



DOCUMENTATION ISG-kernel

Manual **Cycles - Kinematic Optimization**

Short Description:
CYCLES-KINOPT

Preface

Legal information

This documentation was produced with utmost care. The products and scope of functions described are under continuous development. We reserve the right to revise and amend the documentation at any time and without prior notice.

No claims may be made for products which have already been delivered if such claims are based on the specifications, figures and descriptions contained in this documentation.

Personnel qualifications

This description is solely intended for skilled technicians who were trained in control, automation and drive systems and who are familiar with the applicable standards, the relevant documentation and the machining application.

It is absolutely vital to refer to this documentation, the instructions below and the explanations to carry out installation and commissioning work. Skilled technicians are under the obligation to use the documentation duly published for every installation and commissioning operation.

Skilled technicians must ensure that the application or use of the products described fulfil all safety requirements including all applicable laws, regulations, provisions and standards.

Further information

Links below (DE)

<https://www.isg-stuttgart.de/produkte/softwareprodukte/isg-kernel/dokumente-und-downloads>

or (EN)

<https://www.isg-stuttgart.de/en/products/softwareproducts/isg-kernel/documents-and-downloads>

contains further information on messages generated in the NC kernel, online help, PLC libraries, tools, etc. in addition to the current documentation.

Disclaimer

It is forbidden to make any changes to the software configuration which are not contained in the options described in this documentation.

Trade marks and patents

The name ISG®, ISG kernel®, ISG virtuos®, ISG dirigent® and the associated logos are registered and licensed trade marks of ISG Industrielle Steuerungstechnik GmbH.

The use of other trade marks or logos contained in this documentation by third parties may result in a violation of the rights of the respective trade mark owners.

Copyright

© ISG Industrielle Steuerungstechnik GmbH, Stuttgart, Germany.

No parts of this document may be reproduced, transmitted or exploited in any form without prior consent. Non-compliance may result in liability for damages. All rights reserved with regard to the registration of patents, utility models or industrial designs.

General and safety instructions

Icons used and their meanings

This documentation uses the following icons next to the safety instruction and the associated text. Please read the (safety) instructions carefully and comply with them at all times.

Icons in explanatory text

- Indicates an action.
- ⇒ Indicates an action statement.



⚠ DANGER

Acute danger to life!

If you fail to comply with the safety instruction next to this icon, there is immediate danger to human life and health.



⚠ CAUTION

Personal injury and damage to machines!

If you fail to comply with the safety instruction next to this icon, it may result in personal injury or damage to machines.



Attention

Restriction or error

This icon describes restrictions or warns of errors.



Notice

Tips and other notes

This icon indicates information to assist in general understanding or to provide additional information.



Example

General example

Example that clarifies the text.



Programing Example

NC programming example

Programming example (complete NC program or program sequence) of the described function or NC command.



Release Note

Specific version information

Optional or restricted function. The availability of this function depends on the configuration and the scope of the version.

Contents

Preface.....	2
General and safety instructions	3
1 Cycles - Introduction	6
1.1 Note on installing the cycles.....	6
2 Overview.....	8
3 Description	10
3.1 Cycle-specific tool data	10
3.2 Preparations.....	11
3.2.1 CNC controller	11
3.2.2 Machine controller.....	11
3.3 Measurement and calculation cycle	12
3.3.1 General parameters	12
3.3.2 Setting the measurement poses	13
3.4 Kinematic-related arrangement of axes	17
4 Measuring the machine.....	18
4.1 General sequence of measurement run.....	18
4.2 Measuring with measurement subroutines	19
4.3 External measurement	20
4.3.1 Manual measurement with calibration sphere	20
4.3.2 Measuring the TCP (tool centre point).....	21
5 Preparing the calculation cycle.....	22
5.1 Adapting the output file	22
5.2 Adapting the output format.....	22
5.3 Adapting the limit for deviation	23
5.4 Initialising the kinematic calculation	23
5.5 Entering measurement results	24
5.5.1 Direct input of measurement data.....	24
5.5.2 Sphere centre points already calculated.....	26
5.6 Kinematic-specific changes.....	27
5.6.1 KIN_TYP_80	27
5.6.2 KIN_TYP_90	27
6 Executing the calculation cycle	28
6.1 Starting the main program.....	28
7 Checking the calibration.....	29
7.1 Plausibility check of the output.....	29
7.2 Determining missing offsets	32
7.2.1 KIN_TYP_80	32
7.3 Machining test part.....	32
8 Appendix	33
8.1 Suggestions, corrections and the latest documentation.....	33

List of figures

Fig. 1:	Meaning of geometry parameters.....	13
Fig. 2:	Example of distribution of measurement points on a hemisphere.....	20
Fig. 3:	Example of a test part – A or B in zero position	32

1 Cycles - Introduction



Notice

Cycles are additional options and subject to the purchase of a license.

General information

Cycle call

ISG cycles are called using a cycle call:

```
L CYCLE[ NAME="..." @P1 = .. @P2 = .. ...]
```

A cycle is called by specifying the cycle name. It is also possible to parameterise the cycle input parameters to modify cycle behaviour to a special application.

This documentation contains a separate subsection for each cycle where cycle behaviour is described in greater detail. It also contains a list of the input parameters used for the cycle. Finally, a simple programming example describes how to call the cycle.

Selecting the cycle plane

A cycle is programmed independently of the currently valid plane (G17, G18, G19) and independently of the axis names configured in the NC channel.

In the cycle documentation, the axes are described by the following names for the sake of better legibility:

- The X axis describes the 1st main axis
- The Y axis describes the 2nd main axis
- The Z axis describes the 3rd main axis
- A axis is the designation for the rotary axis about the 1st main axis
- B axis is the designation for the rotary axis about the 2nd main axis
- C axis is the designation for the rotary axis about the 3rd main axis



Notice

Cycles can also be used in offset and rotated coordinate systems. These coordinate systems should only be defined using the #CS command. The #ROTATION command is not suitable for use in combination with cycles.

1.1 Note on installing the cycles

Content of the cycle packet

The cycle packet contains the following files and directories:

- Set-up file to execute the cycle set-up.
- Dependency graph: Describes the dependencies (subroutine calls) for each of the cycles.
- Release notes: Description of relevant changes which the release contains.
- Documentation index: Contains the cycle documentation.
- Index of error messages: Contains the cycle error messages.
- Condensed index of cycles: This contains the CNC cycle and example call programs. When the set-up is executed, the files are unpacked and saved on the specified path.

Executing the set-up

To install the cycles, execute the file *isg-cnc-cycles-setup.exe* contained in the cycle packet.

When the cycle set-up is executed, the CNC cycles and the example call programs are unpacked and saved on the specified path.

To execute the cycles, the specified path must be saved in the controller as the subroutine path.



Attention

When the cycle set-up is executed the contents of the previously installed cycle packet are overwritten.

Integrating error messages

The error files must be saved in the TwinCAT set-up to ensure that the correct error messages are output if CNC cycles are incorrectly executed.

The file *TcCncCycleErrors.xml* in the error directory (*error_files*) in the TwinCAT set-up directory (here *C:\TwinCat*) is saved on the following path in order to output error message in the HMI: *C:\TwinCAT\3.1\Target\Resource*.

The file *err_text_cycles_eng.txt* (or *err_text_cycles.txt*) in the error directory (*error_files*) in the TwinCAT set-up directory (here *C:\TwinCat*) is saved on the following path in order to output error messages correctly in the log file: *C:\TwinCAT\3.1\Components\Mc\CNC\Diagnostics*.

Alternatively, the name of the error message file can also be modified by the channel parameter P-STUP-00200.

2 Overview

Task

The “Kinematic Optimisation” cycle (KinOpt) allows an increase in machining accuracy on 5-axis machine tools.

Geometric inaccuracies in machine kinematics are determined by a measurement run. This is followed by a calculation program to define optimised kinematic parameters which can then be entered in the machine configuration.

Effectiveness

The kinematic parameters are improved compared to the drawing dimensions and this leads to an enhanced precision of the workpiece.

Condition

The following is required for the measurement run:

- an absolutely accurate XYZ probe or
- a calibration sphere or
- alternative measuring equipment



Notice

Cycles are additional options and subject to the purchase of a license.



Notice

This additional option is available as of CNC Build V3.1.3068.09..

Restrictions

The method is restricted to the following kinematics:

- KIN_TYP_9: Cartesian CA head
- KIN_TYP_57: BC table
- KIN_TYP_58: AC table
- KIN_TYP_59: Cardanic CA head
- KIN_TYP_60: Cardanic CB head
- KIN_TYP_80: AB table
- KIN_TYP_90: Cartesian AB and CA head, cardanic CA head



Notice

Other kinematics

A number of other kinematics are special cases of the above listed kinematics. For example, the Cartesian CA head only supports the KIN_TYP_9. However, machines which were previously operated as KIN_TYP_10, KIN_TYP_25 or KIN_TYP_28 can also be parameterised as KIN_TYP_9.

Links

Mandatory note on references to other documents

For the sake of clarity, links to other documents and parameters are abbreviated, e.g. [PROG] for the Programming Manual or P-AXIS-00001 for an axis parameter.

For technical reasons, these links only function in the Online Help (HTML5, CHM) but not in pdf files since pdfs do not support cross-linking.

3 Description

3.1 Cycle-specific tool data

Assigning free tool parameters P-TOOL-00007

The free parameters available in the tool data are used in cycles

- to supply additional information,
- execute specific plausibility checks and
- define default values.

Up to 60 free parameters can be defined in the tool data for each tool. A tool parameter is only evaluated and used by V.G.WZ_AKT.P[i] in each cycle if it is entered with a value unequal to 0. Otherwise, either the plausibility check is not executed or any programmed input parameter is invalid.

The following rule applies when assigning currently available cycles:

Tool parameters	Meaning	Unit or value	Effect/special feature	Assignment
[0]	Useful tool length	0.1µm	Error message if final machining depth is greater than parameter value.	optional
[1]	Maximum machining stroke	0.1µm	Warning if feed depth is greater than parameter value. Feed depth is limited to parameter value.	optional
[2]	feed	1 µm/s	Currently not used.	-
[3]	Spindle speed	10 ⁻³ °/s	Only for thread tapping cycle	Optional for thread tapping drill
[4]	Spindle start position	10 ⁻⁴ °	Only for thread tapping cycle	Optional for thread tapping drill
[5]	Pitch	0.1µm	Only for thread tapping cycle	Optional for thread tapping drill
[6]	Identifier for special tool	0, 1, ...	0 = Standard tool 1 = Thread tapping drill with countersink 2 = ...	Obligatory for special tools
[7]	Metric thread size	1...16	Only for thread tapping cycle, Example 6 for M6, ... 10 for M10, etc.	Obligatory for thread tapping drill with countersink Example 6 for M6, ... 10 for M10, etc.
[8]	Overlapping factor	%	Only for pocket milling cycle	optional

3.2 Preparations

3.2.1 CNC controller

Configuring the kinematics



Attention

Collision risk

The measurement run is operated with a selected kinematic. The kinematic must therefore be configured using the drawing dimensions of the machine (or with sufficiently accurate values). The aim of the KinOpt cycle is to supply compensation values for the kinematic parameters.

Naming the axes

The axes must be named as follows in the channel in which the measurement run is operated.

- **X** for the X axis
- **Y** for the Y axis
- **Z** for the Z axis
- **A** for the (optional) rotary axis about **X**
- **B** for the (optional) rotary axis about **Y**
- **C** for the (optional) rotary axis about **Z**



Notice

Naming the axes

For example, the axes can be named using the #SET AX command or by [channel_parameter] (node:SET AX command or by channel parameters).

Supplying the cycles

First execute the installation program for the CNC cycles.

3.2.2 Machine controller

Probe and calibration sphere

The operator carries out the following steps:

- Install calibration sphere on machine table.
- Move rotary axes to 0.
- Clamp probe in tool holder; enter probe tool length.
- Use manual block or JOG to position the probe approximately centrally and at a clearance of several centimetres above the sphere.



Notice

Probe and sphere can be swapped. The layout is irrelevant for the cycle. The documentation only describes the case described above.

3.3 Measurement and calculation cycle

An example program for parameterising the cycle is available for automatic measurement with "SysCalibKinematicDemo.nc".

Appropriate changes for the kinematics to be measured and the measuring equipment used must be made in this program. The parameters to be modified are described in detail below.

3.3.1 General parameters

General parameters include:

- Geometry of the probe and calibration sphere
- Safety distances for pre-positioning and measurement run
- Velocities for pre-positioning and measurement run
- Kinematics used
- Path to save the measurement results in "message.txt" on the default path

Geometry	Unit	Meaning
V.P.RA-DIUS_TOOL	mm	Radius of probe tip (clamped in the spindle)
V.P.SAFETY_POS	mm	Safety distance about the calibration sphere during pre-positioning
V.P.SAFETY_DISTANCE_MEAS	mm	Safety distance around the calibration sphere during measurement Must be large enough to reach measurement speed.
V.P.RA-DIUS_WORK-PIECE	mm	Radius of the calibration sphere (placed on bench as workpiece)
V.P.MEASURING_FEED	mm/min	Velocity for measuring (G100 blocks)
V.P.POSITIONING_FEED	mm/min	Velocity for positioning
V.P.KIN_TYP	-	Kinematic of the machine
V.P.KIN_TYP_VARIANT	-	Variant of the kinematic if, for example, several kinematics can be grouped by one number, as with kinematic 90
#FILE NAME [MSG = "..."]	String	Path to save measurement data
V.L.PRINT (optional)	-	Output of a log in the parameterised file, 0 = no log, 1 = German, 2 = English
V.P.MAX_PRECISION (optional)	mm	Maximum permitted deviation of measurement errors
V.P.LIST_FORMAT	-	Output formats: 0 = XML, 1 = TwinCAT 2, 2 = TwinCAT 3

The radii of the probe tip and calibration sphere only serve to calculate the motion paths and are not used to calculate offsets. It is therefore sufficient to enter approximations for these radii. The meaning of the geometry parameters is described in the figure below.

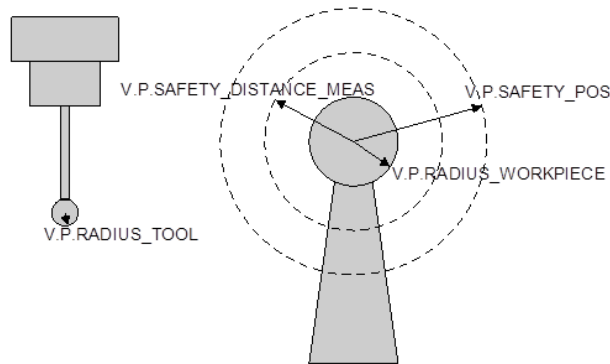


Fig. 1: Meaning of geometry parameters

The first measurement movement is in the negative Z direction. The measurement path is determined by the parameters entered by the operator and is incremental from the starting point:

$$Z = -(V.P.RADIUS_TOOL + V.P.SAFETY_DISTANCE_POS + 2 * V.P.RADIUS_WORKPIECE)$$



Example

Probe radius (as tool) **V.P.RADIUS_TOOL = 3**
Positioning safety distance:
V.P.SAFETY_DISTANCE_POS = 20
Calibration sphere radius (as workpiece):
V.P.RADIUS_WORKPIECE = 15

The resulting NC block is:

```
G100 G91 Z-[3+20+2*15]
```

A distance of -53 mm is traversed in Z, then the distance between the sphere and the probe at the start must be smaller than 53mm.



Example

The safety distance must be large enough to reach the correct measuring velocity. Otherwise, deviations may occur due to the velocity-dependent switching signal.

3.3.2

Setting the measurement poses

The measurement poses are positions of the two rotary axes in which the calibration sphere is measured. There are 2 methods to set the poses:

- Measurement positions are generated by rotating an axis with the second axis held fast.
- Pose to be measured directly across all axis angles.

Enter with 2nd axis held fast

In this case, the individual poses are defined as an array. This must be executed for every rotary axis. One axis is always held fast (V.P.POS_2ND_AXIS[i]) and the second axis is measured at a number of positions between the start and end angles as defined by V.P.NUMBER_SPHERE_MEAS[i]. If the difference between the start and end angles is greater than:

$$[360/V.P.NUMBER_SPHERE_MEAS[i]] * [V.P.NUMBER_SPHERE_MEAS[i] - 1]$$

...the calculated angle is used as the end angle.

Example:

- Start angle = 0
- End angle = 360
- Number of poses = 4
- $[360/4] * 3 = 270$

Therefore, the poses 0, 90, 180 and 270 are measured.

Parameter	Unit	Meaning
V.P.NUMBER_MEAS_PROC	-	Number of measurement processes (arrays)
V.P.MEAS_AXIS[i]	-	Axis to be measured; A=1, B=2, C=3
V.P.NUMBER_SPHERE_MEAS[i]	-	Number of measurement poses
V.P.START_ANGLE_MEAS[i]	°, degrees	Starting position of the moved axis
V.P.END_ANGLE_MEAS[i]	°, degrees	End position of the moved axis
V.P.POS_2ND_AXIS[i]	°, degrees	Position of the second rotary axis



Programing Example

Number of measurements of single rotary axes

The programming example below shows the parameters required to set the measurement poses.

```
#VAR
V.P.NUMBER_MEAS_PROC
#ENDVAR
V.P.NUMBER_MEAS_PROC = 2
#VAR
V.P.MEAS_AXIS[V.P.NUMBER_MEAS_PROC]
V.P.NUMBER_SPHERE_MEAS[V.P.NUMBER_MEAS_PROC]
V.P.START_ANGLE_MEAS[V.P.NUMBER_MEAS_PROC]
V.P.END_ANGLE_MEAS[V.P.NUMBER_MEAS_PROC]
V.P.POS_2ND_AXIS[V.P.NUMBER_MEAS_PROC]
#ENDVAR
(First measurement, here A axis)
(A axis is measured in positions A-45, A0 and A45)
(C axis stands at 0)
V.P.MEAS_AXIS[0] = 1 (axis: 1: A; 2: B; 3: C - depends)
(on kinematic and variant)
V.P.NUMBER_SPHERE_MEAS[0] = 3 (number of poses to be measured)
V.P.START_ANGLE_MEAS[0] = -45 (starting angle [degrees])
V.P.END_ANGLE_MEAS[0] = 45 (end angle [degrees])
V.P.POS_2ND_AXIS[0] = 0 (position of second axis)
(during the measurement)
(Second measurement, here C axis)
(The C axis is measured in positions C0, C90, C180 and C270)
(. The A axis stands at A35)
V.P.MEAS_AXIS[1] = 3
V.P.NUMBER_SPHERE_MEAS[1] = 4
V.P.START_ANGLE_MEAS[1] = 0
V.P.END_ANGLE_MEAS[1] = 360
V.P.POS_2ND_AXIS[1] = 35
```

Direct input of poses

If single poses are entered directly, an array `V.P.MEAS_POSE[3][Number of measurement poses]` must first be defined. It consists of the 3 rotation directions and the number of poses to be measured. The angles of the non-existing rotary axis must be initialised with 0.

The rotary axes are positioned in the sequence in which they were parameterised in the array.



Programing Example

Defining the array

The next programming example describes the parameterisation of 9 poses for a BC kinematic.

```
#VAR
V.P.MEAS_POSE[3][9]
#ENDVAR
(A axis B axis C axis)
V.P.MEAS_POSE[0][0] = 0      V.P.MEAS_POSE[1][0] = 0
V.P.MEAS_POSE[0][1] = 0      V.P.MEAS_POSE[1][1] = 0
V.P.MEAS_POSE[0][2] = 0      V.P.MEAS_POSE[1][2] = 0
V.P.MEAS_POSE[0][3] = 0      V.P.MEAS_POSE[1][3] = 0
V.P.MEAS_POSE[0][4] = 0      V.P.MEAS_POSE[1][4] = 90
V.P.MEAS_POSE[0][5] = 0      V.P.MEAS_POSE[1][5] = 180
V.P.MEAS_POSE[0][6] = 0      V.P.MEAS_POSE[1][6] = 270
V.P.MEAS_POSE[0][7] = 0      V.P.MEAS_POSE[1][7] = 45
V.P.MEAS_POSE[0][8] = 0      V.P.MEAS_POSE[1][8] = 84

V.P.MEAS_POSE[2][0] = -45
V.P.MEAS_POSE[2][1] = 0
V.P.MEAS_POSE[2][2] = 45
V.P.MEAS_POSE[2][3] = 35
V.P.MEAS_POSE[2][4] = 35
V.P.MEAS_POSE[2][5] = 35
V.P.MEAS_POSE[2][6] = 35
V.P.MEAS_POSE[2][7] = 40
V.P.MEAS_POSE[2][8] = 40
```

Minimum number of poses

A minimum number of poses to be approached is required to define the kinematic parameters correctly. This minimum number depends on the kinematic and is specified in the table below:

Kinematic and variant	Structure	Minimum number of poses
Kinematic 9	CA head kinematic (Cartesian)	4 measurement poses
Kinematic 57	BC table kinematic (Cartesian)	5 measurement poses
Kinematic 58	AC table kinematic (Cartesian)	5 measurement poses
Kinematic 59	CA head kinematic (card- anic)	5 measurement poses
Kinematic 60	CB head kinematic (card- anic)	5 measurement poses

Kinematic 80	AB table kinematic (Cartesian)	A and B must each have at least 2 different values
Kinematic 90, variant 0	AB head kinematic (Cartesian)	4 measurement poses
Kinematic 90, variant 2	CA head kinematic (Cartesian)	4 measurement poses
Kinematic 90, variant 8	CA head kinematic (card- anic)	5 measurement poses

This minimum number must be maintained both for automatic and manual measurements.

However, there are also poses which supply no further information about the kinematic.

In addition, the effect of measurement errors on the calculation results is reduced if more poses are measured than required. Therefore, we recommend measuring more poses than the specified minimum number.,

Moreover, the measured poses should be evenly distributed along the entire motion range of the rotary axes.



Notice

Pose selection

In general, possible measurement errors can be averaged out more easily with a large number of poses. This reduces the risk of selecting poses which supply no further information and then lead to a measurement run without a result.

Make sure that the measurement poses cover the largest possible swivel range of the rotary axes.

3.4 Kinematic-related arrangement of axes

The arrangement and naming of the measurement axes is dependent on the kinematic and may have to be generated by the command #SET AX.

In the example below:

- #SET AX commands are specified for the supported kinematics
- the logical axis number (2nd argument in brackets) must be adapted
- a 6th axis exists for kinematic 90



Programing Example

Kinematic-related arrangement of axes

```
(KIN_TYP_9)
#SET AX[X,1,0][Y,2,1][Z,3,2][C,5,3][A,4,4]
(KIN_TYP_57)
#SET AX[X,1,0][Y,2,1][Z,3,2][B,8,3][C,5,4]
(KIN_TYP_58)
#SET AX[X,1,0][Y,2,1][Z,3,2][A,4,3][C,5,4]
(KIN_TYP_59)
#SET AX[X,1,0][Y,2,1][Z,3,2][C,5,3][A,4,4]
(KIN_TYP_60)
#SET AX[X,1,0][Y,2,1][Z,3,2][C,5,3][B,8,4]
(KIN_TYP_80)
#SET AX[X,1,0][Y,2,1][Z,3,2][A,4,3][B,8,4]
(KIN_TYP_90, Variant 0)
#SET AX[X,1,0][Y,2,1][Z,3,2][A,4,3][B,8,4][C,5,5]
(KIN_TYP_90, Variant 2)
#SET AX[X,1,0][Y,2,1][Z,3,2][C,5,3][A,4,4][B,8,5]
(KIN_TYP_90, Variant 8)
#SET AX[X,1,0][Y,2,1][Z,3,2][C,5,3][A,4,4][B,8,5]
```

4 Measuring the machine

The machine can be measured in various ways:

- With the measurement subroutines provided
- External measurement: Manual measurement with calibration sphere
- External measurement: Measuring the TCP (tool centre point)

4.1 General sequence of measurement run

Measuring for pre-positioning

- adjust kinematic offsets and tool length (drawing dimensions)
- the rotary axes are in zero position
- the probe is roughly positioned above the sphere centre point
- the sphere is scanned in the X, Y and Z directions and its centre point is determined

Measuring the compensation values

This step can be executed either by the supplied measurement subroutines or manually.

- the configured poses are approached
- scan the sphere in each of the poses and measure their centre point

Calculating the compensation values

- Call the calculation subroutine. Use the centre points of the sphere measured in the previous step or the TCP positions defined by external measuring equipment
- The calculation may take several minutes
- The results are output in list format

Checking by the operator

- Operator copies **compensation values** to the channel parameter list



Notice

If the measurement is executed automatically using the measurement subroutines provided, no further settings need be made for the evaluation. If this is the case, you can skip the following sections up to “Checking” and “Evaluation”.

4.2

Measuring with measurement subroutines

Secure a calibration sphere to the machining table and clamp a probe in the tool holder. It is also possible to secure the calibration sphere in the spindle and the probe to the table.

Before starting the calibration program, position the probe roughly above the centre point of the calibration sphere. The rotary axes must be in zero position. If the calibration sphere is at a known position, it can be moved towards it from the measurement program.

The kinematic offsets and tool length must be approximately correct (drawing dimensions) since the individual poses are approached using the kinematic transformation.

4.3 External measurement

It is also possible to use externally determined measurement values to calculate the offsets. In this case,

- either a sphere can be measured or
- the TCP can be directly measured if there is a suitable measuring instrument.

4.3.1 Manual measurement with calibration sphere

Apart from the calibration sphere, a 3D dial gauge, a probe or another suitable measuring instrument are required.

The operator swivels the two rotary axes to specific values. Then carefully approach the calibration sphere with the dial gauge manually or with the probe. In other words, the dial gauge approaches the calibration sphere until the gauge indicates a definite deflection. This deflection should be as small as possible (a few 1/100mm) but it should still be large enough so that it is definitely recognisable as a deflection.



Attention

When approaching the calibration sphere, it is important that only linear axes X, Y and Z are moved, not the rotary axes. In addition, make sure that the kinematic transformation is not active (#TRAFO OFF).

If you use a probe, make sure that the scanning movement is perpendicular to the surface to avoid distortions. When the scanning event is recognised, the operator notes down the axis values of X, Y and Z for the related pose.

Repeat the scanning procedure for other points on the sphere surface and in the same pose of the rotary axes. Between 6 and 10 points should be scanned per pose. They should be evenly distributed over a large surface area which is greater than (half) the surface of the calibration sphere. One example (top view) for possible scanning points:

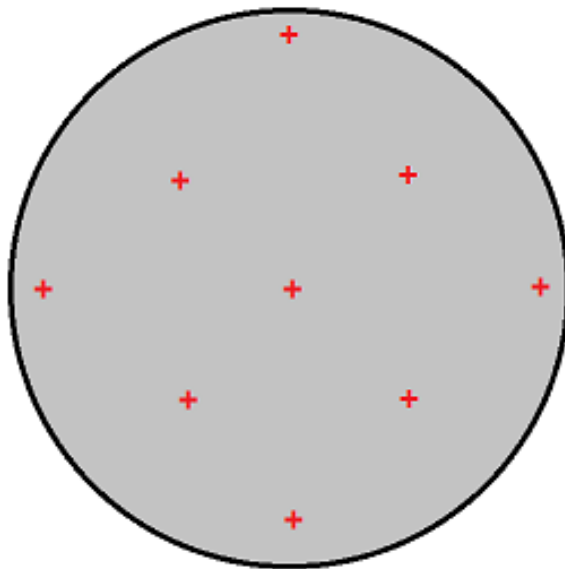


Fig. 2: Example of distribution of measurement points on a hemisphere

In addition, during the scan, make sure that the probe moves as far as possible in the direction of the sphere centre. This is especially possible at points “near the equator” and at the “north pole” since pure X/Y and Z movements are possible in manual mode at those points.

When the measurement values are recorded for a pose, a log should be obtained with the following structure (axis values in mm or degree; use a time stamp).



Example

Measurement values recorded manually

```
X = 103.304 Y = 255.257 Z = 200.000
X = 113.422 Y = 243.257 Z = 200.000
X = 102.089 Y = 233.421 Z = 200.000
X = 87.696 Y = 245.920 Z = 188.835
X = 93.027 Y = 255.847 Z = 188.835
X = 102.861 Y = 258.935 Z = 186.585
```

Repeat this method for other poses. During the process, do not remove the calibration sphere from the table or reposition it. Select the poses from the following aspects:

- The rotary axes must assume several different values. Use the above mentioned minimum number of poses for the supported kinematics.
- The test set-up and the machine construction influence the selection poses due to the restricted range of motion (collision risk).
- A higher number of poses makes it easier to obtain better averaging of measurement errors.

4.3.2

Measuring the TCP (tool centre point)

If there is a measuring instrument for the direct recording of the TCP position, e.g.

- a nest in which a sphere can be immersed, or
- a high-precision camera system etc.

this position can be transferred to the calculation algorithm together with the associated pose.

5 Preparing the calculation cycle

There is a programming example to execute the calculation for every available kinematic without automatic measurement. They are:

- SysCalibFit009V0Demo.nc
- SysCalibFit057V0Demo.nc
- SysCalibFit058V0Demo.nc
- SysCalibFit059V0Demo.nc
- SysCalibFit060V0Demo.nc
- SysCalibFit080V0Demo.nc
- SysCalibFit090V0Demo.nc
- SysCalibFit090V2Demo.nc

5.1 Adapting the output file

The path and name of the output file can be adapted here.



Programing Example

Defining the filename of the output file

```
#FILE NAME [MSG = "calibration_result.txt"]
```

Replace the string "calibration_result.txt" by a filename of your choice.

5.2 Adapting the output format

The list format with the output results can be parameterised with the variable V.P.LIST_FORMAT.



Programing Example

Setting the output format

```
; Format of the output: 0 *.xml (default), 1 TC2, 2 TC3  
V.P.LIST_FORMAT = 1 ( format of the list containing the results )
```

5.3 Adapting the limit for deviation

The variable V.P.MAX_PRECISION specifies the maximum permitted deviation of the kinematic measured.



Programing Example

Setting the maximum error

```
; maximum permitted deviation in mm
V.P.MAX_PRECISION = 0.06
```

This value is the upper limit for the permitted deviation of the calculated solution of an ideal kinematic.

If this deviation is exceeded during calibration (measurement errors, errors in the linear axes, position-dependent errors of the rotary axes), an error message P-ERR-13346 is output. This value must be determined by trial and error.

5.4 Initialising the kinematic calculation

The following parameters must be transferred to the calculation cycle "SysCalibFitInit.ecy":

- @P1: the kinematic used
- @P2: the variant of the kinematic 0 is a kinematic without variant
- @P3: the number of poses measured



Programing Example

Calling the calculation cycle

Number of poses, e.g. number of [A, B] combinations

```
#VAR
V.L.KIN_TYP = 80 ( kinematic measured )
V.L.KIN_VARIANT = 0 ( variant of kinematic measured )
V.L.NUMBER_OF_POSES = 8 ( number of measurement poses )
#ENDVAR
L CYCLE [NAME = "SysCalibFitInit.ecy"      \
      @P1 = V.L.KIN_TYP                    \
      @P2 = V.L.KIN_VARIANT                \
      @P3 = V.L.NUMBER_OF_POSES            \
      ]
```

5.5 Entering measurement results

After initialisation, the calculation is executed with the data entered.

5.5.1 Direct input of measurement data

Enter the measurement results of linear axes X, Y and Z (in mm) for each pose. In addition, specify the related angles of the rotary axes. In the example below, an AB table kinematic was measured in pose A=20 and B=0 degrees. Although there is no C axis in the example, the C position must be transferred with 0.



Programing Example

Example of the direct input of measured data

```
; Defining variables
#VAR
V.L.N = 6          ( number of points measured )
V.L.X[V.L.N]      ( X coordinate of points measured )
V.L.Y[V.L.N]      ( Y coordinate of points measured )
V.L.Z[V.L.N]      ( Z coordinate of points measured )
V.L.RESULT_V[3]   ( array to save calculation results )
V.L.A = 20        ( position of A axis )
V.L.B = 0         ( position of B axis )
V.L.C = 0         ( position of C axis )
#ENDVAR

; List of XYZ coordinates of all points measured in this pose, in mm
V.L.X[0] = 101.807  V.L.Y[0] = 273.174  V.L.Z[0] = 130.068
V.L.X[1] = 116.795  V.L.Y[1] = 258.750  V.L.Z[1] = 130.068
V.L.X[2] = 104.246  V.L.Y[2] = 244.524  V.L.Z[2] = 129.024
V.L.X[3] = 89.606   V.L.Y[3] = 257.804  V.L.Z[3] = 132.980
V.L.X[4] = 96.606   V.L.Y[4] = 269.804  V.L.Z[4] = 133.957
V.L.X[5] = 102.780  V.L.Y[5] = 264.366  V.L.Z[5] = 139.522

; Call cycle to calculate sphere centre point
L CYCLE [NAME = "SysCalcSphere.ecy"]

; Add new coordinates to calculate kinematic
L CYCLE [NAME = "SysCalibFitAddRecord.ecy" \
      @P1 = V.L.RESULT_V[0] \
      @P2 = V.L.RESULT_V[1] \
      @P3 = V.L.RESULT_V[2] \
      @P4 = V.L.A \
      @P5 = V.L.B \
      @P6 = V.L.C \
      ]

; ... Executing input for all poses
```

This step must be executed for every pose measured. The calculation process can then be started from the same NC program as the values entered.



Programing Example

Calling the calculation cycle

```
V.P.MAX_PRECISION = 0.06  (precision)
V.P.LIST_FORMAT   = 1     ( list format for result )
#ENDVAR
```



```
; Call calculation cycle
L CYCLE [NAME = "SysCalibFit.ecy"      \
      @P1 = V.L.LIST                  \
      @P2 = V.L.MAX_PRECISION        \
      ]

; Call to end cycle
L CYCLE [NAME = "SysCalibFitFinalize.ecy"]
```

5.5.2 Sphere centre points already calculated

If the sphere centre points are already known, as shown in the example if a measurement was executed using measuring equipment from third-party providers, they can be used directly for calculation.



Programing Example

Entering sphere centre points for various poses

Initialisation: Number of poses, e.g. number of A,B combinations

```
#VAR
V.L.KIN_TYP = 80          (kinematic measured)
V.L.KIN_VARIANT = 0      (variant of kinematic measured)
V.L.NUMBER_OF_POSES = 8  (number of measurement poses)
#ENDVAR

L CYCLE [NAME = "SysCalibFitInit.ecy" \
        @P1 = V.L.KIN_TYP           \
        @P2 = V.L.KIN_VARIANT       \
        @P3 = V.L.NUMBER_OF_POSES   \
        ]

; Call cycle to add new measurement data
L CYCLE [NAME = "SysCalibFitAddRecord.ecy" \
        @P1 = 102.545449 \ (X coordinate of centre point)
        @P2 = 258.949268 \ (Y coordinate of centre point)
        @P3 = 125.6186118 \ (Z coordinate of centre point)
        @P4 = 20          \ (associated A coordinate)
        @P5 = 0           \ (associated B coordinate)
        @P6 = 0           \ (associated C coordinate)
        ]

; ... Executing input for all poses
; Call fitting cycle
#VAR
V.P.MAX_PRECISION = 0.06 (precision)
V.P.LIST_FORMAT = 1      (list format for result)
#ENDVAR
; Call fitting cycle
L CYCLE [NAME = "SysCalibFit.ecy" \
        @P1 = V.P.LIST_FORMAT \
        @P2 = V.P.MAX_PRECISION \
        ]

; Call to end cycle
L CYCLE [NAME = "SysCalibFitFinalize.ecy"]
```

5.6 Kinematic-specific changes

The kinematic offsets applied here are described in detail in the documentation for Kinematic Transformation KITRA for each of the kinematics.

5.6.1 KIN_TYP_80

Direction of rotation of A axis

Kinematic 80 is used to invert the direction of rotation of the A axis. For these cases, set the kinematic parameter 14 (HD15) to 1. On the other hand, the default value 0 characterises a mathematically positive rotating A axis.



Programing Example

Adapting the rotation direction of the A axis

```
; 0 = positive direction
; 1 = negative direction
; NOTE: in some cases, the previous value of
; V.G.KIN[80].PARAM[14] may have to be restored later.
V.G.KIN[80].PARAM[14] = 1
```



Notice

This may overwrite a globally known V.G. variable. If required, the operator will have to look after restoring the original value later.

Specifying the vertical offset value:

The test set-up described may not be capable of detecting a specific geometry value of the kinematic. This refers to the vertical offset value HD12 + HD6 + HD9 which must be determined by a measurement which is carried out beforehand.

This value is then entered in the main calculation program as follows:



Programing Example

Entering the vertical offset value

```
; The value of HD12 + HD6 + HD9, in mm
V.L.HD12_HD6_HD9 = -3.135
```

5.6.2 KIN_TYP_90

Set the axis programming mode (kinematik[90].param[13]) internally to the default setting of the kinematic variant selected.

6 Executing the calculation cycle

6.1 Starting the main program

After completing these preparations, execute the modified main program:

- SysCalibManual009V0.nc
- SysCalibManual057V0.nc
- SysCalibManual058V0.nc
- SysCalibManual059V0.nc
- SysCalibManual060V0.nc
- SysCalibManual080V0.nc
- SysCalibManual090V0.nc
- SysCalibManual090V2.nc

Since this involves extensive calculations, the runtime may take several minutes. The runtime is dependent on:

- the number of poses measured
- the cycle time of the SDA task and
- system performance

The program generates no path motions.

After the calculation ends, the results must be checked for plausibility and the spindle flange offset in the tool holder must be determined (normally HD1).

7 Checking the calibration

Both the automatic measurement and the calculation output a large amount of information that can be used to check the measurement and calculation results.

7.1 Plausibility check of the output

After a successful run, the program outputs a number of information items:

- estimates for single poses both with automatic and manual measurements
- In addition, the single positions measured for an automatic measurement are output.

For example, the block with the calculation results for the 1st pose is:



Example

Output of the calculation results of a sphere

```
% Centre of sphere [X, Y, Z] = [102.5305658, 244.3467991, 189.8131021]
% Radius R      = 14.94823794
% Variance      = 9.56062833e-006
% Residual[0]   = 0.00154694034
% Residual[1]   = -0.004483877805
% Residual[2]   = 0.003641747138
% Residual[3]   = -0.001544225933
% Residual[4]   = -0.002650543572
% Residual[5]   = 0.003491883191
```

"Centre of sphere" is the centre of the calibration sphere in XYZ coordinates for the pose specified. Radius "R" specifies the radius of the calibration sphere (more exactly: the sum of the radii of the calibration sphere and the dial gauge tip or switching point of the probe tip).

Criterion (A) – radii of the calibration spheres

The calibration sphere radii measured across all poses should be constant except for the measurement precision. If this is not the case, the calibration sphere was not correctly measured in one or several poses. Repeat the measurement for these poses.

Criterion (B) – Residuals

The residuals (Residual[i]) should be within the range of a few 1/100mm. They indicate the amount by which each of the measurement points deviate from the calibration sphere surface. If a measurement point deviates strongly from the sphere surface, reject the affected pose and repeat this measurement.

Calculation output

The 2nd part of the output contains the calculated **Correction values for kinematics P-CHAN-00438 and several additional items of information. This block looks like this, for example for kinematic 80:



Example

Output of calculated kinematic parameters

```
#timestamp
```

```
12.12.2016 12:58:17:000
```

```
#kinematic id                        80
#kinematic id variant                0
#number of measurement records      8
#
#record 0 = [102.5305658,244.3467991,189.8131021, 0,0,0]
#record 1 = [102.545449, 258.949268, 125.6186118, 20,0,0]
#record 2 = [29.18053549,233.8711714,175.957552, -20,-90,0]
#record 3 = [176.2562104,233.7423534,176.3588772, -20.90,0]
#record 4 = [29.20839651,244.3560828,116.1031577, 0,-90.0]
#record 5 = [176.2877121,244.3573191,116.5414503, 0.90,0]
#record 6 = [176.2795252,233.8784364,56.68825135, 20.90,0]
#record 7 = [29.21147408,233.7284348,56.33260263, 20,-90.0]
#
#solution precision                  0.04409219769
#probe position PCS, [mm]           [-0.2058572907, 0, 73.53215927]
#
kinematik[80].corr[1]               E39A4FA5FA49658AFD79AE9E
kinematik[80].corr[2]               E39A4FA5FA49658AFD79AE9E
kinematik[80].corr[3]               E39A4FA5FA49658AFD79AE9E
kinematik[80].corr[4]               E39A4FA5FA49658AFD79AE9E
kinematik[80].corr[5]               6E9B4FA534A809BEB3FDC8FB
kinematik[80].corr[6]               E39A4FA5FA49658AFD79AE9E
kinematik[80].corr[7]               E9954FA50E96594B756EB4F7A79567C2084AC996
kinematik[80].corr[8]               E39A4FA5FA49658AFD79AE9E
kinematik[80].corr[9]               A18A4FA512510AB45F3259BCA79567C2084AC996
kinematik[80].corr[10]              60914FA56EE1CFF3A6AA2953
kinematik[80].corr[11]              AD924FA53196E9D01A53CB3D
```

Besides the time stamp, the kinematic ID and the number of poses, the solution precision is also output, among other items. This value describes in mm the largest deviation between model and measurement values.

The deviation is produced by measurement errors or by errors not mapped in the model, e.g. axis misalignments, etc.



Notice

The correction parameters are output in encrypted form (P-CHAN-00438) and automatically decrypted later when they are read.

Criterion (C) – solution precision

The solution precision should not exceed a certain limit. A limit for this value can be specified in the main program. If this limit is exceeded, an error is output and the measurement equipment, measurement method etc. must be checked.

- **Measurement equipment:** The measurement equipment must measure the shape of the calibration sphere correctly from all directions. The measurement precision obtained must be as small as possible.
- **Measurement method:** The measurement must be taken as perpendicularly to the surface of the calibration sphere as possible to avoid distortion of the contour measured. To ensure that this occurs with the automatic measurement, kinematic parameters which roughly describe the kinematic should be entered previously.
- **Machine errors:** Rotary axis zero positions, the rectangularity of linear axes, the positioning and repeatable precision of all axes and position-dependent errors of the rotary axes

Criterion (D) – kinematic parameters calculated

If the check is successful, the above mentioned block can be copied to the channel parameter list. After the latest import of the channel parameters, the compensation values are available to the kinematic.

**Attention**

Then a careful, slow test run should be executed with the kinematic enabled to check whether the compensation values have the correct effectiveness. This also checks that the transfer of compensation values is error-free and that no unintended collisions occur.

7.2 Determining missing offsets

The measurement method is not capable of determining the offset of the spindle flange in the tool direction (normally HD1). It can only determine HD1 + tool length of the dial gauge or probe for a head kinematic. However, even this is not possible with a table kinematic. For this reason, HD1 must be determined by an alternative method and entered in the channel parameter list.

Due to the lack of kinematic separation, not all offsets can be determined for many kinematics.

7.2.1 KIN_TYP_80

Y offsets are freely movable and are defined by the position of the calibration sphere or probe on the table.

All Y offsets are combined in parameter HD8 in the calculation results.

7.3 Machining test part

The final check of the calibration can be carried out by machining a suitable test part, for example.

- the geometry of the test part should be as simple as possible
- the rotary axes should be used to machine the test part
- the rotary axes should be used independently of each other as far as possible

In the test part, contour parts which were manufactured in various poses with an active transformation should be concatenated without interruption.

The figure shows an example of a geometry to check the result of calibration for a head kinematic – CA or CB. 2 pockets are milled:

- one in X direction
- one in Y direction

Each pocket is machined half under C0 and the other half under C180. The offsets between the C0 and C180 milling operations should disappear if the calculation of the offsets is successful (red).

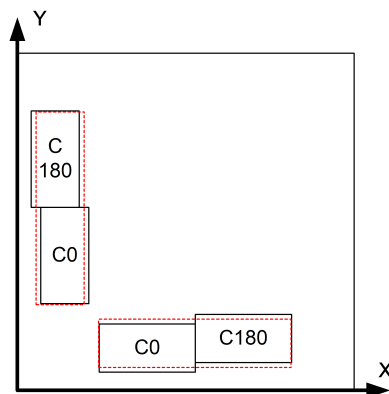


Fig. 3: Example of a test part – A or B in zero position

The offsets must disappear between the pockets in X and Y directions. The pockets without calibration are black and with calibration they are red.

8 Appendix

8.1 Suggestions, corrections and the latest documentation

Did you find any errors? Do you have any suggestions or constructive criticism? Then please contact us at documentation@isg-stuttgart.de. The latest documentation is posted in our Online Help (DE/EN):



QR code link: <https://www.isg-stuttgart.de/documentation-kernel/>

The link above forwards you to:

<https://www.isg-stuttgart.de/fileadmin/kernel/kernel-html/index.html>



Notice

Change options for favourite links in your browser;

Technical changes to the website layout concerning folder paths or a change in the HTML framework and therefore the link structure cannot be excluded.

We recommend you to save the above "QR code link" as your primary favourite link.

PDFs for download:

DE:

<https://www.isg-stuttgart.de/produkte/softwareprodukte/isg-kernel/dokumente-und-downloads>

EN:

<https://www.isg-stuttgart.de/en/products/softwareproducts/isg-kernel/documents-and-downloads>

E-Mail: documentation@isg-stuttgart.de



© Copyright
ISG Industrielle Steuerungstechnik GmbH
STEP, Gropiusplatz 10
D-70563 Stuttgart
All rights reserved
www.isg-stuttgart.de
support@isg-stuttgart.de

