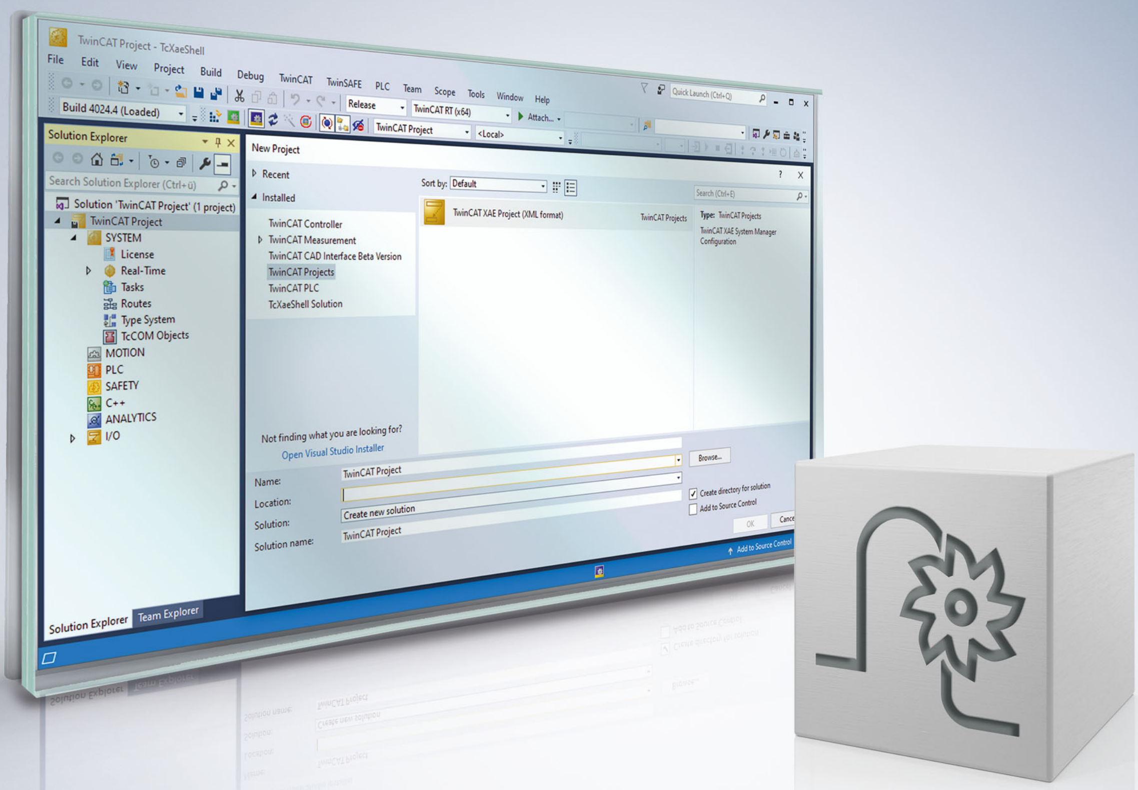
BECKHOFF New Automation Technology

Functional description | EN

TF5292 | TwinCAT 3 CNC

EDM Plus



9/6/2024 | Version: 1.00

Notes on the documentation

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning the components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.
We reserve the right to revise and change the documentation at any time and without prior announcement.
No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH.
Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT technology is patent protected, in particular by the following applications and patents:
EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702
with corresponding applications or registrations in various other countries.

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

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

1. 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.

NOTICE

Restriction or error

This icon describes restrictions or warns of errors.

Tips and other notes

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

General example

Example that clarifies the text.

NC programming example

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

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

List of figures

1 Overview

Task

Electrical Discharge Machining (EDM) is a material removal process for electrically conductive materials. Material removal is based on a process of electrical discharge. Erosion includes wire erosion and die sinking Both types represent a form of spark erosion.

In the die sinking process, the electrode usually has a 'negative shape' which must correspond to the material volume to be removed and the 'positive contour' to be obtained.

This function is available as of CNC Build V3.1.3108.5

This function is an additional option requiring a license.

Effectiveness/possible applications

Die sinking is mainly used for hard materials which would otherwise incur excessive tool costs using mechanical removal methods.

Parameterisation

The CNC must be configured with at least 3 channels in order to use the die sinking technology.

The parameter P-STUP-00033 [▸ 118] and specific channel parameters must be configured for each channel.

Programming

For a description of the extensive programming with its many variants, see the section Programming [▸ 58].

Additional related function descriptions:

- [FCT-C40] Contour look-ahead
- [FCT-C41] Insert stop command
- [FCT-C42] Real-time loops

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.

1.1 Definition of terms

Term	Description/explanation
ACS	Axis coordinate system
Down channel	Main channel that starts the erosion process and defines the machining mode.
Escape channel	Auxiliary channel that executes the electrode escape motion using a defined escape strategy.
Flushing	Long-term or fast electrode escape motion initiated by the PLC to flush the spark gap.
HLI	The High Level Interface (HLI) is an interface between the PLC and the CNC.
MCS	Machine coordinate system: First 3-dimensional Cartesian coordinate system.
Orbit channel	Auxiliary channel where the electrode orbit motion is programmed in the XY plane.
Orbit geometry	Geometry of the workpiece in the XY plane superimposed on the YZ plane during expansion.
Orbit disc	An orbit disc results when the down channel is stopped at a specific height (e.g. by inserting a stop label) and the orbit channel continues to move on its geometry.
PCS	The programming coordinate system is a Cartesian coordinate system which specifies positions in the NC program.
RZ plane	Plane consisting of radius and Z axis
TCP	Tool centre point
GEP	Geometry for erosion on the path
EDM	Electrical discharge machining

2 Description

EDM mainly differs from other forms of machining, such as milling, due to its method of material removal. Instead of mechanical removal, the electrode and the workpiece are electrically charged during erosion and the material is then melted and vaporised by a spark. The gap between the workpiece and the electrode must be continuously controlled during the process in order to achieve optimum machining results.

In die sinking, the workpiece is eroded by an electrode that has the negative shape to produce the workpiece. The workpiece is usually immersed in a tank containing a dielectric. In order to evacuate the removed material, the electrode is quickly retracted from the workpiece and the material is flushed out. This document describes how to set up a die sinking machine using the ISG kernel and what functions the ISG kernel offers for die sinking.

The figure below shows an overview of the individual components involved in motion control in the EDM process:

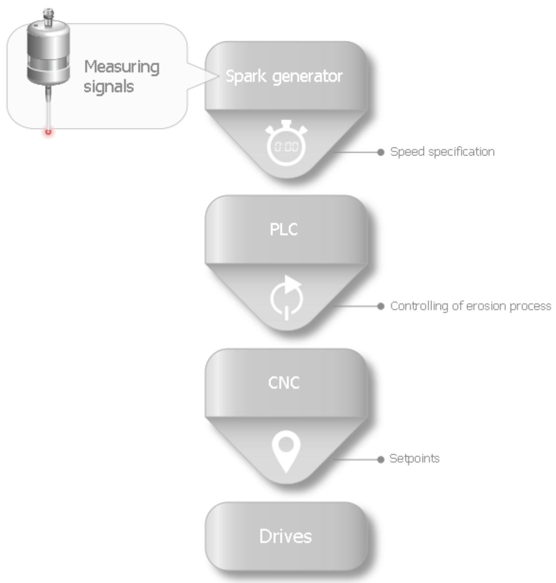


Fig. 1: Overview of components involved in the EDM process

The spark generator is the component that provides the specified velocities to control the gap based on the current values measured on the workpiece. The PLC uses this specified setting depending on the current erosion mode to transfer the necessary data to the CNC to guide the electrode motion. This generates the setpoints for the drives while maintaining maximum dynamics for optimum motion control to achieve optimum machining results.

Phases of the EDM process

The CNC provides 4 processing phases for the EDM process:

1. Positioning when the EDM process is not active
2. Erosion on the path (optional)
3. Erosion with orbiting (optional)
4. Electrode escape after the EDM process

When positioning, the electrode is moved to a position above the starting point for machining.

It can then be decided how the workpiece is to be machined. A distinction is made between erosion on the path, erosion with orbiting or erosion on the path followed by erosion with orbiting.

The machining processes differ in the way the electrode is guided. In erosion on the path, the electrode moves forwards and backwards along a programmed path.

In erosion with orbiting, the motion of the electrode is produced from two asynchronous or synchronised 2D geometries. The escape motion away from the current electrode point is calculated based on an escape strategy.

After EDM machining, the electrode can be retracted from the workpiece.

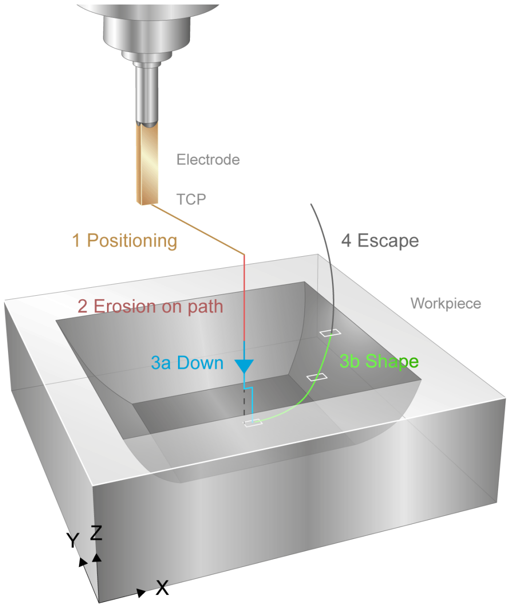


Fig. 2: The 4 phases of the EDM process

Electrode motion for orbiting

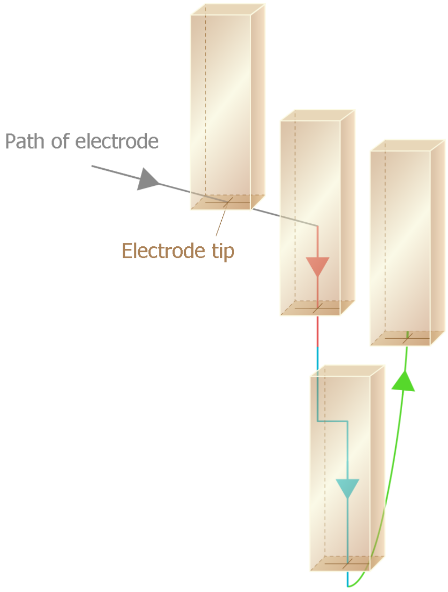


Fig. 3: Programming the TCP = motion of the electrode on the path

Guiding the electrode when orbiting is divided into 2 processes:

1. Down to the RZ plane
2. Orbit motion in the X-Y plane

These processes can be completely asynchronous or partially synchronised depending on the type of EDM used.

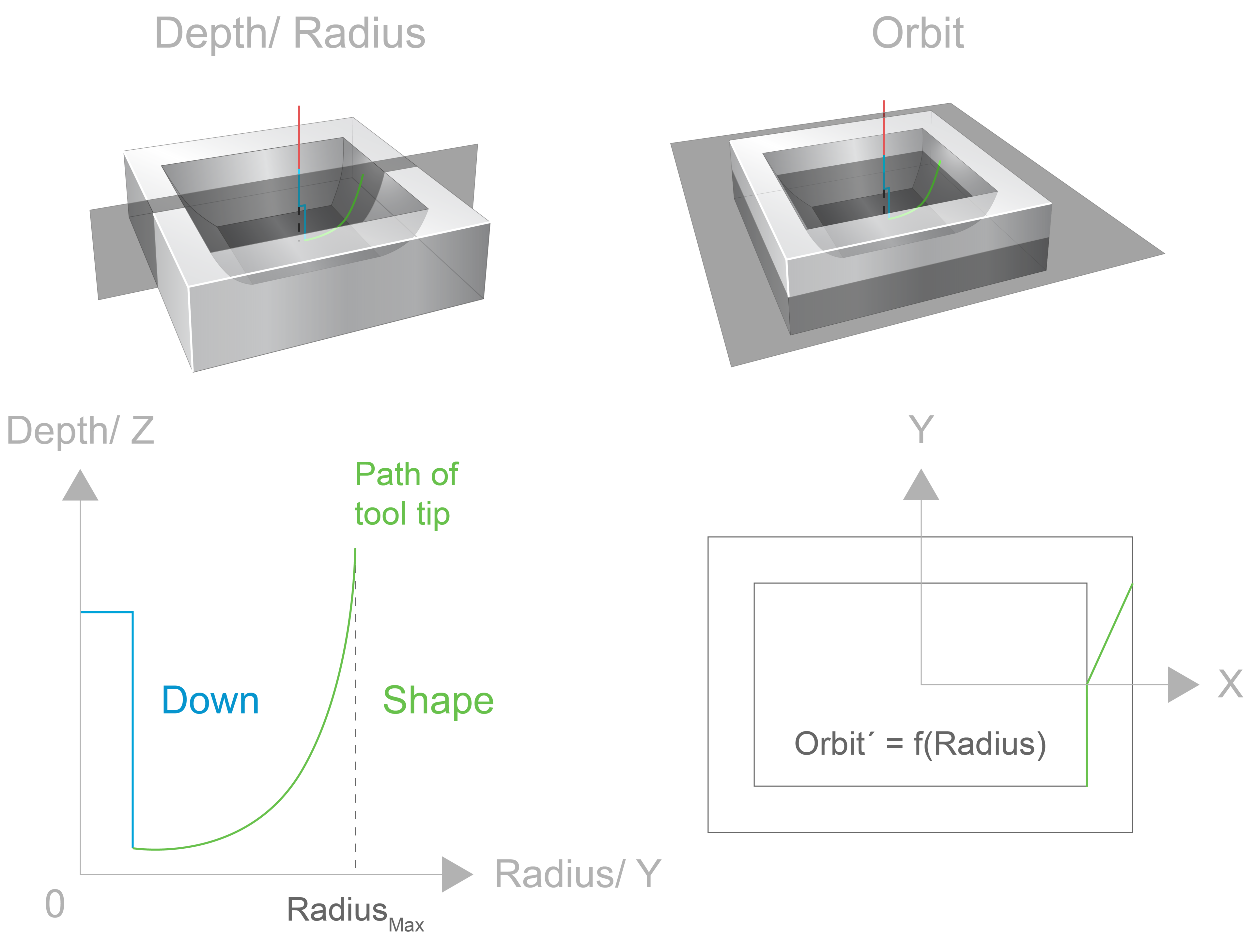


Fig. 4: Division into sinking and orbiting motions

A third process is activated for the escape motion of the electrode. It assumes the spark generator-controlled escape strategy and the flushing process in real time.

CNC architecture for die sinking

In order to fulfil the requirements of the EDM process, three NC channels are used in the CNC control system and together they control the axes of the EDM machine.

The three channels are shown in the figure below: In this document, they are referred to as follows:

1. Down channel
2. Orbit channel
3. Escape channel

Each channel is always assigned its own NC program and has its own status and position display. The down channel is the main channel controlling the orbit and escape channels and establishes a link between the channels.

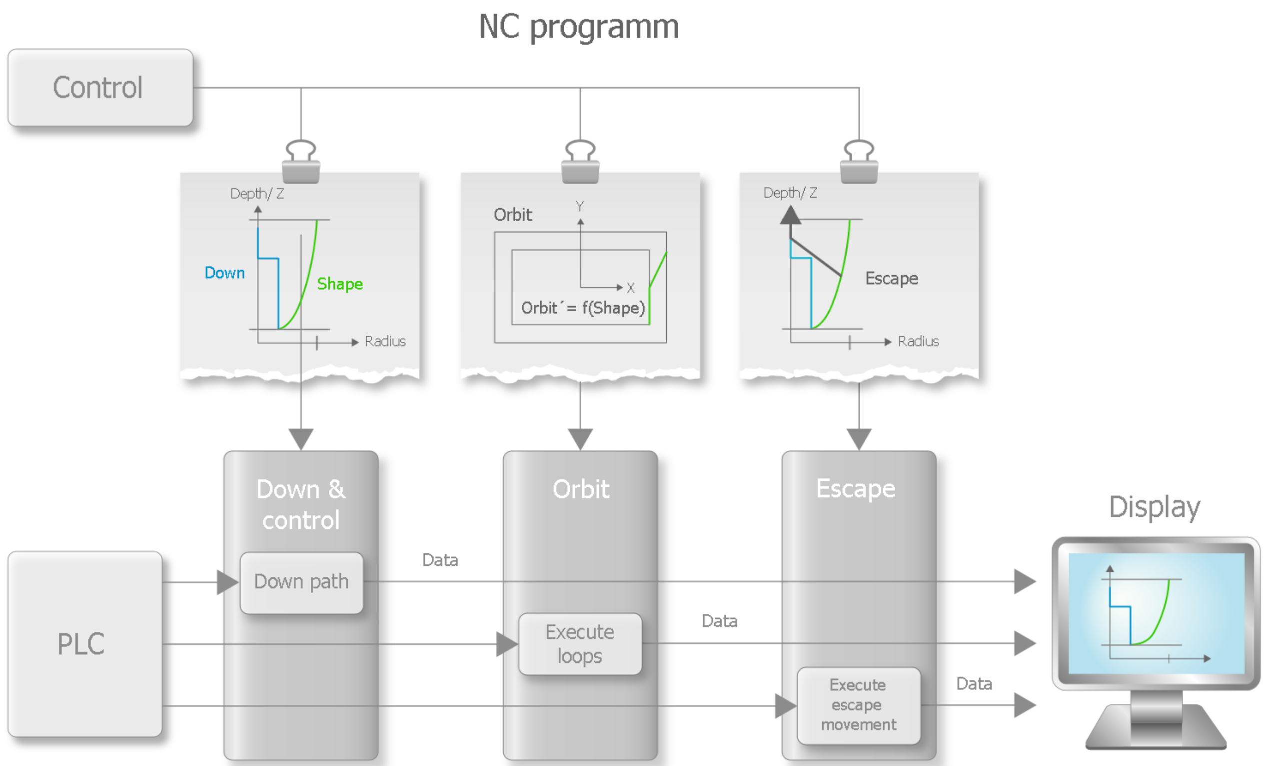


Fig. 5: The three die-sinking channels

The **down channel** assumes the task of the main channel and commands and starts the other two channels. Among other things, the geometry of positioning and erosion on the path are executed in their own NC program. During the orbiting phase, only a two-dimensional contour may be programmed (see figure above).

The **orbit channel** simultaneously performs the motion on its programmed geometry in an endless loop. The X and Y coordinates returned to the down channel depend on the current position in the orbit channel and on the radius transferred from the down channel. The resulting three-dimensional motion is generated by superimposing both of the two-dimensional motions.

The **escape channel** is programmed in the same coordinate system as the down channel. This channel controls the escape motion of the electrode. There are different strategies to calculate an escape path during program runtime based on the current position in the down channel. The escape channel moves along this escape path. This allows the gap size to be set irrespective of program progress in the down channel. The velocity of the escape channel, just like the velocity in the down channel, is a variable that is controlled by the process and is specified by a spark generator. In addition, the escape channel is used to initiate the flushing process by means of a jump movement to enable an optimised flushing process, fast escape and a quick restart on the escape path.

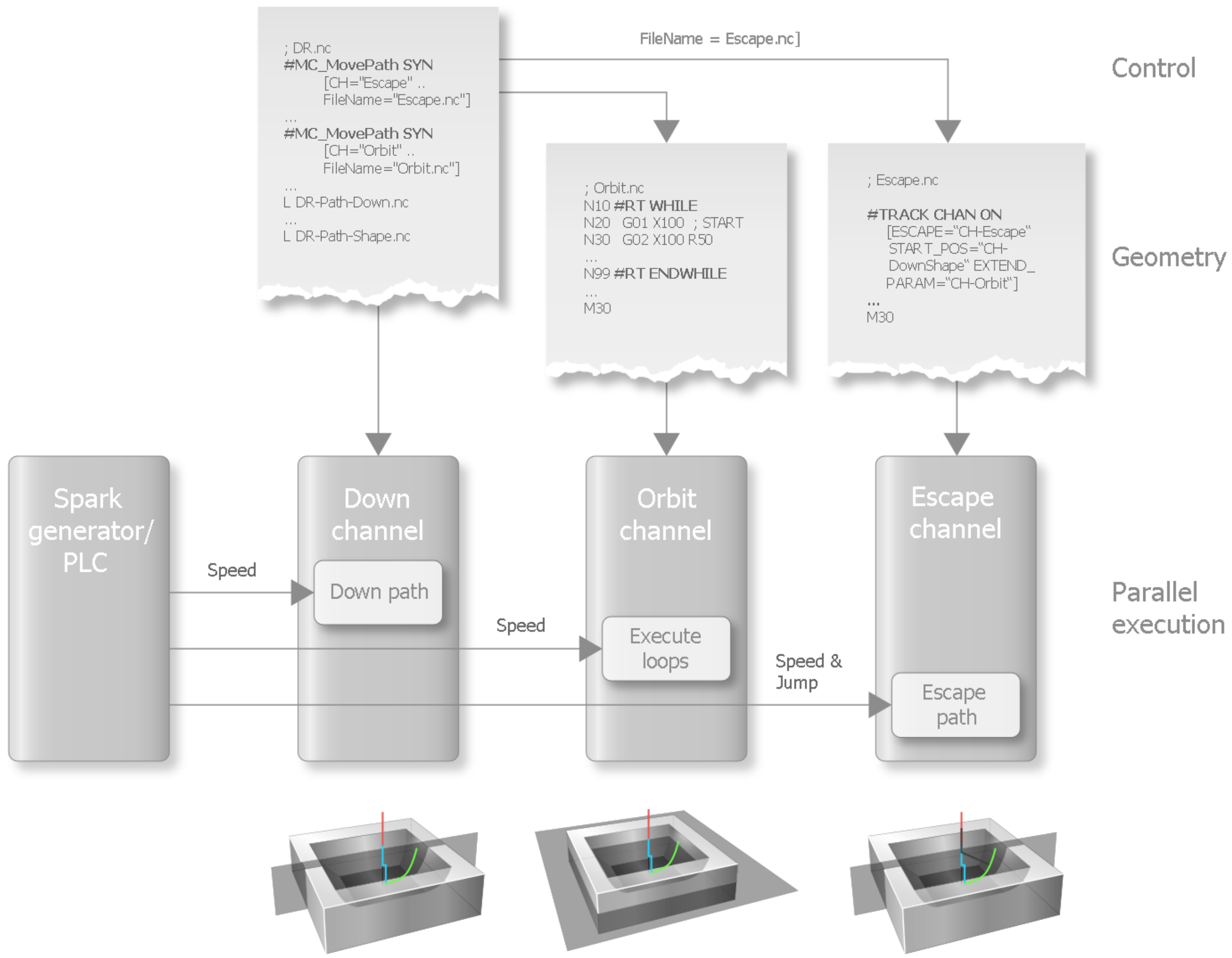


Fig. 6: NC programs of the three channels in the die-sinking process

The approach using separate channels allows progress in the RZ plane to be independent of the progress of the orbit motion and enables constant adjustment of the gap size using the escape channel.

It is also possible to move on an orbit disc at constant position in the down channel and at the current velocity in the orbit channel. By controlling the escape velocity, the gap distance and ultimately surface quality can be optimised at disc height (see Disc management).

Controlling the EDM process

In CNC, the EDM process is entirely controlled by the spark generator via the PLC. The PLC distributes the specified generator velocity via the external velocity interface (ext_command_speed [▸ 108] control unit) to the individual channels.

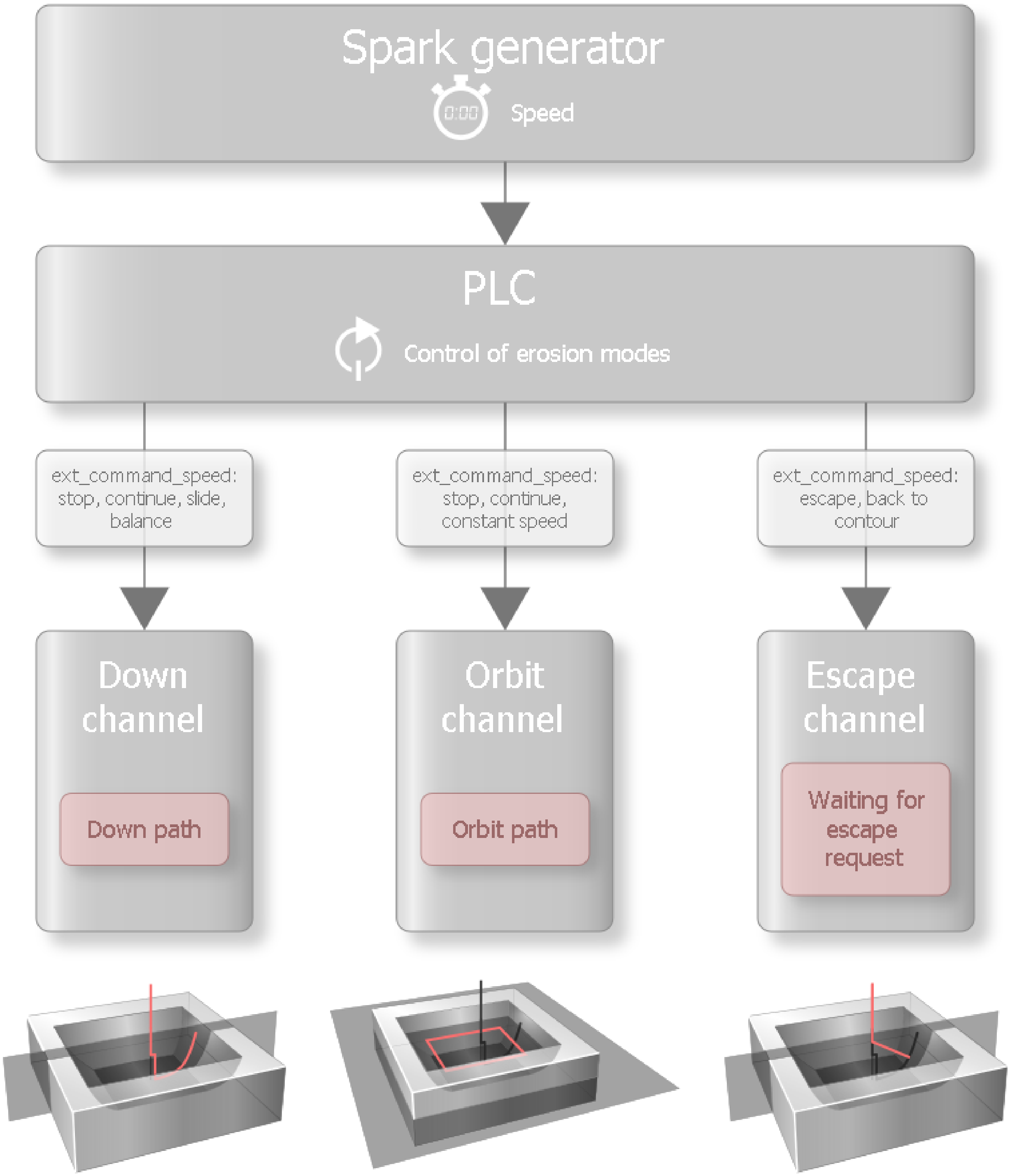


Fig. 7: Controlling the erosion modes

The distributed specified velocities provide the possibility of implementing different erosion modes using the same CNC architecture. Below are two examples and section 3.5 contains more examples that explain exactly how these erosion modes need to be programmed.

Alternating orbiting

Horizontal discs are generated at a specific velocity in the orbit channel and at a constant position in the down channel. Discs can be executed at different heights by motion in the down channel.

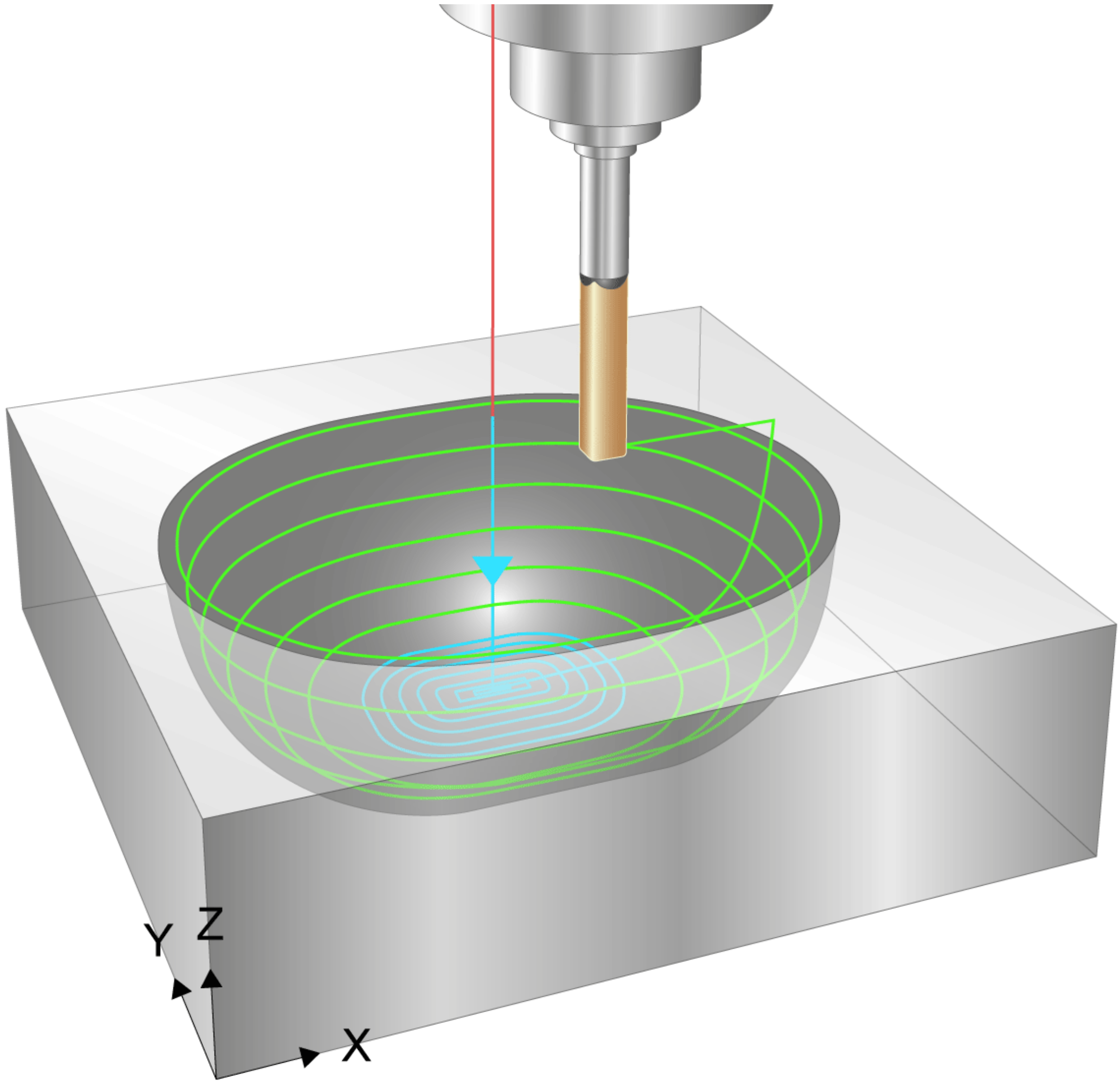


Fig. 8: Alternating orbiting

Star orbiting

As a result of the independent orbit channel, it is also possible to execute a motion on a perpendicular plane at constant position in the orbit and existing motion in the down channel, for example to obtain the required quality of surface finish at the corners of a geometry. A tool motion based on this machining strategy is shown in the figure below.

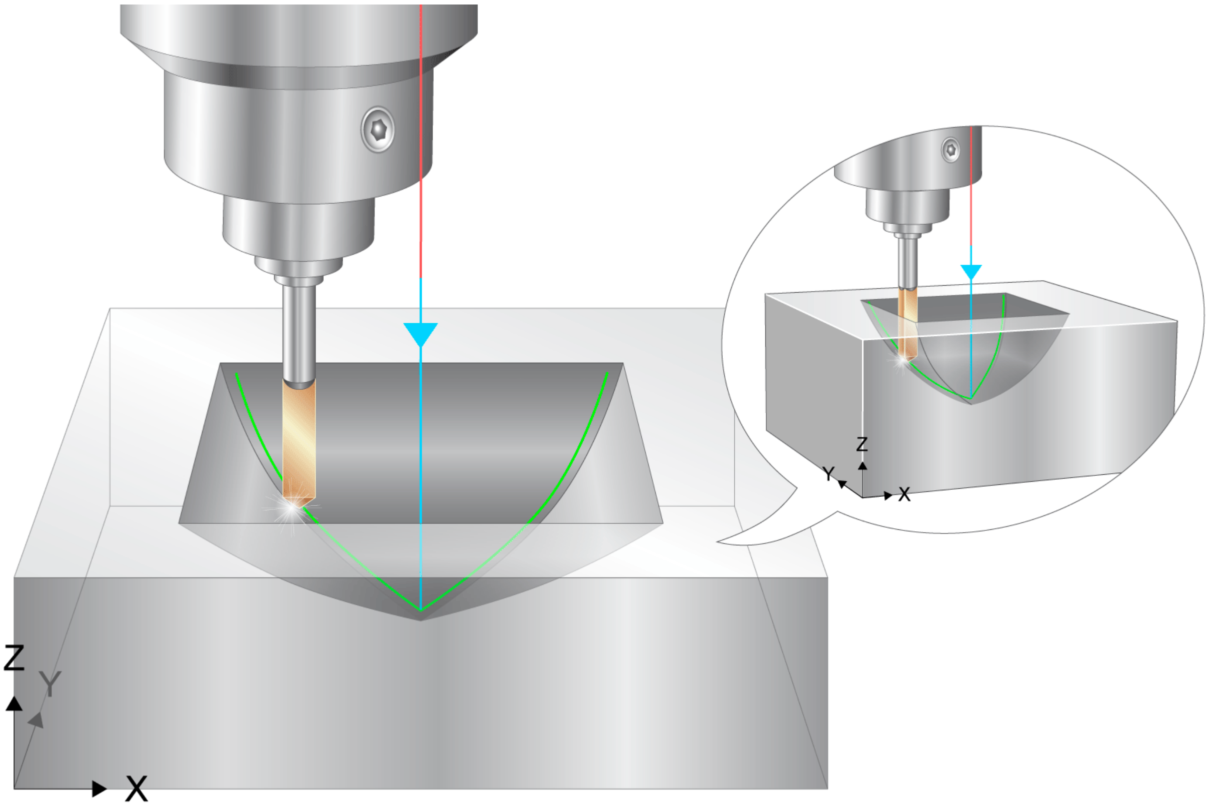


Fig. 9: Star orbiting

2.1 Down channel

A down channel has the function of a master channel. The channel couplings that it triggers implicitly initiate the different phases of the erosion process [▶ 12].

The channel couplings must be executed in a fixed sequence. The sequence for an EDM program is as follows:

1. Start the main NC program in the down channel.
2. Positioning
3. Activate the escape channel.
4. Start the erosion process on a path (optional).
5. Activate the orbit channel and start the erosion process for orbiting. (optional)
6. Deactivate the auxiliary channels.
7. Electrode escape and end of machining

NOTICE
When coupling/decoupling, the channels must be at standstill. If this is not the case, error ID 51021 is output.

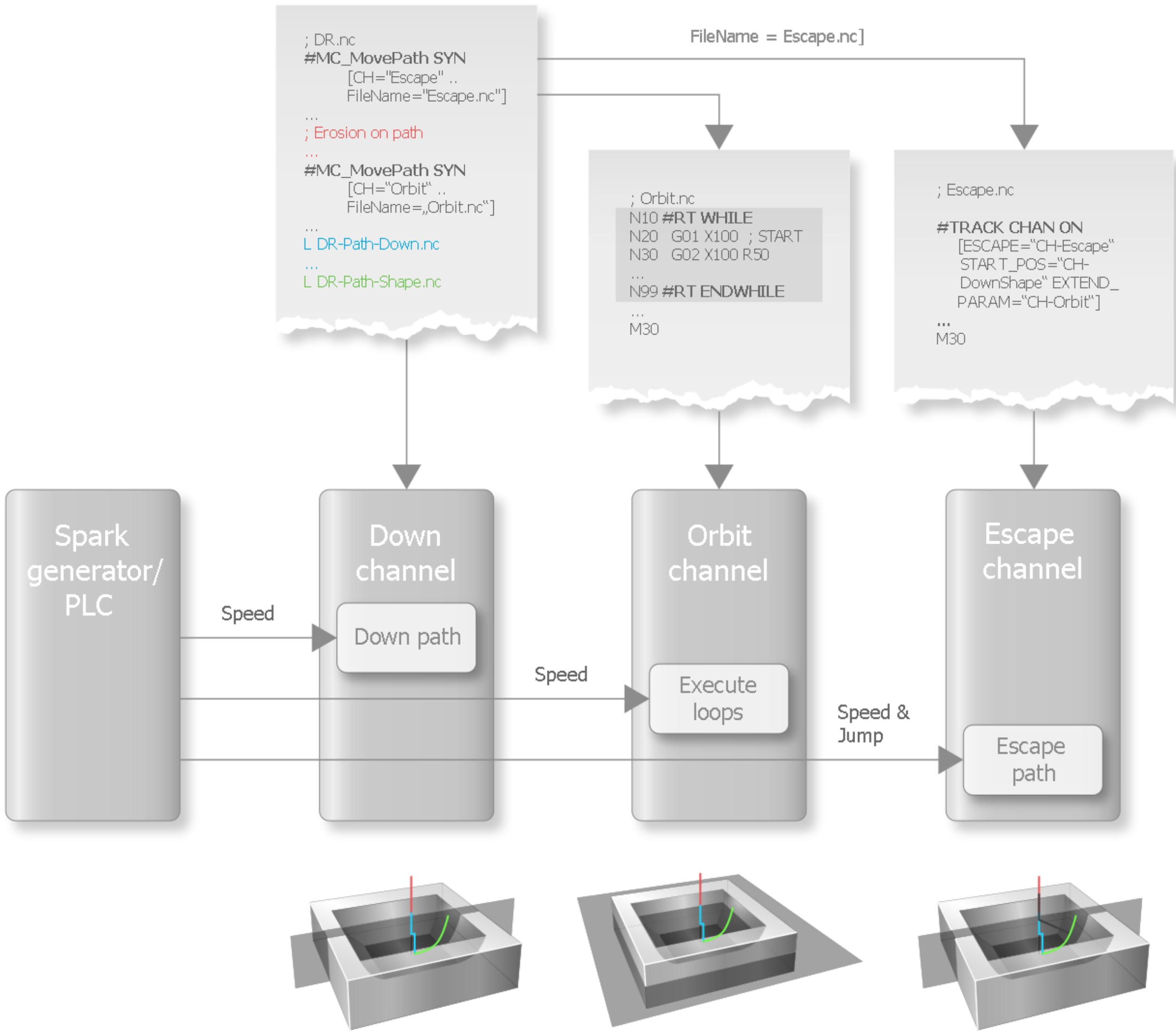


Fig. 10: Machining phases during die sinking

2.1.1 Positioning

Before the erosion process starts, the electrode must be positioned in the down channel above the workpiece. In this phase, the orbit and escape channels are inactive and the down channel acts as a standard NC channel. This means that all axes are physically moveable through the down channel by using coordinate systems and kinematic transformations.

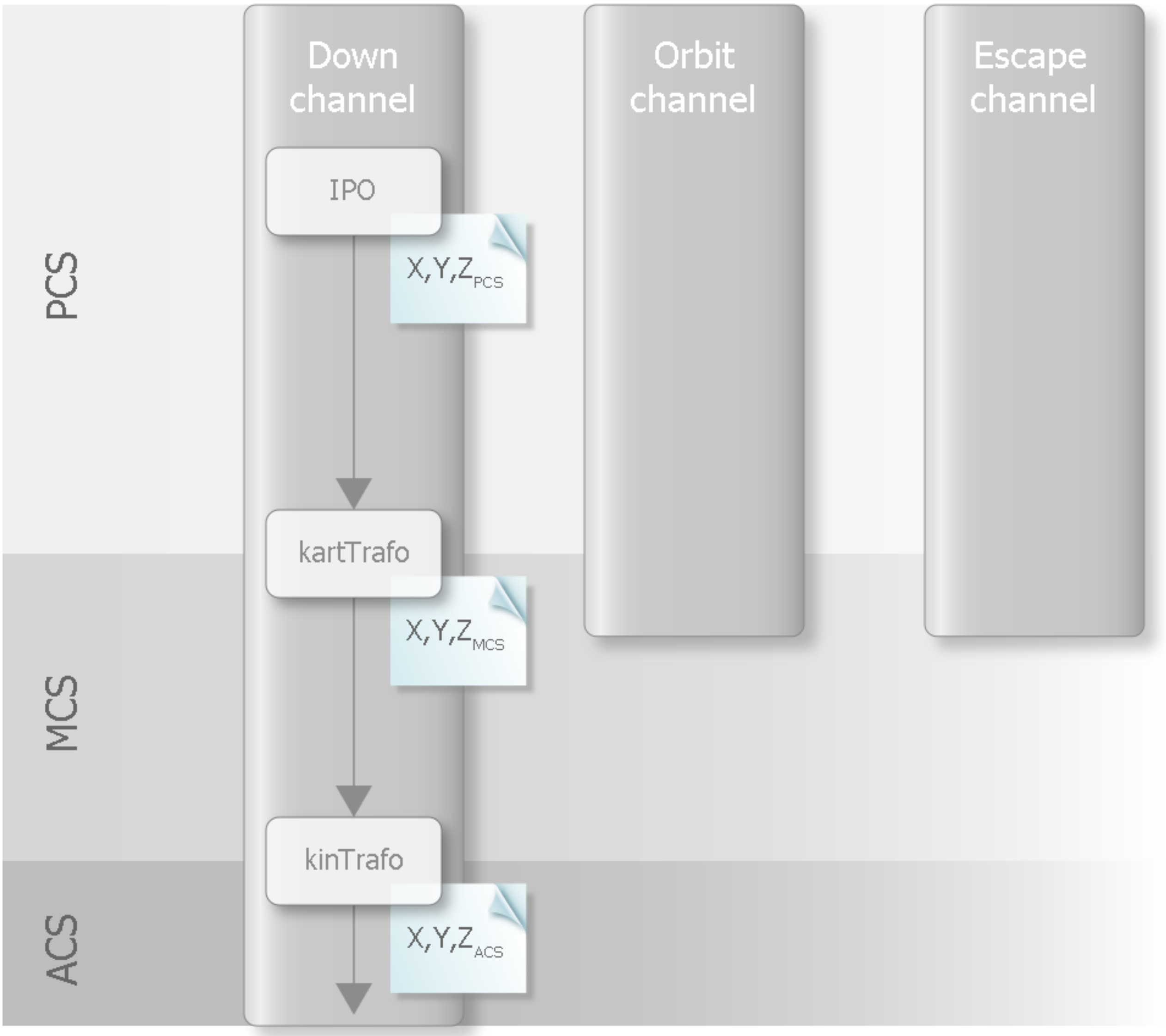


Fig. 11: Overview of down channel during positioning

2.1.2 Erosion on the path

Erosion on the path is the first phase when the erosion process is active. Since the escape channel executes the escape motions of the electrode, it must be activated before the geometry for erosion on the path (GEP).

Activating the escape channel implicitly means the start of the phase for erosion on the path. The orbit channel remains inactive throughout this entire phase. The escape channel can be executed by channel coupling with superimposition of the coordinates of all axes, erosion and flushing motions.

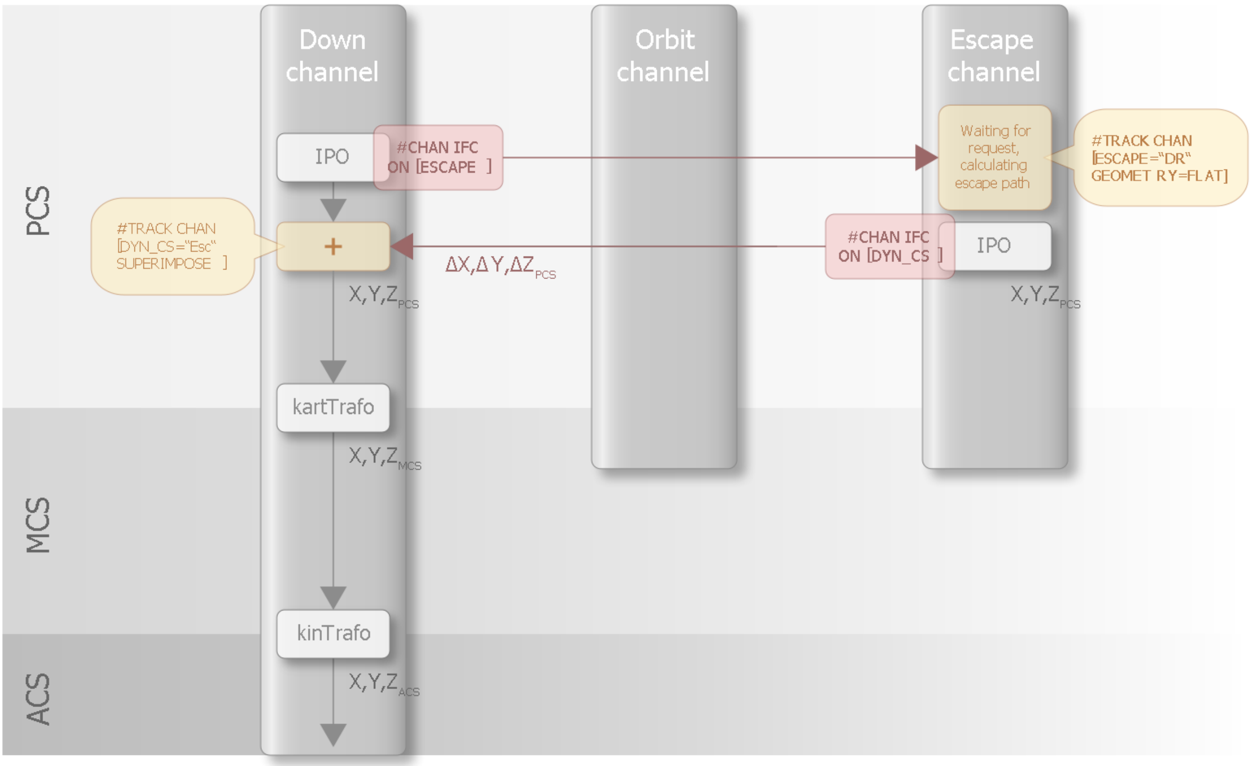


Fig. 12: Channel view for erosion on the path

2.1.3 Orbiting

Orbiting starts when the orbit channel is activated.

From this point on, the programming of the down channel also changes. When the orbit channel is active, the down channel is only programmed in a 2D plane (radius Z, Y-Z) since the orbit channel is superimposed in a 2D plane (X-Y) perpendicular to it. Superimposition generates a 3D motion. For superimposition, the orbit channel uses the radius supplied by the down channel and this scales the orbit geometry.

During the orbiting phase, the escape of the electrode or the flushing motion is controlled by rapidly lifting the electrode through the escape channel.

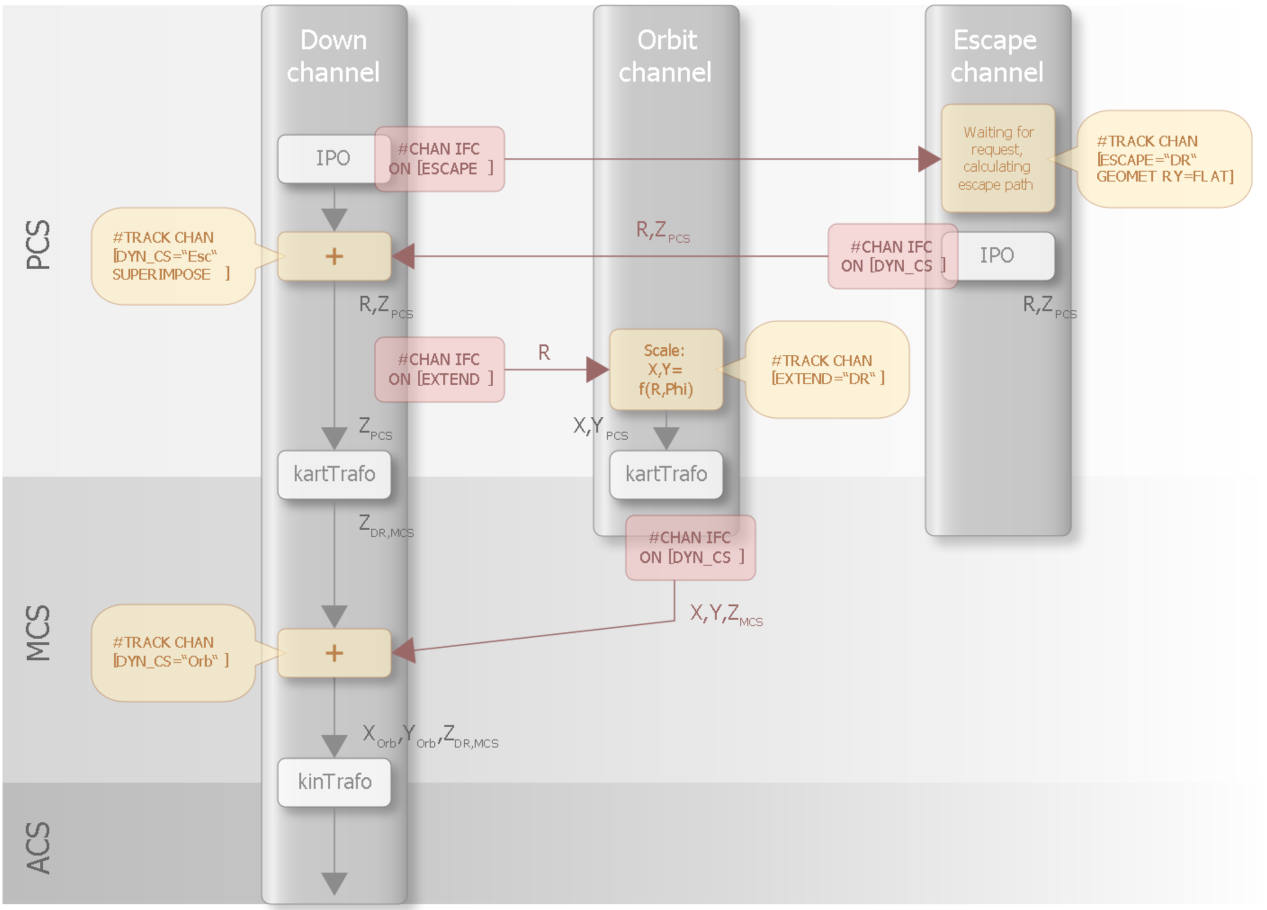


Fig. 13: Overview of channel couplings for orbiting

2.1.4 Ending the machining phase

After the workpiece is produced, the channel couplings are released and the system reverts to the same state as in the positioning phase.

The down channel reverts to a standard NC channel and programming can take place in 3D space. The electrode can then be extracted from the workpiece and moved to its end position.

However, it is also possible to execute a new erosion operation in the same NC program after the channel coupling is released. This must again start with positioning.

It is possible to end the current machining operation prematurely.

See section: Changing the generator settings [▶ 50]

2.1.5 Disc management

During the orbiting phase, the PLC can insert online motion stops ‘’Insert motion stops‘’ function [FCT-C41 Insert stop command, section: Description] during the EDM process. When the orbit channel rotates freely, the surface quality on this “disc” can be detected and optimised.

This function enables execution of the down and orbit geometries using different “discs”. The PLC can control contour accuracy and achieve the required machining quality by inserting these discs into the geometry of the down channel.

For example, the specified velocity of the spark generator can be used as a guide to measure the machining quality attained.

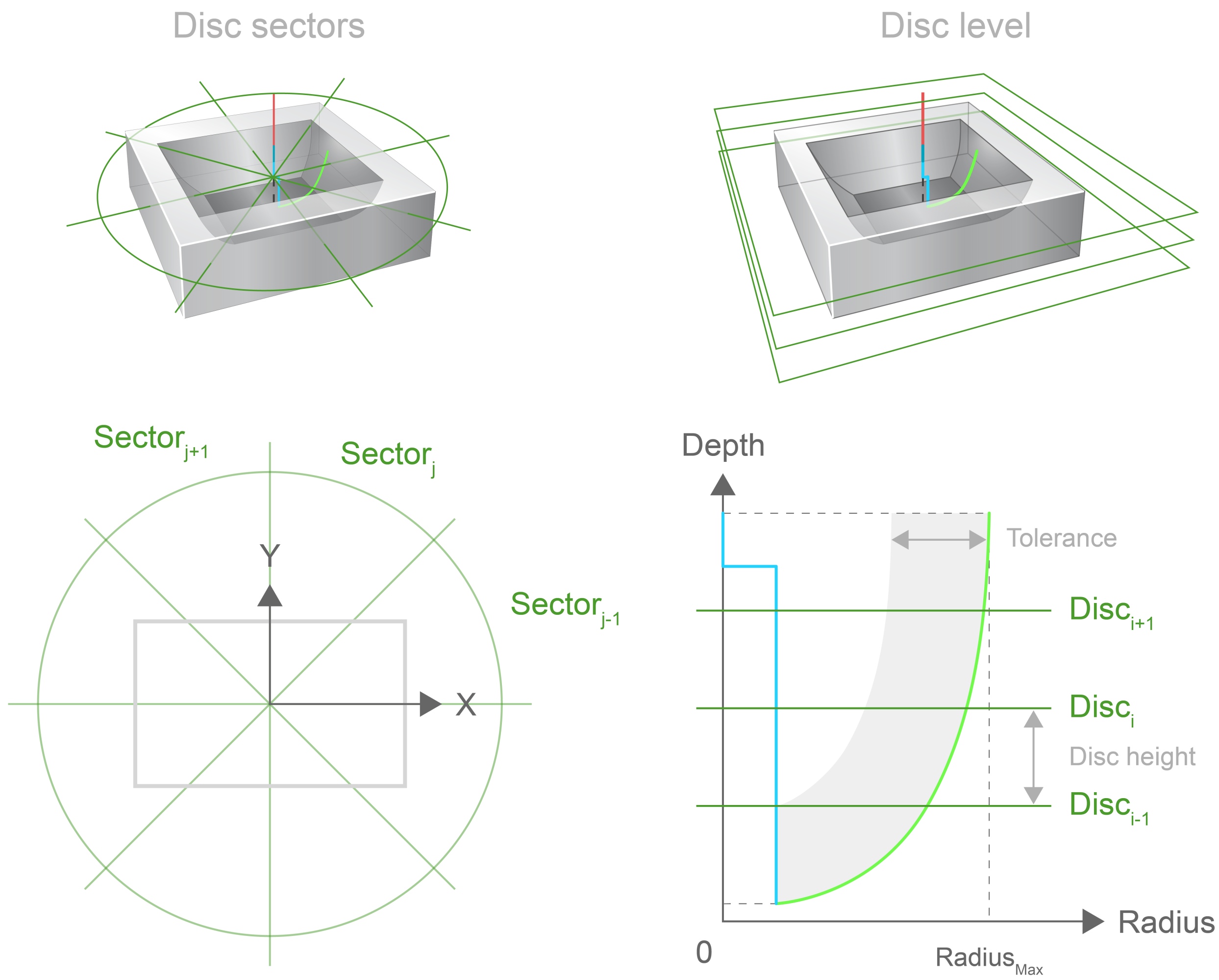


Fig. 14: Monitoring machining quality by inserting discs via the PLC

2.2 Orbit channel

The orbit channel has the task of moving the electrode in the X/Y plane. It therefore transforms the currently supplied radius into its programmed orbit geometry. A distinction is made in this transformation between scaling ($r_{curr} < r_{MaxScal}$) of the orbit geometry and an equidistant adaptation of the orbit geometry ($r_{curr} > r_{MaxScal}$). When the orbit channel is active, the down channel only moves on the R/Z plane. The 3D motion results from the superimposition of the orbit channel.

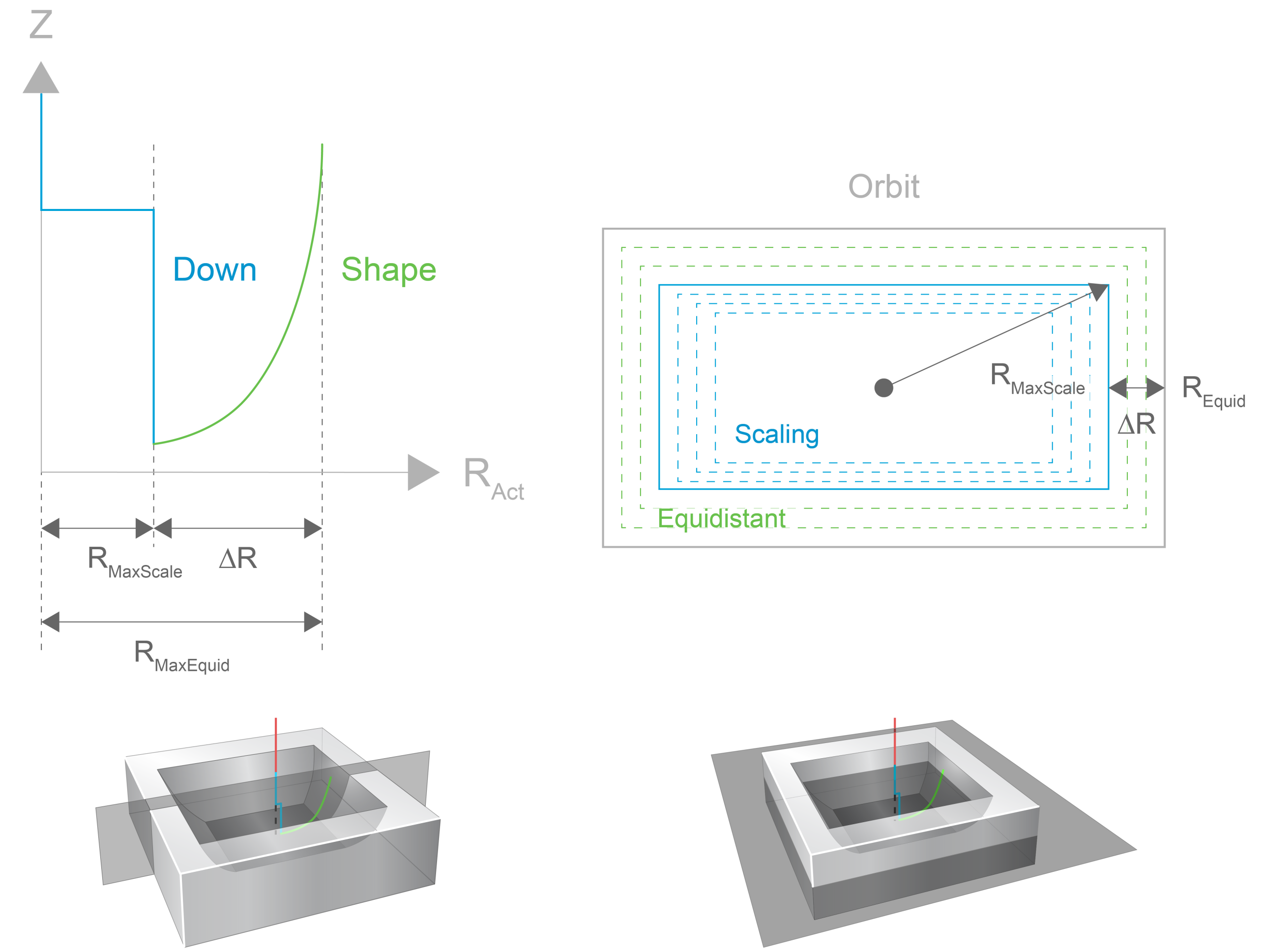


Fig. 15: Adapting the orbit geometry as a function of the current radius of the down channel

The X and Y coordinates superimposed by the orbit channel are dependent on the radius supplied by the down channel and the current PCS position on the geometry of the orbit channel. The channel interface shown in the figure below illustrates this.

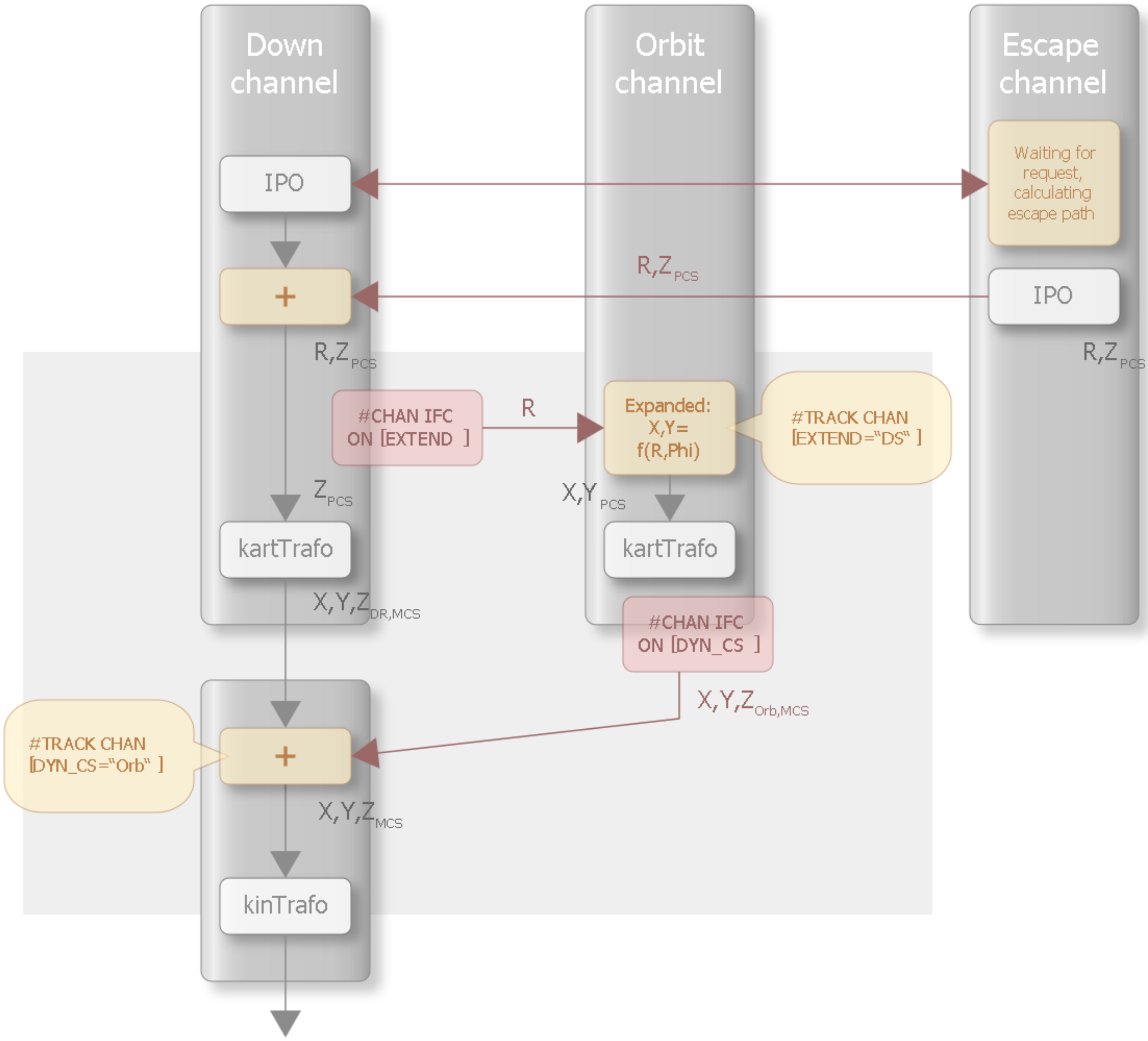


Fig. 16: Depiction of the channel interface

2.2.1 Orbit geometries

The contour programmed in the orbit channel must meet a number of requirements depending on the orbit strategy selected. Here, there are two strategies provided to determine the way in which the contour programmed in the orbit channel is reduced as a function of the radius.

Scaling: $0 <= R <= R_{MaxScal}$

The orbit scales the orbit geometry using the linear scaling factor = R/ $R_{MaxScal}$ in relation to the centre (0/0) as a function of the current radius.

Equidistant: $R_{MaxScal} < R <= R_{MaxEquid}$

A suitable equidistant to the program contour is calculated for the orbit geometry.

The following applies to the scaling and equidistant ranges: $0 <= R_{MaxScal} <= R_{MaxEquid}$.

Scaling can be deactivated by $R_{MaxScal}$ = 0. The equidistant range can be deactivated by $R_{MaxEquid} <= R_{MaxScal}$.

The right-hand part of the figure below shows how the rectangle (grey) programmed in the orbit channel is scaled down to $R_{MaxScal}$ with an equidistant distance. The scaling is then changed from $R_{MaxScal}$ to R=0. The requirements of orbit geometry are explained below.

2.2.1.1 Permitted orbit geometries in “Scaling” mode

Any orbit geometry is permitted for the “Scaling” mode. For example, linear, circular and polynomial (splines) elements may be used in any combination. The origin can also be placed outside the geometry.

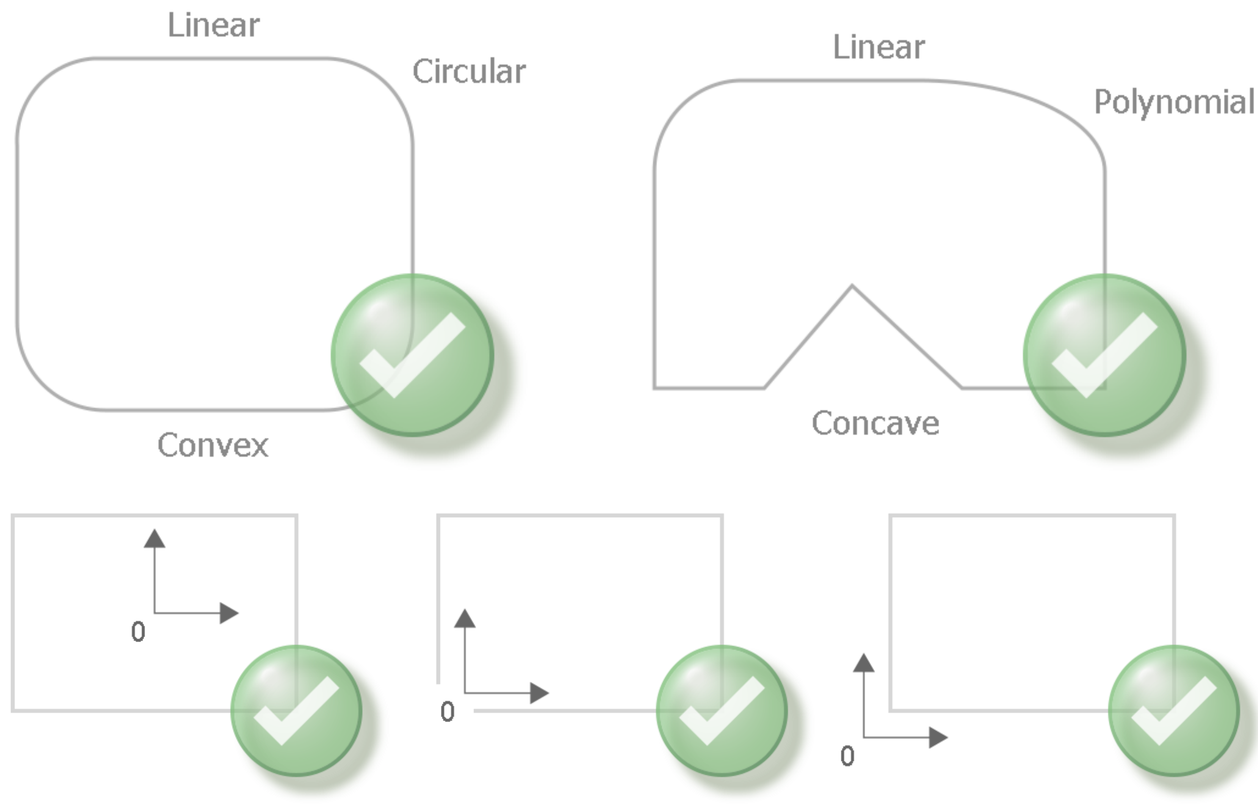


Fig. 17: Permitted geometry elements in “Scaling” mode

2.2.1.2 Permitted orbit geometries in "Equidistant" mode

In "Equidistant" mode, the geometry may only consist of linear and circular elements. No polynomial elements are permitted.

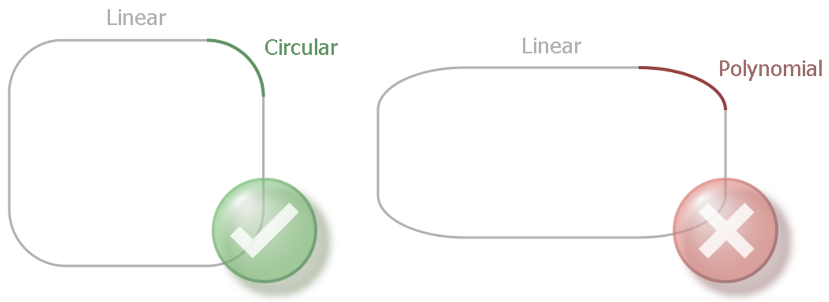


Fig. 18: Permitted geometry elements in Equidistant mode

Geometries must be convex.

If geometries are concave, error ID 50918 is output.

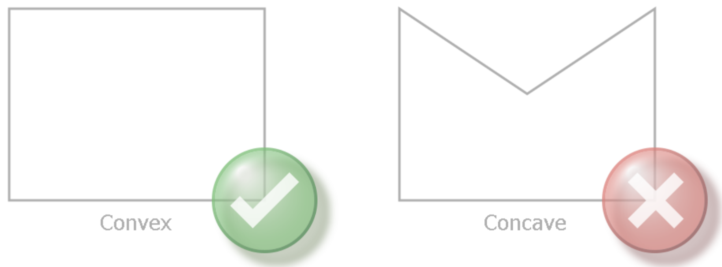


Fig. 19: Permitted geometry shape in Equidistant mode

The origin (0/0) must lie within the orbit scaling range. The positions of the origin are shown below:

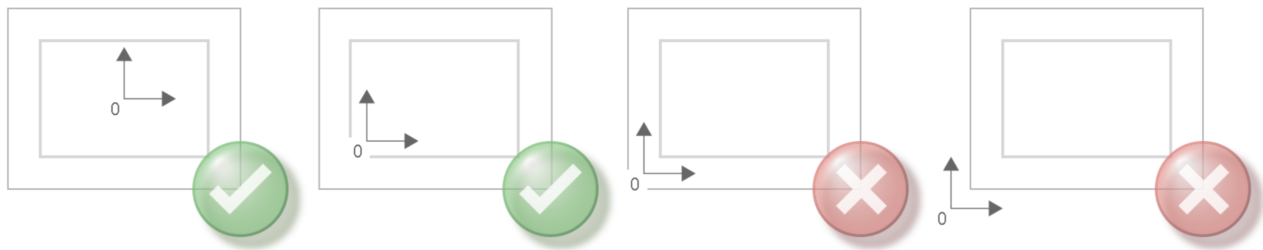


Fig. 20: Permitted positions of the origin (0/0) in Equidistant mode

The equidistant is always calculated on the inside (cf. left/right tool radius compensation, G41/G42) for the specified orbit geometry. This is based on the direction of rotation of the orbit channel which is automatically defined by the direction of the programmed geometry. The centre of rotation (clockwise) is at coordinate (0/0):

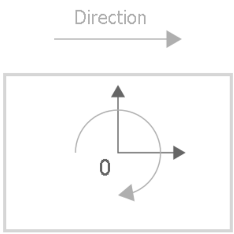


Fig. 21: Orbit direction of rotation (clockwise)

Transitions between two geometries

When an equidistant is calculated, there are certain limiting conditions that apply to the transitions between linear/circular blocks.

A non-tangential transition between two elements, as shown in the figure below, requires two linear elements of defined length at the break point. The length of the elements is:

$$l_{\mathrm{pre}} = l_{post} = \Delta R * \tan(\varphi) \qquad \Delta R = r_{max,Equid} - r_{max,Scale}$$

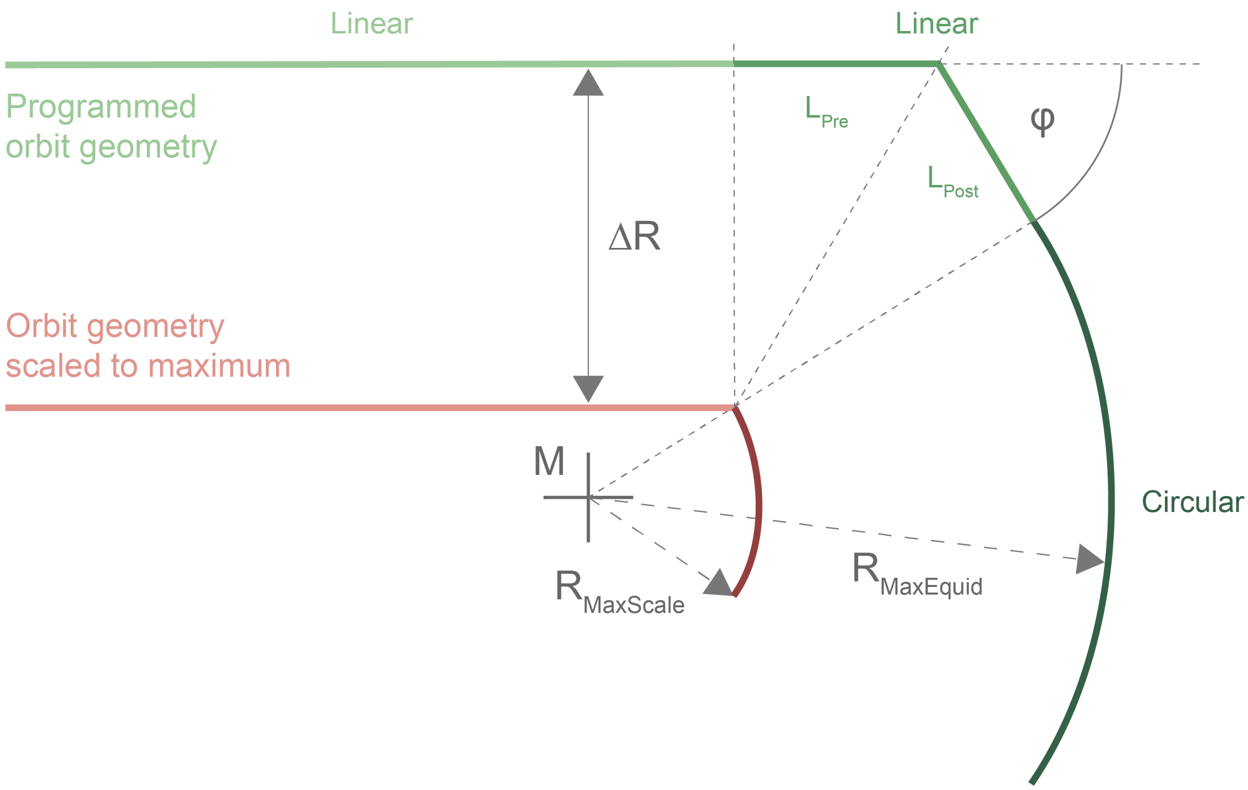


Fig. 22: Non-tangential transitions with two defined linear elements

Every transition to a circular element must be tangential.
If transitions to circles are non-tangential, error ID 50914 is output.

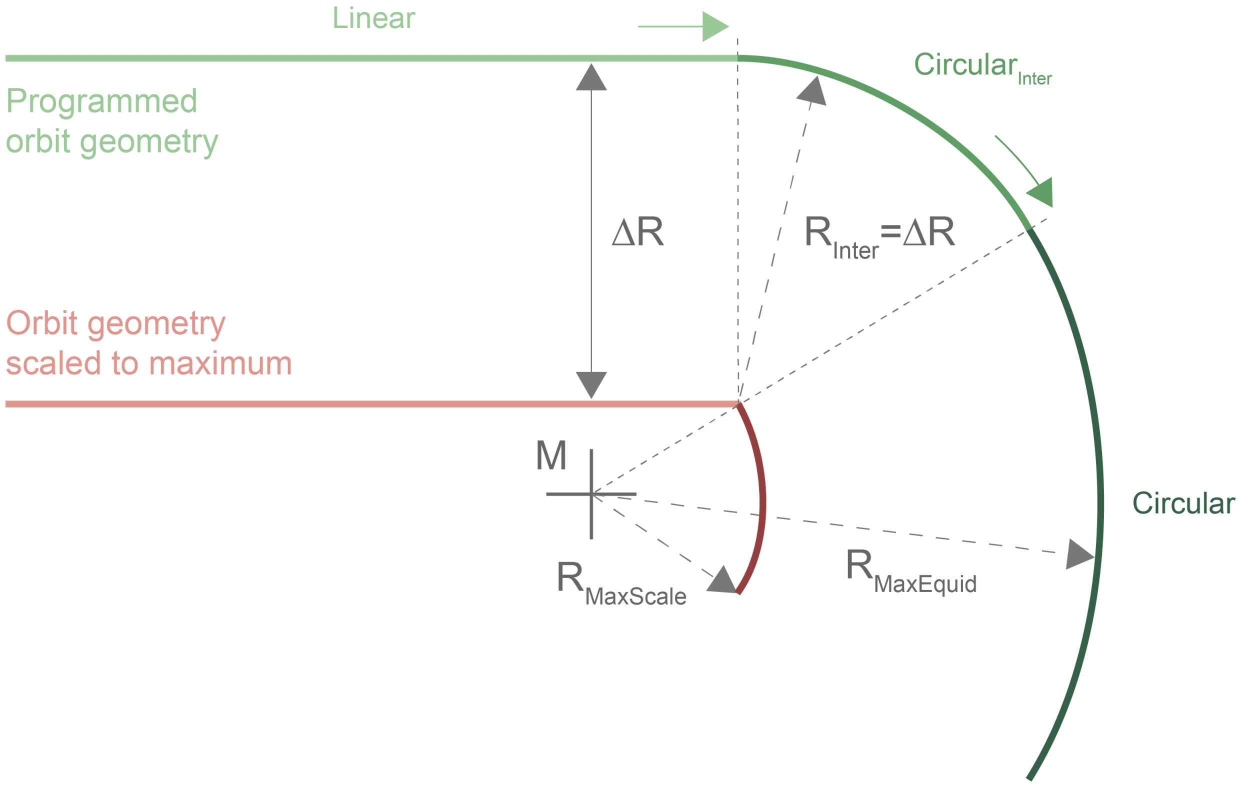


Fig. 23: Tangential circular transition

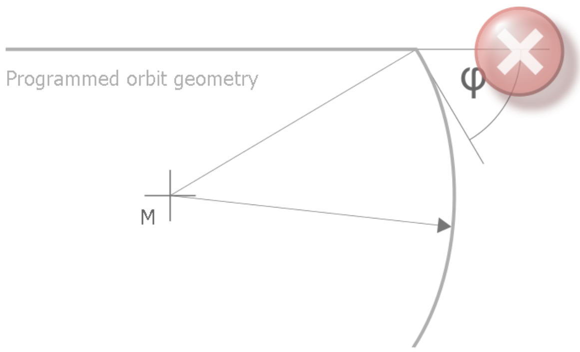
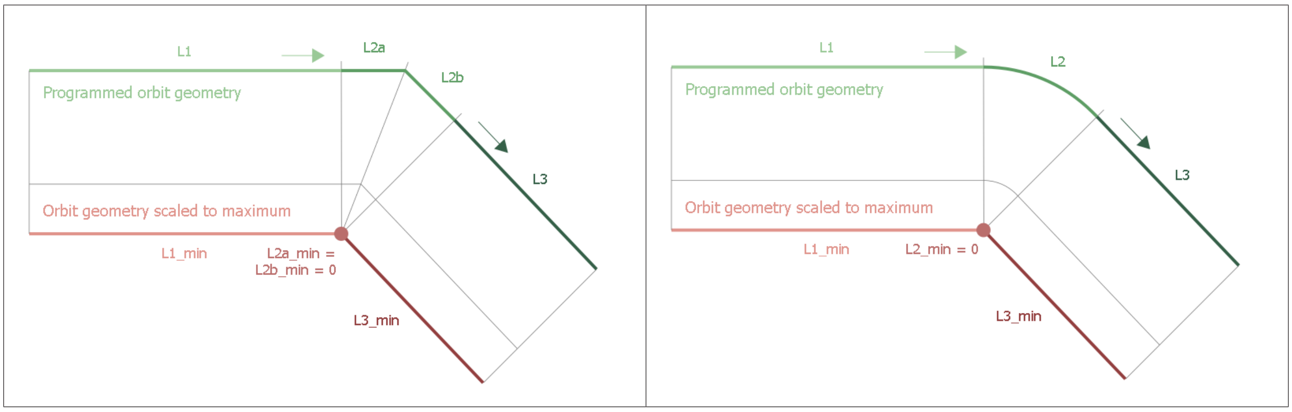


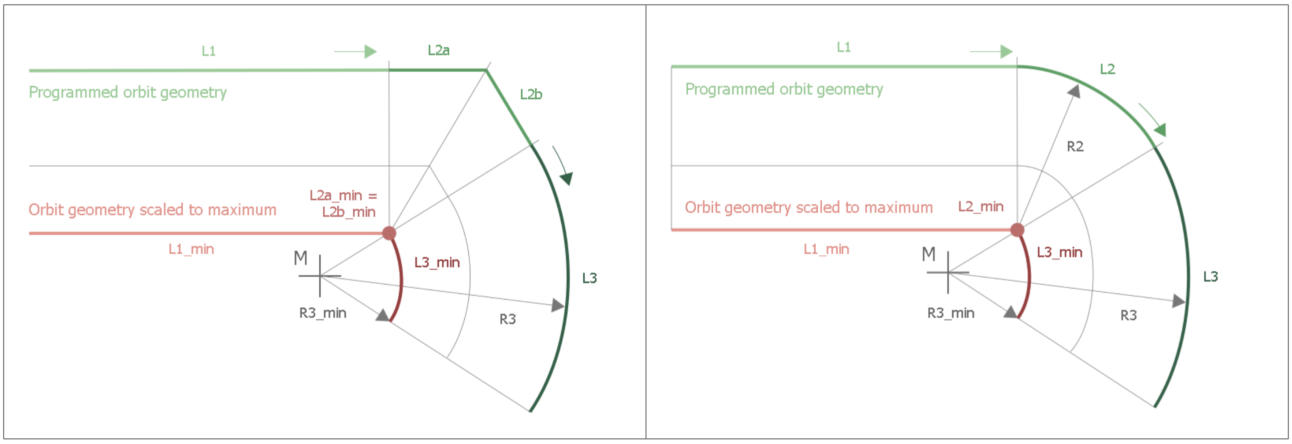
Fig. 24: Invalid non-tangential transition to a circle

Based on the above requirements for orbit geometry in equidistant mode, the following transitions between circular and linear elements are permitted:

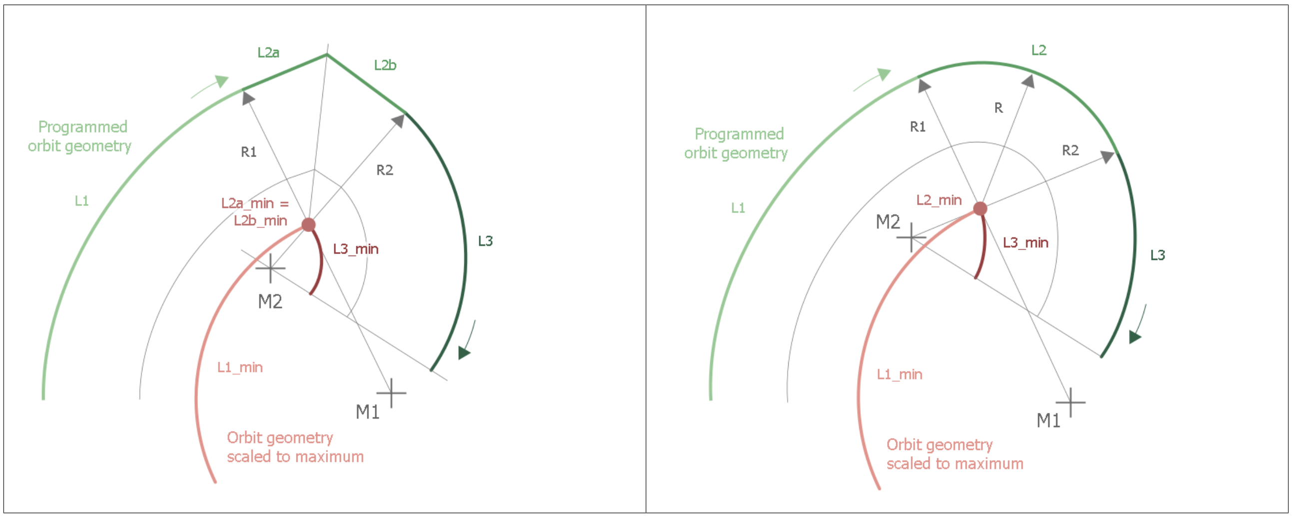
Linear-linear transition:



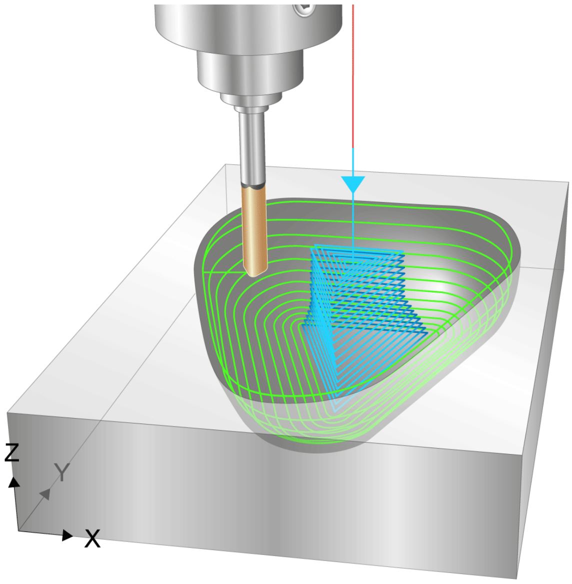
Linear-circular transition:

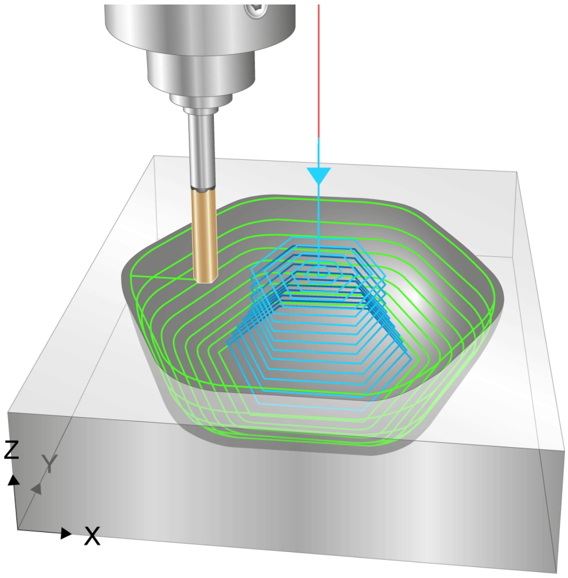


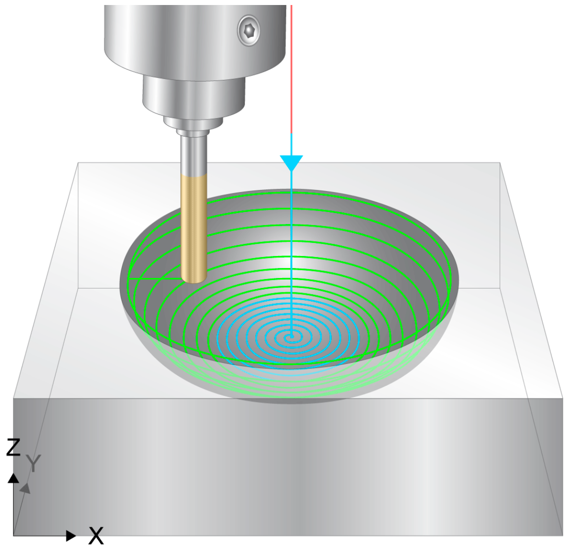
Circular-circular transition:

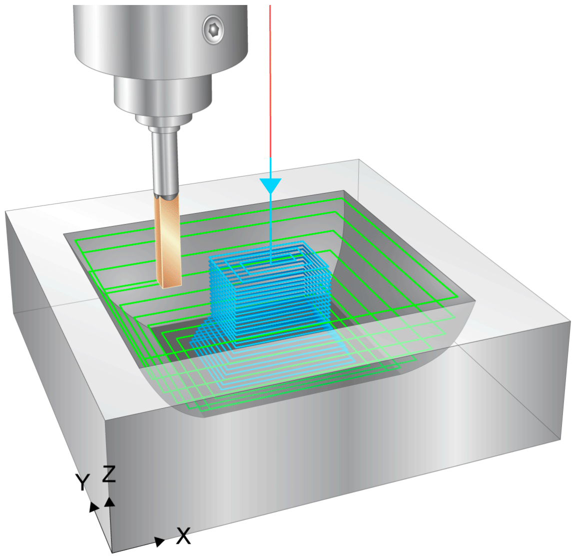


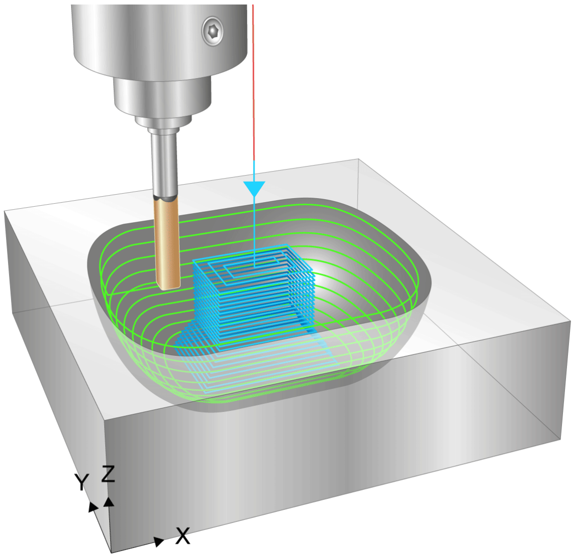
2.2.2 Graphic examples of orbit geometries











Workpieces produced using the same geometry in the down channel and different orbit geometries.

2.3 Escape channel

The escape motion of the electrode is executed in the escape channel and superimposed on the motion of the down channel.

The figure below shows the relationship between channel interfaces and the superimpositions ΔZ and ΔR viewed from the down channel:

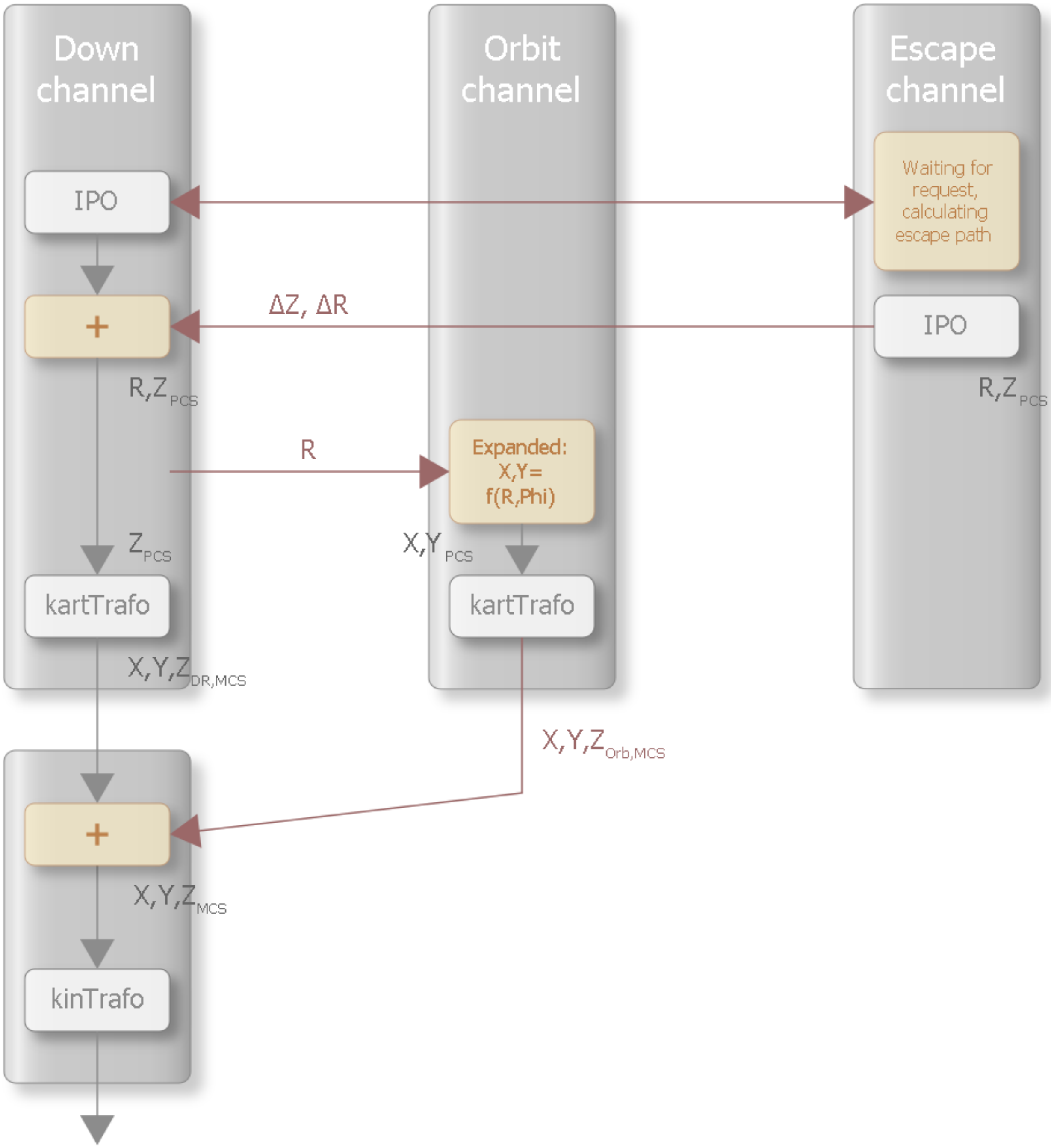


Fig. 25: Superimpositions ΔZ and ΔR viewed from the escape channel

The escape channel must be initialised by an NC program that contains the following:

- The path erosion geometry of the down channel
- Initialisation of the interface to the down channel

At the end of the program, an escape can be commanded by specifying a negative external velocity on the corresponding HLI interface (Specified external path velocity control unit). The escape motion is calculated for each command based on the current position in the down channel.

Various escape strategies are implemented, each of which results in a different escape path:

1. Flat - plane strategy [▶ 36]
2. Alpha - angle strategy [▶ 37]
3. Point - point strategy [▶ 38]
4. Bisector - angle bisector strategy [▶ 39]

All escape strategies refer back to the escape point. By default, the escape point is the end point of the path erosion strategy. If a path erosion was programmed ahead of it, the escape continues with this geometry.

By superimposing the positions of the down and escape channels, the escape channel can be used to adjust the gap between the electrode and the workpiece to meet the process requirements, provided the appropriate escape strategy is selected. The external velocity required for this in the escape channel is specified by the spark generator.

2.3.1 Explanations of escape strategies

2.3.1.1 Flat strategy

In the flat strategy, the motion starts from the current position in the down channel, first at constant Z height up to radius 0, followed by a perpendicular motion up to the escape point.

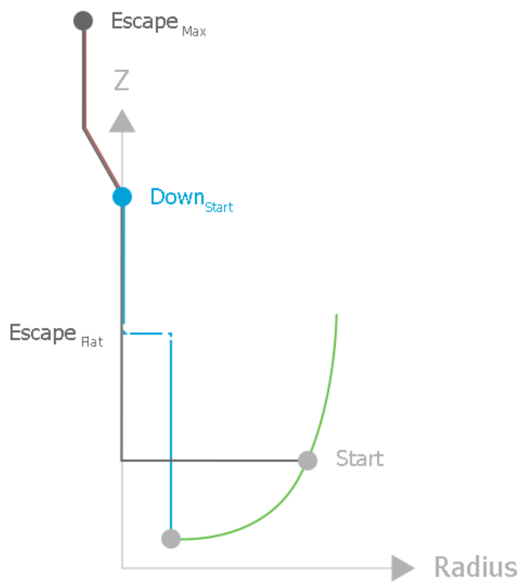


Fig. 26: Flat strategy

2.3.1.2 Angle strategy (alpha strategy)

The alpha strategy features the fact that it specifies the angle from which a linear segment is calculated when viewed in the reverse direction, starting from the current position of the down channel.

- If the straight line intersects the Z axis below the escape point (radius 0), a perpendicular motion takes place from there to the escape point (bottom left figure).
- If radius zero is not reached below the escape point, a horizontal motion takes place from the point of intersection with the horizontal to the escape point (bottom right figure)

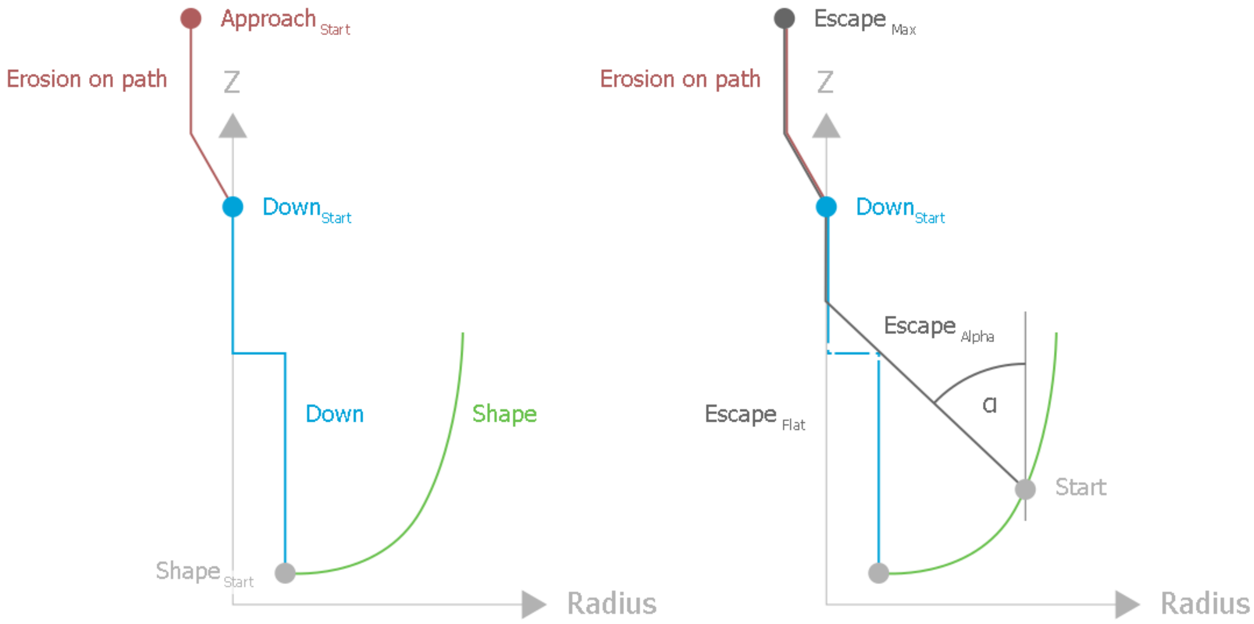


Fig. 27: Angle strategy (alpha strategy)

If an angle of 90 degrees is specified, this corresponds to the behaviour of the Flat strategy [▶ 36].

2.3.1.3 Point strategy

With point strategy, a point must be defined with a Z coordinate that is less than or equal to the $down_{Start}$ (escape point) height and with a radial coordinate that is greater than or equal to 0. There are three possible paths depending on the current position in the down channel:

1. $Z_{point} \leq Z_{start}$ $Flat_{escape}$
2. $Z_{point} > Z_{start}$, $radius_{point} \leq abs(radius_{start})$ linear motion up to the point followed by $flat_{escape}$
3. $Z_{point} > Z_{start}$, $radius_{point} > abs(radius_{start})$ perpendicular motion up to the horizontal of the point followed by $flat_{escape}$.

These three variants are shown in the figure below:

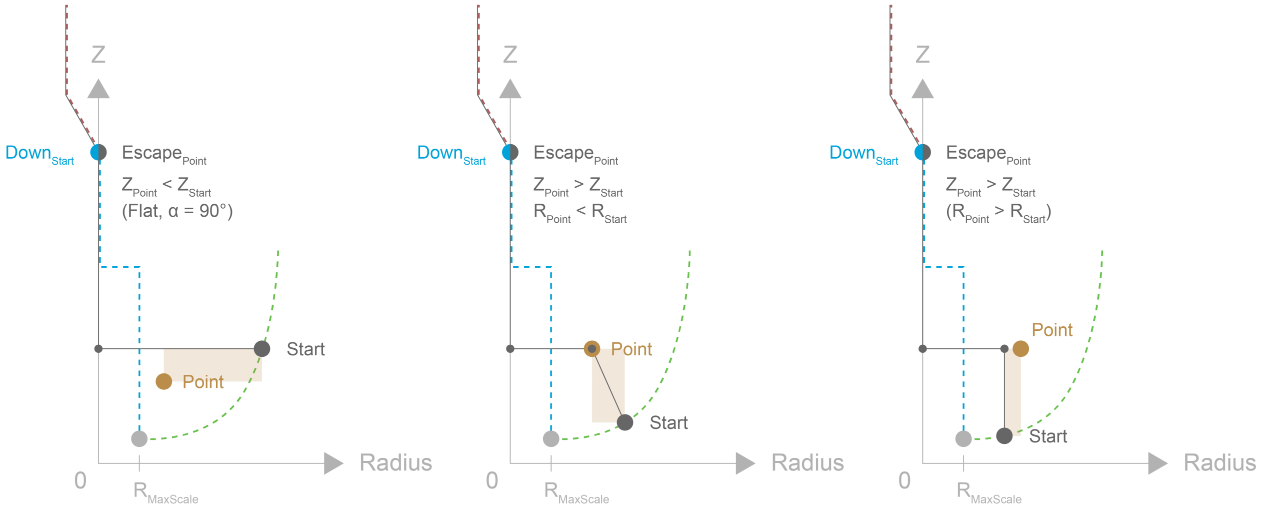


Fig. 28: Three variants of point strategy

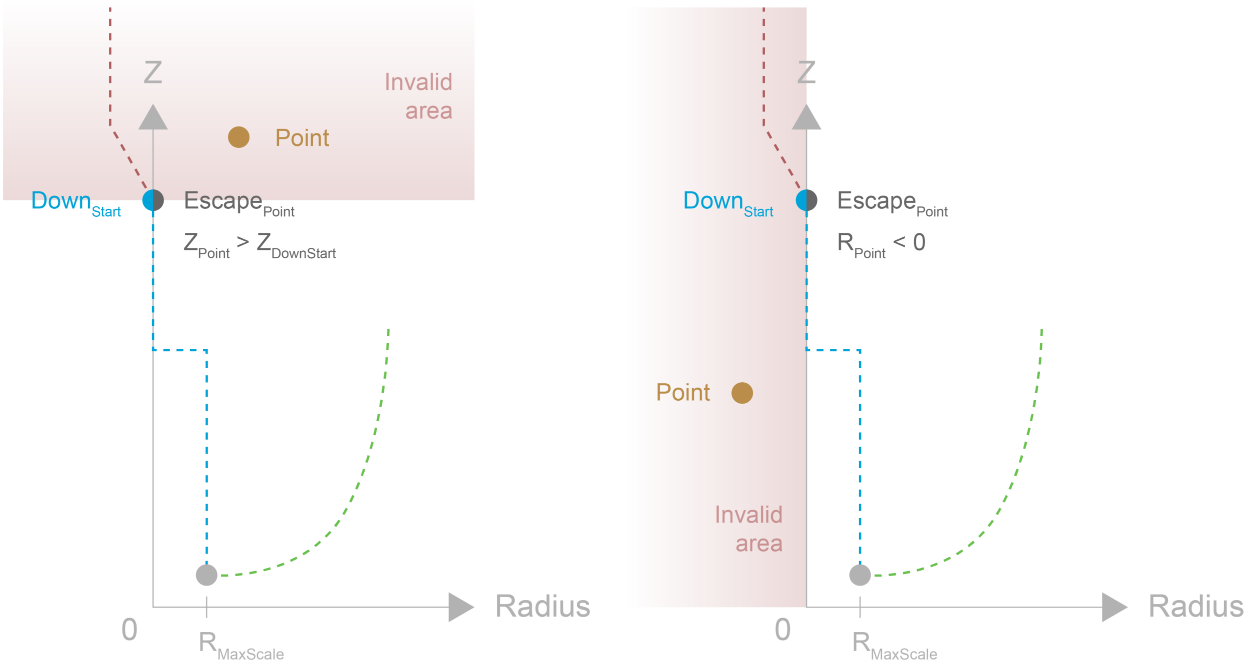


Fig. 29: Invalid positions of points

2.3.1.4 Angle bisector strategy

The bisector escape strategy stores the Z height of the activation ($Z_{Bisector}$). In addition, the user requires the two parameters D1≥ 0 and D2≥ 0. The geometric shape of the escape path executed is dependent on the current height in the down channel (Z_{start}):

1. $Z_{start} <= Z_{Bisector} + D1$
   flat escape
2. $Z_{Bisector} + D1 < Z_{start} <= Z_{Bisector} + D1 + D2$
   motion in direction of angle bisector up to $Z = Z_{Bisector} + D1$,
   followed by a flat escape
3. $Z_{Bisector} + D1 + D2 < Z_{start}$
   motion in direction of angle bisector up to $Z = Z_{Bisector} – D2$,
   followed by a flat escape

The ”direction of the angle bisector” is defined as the direction of the angle bisector from the horizontal and the current tangent of the geometry in the down channel. All three variants are shown in the figures below. The horizontal and the current tangent of the geometry in the down channel are shown in brown if they are used in the respective figure to calculate the angle bisector.

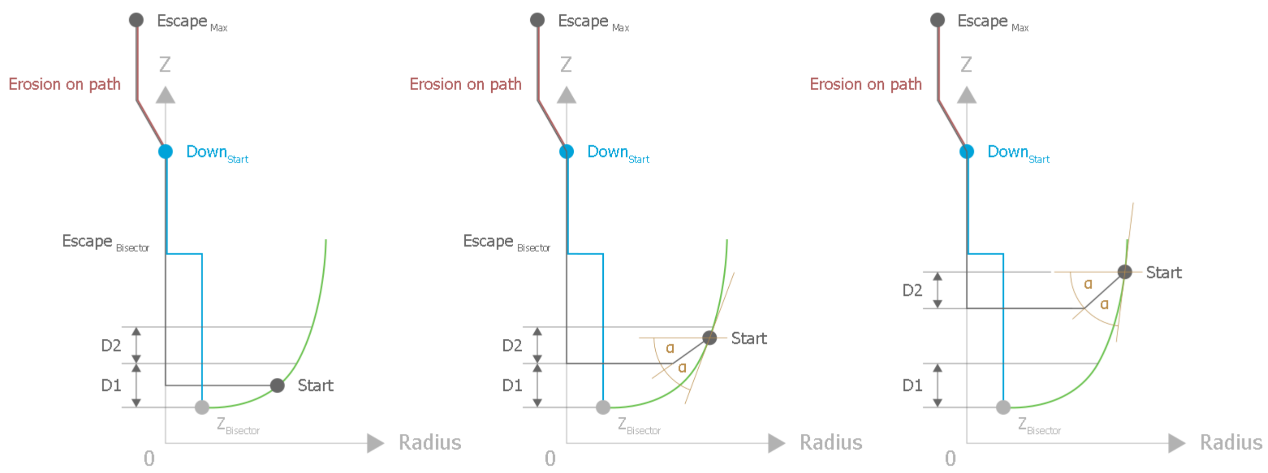


Fig. 30: Possible paths of the angle bisector strategy

As a special case, this strategy results in a horizontal and possibly also a perpendicular motion in the down channel. The escape motion is either identical to the flat motion (‘Start 2’ in the figure below) or perpendicularly downwards. (‘Start 1’) - depending on which direction is closest to the escape motion direction of the previous contour element of the geometry in the down channel.

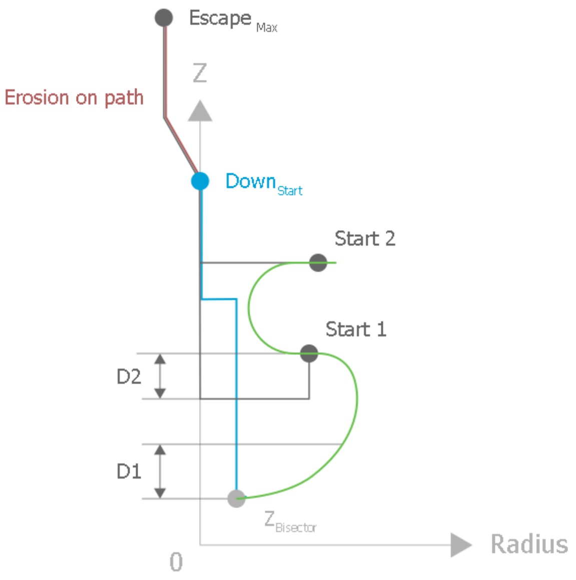


Fig. 31: Special case for angle bisector strategy

In the angle bisector escape strategy, a check is made during program runtime to ensure that the angles 0° and 180° between the tangent and the horizontal are not exceeded. This check takes place above height $Z_{Bisector}$ + D1. If the angle is exceeded, this would result in a jump to geometry planning and therefore also a position jump with an active escape motion. Error ID 51031 is then output. The geometries shown in the figure below are therefore not possible due to the section marked by a solid red line.

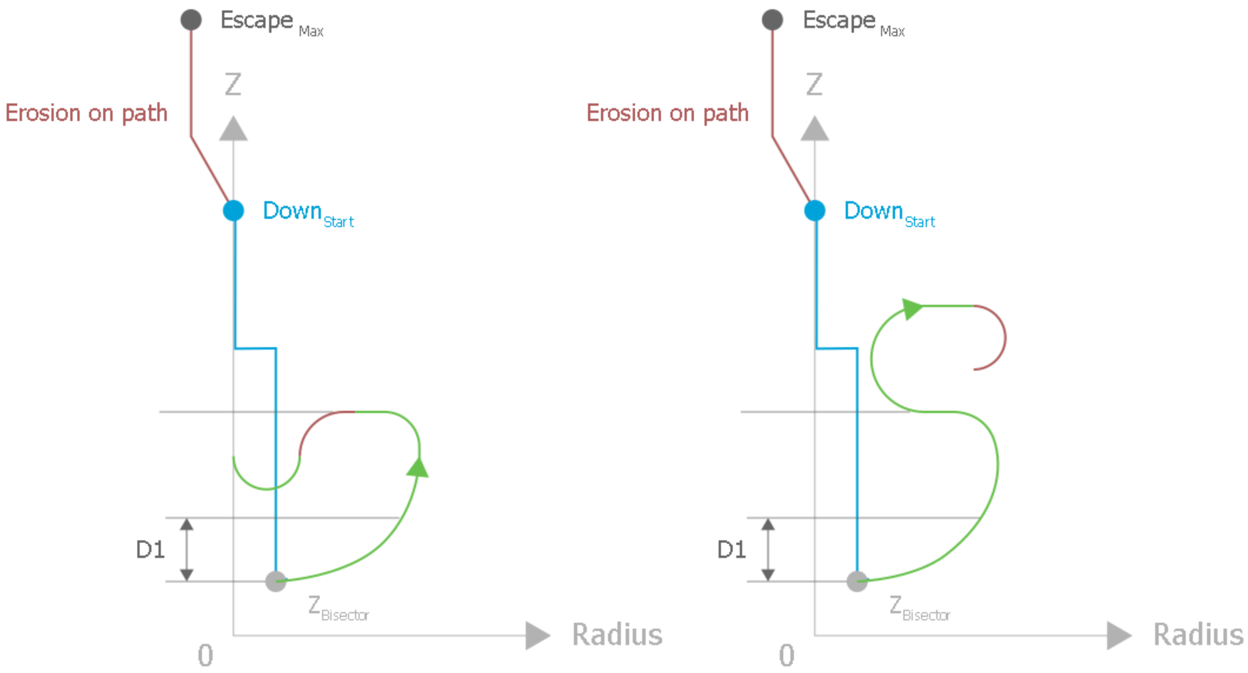


Fig. 32: Boundary geometries with angle bisector strategy

When the angle bisector escape strategy is used, it is important to ensure that the contour in the down channel C1 is continuous as of height $Z_{Bisector}$ + D1. Otherwise, position jumps are possible with an active distance in the escape channel and a motion existing in the down channel.

In addition, it is important to pay attention to the feed rate in the down channel when the distance in the escape channel is active, especially if the geometry in the down channel has small radii. In this case, minor position changes in the down channel are magnified by the escape geometry as if by a lever, so that the superimposed position change resulting from the down and escape channels in one cycle may be greater than the permitted velocity value and thus result in an error.

It should also be noted that this escape strategy may possibly intersect the geometry of the down channel during its escape motion and then remove material that the user does not intend. The user must therefore check the geometry in the down channel as well as the selected parameters D1 and D2 for suitability.

2.3.2 Changing the escape point

By default, the escape point corresponds to the end point of the path erosion geometry. The escape path is calculated depending on the escape strategy up to the escape point. The escape path is then moved back along the path geometry, if it exists.

This escape point can be changed by changing the coordinate system in the down channel.

In this case, note the following:

- The interfaces to the auxiliary channels must be deactivated when the coordinate system is activated.
- The new coordinate system must fulfil X=0 and Y=0 at the current position.

When the escape channel is reactivated and on receipt of an escape request, the current position of the down channel is transferred to the escape channel together with the current coordinate system of the down channel. If the origin of the new coordinate system does not match the end point of the path erosion geometry, the origin becomes the new escape point.

When commands are made, the escape path is calculated using the escape strategy to the new escape point. A linear block is then generated from the escape point to the end point of the path erosion geometry. The linear block has the same properties as if it belonged to the path erosion geometry. Then, if available, the path geometry is further retracted.

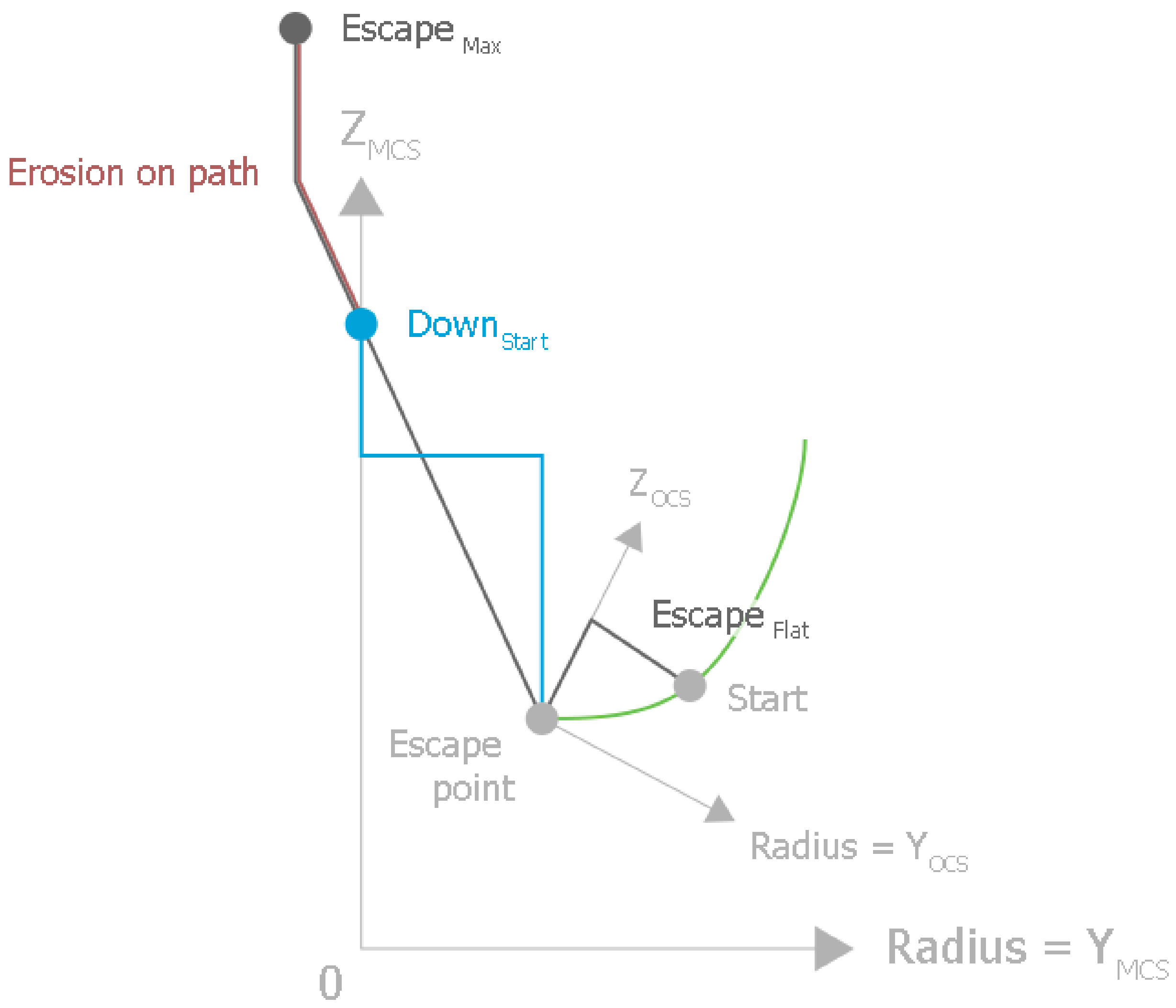


Fig. 33: Flat escape with changed escape point

2.4 Flushing functionality

When the CNC is configured for die sinking, the escape channel enables an escape motion of the electrode on a defined geometry (see section Escape channel [▶ 34]).

The flushing function triggers a special escape motion. The rapid escape of the electrode can be commanded by the Jump control unit [▶ 110] on the HLI if a flushing process is required. The geometry on which the electrode moves during a flushing process is identical to the geometry of an escape that is triggered over the external velocity interface.

Special features of the flushing function include the increased dynamics of the electrode motion and the fact that the distance or height of the escape motion can be commanded. When the specified distance or height is reached in the escape geometry, the electrode is returned to the starting point of the flushing process.

The floating point allows the flushing process to be terminated before the starting point is reached. When the electrode reaches this point, the system is brought to a standstill. It then switches to a normal escape state and the escape motion is controlled over the external velocity interface. It is also possible to abort the flushing process while it is in progress. The abort can be executed by two different dynamics (normal abort, emergency abort).

Special features of a flushing process

- Increased dynamics
- Specified distances and heights for the escape motion

The figure below shows an example of a command sequence for a flushing process:

- The bottom section shows a schematic diagram of the signals for backward motion active, flush active and the flush start command signal on the HLI of the escape channel over time.
- The upper section shows an example of a geometry matching the signals. Only the Z axis moves in this case. The example shows a command during an active escape motion and with premature abort of the flushing process by the floating point.

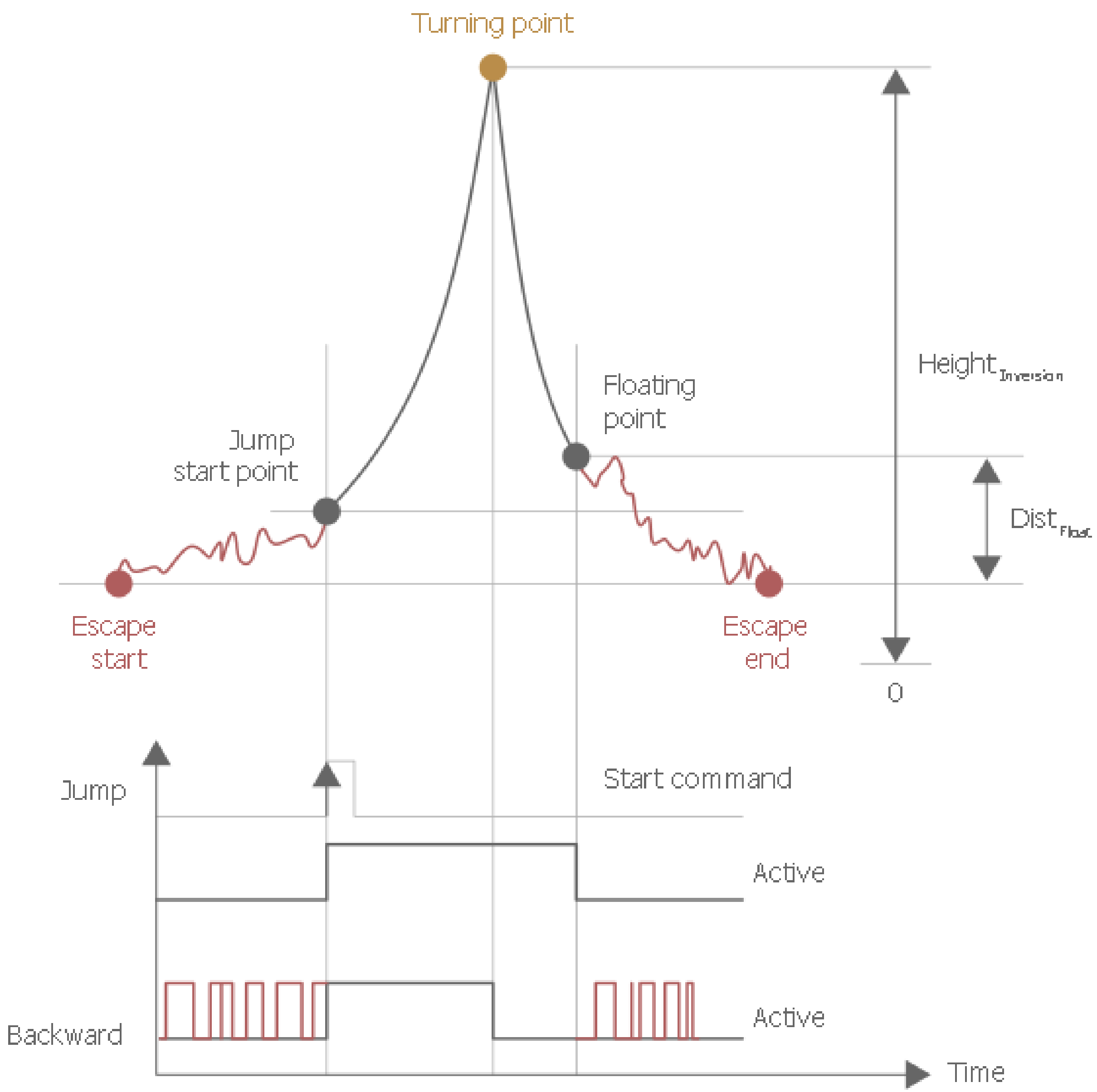


Fig. 34: Position and signal profiles during a flushing process

NOTICE
During a flushing process, the down and orbit channels must be at standstill.
Otherwise, error ID 50927 is output if there is any motion in one of the channels.

2.4.1 Point definitions

The following points can/must be specified for the escape motion:

- Flushing inversion point
- Flushing end point, floating point

2.4.1.1 Inversion point during the orbiting process

The inversion point is defined by the Z height in the programming coordinate system of the down channel or the distance from the starting point.

If the specified Z height is less than the Z height of the starting point, a flushing process is not executed.

The inversion point is defined by the data items inversion_point.distance and inversion_point.height of the (Jump control unit) command:

inversion_point.distance > 0	Define inversion point with inversion_point.distance
inversion_point.distance == 0	Define inversion point with inversion_point.height
inversion_point.distance == HLI_JUMP_ABORT (=-1)	Abort previous jump with specified dynamics
inversion_point.distance <= HLI_JUMP_ABORT_EMERGENCY (=-2)	Abort previous jump with set emergency dynamics

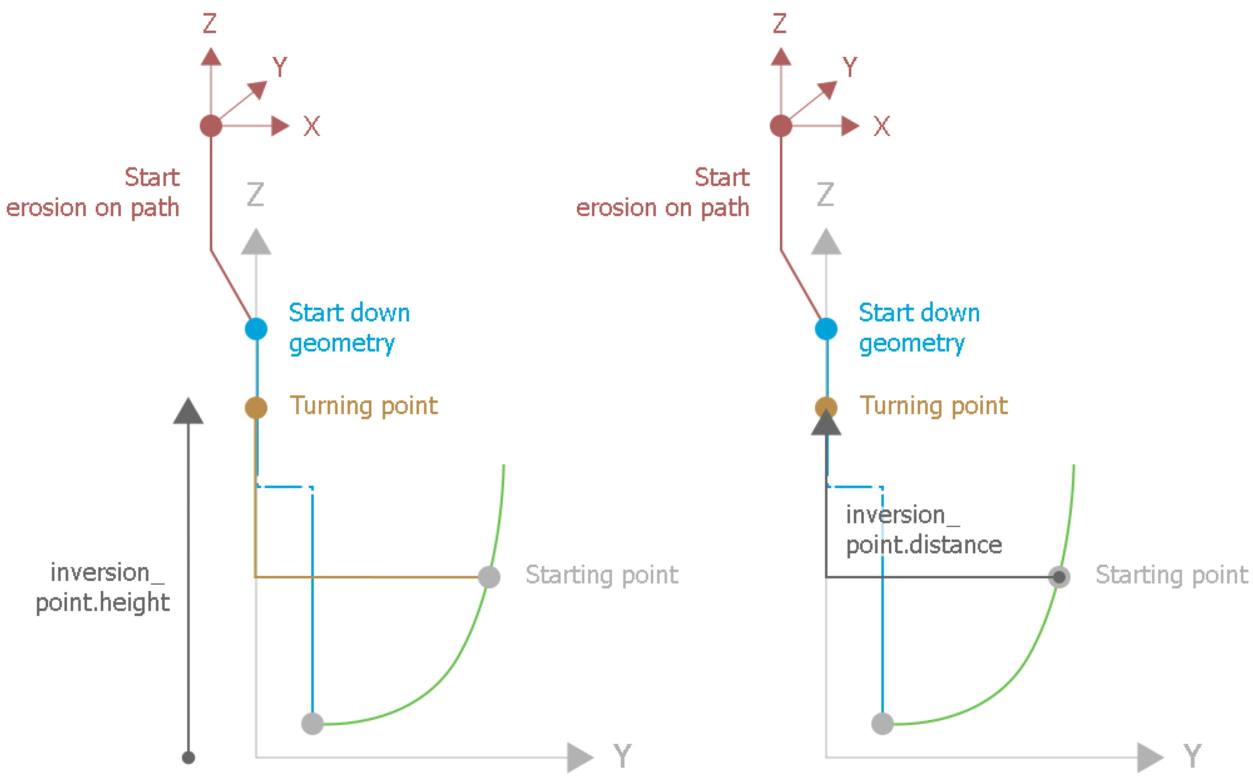


Fig. 35: Positioning the inversion point during the orbiting process

2.4.1.2 Inversion point in erosion on the path function?

If the commanded Z height is greater than the Z height of the starting point of orbiting or the commanded length is longer than the distance to the starting point of orbiting, the height or distance to go is traversed as the distance of the programmed path.

In this case, the electrode is first moved to the starting point of orbiting (see figure: even if the Z height of the starting point is smaller than the current Z height of the down channel) and then the distance to go on the path is traversed.

If no machining was programmed on the path, the starting point of orbiting is also the end point of the flushing process.

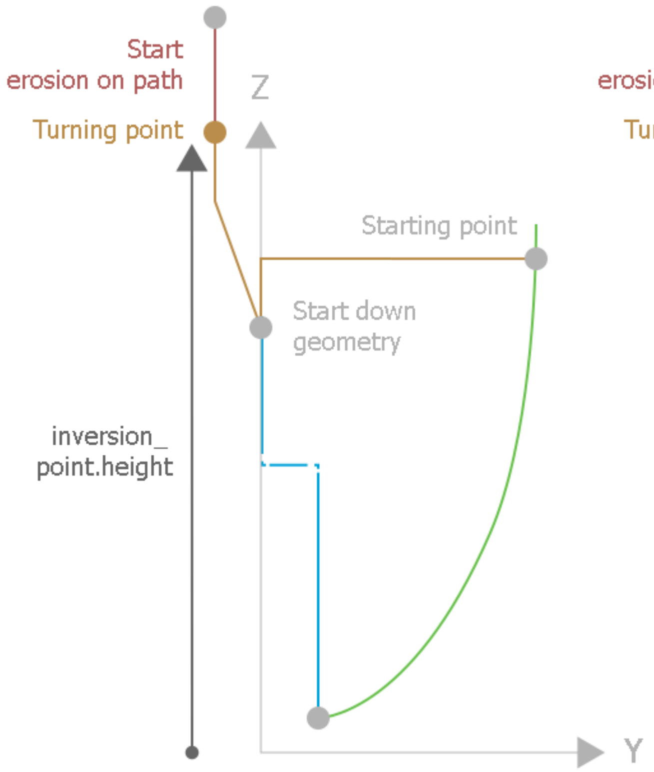


Fig. 36: Inversion point for erosion on the path

2.4.1.3 Flushing process end point, floating point

The floating point defines the end point of the flushing process. It is commanded as the distance from the original starting point of the flushing process. After the inversion point is reached, the electrode continues to move forwards up to the floating point. The distance to go to the original starting point must be traversed at the external velocity commanded by the PLC.

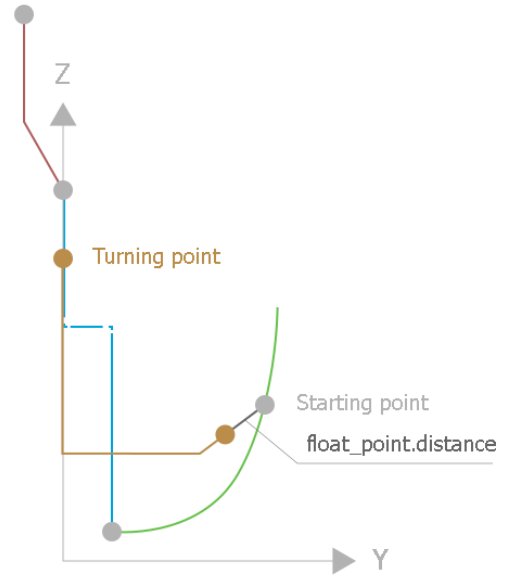


Fig. 37: Position of the end point of flushing

2.4.2 Calculating the dynamics

The down and orbit channels may not move during a flushing process, otherwise error ID 50927 is output. For this reason, the escape channel can take up 100% of the axis dynamics during a jump instead of the usual 50% per channel for die sinking.

The dynamics calculation can be separated into three sectors depending on the erosion geometry:

- Path motion and orbiting at radius = 0
- Orbiting with radius > 0 scaled sector
- Orbiting with radius > 0 in equidistant sector

The flushing motion is always carried out implicitly with a non-linear velocity profile.

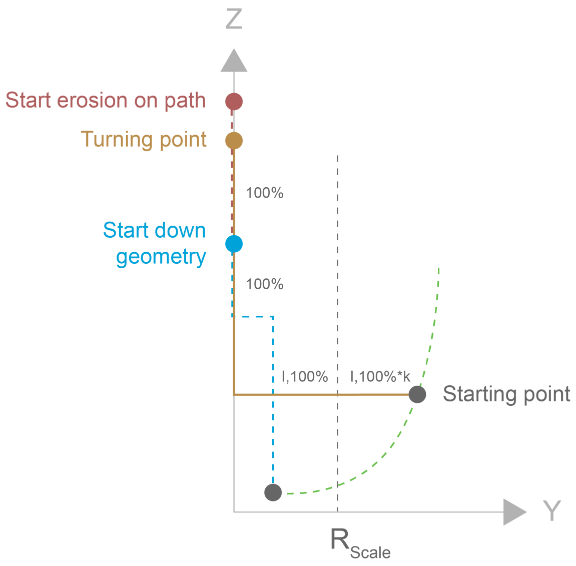


Fig. 38: Classifying the dynamics of the flushing motion

Path motion and orbiting at radius = 0

In the case of path motions and orbiting at radius = 0, the escape channel has all the data required about which axes are moved - the orbit channel has no influence on moving axes. On the path, the escape geometry is a 3D geometry and at radius = 0 there is only motion along the Z axis in the Z/R coordinate system of the down and escape channels. For this reason, these motions can be traversed with maximum dynamics.

Orbiting at radius > 0 in the scaled sector

At radius > 0, the orbit channel moves the real electrode position in a plane perpendicular to the Z axis of the coordinate system in the escape channel. However, the escape channel only moves the Y and Z axes in its coordinate system, but has no data about the axes that are actually moved. For this reason, the dynamics of this motion is limited to 100% of the dynamics of the weakest axis.

Orbiting at radius > 0 in the equidistant sector

In linear corner elements of the equidistant sector, the actual distance traversed can be greater than the change in radius of the escape motion.
If the orbit channel is exactly in the corner, the factor k between the actual distance and the change in radius is sent by the orbit channel to the escape channel. The dynamics of the escape motion is then calculated at 100% dynamics of the weakest axis multiplied by the factor k. This results in a motion that is traversed at 100% dynamic of the weakest axis when the orbit channel is positioned exactly at the corner of the corner element.

If the orbit channel is not positioned exactly on the corner, the motion is less dynamic. The motion always stops at the transition to the scaled sector.

2.5 Premature change of generator settings

If the surface finish of the current machining stage is found to be of sufficient quality, you can change directly to the next machining stage to shorten machining time. Machining should then be aborted at the current point, the electrode removed from the cavity and machining continued at another point in the program, for example with a different orbit geometry and escape strategy.

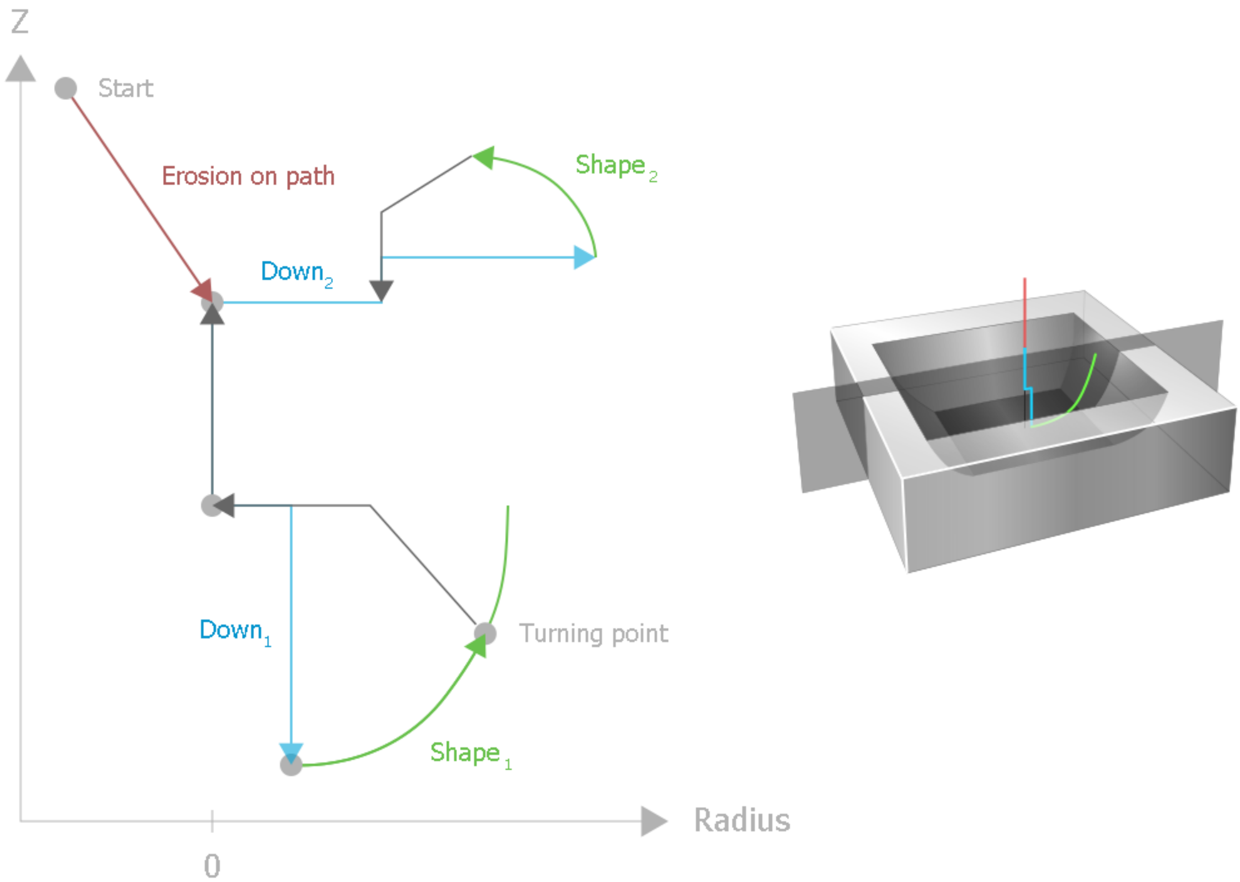


Fig. 39: Changing machining sequences

The "Delete distance to go" function is used to change generator settings prematurely. When jump labels are used, it removes all geometry elements and other NC commands until the selected jump label is reached. The NC program must meet various requirements to enable changes to be made to generator settings using this function.

- The radius of the down channel must be 0 if the orbit channel is to be deactivated and reactivated.
- Restrictions in the "Delete distance to go" function must be considered, for example not all commands can be skipped. Here is a selection of die-sinking commands that cannot be skipped:
  - #CS ON
  - #TRAFO ON
  - #CHANNEL INIT
  - #TRACK CHAN OFF [▶ 97]; the "Delete distance to go" command is aborted with #CHANNEL INIT and #TRACK CHAN OFF [▶ 97] and the following blocks are executed starting from the current position.
- There is a further restriction of the "Delete distance to go" function in connection with circular elements. Circular elements (G02 / G03) are often programmed relative to the end point of the previous block (G162). If the next motion block after the jump label is a circular element, the current position must be the same as if the program had been executed without 'Delete distance to go'. If circular elements are programmed using absolute end points (G161), the current position must be on the circular arc.

One way of avoiding this problem is to program a linear motion block at the starting point of the circular element before the circular element (that follows directly after the selected jump label). This would have distance 0 if executed without ‘Delete distance to go’.

The program sequence for changing generator settings is normally as follows:

1. Execute a flushing process at a possible change position (e.g. end point of erosion on the path or radius = 0). This is where generator settings are changed.
2. Deactivate the escape channel, if required.
3. Deactivate the orbit channel, if required.
4. Reorient the down channel, if required.
5. Reactivate the orbit channel in its new orientation.
6. Reactivate the escape channel.
7. Continue the second geometry.

One way to set the generator change point to the end point of erosion on the path is to program an M01 function at the end of the path geometry. Before a flushing process of sufficient length is commanded, this optional stop in the escape channel must be activated using the corresponding control unit.

A second option for reaching a selected generator change point is to command a flushing process where the required distance and floating distance are identical. A flushing process can then be commanded to end immediately when the selected distance is reached.

By inserting conditions at the start of the second geometry that compare the current position with a specified value, the second geometry can contain different motion sequences depending on whether a generator change has taken place. A programming example for this application is shown in Premature change of generator settings [▶ 87].

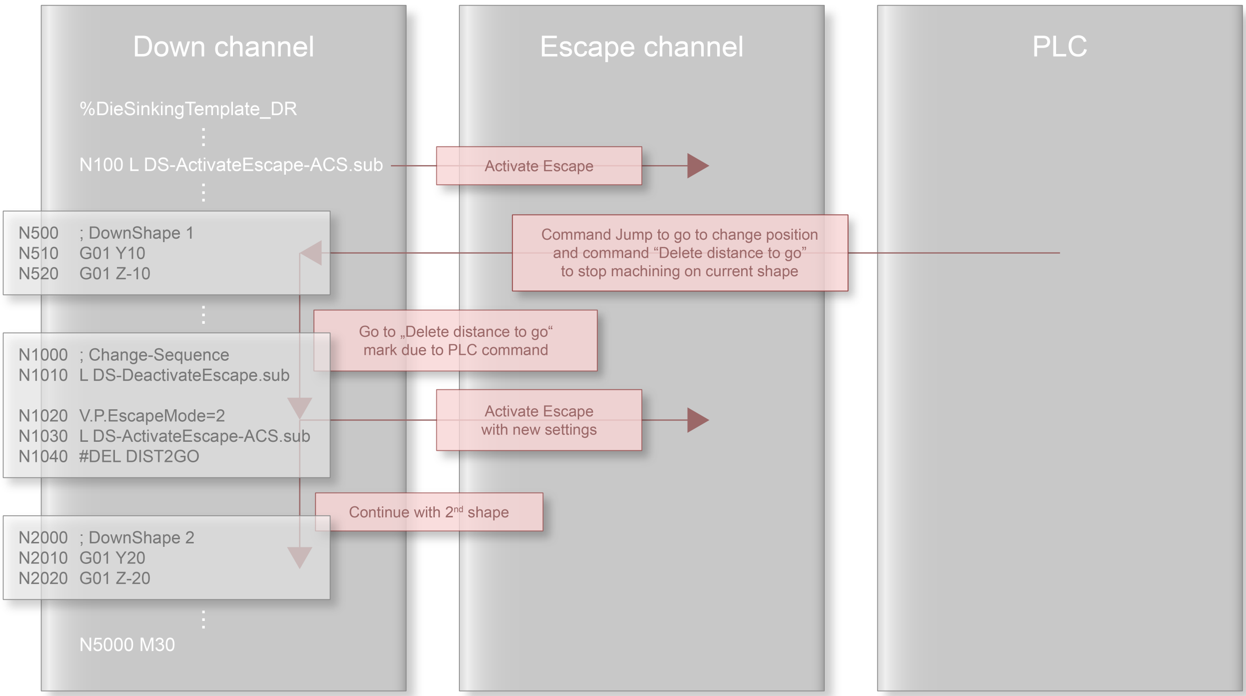


Fig. 40: Generator change sequence

2.6 Display data

2.6.1 Position data

In general, machine drives are controlled via the down channel with an ACS position. All superimpositions can be viewed in the coordinates of each channel.

The figure below provides an overview of the position display data available on the HLI for die sinking:

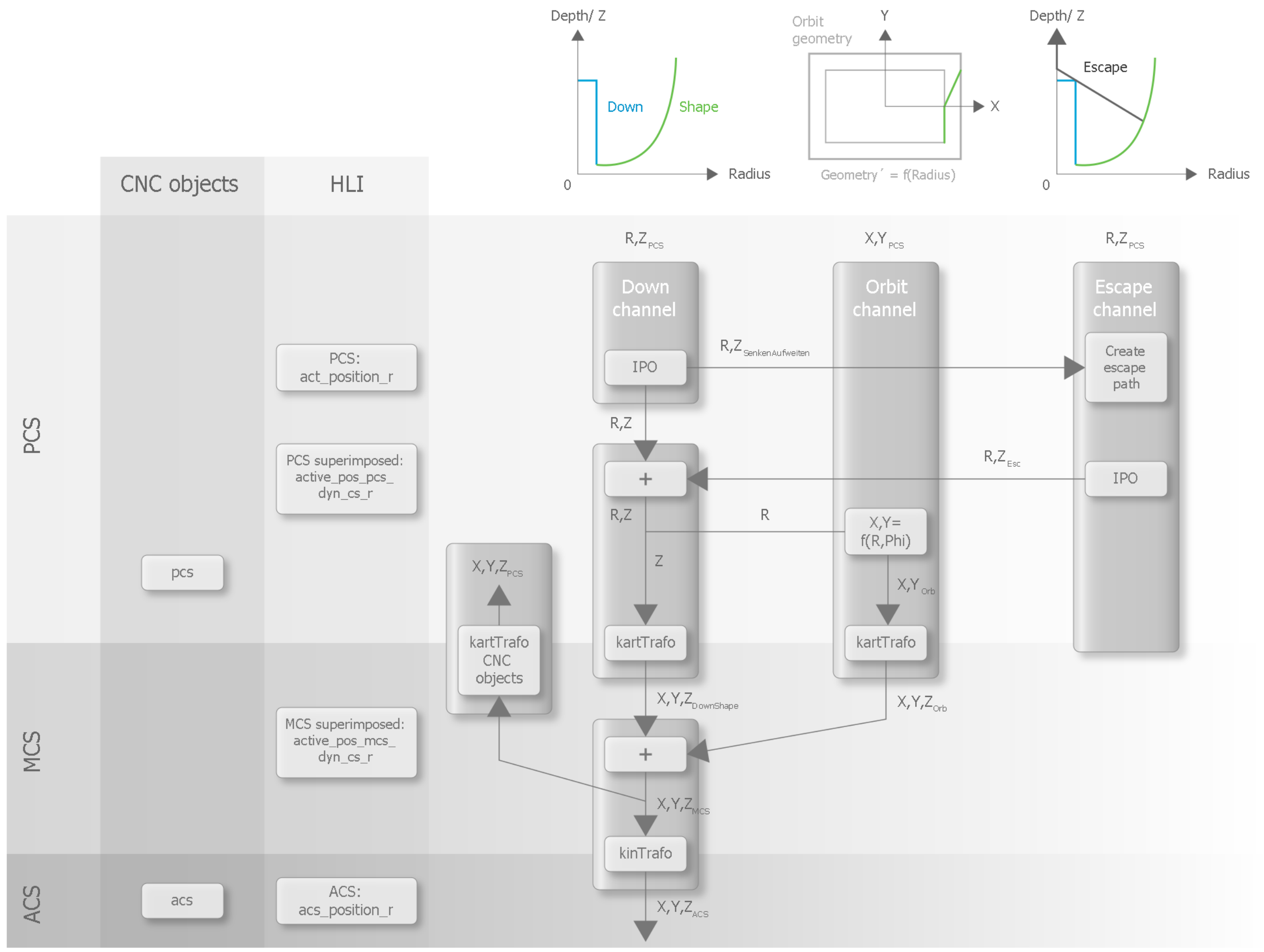


Fig. 41: Position display data

2.6.1.1 Position display via HLI parameters

The command positions of an axis in the currently selected coordinate system (PCS positions) are available on the HLI in act_position_r [▸ 105].

This PCS position is superimposed by the escape channel and its offset. The superimposed position is available on the HLI of the down channel in the data item active_pos_pcs_dyn_cs_r [▸ 105].

The position (active_pos_pcs_dyn_cs_r [▸ 105]) is transferred to the orbit channel. A position is then calculated in the orbit channel using the existing contour which in turn is superimposed on the down channel. The new superimposed position is located on the HLI of the down channel in active_pos_mcs_dyn_cs_r [▸ 105].

The position data at axis level, including all superimpositions based on dynamic coordinate systems, are available on the HLI of the down channel in the data item acs_position_r [▸ 105].

2.6.1.2 Position display via CNC objects

The physical TCP position of the electrode can be read in all defined Cartesian coordinate systems by CNC objects in the 1st channel.

The parameter P-CHAN-00145 [▸ 119] defines how to include step 1 and 2 kinematic transformations in the position display data.

The default value of P-CHAN-00145 [▸ 119] is 0. This means that kinematic transformation steps are only included if they are activated.

If this parameter is initialised with 2, the calculation is already carried out when a transformation step is defined in the channel parameters.

The right-hand side of the figure below shows the calculation of position display data based on axis positions in the ACS for the various parameterisation options in ascending order.

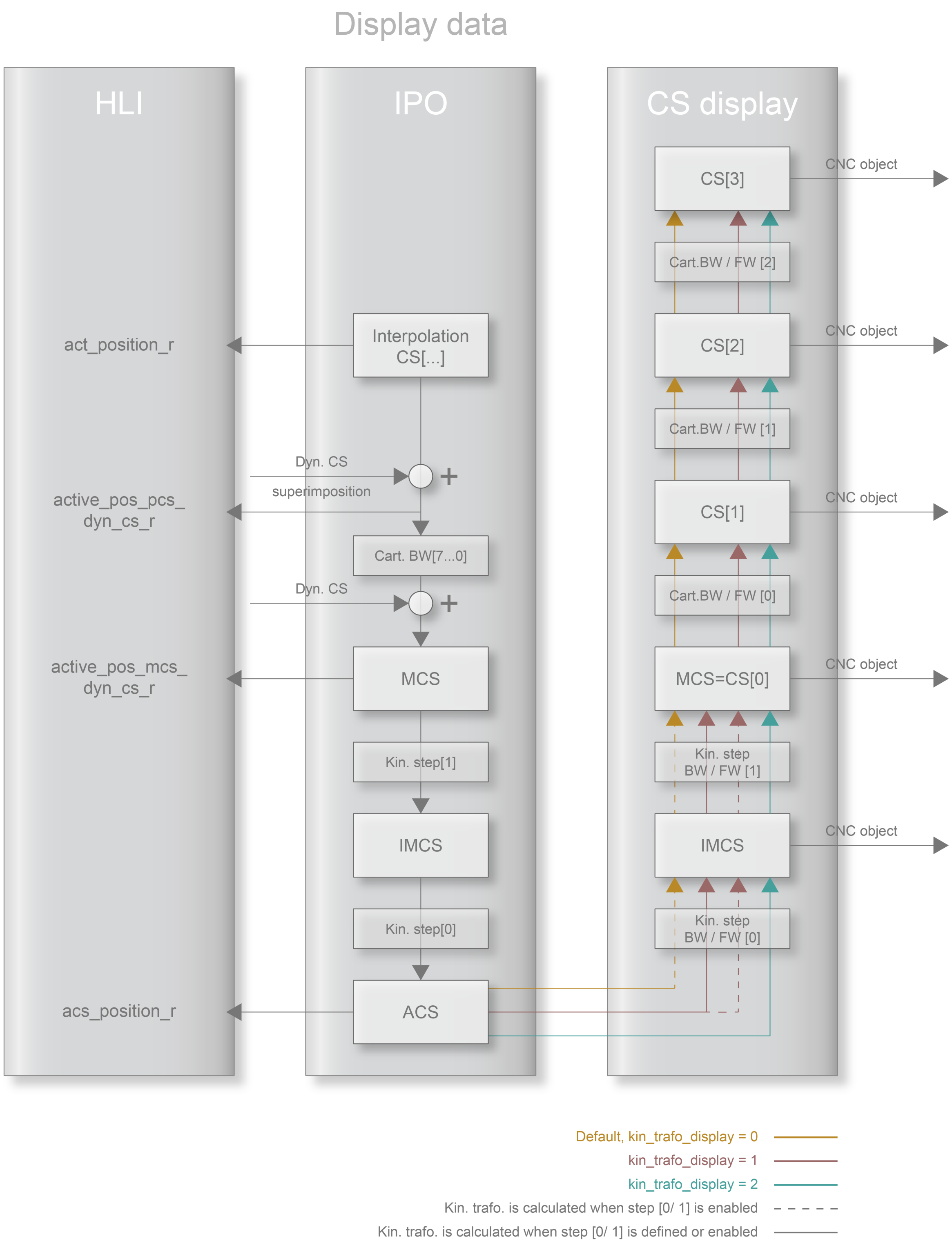


Fig. 42: Overview of display options

Based on the above overview, an example could show how the #CS command defines 3 Cartesian coordinate systems. The command can be used to assign names to them, e.g. ACS, APP and PCS.

The names can be read by the following CNC objects:

- mc_cs_0_name_r
- mc_cs_1_name_r
- mc_cs_2_name_r

The relevant positions of each coordinate system are contained in the following CNC objects:

- mc_cs_0_pos_0_r: The position of the first axis (index 0) in the CS with index 0
- mc_cs_0_pos_1_r
- mc_cs_0_pos_2_r
- mc_cs_0_pos_3_r
- mc_cs_0_pos_4_r
- mc_cs_0_pos_5_r

For the 2nd coordinate system:

- mc_cs_1_pos_0_r to mc_cs_1_pos_5_r.

and for the 3rd coordinate system:

- mc_cs_2_pos_0_r to mc_cs_2_pos_5_r.

2.6.2 Status signals

Status signals of the escape channel

The following signals are available for the escape channel:

- escape_enabled_r [▸ 105]
- program_end_r [▸ 108]
- escape_trigger_is_suspended_r [▸ 106]
- escape_path_length_r [▸ 106]
- escape_strategy_r [▸ 106]

The escape_enabled_r [▸ 105] signal indicates that the escape channel was activated by #TRACK CHAN ON [▸ 97] [ESCAPE] and can be started by the escape start trigger (backward motion by negative external velocity or flush command).

If the interpolator reaches program end, this means that the escape has ended, i.e. no escape is currently executed. The signal is located on the HLI in program_end_r.

The NC command #CHANNEL INTERFACE OFF [▸ 95] [ESCAPE WAIT] is used to temporarily suspend the trigger for escape and flush. A suspended escape trigger is considered by the escape_trigger_is_suspended_r [▸ 106] signal.

The length of the escape path from escape start to escape end without the possibly programmed length of path erosion geometry is indicated by the escape_path_length_r [▸ 106] signal.

The current strategy of the escape channel is indicated by the escape_strategy_r [▸ 106] signal.

Status signals of the orbit channel

The following signals are available for the orbit channel:

- orbit_active_r
- orbit_radius_zero_r
- orbit_wait_extend_ncbl_r

The orbit_active_r status signal indicates that the extend function is active in the orbit channel.

The orbit_radius_zero_r status signal indicates that the radius is 0 in the orbit channel.

The orbit_wait_extend_ncbl status signal indicates that the equidistant calculation of the current block is not yet completed.

Other status signals for die sinking

- jump_acitve_r [▶ 107]
- approach_active_r [▶ 107]

The jump_acitve_r [▶ 107] status signals indicates whether a flushing process is active. The signal is TRUE as soon as a flushing process is commanded by the PLC: When the floating point is reached, the signal reverts to FALSE.

When the system changes to path geometry during an escape or flushing process, this is indicated by the approach_active_r [▶ 107] signal.

Other status signals

The following status signals may also be used for die sinking.

- inside_rt_loop_r [▶ 108]
- ext_command_speed_valid [▶ 108]
- wait_ext_command_speed_r [▶ 108]
- program_end_r [▶ 108] (from interpolator)
- dist_prog_start [▶ 108] (distance to program start)

2.7 Use of real-time loops

Die sinking involves various repetitions of motions and control instructions that take place in the down and orbit channels. In general, these repetitions can be implemented by loops in the NC program. However, this has the following disadvantages:

- Greater system load since the entire NC channel is in use for loops at decoder level.
- Loops can only be influenced at decoder level to a limited extent. To evaluate loop conditions in real time, it is therefore necessary to synchronise interpolation (real-time) and decoding (non-real-time) (#FLUSH WAIT), but this requires a short interruption in process flow.

The real-time loops function therefore provides a solution. Real-time loops are executed at interpolation level with the following syntax:

```
N010 X100 Y100    ; start position
N100 X50 Y0       ; position at loop entry
N200 #RT WHILE
N210 G02 …        ; loop content
N220 G01 …
N290 G01 X50 Y0
N300 #RT ENDWHILE
N400 X100 Y-100  ; loop exit
```

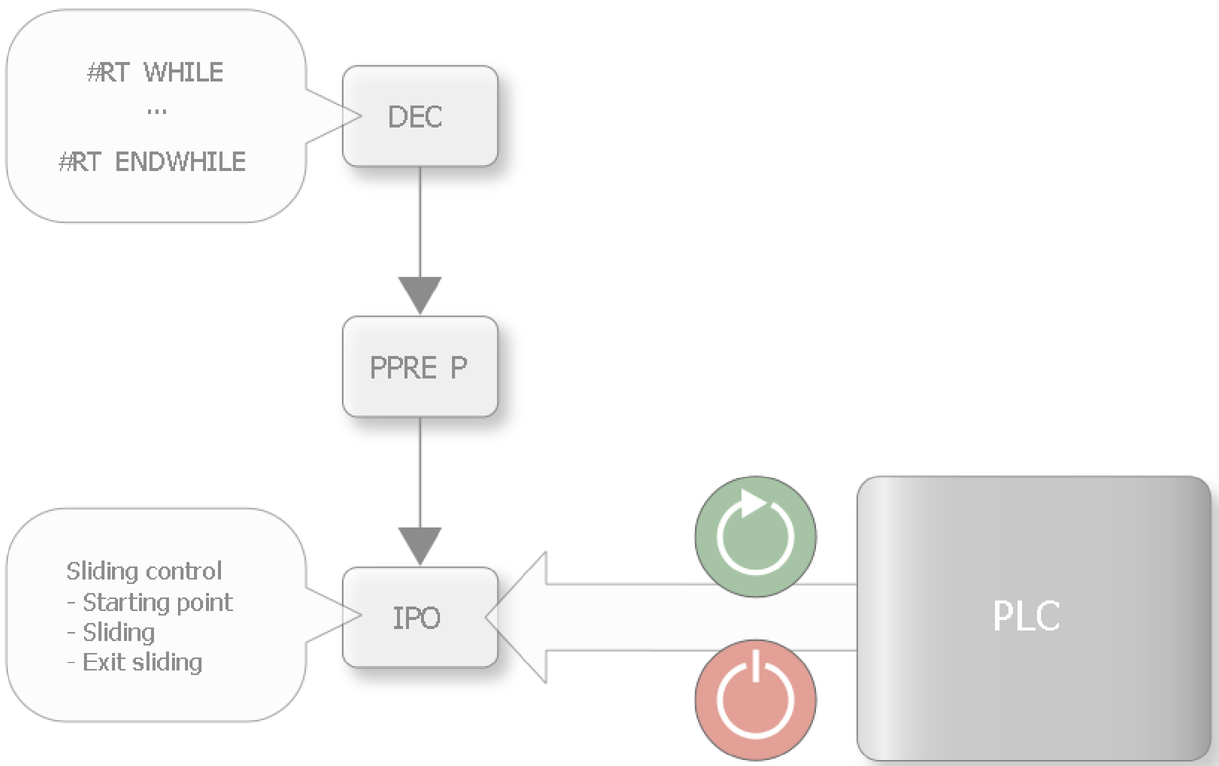


Fig. 43: Real-time loop with PLC control

The use of real-time loops in die sinking provides the following benefits:

- A real-time loop can be used in the down channel during the orbiting phase, e.g. to stop machining immediately during the finishing phase as soon as the required surface finish is achieved. This is not possible with a standard loop since the number of required loop passes is not yet known at the programming stage. See the example programming above.
- In certain machining operations, a change of behaviour may be required between the down and the orbing channels during the process. For example, a change between synchronous orbiting (constant velocity independent of the down channel) or star orbiting during machining can be implemented with real-time loops during the process.

3 Programming

3.1 General channel initialisation/coupling

The individual erosion phases of positioning, erosion on the path, orbiting and end of machining are implicitly defined by channel couplings (see Down channel [▶ 20]). The figures below give an overview of how these channel couplings are implemented in each off the channels in the programming.

It should be noted that the interfaces are always unidirectional, i.e. a channel can write data to an interface that it opens and the partner channel can read data from it. However, in order to receive data from the partner channel, the partner channel must also open an interface. Therefore, every

```
#CHANNEL INTERFACE ON      ; open interface to write
```

NC command in transmitting channel is followed by a

```
#TRACK CHAN ON              ; open interface to read
```

NC command in receiving channel.

The end point of the erosion phase on the path must have the PCS position 0 for the X and Y axes.

The orbit channel can only be decoupled if the down channel is in the centre (radius = 0). The down channel may not be superimposed on the escape channel at this time.

The escape channel can only be decoupled if no escape motion is active at this time. To prevent decoupling of the escape channel while an escape motion is active, use the #CHANNEL INTERFACE OFF [▶ 95] **[ESCAPE WAIT] command.]**

The arrows in the figures below show the direction of information flow.

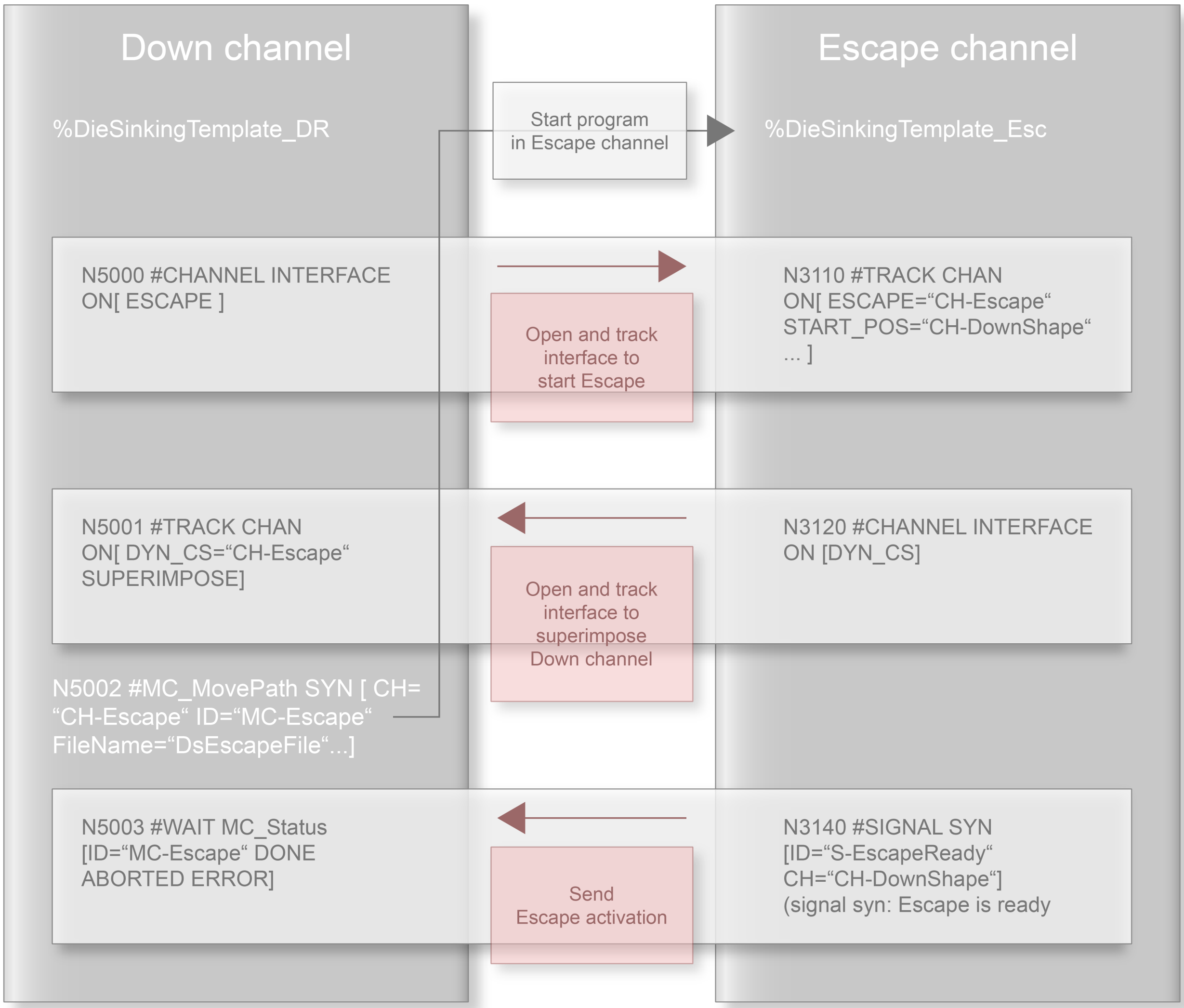


Fig. 44: Channel interfaces between down and escape channels in the NC program

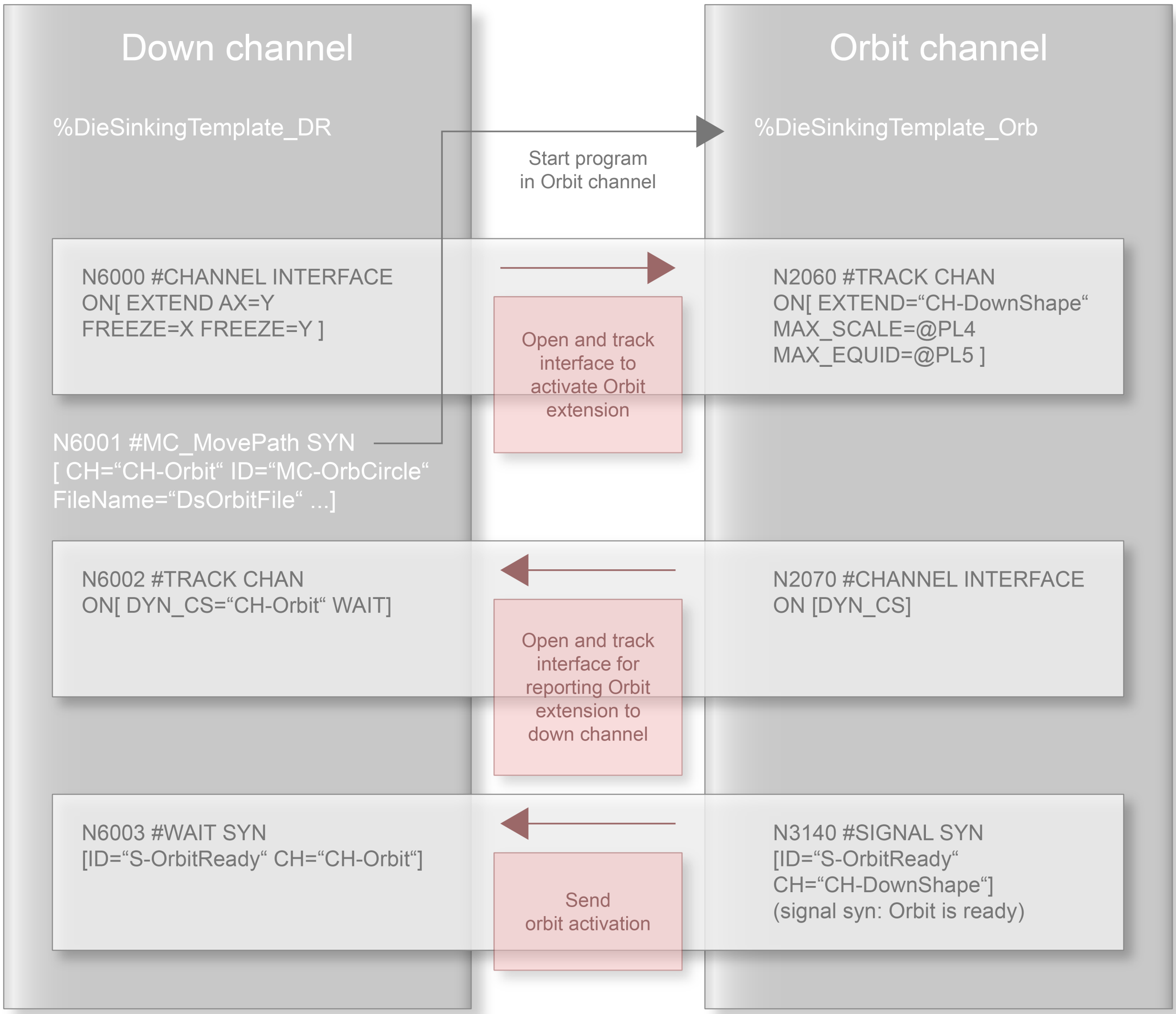


Fig. 45: Channel interfaces between down and orbit channels in the NC program

In summary, this results in the basic overall structure for a standard die sinking program as shown in the figure below.

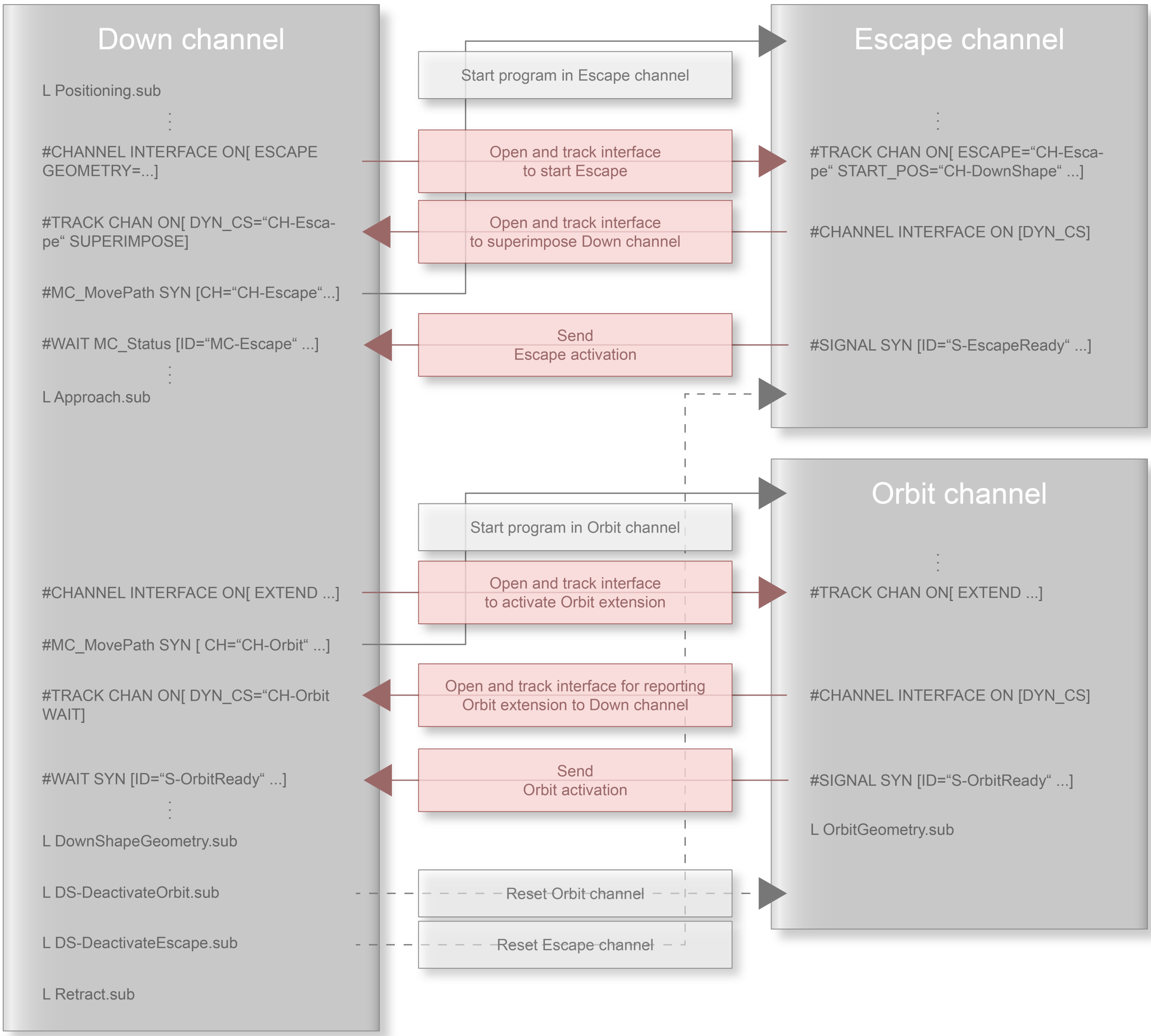


Fig. 46: General overview of channel couplings in a standard die sinking program

The next section presents NC program templates for the down channel. First, the overall program is discussed before the programs are presented for coupling the escape and orbit channels by the down channel (figures / left half of the figure).

This is followed by a presentation of the general program structure including the program sequences for coupling channels from both the orbit channel and the escape channel.

3.2 Standard program structure of down channel

The down channel is the main channel that commands the other channels and controls the die sinking process.

The basic structure of a die sinking program in the down channel is presented in the programming example below.

Basic structure - die sinking program in the down channel

```
%Senkkanal Beispielprogramm

N010 "DsOrbitFile" = "PlanetärkanalProgramm.nc"
N020 "DsEscapeFile" = "RückzugskanalProgramm.nc"

N030 #SLOPE [TYPE=STEP]
N040 L Positionierung.sub

;---- Activate the escape channel -> implicit start of erosion on the path
N050 #CS ON [EAB] [0,0,0, 0,0,0] ; Define CS
N060 L DS-ActivateEscape-ACS.sub

N070 L ErosionAufBahn.sub

; Limit the dynamics
N080 G128=50 ; velocity
N090 G131=50 ; acceleration

;---- Activate the orbit channel -> implicit start of orbiting
N100 #CHANNEL INTERFACE OFF[ESCAPE WAIT]
N110 #CS ON [PCS] [0, 0, 0, 0, 0, 0]
N120 L DS-ActivateOrbit.sub
N130 #CHANNEL INTERFACE ON[ESCAPE]

N140 L SenkAufweitGeometrie.sub

N150 #CHANNEL INTERFACE OFF[ESCAPE WAIT]
;------ Deactivate channels

N160 L DS-DeactivateEscape.sub
N170 L DS-DeactivateOrbit.sub
N180 L RückzugDerElektrode.sub
N190 #CS DEL ALL
N200 M30
```

The macros of the subroutine names for the orbit and escape channels are created in N010 and N020 for later use. The use of macros allows the same subroutine to be called up to activate the other channels [coupling programs [▶ 64]] in different programs. This allows the simple exchange of geometries.

Select the not jerk limited acceleration profile in N030 and the approach to the geometry in N040.

The coordinate system for erosion on the path is defined and activated in N050 – N070, the escape channel is activated and the path geometry is eroded.

NOTICE
It is not possible to change the coordinate system during erosion on the path.

Since the resulting motion consists of the superposition of the motion of the three channels, the dynamics in the individual channels must be limited. This is executed in the down channel in N080 and N090.

Temporary deactivation of the escape commands takes place in N100 (using the #CHANNEL INTERFACE OFF [▶ 95][ESCAPE WAIT]) command. This is done because there must be no motion in the escape channel when the machining coordinate system is changed, the the coupling with the orbit channel is activated and the coupling with the escape and orbit channels are deactivated.

This is followed by the machining coordinate system which is defined and activated in the orbiting phase in N110 and orbit channel activation in N120.

Before actual machining starts in N140, the escape channel is reactivated in N130.

The escape command is temporarily deactivated again in N150 in order to deactivate the escape and orbit channels in N160 and N170.

Finally, the electrode is retracted in N180-N220, all coordinate systems are deleted and the program ends.

3.2.1 Coupling programs

Activation/deactivation and channel coupling/decoupling of the orbit or escape channel can be programmed using standard subroutines in the down channel (this corresponds to the left half in the figures [▶ 58]). The subroutine examples used in the previous section are described below.

The variables used in the programs must be correctly assigned in advance in the down channel program.

Subroutine - Activate escape channel

```
%DS-ActivateEscape-ACS
; activation of Escape channel with saved ACS positions

N5000 $IF V.P.EscapeMode == 0
N5010 #CHANNEL INTERFACE ON[ ESCAPE GEOMETRY=FLAT ]
N5020 $ELSEIF V.P.EscapeMode == 1
N5030 #CHANNEL INTERFACE ON[ ESCAPE GEOMETRY=ALPHA ANGLE=V.P.AlphaAngle]
N5040 $ELSEIF V.P.EscapeMode == 2
N5050 #CHANNEL INTERFACE ON[ ESCAPE GEOMETRY=POINT POINT_Y=V.P.PointY POINT_Z=V.P.PointZ]
N5060 $ELSE
N5070 #CHANNEL INTERFACE ON[ ESCAPE GEOMETRY=BISECTOR BISECTOR BISEC_D1 = V.P.BisecD1 BISEC_D2 =
V.P.BisecD2]
N5080 $ENDIF
N5090 #TRACK CHAN ON[ DYN_CS="CH-Escape" SUPERIMPOSE]

; -- Start the program in the escape channel
N5100 #MC_MovePath SYN [ CH="CH-Escape" ID="MC-Escape" FileName="DsEscapeFile" \
@PL1=V.P.Appr_Start_ACS_X @PL2=V.P.Appr_Start_ACS_Y @PL3=V.P.Appr_Start_ACS_Z \
@PL4=V.P.Appr_Start_TRANS_X, @PL5=V.P.Appr_Start_TRANS_Y, @PL6=V.P.Appr_Start_TRANS_Z, \
@PL7=V.P.Appr_Start_ROT_X, @PL8=V.P.Appr_Start_ROT_Y, @PL9=V.P.Appr_Start_ROT_Z \
@PL10=V.P.Appr_Start_ACS_A @PL11=V.P.Appr_Start_ACS_B @PL12=V.P.Appr_Start_ACS_C
@PL13=V.P.EscapeMode \
@PL14=V.P.AlphaAngle @PL15=V.P.PointY @PL16=V.P.PointZ]

N5110 #WAIT MC_Status [ID="MC-Escape" DONE ABORTED ERROR]
N5120 V.P.McStatus = MC_STATUS ["MC-Escape"]
N5130 $IF V.P.McStatus != "MC_DONE"
N5140 #ERROR [ID455 MID0 RC2 PV1=V.P.McStatus PV2="MC_DONE" PM1=2 PM2=3]
N5150 $ENDIF
N5160 M17
```

Subroutine - Activate orbit channel

```
%DS-ActivateOrbit
N6000 #CHANNEL INTERFACE ON [EXTEND AX=Y FREEZE=X FREEZE=Y ]

; Start the program in the orbit channel
N6001 #MC_MovePath SYN [ CH="CH-Orbit" ID="MC-OrbCircle" FileName="DsOrbitFile" \
@PL1=V.G.SELECTED_CS.ROT.X @PL2=V.G.SELECTED_CS.ROT.Y \
@PL3=V.G.SELECTED_CS.ROT.Z @PL4=V.P.OrbitMaxScale \
@PL5=V.P.OrbitMaxEquid
N6002 #TRACK CHAN ON[ DYN_CS="CH-Orbit" WAIT]
N6003 #WAIT SYN [ID="S-OrbitReady" CH="CH-Orbit"]
N6004 #BACKWARD STORAGE CLEAR
N6005 M17
```

Subroutine - Deactivate escape channel

```
%DS-DeactivateEscape.sub
N7005 #CHANNEL INTERFACE OFF [ESCAPE]
N7006 #TRACK CHAN OFF[DYN_CS SUPERIMPOSE]
N7007 #MC_GroupResetForced SYN [ CH="CH-Escape" ID="MC-Reset"] ( reset escape channel
N7008 #WAIT MC_Status [ID="MC-Reset" DONE ABORTED ERROR]
N7009 M17
```

Subroutine - Deactivate orbit channel

```
%DS-DeactivateOrbit.sub
N7000 #CHANNEL INTERFACE OFF [EXTEND]
N7001 #TRACK CHAN OFF [DYN_CS]
N7002 #MC_GroupResetForced SYN [ CH="CH-Orbit" ID="MC-Reset" ] ( reset orbit channel
N7003 #WAIT MC_Status [ID="MC-Reset" DONE ABORTED ERROR]
N7004 M17
```

3.2.2 Escape strategy

Different parameters are required depending on the escape strategy to be started. They must be specified in the following command:

#CHANNEL INTERFACE ON [ESCAPE GEOMETRY= <FLAT | ALPHA | POINT | BISECTOR> [ANGLE=..] [POINT_Y=..] [POINT_Z=..] [BISEC_D1=..] [BISEC_D2=..]]

3.3 Standard program structure of orbit channel

Below is an example of an NC program for an orbit channel which is started from the down channel in the orbit channel as shown in the orbit channel coupling.

Standard program structure of orbit channel

```
%DieSinkingTemplate_Orb
N2010 G128=50 ; speed
N2020 G131=50 ; acceleration
N2030 G133=50 ; ramp times

N2040 G00 X0 Y0 Z0
N2050 #TRACK CHAN ON [EXTEND="CH-DownShape" MAX_SCALE=@PL4 MAX_EQUID=@PL5]
N2060 #CHANNEL INTERFACE ON [DYN_CS]
N2070 #BACKWARD STORAGE CLEAR

(-- Same orientation in orbit channel as in down channel--)
N2080 #CS ADD [PCS] [0, 0, 0, @PL1, @PL2, @PL3]
N2090 #CS SELECT [PCS]

N2100 L OrbitStartingPoint.sub

N2110 #RT CYCLE DELETE [ ID="S-Orbit-RT-Loop" ]
N2120 #RT CYCLE [ ID="S-Orbit-RT-Loop", SCOPE=GLOBAL ]
```

```
N2130 V.RTG.LOOP.ENABLED = TRUE
N2140 #RT CYCLE END

N2150 #SIGNAL SYN[ID="S-OrbitReady" CH="CH-DownShape"]

N2160 #RT WHILE
N2170 L Planetärgeometrie.sub
N2180 #RT ENDWHILE
N2190 #CS DEL ALL
N2200 M30
```

The dynamics are limited in N2010 - N2030 due to the superimposed channel constellation.

The channel is moved to its start position in N2040.

The channel coupling to the down channel is created in N2050 by the orbit channel. The command is the counterpart of block N6000 in orbit channel coupling. It also defines the maximum scaling radius and the maximum equidistant distance. These parameters are transferred from the down channel when the program is called.

The interface of the orbit channel to the down channel is opened in N2060. The X and Y MCS coordinates are transferred from the orbit channel to the down channel via this interface.

N2070 ensures that it is not possible to move backwards over this block.

A machining coordinate system is generated and selected in N2080 and N2090.

N2100 contains the motion to the starting point of the actual orbit geometry.

The second motion block is implicitly interpreted as an orbit geometry block and must therefore be part of the orbit geometry.

N2110-N2140 contain the definition of the real-time cycle for the endless real-time loop in N2160-N2180 [FCT-C32 Real-time cycles].

N2150 sends a signal to the down channel so that it can continue machining. The down channel waits for this signal in block N6002, as described in the NC program in the section Coupling programs [▶ 64].

With an equidistant orbit geometry, the orbit channel is only ready when a closed orbit geometry is present in the interpolator of the CNC. This is indicates on the HLI by the "orbit_active_r [▶ 107]" signal. With an equidistant orbit geometry, machining in the down channel should only be continued after this signal is set to True.

The actual orbit geometry is generated in an endless real-time loop in N2160-N2180.

All machining coordinate systems of the orbit channel are deleted in N2190.

3.3.1 Fast positioning in the orbit channel

With star orbiting, the orbit channel is at fixed positions while the down channel expands and returns to the centre. As soon as the down channel reaches the centre, it should move to a new position in the orbit channel as fast as possible so that the down channel can expand again at the new position. The function described below is designed to implement fast positioning as quickly as possible.

#CHANNEL SET [FAST_FORWARD_IN_CENTER=ON/OFF]

It enables the position in the orbit channel to be set to the last position before channel synchronisation at radius 0 instead of doing this by interpolation. This is permitted since no physical axis motions are executed during a motion in the orbit channel at radius zero. Channel synchronisation is executed using the “SIGNAL SYN” and “WAIT SYN” commands.

This command only works for motions of the three Cartesian spatial axes. An error is output if additional axes are moved.

If the radius is not 0, this program command has no effect.

Below is a program excerpt in which the presented command is used:

```
N100 X10 Y0 F1000
N110 #CHANNEL SET [FAST_FORWARD_IN_CENTER=ON]
N120 $WHILE 1
N130 #WAIT SYN[ID 1000 CH1]
N140 G02 X-10 Y0 R10
N150 #SIGNAL SYN[ID 2000 CH1]
N160 #WAIT SYN[ID 1000 CH1]
N170 G02 X10 Y0 R10
N180 #SIGNAL SYN[ID 2000 CH1]
N190 $ENDWHILE
N200 #CHANNEL SET [FAST_FORWARD_IN_CENTER=OFF]
```

3.4 Escape channel

The escape channel executes electrode escape motions that are superimposed on the motion of the down channel. The escape motion is triggered by commanding a negative velocity or a flushing process in the escape channel.

A standard program of the escape channel is shown below.

Standard program - escape channel

```
%Rückzugskanal_Beispielprogramm
N3030 X[SET_POSITION POS=@PL1] Y[SET_POSITION POS=@PL2] Z[SET_POSITION POS=@PL3]
N3040 A[SET_POSITION POS=@PL10] B[SET_POSITION POS=@PL11] C[SET_POSITION POS=@PL12]

N3050 #SLOPE [TYPE=STEP]
N3060 #ESCAPE PATH DEF BEGINN3070 #CS ON [EAB] [@PL4,@PL5,@PL6,@PL7,@PL8,@PL9]
N3080 #ESCAPE PATH BACKWARD STOP
N3090 L ErosionAufBahn.sub
N3100 #ESCAPE PATH POST SEQUENCE
N3110 #CS DEL ALL
N3120 #ESCAPE PATH DEF END

N3130 G128=50 ; velocity
N3140 G131=50 ; acceleration

N3150 #TRACK CHAN ON[ ESCAPE="CH-Escape" START_POS="CH-DownShape" EXTEND_PARAM="CH-Orbit"]
N3160 #CHANNEL INTERFACE ON [DYN_CS]
N3170 #SIGNAL SYN [ID="S-EscapeReady" CH="CH-DownShape"]
N3180 M30
```

This program is started by the down channel when the escape channel is coupled (see N5002 in section Coupling programs [▶ 64]).

N3030 and N3040 set the positions of the escape channel without a motion occurring. The positions to be set are transferred from the down channel at program start and correspond to the positions at the start of the geometry for erosion on the path.

The not jerk limited acceleration profile is selected in N3050.

In the escape channel, N3060 marks that the geometry description follows for erosion on the path. This motion is only saved by this mark so that the escape motion can be executed on this path.

The transfer parameters from the down channel in N3070 select the identical coordinate system as in the down channel.

The command in block number N3080 indicates that it is not permitted to retract further than this point.

Block number N3090 calls the identical subroutine containing the geometry for erosion on the path as in the down channel.

The end of the geometry motion blocks is marked in N3100. Switch and additional functions, e.g. M01, can still be programmed.

The coordinate system is deselected in N3110 and the entire end of the geometry is marked in N3120.

NOTICE

The coordinate systems must be deselected at the end of the geometry for erosion on the path.

As in the other programs, the dynamics are limited in N3120 and N3130 due to the superimposed channel constellation.

The escape strategy is parameterised in N3140. Here the channel from which the escape channel receives its start position must be specified as well as the number of the channel that acts as the orbit channel.

The escape channel opens an interface in N3150. This is the counterpart to N5001 (see section Coupling programs [▶ 64]).

In block number N3160, a signal is sent to the down channel to indicate that the escape channel is ready.

3.5 Applications

This section contains programming examples of how various applications can be implemented.

The PLC can specify the velocity to achieve different erosion modes together with specific geometries for various machining operations. The table below gives an overview of the different velocities specified for the erosion modes described in this section.

	Down channel	Orbit channel	Escape channel	Escape start condition
Half spherical orbiting	V_{prog}	V_{prog}, start / stop	V_{ext}	V_{ext} <0
Star orbiting	V_{ext}	V_{prog}, start / stop	V_{ext}	V_{ext} <0 and down (& orbit channel) stopped
Asynchronous orbiting	V_{ext}	V_{prog}	V_{ext}	V_{ext} <0 and down (& orbit channel) stopped
Alternating orbiting	V_{ext}	V_{ext}	V_{ext}	V_{ext} <0 and down (& orbit channel) stopped

Controlling erosion modes by specifying the velocities:

V_{prog} Programmed velocity

V_{ext} Velocity via external velocity interface

V_{Gen} Specified generator velocity

3.5.1 Asynchronous orbiting

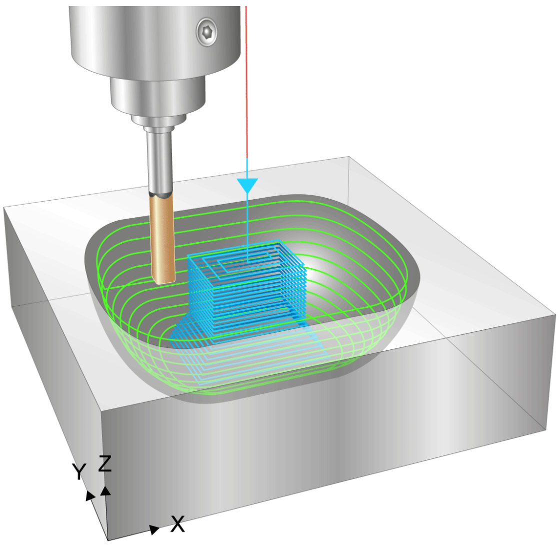


Fig. 47: Application - asynchronous orbiting

Process control by PLC

Down channel	Orbit channel	Escape channel	Escape start condition

V_{ext} =0, when dist_prog_start$_{Esc}$ > Sliding distance or V_{Gen} < 0 V_{ext} = V_{Gen}, when V_{Gen}> 0 and dist_prog_start$_{Esc}$ < Sliding distance Use of Insert command CU [▸ 114] for erosion on discs	V_{prog}	V_{ext}	V_{ext} <0 and down (& orbit channel) stopped

In asynchronous orbiting, the program flows in the down and orbit channels in parallel and independently of each other. No further synchronisation takes place between these two channels. The velocity in the channels is completely asynchronous.

A motion in the escape channel is executed at negative generator velocity and when the down channel is at standstill. As the contour is approached, the escape motion and the program progress of the down channel can be superimposed within a sliding distance. The Insert command CU [▸ 114] can insert motion stops online in the down channel. While the down channel is at standstill, a disc is eroded due to the parallel motion of the orbit channel.

Geometry using NC programming

In this erosion mode, the geometries of the down and orbit channels are programmed independently of each other. The programming example below shows the two geometries of the down and orbit channels resulting in the geometry shown in the figure.

The geometry in the down channel is programmed in the subroutine “SenkenAufweitenRückzug.sub” taken from the programming example of the basic structure [Standard program structure of down channel [▸ 62]].

The geometry in the orbit channel is programmed in the subroutine “OrbitGeometry.sub” taken from the programming example of the basic structure [Standard program structure of orbit channel [▸ 65]].

Down channel - asynchronous orbiting

```
; SenkenAufweitenRückzug geometry
N0380 G19 G91
N0410 G01 Y20 ; expand
N0450 G01 Z-26 ; down
N0520 G01 Y30 Z-36 G90 ; quadrant orbiting
N0560 G03 Y60 Z-6 J0 K30
N0580 G01 Z0 F50 ; escape
N0620 G01 Y0 F200
N0630 M17
```

Orbit channel - asynchronous orbiting

```
; orbit motion, rounded rectangle
;P50 (* radius of corner rounding *)
;P100 (* X outer corner point *)
;P200 (* X inner corner point *)

;P300 (* Y inner corner point *)
;P400 (* Y outer corner point *)

N2010 G03 XP200 YP400 RP50
N2020 G01 X-P200 YP400
N2030 G03 X-P100 YP300 RP50
N2040 G01 X-P100 Y-P300
N2050 G03 X-P200 Y-P400 RP50
N2060 G01 XP200 Y-P400
N2070 G03 XP100 Y-P300 RP50
N2080 G01 XP100 YP300
N2090 M17
```

3.5.2 Star orbiting

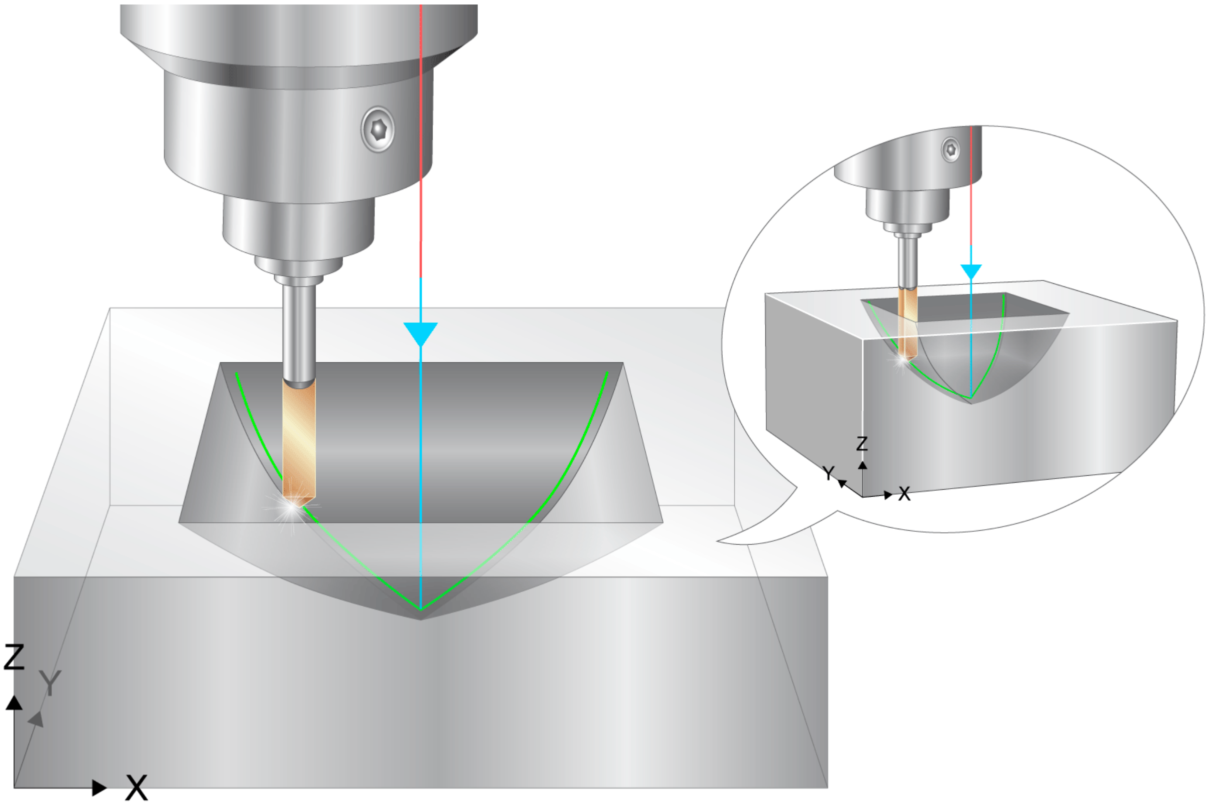


Fig. 48: Application - star orbiting

Process control by PLC

Down channel	Orbit channel	Escape channel	Escape condition
V_{ext} =0, when dist_prog_start$_{Esc}$ > Sliding distance or vGen < 0 V_{ext} = V_{Gen}, ,when V_{Gen} > 0 and dist_prog_start$_{Esc}$ < Sliding distance	V_{prog}, start / stop	V_{ext}	V_{ext} <0 and down (& orbit channel) stopped

The resulting geometry of the “star orbiting” machining strategy is shown in the figure in the section Superimposing geometry and escape motions [▶ 78]. This operation mode is designed to finish-machine cavities in corners. It requires the orbit channel to remain in position in each corner while the down channel expands. After expanding, the down channel must wait at the centre until the orbit channel has positioned itself in the next corner before the down channel can expand again. Synchronising the two channels can be achieved using the #SIGNAL and #WAIT commands, as shown in the following program excerpts.

One way of organising velocity planning is for the spark generator to specify an external velocity for the escape channel as usual. The velocity of the down channel is also specified by the spark generator, but if a negative velocity is specified by the spark generator, it is overwritten in the PLC by zero velocity. Consequently, escape motions are only executed by the escape channel.

Down channel - star orbiting

```
% Sternförmiges Aufweiten
N0290 G01 Z30 F1500 ; down
N0300 G19
N0310 $WHILE V.P.SliceCounter < 4 ; 3 corners
N0320 #SIGNAL SYN [ID 1000 CH2] ; signal to move to next corner
N0330 #WAIT SYN [ID 2000 CH2] ; wait until the orbit channel reaches the next corner
```

```
N0340 G03 Y30 Z60 J0 K30 ; expand
N0350 G01 Z70
N0360 G01 Z60
N0370 G02 Y0 Z30 J-30 K0 ; move back to centre
N0380 V.P.SliceCounter=V.P.SliceCounter+1
N0390 $ENDWHILE
```

Orbit channel - star orbiting

```
%L Kanalsynchronisierung
N2000 #SIGNAL SYN[ID 2000 CH1]
N2010 #WAIT SYN[ID 1000 CH1]
N2020 M17

N2170 #CHANNEL SET [FAST_FORWARD_IN_CENTER=ON]
(----- Orbit geometry -----)
N2180 G01 XP1 Y-P2 ; first corner
N2190 LL channel synchronisation; wait for signal to move to next corner
N2200 X0
N2210 X-P1
N2220 LL channel synchronisation

…
N2380 #CHANNEL SET [FAST_FORWARD_IN_CENTER=OFF]
N2390 LL channel synchronisation
(----- End of orbit geometry -----)

N2400 #TRACK CHAN OFF [EXTEND]
N2410 #CS DEL ALL

N2420 M30
```

3.5.3 Half spherical orbiting

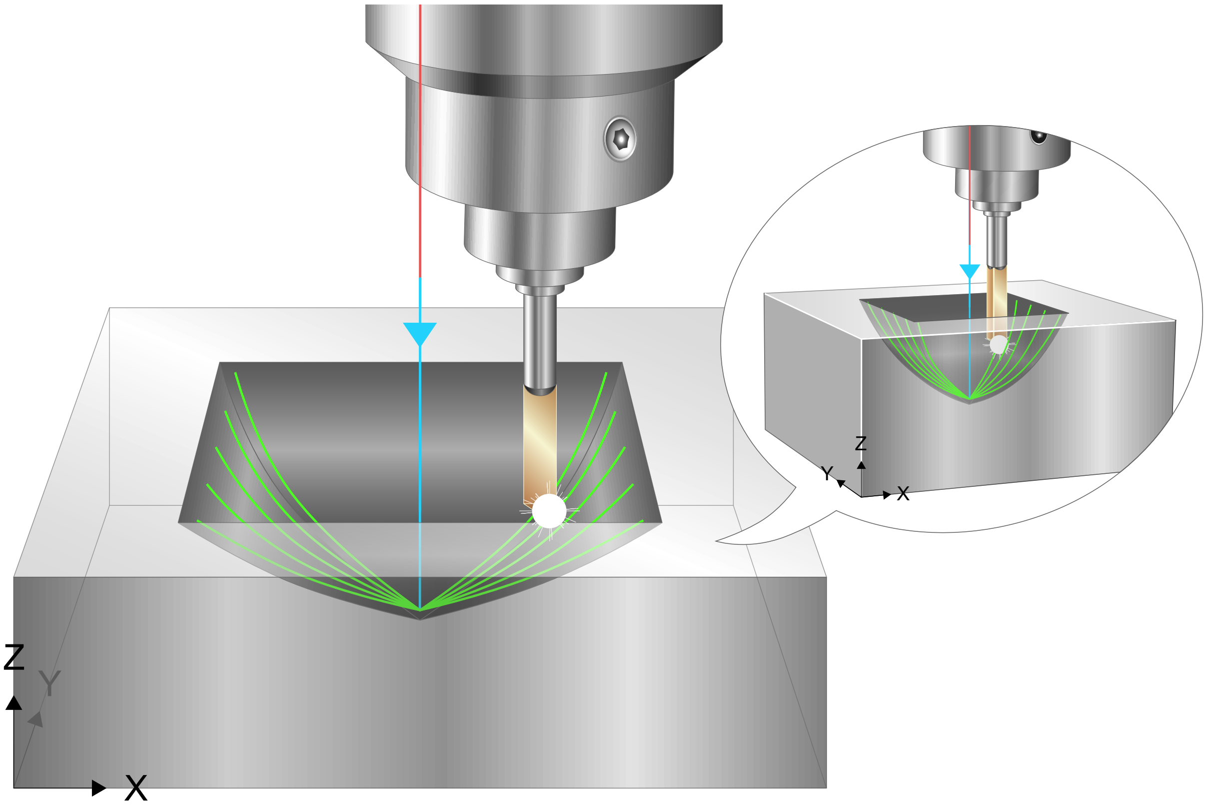


Fig. 49: Application - half spherical orbiting

Process control by PLC

Down channel	Orbit channel	Escape channel	Escape condition
V_{prog}	V_{prog}, start / stop	V_{ext}	V_{ext} <0 and down (& orbit channel) stopped

In half spherical orbiting, the down channel moves at a constant velocity. Another difference compared to star orbiting is that the down geometry has both a sector in the positive radius direction and one in the negative radius direction.

The example program uses the same contour in the down channel as for star orbiting erosion. However, a real-time loop is also used instead of a normal loop. This offers flexibility to increase or decrease the number of loops depending on the progress of the process.

Down channel - half spherical orbiting

```
% Halb-sphärisches Planetär-Aufweiten

N0270 #RT CYCLE DELETE [ID = 4711 ] ; define the real-time loops
N0280 #RT CYCLE [ID=4711 SCOPE = GLOBAL]
N0290 $IF V.E.RtLoopEnable != 0
N0300 V.RTG.LOOP.ENABLED = TRUE
N0310 $ELSE
N0320 V.RTG.LOOP.ENABLED = FALSE
N0330 $ENDIF
N0340 #RT CYCLE END

N0350 #FLUSH WAIT
N0360 G19

#RT WHILE
N0410 G01 X0 Y0 Z30 F1000 ; down
N0420 G03 Y30 Z60 J0 K30 ; expand with positive radius
N0430 G01 Z70
N0440 G01 Z60
N0450 G02 Y0 Z30 J-30 K0 ; move to centre

N0360 #OPTIONAL EXECUTION ON [SIMULATE] ; optional change to the next disc position
N0370 #SIGNAL SYN [ID "S-OrbitSectorReq" CH2] ; signal for orbit channel to move to the next
position
N0380 #WAIT SYN [ID "S-OrbitSectorOk" CH2] ; wait until orbit channel is in position
N0390 #OPTIONAL EXECUTION OFF

N0460 G02 Y-30 Z60 J0 K30 ; expand with negative radius
N0470 G01 Z70
N0480 G01 Z60
N0490 G03 Y0 Z30 J30 K0 ; move to centre
N0500 #RT ENDWHILE

N0550 G01 Y0 F100 ; end machining
N0560 M17
```

Orbit channel - half spherical orbiting

```
%L Kanalsynchronisierung
N2000 #SIGNAL SYN[ID "S-OrbitSectorOk" CH1]
N2010 #WAIT SYN[ID "S-OrbitSectorReq" CH1]
N2020 M17

N2180 $WHILE 1
N2190 #WAIT SYN[ID "S-OrbitSectorReq" CH1]
N2200 G01 XP1 Y-P2 F2000

N2210 LL channel synchronisation
N2220 X-P1

(-- divided orbit geometry --)
N0400 ...
N2410 #SIGNAL SYN[ID "S-OrbitSectorOk" CH1]
N2420 $ENDWHILE
```

3.5.4 Alternating orbiting

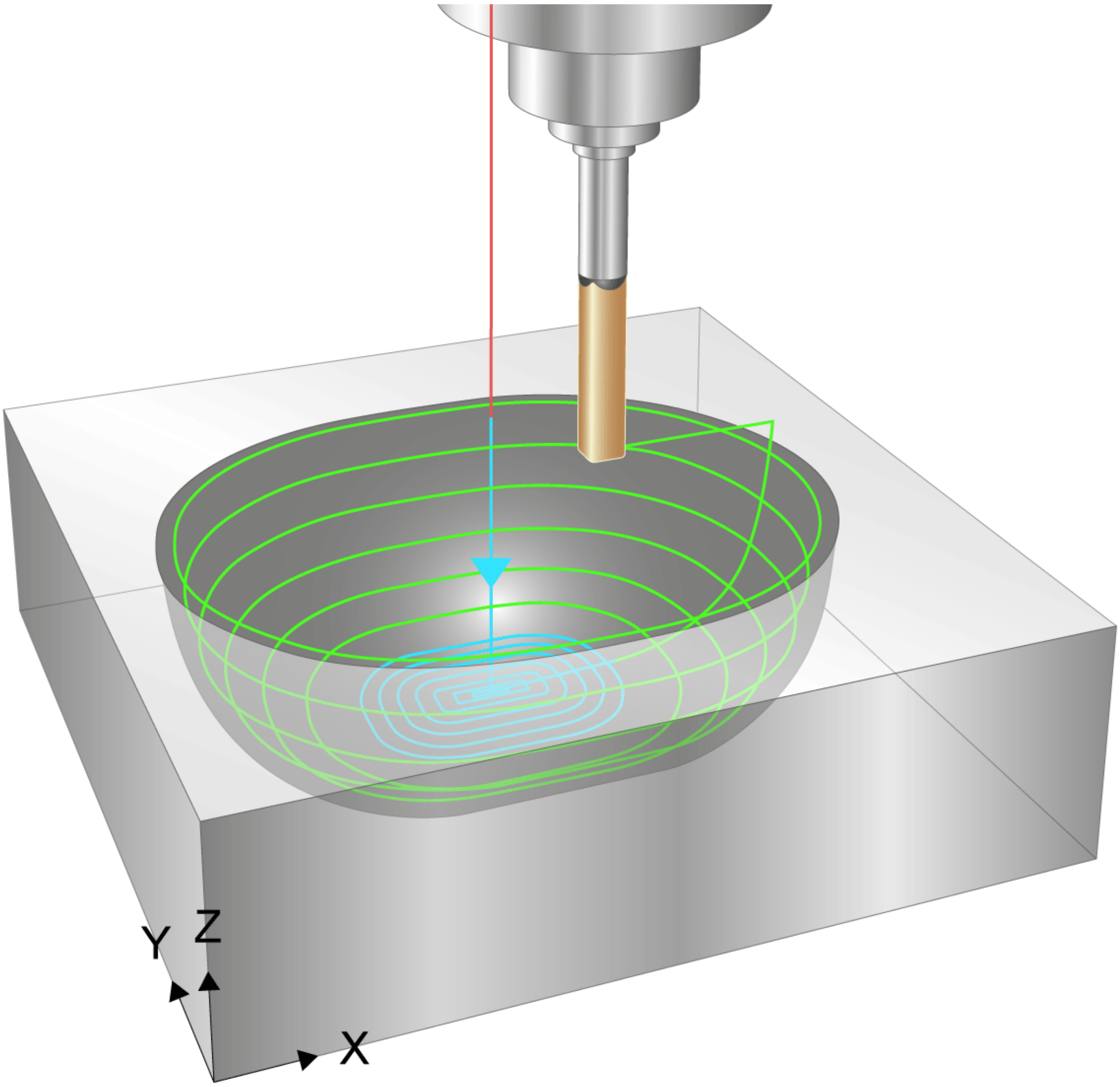


Fig. 50: Application - alternating orbiting

Process control by PLC

Down channel	Orbit channel	Escape channel	Escape condition
V_{ext} =0, when dist_prog_start$_{Esc}$ > Sliding distance or v_{Gen} < 0 V_{ext} = V_{Gen}, when V_{Gen}> 0 and dist_prog_start$_{Esc}$ < Sliding distance	V_{ext}	V_{ext}	V_{ext}<0 and down (& orbit channel) stopped

In alternating orbiting, the down channel and the orbit channel always move in alternating sequence. Channel synchronisation always takes place between the down and orbit channels. The down channel moves to the next position at which a disc is eroded. After it reaches the position, it synchronises with the orbit channel and the orbit channel erodes a complete disc. When the entire disc has been eroded, the orbit channel synchronises with the down channel and the process starts again from the beginning.

3.5.5 Superimposing geometry and escape motions

Control by the PLC

The PLC can fully monitor and control the synchronisation of the down channel and escape channels, i.e. the PLC can achieve the required response using the appropriate interfaces.

- The PLC is the master.
- The down channel and escape channel are slaves.

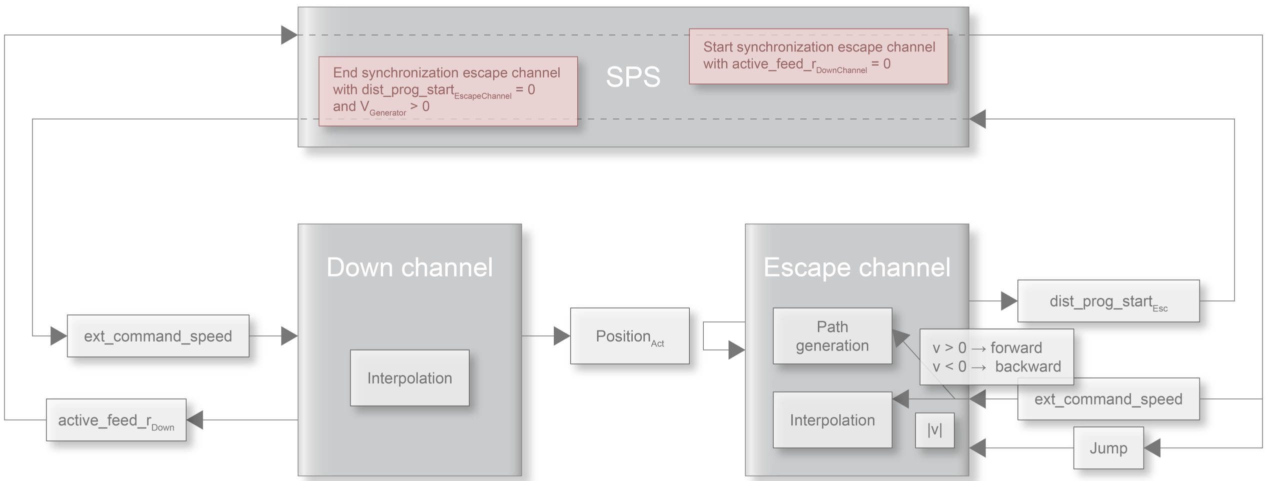


Fig. 51: The PLC controls down and escape motions.

In standard machining operations, the escape channel only superimposes the down channel until the down channel is at standstill. This condition is monitored by the CNC while erosion takes place on the path and if ignored, it results in an error message and a machining process abort.

During the orbiting phase, it is possible to simultaneously move and superimpose the two channels. The escape path is adapted online to the changed start position of the down channel. Superimposition results in a type of looping of the escape and geometry motions.

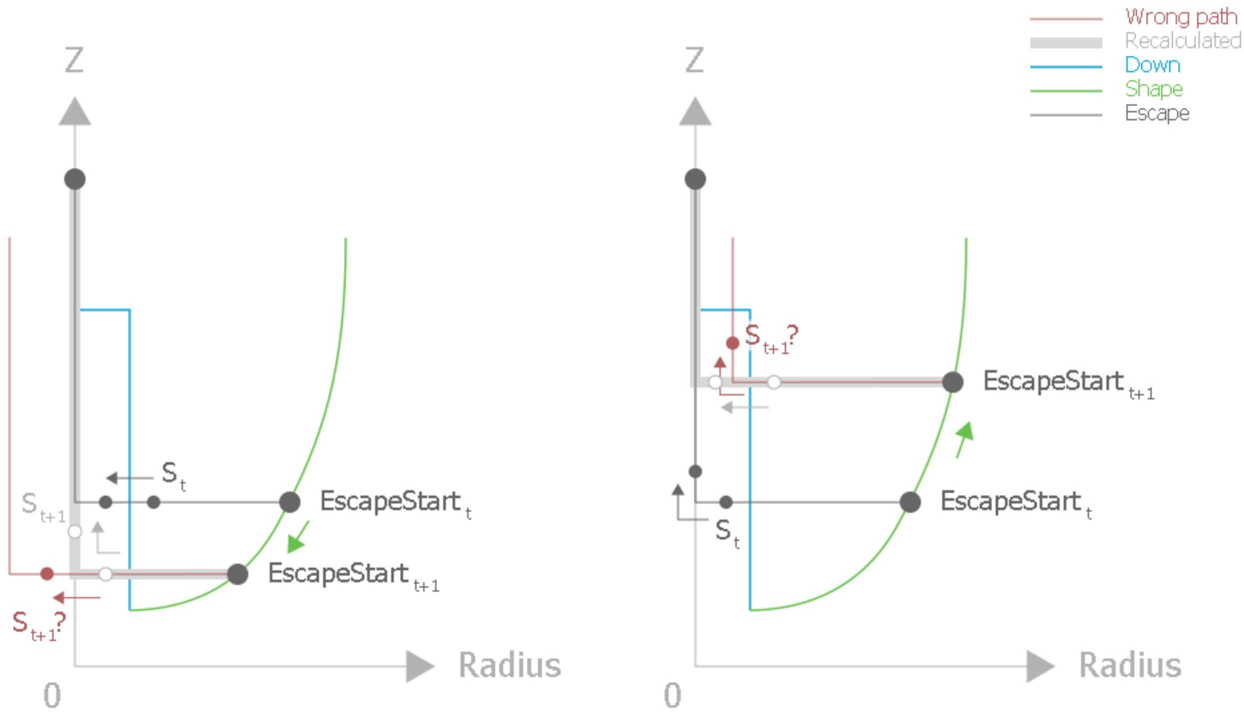


Fig. 52: Adapting the escape path with simultaneous change in position of the down channel

However, superimposition may lead to dynamic problems. In this machining operation, the electrode can be “pushed” over corners without a motion stop at the bending point. In the worst case, this can result in additional axis acceleration.

With

$$a_{add} = \frac{v_B}{T_{cyc}}$$

v_B : path velocity in down channel

T_{cyc} : Cycle time

$$v_B = 100\frac{mm}{min} = 1{,}66 * 10^{-3}\frac{mm}{s}$$: standard value for path velocity.

Regardless of the programmed feed rate or the external commanded velocity specified by the PLC, the maximum path velocity in the down channel can be limited to this standard value by the command #VECTOR LIMIT ON [VEL=100].

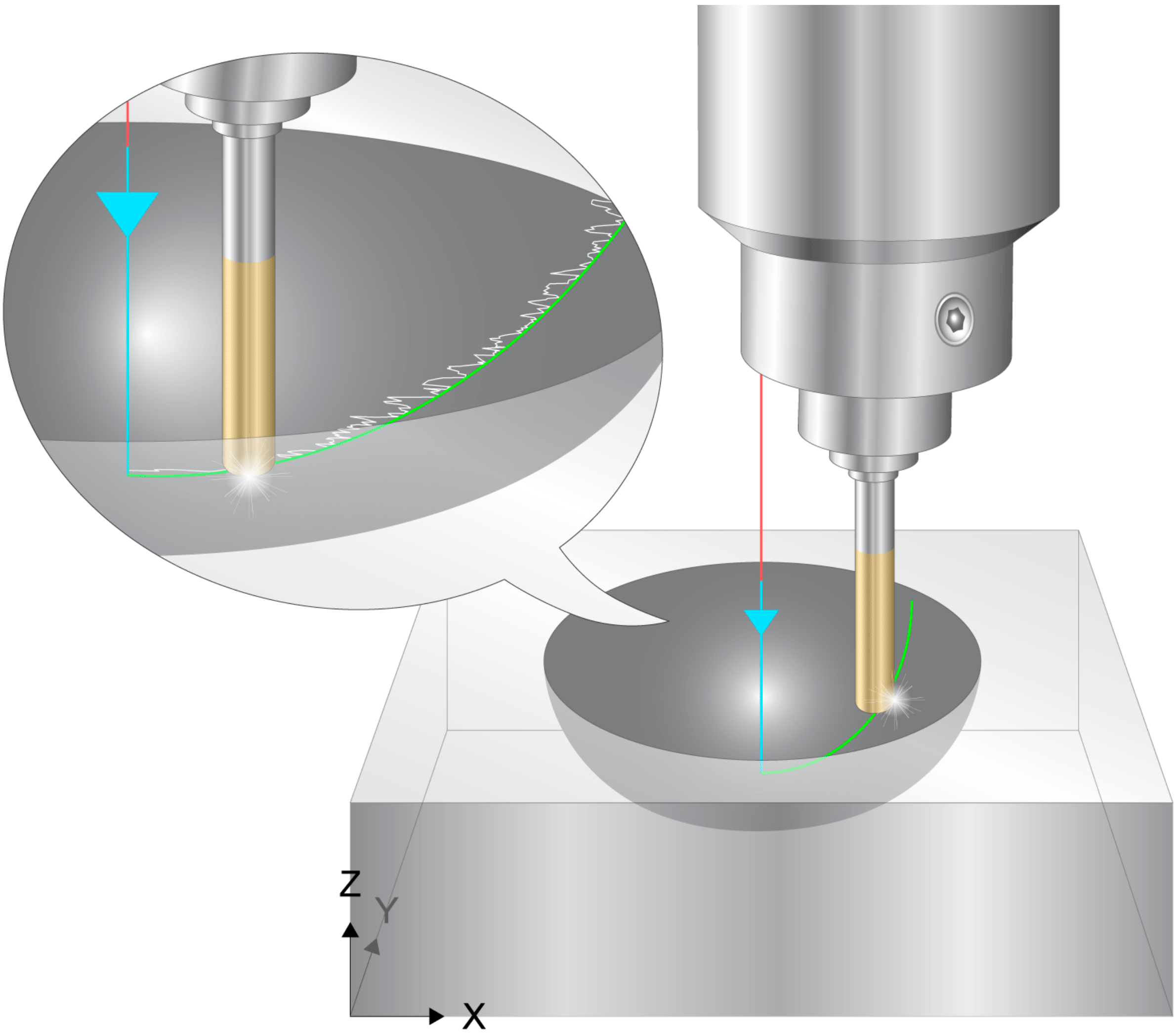


Fig. 53: Application example - superimposed escape motion

Sliding end of the escape motion

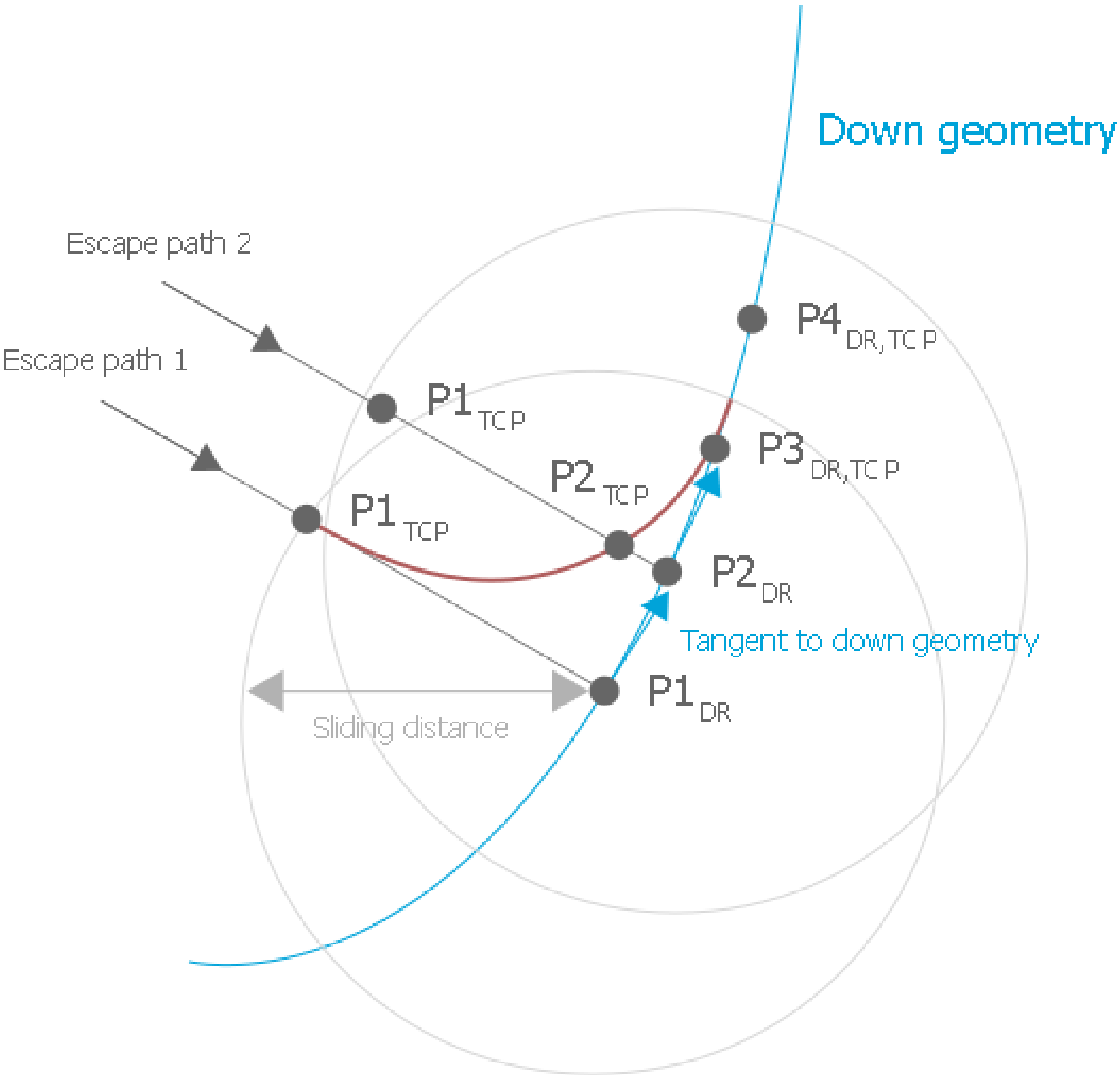


Fig. 54: Geometric diagram of superimposed escape motion

When the electrode returns to the contour after withdrawing through the escape channel, the simultaneous commanded velocity specified by the PLC on the down and orbit channels allows a soft return to the workpiece as of a specified distance.

This distance defines a circular area about the starting point of the down channel and can be defined by the dist_prog_start control unit [▶ 108] [HLI, section: Status information of a channel] of the escape channel.

3.6 Dynamics limitations

In the orbiting function, the resulting electrode motion is obtained from superposition of the three channels. The dynamics of the individual channels must be limited in these phases so that superposition does not result in a violation of axis dynamics. Since only the down or escape channel may move the electrode during erosion on the path, a weighting is not necessary in this phase.

The figure below shows the radial, tangential and total velocity vectors at sample points in the orbit and down channels.

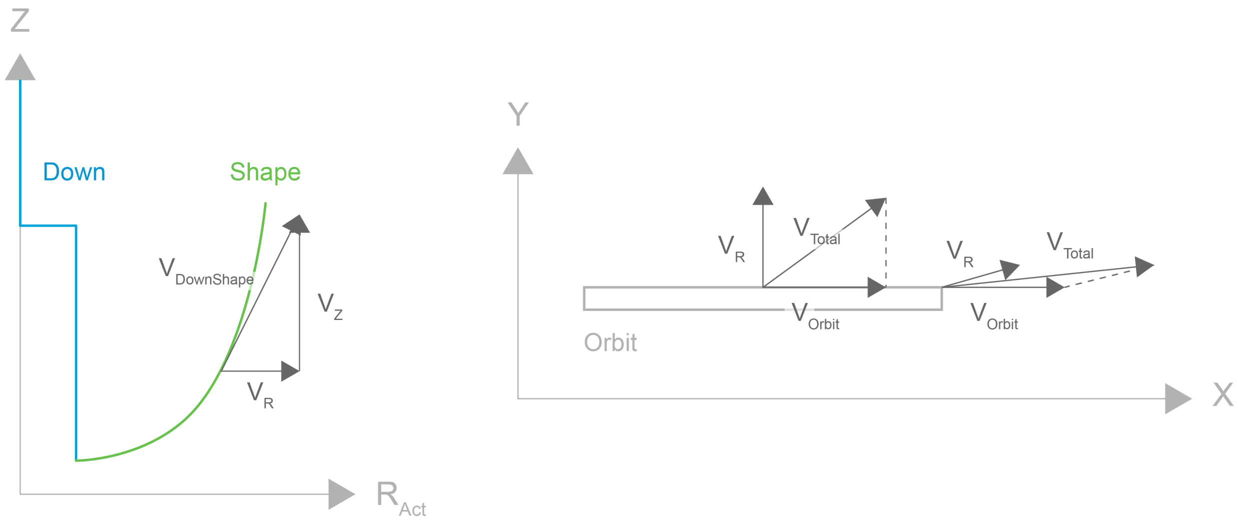


Fig. 55: Radial, tangential and total velocity vectors at sample points in the orbit and down channels.

The following applies to the path velocity vector in the down channel: $\overrightarrow{v_{DownShape}} = \overrightarrow{v_Z} + \overrightarrow{v_R}$

The same applies to the path velocity vector in the orbit channel: $\overrightarrow{v_{Orbit}} = \overrightarrow{v_X} + \overrightarrow{v_Y}$

Since the two channels are superimposed, the total velocity is therefore: $|\overrightarrow{v_{total}} = \overrightarrow{v_R} + \overrightarrow{v_{orbit}}$

Due to the dynamic limitations of the machine, the following applies: $\left|\overrightarrow{v_{R,max}}\right| = v_{max,axis}$ and

$$\left|\overrightarrow{v_{Orbit,max}}\right| = v_{max,axis}$$

For the maximum total velocity $\overrightarrow{v_{total}}$ the two cases shown on the right in the above figure are the result:

Case 1: $\overrightarrow{v_R} \perp \overrightarrow{v_{orbit}} \rightarrow |\overrightarrow{v_{total}}| = \sqrt{2} \cdot v_{max,axis}$

Case 2: $\overrightarrow{v_R} \parallel \overrightarrow{v_{orbit}} \rightarrow |\overrightarrow{v_{total}}| = 2 \cdot v_{max,axis}$

It is advisable to limit the maximum dynamic range of each channel to half of its maximum value to ensure that the maximum dynamic range is not exceeded at any time.

This can be achieved using the code lines

```
N0150 G128=50 ; velocity
N0160 G131=50 ; G1, G2, G3 acceleration
N0170 G231=50 ; G0 acceleration
N0180 G133=200 ; G1, G2, G3 ramp times
```

These lines were already used in previous programming examples.

Alternatively, dynamic values can also be limited in each channel using the #VECTOR LIMIT ON command. However, it must be noted that the effective feedhold accelerations are also limited depending on the P-CHAN-00097 command. When the stop command is used in critical cases where an overshoot of maximum axis accelerations is insignificant and deceleration is more important within the shortest possible time, feedhold can be commanded using the e_feedhold control unit.

It must also be noted that the maximum path velocity must be limited for simultaneous motion in the down and escape channels.

3.7 Using kinematics in die sinking

It is only possible to use kinematics in the die sinking configuration under the following two conditions:

1. The forward and backward transformation of the kinematic used must be unique. If a TcCOM transformation is used, the Boolean value f_UniqueTrafo must be set since a TcCOM transformation is normally not interpreted as unique.
2. Motions where kinematics have an influence on axis motions may only be programmed for erosion on the path (in the following example, motion of the C axis). The influence In the down channel geometry cannot be considered due to the 2D superposition of the down/escape channels.

In order to use kinematics, the kinematics in the down channel must be selected before activation of the escape channel. The identical kinematics must be activated in the escape channel before the geometry. It may not be deselected in the escape channel before program end.

Offset of the electrode centre point in relation to the C axis

In subsequent NC programs for the down and escape channels, the kinematics in the figure are defined as TcCOM transformation ID 503 with the radial offset in parameter 4. With this kinematic, a compensation motion must be executed when the C axis is moved so that the electrode centre point remains at the same position.

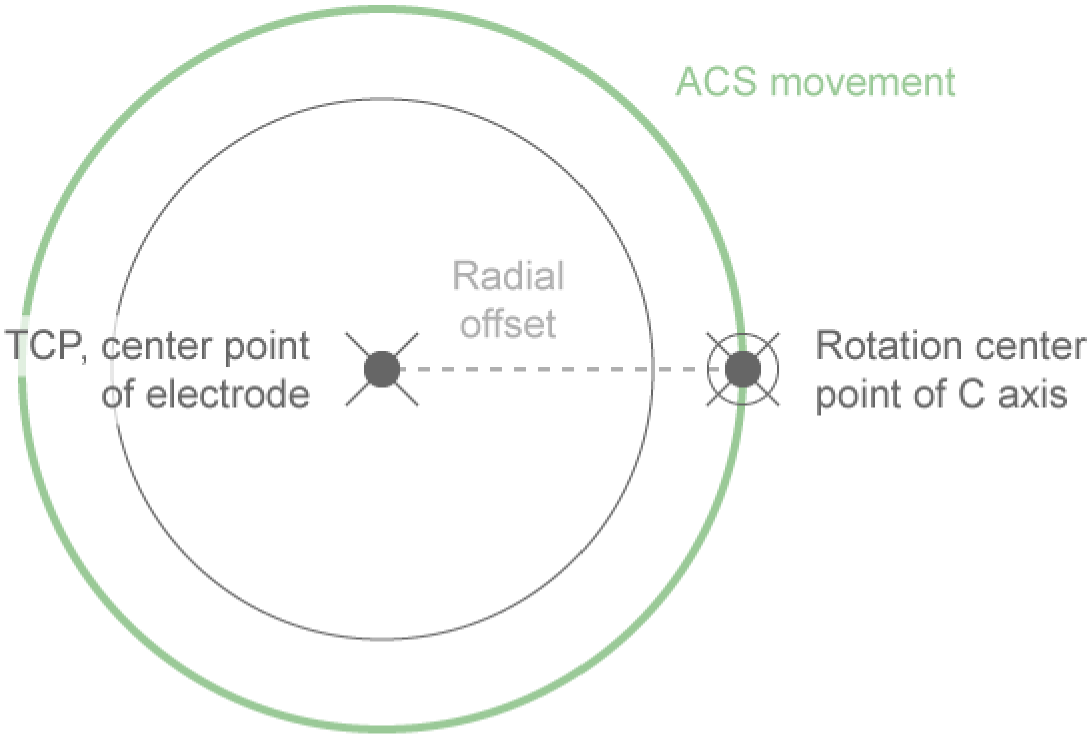


Fig. 56: C axis motion with transformation

Down channel

```
%Test_Kinematic
N0550 #KIN ID[503]
N0560 V.G.KIN[503].PARAM[4] = 100000 ;radial offset
N0580 #TRAFO ON

N0590 G00 Y-7.07 X0 Z17.07 C10 ;positioning
; ----------------------------------------------
N0610 #CS ON [APP] [0, 0, 0, 0, 0, 0] ;define CS for erosion on the path
N0620 L DS-ActivateEscape.sub
N0630 LL EDMOn
N0640 G01 Z7.07 C20 F30 ;path geometry
N0645 G01 Y0 Z0 C0

N0650 "Escape_WaitFini_DisableStart"
N0660 #CS ON [PCS] [0, 0, 0, 0, 0, 0] ;define CS for orbiting
N0670 L DS-ActivateOrbit.sub
…
M30
```

Escape channel

```
%Test_Kinematic_Escape
N040 X[SET_POSITION POS=@PL1] Y[SET_POSITION POS=@PL2] Z[SET_POSITION POS=@PL3]\
       A[SET_POSITION POS=@PL10] B[SET_POSITION POS=@PL11] C[SET_POSITION POS=@PL12]

N050 #SET SLOPE PROFIL [0]
N070 #KIN ID[503]
N080 V.G.KIN[503].PARAM[4] = 100000 ;radial offset
N090 #TRAFO ON

N095 #ESCAPE PATH DEF BEGIN
N100 #CS ON [APP] [@PL4,@PL5,@PL6,@PL7,@PL8,@PL9]
N120 #ESCAPE PATH BACKWARD STOP
N130 G01 Z7.07 C20 F30 ;path geometry
N140 G01 Y0 Z0 C0
N150 #CS DEL ALL
N170 #ESCAPE PATH DEF END

N180 #TRACK CHAN ON [ESCAPE="CH-Escape" START_POS="CH-DownShape"\
                EXTEND_PARAM="CH-Orbit"] ;activate escape channel

N190 #CHANNEL INTERFACE ON [DYN_CS]
N200 #CHANNEL SET [EXT_FEEDRATE_RESOLUTION=nm/s EXT_FEEDRATE_WAIT=1]

N210 #SIGNAL SYN [ID="S-EscapeReady" CH="CH-DownShape"] ;Signal
                                     ;syn: Escape channel ready
N220 M30
```

3.8 Premature change of generator settings

In die sinking, the machining time can be determined by the process using specific erosion parameters. In this case, the NC program may not contain the required geometry before machining. It is therefore possible to abort an ongoing machining operation prematurely and start the next machining operation with new parameters.

If the point of the parameter change is in the geometry of the ‘’Erosion on the path‘’ phase, an M01 can be programmed at this point. This allows the command to be specified for a flushing motion of sufficient length. If the optional stop in the escape channel is activated by the relevant control unit, the system stops at this point.

A second option is to move to a selected generator change point by commanding a flushing motion where the required distance and floating distance are identical. A flushing motion can then be commanded to end immediately when the selected distance is reached.

As soon as the change point is reached via the escape channel, the remaining geometry can be deleted using the Delete distance to go control unit. The escape channel can then be initialised with the new generator parameters and machining can be continued in the down channel.

Changing the generator settings

An example of how to change generator settings is shown below. During the first geometry after N0100, the PLC commands a ‘Delete distance to go". This causes a jump to N0410 and inbetween, the escape channel is decoupled and recoupled with a different strategy. N0220 is taken into account since the current position (Y0 Z-40) fulfils the loop condition.

```
%L DownShape1
N0100 G01 Y0 Z-40 ; generator change is executed after this block
N0110 G01 Y0 Z-80
N0120 G01 Y10 Z-120
N0140 G01 Y10 Z-150
N0150 M17

%L DownShape2
N0210 $IF [V.A.ACT_POS.Y == 0.0] AND [-80.0 < V.A.ACT_POS.Z]
N0220 G01 Y0 Z-120
N0230 $ENDIF
N0240 G01 Y30 Z-150
N0250 M17


%Change Generator
;---- Main program
N0370 LL DownShape1 ;---- first geometry

N0380 L DS-DeactivateEscape.sub
N0390 V.P.EscapeMode = 1 ; variable for escape strategy
N0400 L DS-ActivateEscape-ACS.sub
N0410 #DEL DIST2GO

N0420 LL DownShape2 ;---- second geometry
N0430 M30
```

3.9 Rotary axes with erosion on the path

The program example below for a screw electrode shows how a rotary axis can be used in the geometry for erosion on the path. The #FGROUP command is used to add the C axis to the feed group.

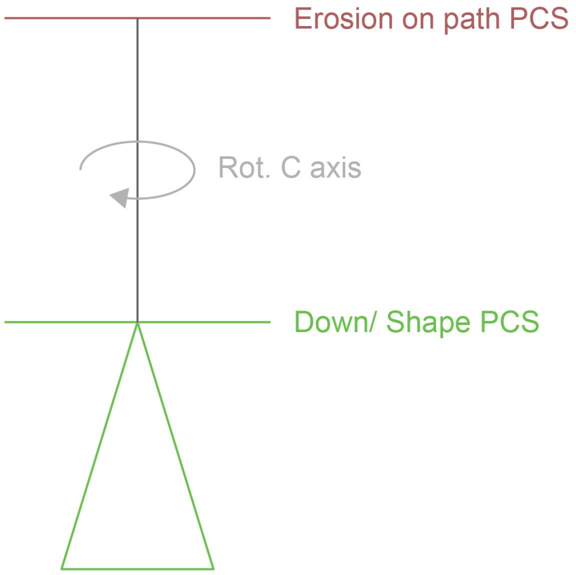


Fig. 57: Using a screw electrode

Down channel - rotary axes with erosion

```
%L Geometrie
N0080 G01 G90 X2 Y2 Z-193 C10 F200
N0090 G01 A10 B20 C20
N0100 G01 G90 X0 Y0 Z-196 A0 B0 C10 F200
N0110 M17
…
N0650 A[MODULO=OFF] B[MODULO=OFF] C[MODULO=OFF]
N0660 #FGROUP [X, Y, Z, C]
; --------------------------------------------
N0670 #CS ON [APP] [0, 0, 0, 0, 0, 0] ; select CS
N0675 L DS-Save-Appr-ACS-Pos.sub
N0680 L DS-ActivateEscape-ACS.sub
N0690 LL EDMOn

N0710 LL Geometry
```

Escape channel - rotary axes with erosion

```
N3030 #SET SLOPE PROFIL [0]
N3035 A[MODULO=OFF] B[MODULO=OFF] C[MODULO=OFF]
N3036 #FGROUP [X, Y, Z, C]

N3060 #ESCAPE PATH DEF BEGIN ; erosion on the path
N3070 #CS ON [APP] [@PL4, @PL5, @PL6, @PL7, @PL8, @PL9]
N0375 #ESCAPE PATH BACKWARD STOP
N0080 G01 G90 X2 Y2 Z-193 C10 F200
N0090 G01 A10 B20 C20
N0100 G01 G90 X0 Y0 Z-196 A0 B0 C10 F200
N3100 #CS DEL ALL
N3110 #ESCAPE PATH DEF END
```

3.10 Coordinate systems

3.10.1 Coordinate systems for erosion on the path

If a coordinate system is selected for erosion on the path, the identical coordinate system must be programmed in the down and escape channels.

The coordinate system must be selected in the down channel before the escape channel is activated. Activation of the escape channel implicitly indicates the start of the erosion phase [Escape channel [▶ 68]].

The following program example consists of the programs in the sections Standard program structure - down channel [▶ 62] and the escape channel [▶ 68] and only lists the specific lines for the coordinate system for erosion on the path.

It is not possible to change the coordinate system when erosion on the path is active.

Using a coordinate system for erosion on the path

Down channel

```
%Senkkanal_Beispielprogramm

…

;---- Activate the escape channel -> implicit start of erosion on the path
N050 #CS ON [EAB] [V.P.Appr_Start_TRANS_X, V.P.Appr_Start_TRANS_Y,0, V.P.Appr_Start_ROT_X,0,0] ;
define CS

…
N5100 #MC_MovePath SYN [ CH="CH-Escape" ID="MC-Escape" FileName="DsEscapeFile" \
@PL1=V.P.Appr_Start_ACS_X @PL2=V.P.Appr_Start_ACS_Y @PL3=V.P.Appr_Start_ACS_Z \
@PL4=V.P.Appr_Start_TRANS_X, @PL5=V.P.Appr_Start_TRANS_Y, @PL6=V.P.Appr_Start_TRANS_Z, \
@PL7=V.P.Appr_Start_ROT_X, @PL8=V.P.Appr_Start_ROT_Y, @PL9=V.P.Appr_Start_ROT_Z \
@PL10=V.P.Appr_Start_ACS_A @PL11=V.P.Appr_Start_ACS_B @PL12=V.P.Appr_Start_ACS_C
@PL13=V.P.EscapeMode \
@PL14=V.P.AlphaAngle @PL15=V.P.PointY @PL16=V.P.PointZ]

…
N070 L ErosionAufBahn.sub
```

Escape channel

```
%Rückzugskanal
…

N3060 #ESCAPE PATH DEF BEGIN
N3070 #CS ON [EAB] [@PL4,@PL5,@PL6,@PL7,@PL8,@PL9]

L ErosionAufBahn.sub
N3100 #CS DEL ALL
N3110 #ESCAPE PATH DEF END
```

3.10.2 Coordinate system for orbiting

A different coordinate system for orbiting can be used for erosion on the path. To ensure that the orbit motion is executed perpendicular to the geometry in the down channel, the coordinate system in the orbit channel must also be rotated. These parameters can be specified when the orbit channel is activated.

The important thing here is that the coordinate system in the orbit channel must remain at the origin and only be rotated.

If several down geometries with different coordinate systems are executed, the orbit channel must first be deactivated and then activated with the new coordinate system

Using a coordinate system for orbiting

Down channel

```
%Senkkanal
…

N110 #CS ON [PCS] [V.A.ACT_POS.X, V.A.ACT_POS.Y, V.A.ACT_POS.Z, 45, 0, 0] ( Define CS for orbiting
N6000 #CHANNEL INTERFACE ON [EXTEND AX=Y FREEZE=X FREEZE=Y ]

; transfer the CS rotations of the current CS in parameter @PL1-@PL3
N6001 #MC_MovePath SYN [ CH="CH-Orbit" ID="MC-OrbCircle" FileName=V.E.OrbitFile
@PL1=V.G.SELECTED_CS.ROT.X @PL2=V.G.SELECTED_CS.ROT.Y \ @PL3=V.G.SELECTED_CS.ROT.Z
@PL4=V.E.OrbitMaxScale @PL5=V.E.OrbitMaxEquid]
```

Orbit channel

```
%Planetärkanal

…

(- same orientation as in down channel
N2250 #CS ADD [PCS] [0, 0, 0, @PL1, @PL2, @PL3]
N2260 #CS SELECT [PCS]
```

3.11 Different gap distances in radial and axial directions

If the material for rounded corners (e.g. quadrants) is removed with a cylindrical erosion tool, the problem usually occurs that different gap distances are set in the spark generator in the radial and axial tool directions. Therefore, the TCP of the erosion tool cannot be moved along a circular path.

In order to erode rounded corners, the point where the tool contacts the workpiece must be included when calculating the TCP path. The resulting path usually approximates an ellipse.

A reduced B-spline type is used to approximate the ellipse. This spline type is defined using the command #SPLINE TYPE REDBSPLINE.

This type allows a maximum of four programmed points in the NC program between #SPLINE ON and OFF, but this is sufficient to approximate the ellipse if the points are selected appropriately.

Using the reduced B-spline

```
%L DownShape
N0240 #SPLINE TYPE REDBSPLINE
N0250 #SPLINE ON
N0260 Y0.018078 Z-0.0127
N0270 Y0.057658 Z-0.009547
N0280 Y0.0767 Z-0.0029934
N0290 Y0.0767 Z0
N0300 #SPLINE OFF
N0310 #SPLINE ON
(---- 2nd B-spline half ellipse ----)
N0320 Y0.0767 Z-0.0029934
N0330 Y0.057658 Z-0.009547
N0340 Y0.018078 Z-0.0127
N0350 Y0 Z-0.0127
N0360 #SPLINE OFF
N0370 M17
```

3.12 Limitations, errors and checks

The CNC performs a number of checks to prevent operational errors. An extract of the checks relevant for die sinking is listed below:

- When the escape channel is activated, the radius must be zero [ID 50916].
- The radius is unequal to 0 when the orbit channel approaches the geometry [ID 50922].
- Escape/restart position not found [ID 50926].
- At the end of the path in the escape channel, the X/Y position must be zero [ID 50931 and ID 50932].
- The escape channel must be activated before the orbit channel is activated in order to mark the geometry for erosion on the path [ID 50933].
- The geometry for erosion on the path must be identical in the down and escape channels [ID 50950, ID 50934, ID 50951].
- The geometry for erosion on the path must be defined in the escape channel before the NC command #TRACK CHAN ON [Escape] [ID 50949].
- No single axis may be moved after erosion on the path [ID 50839].
- A CS may not be active in the escape channel, neither at the start nor at the end of the geometry for erosion on the path [ID 50945 and ID 50946].
- It is not possible to change the CS within the geometry for erosion on the path. It is only possible to change the CS after the start or end of the geometry [ID 50947].
- The position in the down channel changed during an escape motion during erosion on the path [ID 50956].
- A change of CS in the down channel is only possible when the orbit channel is deactivated. If a change of CS is necessary in the down channel, the orbit channel must be deactivated, the CS change must be executed in the down channel and finally the orbit channel is reactivated with the correct CS [ID 50984].
- The specified start position for erosion of the path of the down channel is not within the geometry defined in the escape channel [ID 50995].
- When the points for the point strategy are defined, the specified Y coordinate may not be less than zero and the specified Z coordinate may not be above the end of the geometry for erosion on the path [ID 51006 and ID 51007].
- Before #TRACK CHAN of the orbit channel is deactivated, the channel interface must first be deactivated [ID 51013].
- Before #TRACK CHAN of the escape channel is deactivated, the channel interface must first be deactivated [ID 51014].
- Radius is not 0 before a complete orbit geometry is decoded [ID 51019].
- The coupled master channel moves during channel initialisation of the slave channel [ID 51021].
- The escape channel shifted in the geometry for erosion on the path due to the sliding motion of the down channel [ID 51024].
- When #TRACK CHAN ON [ESCAPE] is activated, no commanded escape strategy of the master channel was found [ID 51033].
- During the escape strategy of the angle bisector, the geometry in the down channel exceeded 180 degrees [ID 51031].

3.13 Special die sinking commands

3.13.1 Opening channel interface to write

#CHANNEL INTERFACE ON [DYN_CS]
#CHANNEL INTERFACE OFF [DYN_CS]

DYN_CS When the “DYN_CS” channel interface is activated, the current channel cyclically writes the axis positions of the first 6 axes at the MCS plane to the interface (e.g. 3 translatory and 3 rotary axis positions).

"TRACK_CS" continues to be supported instead of "DYN_CS" for reasons of downward compatibility.

Programming:

```
#CHANNEL INTERFACE ON [DYN_CS]
;..
#CHANNEL INTERFACE OFF [DYN_CS]
```

NOTICE

If the NC command #CHANNEL INTERFACE OFF is programmed without specifying the interface, e.g. [DYN_CS], error ID 20509 is output.

#CHANNEL INTERFACE ON [EXTEND AX=<axis_name> { **FREEZE**=<axis_name> } **]**
#CHANNEL INTERFACE OFF [EXTEND]

EXTEND The current radius is transferred to the coupled slave channel over the “EXTEND” channel interface. (#TRACK CHAN ON [EXTEND] [▸ 97]) This command marks the end of the ‘’Erosion on the path‘’ phase.

AX Name of the axis used as the radius to calculate the scaling factor or the equidistant value.

$Factor_{Scale}$ = axis $position_{PCS}$ / MAX_SCALE

FREEZE The PCS position of a "frozen" axis no longer changes. As a result, the PCS input coordinate remains constant in the Cartesian transformation in the MCS. The MCS position of this type of axis can be superimposed by the input of another channel. In die sinking, the orbit channel calculates the MCS positions of these axes. The orbit channel uses the current PCS positions, the axis defined by AX of this channel to calculate the MCS positions of the "frozen" axis.

#CHANNEL INTERFACE ON [ESCAPE GEOMETRY= <FLAT | ALPHA | POINT | BISECTOR> [ANGLE=..] [POINT_Y=..] [POINT_Z=..] [BISEC_D1=..] [BISEC_D2=..]]
#CHANNEL INTERFACE OFF [ESCAPE [**WAIT**] **]**

ESCAPE The channel writes its current PCS positions to the interface by activating the "ESCAPE" channel interface. These positions can be used by a coupled slave channel to calculate the escape geometry (see #TRACK CHAN ON [ESCAPE ...] [▸ 97]).

GEOMETRY=.. This keyword and the value that follows can be used to select the escape strategy set in the slave channel. No check is made whether a setting can be made at this time.

Compared to #CHANNEL INTERFACE SET [ESCAPE GEOMETRY ...], [▸ 95] the system does not wait until the strategy is set in the slave channel. The selection includes:

- FLAT: Flat strategy
- ALPHA: Alpha strategy
- POINT: Point strategy

- BISECTOR: Angle bisector strategy

ANGLE=.. Angle starting from the perpendicular for the ALPHA escape strategy

POINT_Y=.. Y coordinate (in the PCS) of the point definition for the POINT escape strategy

POINT_Z=.. Z coordinate (in the PCS) of the point definition for the POINT escape strategy

BISEC_D1=.. D1 parameter for the BISECTOR escape strategy

BISEC_D2=.. D2 parameter for the BISECTOR escape strategy

WAIT While the interface is deactivated, the channel waits until the active escape motion is completed. Any further escape command is suppressed until the interface is reactivated

#CHANNEL INTERFACE SET [ESCAPE GEOMETRY= <FLAT | ALPHA | POINT | BISECTOR> [ANGLE=..] [POINT_Y=..] [POINT_Z=..] [BISEC_D1=..] [BISEC_D2=..]]

The NC command can be used to set a value in another channel.
The following values can be set in another channel:

ESCAPE GEOMETRY=.. Sets the geometry of the escape strategy of the slave channel. A check is made whether a change is possible.
A strategy change is only possible if

- the radius in the down channel is zero,
- erosion is taking place on the path,
- the escape channel is not active
- or the geometries at the current point have the identical path.

The system waits until a change is possible before continuing. The corresponding escape channel indicates its change request via the status signal bahn_state.escape_strategy_wait_change_r [▸ 106] an.

- FLAT: Flat strategy
- ALPHA: Alpha strategy
- POINT: Point strategy
- BISECTOR: Angle bisector strategy

ANGLE=.. Angle starting from the perpendicular for the ALPHA escape strategy

POINT_Y=.. Y coordinate (in the PCS) of the point definition for the POINT escape strategy

POINT_Z=.. Z coordinate (in the PCS) of the point definition for the POINT escape strategy

BISEC_D1=.. D1 parameter for the BISECTOR escape strategy

BISEC_D2=.. D2 parameter for the BISECTOR escape strategy

WAIT While the interface is deactivated, the channel waits until the active motion of the escape motion is completed. Any further escape command is suppressed until the interface is reactivated.

3.13.2 Opening channel interface to read

#TRACK CHAN ON [DYN_CS=.. [SUPERIMPOSE] [FILTER=..] [WAIT]]

DYN_CS=..	CNC channel number as the source of the dynamic coordinate system to be followed on the PCS plane. The supplied coordinates are superimposed on the MCS or PCS plane (see SUPERIMPOSE) in the same way as for an axis offset. Range of the channel number [1; 12]
SUPERIMPOSE	Superimposes the dynamic coordinate system on the PCS plane. If superimposition is not used, the values are read in the machine coordinate system (MCS).
FILTER=..	Specifies the number of cycles across which a possible position jump is smoothed when coupling or decoupling. Value = 0 : Filter is off Value > 0 : Filter is enabled with filter cycles specified explicitly. If FILTER is not specified, the filter is enabled with a default value of 200.
WAIT	When the filter is active, the program waits until the coupling is completely activated before executing the next NC line. If this mode is not specified (default), coupling is executed on the fly.

#TRACK CHAN ON [EXTEND=.. MAX_SCALE=.. MAX_EQUID=..]

EXTEND=..	CNC channel number of the master channel used to read the radius. The current channel uses the radius to calculate the scaling or equidistant geometry. Range of the channel number [1; 12]
MAX_SCALE=..	Maximum scaling radius, normalising value to calculate the current scaling factor. Scaling is disabled at MAX_SCALE=0 (default).
MAX_EQUID=..	Maximum equidistant radius to calculate the extend function. Extend equidistant is disabled at MAX_EQUID=0 (default).

#TRACK CHAN ON [ESCAPE=.. START_POS=.. EXTEND_PARAM=..]

ESCAPE=..	CNC channel number of escape channel. Range of the channel number [1; 12]
START_POS=..	Specified CNC channel number. This channel (normally the down channel) uses the start position. The channel interface of the specified channel must first be activated by #CHANNEL INTERFACE ON [ESCAPE] [▸ 95] Without a definition of START_POS the start position of the ESCAPE channel is used.
EXTEND_PARAM =..	Specified CNC channel number. The escape calculation is executed depending on the extend parameters (e.g. MAX_SCALE). Therefore, this requires a reference to the EXTEND channel [see #CHANNEL INTERFACE ON [EXTEND] [▸ 95]].

#TRACK CHAN OFF [[DYN_CS | EXTEND | ESCAPE] [SUPERIMPOSE] NO_POS_SYNC]]

DYN_CS	Deactivates the DYN_CS channel interface of this channel.
EXTEND	Deactivates the EXTEND channel interface of this channel.
ESCAPE	Deactivates the ESCAPE channel interface of this channel.
SUPERIMPOSE	Deactivates the PCS superimposition of the dynamic coordinate system.
NO_POS_SYNC	Suppresses position synchronisation of the channel when the channel interface is deactivated. When the channel interface is deactivated, the position of the channel is synchronised by default (see #CHANNEL INIT [CMDPOS]). This option can be used in the command to suppress this synchronisation, for example for successive rotations while the interface is active. When the channel interface is disabled, the position of this channel must be synchronised (see #CHANNEL INIT [CMDPOS]). This extension can be selected in the NC command to suppress this synchronisation, for example for successive rotations while the interface is active.

If neither DYN_CS, EXTEND nor ESCAPE are specified in the NC command #TRACK CHAN OFF, all channel interfaces are disabled.

3.13.3 Setting function-specific parameters in the channel

3.13.3.1 Synchronisation with an external velocity interface

This command determines

- the interpretation of read velocity values by the “External commanded path velocity“ control unit.
- defines the “wait response” of the interpolation with an external commanded velocity specified by this control unit.

#CHANNEL SET [EXT_FEEDRATE_RESOLUTION=.. EXT_FEEDRATE_WAIT=..]

EXT_FEEDRATE_RESOLUTION=..	Determines the velocity resolution of the external commanded velocity on the HLI. Permitted values are um/s and nm/s, corresponding to the units [µm/s] and [nm/s].
EXT_FEEDRATE_WAIT=..	This options determines the “wait response” between interpolation and the “External commanded velocity” control unit. 0: Interpolation does not wait for activation of the external commanded velocity. 1: Interpolation is stopped and waits for the next motion until the external commanded velocity is activated on the HLI and the commanded feed rate is >0 (default).

#CHANNEL SET with external commanded velocity

```
;..
#CHANNEL SET [ EXT_FEEDRATE_RESOLUTION=nm/s EXT_FEEDRATE_WAIT=1]
; or
#CHANNEL SET [ EXT_FEEDRATE_RESOLUTION=um/s EXT_FEEDRATE_WAIT=0]
```

3.13.3.2 Fast positioning of the orbit channel

This command is special for the die sinking function.

#CHANNEL SET [FAST_FORWARD_IN_CENTER= ON/ OFF **]**

The command permits the fast positioning of the orbit channel at radius zero supplied by the down channel.

The orbit channel positions in a cycle up to the end position of the next motion stop point. The orbit channel can then skip motions in a cycle without exceeding an acceleration.

If the radius is unequal to 0, the orbit channel interpolates the geometry at the applied feed rate.

3.13.4 Defining the escape geometry in the escape channel

This command marks the start of the geometry description for erosion on the path in the escape channel.

#ESCAPE PATH DEF BEGIN

This command marks that it is not permitted to retract further than this point in the escape channel.

#ESCAPE PATH BACKWARD STOP

This command marks that no further motion blocks will follow in the geometry sequence in the escape channel. There may still be control information.

#ESCAPE PATH POST SEQUENCE

This command marks the end of the geometry description for erosion on the path in the escape channel.

#ESCAPE PATH DEF END

3.14 PLCopen group programming

3.14.1 #MC_MovePath command

See also [MCP-P4// MC_MovePath]

A CNC program running in the commanding channel starts a job in another channel “agent” using the #MC_MovePath command.

> **Available as of V3.1.3110.**

```
Syntax of the NC command:
```

#MC_MovePath [SYN] **[CH=.. JobID=.. FileName=.. @PL<1…20>=..**
[InitializeOnActualPosition=..]
[SetDefaultConfig=..]
[ReportSceneSampling=..]
[ReportRunTimeMeasure=..]
[ReportAxesPositionSample=.]. **]**

SYN	Synchronous command execution of the ISO program. Before the command is executed, an implicit channel synchronisation takes place (implicit #FLUSH WAIT).
CH=..	Logical channel number of the channel in which the job is executed. The link between the CNC channel number and the CNC channels is specified in the start-up list. (See Parameter descriptions [▶ 118]).
JobID=..	User-specific job identification number (job ID). Every number must be unique in the commanding program of the master channel. For example, this ID is used in the #WAIT MC_Status command for job identification.
FileName=<Name>	Filename of the ISO program which is to be started.
@P<i>=..	Parameter transfer to the commanded ISO program. These parameters can be accessed in the called “main program” by @PL<i>.
InitializeOnActualPosition=..	Requests current positions at program start. See Channel operation mode- SUPPRESS_POSITION_REQUEST If InitializeOnActualPosition is not used, the current configuration specified applies. ON: Request the position regardless of the current configuration. OFF: “No position request” regardless of the current configuration. USE_ACTUAL: The current configuration specified applies
SetDefaultConfig =..	Initialises decoder working data at program start. See Channel operation mode- SUPPRESS_PROG_START_INIT If SetDefaultConfig is not used, the current configuration specified applies. ON: Initialisedecoder working data. OFF: Deselect initialisation of decoder working data. USE_ACTUAL: The current configuration specified applies

ReportSceneSample=..	Enables the interface to log scene data. See Channel operation mode- BEARB_MODE_SCENE If ReportSceneSample is not used, the current configuration specified applies.	
	ON	Log scene data.
	OFF	Disable scene data logging.
	USE_ACTUAL	The current configuration specified applies
ReportRunTimeMeasure=..	Enables the interface to log time stamps. See Channel operation mode - ONLINE_PROD_TIME If ReportRunTimeMeasure is not used, the current configuration specified applies.	
	ON	Generate time stamp data.
	OFF	Disable logging of time stamp data.
	USE_ACTUAL	The current configuration specified applies
ReportAxesPositionSample=..	Enables the interface to log axis positions. See Channel operation mode- ON_LINE If ReportAxesPositionSample is not used, the current configuration specified applies.	
	ON	Generate log data of axis positions.
	OFF	Disable logging of axis positions.
	USE_ACTUAL	The current configuration specified applies.

After reading in #MC_MovePath, the interpretation of the CNC program is not continued until the job commanded by the command is successfully stored in the job queue of the channel agent.

Starting a program in another channel

The executed program starts the SlaveFile.nc program with the 3 listed parameters in channel 3. These parameters can be used in the SlaveFile.nc program (e.g. by G01 X@PL1).

```
%ExampleMC_MovePath
N010 "CH-Slave"= "3"
N020 "MC-Slave"= "633"

N030 #MC_MovePath SYN[ CH="CH-Slave" ID="MC-Slave" \
    FileName=”SlaveFile.nc” \
    @PL1=V.A.ACS.ABS.X @PL2=V.A.ACS.ABS.Y @PL3=V.A.ACS.ABS.Z ]

N040 M30
```

3.14.2 #MCV_GroupResetForced command

See also [MCP-P4 MCV_GrpResetForced]

The MCV_GrpResetForced command forces a channel reset in another channel.

Available as of V3.1.3110.

```
Syntax:
```

#MCV_GrpResetForced [SYN] **[CH=.. JobID=..]**

SYN	Synchronous command execution of the ISO program. Before the command is executed, an implicit channel synchronisation takes place (implicit #FLUSH WAIT). Without SYN the command is executed without channel synchronisation.
CH=..	Logical channel number P-STUP-00208 [▸ 118] of the channel in which the job is executed.
JobID=..	User-specific job identification number(JobID). Every number must be unique in the current commanding program of the master channel. For example, this JobID is used in the #WAIT MC_Status command for job identification.

Executes a reset in another channel

The executed program resets channel 3.

```
%ExampleMCV_GrpResetForced
N010 N030 #MCV_GrpResetForced SYN [ CH=3 ID=634]
N020 M30
```

3.14.3 #WAIT MC_Status command

The #WAIT MC_Status command makes the system wait for a job acknowledgement. Several events can be specified for job acknowledgements. The command return value can be used to continue program execution.

Available as of V3.1.3110.

```
Syntax of the NC command:
```

#WAIT MC_Status [JobID=.. [MC_NEW][MC_BUSY][MC_ACTIVE][MC_DONE]
[MC_ABORTED][MC_ERROR] **]**

JobID=..	User-specific JobID/job identification number. Every number must be unique in the commanding program.
MC_NEW	New job received but not yet processed in the commanded channel.
MC_BUSY	New job waits for execution in the commanded channel.
MC_ACTIVE	The job is active in the commanded channel but was not completed.
MC_DONE	The job was successfully completed in the commanded channel.
MC_ABORTED	The job was aborted in the commanded channel before completion. An abort can be forced by a reset command, e.g. #MC_GroupResetForced.
MC_ERROR	An error occurred while the job is in progress.

Properties:

- Job IDs are stored in a history when #WAIT MC_Status is called in order to return a correct status in a “later” request even after the job is completed/cancelled. If the job ID specified in JobID is unknown because it was never commanded or is no longer in the history, an error is output.
- The job ID history is deleted at program end. It is therefore no longer possible to synchronise with jobs from previous (client) jobs. It is also not possible to synchronise with jobs from another client.

If no expected status is defined for #WAIT MC_Status, an implicit MC_DONE and MC_ABORTED apply.

Waiting for job state

The program starts a “SlaveFile”.nc program in logical channel 3 with job 633, then waits for the job to be completed with the alternative states MC_DONE, MC_ABORTED or MC_ERROR. If none of the states is reached, the program stops.

```
%Example MC_Wait
N010 #MC_MovePath SYN[ CH=3 JobID=633 FileName=”SlaveFile.nc”]
N020 #WAIT MC_Status [JobID=633 MC_DONE MC_ABORTED MC_ERROR]
N100 M30
```

3.14.4 #MCV_WAIT_STATUS command

After one or more previous #WAIT MC_Status commands, the last valid return value of #WAIT MC_Status can be read out by the MCV_WAIT_STATUS function and used to continue program execution.
The numerical values stored in the constants apply as return values.

The status constants MC_NEW, MC_BUSY, MC_ACTIVE, MC_DONE, MC_ERROR and MC_ABORTED are available in the CNC code. They are used to compare a stored variable after a #MCV_WAIT_STATUS.

```
Syntax:
```

#MCV_WAIT_STATUS [JobID]

<JobID>	User-specific JobID/job identification number. The number must be unique in the commanding program.

Waiting for a job signal

The program starts a “SlaveFile”.nc program in logical channel 3 with job 633, then waits for the job to be completed with the alternative states MC_DONE, MC_ABORTED or MC_ERROR.The valid state when #WAIT MC_Status is enabled can be analysed.

```
%Example MC_Wait
N010 #MC_MovePath SYN[ CH=3 JobID=633 FileName=”SlaveFile.nc”]
N010 #WAIT MC_Status [JobID=633 MC_DONE MC_ABORTED MC_ERROR]
N020 V.P.McStatus = MCV_WAIT_STATUS [633]
N030 $IF V.P.McStatus != MC_DONE
N040   #ERROR [ID455 MID0 RC2 PV1=V.P.McStatus \
              PV2=MC_DONE PM1=3 PM2=633]
N050 $ENDIF
N100 M30
```

4 PLC interface

4.1 Position display

Command position (PCS)	
Description	Position preset in the current cycle as setpoint.
Signal flow	CNC → PLC
ST path	gpCh[*channel_idx*]^.bahn_state.coord_r[*axis_idx*].**act_position_r**
Data type	DINT
Unit	0.1 µm
Access	PLC is reading

PCS position after a dynamic CS superimposition	
Description	Interpolated position in the selected PCS, including the superimposition of the dynamic coordinate system on the channel interface.
Signal flow	CNC □ PLC
ST path	gpCh[*channel_idx*]^.bahn_state.coord_r[*axis_idx*].**active_pos_pcs_dyn_cs_r**
Data type	DINT
Unit	0.1 µm
Access	PLC is reading
Special feature	Available as of V3.1.3105.1

MCS position after all dynamic CS superimpositions	
Description	Interpolated position in the MCS, including the superimposition of all dynamic coordinate systems on the channel interface.
Signal flow	CNC □ PLC
ST path	gpCh[channel_idx]^.bahn_state.coord_r[axis_idx].**active_pos_mcs_dyn_cs_r**
Data type	DINT
Unit	0.1 µm
Access	PLC is reading
Special feature	Available as of V3.1.3105.1

Command position (ACS) from interpolator	
Description	Command position in the axis coordinate system which is refreshed each interpolator cycle.
Signal flow	CNC → PLC
ST path	gpCh[*channel_idx*]^.bahn_state.coord_r[*axis_idx*].**acs_position_r**
Unit	0.1 µm
Data type	DINT
Access	PLC is reading

4.2 Die sinking status signals

Escape channel is activated	
Description	Escape channel was activated by #TRACK CHAN ON [▸ 97] [ESCAPE] and can be started by a trigger signal. The trigger signal for escape start can be provided by backward motion at negative external velocity or by commanding a flushing process.
Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.bahn_state.**escape_enabled_r**

Data type	BOOL
Value range	[TRUE : ESCAPE active, FALSE]
Access	PLC is reading

Trigger consideration suspended for escape motions	
Description	Consideration of the trigger for escape motions can be temporarily deactivated by the down channel. This status signal indicates whether an escape motion can be started. • Suspend trigger consideration with NC command: #CHANNEL INTERFACE OFF[ESCAPE WAIT] [▸ 95] • Activate trigger consideration with NC command: #CHANNEL INTERFACE ON[ESCAPE] [▸ 95]
Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.bahn_state.**escape_trigger_is_suspended_r**
REALDatentyp	BOOL
Value range	[TRUE = Trigger consideration suspended FALSE]
Access	PLC is reading

Length of the escape path	
Description	This data item indicates the actual length of the escape path. The length of the geometry for "Erosion on the path" is not included. The dynamically generated path has an individual length dependent on the starting point of the escape, the escape strategy (FLAT, ALPHA, etc.) and the escape end. The length can be read by this data item. The length is independent of the distance traversed in the escape channel. This value is updated cyclically from the point when the orbit channel is activated (i.e. when the "Erosion on the path" phase is implicitly terminated).
Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.bahn_state.**escape_path_length_r**
Data type	LREAL
Value range	[0 <= length]
Access	PLC is reading

Current escape strategy	
Description	This data item shows the currently selected escape strategy.
Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.**bahn_state.escape_strategy_r**
Data type	UINT
Value range	Possible values are: • 0- EDM_ESC_GEOM_NONE • 1- EDM_ESC_GEOM_FLAT • 2- EDM_ESC_GEOM_ALPHA • 3- EDM_ESC_GEOM_POINT • 4- EDM_ESC_GEOM_BISECTOR
Access	PLC is reading

Change request for escape strategy active	
Description	This data item shows that a change request was identified for the escape strategy, If it is impossible to change the strategy, the data item remains TRUE until the escape strategy is changed.

Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.bahn_state.**escape_strategy_wait_change_r**
Data type	BOOL
Value range	[TRUE / FALSE]
Access	PLC is reading

Flushing motion is active	
Description	The escape channel executes a flushing motion of the PLC. When the floating point is reached, the signal reverts to FALSE.
Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.bahn_state.**jump_acitve_r**
Data type	BOOL
Value range	[TRUE : Flushing motion active, FALSE]
Access	PLC is reading

Erosion on the path is active	
Description	This data item only indicates in the escape channel whether the escape channel is in the "Erosion on the path" phase or has entered this phase as a result of the escape. In the latter case, the signal is set to TRUE.
Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.bahn_state.**approach_active_r**
Data type	BOOL
Value range	[TRUE : “Erosion on the path” phase active, FALSE]
Access	PLC is reading

Orbit channel active	
Description	The element indicates whether the geometry in the orbit channel is fully decoded. If the value is set to TRUE, the down channel can increase the radius from 0.
Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.bahn_state.**orbit_active_r**
Data type	BOOL
Value range	[TRUE : Orbit channel active, FALSE]
Access	PLC is reading

Radius in the orbit channel is 0	
Description	This element indicates whether the commanded radius in the orbit channel is 0. If the value is 0, the interface to the down channel can be deactivated, for example.
Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.bahn_state.orbit_radius_zero_r
Data type	BOOL
Value range	[TRUE : Radius in the orbit channel is 0, FALSE]
Access	PLC is reading

Wait for equidistant calculation in the block	
Description	The element indicates that the equidistant calculation has not yet finished in the current motion block.
Signal flow	CNC → PLC
ST path	gpCh[channel_idx]^.bahn_state.orbit_wait_extend_ncbl_r

Data type	BOOL
Value range	[TRUE : Wait state, equidistant for this motion block not yet calculated. FALSE]
Access	PLC is reading

4.2.1 Other status signals

Real-time loop active	
Description	The datum indicates whether current machining is located in a #RT WHILE - #RT ENDWHILE sequence.
Signal flow	CNC → PLC
ST path	gpCh[*channel_idx*]^.bahn_state.**inside_rt_loop_r**
Data type	BOOL
Value range	[TRUE = Machining within #RT WHILE #RT ENDWHILE, FALSE]
Access	PLC is reading
Special feature	Date available as of CNC Build V3.1.3105.01 and higher

Waiting for external command velocity	
Description	The NC channel waits for an external command velocity.
Signal flow	CNC → PLC
ST path	gpCh[*channel_idx*]^.bahn_state.**wait_ext_command_speed_r**
Data type	BOOL
Value range	[TRUE = channel is waiting FALSE = external velocity exists]
Access	PLC is reading

End of program reached	
Description	This status information indicates that the **interpolator** has reached program end or that no NC program is currently under in execution.
Signal flow	CNC → PLC
ST path	gpCh[*channel_idx*]^.bahn_state.**program_end_r**
Data type	BOOL
Value range	[TRUE = End of program reached, FALSE]
Access	PLC is reading

Currently covered path in the NC program (PCS)	
Description	Reads the current distance covered in the NC program since program start or since the last # DISTANCE PROG START CLEAR NC command. The calculation is based on the current position in the current NC block.
Signal flow	CNC → PLC
ST path	gpCh[*channel_idx*]^.bahn_state.**dist_prog_start**
Data type	UDINT (* LREAL)
Unit	0.1 µm
Access	PLC is reading
Special features	* As of CNC Build V3.1.3104.01 the data element is provided in LREAL forma.

Activation of external path velocity	
Description	Activate the velocity commanded in the ext_command_speed control unit. To reach the commanded velocity, all axes involved in the motion are accelerated or decelerated.

	If this value is TRUE, the sign is considered in the current path feed (active_feed_r control unit).
Data type	MC_CONTROL_BOOL_UNIT, see description Control unit
Access	PLC reads request_r + state_r and writes command_w + enable_w
ST Path	gpCh[*channel_idx*]^.bahn_mc_control.**ext_command_speed_valid**
Commanded, requested and return values	
ST Element	**.command_w** **.request_r** **.state_r**
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST Element	**.enable_w**

4.3 Escape motion (flushing process)

4.3.1 Escape motion control unit (flushing process)

Jump	
Description	The PLC can used this control unit to command an escape motion of the electrode during the runtime in the escape channel in a die sinking process. The control unit must then be enabled by enable_w = TRUE.
Data type	MC_CONTROL_JUMP_UNIT
ST path	gpCh[*channel_idx*]^. bahn_mc_control.**jumping**
Command data	
ST element	**.command_w**
Data type	HLI_JUMPING_COMMAND
Access	PLC is writing
Return values	
ST element	**.state_r**
Data type	HLI_JUMPING_STATE
Access	PLC is reading
Activation	
ST element	**.enable_w**
Data type	BOOL
Access	PLC is writing
Value range	[TRUE/FALSE] ; TRUE: Control unit in PLC activated
Flow control of commanded value	
ST element	**.command_semaphor_rw**
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	**Consumption data item**
Access	PLC sets the **command_semaphor_rw** to TRUE for a new command_w. The CNC sets the **command_semaphor_rw** to FALSE after reading the command_w data.

4.3.2 User data

Unassigned dynamic values are assigned the maximum possible value.

If subsequent dynamic values are not assigned or are assigned values less than or equal to zero, this is ignored and the maximum possible value is used instead.

- With velocity values, the maximum possible velocity is used as for a G0 motion.
- With acceleration values, the maximum acceleration value is used for the axis that has the lowest dynamics during motion on the specified path.
- With jerk values, the jerk that has a ramp time of 1 cycle is used.

Commanded data – flushing motion	
Description	Control data for flushing motion
ST path	gpCh[*channel_idx*]^. bahn_mc_control.jumping.**command_w**
ST name	HLI_JUMPING_COMMAND
ST element	**.inversion_point.distance**
Data type	DINT
Description/ special features	The escape motion is executed up to the specified distance on the escape path. Unit [0.1 µm] The following special cases can be parameterised: • **inversion_point.distance == 0**: inversion_point.height used. • **inversion_point.distance == HLI_JUMP_ABORT (= -1)**: Abort the current flushing motion; deceleration dynamics correspond to the dynamics entered in the forward and backward directions. **inversion_point.distance <= HLI_JUMP_ABORT_EMERGENCY (= -2)**: Emergency abort of current flushing motion; deceleration dynamics correspond to the dynamics set in the forward and backward directions.
ST element	**.inversion_point.height**
Data type	DINT
Description/ special features	The escape motion is executed up to the absolute height in the Z axis. If the specified Z height is less than the current height, no flushing motion is executed. Unit [0.1 µm]
ST element	**.float_point.distance**
Data type	DINT
Description/ special features	If float_point.distance > 0, the flushing motion is aborted at the specified distance before the end when re-approaching the workpiece. Unit [0.1 µm]
ST element	**.float_point.velocity**
Data type	UDINT
Description/ special features	Velocity that is to be attained on reaching the float_point. This parameter is not available. Unit [0.1 mm/min]
ST element	**.dyn_forward.velocity**

Data type	UDINT
Description/ special features	Specified velocity for the flushing motion when moving in the forward direction. Unit [1 µm/s]
ST element	**.dyn_forward.acceleration**
Data type	UDINT
Description/ special features	Specified acceleration for the flushing motion when moving in the forward direction. Unit [mm/s²]
ST element	**.dyn_forward.jerk**
Data type	UDINT
Description/ special features	Specified jerk for the flushing motion when moving in the forward direction (up to jerk while reducing the negative acceleration) Unit [mm/s³]
ST element	**.dyn_forward.jerk_final**
Data type	UDINT
Description/ special features	Specified jerk for the flushing motion when moving in the direction of the workpiece to reduce negative acceleration in the final motion. This parameter enables smooth transition from the flushing motion to the rest of the erosion process. Unit [mm/s²]
ST element	**.dyn_backward.jerk**
Data type	UDINT
Description/ special features	Specified jerk for the flushing motion when moving in the backward direction. Unit [mm/s²]
ST element	**.dyn_backward.acceleration**
Data type	UDINT
Description/ special features	Specified acceleration for the flushing motion when moving in the backward direction. Unit [mm/s²]
ST element	**.dyn_backward.velocity**
Data type	UDINT
Description/ special features	Specified velocity for the flushing motion when moving in the backward direction. Unit [1 µm/s]
ST element	**.dyn_backward.jerk_final**
Data type	UDINT
Description/ special features	Specified jerk for the flushing motion when moving in the direction of the workpiece to reduce negative acceleration. Unit [mm/s³]
ST element	**.dyn_emergency.acceleration**
Data type	UDINT
Description/ special features	Specified acceleration for deceleration in the event of an emergency abort of the flushing motion. Unit [mm/s²]
ST element	**.dyn_emergency.jerk**
Data type	UDINT

Description/ special features	Specified jerk for deceleration in the event of an emergency abort of the flushing motion. Unit [mm/s²]

State data – flushing motion	
Description	State data of the flushing motion control unit.
ST path	gpCh[*channel_idx*]^.bahn_mc_control.jumping.**state_r**
ST name	HLI_JUMPING_STATE
ST element	**.float_distance**
Data type	UDINT
Description	Distance between the positions of the Z axis in the event of an abort of the flushing motion and the start/end point of the flashing motion. Unit [0.1 µm]
ST element	**.is_active**
Data type	BOOL
Description	This data item indicates whether a flushing motion is active.

4.4 Insert stop marks

This function is available as of CNC Build V3.1.3105.01.

4.4.1 Control unit – Insert stop marks (insert command)

Insert stop marks	
Description	During runtime, the PLC can use this control unit to set stop points in the future NC program. The control unit must then be enabled by enable_w = TRUE. The PLC signals a new assignment of command_2 by setting **command_semaphor_rw** to TRUE. The CNC sets the command_semaphor_rw to FALSE after the command_w data is read.
Data type	MC_CONTROL_INSERT_CMD_UNIT
ST path	gpCh[*channel_idx*]^. bahn_mc_control.**insert_cmd**
Command data	
ST element	**.command_w**
Data type	HLI_INSERT_CMD_COMMAND [▶ 114]
Access	PLC writes command_w
Return values	
ST element	**.state_r**
Data type	HLI_INSERT_CMD_STATE [▶ 115]
Access	PLC reads
Activation	
ST element	**.enable_w**
Data type	BOOL
Access	PLC writes
Value range	[TRUE/FALSE] ; TRUE: CU enabled in PLC
Flow control of commanded value	
ST element	**.semaphor_rw**
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	**Consumption data item**
Access	TRUE : PLC triggers on new request FALSE : CNC has read new request

4.4.2 User data

Command data – Insert command	
Description	Control data for inserted STOP command
ST path	gpCh[*channel_idx*]^.bahn_mc_control.insert_cmd.**command_w**
ST name	HLI_INSERT_CMD_COMMAND
ST element	**.dist_or_pos**
Data type	LREAL
Description/ special features	Relative/absolute distance or axis position at which the stop is to be inserted. [0.1 µm] The default setting is DISTANCE mode, the POSITION mode is active at a value of “.axis_nr“ > 0

ST element	**.rel_abs_mode**
Data type	UINT
Description/ special features	Value = 0 for relative distance with automatic insertion of a new stop when the current stop is reached Value = 1 for one-off insertion at the absolute distance Value = 2 for one-off insertion at the relative distance
ST element	**.axis_nr**
Data type	UINT
Description/ special features	Value = 0 for DISTANCE mode Value > 0 for POSITION mode: Stop is inserted at the commanded axis position of the axis with the logical axis number = “.axis_nr”.
ST element	**.m_function_nr**
Data type	UINT
Description/ special features	Number of the inserted M function • Value = 0 for M00 programmed stop • Value = 1 for M01 optional stop (enabled/disabled with m01_stop_enable CU) • Value > 1 for M<m_function_nr> with MVS_SVS synchronisation
ST element	**.add_nr**
Data type	DINT
Description/ special features	Optional additional value of the M function; it is specified as a negative or positive integer.

Status data – Insert command	
Description	Status data of the insert command control unit
ST path	gpCh[*channel_idx*]^.bahn_mc_control.insert_cmd.**state_r**
ST name	HLI_INSERT_CMD_STATE
ST element	**.distance_of_next_stop**
Data type	LREAL
Description	Absolute distance (dist_prog_start) of the feed axes (#FGROUP) at the next stop [0.1 µm] Value >= 0 : Stop found, axis positions correct Value = -1 : Stop still not found in the NC program
ST element	**.position_at_next_stop[*idx*]**
Data type	ARRAY[0..HLI_CS_AXIS_MAXIDX] OF DINT
Description	When • command_w.axis_nr = 0, the axis positions X, Y, Z are assigned accordingly at the next stop in the PCS [0.1 µm] • command_w.axis_nr > 0; .position_at_next_stop[0] contains the axis position of the commanded axis at the next stop position_at_next_stop[1]=0 and position_at_next_stop[2]=0
ST element	**.state**
Data type	DINT
Description	Status if the current command_w. • Value =0: no stop commanded

	• Value=1: Stop commanded but not yet reached

5 Parameter

5.1 Overview

Start-up parameter

ID	Parameter	Description
P-STUP-00033	fb_storage_size	Memory size for backward motion
P-STUP-00182	schedule.config	Scheduling the CNC
P-STUP-00208	jobmanager.group[i].cnc_slave[j].log_id	Logical ID of the slave channel
P-STUP-00209	jobmanager.group[i].cnc_slave[j].channel_id	Channel number of the slave channel

Channel parameters

ID	Parameter	Description
P-CHAN-00145	kin_trafo_display	Activation of TCP display data
P-CHAN-00430	no_backward_before_prg_end	Suppress escape motions
P-CHAN-00650	configuration.interpolator.function	Definition of interpolator functionalities (alternatively P-STUP-00070)

5.2 Parameter descriptions

Start-up parameter

P-STUP-00033	Memory size for backward motion
Description	This parameter defines the memory size in bytes used for backward motion on the path. During start-up, the NC checks whether the required minimum size is available. If this is not the case, a warning is output and the memory size is set to the required minimum value. If the size is set to 0, the “forward/ backward motion on the path” function is not available. The maximum size is only limited by the resources available on the PC.
Parameter	fb_storage_size[i] where i = 0 to 11 (maximum number of channels: 12, application-specific)
Data type	UNS32
Data range	0 ... MAX(UNS32)
Dimension	----
Default value	0
Remarks	

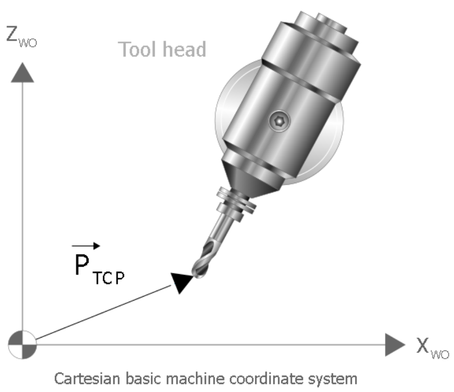
P-STUP-00182	Scheduling the NC channel
Description	This parameter defines channel scheduling for the CNC. For the die-sinking function, this value need only be parameterised in the down channel using DIE_SINKING.
Parameter	schedule_config
Data type	UNS32
Data range	DEFAULT default scheduling DIE_SINKING optimised scheduling for die sinking.
Dimension	-
Default value	DEFAULT
Remarks	Available as of CNC Build V3.1.3105.01

P-STUP-00208	Logical ID of a channel agent in a Job Manager group
Description	The parameter defines the logical ID of a channel agent. Every agent (slave) in a Job Manager group is invoked by commands from the client by its logical ID "log_id" at the start. Due to the necessary uniqueness, no second channel agent may use the same logical ID in the same Job Manager group.
Parameter	Jobmanager.group[i].cnc_slave[j].log_idwhere where i = 0, 1 (index of Job Manager group, max. 1) where j = 0..n (index of a continuous list element. n: applications-specific)
Data type	UNS16
Data range	1 … 65536
Dimension	----
Default value	0
Remarks	There are two types of agent: CNC channels and PLC units. The logical ID always refers to a particular type

P-STUP-00209	Channel agent in a Job Manager group
Description	The channel agent (slave) designated by this parameter in a Job Manager group corresponds to an existing CNC channel. It cannot be assigned to any other group, neither as agent (slave) or as client (master). The number used must correspond to an existing channel number.
Parameter	jobmanager.group[i]. cnc_slave[j].channel_id (application-specific) where i = 0, 1 (index of Job Manager group, max. 1) where j = 0..n (index of a continuous list element. n: application-specific)
Data type	UNS16

Data range	1 … (application-specific)
Dimension	----
Default value	0
Remarks	The channel agent behaves like a "normal" channel. It has the additional property of being requested by any master in the same Job Manager group to execute a job. Job completion is signalled back to the client in the controller.

Channel parameters

P-CHAN-00145	**Activation of TCP display data**
Description	This parameter is used to activate W0 display data (TCP position referred to the Cartesian basic coordinate system of the machine - MCS). The TCP position is calculated dependent on the active kinematic ID based on the current command axis coordinates, the selected tool (length) and the kinematic offset parameters. The calculation also takes place when transformation is inactive. All axes in the kinematic structure must exist in the channel. 
Parameter	kin_trafo_display
Data type	UNS16
Data range	0: MCS display function inactive (default) 1: MCS display function active 2: MCS display function active (only for multistep transformation, see Addendum)
Dimension	----
Default value	0
Remarks	The axes must be homed to obtain the correct display. Programmed tool offsets (V.G.WZ_AKT.V.*) are only considered if they are followed by the programming of #KIN ID[<kinematic-ID>]. **Note:** As of CNC Build V3.1.3105 the data type of the parameter changed from BOOLEAN to UNS16.

Additions for multi-step kinematic transformations

A kinematic step can be defined in the parameter lists or in the NC program. A kinematic step is activated by programming the NC command #TRAFO ON.

For multi-step transformations see Concatenating transformations, multistep transformations.

Data value 0 (default):

Kinematic transformations are only executed to display axis positions if they are activated.

	PCS not active	**PCS active**
Kin. step 0 = defined, Kin. step 1 = defined	MCS = ACS	MCS = ACS PCS = f(ACS, CS_{active})
Kin. step 0 = active, Kin. step 1 = defined	MCS = f(kin. step 0)	MCS = f(kin. step 0) PCS = f(MCS, CS_{active})
Kin. step 0 = defined, Kin. step 1 = active	MCS = f(kin. step 1)	MCS = f(kin. step 1) PCS = f(MCS, CS_{active})
Kin. step 0 = active, Kin. step 1 = active	MCS = f(kin. step 0, kin. step 1)	MCS = f(kin. step 0, kin. step 1) PCS = f(MCS, CS_{active})

Data value 1:

Kinematic transformations are always executed to display axis positions as soon as they are activated. Defined Cartesian transformations are executed for display based on the axis coordinates.

	PCS not active	**PCS active**
Kin. step 0 = defined, Kin. step 1 = defined	MCS = f(kin. step 0, kin. step 1)	MCS = f(kin. step 0, kin. step 1) PCS = f(ACS, CS_{def})
Kin. step 0 = active, Kin. step 1 = defined	MCS = f(kin. step 0, kin. step 1)	MCS = f(kin. step 0, kin. step 1) PCS = f(ACS, CS_{def})
Kin. step 0 = defined, Kin. step 1 = active	MCS = f(kin. step 0, kin. step 1)	MCS = f(kin. step 0, kin. step 1) PCS = f(ACS, CS_{def})
Kin. step 0 = active, Kin. step 1 = active	MCS = f(kin. step 0, kin. step 1)	MCS = f(kin. step 0, kin. step 1) PCS = f(ACS, CS_{def})

Data value 2:

Kinematic transformations are always executed to display axis positions as soon as they are activated. Defined Cartesian transformations are executed for display based on the TCP coordinates.

	PCS not active	**PCS active**
Kin. step 0 = defined, Kin. step 1 = defined	MCS = f(kin. step 0, kin. step 1)	MCS = f(kin. step 0, kin. step 1) PCS = f(MCS, CS_{def})
Kin. kin. step 0 = active, Kin. kin. step 1 = defined	MCS = f(kin. step 0, kin. step 1)	MCS = f(kin. step 0, kin. step 1) PCS = f(MCS, CS_{def})
Kin. kin. step 0 = defined, Kin. kin. step 1 = active	MCS = f(kin. step 0, kin. step 1)	MCS = f(kin. step 0, kin. step 1) PCS = f(MCS, CS_{def})
Kin. kin. step 0 = active, Kin. kin. step 1 = active	MCS = f(kin. step 0, kin. step 1)	MCS = f(kin. step 0, kin. step 1) PCS = f(MCS, CS_{def})

P-CHAN-00430	**Suppress escape motions**
Description	Parameter for escape channel in die sinking. Suppresses requests to trigger an escape motion (negative external velocity with flushing motion) until the initialisation program ends.
Parameter	no_backward_before_prg_end

Data type	BOOLEAN
Data range	TRUE/FALSE
Dimension	-
Default value	FALSE
Remarks	Available as of V3.1.3108.5. Parameter is only effective in the escape channel if the escape channel was initialised before the first escape motion.

P-CHAN-00650	Definition of interpolator functionalities
Description	This parameter defines individual functionalities and the size of the look-ahead buffer in the interpolator, i.e. it defines the number of blocks required to calculate deceleration distance and dynamic planning.
Parameter	configuration.interpolator.function
Data type	STRING
Data range	See Parameter descriptions [▶ 121]
Dimension	----
Default value	FCT_IPO_DEFAULT
Remarks	

Interpolation function table

Identifier	Description
FCT_IPO_DEFAULT	FCT_LOOK_AHEAD_STANDARD
FCT_LOOK_AHEAD_LOW	30 blocks
FCT_LOOK_AHEAD_STANDARD	120 blocks
FCT_LOOK_AHEAD_HIGH	190 blocks
FCT_LOOK_AHEAD_CUSTOM	Any number of look-ahead blocks in the interval [0; 200]. Specification by parameter P-CHAN-00653.
FCT_SYNC	Synchronisation of an axis on a path group. Example: FCT_IPO_DEFAULT | FCT_SYNC
FCT_LOOK_AHEAD_OPT	The path velocity curve can be further improved for HSC machining by additional calculations. This generally reduces machining time. The additional calculations place greater demands on the controller hardware.
FCT_LIFT_UP_TIME	Automatic lifting/lowering of an axis (time-based coupling). Example: FCT_IPO_DEFAULT | FCT_LIFT_UP_TIME
FCT_SHIFT_NCBL	Path-controlled offset of M functions (dwell time). Example: FCT_IPO_DEFAULT | FCT_SHIFT_NCBL
FCT_CALC_STATE_AT_T	Calculation of path velocity at a time in the future. Function only available in combination with HSC slope and only as of V3.1.3057.0 Example: FCT_IPO_DEFAULT | FCT_CALC_STATE_AT_T
FCT_CALC_TIME	Calculation of interpolation time to next feed block (G01,G02,G03). Example: FCT_IPO_DEFAULT | FCT_CALC_TIME
FCT_CONTOUR_LAH	Contour look-ahead: advance output of motion blocks to the PLC as of V3.1.3104.07
FCT_DYN_POS_LIMIT	Dynamic limitation of axis positions
FCT_EXTENSION_EQUIDIST	Die-sinking EDM Planetary expansion

The look-ahead buffer size values specified above apply as of CNC Builds V2.11.2800 and higher; the following settings apply to CNC Build V2.11.20xx:

FCT_LOOK_AHEAD_LOW	30 blocks
FCT_LOOK_AHEAD_STANDARD	70 blocks

FCT_LOOK_AHEAD_HIGH	120 blocks

5.3 General

In order to use the escape and jumping functions, the parameter P-STUP-00033 [▸ 118] must be set for all channels.

The down channel activates the escape and orbit channels via the Job Manager and must therefore be configured as master in P-STUP-00208 [▸ 118] with the corresponding log_id and in P-STUP-00209 [▸ 118] with the corresponding channel_id.

The down and orbit channels are activated via the Job Manager and must therefore be configured as slave in P-STUP-00208 [▸ 118] with the corresponding log_id and in P-STUP-00209 [▸ 118] with the corresponding channel_id.

For channels where the contour look-ahead function is used, the parameter P-CHAN-00650 [▸ 121] must be configured with FCT_CONTOUR_LAH (alternatively with P-STUP-00070).

Example for configuring the Job Manager with the following channel arrangement: down channel = 1, orbit channel=2, escape channel=3

jobmanager.group[0].master[0].log_id 4715

jobmanager.group[0].master[0].channel_id 1 # Link to CNC channel 1

----

jobmanager.group[0].cnc_slave[0].log_id 2

jobmanager.group[0].cnc_slave[0].channel_id 2 # Channel 2

jobmanager.group[0].cnc_slave[1].log_id 3

jobmanager.group[0].cnc_slave[1].channel_id 3 # Channel 3

5.4 Down channel

The parameter P-STUP-00182 [▸ 118] defines the CNC channel scheduling. For die sinking, this value need only be parameterised in the down channel using DIE_SINKING.

```
schedule_config   DIE_SINKING ( P-STUP-00182 )
```

5.5 Orbit channel

The orbiting functionality must be activated by P-CHAN-00650 [▸ 121] (alternatively P-STUP-00070) for the orbit channel. When this parameter is configured with **FCT_EXTENSION_EQUIDIST**, the command #TRACK CHAN ON [▸ 97] [EXTEND="CH-DownShape" transforms the orbit axes with the radius provided by the down channel.

5.6 Escape channel

The parameter P-CHAN-00430 [▸ 123] suppresses requests to trigger an escape motion (negative external velocity, Jump) until the initialisation program ends.

P-CHAN-00430	**Suppress escape motions**
Description	Parameter for escape channel in die sinking. Suppresses requests to trigger an escape motion (negative external velocity with flushing motion) until the initialisation program ends.
Parameter	no_backward_before_prg_end

Data type	BOOLEAN
Data range	TRUE/FALSE
Dimension	-
Default value	FALSE
Remarks	Available as of V3.1.3108.5. Parameter is only effective in the escape channel if the escape channel was initialised before the first escape motion.

Index

6 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Download finder

Our download finder contains all the files that we offer you for downloading. You will find application reports, technical documentation, technical drawings, configuration files and much more.

The downloads are available in various formats.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on our internet page: www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49 5246 963-157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49 5246 963-460
e-mail:	service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49 5246 963-0
e-mail:	info@beckhoff.com
web:	www.beckhoff.com

More Information:
www.beckhoff.com/TF5292

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com