the axis.

MD62950 ROBX_EXT_AX_OFFSET[<index>] = <value>

<index>	Meaning: Work offsets in the basic coordinate system (BCS)
0	Work offset of the 1st linear additional axis
1	Work offset of the 2nd linear additional axis
2	Work offset of the 3rd linear additional axis

8.1.3 Programming

Programming

When programming the traversing motion of the robot, the positions of the linear additional axes must be specified in addition to the Cartesian position of the robot.

In this case, a distinction should be made between the following cases:

Programming	Traversing motion
The robot position is programmed using the position and orientation of the tooltip	Only the robot axes traverse/move. The position of the base point remains unchanged.
Only linear additional axes are programmed	Only the linear additional axes traverse. The position of the base point changes. The tooltip (TCP) remains stationary. To achieve this, the robot axes must compensate for the changing position of the base point.
The robot position is programmed using the position and orientation of the tooltip and the linear additional axes.	The traversing motion is split up between the linear additional axes and the robot axes. The positions of the robot axes and also the base point change.

Example

Robot axes X, Y, Z (linear axes) and A, B, C (rotary axes) as well as the linear additional axis Z10 are programmed in block N100:

```
N100 G1 X100 Y20 Z1000 A0 B60 C90 Z_EXT=800 F10000
```

The traversing motion of the machine in the direction of the Z coordinate is split up as follows:

1. The robot base point is traversed to position Z = 800 mm using linear additional axis Z_EXT.
2. The robot itself traverses through the difference from Z1000 - Z_EXT = 1000 - 800 = 200 mm to the programmed end point Z = 1000 mm.

8.2 Additional axes in the workpiece coordinate system (WCS)

8.2.1 Function description

In addition to the robot, the workpiece can be traversed in the workpiece coordinate system (WCS) using up to 3 rotary additional axes.

The following diagram shows as example a robot with 6 rotary axes A1 to A6 and 2 rotary axes E1 and E2 as additional axes. The additional axes move a rotary/swivel table on which the workpiece is located.

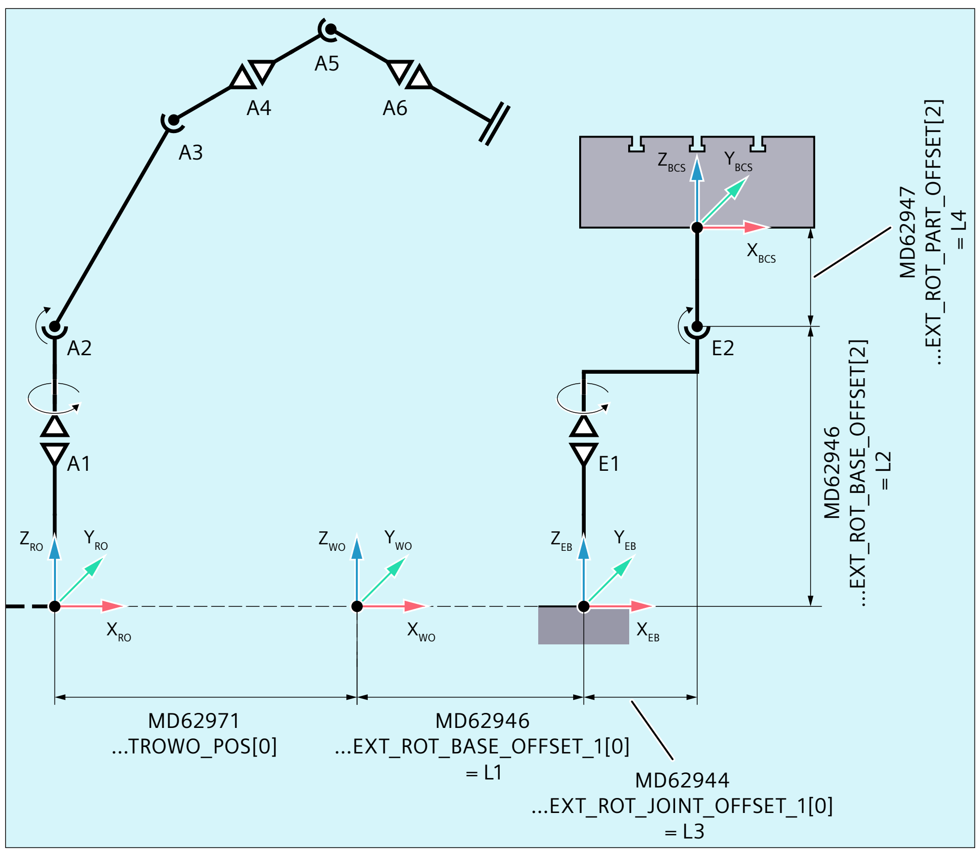


Figure 8-2 Kinematics, additional axes in the WCS

8.2.2 Commissioning

8.2.2.1 Machine data of the standard system

The rotary additional axes are parameterized as input axes of the ROBX transformation in MD24110 $MC_TRAFO_AXES_IN_1 in indices 9 to 11.

See Chapter Axis assignment of the transformation (Page 17).

8.2.2.2 Machine data of the ROBX transformation

Direction vectors of rotary additional axes

The direction vectors for the rotary additional axes, which rotate the workpiece, are parameterized in the basic coordinate system (BCS), which corresponds to the world coordinate system (WO), using the machine data. The positive direction of rotation of the additional axes is obtained according to the "Right-hand rule".

- MD62940 $MC_ROBX_EXT_ROT_AX_VECTOR_1[<index>] = <value>

<index>	Meaning: Direction vector, 1st rotary additional axis (WCS)
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

- MD62941 $MC_ROBX_EXT_ROT_AX_VECTOR_2[<index>] = <value>

<index>	Meaning: Direction vector, 2nd rotary additional axis (WCS)
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

- MD62942 $MC_ROBX_EXT_ROT_AX_VECTOR_3[<index>] = <value>

<index>	Meaning: Direction vector, 3rd rotary additional axis (WCS)
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

Angle offset

Using the machine data, an angle offset for the rotary axes can be parameterized in the workpiece coordinate system.

Starting from the mechanical zero point, the angle offset must be entered referred to the mathematical positive direction of rotation of the axis. The mathematical positive direction of rotation of the axis is obtained according to the "Right-hand rule" as rotation around the corresponding direction vector of the axis.

MD62943 $MC_ROBX_EXT_ROT_AX_OFFSET[<index>] = <value>

<index>	Meaning:
0	Angle offset of the 1st rotary additional axis
1	Angle offset of the 2nd rotary additional axis
2	Angle offset of the 3rd rotary additional axis

Offset between the rotary additional axes

Using the machine data, an offset can be parameterized between the rotary additional axes in the workpiece coordinate system (WCS). The offset must be specified in the mathematical zero position of the rotary axes referred to the world coordinate system (WO).

Offset between the 1st and 2nd rotary additional axes

MD62944 $MC_ROBX_EXT_ROT_JOINT_OFFSET_1[<index>] = <value>

<index>	Meaning:
0	X coordinate of the offset vector
1	Y coordinate of the offset vector
2	Z coordinate of the offset vector

Offset between the 2nd and 3rd rotary additional axes

MD62945 $MC_ROBX_EXT_ROT_JOINT_OFFSET_2[<index>] = <value>

<index>	Meaning:
0	X coordinate of the offset vector
1	Y coordinate of the offset vector
2	Z coordinate of the offset vector

Offset between the base point and the 1st rotary additional axes

The offset between the base point of the robot and the base point of the 1st rotary additional axis, which rotates the workpiece, is parameterized using the machine data. The offset must be specified referred to the directions of the world coordinate system (WO).

MD62946 $MC_ROBX_EXT_ROT_BASE_OFFSET[<index>] = <value>

<index>	Meaning:
0	X coordinate of the offset vector
1	Y coordinate of the offset vector
2	Z coordinate of the offset vector

Offset between the last rotary additional axis and the basic coordinate system (BCS)

The offset between the last rotary additional axis, which rotates the workpiece, and the basic coordinate system (BCS) is parameterized using the machine data. The offset must be specified referred to the directions of the world coordinate system (WO).

MD62947 $MC_ROBX_EXT_ROT_PART_OFFSET[<index>] = <value>

<index>	Meaning:
0	X coordinate of the offset vector
1	Y coordinate of the offset vector
2	Z coordinate of the offset vector

8.2.3 Example

For the robot example from Chapter Function description (Page 68), the following values should be set for the machine data listed in the table:

Number: MD	Identifier: $MC_	Value
62940	ROBX_EXT_ROT_AX_VECTOR_1	[0.0, 0.0, 1.0]
62941	ROBX_EXT_ROT_AX_VECTOR_2	[0.0, 1.0, 0.0]
62944	ROBX_EXT_ROT_JOINT_OFFSET_1	[L3, 0.0, 0.0]
62946	ROBX_EXT_ROT_BASE_OFFSET	[L1, 0.0, L2]
62947	ROBX_EXT_ROT_PART_OFFSET	[0.0, 0.0, L4]
62943	ROBX_EXT_ROT_AX_OFFSET	[0.0, 0.0, 0.0]

8.3 Additional axes in the flange coordinate system

8.3.1 Function description

In the flange coordinate system (FL), the tool can be rotated using up to 3 rotary additional axes.

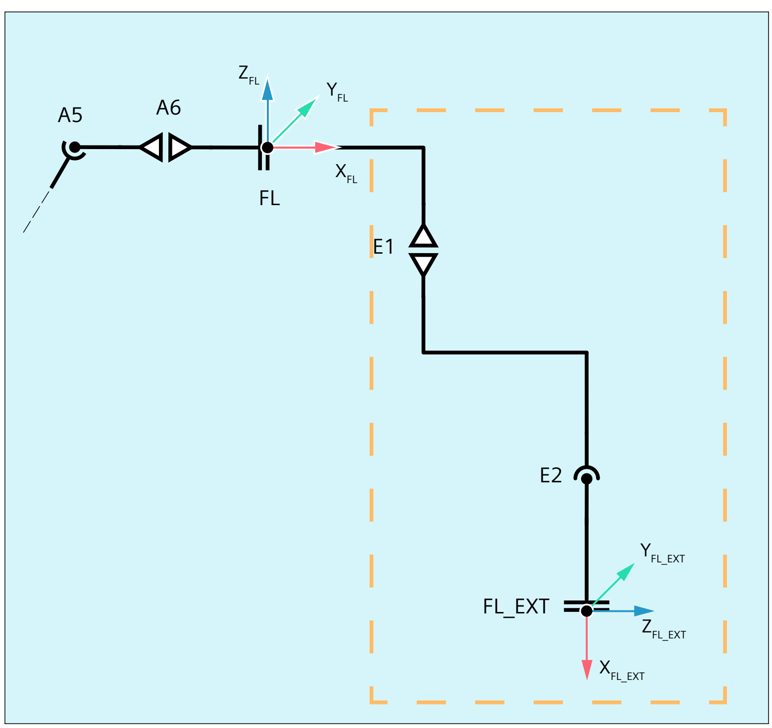


Figure 8-3 Principle of additional axes in the flange coordinate system

Direction specification

The directions of these additional axes are specified in relation to the flange coordinate system.

Avoiding redundancy

A certain redundancy is obtained at the machine as a result of these additional axes in the flange coordinate system. However, the redundancy is avoided as these additional axes must be programmed as additional input axes of the transformation in addition to Cartesian position and orientation.

Constraint

The function of rotary additional axes in the flange coordinate system (FL) is only permitted in conjunction with **6-axis** robots.

If additional axes are parameterized in the flange coordinate system for robots equipped with less than 6 axes, when the transformation is activated, then alarm 75373 "ROBX: Additional axes not permitted in the flange coordinate system".

8.3.2 Commissioning

8.3.2.1 Machine data of the standard system

The rotary additional axes in the flange coordinate system are parameterized as input axes of the ROBX transformation in MD24110 $MC_TRAFO_AXES_IN_1 in indices 12 to 14.

See Chapter Axis assignment of the transformation (Page 17).

8.3.2.2 Machine data of the ROBX transformation

Direction vectors of rotary additional axes

The direction vectors of the rotary additional axes are parameterized in the flange coordinate system (FL) using the machine data:

- MD62973 $MC_ROBX_EXT_FL_ROT_AX_VECTOR_1[<index>] = <value>

<index>	Meaning: 1st rotary additional axis (FL)
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

- MD62974 $MC_ROBX_EXT_FL_ROT_AX_VECTOR_2[<index>] = <value>

<index>	Meaning: 2nd rotary additional axis (FL)
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

- MD62975 $MC_ROBX_EXT_FL_ROT_AX_VECTOR_3[<index>] = <value>

<index>	Meaning: 3rd rotary additional axis (FL)
0	X coordinate of the direction vector
1	Y coordinate of the direction vector
2	Z coordinate of the direction vector

The positive direction of rotation of the rotary additional axes is obtained according to the "Right-hand rule".

Angle offset of rotary additional axes

The angle offsets for the rotary additional axes are parameterized in the flange coordinate system (FL) using the machine data:

Starting from the mechanical zero point, an angle offset must be entered referred to the mathematical positive direction of rotation of the rotary axis. The mathematical positive direction of rotation of the rotary axis is obtained according to the "Right-hand rule" as rotation around the corresponding direction vector of the rotary axis.

MD62976 $MC_ROBX_EXT_FL_ROT_AX_OFFSET[<index>] = <value>

<index>	Meaning: Angle offset
0	Angle offset of the 1st rotary additional axis
1	Angle offset of the 2nd rotary additional axis
2	Angle offset of the 3rd rotary additional axis

Offsets between the rotary additional axes

The offset vectors between the rotary additional axes are parameterized in the flange coordinate system (FL) using the machine data:

The offset vectors must be specified in the mathematical zero position of the rotary axes referred to the flange coordinate system (FL).

- MD62977 $MC_ROBX_EXT_FL_ROT_JOINT_OFF_1[<index>] = <value>

<index>	Meaning: Offset vector, 1st → 2nd additional axis
0	X coordinate of the offset vector
1	Y coordinate of the offset vector
2	Z coordinate of the offset vector

- MD62978 $MC_ROBX_EXT_FL_ROT_JOINT_OFF_2[<index>] = <value>

<index>	Meaning: Offset vector, 2nd → 3rd additional axis
0	X coordinate of the offset vector
1	Y coordinate of the offset vector
2	Z coordinate of the offset vector

Base offset:

The offset factor of the base point of the 1st rotary additional axis referred to the flange coordinate system (FL) is parameterized using the machine data.

MD62979 $MC_ROBX_EXT_FL_ROT_BASE_OFFSET[<index>] = <value>

<index>	Meaning: Base point offset vector of the 1st rotary additional axis
0	X coordinate of the offset vector
1	Y coordinate of the offset vector
2	Z coordinate of the offset vector

8.3.3 Example

In the following example, for a 6-axis robot, starting from the original flange (FL), two additional axes are attached, rotary additional axes E1 and E2.

The direction vectors of the additional axes, referred to the flange coordinate system (FL), are:

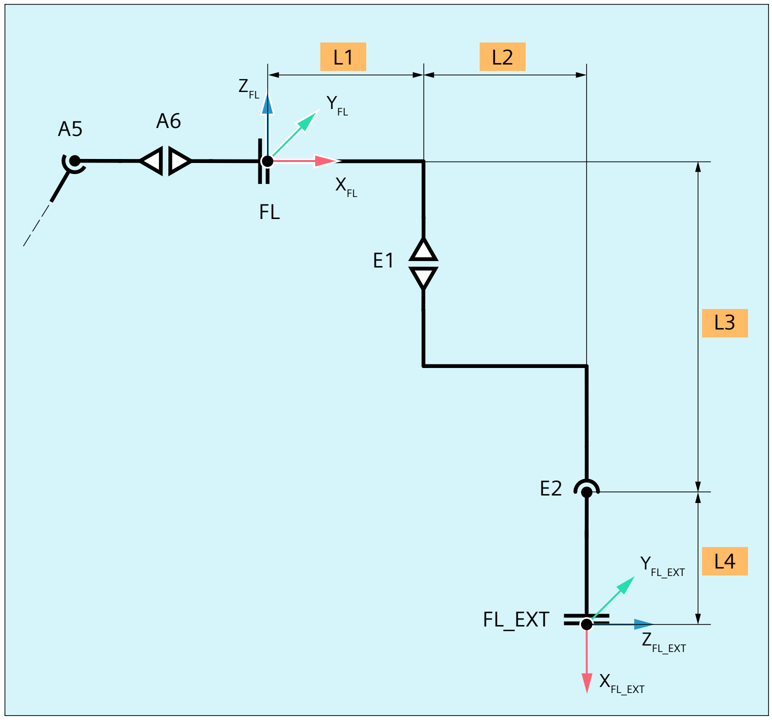
- Additional axis E1: in the direction of coordinate axis Z_{FL} = [0.0, 0.0, **1.0**]
- Additional axis E2: in the direction of coordinate axis Y_{FL} = [0.0, **1.0**, 0.0]

Offsets L1 - L4 of the additional axes can be taken from the following diagram.

The additional axes have no offset with respect to their mathematical basic position: [0.0, 0.0, 0.0]

The physical flange coordinate system (FL_EXT) is rotated through 90° around the Y coordinate axis of the original flange coordinate system (FL): [0.0, **90.0**, 0.0]

The tool is attached to flange FL_EXT.



L1 MD62979, $MC_ROBX_EXT_FL_ROT_BASE_OFFSET[0]

L2 MD62977, $MC_ROBX_EXT_FL_ROT_JOINT_OFF_1[0]

L3 MD62977, $MC_ROBX_EXT_FL_ROT_JOINT_OFF_1[2]

L4 MD62965, $MC_ROBX_TTCFL_POS[2]

Figure 8-4 Example of additional axes in the flange coordinate system

Overview

Number: MD	Identifier: $MC_ROBX_	Value
62973	EXT_FL_ROT_AX_VECTOR_1	[0.0, 0.0, **1.0**]
62974	EXT_FL_ROT_AX_VECTOR_2	[0.0, **1.0**, 0.0]
62977	EXT_FL_ROT_JOINT_OFF_1	[**L2**, 0.0, **-L3**]
62979	EXT_FL_ROT_BASE_OFFSET	[**L1**, 0.0, 0.0]
62976	EXT_FL_ROT_AX_OFFSET	[0.0, 0.0, 0.0]
62965	TTCFL_POS	[0.0, 0.0, **-L4**]
62966	TTCFL_RPY	[0.0, **90.0**, 0.0]

9 Gripper-referred motion control

9.1 Function description

9.1.1 Gripper-referred motion control

Function "Gripper-referred motion control" allows a workpiece, which is held at the robot flange using a gripper, to be machined using a fixed tool. Programming is the same as for a fixed workpiece.

9.1.2 Coordinate systems

For function "Gripper-referred motion control", the basic coordinate system (BCS) or the world coordinate system (WO) and the flange coordinate system (FL) are essentially interchanged regarding the programming. Instead of defining the robot base point, the fixed **tool**flange is defined using the machine data to define the world coordinate system (WO). See Chapter ROBX-specific frame: World coordinate system for the machine base point (Page 20).

Several fixed tools

If the workpiece is machined using several fixed tools, then the robot base point can be parameterized using the machine data to define the ROBX-specific frame for the first internal coordinate system. The position of the fixed tool adapters are then defined using orientable tool carriers, tool adapters or directly by specifying the tool length data referred to the original basic coordinate system (BCS).

Workpiece coordinate system (WCS)

The workpiece coordinate system (WCS) is defined, referred to the flange coordinate system (FL). A programmable frame acts, referred to the robot flange coordinate system.

Tool lengths

Depending on machine data MD62949 $MC_ROBX_TOOL_DIR, also for function "Gripper-referred motion control" the tool lengths are taken into account negated.

Example: Coordinate systems for a 6-axis robot

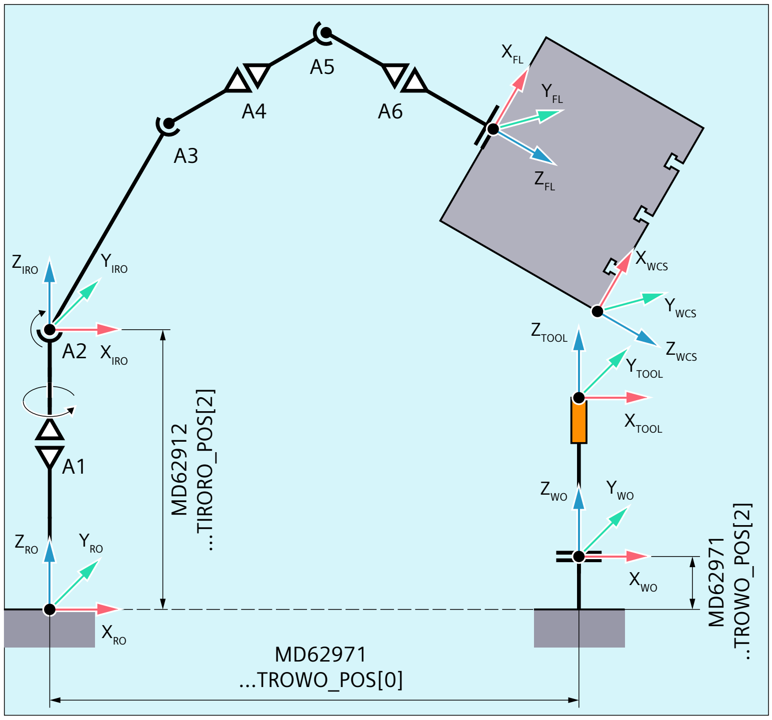


Figure 9-1 Kinematics, gripper-referred motion control

9.1.3 Additional axes

Function "Gripper-referred motion control" can also be activated, if Additional axes in the basic coordinate system (BCS) (Page 65) are available.

The additional axes are offset taking into account in the basic coordinate system (BCS) referred to the first internal coordinate system (IRO) of the robot.

9.1.4 Status display

When transformation is **active**, the current status of function "Gripper-referred motion control" can be read via a system variable, type $A_DBD.

Defining system variables

The system variable to read status information is defined using machine data:

MD62952 $MC_ROBX_A_DBD_START_INDEX = <**index**>

The parameterized index must be a multiple of 4: 0, 4, 8, ...

Reading status information

The status read via system variable $A_DBD[<**index**>] contains the following information:

<Status> = $A_DBD[<**index**>]

<status>	Value	Meaning
Bit 0	0	Function "Gripper-referred motion control" is **not** active.
	1	Function "Gripper-referred motion control" is active.

Note

Validity of the status information

It only makes sense to read the status information when transformation is active. After the transformation was deactivated via `TRAFOOF` or for a channel reset, the system variable only supplies the last valid status for an active transformation.

9.2 Programming

9.2.1 Activation

Function

Function "Gripper-referred motion control" is activated using procedure `CC_GRIP_REL_MOVE_ON( )`.

Syntax

```
CC_GRIP_REL_MOVE_ON( )
```

Meaning

Activating function "Gripper-referred motion control"

Constraints

- The function is only internally activated after first deselecting and reselecting the transformation via `TRAORI` or by assigning a new tool.
- It is only permissible to activate the function for 6-axis robots. If the function is activated for robots equipped with less than 6 axes, then alarm 75321 is displayed.

9.2.2 Deactivating

Function

Function "Gripper-referred motion control" is deactivated using procedure `CC_GRIP_REL_MOVE_OFF( )`.

Syntax

```
CC_GRIP_REL_MOVE_OFF( )
```

Meaning

Deactivating function "Gripper-referred motion control"

Constraints

The function is only internally deactivated after first deselecting and reselecting the transformation via `TRAORI` or by assigning a new tool.

9.3 Commissioning

9.3.1 RESET behavior

Using the machine data, the RESET behavior of function "Gripper-referred motion control" is parameterized:

MD62951 ROBX_GRIP_REL_RESET_MODE = <value>

<value>	Meaning
0	The current status is kept
1	"Gripper-referred motion control" function: **Off** or fixed work**piece** is selected
2	"Gripper-referred motion control" function: **On** or fixed **tool** is selected

Power on reset

After a power on reset, even if <value> is set = 0, function "Gripper-referred motion control" is always **deactivated** or a fixed **workpiece** is selected.

10 Singularity handling for 6-axis kinematics

10.1 Function description

10.1.1 Target group

The target group for function "Singularity handling" are the following persons, who operate robots with 6-axis kinematics in conjunction with the ROBX transformation:

- System integrator
- Machine tool OEM
- End user

Application advantage

Function "Singularity handling" offers users the advantage that it allows program code to be written more simply and faster as the SINUMERIK control system automatically detects the robot-specific singular positions and circumvents them. Previously, this was the task of the machine operator and/or programmer.

Notes

The function does not replace the preliminary simulation of a programmed machining operation in a robot-CAM system, e.g. "NX CAM Robotics", in order to already identify possible singularities before machining. If possible, these can then be avoided using the appropriate programming.

Also with the function active, it is not recommended that you move through singular positions when machining a workpiece, as path deviations and thus contour violations can occur.

10.1.2 System requirements

- The function can run on the following HW systems:
  - NCU720.3
  - NCU730.3
  - NCU1750
  - NCU 1760
- SW release SINUMERIK >= 4.93
- SW version CC ROBX/ROBX_AR >= 6.1
- Active "Advanced Surface" function

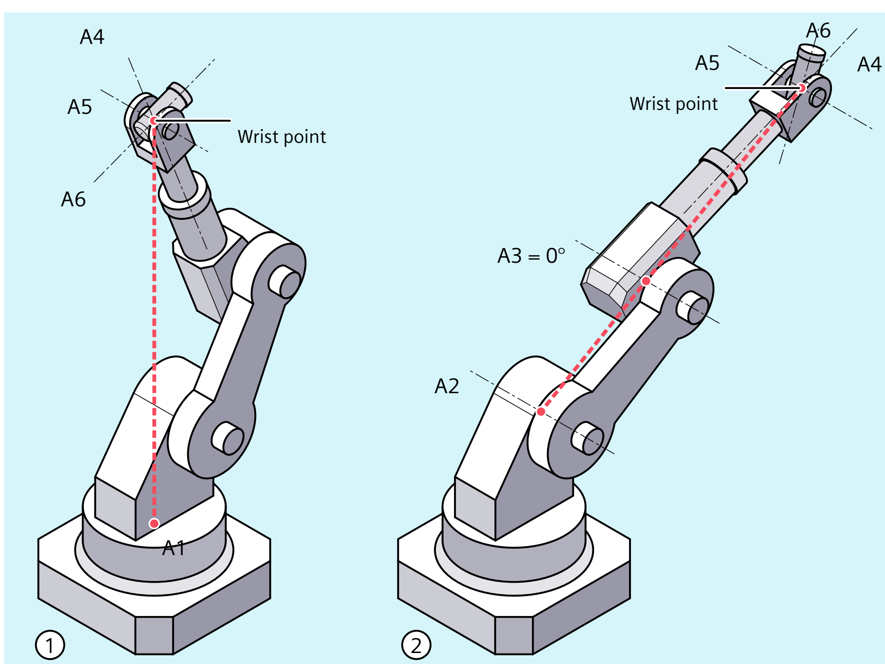
- The function is limited to the following number of axes:
  - 6 robot axes
  - 3 additional axes
- Depending on the performance, the function can be activated in just one channel.

10.1.3 Singularity positions

Singularity positions occur for robots with at least 6 robot axes. These are characterized by at least one of the following phenomena:

- The positions of the robot axes cannot be unambiguously determined from the position and orientation of the tooltip (TCP).
- The programmed path of the tooltip (TCP) in the working space of the robot results in a position of the robot axes between the start and end points so that for one of the axes involved, and axis velocity close to infinity would be necessary in order to achieve a constant path velocity.
- The position of the axes reduces the number of possible degrees of freedom so that from this position, motion to the programmed path or tooltip (TCP) is no longer possible and path and/or contour deviations occur.

Singularity position: alpha1 and alpha2 singularity



① alpha1 singularity: Wrist point and axis 1 are in alignment

② alpha2 singularity: Axis A3 = 0°

Figure 10-1 alpha1 and alpha2 singularity

Singularity position: alpha5 singularity

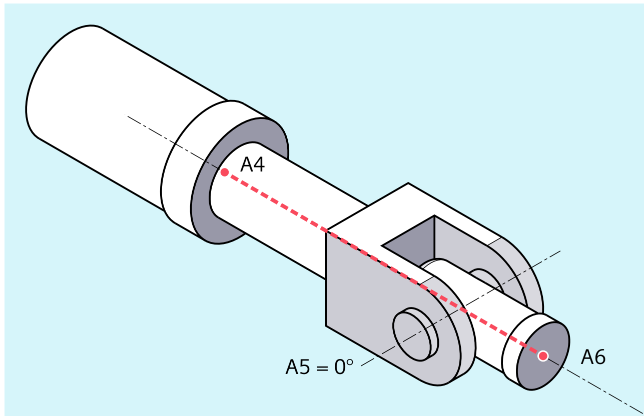


Figure 10-2 alpha5 singularity: Axis A5 = 0°

Gimbal lock

The Gimbal-lock effect is a singularity position that occurs when using Euler angles if angle B is 0°. As a result, the combination of the 3 orientation angles A, B and C loses a possible degree of freedom.

10.1.4 Behavior in the vicinity of a singularity

10.1.4.1 JOG mode

If an impermissible velocity increase occurs when traversing the robot in the JOG mode as a result of a singularity position, then alarm 75324 "ROBX: Excessive velocity in the vicinity of the pole" displayed and traversing motion stopped.

Since triggering alarm 75324 would slow down the robot axes and any additional axes on different braking ramps, alarm 75326 "ROBX: Block %2, emergency stop" is also output. This results in axis-specific braking of all axes over the respective axis-specific parameterized time for the maximum duration of the braking ramp in the event of errors (MD36610, $MA_AX_EMERGENCY_STOP_TIME).

10.1.4.2 AUTOMATIC / MDI mode

Response without function "Singularity handling"

Path in the vicinity of a singularity

If the programmed path in the automatic or MDI mode runs in the vicinity of a singularity, it can occur that one or more of the axes traverse at a very high velocity. Alarm 10910 "Irregular velocity run in a path axis" is displayed, and the path velocity is then reduced to a value, so that the maximum axis velocity is not exceeded.

Path directly through a singularity

If the programmed path runs directly through a singularity, alarm 10911 "Transformation does not permit travel through the pole" is displayed and machining is canceled.

Response with "Singularity handling" function

Detecting the singularity

Function "Singularity handling" checks the program path or the traversing block for singular positions. The value parameterized in setting data SD41670 $SN_SINGULARITY_THRESHOLD (Page 85) is used as threshold value to detect a singularity. The section of the path in which the singularity occurs is localized based on the threshold value.

Handling a singularity

Depending on the position of the singularity within the block, the block, in which the singularity occurs, is split up into 2 or 3 subblocks. Where a path section contains a singularity. See the following diagram.

The traversing block with the path section, which contains the singularity, is traversed using interpolation type "point-to-point" (`PTP`) and continuous-path mode (`G64`). The other path sections are traversed using interpolation type "Path motion" (`CP`) and the programmed G function (e.g. `G645` or `G60`).

When function "Single block" is active, the intermediate points inserted by "Singularity handling" are taken into account. An exact stop is also realized at the block end of intermediate blocks that have been inserted.

Typical block distribution

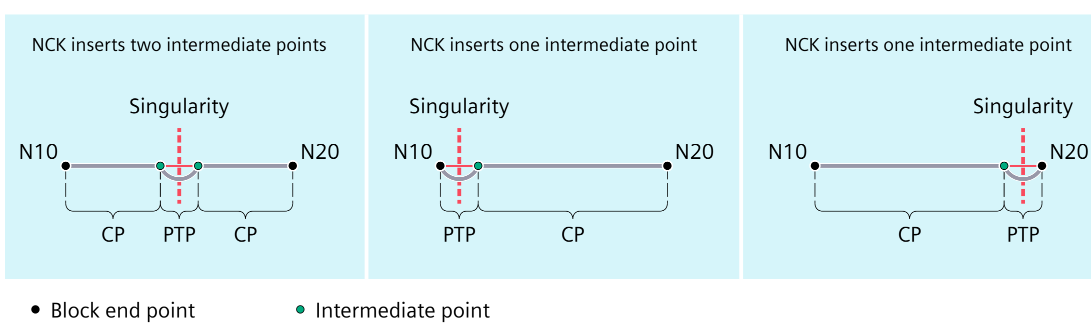


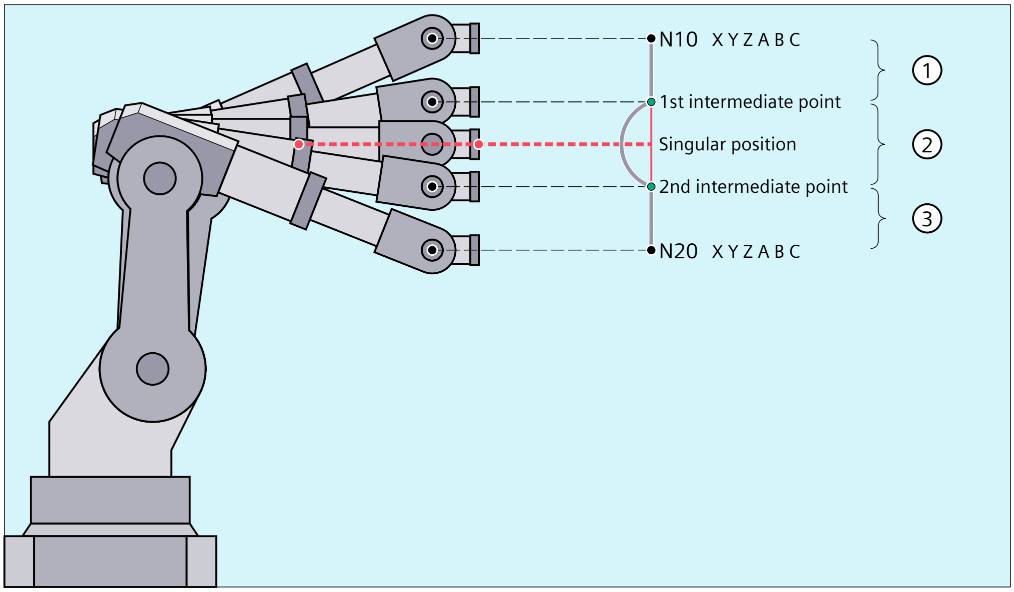
Figure 10-3 Typical block distribution

10.1.4.3 Example

The function is explained in the following diagram using the example of traversing through the alpha-5 singularity (axis RA5=0°) of a 6-axis robot in traversing block N10.

Based on threshold value (SD41670 $SN_SINGULARITY_THRESHOLD) a singularity is detected in block N10. In this path section, 2 intermediate points are inserted so that 3 intermediate blocks N10_1, N10_2 and N10_3 are created. Intermediate block N10_2, which contains the singularity, is traversed using interpolation type point-to-point (`PTP`) and continuous-path mode (`G64`).

Irrespective of whether at the start of the block, end of the block or somewhere in between, the singularity is always detected and is circumvented with the inserted "PTP with G64 block".



① 1st intermediate block: `N10_1 CP G6x`
② 2nd intermediate block: `N10_2 PTP G64`
③ 3rd intermediate block: `N10_3 CP G6x`

Figure 10-4 Singularity and block split

10.2 Commissioning

10.2.1 Activation

Using the machine data, function "Singularity handling" is activated for the ROBX transformation:

MD28592 $MC_MM_NUM_SINGULARITY_BLOCKS = <value>

<value>	Meaning:
0	Function "Singularity handling" switched **off**
1	Function "Singularity handling" switched **on**

10.2.2 Precision criterion

Using the setting data, the accuracy criterion to detect a singularity on the programmed path is set:

SD41670 $SN_SINGULARITY_THRESHOLD = <value>

Recommended setting: $2.0*10^{-6}$

Constraints

The higher the higher the selected value, the greater the probability that a non-singular position is detected as such. As a consequence, a singularity handling with corresponding path deviations is unnecessarily executed.

If the value is selected too low, then it is possible that the function does not identify an existing singularity.

10.2.3 Minimum clearance

Using the setting data, the minimum clearance to the singularity point on the programmed path is set:

SD41671 $SN_MIN_DIST_TO_SINGULARITY = <value>

Recommended setting: $1.0*10^{-3}$

Constraints

Traversing blocks, whose positions lie between the accuracy criteria (SD41670 $SN_SINGULARITY_THRESHOLD (Page 85)) and the minimum clearance to the singularity point parameterized here are more intensively analyzed.

10.3 Constraints

"Advanced Surface" function

Function "Singularity handling" does not support smoothing with G64 when function "Advanced Surface" is active. When changing from "Path motion" (`CP`) to interpolation type "Point-to-point" (`PTP`), an exact stop is always made at the end of the block.

"Singularity handling" function

Superimposed traversing motion in the basic coordinate system (BCS) (e.g. $AA_OFF, TOFFL, DRV shifting/offset) is not taken into consideration by the "Singularity handling" function. As a consequence, this superimposed traversing motion can result in traversing motion in the vicinity of or through singularities even if function "Singularity handling" is active.

Singular block too long

If a programmed traversing block is completely singular and the path length ≥ 30 mm, then alarm 75327 "Singular block too long" is displayed and machining is canceled.

Functions that are excluded

Depending on the performance, function "Singularity handling" cannot be used in conjunction with the following functions:

- "Top Surface" function
- "Collision detection" function

11 Data lists

11.1 Machine data, short

11.1.1 General machine data

Number: MD	Identifier: $MN_	Description
10620	EULER_ANGLE_NAME_TAB[n]	Name of Euler angle
19610	TECHNO_EXTENSION_MASK[3].Bit 16	Option data for ROBX transformation, (0x10000H)

11.1.2 Channel-specific machine data

Number: MD	Identifier: $MC_	Description
28090	MM_NUM_CC_BLOCK_ELEMENTS	Number of block elements for compile cycles (**1** Block element is additionally required!)
28100	MM_NUM_CC_BLOCK_USER_MEM	Size of the memory for compile cycles block elements [kB] (**approx. 100 kB** is additionally required!)
28105	MM_NUM_CC_HEAP_MEM	Size of the dynamic memory for compile cycles in the channel [kB] (**100 kB** is additionally required!)
21100	ORIENTATION_IS_EULER	Angle definition for orientation programming
21102	ORI_DEF_WITH_G_CODE	Defining the orientation axes via G code
21104	ORI_IPO_WITH_G_CODE	G code for orientation interpolation
21120	ORIAX_TURN_TAB_1	Reference axis definition for orientation axes
21130	ORIAX_TURN_TAB_2	Reference axis definition for orientation axes
21110	X_AXIS_IN_OLD_X_Z_PLANE	Coordinate system with automatic FRAME definition
24100	TRAFO_TYPE_1	Definition of the transformation
24110	TRAFO_AXES_IN_1	Axis assignment for transformation 1
24120	TRAFO_GEOAX_ASSIGN_TAB_1	Assignment of geometry axes to the channel axes
24130	TRAFO_INCLUDES_TOOL_1	The transformation executes tool handling
24580	TRAFO5_TOOL_VECTOR_1	Orientation vector direction
24585	TRAFO5_ORIAX_ASSIGN_TAB_1	Orientation axis/channel axis assignment
28592	MM_NUM_SINGULARITY_BLOCKS	Activating the NCK preprocessing station to handle singularities

11.1.3 Channel-specific machine data for ROBX

See Chapter "Channel-specific machine data for ROBX (Page 90)"

11.2 Machine data, long

11.2.1 Channel-specific machine data for ROBX

62900	ROBX_KINCLASS				00.00.00	
	Kinematic category				NEWCONF	
						DWORD
The following kinematic types can be specified: • Standard transformation: 1 • Special transformation: 2						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		1	1	2	KEY-SWITCH_0/ OEM_LOW	

62901	ROBX_AXES_TYPE				00.00.00	
	Axis type for transformation [axis No.]: 0...5				NEWCONF	
						DWORD
This machine data defines the axis type used in the transformation. The following axis types are possible: • Linear axis: 1 • Triangular/trapezoidal spindle drive: 2 • Rotary axis: 3 • Endlessly rotating rotary axis: 4						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	6	1, 1, 1, 3, 3, 3	1	4	KEY-SWITCH_0/ OEM_LOW	

62902	ROBX_SPECIAL_KIN				00.00.00	
	Special kinematic type				NEWCONF	
						DWORD

62902	ROBX_SPECIAL_KIN				00.00.00	
This machine data defines the special kinematics type. The following special kinematics are available: • No special kinematics:1 • Articulated joint 5-axis machine with axis 2 coupled to axis 3: 2						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		1			KEY-SWITCH_0/ OEM_LOW	

62903	ROBX_MAIN_AXES				00.00.00	
	Basic axis identifier				NEWCONF	
						DWORD
This machine data defines the type of the basic axis layout. The basic axes are normally considered to be the first 3 axes. The following basic axis arrangements are included: • SS (gantry): 1 • CC (Scara): 2 • NR (articulated arm): 3 • SC (Scara): 4 • RR (articulated arm): 5 • CS (Scara): 6 • NN (articulated arm): 7						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		1	1	7	KEY-SWITCH_0/ OEM_LOW	

62904	ROBX_WRIST_AXES				00.00.00	
	Wrist axis identifier				NEWCONF	
						DWORD
This machine data defines the robot wrist type. Axes 4 to 6 are normally called robot wrist. The following wrist types are included: • No wrist: 1 • Central wrist: 2 • Inclined wrist axis: 3 • Angled wrist axis: 5 • Inclined wrist axis: 6						

62904	ROBX_WRIST_AXES				00.00.00	
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		1	1	6	KEY-SWITCH_0/ OEM_LOW	

62905	ROBX_NUM_AXES				00.00.00	
	Number of transformed axes				NEWCONF	
						DWORD
This machine data defines the number of axes used in the transformation. Kinematics with a maximum of 6 axes are supported.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		4	4	6	KEY-SWITCH_0/ OEM_LOW	

62906	ROBX_A4PAR				00.00.00	
	Axis 4 is parallel/anti-parallel to the last basic axis				NEWCONF	
						DWORD
This machine data defines whether the 4th axis is parallel/antiparallel to the last rotatory basic axis. This machine data is only relevant for kinematics with more than 3 axes. • Axis 4 is parallel/anti-parallel: 1 • Axis 4 is not parallel 0						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		0	0	1	KEY-SWITCH_0/ OEM_LOW	

62907	ROBX_MAIN_LENGTH_AB				00.00.00	
mm	Basic axis lengths A and B, n = 0...1				NEWCONF	
						DOUBLE
This machine data defines the basic axis lengths A and B. These lengths are specifically defined for each basic axis type. • n = 0: Basic axis length A • n = 1: Basic axis length B						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	2	0.0, 500.0			KEY-SWITCH_0/ OEM_LOW	

62908	ROBX_TX3P3_POS				00.00.00	
mm	Attaching the wrist (position component), n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the position component of frame TX3P3 frame, which connects the basic axes with the wrist. • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62909	ROBX_TX3P3_RPY				00.00.00	
Degrees	Attaching the wrist (rotation component), n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the orientation component of frame TX3P3 frame, which connects the basic axes with the wrist. • Index 0: Rotation around RPY angle A • Index 1: Rotation around RPY angle B • Index 2: Rotation around RPY angle C						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62910	ROBX_TFLWP_POS				00.00.00	
mm	Frame between the wrist point and flange coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the position component of frame TFLWP, which connects the wrist point to the flange. • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62911	ROBX_TFLWP_RPY				00.00.00	
Degrees	Frame between the wrist point and flange coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the orientation component of frame TFLWP, which connects the wrist point to the flange. • Index 0: Rotation around RPY angle A • Index 1: Rotation around RPY angle B • Index 2: Rotation around RPY angle C						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, -90.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62912	ROBX_TIRORO_POS				00.00.00	
mm	Frame between the base point and internal coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the position component of frame TIRORO, which connects the base point coordinate system (RO) defined by the user with the internal transformation coordinate system (IRO). The base point coordinate system (RO) can be specified referred to a world coordinate system (WO). See: • MD62971 $MC_ROBX_TROWO_POS • MD62972 $MC_ROBX_TROWO_RPY The world coordinate system corresponds to the basic coordinate system (BCS). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62913	ROBX_TIRORO_RPY		00.00.00	
Degrees	Frame between the base point and internal coordinate system, n = 0...2		NEWCONF	
				DOUBLE

62913	ROBX_TIRORO_RPY				00.00.00	
This machine data defines the orientation component of frame TIRORO, which connects the base point coordinate system defined by the user with the internal transformation coordinate system. The base point coordinate system (RO) can be specified referred to a world coordinate system (WO). The world coordinate system corresponds to the basic coordinate system (BCS). • Index 0: Rotation around RPY angle A • Index 1: Rotation around RPY angle B • Index 2: Rotation around RPY angle C						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62914	ROBX_DHPAR4_5A				00.00.00	
mm	Parameter A to configure the wrist, n = 0...1				NEWCONF	
						DOUBLE
This machine data defines length a. • n = 0: Transition, axis 4 to 5 • n = 1: Transition, axis 5 to 6						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	2	0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62915	ROBX_DHPAR4_5D				00.00.00	
mm	Parameter D to configure the wrist, n = 0...1				NEWCONF	
						DOUBLE
This machine data defines length d. • n = 0: Transition, axis 4 to 5 • n = 1: Transition, axis 5 to 6						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	2	0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62916	ROBX_DHPAR4_5ALPHA				00.00.00	
Degrees	Parameter ALPHA to configure the wrist, n = 0...1				NEWCONF	
						DOUBLE
This machine data defines angle alpha. • n = 0: Transition, axis 4 to 5 • n = 1: Transition, axis 5 to 6						

62916	ROBX_DHPAR4_5ALPHA				00.00.00	
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	2	-90.0, 90.0			KEY-SWITCH_0/OEM_LOW	

62917	ROBX_MAMES				00.00.00	
	Offset math. to mech. zero point [Axis No.]: 0...5				NEWCONF	
						DOUBLE
Using this machine data the zero point for a rotary or translatory axis can be adapted to the mathematical zero point, specified by the transformation. In this case, the offset should be entered starting from the mechanical zero point referred to the mathematical positive direction of rotation of the axis.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	6	0.0, 0.0, 0.0, 0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62918	ROBX_AXES_DIR				00.00.00	
	Adapting the phys. and math. direction of rotation [axis No.]: 0...5				NEWCONF	
						DWORD
This machine data can be used to match the mathematical direction of rotation to the physical direction of rotation of the axes. • +1: Direction of rotation is the same • -1: Direction of rotation is different						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	6	1, 1, 1, 1, 1, 1	-1	1	KEY-SWITCH_0/OEM_LOW	

62919	ROBX_DIS_WRP				00.00.00	
mm	Mean distance between wrist point and singularity				NEWCONF	
						DOUBLE
This machine data can be used to enter a limit value for the distance between the wrist point and the singularity. Note: Currently not active!						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		10.0	0.00001	999999.9999	KEY-SWITCH_0/OEM_LOW	

62920	ROBX_AXIS_SEQ				00.00.00	
	Rearrangement of axes				NEWCONF	
						DWORD
Using this machine data, the sequence of the axes can be interchanged in order to internally change a kinematic to a standard kinematic.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	6	1, 2, 3, 4, 5, 6	1	6	KEYSWITCH_0/ OEM_LOW	

62921	ROBX_SPIN_ON				00.00.00	
	Configuration includes triangular or trapezoidal spindles				NEWCONF	
						DWORD
This machine data defines whether triangular spindles or trapezoidal connections are available. • 0: none available • 1: available **Note** This function is presently not supported. MD62621 must be set to 0. Machine data MD62622 to MD62628 are therefore not active!						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		0	0	1	KEYSWITCH_0/ OEM_LOW	

62922	ROBX_SPIND_AXIS				00.00.00	
	Axis that acts on the triangular spindle, n = 0...2				NEWCONF	
						DWORD
This machine data defines on which axis a triangular spindle acts. A maximum of 3 triangular spindles can be available. • n = 0: 1st triangular spindle • n = 1: 2nd triangular spindle • n = 2: 3rd triangular spindle						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0, 0, 0			KEYSWITCH_0/ OEM_LOW	

62923	ROBX_SPINDLE_RAD_G				00.00.00	
mm	Length G for triangular spindle, n = 0...2				NEWCONF	
						DOUBLE

62923	ROBX_SPINDLE_RAD_G				00.00.00	
This machine data defines length G for the n-th triangular spindle.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62924	ROBX_SPINDLE_RAD_H				00.00.00	
mm	Length H for triangular spindle, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines length H for the n-th triangular spindle.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62925	ROBX_SPINDLE_SIGN				00.00.00	
	Signs for triangular spindle, n = 0...2				NEWCONF	
						DWORD
This machine data defines the sign for adapting the direction of rotation for the n-th triangular spindle.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	1, 1 ,1	-1	1	KEY-SWITCH_0/OEM_LOW	

62926	ROBX_SPINDLE_BETA				00.00.00	
Degrees	Angle offset for triangular spindle, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the offset angle b for adapting the zero point for the n-th triangular spindle.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62927	ROBX_TRP_SPIND_AXIS				00.00.00	
	Axes driven via a trapezoidal spindle, n = 0...1				NEWCONF	
						DWORD
This machine data defines which axes are driven via a trapezoidal connection. • n = 0: axis driven via trapezoid • n = 1: coupling axis						

62927	ROBX_TRP_SPIND_AXIS				00.00.00	
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard	2	0, 0			KEY-SWITCH_0/OEM_LOW	

62928	ROBX_TRP_SPIND_LEN				00.00.00	
mm	Trapezoidal lengths, n = 0...3				NEWCONF	
NO_GROUP						DOUBLE
This machine data specifies the lengths of the trapezoidal connection.						
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard	4	0.0, 0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62929	ROBX_VELCP				00.00.00	
mm/min	cartesian velocity [No.]: 0...2				IMMEDIATE	
						DOUBLE
Using this machine data, a velocity setpoint for the Cartesian directions can be specified for motion blocks with G0. • n = 0: Velocity in the x direction • n = 1: Velocity in the y direction • n = 2: Velocity in the z direction						
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard	3	600000.0, 600000.0, 600000.0			KEY-SWITCH_0/OEM_LOW	

62930	ROBX_ACCCP				00.00.00	
m/s^2	cartesian acceleration rates [No.]: 0...2				IMMEDIATE	
						DOUBLE
Using this machine data, an acceleration setpoint for the Cartesian directions can be specified for motion blocks with G0. • n = 0: Acceleration in the x direction • n = 1: Acceleration in the y direction • n = 2: Acceleration in the z direction						
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard	3	0.5, 0.5, 0.5	0.001	100000	KEY-SWITCH_0/OEM_LOW	

62931	ROBX_VELORI				00.00.00	
rpm	Orientation angle velocities [No.]: 0...2				IMMEDIATE	
						DOUBLE
Using this machine data, a velocity setpoint for the orientation angle can be specified for motion blocks with G0. • n = 0: Velocity angle A • n = 1: Velocity angle B • n = 2: Velocity angle C						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	1.6666, 1.6666, 1.6666			KEY-SWITCH_0/ OEM_LOW	

62932	ROBX_ACCORI				00.00.00	
rev/s²	Orientation angle-acceleration rates [No.]: 0...2				IMMEDIATE	
						DOUBLE
Using this machine data, an acceleration setpoint for the orientation angle can be specified for motion blocks with G0. • n = 0: Acceleration angle A • n = 1: Acceleration angle B • n = 2: Acceleration angle C						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.00277, 0.00277, 0.00277	0.001	100000	KEY-SWITCH_0/ OEM_LOW	

62934	ROBX_DYN_LIM_REDUCE				00.00.00	
	Reduction factor for the velocity controller				NEWCONF	
						DOUBLE
Using this machine data, a reserve can be specified for the maximum velocity so that an excessive velocity increase by the velocity controller does not result in the maximum velocity being exceeded. The value should be seen as a factor that acts on the maximum velocity.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		1.0	0.001	1.0	KEY-SWITCH_0/ OEM_LOW	

62935	ROBX_VEL_FILTER_TIME		00.00.00	
s	Velocity controller time constant		NEWCONF	
				DOUBLE

62935	ROBX_VEL_FILTER_TIME				00.00.00	
This machine data allows the time constant for the velocity controller in the interpolator to be set. This can be used to prevent the controller oscillating.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		0.024	0.0	100.0	KEY-SWITCH_0/ OEM_LOW	

62936	ROBX_CC_TOA_START_NUM				00.00.00	
	Start number for tool orientations				NEWCONF	
						DWORD
This machine data defines the number n of the 1st parameter $TC_DPCn, from which the OEM tool parameters are read-in for the orientation of the tool from the compile cycle. The OEM tool parameters are not read-in if the machine data is set to 0.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		0	-1	8	KEY-SWITCH_0/ OEM_LOW	

62937	ROBX_EXT_AXIS_VECTOR_1				00.00.00	
	Direction vector of the first additional axis, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the third direction vector of the first, linear additional axis regarding the basic coordinate system (BCS). The basic coordinate system (BCS) corresponds to the world coordinate system (WO). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 1.0	-1.0	1.0	KEY-SWITCH_0/ OEM_LOW	

62938	ROBX_EXT_AXIS_VECTOR_2				00.00.00	
	Direction vector of the second additional axis, n = 0...2				NEWCONF	
						DOUBLE

62938	ROBX_EXT_AXIS_VECTOR_2				00.00.00	
This machine data defines the third direction vector of the second additional axis regarding the basic coordinate system (BCS). The basic coordinate system (BCS) corresponds to the world coordinate system (WO). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 1.0, 0.0	-1.0	1.0	KEYSWITCH_0/OEM_LOW	

62939	ROBX_EXT_AXIS_VECTOR_3				00.00.00	
	Direction vector of the third additional axis, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the third direction vector of the additional axis regarding the basic coordinate system (BCS). The basic coordinate system (BCS) corresponds to the world coordinate system (WO). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 1.0	-1.0	1.0	KEYSWITCH_0/OEM_LOW	

62940	ROBX_EXT_ROT_AX_VECTOR_1				00.00.00	
	Direction vector of the first additional axis for a moved workpiece, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the direction vector of the first additional axis that rotates the workpiece. The direction is specified, referred to the world coordinate system (WO). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	1.0, 0.0, 0.0	-1.0	1.0	KEYSWITCH_0/OEM_LOW	

62941	ROBX_EXT_ROT_AX_VECTOR_2				00.00.00	
	Direction vector of the second additional axis for a moved workpiece, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the direction vector of the second additional axis that rotates the workpiece. The direction is specified, referred to the world coordinate system (WO). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	1.0, 0.0, 0.0	-1.0	1.0	KEY-SWITCH_0/ OEM_LOW	

62942	ROBX_EXT_ROT_AX_VECTOR_3				00.00.00	
	Direction vector of the third additional axis for a moved workpiece, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the direction vector of the third additional axis that rotates the workpiece. The direction is specified, referred to the world coordinate system (WO). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	1.0, 0.0, 0.0	-1.0	1.0	KEY-SWITCH_0/ OEM_LOW	

62943	ROBX_EXT_ROT_AX_OFFSET	00.00.00	
Degrees	Offset of the additional axes for a moved workpiece, n = 0...2	NEWCONF	
			DOUBLE
This machine data defines the rotary axis offset of the additional axes that rotate the workpiece. The deviation to be entered corresponds to the difference between the mechanical zero point and the mathematically positive direction of rotation of the axis. • Index 0: 1st additional axis • Index 1: 2nd additional axis • Index 2: 3rd additional axis			

62943	ROBX_EXT_ROT_AX_OFFSET				00.00.00	
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62944	ROBX_EXT_ROT_JOINT_OFFSET_1				00.00.00	
mm	Offset between the first and second additional axis, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the offset between the first and the second additional axes that rotate the workpiece. The offset must be specified in the mathematical zero position of the additional axes, referred to the directions of the world coordinate system (WO). • Index 0: x component • Index 1: y component • index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62945	ROBX_EXT_ROT_JOINT_OFFSET_2				00.00.00	
mm	Offset between the second and third additional axes, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the offset between the second and the third additional axes that rotate the workpiece. The offset must be specified in the mathematical zero position of the additional axes, referred to the directions of the world coordinate system (WO). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62946	ROBX_EXT_ROT_BASE_OFFSET				00.00.00	
mm	Offset between the robot base point and base point of the additional axes, n = 0...2				NEWCONF	
						DOUBLE

62946	ROBX_EXT_ROT_BASE_OFFSET				00.00.00	
This machine data defines the offset between the robot base point and the base point of the first additional axis that rotates the workpiece. The offset must be specified referred to the directions of the world coordinate system (WO). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62947	ROBX_EXT_ROT_PART_OFFSET				00.00.00	
mm	Offset between the last additional axis and the basic coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the offset between the last additional axis, which rotates the workpiece, and the basic coordinate system (BCS). The offset must be specified referred to the directions of the world coordinate system (WO). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62948	ROBX_TFL_EXT_RPY				00.00.00	
Degrees	Adapting the tool orientation, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the orientation component of the frame, which can be used to adapt the tool direction. • Index 0: Rotation around RPY angle A • Index 1: Rotation around RPY angle B • Index 2: Rotation around RPY angle C						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62949	ROBX_TOOL_DIR				00.00.00	
	Definition of the tool direction for 5/6-axis robots				NEWCONF	
						DWORD
This machine data defines whether the tool direction for 5-axis machines should be defined according to the robotic or NC convention. This also influences how the rotation sequence of the virtual orientation axes must be defined. • 0: Robotic convention, tool direction is X, rotation sequence is Z, Y, X • +1: NC convention, tool direction is Z, rotation sequence is X, Y, Z, tool length is negated • -1: Robotic convention, tool lengths are negated						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		0	-1	1	KEY-SWITCH_0/OEM_LOW	

62950	ROBX_EXT_AX_OFFSET				00.00.00	
mm	Work offset of the additional axes, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the work offset of the additional axes that move the robot or the workpiece. • Index 0: 1st additional axis • Index 1: 2nd additional axis • Index 2: 3rd additional axis						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/OEM_LOW	

62951	ROBX_GRIP_REL_RESET_MODE				04.00.00	
	Reset behavior, gripper-referred motion				RESET	
						DWORD
This machine data defines the reset response for gripper-referred interpolation • 0: Active setting remains valid • 1: Gripper-referred interpolation off • 2: Gripper-referred interpolation on						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		0	0	2	KEY-SWITCH_0/OEM_LOW	

62952	ROBX_A_DBD_START_INDEX				04.00.00	
	Start index for state output via $A_DBD variable				POWER ON	
						DWORD
This machine data defines the start index for the $A_DBD status variable. Value: • -1: The state of the gripper-referred interpolation is not output. • 0, 4, 8, ... : The state of the gripper-referred interpolation is output in $A_DBD[<value>]. Note: The index must be a multiple of 4.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		-1	-1	4088	KEY-SWITCH_0/ OEM_LOW	

62953	ROBX_SPECIAL_FEATURE_MASK				05.02.00	
	Activating special features				RESET	
						DWORD
Special features can be activated using this machine data. • Bit 0 = 0: For 4-axis SCARA kinematics, the C angle is controlled in the BCS in the range between -180 degrees and +180 degrees. • Bit 0 = 1: For 4-axis SCARA kinematics, the C angle is controlled in the BCS in absolute terms						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		0			KEY-SWITCH_0/ OEM_LOW	

62954	ROBX_C_ANGLE_OFFSET				05.02.00	
Degrees	Offset angle for C angle calculation in the BCS				NEWCONF	
						DOUBLE
Using this machine data, an offset angle can be entered to calculate the C angle in the BCS for 4-axis SCARA kinematics.						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard		0.0			KEY-SWITCH_0/ OEM_LOW	

62964	ROBX_SW_LIMITS_PARALLEL				05.05.00	
Degrees	Limit angle for self collision monitoring				NEWCONF	
						DOUBLE
Using this machine data, for robots with a parallelogram structure, the upper and lower limit angle can be set to check for self collisions.						

62964	ROBX_SW_LIMITS_PARALLEL				05.05.00	
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard	2	-1000.0, 1000.0			KEY-SWITCH_0/ OEM_LOW	

62965	ROBX_TTCFL_POS				06.00.00	
mm	Frame between flange and tool carrier coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the position component of frame TTCFL, which connects the flange to the tool carrier. • Index 0: x component • Index 1: y component • Index 2: z component						
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62966	ROBX_TTCFL_RPY				06.00.00	
Degrees	Frame between flange and tool carrier coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the orientation component of frame TTCFL, which connects the flange to the tool carrier. • Index 0: Rotation around RPY angle A • Index 1: Rotation around RPY angle B • Index 2: Rotation around RPY angle C						
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62967	ROBX_ITERMAXCOUNT	06.01.00	
	Maximum number of iteration runs for a transformation	IMMEDIATE	
			DWORD
This machine data defines the maximum number of iteration runs for a transformation.			

62967	ROBX_ITERMAXCOUNT				06.01.00	
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard		8	3	18	KEY-SWITCH_0/ OEM_LOW	

62968	ROBX_ITERPRECISION				06.01.00	
	Termination barrier for internal target function				IMMEDIATE	
						DOUBLE
Using this machine data, a termination barrier can be entered for the internal target function.						
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard		1.0e-6	1.0e-16	100.0	KEY-SWITCH_0/ OEM_LOW	

62969	ROBX_ITERLIMIT				06.01.00	
	Limit angle for handling a singular position				RESET	
						DOUBLE
Using this machine data, a limit angle for handling a singular position can be entered.						
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard		3.0	0.0	10.0	KEY-SWITCH_0/ OEM_LOW	

62970	ROBX_ITERLIMIT2				06.01.00	
	Help angle to retract from a singular position:				RESET	
						DOUBLE
Using this machine data a help angle can be entered to retract from a singular position.						
System	Dimension	Default value	Minimum value	Maximum value	Protection	Class
Standard		0.0	0.0	10.0	KEY-SWITCH_0/ OEM_LOW	

62971	ROBX_TROWO_POS				00.00.00	
mm	Frame between the world and base point coordinate system, n = 0...2				NEWCONF	
						DOUBLE

62971	ROBX_TROWO_POS				00.00.00	
This machine data defines the position component of frame TROWO, which connects the world coordinate system (WO) / basic coordinate system (BCS) with the base point coordinate system (RO) of the robot. • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62972	ROBX_TROWO_RPY				00.00.00	
Degrees	Frame between world and base point coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the orientation component of frame TROWO, which connects the world coordinate system (WO) / basic coordinate system (BCS) with the base point coordinate system (RO) of the robot. • Index 0: Rotation around RPY angle A • Index 1: Rotation around RPY angle B • Index 2: Rotation around RPY angle C						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62973	ROBX_EXT_FL_ROT_AX_VECTOR_1				00.00.00	
	Direction vector of the first additional axis in the flange coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the direction vector of the first additional axis in the flange coordinate system to rotate the tool. The direction is specified, referred to the flange coordinate system (FL). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	1.0, 0.0, 0.0	-1.0	1.0	KEY-SWITCH_0/ OEM_LOW	

62974	ROBX_EXT_FL_ROT_AX_VECTOR_2				00.00.00	
	Direction vector of the second additional axis in the flange coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the direction vector of the second additional axis in the flange coordinate system to rotate the tool. The direction is specified, referred to the flange coordinate system (FL). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 1.0, 0.0	-1.0	1.0	KEY-SWITCH_0/ OEM_LOW	

62975	ROBX_EXT_FL_ROT_AX_VECTOR_3				00.00.00	
	Direction vector of the third additional axis in the flange coordinate system, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the direction vector of the third additional axis in the flange coordinate system to rotate the tool. The direction is specified, referred to the flange coordinate system (FL). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 1.0	-1.0	1.0	KEY-SWITCH_0/ OEM_LOW	

62976	ROBX_EXT_FL_ROT_AX_OFFSET		00.00.00
Degrees	Offset of the additional axes in the flange coordinate system, n = 0...2		NEWCONF
			DOUBLE
This machine data defines the rotary axis offset of the additional axes that rotate the tool. The deviation to be entered corresponds to the difference between the mechanical zero point and the mathematically positive direction of rotation of the axis. • Index 0: 1st additional axis • Index 1: 2nd additional axis • Index 2: 3rd additional axis			

62976	ROBX_EXT_FL_ROT_AX_OFFSET				00.00.00	
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62977	ROBX_EXT_FL_ROT_JOINT_OFF_1				00.00.00	
mm	Offset between the first and second additional axis, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the offset between the first and the second additional axes that rotate the tool. The offset must be specified in the mathematical zero position of the assitional axes, referred to the directions of the flange coordinate system (FL). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62978	ROBX_EXT_FL_ROT_JOINT_OFF_2				00.00.00	
mm	Offset between the second and third additional axis, n = 0...2				NEWCONF	
						DOUBLE
This machine data defines the offset between the second and the third additional axes that rotate the workpiece. The offset must be specified in the mathematical zero position of the additional axes, referred to the directions of the flange coordinate system (FL). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

62979	ROBX_EXT_FL_ROT_BASE_OFFSET				00.00.00	
mm	Offset between the flange and base point of the additional axes, n = 0...2				NEWCONF	
						DOUBLE

62979	ROBX_EXT_FL_ROT_BASE_OFFSET				00.00.00	
This machine data defines the offset between the flange and the base point of the first additional axis that rotates the tool. The offset must be specified referred to the directions of the flange coordinate system (FL). • Index 0: x component • Index 1: y component • Index 2: z component						
System	**Dimension**	**Default value**	**Minimum value**	**Maximum value**	**Protection**	**Class**
Standard	3	0.0, 0.0, 0.0			KEY-SWITCH_0/ OEM_LOW	

11.3 Setting data

11.3.1 NC-specific setting data for ROBX

Number: SD	Identifier: $SN_	Description
41670	SINGULARITY_THRESHOLD	Precision criterion
41671	MIN_DIST_TO_SINGULARITY	Minimum clearance

11.4 PLC signals

Channel-specific PLC signals

Basic Program Plus	Basic Program		Description
<Chan>.basic.out.ptpTravers	LBP_Chan*.A_PTP_Travel	DB21, ...DBX29.4	Activate PTP travel
<Chan>.basic.in.ptpTraversActive	LBP_Chan*.E_PTP_Travel	DB21, ...DBX317.6	PTP travel active
<Chan>.basic.in.transformationActive	LBP_Chan*.E_Transform	DB21, ...DBX33.6	Transformation active
<Chan>.basic.in.transformationActiveNumber	LBP_Chan*.E_TransformNo	DB21, ...DBB388	Number of the active transformation

11.5 Alarms

75310	%?C{channel %1: %}ROBX: incorrect MD configuration, error for MD: %2
Explanation	The following error was detected in the machine data of the ROBX transformation: • ROBX_IRORO: The orientation entered in MD62913 $MC_ROBX_TIRORO_RPY is not permissible. • ROBX_TFLWP: The orientation entered in MD62911 $MC_ROBX_TFLWP_RPY is not permissible. • ROBX_TX3P3: The orientation entered in MD62909 $MC_ROBX_TX3P3_RPY is not permissible. • ROBX_MAIN_LENGTH_AB: The value entered in MD62907 $MC_ROBX_MAIN_LENGTH_AB is incorrect.
Response	--NOREADY--, --SHOWALARM--
Remedy	Correct machine data
Clear criterion	--POWERONCLEAR--
Parameter %1	Channel number
Parameter %2	MD name

75311	%?C{channel %1: %}ROBX: Number of axes/axis assignment inconsistent
Explanation	When selecting the transformation, an incorrect axis assignment is detected: The axes entered in MD24110 $MC_TRAFO_AXES_IN_1 do not match what has been entered in MD62905 $MC_ROBX_NUM_AXES.
Response	--INTERPRETERSTOP--, --SHOWALARM--
Remedy	Correct machine data
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75312	%?C{channel %1: %}ROBX: Incorrect TRAFO_TYPE_ : use 4100
Explanation	The transformer type entered in MD24xx0 $MC_TRAFO_TYPE_x is incorrect
Response	--INTERPRETERSTOP--, --SHOWALARM--
Remedy	TRAFO_TYPE 4100 must be used
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75313	%?C{channel %1: %}ROBX: Block %2 software limit switch axis %3 direction %4 for block preparation exceeded
Explanation	For block preparation in preprocessing, with a transformation active, a violation of the software limit switch of an axis is detected.
Response	--COMPBLOCKWITHREORG--, --LOCALREACTION--, --NONCSTART--, --SETVDI--, --SHOWALARM--, --STOPATENDBYALARM--
Remedy	Programming a position that is located before the software limit switch.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Block number
Parameter %3	Axis
Parameter %4	Direction

75314	%?C{channel %1: %}ROBX: MD change when transformation is active not permissible
Explanation	An attempt was made to change machine data via NEWCONF with a transformation active. This is not permissible, as changing the machine data directly effects the actual axis position, which is transformed from the basic coordinate system into the machine coordinate system in realtime. Changing the transformation data with the transformation active would result in a jump in the axis positions.
Response	--INTERPRETERSTOP--, --SHOWALARM--
Remedy	Deactivate the transformation via TRAFOOF before the machine data are accepted via NEWCONF.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75315	%?C{channel %1: %}ROBX: Tool parameter incorrect interpreter
Explanation	When interpreting the block, an incorrect tool parameter is detected:
Response	--INTERPRETERSTOP--, --SHOWALARM--
Remedy	Correct tool parameters.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75316	%?C{channel %1: %}ROBX: Unreachable position of interpreter
Explanation	When interpreting the block, a position that cannot be approached is detected:
Response	--INTERPRETERSTOP--, --SHOWALARM--
Remedy	Correct part program.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75317	%?C{channel %1: %}ROBX: Block: %2, tool parameter incorrect for block preparation
Explanation	When preparing the block, incorrect tool parameters are detected:
Response	--INTERPRETERSTOP--, --LOCALREACTION--, --NONCSTART--, --SETVDI--, --SHOWALARM--
Remedy	Correct tool parameters.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Block number

75318	%?C{channel %1: %}ROBX: Block: %2, unreachable position on block editing
Explanation	When preparing the block, a position that cannot be approached is detected:
Response	--INTERPRETERSTOP--, --LOCALREACTION--, --NONCSTART--, --SETVDI--, --SHOWALARM--
Remedy	Correct part program.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Block number

75319	%?C{channel %1: %}ROBX: Option for ROBX transformation not set
Explanation	The option bit for the ROBX transformation in $ON_TECHNO_EXTENSION_MASK is not set. As a consequence, the transformation is not selected. From ROBX SOFTWARE RELEASE 06.01.00, this alarm is no longer output as the option bit for RODI is no longer interrogated.
Response	--NOREADY--, --SHOWALARM--
Remedy	Set option for ROBX transformation.
Clear criterion	--POWERONCLEAR--
Parameter %1	Channel number

75320	%?C{channel %1: %}ROBX: Tool parameters incorrect on interpolation
Explanation	When interpolating, incorrect tool parameters are detected:
Response	--NONCSTART--, --SHOWALARM-- , --STOPBYALARM--
Remedy	Correct tool parameters.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75321	%?C{channel %1: %}ROBX: Gripper-referred motion control not permitted
Explanation	An attempt is being made to activate the "Gripper-referred motion control" function on a machine with less than 6 axes or for 5-axis robots that are not type SS, CC, or SC and whose 4th axis is not parallel to the 3rd axis.
Response	--SHOWALARM--, -- INTERPRETERSTOP --
Remedy	Correct the part program or adapt the reset behavior via MD 62951 ROBX_GRIP_REL_RESET_MODE
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75322	%?C{channel %1: %}ROBX: Error %2 on creation of CC block elements
Explanation	The following error occurred when creating CC block elements in the interpreter: • 1: No block memory for CC block elements available • 2: No block available
Response	--INTERPRETERSTOP--, --SHOWALARM--
Remedy	Increase machine data MD28100 $MC_MM_NUM_CC_BLOCK_USER_MEM or MD 28090 $MC_MM_NUM_CC_BLOCK_ELEMENTS.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Error number

75323	%?C{channel %1: %}ROBX: Software limit switch axis %2 direction %3 violated with transformation
Explanation	When traversing with a transformation active in the JOG mode, a violation of the software limit switch of an axis is detected.
Response	--NONCSTART--, --SETVDI-- , --SHOWALARM-- , --STOPBYALARM--
Remedy	JOG travel in the opposite direction.

Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Axis

75324	%?C{channel %1: %}ROBX: Excessive velocity error code %2 old value %3 new value %4
Explanation	When traversing with a transformation active, especially in the JOG mode in the vicinity of a pole, the velocity of the machine axes involved can excessively increase. An alarm is initiated if the velocity or acceleration increases excessively for an output axis of the transformation (MCS). Error code: • 0 : Position step • 1: Excessive speed increase • 2: Excessive acceleration increase Depending on whether a position step or an excessive velocity or acceleration increase is involved, as 3rd and 4th parameters the old and the new values for position, velocity or acceleration are output.
Response	--NONCSTART--, --SETVDI-- , --SHOWALARM-- , --STOPBYALARM--
Remedy	Reduce velocity. Avoid traversing in JOG in the vicinity of a pole
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Error code
Parameter %3	Old value
Parameter %4	New value

75325	%?C{channel %1: %}ROBX: Block %2, unreachable position on interpolation
Explanation	When interpolating the block, a position that cannot be approached is detected:
Response	--NONCSTART--, --SHOWALARM-- , --STOPBYALARM--
Remedy	Correct part program.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Block number

75326	%?C{channel %1: %}ROBX: Block %2, emergency stop
Explanation	Follow-on alarm to alarms 75324 and 75325. Since triggering of alarm **75324** slows down the robot axes and any additional axes on different braking ramps, triggering alarm 75326 brakes all axes on the braking ramp specified by the Emergency Stop Time (MD36610, $MA_AX_EMERGENCY_STOP_TIME). Since traversing near a singularity can lead to high axis speeds and the resulting triggering of the software limit switch monitoring, alarm 75326 is issued as a follow-up alarm to alarm **75325** to prevent an axle from traversing to the software limit switch.
Response	--NONCSTART--, --SHOWALARM--, --STOPBYALARM--, --SETVDI--, --FOLLOWUP--
Remedy	Reduce velocity. Avoid traversing in JOG in the vicinity of a pole
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75327	%?C{channel %1: %}ROBX: Block: %2, singular block is too long
Explanation	The path distance while the robot is in the singular range is too long (> 30 mm).
Response	--SHOWALARM--
Remedy	Correct the part program and split the block.
Clear criterion	--CANCELCLEAR--
Parameter %1	Channel number
Parameter %2	Block number

75328	%?C{channel %1: %}ROBX: Block %2 limit angle in %3 direction for parallelogram exceeded during block preparation
Explanation	For block preparation in preprocessing, with a transformation active, for a robot with parallelogram structure ($MC_ROBX_SPECIAL_KIN = 2), a violation of the limit angle specified in MD 62964 is detected.
Response	--COMPBLOCKWITHREORG--, --LOCALREACTION--, --NONCSTART--, --SETVDI--, --SHOWALARM--, --STOPATENDBYALARM--
Remedy	Programming a position that is located within the limit angle range.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Block number
Parameter %3	Direction

75329	%?C{channel %1: %}ROBX: Limit angle for parallelogram exceeded
Explanation	When traversing with a transformation active in the JOG mode, or when traversing without transformation for a robot with parallelogram structure (MD62902 $MC_ROBX_SPECIAL_KIN = 2), a violation of the limit angle specified in MD62964 is detected.
Response	--NONCSTART--, --SETVDI--, --SHOWALARM--, --STOPBYALARM--
Remedy	JOG travel in the opposite direction or programming a position that is located within the limit angle range.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75334	%?C{channel %1: %}ROBX: Block %2 error %3 on creation of CC block elements
Explanation	The following error occurred when creating CC block elements for the block preparation: • 1: No block memory for CC block elements available • 2: No block available
Response	--INTERPRETERSTOP--, --NONCSTART--, --SETVDI--, --SHOWALARM--
Remedy	Increase machine data MD28100 $MC_MM_NUM_CC_BLOCK_USER_MEM.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Block number
Parameter %3	Error number

75335	%?C{channel %1: %}ROBX: Transformation prevents movements with unreferenced axes
Explanation	In order to guarantee that transformation functions correctly, the axes, involved in the transformation, must be referenced before selecting the transformation. This alarm is output at the first traversing motion after transformation has been selected without referenced axes.
Response	--NONCSTART--, --SETVDI--, --SHOWALARM--, --STOPBYALARM--
Remedy	Reference the machine axes, deselect transformation and select again.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75370	%?C{channel %1: %}ROBX: Unreachable position interpreter Alpha 5
Explanation	When interpreting the block, axis 5 comes closer to 0 than MD62970 $MC_ROBX_ITERLIMIT2.
Response	--INTERPRETERSTOP--, --SHOWALARM--
Remedy	Correct part program.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

75371	%?C{channel %1: %}ROBX: Block: %2, unreachable position with block preparation Alpha 5
Explanation	When preparing the block, axis 5 comes closer to 0 than MD62970 $MC_ROBX_ITERLIMIT2.
Response	--INTERPRETERSTOP--, --NONCSTART--, --SETVDI--, --SHOWALARM--
Remedy	Correct part program.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Block number

75372	%?C{channel %1: %}ROBX: Block %2, unreachable position with interpolation Alpha 5
Explanation	When interpolating the block, axis 5 comes closer to 0 than MD62970 $MC_ROBX_ITERLIMIT2.
Response	--NONCSTART--, --SHOWALARM--, --STOPBYALARM--
Remedy	Correct part program.
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number
Parameter %2	Block number

75373	%?C{channel %1: %}ROBX: Additional axes not permitted in the flange coordinate system
Explanation	When selecting the transformation, an incorrect axis assignment is detected: Additional axes are not permitted in the flange coordinate system for robots with less than 6 axes
Response	--INTERPRETERSTOP-- , --SHOWALARM--
Remedy	Correct machine data
Clear criterion	--RESETCLEAR--
Parameter %1	Channel number

List of abbreviations

A

A	
A	Output
AFIS	Automatic Filter Switch: Automatic filter switch
ASCII	American Standard Code for Information Interchange: American coding standard for the exchange of information
ASIC	Application Specific Integrated Circuit: User switching circuit
ASUP	Asynchronous subprogram
AUTO	Operating mode "Automatic"
AUXFU	Auxiliary Function: Auxiliary function
AWL	Statement list

B	
BAG	Mode group
BCS	Basic coordinate system
BCD	Binary Coded Decimals: Decimal numbers encoded in binary code
BICO	Binector Connector
BIN	Binary Files: Binary files
BKS	Basic coordinate system
BM	Operating alarm
BO	Binector Output
BTSS	Operator panel interface

C	
CLC	Clearance control
CNC	Computerized Numerical Control: Computerunterstützte numerische Steuerung
COM	Communication
CP	Communication Processor
CP	continuos path
CPU	Central Processing Unit: Central processing unit
CST	Configured Stop: Configured stop

D	
DDS	Drive Data Set: Drive data set
DIR	Directory: Directory
DO	Drive Object
DRF	Differential Resolver Function: Differential resolver function (handwheel)

D	
DRY	Dry Run: Dry run feedrate
DWORD	Double word (currently 32 bits)

E	
E	Input
EES	Execution from External Storage
E/A	Input/Output
ESR	Extended stop and retract
ETC	ETC key ">"; Softkey bar extension in the same menu

F	
FDD	Feed Disable: Feed disable
FdStop	Feed Stop: Feed stop
FIFO	First In First Out: Memory that works without address specification and whose data is read in the same order in which they was stored
FL	Flange coordinate system
FM	Error message
FUP	Control system flowchart (PLC programming method)
FW	Firmware

G	
GEO	Geometry, e.g. geometry axis
GP	Basic program (PLC)
GUD	Global User Data: Global user data

H	
HEX	Abbreviation for hexadecimal number
HiFu	Auxiliary function
HMI	Human Machine Interface: SINUMERIK user interface
HSA	Main spindle drive
HP	Wrist point coordinate system
HT	Handheld Terminal
HW	Hardware

I	
IBN	Commissioning
INC	Increment: Increment
INI	Initializing Data: Initializing data

I	
IPO	Interpolator
IRO	Internal robot coordinate system

J	
JOG	JOGging: Setup mode

K	
KOP	Ladder diagram (PLC programming method)

L	
LED	Light Emitting Diode: Light-emitting diode
LMS	Position measuring system
LR	Position controller

M	
Main	Main program: Main program (OB1, PLC)
MCP	Machine Control Panel: Machine control panel
MD	Machine Data
MDA	Manual Data Automatic: Manual input
MDS	Motor Data Set: Motor data set
MELDW	Message word
MKS	Machine coordinate system
MM	Motor Module
MPF	Main Program File: Main program (NC)
MPI	Multi Point Interface
MSTT	Machine control panel

N	
NC	Numerical Control: Numerical control with block preparation, traversing range, etc.
NCU	Numerical Control Unit: NC hardware unit
NCK	Numerical Control Kernel
NCSD	NC Start Disable
NST	Interface signal
NV	Work offset
NX	Numerical Extension: Axis expansion module

O	
OB	Organization block in the PLC
OEM	Original Equipment Manufacturer
OP	Operation Panel: Operating equipment

P	
p3, q3, r3	Coordinate system of the last basic axis
PCU	PC Unit: PC box (computer unit)
PG	Programming device
PI	Program instance
PLC	Programmable Logic Control: Controller
PN	PROFINET
PO	Power On
POS	Position/positioning
PPO	Parameter process data object: Cyclic data telegram for PROFIBUS DP transmission and "Variable speed drives" profile
PPU	Panel Processing Unit (central hardware for a panel-based CNC, e.g. SINUMERIK 828D)
PROFIBUS	Process Field Bus: Serial data bus
PRT	Program test
PTP	Point to Point: Point-to-point
PZD	Process data: Process data part of a PPO

R	
REF	REFerence point approach function
REPOS	REPOSition function
RESU	Retrace support
RID	Read In Disable
RO	Robot or base point coordinate system (= basic coordinate system)
RP	R Parameter, arithmetic parameter, predefined user variable
RPY	Roll, pitch, yaw: Angle definition for describing the orientation

S	
SA	Synchronized action
SBL	Single Block: Single block
SBT	Safe Brake Test
SCC	Safety Control Channel
SCL	Structured Control Language
SD	Settingdatum or setting data
SDI	Safe Direction
SERUPRO	Search-Run by Program Test: Block search via program test
SIC	Safety Info Channel

S	
SKP	Skip: Function for skipping a part program block
SLP	Safe Limited Position
SLS	Safely Limited Speed
SMI	Sensor Module Integrated
SOS	Safe Operating Stop
SPF	Sub Program File: Subprogram (NC)
SS1	Safe Stop 1
SS2	Safe Stop 2
STO	Safe Torque Off
STW	Control word
SUG	Grinding wheel peripheral speed
SW	Software

T	
TC	Tool carrier coordinate system
TCP	tool center point (tool tip)
TCS	tool coordinate system
TCU	Thin Client Unit
TIA	Totally Integrated Automation
TM	Terminal Module (SINAMICS)
TO	Tool Offset: Tool offset
TOA	Tool Offset Active: Identifier (file type) for tool offsets
TOFF	Online tool length offset
TRANSMIT	Transform Milling Into Turning: Coordination transformation for milling operations on a turning machine

U	
UP	Subprogram
USB	Universal Serial Bus

V	
VDI	Internal communication interface between NC and PLC

W	
WCS	workpiece coordinate system
WKS	Workpiece coordinate system
WO	World coordinate system (world)
WPD	Work Piece Directory: Workpiece directory
WZ	Tool

W	
WZ	Tool coordinate system
WZV	Tool management

X	
x3, y3, z3	Coordinate system of the first wrist axis

Z	
ZSW	Status word (of drive)