



# DOCUMENTATION ISG-kernel

## **Functional description Volumetric compensation**

Short Description:  
FCT-C26

# 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.

# Table of contents

<b>Preface.....</b>	<b>2</b>
<b>General and safety instructions .....</b>	<b>3</b>
<b>1 Overview.....</b>	<b>6</b>
<b>2 Description .....</b>	<b>8</b>
<b>3 Configuration .....</b>	<b>9</b>
3.1 General configuration (obligatory).....	9
3.2 Tool length (optional) .....	12
3.3 Modification of compensation values (optional) .....	13
3.4 Alternative model (optional) .....	13
3.5 Diagnosis (optional) .....	14
3.6 Linkage to channel (optional).....	14
3.7 Access to configuration and state data with V.G variables .....	15
<b>4 Programming .....</b>	<b>16</b>
4.1 Activation (#VOLCOMP ON/OFF).....	16
4.2 Behaviour on reset and in the event of errors.....	19
<b>5 Notes on measurement.....</b>	<b>20</b>
5.1 Pre-compensation .....	20
5.2 Influence of measuring strategy on configuration .....	20
5.2.1 Workpiece axes .....	20
5.2.2 Cartesian pre-compensation.....	21
5.2.3 Coordinate system offset, rotation centre .....	22
<b>6 Error parameters .....</b>	<b>23</b>
6.1 Parametric model according to DIN ISO 230 and ISO TR 16907 .....	23
6.2 List of parameters supported .....	25
6.3 File formats supported .....	26
6.3.1 Units of measure.....	26
6.3.2 ISG CSV format .....	27
6.3.3 Etalon Exchange Format .....	28
6.4 Interpolation of parameter values.....	30
6.4.1 Interpolation mode .....	30
6.4.2 Constancy in the case of modulo passages, parameter splicing .....	30
<b>7 Examples .....</b>	<b>32</b>
7.1 Cartesian compensation of XYZ kinematics .....	32
7.2 Compensation of head kinematics CYXZ, with Cartesian pre-compensation.....	33
<b>8 Appendix .....</b>	<b>35</b>
8.1 Suggestions, corrections and the latest documentation.....	35

## List of figures

Fig. 1:	for a 5-axis CB machine .....	11
Fig. 2:	5-axis machine XYZBC with rotary and swivel axis in the workpiece.....	21
Fig. 3:	Position errors of the C axis, without C0C.....	23
Fig. 4:	Component errors of the X axis.....	24

# 1 Overview

## Task

The term "volumetric compensation" denotes compensation of machine tools' geometric errors in the overall work volume. It is based on a parametric machine model that is standardised to the standards DIN ISO 230-1/7 "Test code for machine tools" and ISO/TR 16907 "Machine tools — Numerical compensation of geometric errors". Chapter "Notes on measurement and use [► 20]" provides a short overview of the model.

The purpose of compensation is to balance out errors in tool positioning and orientation by correcting the axis setpoints, thus enhancing the quality of the workpiece.



### Release Note

**This functionality has been available since CNC Build V3.00.xx.**



### Notice

**This function is an additional option requiring a license.**



### Notice

Volumetric Compensation only compensates for tool position. Errors resulting from incorrect positioning are compensated but not tool orientation.

## Properties

Volumetric Compensation can be used for all machines that can be described by serial kinematics with precisely 3 linear axes X, Y, Z and up to three rotary axes A, B, C.

Gantry combinations are supported (transparently for the user) by the fact that the slave axes are supplied with the same compensation values as their master axes.

## Programming

Volumetric Compensation can be activated and deactivated in the NC program by the **#VOL-COMP** command.

---

***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.

## 2 Description

### General approach

---

The first step towards use of Volumetric Compensation consists of measuring the machine to determine the parameters for the machine model. There are several vendors of appropriate hardware and software on the market. The ISG controller supports:

- ISG CSV format
- Etalon® Exchange Format (see Section “File formats supported [► 26]”).

After measurement, the parameters are available in one or more file(s). These parameter files can be loaded to the ISG kernel via a file interface. Volumetric Compensation is configured via the start-up list and the axis parameter lists.

Volumetric Compensation can be

- activated
- deactivated
- by a CNC programming command and the parameter files can be read in again.



## 3 Configuration

### 3.1 General configuration (obligatory)

Volumetric Compensation(s) takes place

- in the start-up list "hochlauf.lis" [STUP//Chapter Volumetric Compensation] and
- in the compensation-specific lists [VOLC].

Up to five compensations can be configured for each controller. Configuration of more than only one compensation makes sense on multi-channel machines, for example. Moreover, the measuring strategy can influence whether you have to work with one or more compensations; see Section "Influence of measuring strategy on configuration". [► 20]

#### Axis assignment

Volumetric Compensation requires the presence of

- 3 translatory axes and
- up to 3 rotary axes.

The ISO standard stipulates the letters X, Y, Z, A, B and C as axis names.

Which physical axis assumes the role of the X axis, the Y axis, etc. is assigned by the logical axis number.



#### Attention

This axis assignment must match the axis identifier when the machine is measured. Otherwise, the compensation values are assigned to the wrong axes and this results in an inexact TCP position during machining.



#### Programming Example

##### Axis assignment in compensation list

```
# Axis assignment for 0-th compensation, via logical axis number
```

```
X      1
Y      2
Z      3
C      7
A     19
```

#### Number of records

For memory allocation, a bound must be specified for the number of records to be read. An error is issued if this number is exceeded when the parameter files are read in.



## Programing Example

### Example – number of records in "hochlauf.lis"

```
# Upper bound for the number of data records
vol_comp[0].max_records    2000
```

## Parameter files

Up to 10 parameter files can be used for each compensation. A file name and the file format used must be specified for each parameter file.

The following names are available for the file format.

Name	File format
ISG_CSV	ISG CSV file format, see section “ISG CSV format [► 27]”
ETALON_EXCHANGE	Etalon Exchange Format, see section “Etalon Exchange Format [► 28]”



## Programing Example

### 2 parameter files in compensation list

```
# First parameter file, ISG CSV format
file[0]          C:\volcomp\datei01.csv
file_format[0]   ISG_CSV

# Second parameter file, Etalon exchange format
file[1]          C:\volcomp\datei02.exc
file_format[1]   ETALON_EXCHANGE
```

## Kinematic model

Further information is needed to completely describe the machine kinematic model. In particular, the sequence of axes in the kinematic chain must be specified.

Under certain circumstances, the number of axes located on the workpiece side must be specified. Whether this value must be specified depends on the measuring strategy, see section “Influence of the measurement strategy on configuration”. [► 20]



## Programing Example

### Kinematic model in compensation list

```
# Example: a 5-axis CB machine, table-table kinematics
# ID57 in the ISG library for Kinematic Transformations

# the sequence of the axes is CBYXZ
chain          CBYXZ

# (the first) two axes CA are located on the workpiece side
workpiece_axes 2
```



### Notice

Specifying the kinematic chain starts with the workpiece and then runs along the axes up to the tool. The sequence in which the axes occur specifies the **kinematic chain**. In this example, we have a **CBYXZ**kinematic.

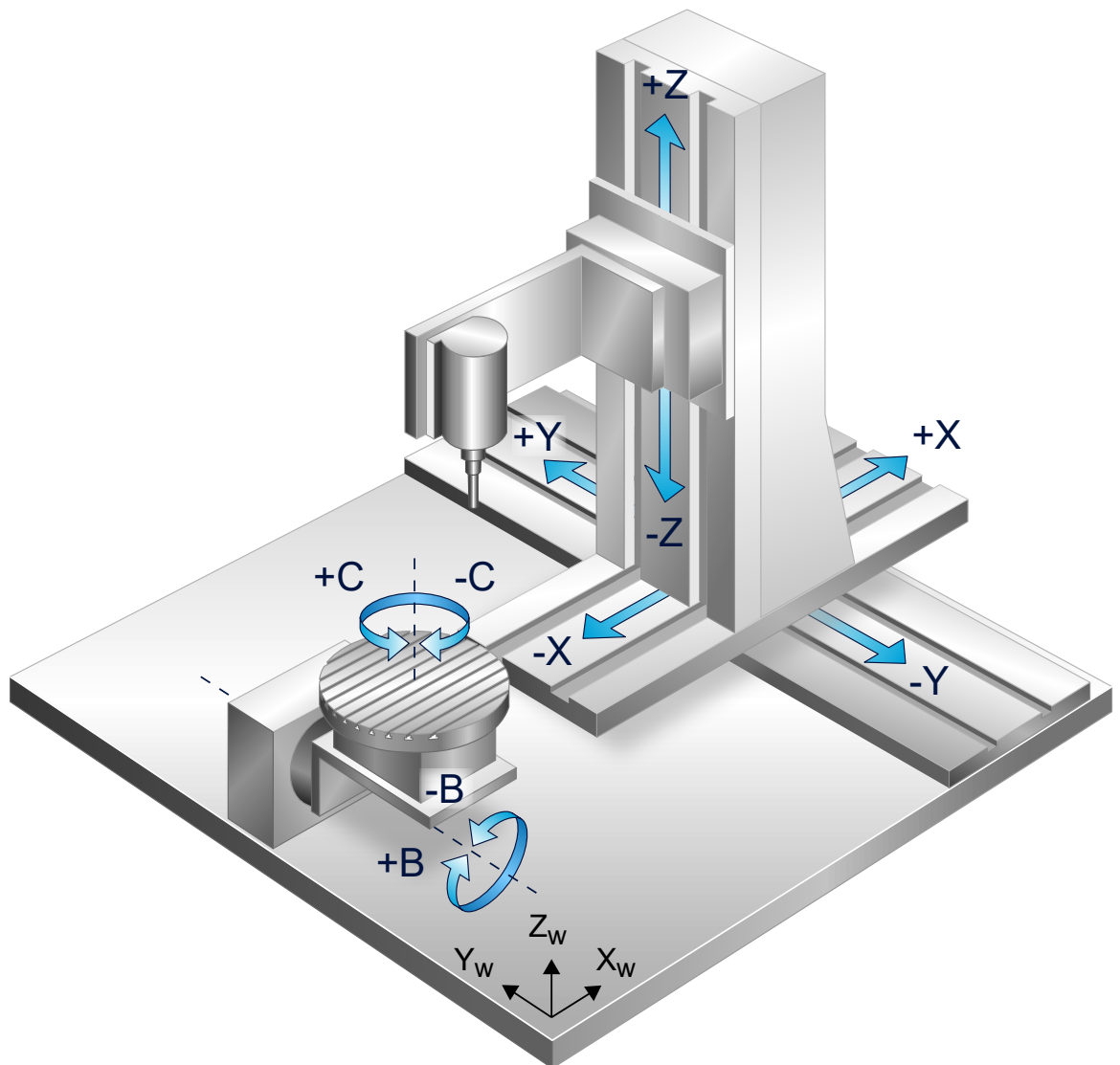


Fig. 1: Kinematic chain for a 5-axis CB machine

## Limit value

A limit for the compensation value must be specified in the axis parameters for each of the axes to be compensated (previously X, Y, Z).

The `lr_param.vol_comp.s_limit` (P-AXIS-00525) parameter defines the maximum allowed compensation value for this axis. If this value is exceeded, an error is issued and the axis is stopped in a controlled manner.



### Attention

If this limit is not defined, an error is issued when compensation is activated.



### Programing Example

#### Limit in "achsmid1.lis"

```
# Limit of 1mm for compensation values, in 0.1um
lr_param.vol_comp.s_limit      10000
```



### Attention

The `s_limit` limit must also be defined for gantry axes whose master axes are involved in Volumetric Compensation. Otherwise, an error is output when a compensation value is applied.

## 3.2

### Tool length (optional)

Tool length generally has an influence on compensation values. By default, calculation is performed with tool length 0. In order to incorporate tool length as well, the `N0X`, `N0Y`, `N0Z` parameters must be set in a parameter file. See Section "List of protected parameters" [► 25].

The vector [`N0X`, `N0Y`, `N0Z`] must be scaled and specifies tool orientation at zero position (from the flange to the tool tip). Except for the sign, this is typically the direction of the last axis in the kinematic chain, i.e. [0, 0, -1] for the example machine CBYXZ.

Tool length is automatically monitored and kept updated.

### 3.3 Modification of compensation values (optional)

The compensation values calculated by the compensation function can also be manipulated for each axis. You can either negate the compensation values or set them to 0.



#### Programming Example

##### Modification in compensation list

```
# Example: Negate compensation values for X and Y axes,  
# Axis sequence makes no difference  
  
negate                XY  
  
# Example: Set compensation values for Z axis to zero  
set_to_zero           Z
```

### 3.4 Alternative model (optional)



#### Notice

The application requires expert knowledge and is not necessary in normal use. This alternative model may prove useful for customers to develop their own measurement cycles.

Position errors in the rotary axes (see Section “Parametric model in accordance with DIN ISO 230 and ISO TR 16907 [► 23]”) are regarded as “average values” of the associated component errors in the literature\* on measurement/calibration of machine tools.

In this model, the rotation takes place around an **ideal** axis and afterwards the motion is subjected to location and component errors in the same manner. In default mode, the ISG implementation follows this model and is therefore compliant with the calibration models of third-party vendors.

However, the parameter in the compensation list

```
alternative_model      1
```

switches to an alternative model for the position errors of rotary axes. In this alternative model, the main rotation actually takes place about a distorted axis and any component errors then affect the rotary motion.

\* Soichi Ibaraki, Yusuke Ota, "A machining test to calibrate rotary axis error motions of five-axis machine tools and its application to thermal deformation test", International Journal of Machine Tools & Manufacture 86 (2014), 81-88

## 3.5 Diagnosis (optional)

For diagnostic purposes, you can create a file that contains a sorted list of all parameter records read in.

The file is in the "ECMA-376 Office Open XML File Format – spreadsheet" and is fully editable in versions of Microsoft® Office 2010® or higher or in Office 2013® or higher. This file is created during start-up of the controller and whenever compensation is updated.



### Programing Example

#### Creating a diagnostic file in compensation list

```
# Specify path to the diagnostic file
diagnosis_file    C:\volcomp\diagnosis_00.xml
```



### Notice

The function to create a diagnostic file is not available in TwinCAT.

## 3.6 Linkage to channel (optional)

Compensation may be linked to the channel in which it is activated. This has the effect that only axes contained in the channel are compensated. In addition, this linkage affects behaviour at program end; see Section "Behaviour on reset and in the event of errors [► 16]".



### Programing Example

#### Linkage to channel in compensation list

```
# Link to channel (Boolean)
associate_to_channel    1
```

## 3.7 Access to configuration and state data with V.G variables

### V.G variables

---

Read and write access to some configuration and state data can be made directly from the NC program with V.G variables.

### Path and file name

---

The **V.G.VOLCOMP[i].FILE[j]** parameter permits read and write access to the path and name of the parameter file under Index 'j' of Volumetric Compensation with Index 'i'. It therefore includes the value of the compensation parameter P-VOLC-00012 (file[j]). The data type is STRING.

### Activated

---

The **V.G.VOLCOMP[i].ACTIVE** parameter contains the information whether Volumetric Compensation is currently active with Index 'i'. Write access is not possible. The data type is BOOLEAN.

## 4 Programming



### Attention

Two preconditions must be fulfilled to use a previously configured compensation.

- Each of the axes involved in compensation must be referenced and error-free.
- The axes should also be referenced before the machine is measured.

The referencing requirement does not apply to axes with absolute encoders.

### 4.1 Activation (#VOLCOMP ON/OFF)

When these preconditions are met, Volumetric Compensation can be activated. This takes place either

- automatically (default) or
- manually by an NC command.

This behaviour is configured by the parameter P-VOLC-00017 (manual\_activation) in the compensation list.



### Programing Example

#### Configure activation in the compensation list

```
# Compensation is either
# - activated automatically as soon as all participating axes allow # it
# (error-free, referenced, etc.) or
# -manually by the NC command #VOLCOMP ON/OFF
manual_activation      0

# Compensation is exclusively activated/deactivated by the NC command
# #VOLCOMP ON/OFF
manual_activation      1
```

### Manual activation/deactivation

Volumetric Compensation can be activated in the NC program by the **#VOLCOMP** command.

**#VOLCOMP ON | OFF [ ID=.. [ NO\_MOVE ] { \ } ]**

modal

Or only in combination with ON

**#VOLCOMP ON [ ID=.. [NO\_MOVE] [CFACT\_X,Y,Z=.. ] { \ } ]**

modal

or

**#VOLCOMP [ ID=.. UPDATE { \ } ]**

modal



ON	Activate Volumetric Compensation
OFF	Deactivate Volumetric Compensation
ID=..	Identifier of the compensation to be programmed (0 – 4)
NO_MOVE	If the keyword NO_MOVE is specified, the axes do not move when compensation is activated/deactivated. Therefore, correction values are not increased, only the position controller setpoints are corrected.
CFACT_X, CFACT_Y, CFACT_Z	Correction factor to weight the volumetric compensation value of an axis accordingly. All factor are always initialised with 1. A programmed factor value is non-modal. If you require it to act with a subsequent selection, you must reprogram the factor value. Permitted values with a interval of [0.0;1.0]
UPDATE	Update a specific compensation. The affected compensation must be deactivated beforehand.
\	Separator ("backslash") for clear programming of the command over multiple lines.

The NC command below activates and deactivates all Volumetric Compensations in a single step:

<b>#VOLCOMP ON ALL</b>	modal
or	
<b>#VOLCOMP OFF ALL</b>	modal



## Programing Example

### Activate/deactivate compensation in the NC program

```

; Activate compensation with ID0
N10 #VOLCOMP ON [ID0]
...

:
; Deactivate compensation with ID0
...
N90 #VOLCOMP OFF [ID0]
```



### Attention

The machine stops when a #VOLCOMP commands is used. Therefore, it is advisable to plan the use of #VOLCOMP commands only for times when the machine is supposed to be in standstill anyway.

If NO\_MOVE is not specified, the NC command #VOLCOMP ON/OFF causes the immediate activation/deactivation of compensation values, i.e. the participating axes move. Correction values are then filtered by a filter whose size and duration are set in the parameter P-VOLC-00018 (filter\_steps).

If UPDATE is specified, the parameters are read from the configured files again. However, the related compensation must first be deactivated.

An update also includes the configuration from the compensation lists (participating axes, kinematic chain etc.) in addition to the error parameters.

Before an update, the compensation is implicitly deactivated, as required.



### Programing Example

#### Reload parameters to the NC program

```
; Deactivate compensation
; ...
N090 #VOLCOMP OFF [ID0]
; Update compensation parameters
N100 #VOLCOMP [ID0 UPDATE]
; Reactivate compensation
N110 #VOLCOMP ON [ID0]
```

## 4.2 Behaviour on reset and in the event of errors

### Reset

---

A controller reset disables all Volumetric Compensations without causing additional axis movements.

### Error case

---

If an axis assumes an error state, every compensation in which the axis participates is marked as incorrect. Compensation values are removed from the axis without causing an additional axis movement. The affected compensations are deactivated; the other participating axes are stopped in a controlled fashion and their compensation values are reduced.

### Response

---

After reset, deactivated compensations are re-activated by the **#VOLCOMP** command.

### Program end

---

If compensations are still active at program end (M30) and they are linked to the current channel, an error is output and the compensation values of the affected axes are reduced without any additional movement.

## 5 Notes on measurement

### 5.1 Pre-compensation

#### Pre-compensation

---

Volumetric Compensation is regarded as an additional function block in addition to the conventional geometric compensations to enhance machine accuracy even further. Conventional compensations include:

- Leadscrew error compensation
- Cross compensation
- Plane compensation

For this reason, it is advisable to activate these conventional compensations in advance, i.e. to execute a pre-compensation and then add the measurement and finally apply Volumetric Compensation.

### 5.2 Influence of measuring strategy on configuration

The measuring strategy applied to parameter determination, the error model used and the other configuration factors of the machine all influence how Volumetric Compensation must be configured.

In general, the following must match:

- measuring strategy and
- configuration of the Volumetric Compensation

#### 5.2.1 Workpiece axes

In general, it makes a difference whether an axis lies on the tool or the workpiece side. For Volumetric Compensation, this distinction is made via the workpiece axes parameter. By default, this value is assigned the value 0 and so all axes lie in the tool.

Due to settings on the machine, an axis on the workpiece side may already be considered elsewhere, e.g. as a result of the vz\_stellgr or vz\_istw axis parameters. In this case, the axis lies in the tool – also for the measuring technician – and the axis should also lie in the tool for Volumetric Compensation, i.e. workpiece\_axes should remain at 0.

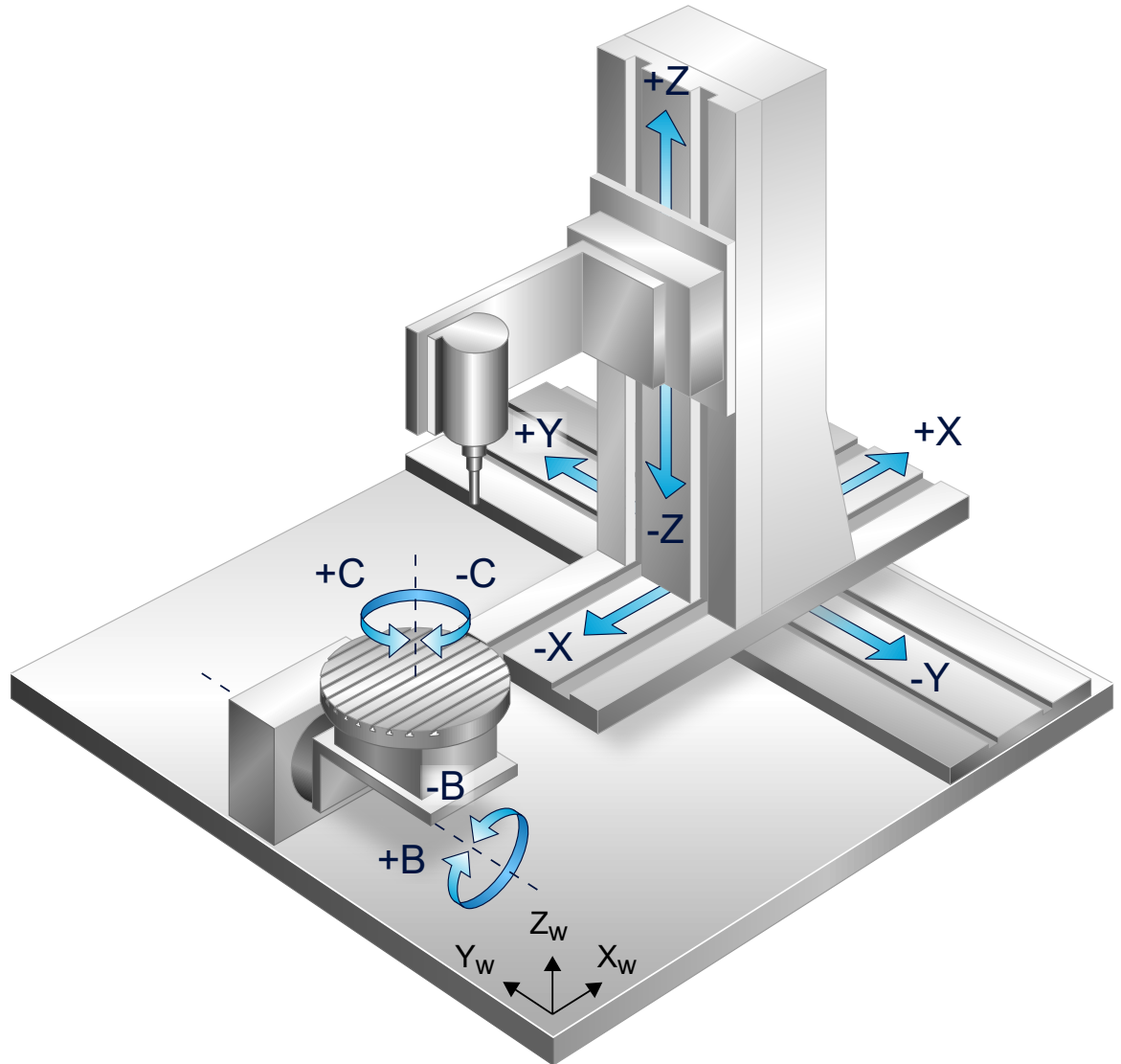
Rule of thumb: if the measuring technician considers an axis to be on the tool side, the axis should also be treated as such when configuring Volumetric Compensation and vice versa.

## 5.2.2

### Cartesian pre-compensation

The sequence for measuring the axes plays an important role.

In the example, let us assume there is a 5-axis machine XYZBC with rotary and swivel axis in the workpiece.



**Fig. 2: 5-axis machine XYZBC with rotary and swivel axis in the workpiece**

If all 5 axes are measured simultaneously, one Volumetric Compensation CBYXZ is sufficient.

However, the following sequence is also possible:

- The axes YXZ are measured and compensated.
- Then the B and C axes are measured and compensated.

2 compensations are used:

- YXZ, Cartesian parameter file only
- CBYXZ, parameter files for the B and C axes only

This is how the measuring strategy of Cartesian pre-compensation used is mapped.

### 5.2.3

#### Coordinate system offset, rotation centre

Some measuring technicians do not include the error parameters of rotary axes in the machine coordinate system but in the system for the rotation centre of the axis (for example, in the tool table).

If the rotation centre of the C axis lies at  $[0, 0, -550]$ , for example, the method of measuring the machine with  $[X, Y, Z]$  commands may result in the output of compensation values that would actually be expected with  $[X, Y, Z \pm 550]$ .

This problem can be solved by implementing a coordinate system offset on the axis side. The parameters  $IDX, IDY, IDZ$  can be used for this; see Section "Workpiece axes". [► 20]



#### Attention

In the event that the C axis was compensated in a different coordinate system than the Cartesian axes, it is imperative to configure two compensations. Otherwise, the machine cannot be compensated at the exact position.

## 6 Error parameters

### 6.1 Parametric model according to DIN ISO 230 and ISO TR 16907

Classic compensations (leadscrew error, crossing and error compensation) determine their compensation values by means of a simple lookup operation in one or multi-dimensional tables. The compensation value of the affected axis then depends on:

- the current value of the axis itself (ssfk) or
- only the value of another axis (crosscomp) or
- the values of two axes (crosscomp2).

The decisive drawback of this approach is that it cannot be generalised in practice for higher dimensions since the required lookup tables would be too unwieldy and measuring the machine would take too long. A 5-dimensional table would be needed to compensate adequately for a 5-axis machine ("crosscomp5"). 10 interpolation points per axis would result in  $10^5$  grid points.

The parametric approach avoids this weakness by including information about the machine that is ignored by the lookup approach. A kinematic model of the (inexact) machine is created by using the given parameters. Then the compensated axis values are calculated from the known target position of the tool by inverting the model.

The Volumetric Compensation model works with two kinds of error parameters:

- Position errors
- Component errors

#### Position errors

The position errors of an axis describe how strongly the position and orientation of an axis deviate from their ideal values (position deviation is only relevant for rotary axes). In this way, global aspects of the machine, for example the rectangularity of axes, can be detected.

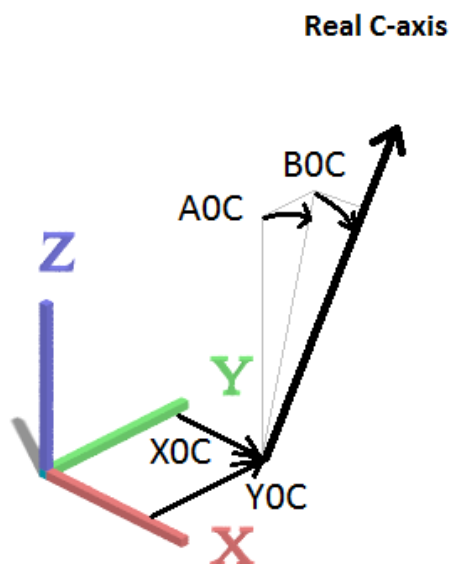


Fig. 3: Position errors of the C axis, without C0C

## Component errors

Component errors specify which deviations a slide experiences when it is guided along an axis. The slide can be shifted with respect to its target position in each of the three spatial directions ( $EX^*$ ,  $EY^*$ ,  $EZ^*$ ) and can be inclined with respect to its target orientation ( $EA^*$ ,  $EB^*$ ,  $EC^*$ ). These errors are typically provided in a one-dimensional lookup table.

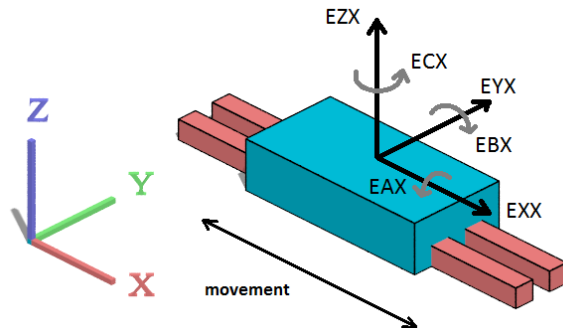


Fig. 4: Component errors of the X axis



## 6.2 List of parameters supported

### DIN ISO 230

The list below contains all error parameters that are supported in compliance with the DIN ISO 230 standard.

The number of position errors is redundant: only a subset of position errors is required to describe a machine completely. In addition, there are several options to choose this type of descriptive subset.

Position errors	Component errors
X0X, Y0X, Z0X, A0X, B0X, C0X	EXX, EYX, EZX, EAX, EBX, ECX
X0Y, Y0Y, Z0Y, A0Y, B0Y, C0Y	EXY, EYY, EYZ, EAY, EBY, ECY
X0Z, Y0Z, Z0Z, A0Z, B0Z, C0Z	EXZ, EYZ, EZZ, EAZ, EBZ, ECZ
X0A, Y0A, Z0A, A0A, B0A, C0A	EXA, EYA, EZA, EAA, EBA, ECA
X0B, Y0B, Z0B, A0B, B0B, C0B	EXB, EYB, EZB, EAB, EBB, ECB
X0C, Y0C, Z0C, A0C, B0C, C0C	EXC, EYC, EYC, EAC, EBC, ECC

### ISG parameters

The ISG kernel uses the following additional parameters to describe the kinematic model of the machine or provides other functions.

Parameter	Description
P0X, P0Y, P0Z	Zero position of the tool (TCP)
N0X, N0Y, N0Z	Zero orientation of the tool, scaled vector
PXA, PYA, PZA PXB, PYB, PZB, PXC, PYC, PZC	Centre of rotation of A axis Centre of rotation of B axis Centre of rotation of C axis
VXX, VYX, VZX VXY, VYY, VZY VXZ, VYZ, VZZ VXA, VYA, VZA VXB, VYB, VZB VXC, VYC, VZC	for cardanic machines Modified orientation of the X axis Modified orientation of the Y axis Modified orientation of the Z axis Modified orientation of the A axis Modified orientation of the B axis Modified orientation of the C axis
IDX, IDY, IDZ, IDA, IDB, IDC	Additive offsets to the programmed axis values
LNx, LNY, LNZ, LNA, LNB, LNC	Logical axis numbers, overwrites compensation parameters X, Y.. from Section 2.1 [► 9]. This allows to note the logical axis numbers of the measured axes in the compensation file.

## 6.3 File formats supported

### 6.3.1 Units of measure

#### General

---

The following units of measure apply to all file formats.

#### Parameter values

---

The values of parameters must be specified in millimetres for offsets or rad (e.g.  $[0.2\pi]$ ) for angles.

#### Arguments

---

While position errors are global quantities and have only one value, component errors are typically specified as a function of the current axis value. For example, as a unidimensional lookup table, the error EYX generally depends on X. The arguments of the component errors (in this case X) must be specified in millimetres (for translation axes) or in degrees (e.g.  $[0^\circ, 360^\circ]$ , for rotation axes).

The following arguments are currently available.

Argument	Description
X	Axis value of the X axis, millimetres
Y	Axis value of the Y axis, millimetres
Z	Axis value of the Z axis, millimetres
A	Axis value of the A axis, in degrees
B	Axis value of the B axis, in degrees
C	Axis value of the C axis, in degrees

## 6.3.2 ISG CSV format

### Description

The simplest and most flexible file format is a tabular ISG CSV format.

In terms of its structure it is very similar to an Excel spreadsheet.

- The columns are named in the header. The header can contain both argument and parameter names.
- The actual records are specified in the rows below.
- Row comments start by a double slash "//".
- A "file break" can be created with a hash "#". After this break, a new header and the associated records can be specified.



### Programing Example

#### Machine position errors

```
// The header only contains position errors.
Y0A    Z0A    X0B    ...

// Position errors are typically (but not necessarily)
// constant. Therefore, one data row is usually sufficient here.
0.1     0.2     0.11    ...
```



### Programing Example

#### Component errors of the X axis

```
// The header contains the argument "X" and component errors.
X      EXX    EYX    EZX    EAX    EBX    ECX
0.0     0.02   0.01   0.07   0.00   0.0042 0.003
10.0    0.00  -0.1    0.01   0.00   0.0011 0.002
...
// Further records depending on X
```

### Extended functional dependencies

Any functional dependencies can be implemented in the ISG CSV format. For example, it is conceivable that two data blocks for component errors in the X axis, one each for  $Z = 0$  and one for  $Z = 100$ , must be specified due to sag effects on the machine. Parameters are interpolated as described in Section "File formats supported [► 26]".

Any higher-dimensional dependencies can be implemented with this method.



## Programing Example

### Component errors of the X axis for Z = 0 and Z = 100

```
// The argument "Z" now crops up in the header.
Z      X      EXX      EYX      EZX      EAX      EBX      ECX

// In the first block Z is always = 0.
0.0    0.0    0.02    0.01    0.07    0.00    0.0042 0.003
0.0    10.0    0.00   -0.1    0.01    0.00    0.0011 0.002
...

// In the second block Z is always = 100.
100.0  0.0    0.01    0.02    0.03    0.07    0.0040 0.003
100.0  10.0    0.03   -0.09   0.03    0.06    0.0015 0.000
...
```

### 6.3.3

## Etalon Exchange Format

### Description

Etalon AG® is a vendor of hardware and software for the ISO-compliant measurement of machine tools. Measurement returns a set of parameter files in the proprietary exchange format. This format is similar to the format of Microsoft Windows INI files.

Row comments start by a double slash "//".



### Attention

Etalon offers output in the exchange format in two variants: "Vendor" or "Kinematic". The "Vendor" variant must always be selected here.



## Programing Example

### Position errors

```
// Every parameter is specified in a block.
// The block begins with the name of the parameter in square brackets.

[C0B]
VALUE = 0.000022355737

[A0B]
VALUE = 0.000047622950
```



## Programing Example

```
// Lookup tables are specified via a 2D array called "Gridpoints".

[EXY]
Gridpoints = {
    -280.0 -0.001358635132
    -270.0 -0.000240680764
    -260.0 0.000877273605
    ... // Further records in the form of "Parameter argument"
    260.0 -0.001297146744
    270.0 -0.000316265101
} // end of EXY
```

## 6.4 Interpolation of parameter values

### 6.4.1 Interpolation mode

A specific interpolation mode can be set or forced via the interpolation compensation parameter.

- GRID (default) – linear interpolation, expects the parameter values on a grid
- SCATTER – non-linear interpolation, any distribution of parameters
- AUTO – analyses whether the parameters lie on a grid and selects GRID or SCATTER accordingly
- PATH: Error parameters are specified at the corner points of a polygon. Interpolation takes place in linear sections along the polygon.
- PATH\_XY: Error parameters are specified at the corner points of a polygon in the XY plane.

### 6.4.2 Constancy in the case of modulo passages, parameter splicing

#### Constancy

---

In the case of rotary axes, the axis values typically jump between two cycles, from just less than 360° to just over 0°, e.g. for a 360° transition. This may then result in interpolated parameter values jumping.

Users have two options to prevent jumping.

#### Splicing by hand

---

Users can adapt the parameter files by adding a new record for 360° for every affected parameter where the same parameter value is set as in the case of the 0° block.



#### Programming Example

##### EXC – Etalon Exchange Format

```
[EXC]
Gridpoints = {
    0.0 -0.000833484486
    10.0 -0.000845857719
...
    340.0 -0.001687237526
    350.0 -0.001699610758
    360.0 -0.000833484486// <- duplicate of the 0° row
} // end of EXC
```

## Splicing in the kernel

---

Splicing the parameter values can also be executed by the kernel itself by setting `enable_splicing` to 1.

The precondition is that the applicable rotary parameter is provided as a 1D table. There is no automatic splicing for higher-dimensional tables.

In addition, steps must be taken to ensure that enough memory is reserved for the new parameter sets to be added, i.e. that the chosen value for `vol_comp[i].max_records` is large enough.

## 7 Examples

### 7.1 Cartesian compensation of XYZ kinematics



#### Programing Example

##### Configuration in start-up list

```
vol_comp[0].max_records      1000
vol_comp[0].file_name        C:\volcomp\vol_comp_0.lis
```



#### Programing Example

##### Configuration of XYZ kinematics in "vol\_comp\_0.lis"

```
X          1
Y          2
Z          3

chain      YXZ

file[0]     C:\kartesisch_YXZ.exc
file_format[0] ETALON_EXCHANGE
```



## 7.2 Compensation of head kinematics CYXZ, with Cartesian pre-compensation



### Programing Example

#### Configuration in start-up list

```
# Cartesian pre-compensation
vol_comp[0].max_records      1000
vol_comp[0].file_name        C:\volcomp\vol_comp_0.lis

# second compensation, for C axis
vol_comp[1].max_records      1000
vol_comp[1].file_name        C:\volcomp\vol_comp_1.lis
```



### Programing Example

#### Configuration YXZ kinematics in "vol\_comp\_0.lis"

```
X          1
Y          2
Z          3

chain      YXZ

file[0]     C:\kartesisch_YXZ.exc
file_format[0] ETALON_EXCHANGE
```



### Programing Example

#### Configuration CYXZ kinematics in "vol\_comp\_1.lis"

```
X          1
Y          2
Z          3
C          5

chain      CYXZ

file[0]     C:\C_achse.exc
file_format[0] ETALON_EXCHANGE
```



## Programing Example

### Parameter file C:\C\_axis.exc, including parameter splicing

```
// Etalon Exchange Format
// Created by TRAC-CAL V30 Build: 0017 dated 2015-04-28
// Parametric errors of one rotary axis by mulitlateration
// Date: 24-Jun-2015
// -----
//

[HEADER]
MACHINE_TYPE = Exeron HSC 600
MACHINE_NO   =
CUSTOMER     = ETALON AG
DATE_OF_CALIBRATION = -
OPERATOR     = Etalon master
FILE_TYPE    = rotary
USERSPECIFIC_COMMENT = -

[EZC]
Gridpoints = {
    0.0 -0.000423728477
    5.0 -0.000351288461
   10.0 -0.000278665223
   15.0 -0.000205672612
   20.0 -0.000132112763
:
   335.0 -0.000001094701
   340.0 -0.000120503607
   345.0 -0.000239848892
   350.0 -0.000359170264
   355.0 -0.000478491636
   360.0 -0.000423728477
} // end of EZC
// The example is curtailed at this point
```

## 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](mailto: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](mailto:documentation@isg-stuttgart.de)



© Copyright  
ISG Industrielle Steuerungstechnik GmbH  
STEP, Gropiusplatz 10  
D-70563 Stuttgart  
All rights reserved  
[www.isg-stuttgart.de](http://www.isg-stuttgart.de)  
[support@isg-stuttgart.de](mailto:support@isg-stuttgart.de)

