DOCUMENTATION ISG-kernel

Manual
Configuration list for scene visualization and collision monitoring

Short Description:
SECO

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

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.

---

i 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

1 General description

1.1 Links 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.

1.2 Structure and layout of the list

The parameters in the list for scene visualisation and collision monitoring contain general and specific data and references to files for visualisation and the configuration of graphical (machine) frames (so-called scenes) and for the collision monitoring of these graphical objects. This data is adopted in the so-called scene database for initialisation at start-up.

Value ranges of parameters are defined by specifying a limit resulting from the data width, such as MAX(UNS16) etc.

In addition, all scene parameters can be changed or expanded by special commands directly in NC program [PROG].

1.3 Syntax and interpretation of ASCII list file

An interpreter copies the entries in the ASCII list file into identical internal structures which are then checked for plausibility. To ensure reliable controller start-up every time, any defective entries found by the plausibility check are replaced by default values.

Unknown entries are not taken over. These irregularities are displayed by warning messages. We advise you to investigate the cause for these warning messages and remove defective entries from the ASCII list file.

Notice

The following agreement applies to BOOLEAN data:

Value	Meaning
0	Definition of FALSE
1	Definition of TRUE

Notice

The following agreement applies to STRING data:

If a character string containing characters with a special meaning in ASCII lists (e.g. comment characters, spaces [▷ 7]) is assigned to a STRING type list parameter, this character string must be defined in inverted commas ″..″ (available as of V3.1.3081.0, V3.1.3108.0).

```
example[0].name "STRING_WITH_COMMENT( # /*)_CHARACTERS"
```

Trailing spaces are discarded on import. The entry..

```
example[0].name "STRING_WITH_POST_SPACES "
```

..has the same meaning as

```
example[0].name "STRING_WITH_POST_SPACES"
```

If the character string only contains characters without any special meaning, no inverted commas are required.

```
example[0].name STRING_WITH_STANDARD_CHARACTERS!
```

1.4 Comments in the ASCII list file

Comments can be in an entire line or can be added at the end of a line.

With a comment spanning an entire line, the comment character "'#'" must be placed at the start of the line and followed by a blank.

If a comment is to be inserted at the end of a line, only a blank is required before the comment. Blank lines are also possible.

Example

Comments in the ASCII list file

```
#
***********************************************************************
# Data
#
***********************************************************************
#
# List comments after numerical values

dummy[1] 1 Comment
dummy[2] 1 # Comment
dummy[3] 1 ( Comment
dummy[4] 1 /* Comment
...
...
```

However, if a character string was assigned to the list parameter as a value in the line, any following comment <u>must</u> be opened by a bracket '('. The comment characters space, # and /* are not permitted.

If a '(' itself is part of the character string, the character string must be defined in inverted commas ″..″ (available as of V3.1.3081.0, V3.1.3108.0).

```
# List comments after strings

beispiel[0].bezeichnung STRING_1 (comment requires a '(' bracket!)

beispiel[1].bezeichnung″ STRING_(2)″ (comment requires a '(' bracket!)
```

2 Description of elements

2.1 General parameters

2.1.1 File name of the status image (P-SECO-00001)

P-SECO-00001	Filename of the state image
Description	This element specifies the name of the file to which the current status image is written in case of a collision. The string contains the path and the filename. It can be specified as both absolute and relative.
Parameter	collision_detected_outputfile
Data type	STRING
Data range	Maximum 135 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

2.1.2 Path to the enveloping bodies (P-SECO-00002)

P-SECO-00002	Path to the enveloping bodies
Description	This element specifies the path to the enveloping bodies which are monitored for collision. It can be specified as both absolute and relative. The length of the path is limited to 103 characters.
Parameter	path_to_geometry
Data type	STRING
Data range	Maximum of 103 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

2.1.3 Activating collision monitoring (P-SECO-00003)

P-SECO-00003	Activating collision monitoring
Description	This element always enables collision monitoring for the channel.
Parameter	use_collision_detection
Data type	BOOLEAN
Data range	0 / 1
Dimension	----
Default value	0
Remarks	

2.1.4 Reaction after collision (P-SECO-00004)

P-SECO-00004	Reaction after collision
Description	This parameter defines the reaction of the controller in case of a collision. The parameter entry is only set after each CNC reset. The reaction can be changed in the NC program by the #COLL SET [REACTION=..] command.
Parameter	collision_reaction
Data type	SGN32
Data range	**1 – Abort before collision (default)** 2 – Wait for user release before collision 3 – Abort after collision 4 – Wait for user release after collision
Dimension	----
Default value	1
Remarks	

2.1.5   Consistency check of data models (P-SECO-00005)

P-SECO-00005	Consistency check of data models
Description	This parameter influences the consistency check of data models in the collision library.
	By default, the consistency of the data model is checked. The check can be disabled if the data models (DCP files) are always verified. This reduces the process time when new geometries are read in.
Parameter	mode
Data type	SGN32
Data range	**0 : Debug mode; the consistency of the data model is checked (default)**
	!= 0 : Release mode; the data model is no longer checked for consistency
Dimension	----
Default value	0
Remarks	

2.1.6   Suffix of colliding objects (P-SECO-00006)

P-SECO-00006	Suffix of colliding objects
Description	This parameter specifies a file extension (suffix) which, in the event of a collision, extends the two colliding graphical objects with the suffix and also creates these files.
	This highlights the colliding objects in colour. If motion continues after collision (collision_reaction = 2 / 4), the two created graphical objects are deleted after user confirmation.
Parameter	collided_gobject_suffix
Data type	STRING
Data range	Maximum of 39 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

2.1.7 Path to collision geometry (P-SECO-00007)

P-SECO-00007	Path to collision geometry
Description	This parameter specifies a relative path that is added to the filename of the colliding graphical objects. The path is used to search for the description of the temporarily saved graphical objects (complete file:= <path_to_geometry><path_to_collided_geometry><GOBJECT::filename>). The graphical description of the colliding objects can be highlighted in colour and should be slightly larger than the original object since the additional graphical object is displayed at the same position and would otherwise be poorly visible.
Parameter	path_to_collided_geometry
Data type	STRING
Data range	Maximum of 103 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

2.1.8 Coordinate system of an element

Elements such as a graphical object or a node point each have a coordinate system that consists of translation and rotation.

These elements have the following translation and rotation parameters in their structures

2.1.8.1 Translation (P-SECO-00025)

P-SECO-00025	Definition of translation
Description	This parameter defines the translation of an element in Cartesian form (x/y/z). The parameter to translate an element is used in the following structures: • gobject[i].cs.position[j] • linkpoint[i].cs_init.position[j] • linkpoint[i].cs_actual.position[j]
	position[j] where j=0,1,2
Data type	REAL64
Data range	MIN(REAL64) ≤ P-SECO-00025 ≤ MAX(REAL64)
Dimension	[0.1 μm]
Default value	0.0
Remarks	

2.1.8.2 Rotation (P-SECO-00026)

P-SECO-00026	Definition of rotation
Description	The static rotation of an element in relation to the node point about the three main axes is described in this parameter as a 3 x 3 $$\begin{bmatrix} rotxx & rotxy & rotxz \\ rotyx & rotyy & rotyz \\ rotzx & rotzy & rotzz \end{bmatrix}$$ rotation matrix. The parameter to rotate an element is used in the following structures: • gobject[i].cs.orientation[j][k] [▷ 15] • linkpoint[i].cs_init.orientation[j][k] • linkpoint [i].cs_actual.orientation[j][k]
Parameter	orientation[j][k] where j and k =0,1,2
Data type	REAL64
Data range	-1 <= P-SECO-00026 <= 1
Dimension	----
Default value	0.0
Remarks	To specify orientation[j][k] j : Line in the matrix k : Column in the matrix

2.2 Data of a graphical object (gobject[i].*)

The data of a graphical object are defined in this structure. The number of graphical objects is limited to 512.

Structure name	Index
gobject[i]	0 ≤ i ≤ 511

2.2.1 Name of the graphical object (P-SECO-00020)

P-SECO-00020	Name of the graphical object
Description	This parameter defines the name of a graphical object.
Parameter	gobject[i].name
Data type	STRING
Data range	Maximum of 39 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

Example taken from list:

```
:
gobject[0].name                        Ausleger_Einfach
gobject[0].filename                     Ausleger_01
gobject[0].linkpoint                    A_ACHSE
gobject[0].cs.position[0]               500000
gobject[0].cs.position[1]               0
gobject[0].cs.position[2]               0
gobject[0].cs.orientation[0][0]         1
gobject[0].cs.orientation[0][1]         0
gobject[0].cs.orientation[0][2]         0
gobject[0].cs.orientation[1][0]         0
gobject[0].cs.orientation[1][1]         1
gobject[0].cs.orientation[1][2]         0
gobject[0].cs.orientation[2][0]         0
gobject[0].cs.orientation[2][1]         0
gobject[0].cs.orientation[2][2]         1
gobject[0].key[0]                       Colour
gobject[0].value[0]                     red
gobject[0].key[1]                       Shading
gobject[0].value[1]                     grey
:
```

2.2.2 File name of geometric data (P-SECO-00021)

P-SECO-00021	Filename of geometric data
Description	This parameter defines the name of the file which stores the information required to describe the geometry of the graphical object. Only the filename is specified with no file extension or path.
Parameter	gobject[i].filename
Data type	STRING
Data range	Maximum of 135 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

i **Notice**

Only the filename is specified with no file extension or path. The file extension and the path must be managed by each client.

The following applies to collision monitoring:

The path to the geometric files is specified in the start-up list.

The geometry must be available as a DCP file for collision monitoring and as a VRML file to visualise the collision scene.

Example taken from list:

```
:
gobject[0].name                        Ausleger_Einfach
gobject[0].filename                     Ausleger_01
gobject[0].linkpoint                    A_ACHSE
```

```
gobject[0].cs.position[0]              500000
gobject[0].cs.position[1]              0
gobject[0].cs.position[2]              0
gobject[0].cs.orientation[0][0]        1
gobject[0].cs.orientation[0][1]        0
gobject[0].cs.orientation[0][2]        0
gobject[0].cs.orientation[1][0]        0
gobject[0].cs.orientation[1][1]        1
gobject[0].cs.orientation[1][2]        0
gobject[0].cs.orientation[2][0]        0
gobject[0].cs.orientation[2][1]        0
gobject[0].cs.orientation[2][2]        1
gobject[0].key[0]                      Colour
gobject[0].value[0]                    red
gobject[0].key[1]                      Shading
gobject[0].value[1]                    grey
:
```

2.2.3   Node point name (P-SECO-00022)

P-SECO-00022	Node point name
Description	This parameter defines the name of the node (LINKPOINT) of the kinematic chain. It is attached to the graphical object. Here, for example, enter the axis name to which the kinematic chain is attached. If the graphical name is not attached to a node point, enter GROUND.
Parameter	gobject[i].linkpoint
Data type	STRING
Data range	Maximum of 39 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

Example taken from list:

```
:
gobject[0].name                        Ausleger_Einfach
gobject[0].filename                     Ausleger_01
gobject[0].linkpoint                   A_ACHSE
gobject[0].cs.position[0]              500000
gobject[0].cs.position[1]              0
gobject[0].cs.position[2]              0
gobject[0].cs.orientation[0][0]        1
gobject[0].cs.orientation[0][1]        0
gobject[0].cs.orientation[0][2]        0
gobject[0].cs.orientation[1][0]        0
gobject[0].cs.orientation[1][1]        1
gobject[0].cs.orientation[1][2]        0
gobject[0].cs.orientation[2][0]        0
gobject[0].cs.orientation[2][1]        0
gobject[0].cs.orientation[2][2]        1
gobject[0].key[0]                      Colour
gobject[0].value[0]                    red
gobject[0].key[1]                      Shading
gobject[0].value[1]                    grey
```

:

2.2.4 Orientation of a graphical object (gobject[i].cs.*)

The parameter "gobject[i].cs.*" defines the orientation of the graphical object with the number "i". Orientation is defined by translation and rotation (defined as a matrix) with respect to the related basic node point coordinate system.

2.2.4.1 Definition of translation and rotation

The translation and rotation of a graphical object is executed in the coordinate system structure. This consists of the parameters P-SECO-00025 [▸ 11] and P-SECO-00026 [▸ 12].

```
gobject[i].cs.position[j]         where j =0,1,2
gobject[i].cs.orientation[j][k]   where j and k =0,1,2
```

Example taken from list:

```
:
gobject[0].name                   Ausleger_Einfach
gobject[0].filename               Ausleger_01
gobject[0].linkpoint              A_ACHSE
gobject[0].cs.position[0]         500000
gobject[0].cs.position[1]         0
gobject[0].cs.position[2]         0
gobject[0].cs.orientation[0][0]   1
gobject[0].cs.orientation[0][1]   0
gobject[0].cs.orientation[0][2]   0
gobject[0].cs.orientation[1][0]   0
gobject[0].cs.orientation[1][1]   1
gobject[0].cs.orientation[1][2]   0
gobject[0].cs.orientation[2][0]   0
gobject[0].cs.orientation[2][1]   0
gobject[0].cs.orientation[2][2]   1
gobject[0].key[0]                 Colour
gobject[0].value[0]               red
gobject[0].key[1]                 Shading
gobject[0].value[1]               grey
:
```

If no rotation is to be executed, specify the unit matrix

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(see example below).

Example taken from list for unit matrix:

```
:
gobject[0].name                   Ausleger_Einfach
gobject[0].filename               Ausleger_01
gobject[0].linkpoint              A_ACHSE
gobject[0].cs.position[0]         500000
gobject[0].cs.position[1]         0
gobject[0].cs.position[2]         0
gobject[0].cs.orientation[0][0]   1
gobject[0].cs.orientation[0][1]   0
gobject[0].cs.orientation[0][2]   0
```

```
gobject[0].cs.orientation[1][0]          0
gobject[0].cs.orientation[1][1]          1
gobject[0].cs.orientation[1][2]          0
gobject[0].cs.orientation[2][0]          0
gobject[0].cs.orientation[2][1]          0
gobject[0].cs.orientation[2][2]          1
gobject[0].key[0]                        Colour
gobject[0].value[0]                      red
gobject[0].key[1]                        Shading
gobject[0].value[1]                      grey
:
```

2.2.4.2   Additional characteristics of graphical objects: key/value pairs

The parameters "key" (P-SECO-00027) and "value" (P-SECO-00028) assign additional so-called key/ value pairs to the graphical object They describe properties which are evaluated during visualisation, for example.

Attention: "key" and "value" must always form a pair of properties (j=j).

P-SECO-00027	Key (category)
Description	This parameter defines a key for a value pair.
Parameter	gobject[i].key[j] where j=0 to 4
Data type	STRING
Data range	Maximum of 39 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

P-SECO-00028	Value
Description	This parameter defines a value for a value pair.
Parameter	gobject[i].value[j] where j=0 to 4
Data type	STRING
Data range	Maximum of 39 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

Example taken from list:

```
:
gobject[0].name                    Ausleger_Einfach
gobject[0].filename                Ausleger_01
gobject[0].linkpoint               A_ACHSE
gobject[0].cs.position[0]          500000
gobject[0].cs.position[1]          0
gobject[0].cs.position[2]          0
gobject[0].cs.orientation[0][0]    1
gobject[0].cs.orientation[0][1]    0
gobject[0].cs.orientation[0][2]    0
gobject[0].cs.orientation[1][0]    0
gobject[0].cs.orientation[1][1]    1
gobject[0].cs.orientation[1][2]    0
gobject[0].cs.orientation[2][0]    0
gobject[0].cs.orientation[2][1]    0
gobject[0].cs.orientation[2][2]    1
gobject[0].key[0]                  Colour
gobject[0].value[0]                red
gobject[0].key[1]                  Shading
gobject[0].value[1]                grey
:
```

2.3   Data of a node point (linkpoint[i].*)

This structure defines the data of a node point. The number of node points is limited to 64.

Structure name	Index
linkpoint[i]	0 ≤ i ≤ 63

2.3.1   Node point name (P-SECO-00040)

P-SECO-00040	Node point name
Description	This parameter defines the name of a node point.
Parameter	linkpoint[i].name
Data type	STRING
Data range	Maximum of 39 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

Example taken from list:

```
:
linkpoint[0].name                         X_ACHSE
linkpoint[0].mountpoint                   GROUND
linkpoint[0].input_log_ax_nr              1
linkpoint[0].cs_init.position[0]          0.000000
linkpoint[0].cs_init.position[1]          0.000000
linkpoint[0].cs_init.position[2]          0.000000
linkpoint[0].cs_init.orientation[0][0]    1.000000
linkpoint[0].cs_init.orientation[0][1]    0.000000
linkpoint[0].cs_init.orientation[0][2]    0.000000
linkpoint[0].cs_init.orientation[1][0]    0.000000
linkpoint[0].cs_init.orientation[1][1]    1.000000
linkpoint[0].cs_init.orientation[1][2]    0.000000
linkpoint[0].cs_init.orientation[2][0]    0.000000
linkpoint[0].cs_init.orientation[2][1]    0.000000
linkpoint[0].cs_init.orientation[2][2]    1.000000
linkpoint[0].trans_rot                    TRANS_X
linkpoint[0].visible                      0
linkpoint[0].inverse                      0
:
```

2.3.2   Name of the previous node point (P-SECO-00041)

P-SECO-00041	Name of the previous node point
Description	Name of the existing node point of the kinematic chain to which the new node point is attached (P-SECO-00040 [▸ 18]). The system-internal basic node point GROUND forms the initial point of a kinematic chain.
Parameter	linkpoint[i].mountpoint
Data type	STRING
Data range	Maximum of 39 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

Example taken from list:

```
:
linkpoint[0].name                        X_ACHSE
linkpoint[0].mountpoint                  GROUND
linkpoint[0].input_log_ax_nr             1
linkpoint[0].cs_init.position[0]         0.000000
linkpoint[0].cs_init.position[1]         0.000000
linkpoint[0].cs_init.position[2]         0.000000
linkpoint[0].cs_init.orientation[0][0]   1.000000
linkpoint[0].cs_init.orientation[0][1]   0.000000
linkpoint[0].cs_init.orientation[0][2]   0.000000
linkpoint[0].cs_init.orientation[1][0]   0.000000
linkpoint[0].cs_init.orientation[1][1]   1.000000
linkpoint[0].cs_init.orientation[1][2]   0.000000
linkpoint[0].cs_init.orientation[2][0]   0.000000
linkpoint[0].cs_init.orientation[2][1]   0.000000
linkpoint[0].cs_init.orientation[2][2]   1.000000
linkpoint[0].trans_rot                   TRANS_X
linkpoint[0].visible                     0
linkpoint[0].inverse                     0
:
```

2.3.3   Assigned logical axis number (P-SECO-00042)

P-SECO-00042	Assigned logical axis number
Description	Logical axis number of the axis which causes a movement of the new node point (P-SECO-00040 [▸ 18]). The type and method of the axis link is specified in the parameter P-SECO-00047 [▸ 21].
Parameter	input_log_ax_nr
Data type	UNS16
Data range	0 ≤ P-SECO-00042 ≤ MAX(UNS16)
Dimension	----
Default value	0
Remarks	

Example taken from list:

```
:
linkpoint[0].name                               X_ACHSE
linkpoint[0].mountpoint                         GROUND
linkpoint[0].input_log_ax_nr                    1
linkpoint[0].cs_init.position[0]                0.000000
linkpoint[0].cs_init.position[1]                0.000000
linkpoint[0].cs_init.position[2]                0.000000
linkpoint[0].cs_init.orientation[0][0]          1.000000
linkpoint[0].cs_init.orientation[0][1]          0.000000
linkpoint[0].cs_init.orientation[0][2]          0.000000
linkpoint[0].cs_init.orientation[1][0]          0.000000
linkpoint[0].cs_init.orientation[1][1]          1.000000
linkpoint[0].cs_init.orientation[1][2]          0.000000
linkpoint[0].cs_init.orientation[2][0]          0.000000
linkpoint[0].cs_init.orientation[2][1]          0.000000
linkpoint[0].cs_init.orientation[2][2]          1.000000
linkpoint[0].trans_rot                          TRANS_X
linkpoint[0].visible                            0
linkpoint[0].inverse                            0
:
```

2.3.4 Orientation of the node point (linkpoint[i].cs_init.*)

The structure "linkpoint[i].cs_init.*" defines the orientation of the node point where the number "i" is relative to the previous node point defined in the list. The orientation is defined by relative translation and rotation (in matrix notation) in relation to the previous node point specified as the mount point (P-SECO-00041 [▷ 19]).^

The translation and rotation of a graphical object is executed in the coordinate system structure. This consists of the parameters P-SECO-00025 [▷ 11] and P-SECO-00026 [▷ 12].

```
linkpoint[i].cs_init.position[j] where j =0,1,2
linkpoint[i].cs_init.orientation[j][k] where j and k =0,1,2
```

Example taken from list:

```
:
linkpoint[0].name                               X_ACHSE
linkpoint[0].mountpoint                         GROUND
linkpoint[0].input_log_ax_nr                    1
linkpoint[0].cs_init.position[0]                0.000000
linkpoint[0].cs_init.position[1]                0.000000
linkpoint[0].cs_init.position[2]                0.000000
linkpoint[0].cs_init.orientation[0][0]          1.000000
linkpoint[0].cs_init.orientation[0][1]          0.000000
linkpoint[0].cs_init.orientation[0][2]          0.000000
linkpoint[0].cs_init.orientation[1][0]          0.000000
linkpoint[0].cs_init.orientation[1][1]          1.000000
linkpoint[0].cs_init.orientation[1][2]          0.000000
linkpoint[0].cs_init.orientation[2][0]          0.000000
linkpoint[0].cs_init.orientation[2][1]          0.000000
linkpoint[0].cs_init.orientation[2][2]          1.000000
linkpoint[0].trans_rot                          TRANS_X
linkpoint[0].visible                            0
linkpoint[0].inverse                            0
:
```

If no rotation is to be executed, specify the unit matrix

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(see example below).

Example taken from list for unit matrix:

```
:
linkpoint[0].name                          X_ACHSE
linkpoint[0].mountpoint                    GROUND
linkpoint[0].input_log_ax_nr               1
linkpoint[0].cs_init.position[0]           0.000000
linkpoint[0].cs_init.position[1]           0.000000
linkpoint[0].cs_init.position[2]           0.000000
linkpoint[0].cs_init.orientation[0][0]     1.000000
linkpoint[0].cs_init.orientation[0][1]     0.000000
linkpoint[0].cs_init.orientation[0][2]     0.000000
linkpoint[0].cs_init.orientation[1][0]     0.000000
linkpoint[0].cs_init.orientation[1][1]     1.000000
linkpoint[0].cs_init.orientation[1][2]     0.000000
linkpoint[0].cs_init.orientation[2][0]     0.000000
linkpoint[0].cs_init.orientation[2][1]     0.000000
linkpoint[0].cs_init.orientation[2][2]     1.000000
linkpoint[0].trans_rot                     TRANS_X
linkpoint[0].visible                       0
linkpoint[0].inverse                       0
:
```

2.3.5 Type of axis movement (P-SECO-00047)

P-SECO-00047	Type of axis movement
Description	This parameter defines whether an axis movement P-SECO-00042 [▶ 19] causes a translation in X / Y / Z or a rotation about X / Y / Z.
Parameter	linkpoint[i]. trans_rot
Data type	UNS16
Data range	0 : NONE 1 : TRANS_X, 2 : TRANS_Y 3 : TRANS_Z 4 : ROT_X 5 : ROT_Y 6 : ROT_Z
Dimension	---
Default value	0
Remarks	

Example taken from list:

```
:
```

```
linkpoint[0].name                              X_ACHSE
linkpoint[0].mountpoint                        GROUND
linkpoint[0].input_log_ax_nr                   1
linkpoint[0].cs_init.position[0]               0.000000
linkpoint[0].cs_init.position[1]               0.000000
linkpoint[0].cs_init.position[2]               0.000000
linkpoint[0].cs_init.orientation[0][0]         1.000000
linkpoint[0].cs_init.orientation[0][1]         0.000000
linkpoint[0].cs_init.orientation[0][2]         0.000000
linkpoint[0].cs_init.orientation[1][0]         0.000000
linkpoint[0].cs_init.orientation[1][1]         1.000000
linkpoint[0].cs_init.orientation[1][2]         0.000000
linkpoint[0].cs_init.orientation[2][0]         0.000000
linkpoint[0].cs_init.orientation[2][1]         0.000000
linkpoint[0].cs_init.orientation[2][2]         1.000000
linkpoint[0].trans_rot                         TRANS_X
linkpoint[0].visible                           0
linkpoint[0].inverse                           0
          :
```

2.3.6   Handling node point movement (P-SECO-00048)

P-SECO-00048	Handling node point movement
Description	This parameter defines whether the movement of a node point should be output to the controller interface or not (so-called DataFactory).
Parameter	linkpoint[i].visible
Data type	BOOLEAN
Data range	0 - Movements of the node point are cyclically displayed when logged (e.g. via Data Factory).
	1 - Movements of the node point are cyclically displayed when logged (e.g. via Data Factory).
Dimension	----
Default value	0
Remarks	P-SECO-00048 is always set with the element GROUND..

Example taken from list:

```
          :
linkpoint[0].name                              X_ACHSE
linkpoint[0].mountpoint                        GROUND
linkpoint[0].input_log_ax_nr                   1
linkpoint[0].cs_init.position[0]               0.000000
linkpoint[0].cs_init.position[1]               0.000000
linkpoint[0].cs_init.position[2]               0.000000
linkpoint[0].cs_init.orientation[0][0]         1.000000
linkpoint[0].cs_init.orientation[0][1]         0.000000
linkpoint[0].cs_init.orientation[0][2]         0.000000
linkpoint[0].cs_init.orientation[1][0]         0.000000
linkpoint[0].cs_init.orientation[1][1]         1.000000
linkpoint[0].cs_init.orientation[1][2]         0.000000
linkpoint[0].cs_init.orientation[2][0]         0.000000
linkpoint[0].cs_init.orientation[2][1]         0.000000
linkpoint[0].cs_init.orientation[2][2]         1.000000
```

```
linkpoint[0].trans_rot                              TRANS_X
linkpoint[0].visible                                0
linkpoint[0].inverse                                0
    :
```

2.3.7 Effect of axis movement on node point movement (P-SECO-00049)

P-SECO-00049	Effect of axis movement on node point movement
Description	This parameter defines whether the programmed axis movements should have an inverting effect on the translation or rotation of the node point.
Parameter	linkpoint[i].inverse
Data type	BOOLEAN
Data range	0 - Axis movements have no inverting effect (default).
	1 - Axis movements have an inverting effect.
Dimension	----
Default value	0
Remarks	

Example taken from list:

```
    :
linkpoint[0].name                                   X_ACHSE
linkpoint[0].mountpoint                             GROUND
linkpoint[0].input_log_ax_nr                        1
linkpoint[0].cs_init.position[0]                    0.000000
linkpoint[0].cs_init.position[1]                    0.000000
linkpoint[0].cs_init.position[2]                    0.000000
linkpoint[0].cs_init.orientation[0][0]              1.000000
linkpoint[0].cs_init.orientation[0][1]              0.000000
linkpoint[0].cs_init.orientation[0][2]              0.000000
linkpoint[0].cs_init.orientation[1][0]              0.000000
linkpoint[0].cs_init.orientation[1][1]              1.000000
linkpoint[0].cs_init.orientation[1][2]              0.000000
linkpoint[0].cs_init.orientation[2][0]              0.000000
linkpoint[0].cs_init.orientation[2][1]              0.000000
linkpoint[0].cs_init.orientation[2][2]              1.000000
linkpoint[0].trans_rot                              TRANS_X
linkpoint[0].visible                                0
linkpoint[0].inverse                                0
    :
```

2.4 Data of a group (group[i].*)

This structure defines the data of a group. Several graphical objects are included in a group. This is intended to simplify programming. The number of groups is limited to 64.

The number of graphical objects in a group is limited to 128.

Structure name	Index
group[i].	0 ≤ i ≤ 63

2.4.1   Name of the group (P-SECO-00060)

P-SECO-00060	Name of the group
Description	This parameter defines the name of a group.
Parameter	group[i].name
Data type	STRING
Data range	Maximum of 39 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

```
       :
group[0].name              Machine
group[0].gobject[0]        Arm
group[0].gobject[1]        Milling cutter
       :
```

2.4.2   Names of the graphical objects (P-SECO-00061)

P-SECO-00061	Names of the graphical objects
Description	This parameter specifies the name of a graphical object that should be part of the group.
Parameter	group[i].gobject[j] where j = 0 to 127
Data type	STRING
Data range	Maximum of 39 characters
Dimension	----
Default value	*
Remarks	* Note: The default value of variables is a blank string.

```
       :
group[0].name              Machine
group[0].gobject[0]        Arm
group[0].gobject[1]        Milling cutter
       :
```

3   Example of assignment of a scene and collision list

```
# *****************************************
#
# Configuration for scene definition
# and collision monitoring
#
collision_detected_outputfile D:\Scene\Actual\collision_detected.lis
path_to_geometry D:\Scene\Listen\Gobjects\Rough\
use_collision_detection 1
#
#**********************************************************
# Gobjects
#**********************************************************
gobject[0].name Ausleger_Einfach
gobject[0].filename Ausleger_01
gobject[0].linkpoint A_AXIS
gobject[0].cs.position[0] 500000
gobject[0].cs.position[1] 0
gobject[0].cs.position[2] 0
gobject[0].cs.orientation[0][0] 1
gobject[0].cs.orientation[0][1] 0
gobject[0].cs.orientation[0][2] 0
gobject[0].cs.orientation[1][0] 0
gobject[0].cs.orientation[1][1] 1
gobject[0].cs.orientation[1][2] 0
gobject[0].cs.orientation[2][0] 0
gobject[0].cs.orientation[2][1] 0
gobject[0].cs.orientation[2][2] 1
gobject[0].key[0] Colour
gobject[0].value[0] red
gobject[0].key[1] Shading
gobject[0].value[1] grey

gobject[1].name Milling cutter
gobject[1].filename toolbody102
gobject[1].linkpoint A_AXIS
gobject[1].cs.position[0] 1000000
gobject[1].cs.position[1] 0
gobject[1].cs.position[2] 0
gobject[1].cs.orientation[0][0] 1
gobject[1].cs.orientation[0][1] 0
gobject[1].cs.orientation[0][2] 0
gobject[1].cs.orientation[1][0] 0
gobject[1].cs.orientation[1][1] 1
gobject[1].cs.orientation[1][2] 0
gobject[1].cs.orientation[2][0] 0
gobject[1].cs.orientation[2][1] 0
gobject[1].cs.orientation[2][2] 1
gobject[1].key[0] Colour
gobject[1].value[0] blue
gobject[1].key[1] Shading
gobject[1].value[1] true

#**********************************************************
# Linkpoints
#**********************************************************
linkpoint[0].name X_AXIS
linkpoint[0].mountpoint GROUND
linkpoint[0].input_log_ax_nr 1
linkpoint[0].cs_init.position[0] 0.000000
linkpoint[0].cs_init.position[1] 0.000000
linkpoint[0].cs_init.position[2] 0.000000
```

```
linkpoint[0].cs_init.orientation[0][0] 1.000000
linkpoint[0].cs_init.orientation[0][1] 0.000000
linkpoint[0].cs_init.orientation[0][2] 0.000000
linkpoint[0].cs_init.orientation[1][0] 0.000000
linkpoint[0].cs_init.orientation[1][1] 1.000000
linkpoint[0].cs_init.orientation[1][2] 0.000000
linkpoint[0].cs_init.orientation[2][0] 0.000000
linkpoint[0].cs_init.orientation[2][1] 0.000000
linkpoint[0].cs_init.orientation[2][2] 1.000000
linkpoint[0].trans_rot TRANS_X
linkpoint[0].visible 0
linkpoint[0].inverse 0
#*************************************************************
linkpoint[1].name Y_AXIS
linkpoint[1].mountpoint X_AXIS
linkpoint[1].input_log_ax_nr 2
linkpoint[1].cs_init.position[0] 0.000000
linkpoint[1].cs_init.position[1] 0.000000
linkpoint[1].cs_init.position[2] 0.000000
linkpoint[1].cs_init.orientation[0][0] 1.000000
linkpoint[1].cs_init.orientation[0][1] 0.000000
linkpoint[1].cs_init.orientation[0][2] 0.000000
linkpoint[1].cs_init.orientation[1][0] 0.000000
linkpoint[1].cs_init.orientation[1][1] 1.000000
linkpoint[1].cs_init.orientation[1][2] 0.000000
linkpoint[1].cs_init.orientation[2][0] 0.000000
linkpoint[1].cs_init.orientation[2][1] 0.000000
linkpoint[1].cs_init.orientation[2][2] 1.000000
linkpoint[1].trans_rot TRANS_Y
linkpoint[1].visible 1
linkpoint[1].inverse 0
#*************************************************************
linkpoint[2].name Z_AXIS
linkpoint[2].mountpoint Y_AXIS
linkpoint[2].input_log_ax_nr 2
linkpoint[2].cs_init.position[0] 0.000000
linkpoint[2].cs_init.position[1] 0.000000
linkpoint[2].cs_init.position[2] 0.000000
linkpoint[2].cs_init.orientation[0][0] 1.000000
linkpoint[2].cs_init.orientation[0][1] 0.000000
linkpoint[2].cs_init.orientation[0][2] 0.000000
linkpoint[2].cs_init.orientation[1][0] 0.000000
linkpoint[2].cs_init.orientation[1][1] 1.000000
linkpoint[2].cs_init.orientation[1][2] 0.000000
linkpoint[2].cs_init.orientation[2][0] 0.000000
linkpoint[2].cs_init.orientation[2][1] 0.000000
linkpoint[2].cs_init.orientation[2][2] 1.000000
linkpoint[2].trans_rot TRANS_Z
linkpoint[2].visible 1
linkpoint[2].inverse 0
#*************************************************************
linkpoint[3].name A_AXIS
linkpoint[3].mountpoint Z_AXIS
linkpoint[3].input_log_ax_nr 2
linkpoint[3].cs_init.position[0] 0.000000
linkpoint[3].cs_init.position[1] 0.000000
linkpoint[3].cs_init.position[2] 0.000000
linkpoint[3].cs_init.orientation[0][0] 1.000000
linkpoint[3].cs_init.orientation[0][1] 0.000000
linkpoint[3].cs_init.orientation[0][2] 0.000000
linkpoint[3].cs_init.orientation[1][0] 0.000000
linkpoint[3].cs_init.orientation[1][1] 1.000000
linkpoint[3].cs_init.orientation[1][2] 0.000000
```

```
linkpoint[3].cs_init.orientation[2][0] 0.000000
linkpoint[3].cs_init.orientation[2][1] 0.000000
linkpoint[3].cs_init.orientation[2][2] 1.000000
linkpoint[3].trans_rot ROT_X
linkpoint[3].visible 1
linkpoint[3].inverse 0
#*************************************************************
linkpoint[4].name C_AXIS
linkpoint[4].mountpoint A_AXIS
linkpoint[4].input_log_ax_nr 2
linkpoint[4].cs_init.position[0] 0.000000
linkpoint[4].cs_init.position[1] 0.000000
linkpoint[4].cs_init.position[2] 0.000000
linkpoint[4].cs_init.orientation[0][0] 1.000000
linkpoint[4].cs_init.orientation[0][1] 0.000000
linkpoint[4].cs_init.orientation[0][2] 0.000000
linkpoint[4].cs_init.orientation[1][0] 0.000000
linkpoint[4].cs_init.orientation[1][1] 1.000000
linkpoint[4].cs_init.orientation[1][2] 0.000000
linkpoint[4].cs_init.orientation[2][0] 0.000000
linkpoint[4].cs_init.orientation[2][1] 0.000000
linkpoint[4].cs_init.orientation[2][2] 1.000000
linkpoint[4].trans_rot ROT_Z
linkpoint[4].visible 1
linkpoint[4].inverse 0

#*************************************************************
# Groups
#*************************************************************
group[0].name Machine
group[0].gobject[0] Arm
group[0].gobject[1] Milling cutter
#*************************************************************

End
```

4 Appendix

4.1 References

[PROG] Programming manual

4.2 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

---

i | **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

Keyword index

P