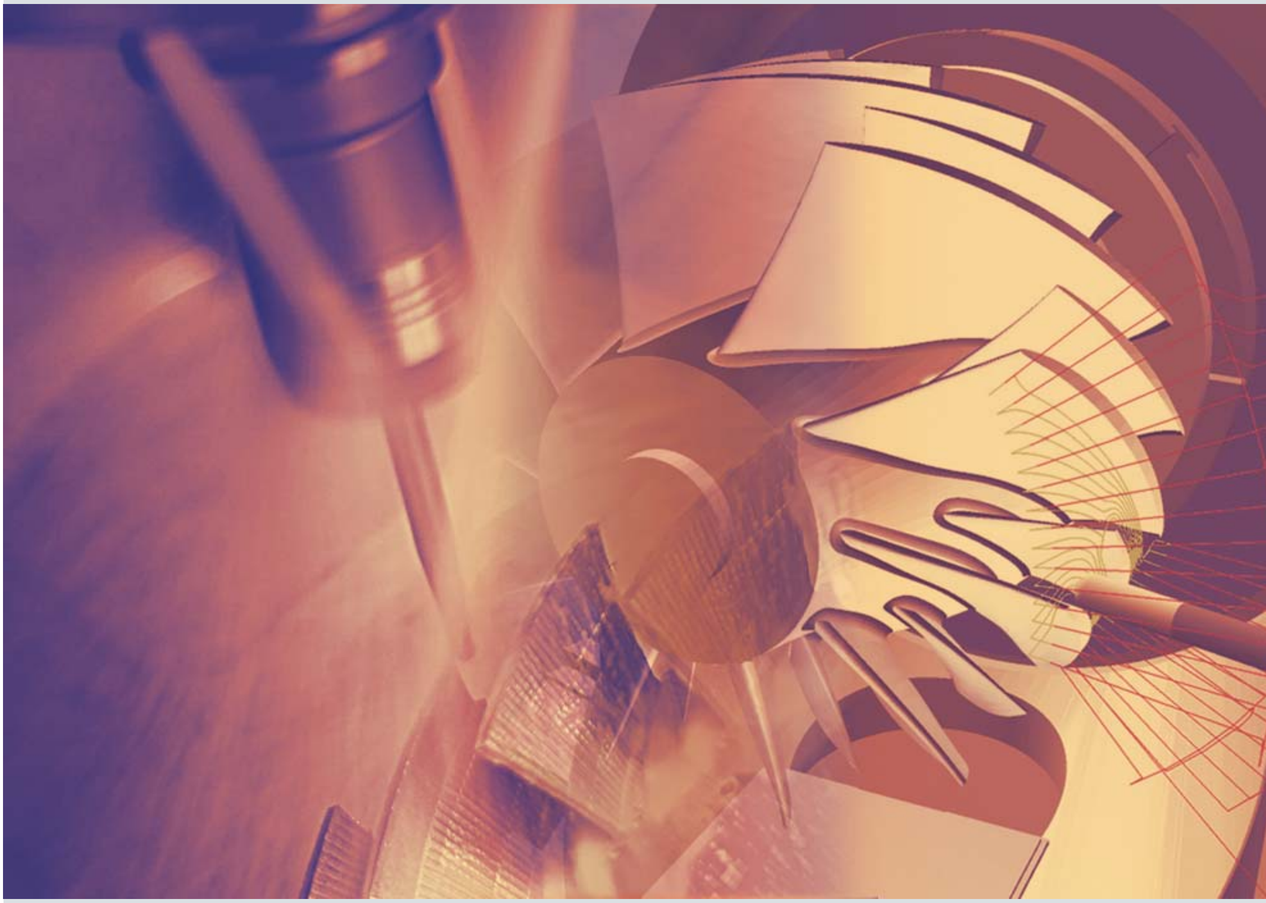
Milling with SINUMERIK

5-axis machining

Manual



SINUMERIK

SIEMENS

SIEMENS

SINUMERIK
5-axis machining

Manual

Valid for:

Control systems

SINUMERIK 840D
SINUMERIK 840D sl
SINUMERIK 840Di

Edition 05/2009

DocOrderNo. 6FC5095-0AB10-0BP1

SINUMERIK® documentation

Printing history, registered trademarks

Brief details of this edition and previous editions are listed below.
The status of each edition is shown by the code in the "Remarks" column.

Status code in the "Remarks" column:

A .... New documentation
B .... Unrevised reprint with new order number
C .... Revised version with new edition status

If substantive changes have been made on a particular page since the last edition, this is indicated by the new version code in the header on that page.

Edition	Order No.	Remarks
05/2009	6FC5095-0AB10-0BP1	A

Registered trademarks

SIMATIC, SIMATIC HMI, SIMATIC NET, SIROTEC, SINUMERIK, SIMODRIVE and SINAMICS are registered trademarks of Siemens AG. Other names in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

More information is available on the Internet at:
http://www.siemens.com/sinumerik

This document was created using a number of layout and graphic tools.

Other functions not described in this documentation might be executable in the control. This does not, however, represent an obligation to supply such functions with a new control or when servicing.

We have checked that the contents of this document correspond to the hardware and software described. However, differences cannot be ruled out and we can assume no liability for ensuring full consistency. Nevertheless, the information contained in this document is reviewed regularly and any necessary changes will be included in subsequent editions. Suggestions for improvement are welcome.

Technical data subject to change.

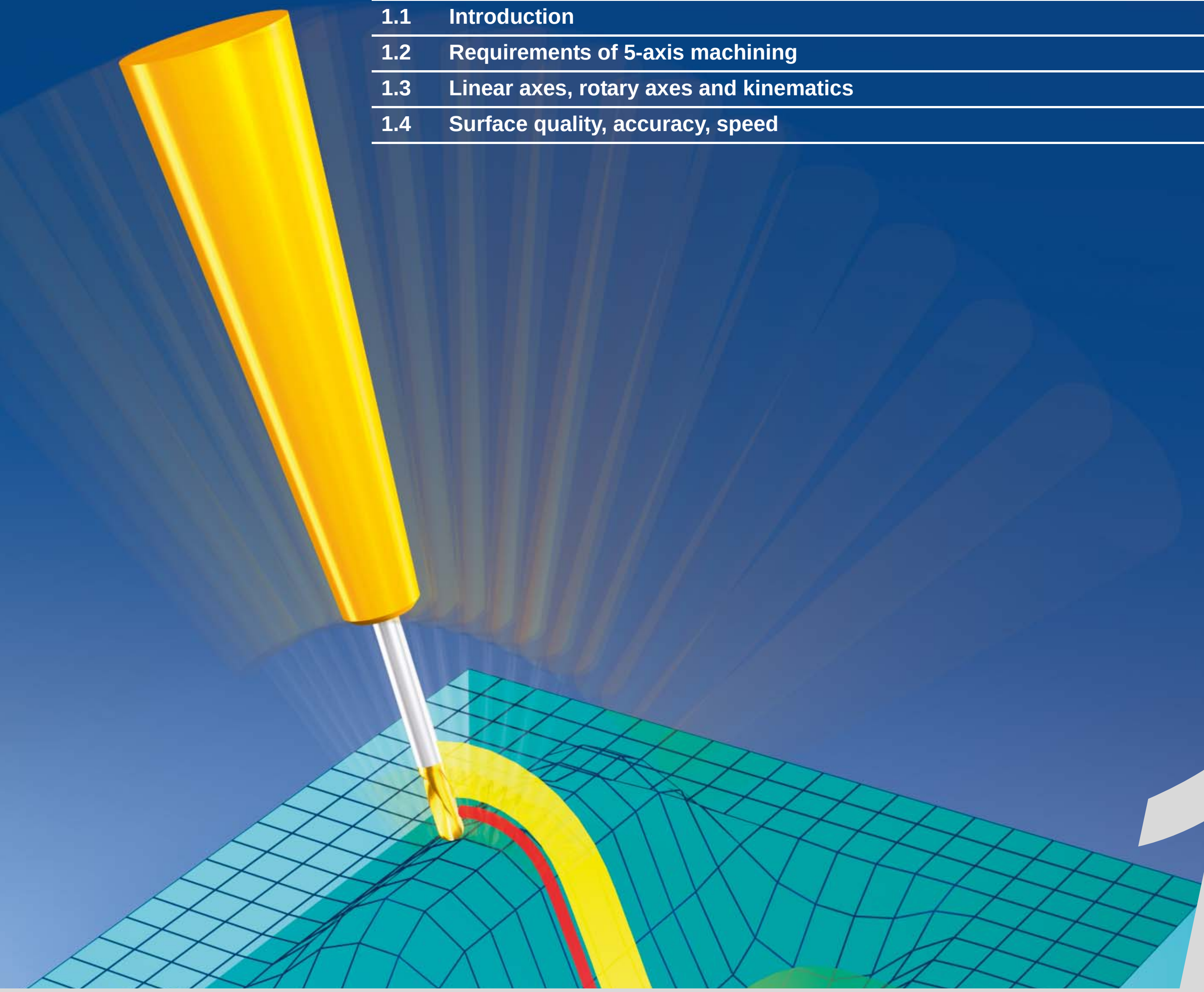
Order No.

Siemens-Aktiengesellschaft

Contents

Page

Basic information



1

1.1 Introduction

With multi-axis machining, the main objectives are to achieve perfect surface quality, precision, and speed without any need for remachining. Within this context, workflow is typically characterized by the CAD-CAM-CNC process chain. From the CAD system right through to the control system, Siemens can offer an integrated solution for these requirements in the form of its SINUMERIK products.

SINUMERIK controls are equipped with powerful, advanced functions which, when intelligently used, make the whole process of multi-axis programming and machining (particularly 5-axis machining) considerably easier while at the same time improving the results of production.

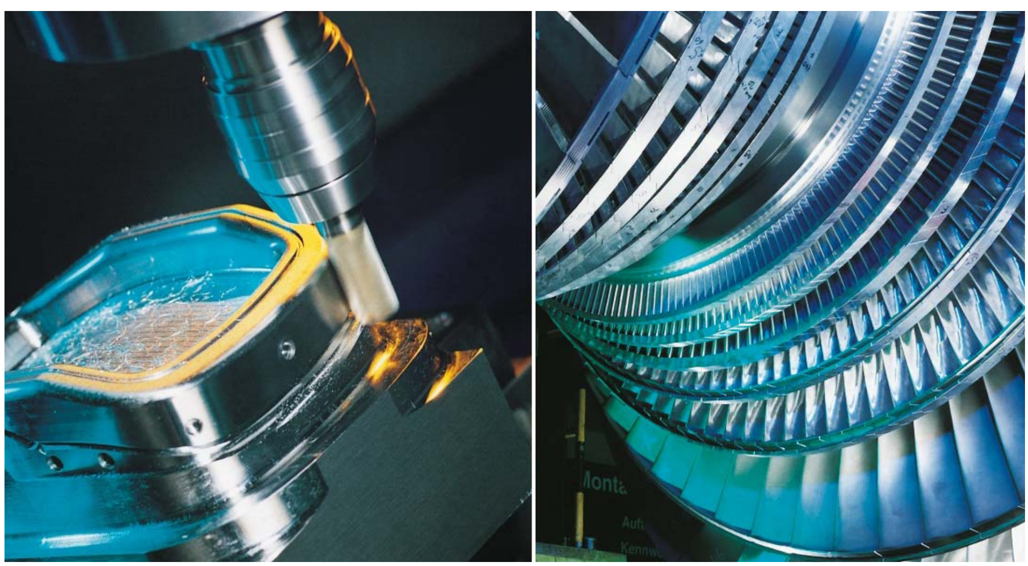
This manual is a compact resource aimed at experts working in industrial and research environments who need to get acquainted with the key basic principles of multi-axis machining. Building on this, it also provides machine users with practical tips so that they can organize their work efficiently and provides programmers with an insight into the functions of the control and the CAD/CAM system.

As regards the various application areas of multi-axis machining, the dedicated functions that have been specifically developed for each area are briefly introduced and considered in context.

Although many of the aspects cannot be covered in depth by this manual, you will find additional information in the appropriate Sinumerik documentation and relevant literature (See "Further information/documentation" on page 108.)

This manual is designed to supplement the Tool and Mold Making (3 axes) manual, which deals with the typical functions of 3-axis machining. Please refer to this original manual for more in-depth information about the basic principles and functions involved.

Range of 5-axis machining



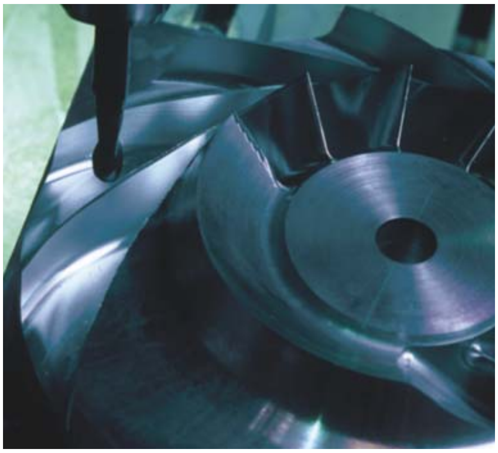
1.2 Requirements of 5-axis machining

Free-form surfaces, mold making

Design standards in all application areas are becoming increasingly more demanding. Expectations in terms of ergonomics, the air drag coefficient (CW value) or simply aesthetic appeal are creating a need for more complex surface geometries to be achieved in less time and with greater precision. The design primarily comes from CAD systems, the machining programs from CAM stations.

Nevertheless, the skilled machine tool operator still has overall responsibility (in terms of technology) for the quality of the mold and the complete tool.

Driving gear and turbine components, e.g. impellers

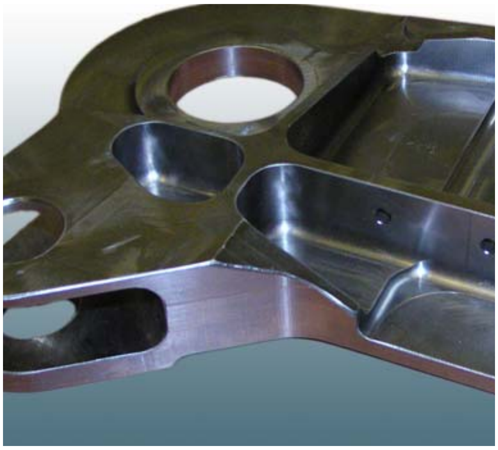


With SINUMERIK, Siemens can provide CNC systems that are perfectly suited to the demands of 5-axis machining as well as HSC applications:

- Simple to operate
- User-friendly programming at the machine
- Optimum performance throughout the CAD - CAM - CNC process chain
- Maximum control over quality at the machine
- Optimized 5-axis functions

Different requirements depending on the application area

Structural parts, aviation industry



Depending on the application, the requirements imposed on the control will vary and a whole range of different functions may be demanded.

Within this context, 5-axis machining can be broken down into three broad areas:

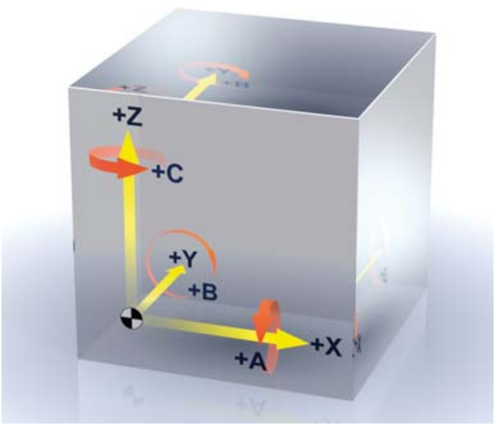
- Free-form surfaces (mold making)
- Turbine and driving gear components (impellers, blisks)
- Structural parts (aviation industry)

SINUMERIK can provide optimum support for each of these areas.

1.3 Linear axes, rotary axes and kinematics

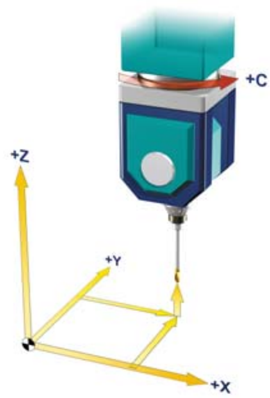
1.3.1 Axes and programming

Linear and rotary axes



To accommodate machining scenarios involving tools set at an angle or in order to mill geometries located anywhere in space, the three linear axes X, Y and Z are required along with two of the rotary axes A, B or C. It must be possible to control the axes simultaneously.

Tool tip motion

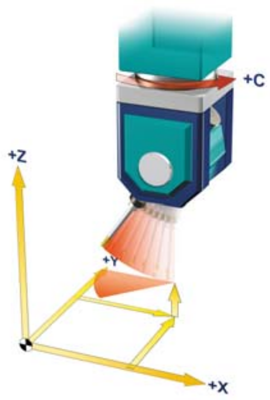


The tool position is approached in space using linear axes X, Y and Z. This enables the tool tip to adopt any position.

With 3-axis machining, you program the three linear axes to achieve the desired machining operations. The contour is milled line by line by moving the three linear axes.

If the tool also has to be set at an angle, you will need rotary axes as well.

Setting the tool at an angle



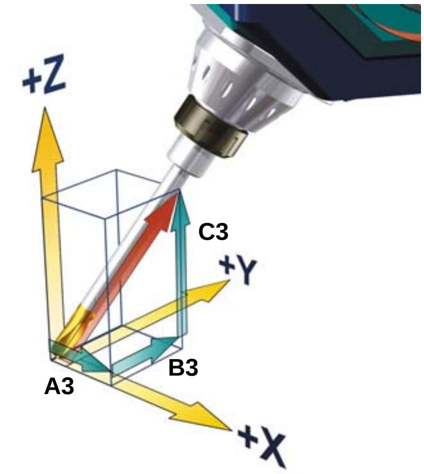
The inclination of the tool (i.e. the tool orientation) is changed using two axes of rotation, e.g. B and C.

This is necessary, for example, if the tool needs to be angled in relation to the machining surface or if you want to mill a pocket with inclined walls within the context of circumferential milling.

Using three linear axes and two rotary axes, theoretically any point in space can be approached with any tool orientation. This is the basis of 5-axis machining.

CNC programming options

Direction vector programming



For the purpose of 5-axis machining, the orientation of the tool needs to be defined in addition to the position setpoint of the machining point. The position setpoint is defined in the CNC program by means of the coordinate axes X, Y, and Z.

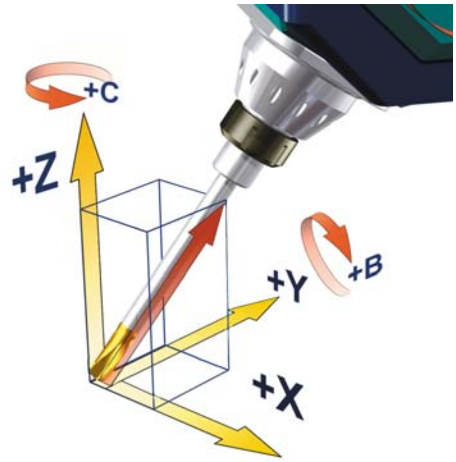
When describing the tool orientation, it makes sense to specify the direction vector A3, B3, and C3 so that the orientation can be programmed independently of the machine kinematics.

This enables the position setpoint and tool orientation to be defined in a unique manner.

The example shows the tool in the position (0,0,0) as a diagonal of a cube (35.26° in relation to X-Y plane).

N100 G1 X0 Y0 Z0 A3=1 B3=1 C3=1

Rotary axis programming



The same position can be achieved as with tool orientation by specifying it on the basis of the rotary axis positions.

The position in the above example would be expressed as follows:

N100 G1 X0 Y0 Z0 B=54.73561 C=45

The example shows the tool in the position (0,0,0) as a diagonal of a cube (35.26° in relation to X-Y plane).

NOTE

In addition to programming based on the direction vector and rotary axis positions, other forms of angle programming are also common. These include, for example, Euler or RPY angles. Further information regarding this can be found in Section *"Tool orientation" on page 50*.

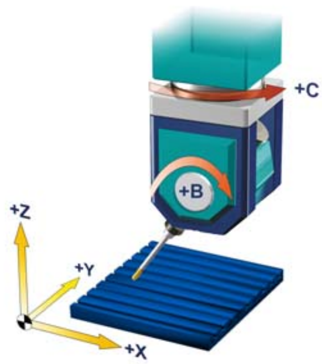
1.3.2 Kinematics of 5-axis machining centers

A 5-axis machine can control tool motion in 5 axes. These are the 3 linear axes (with which you will already be familiar) and an additional 2 rotary axes. There are different kinematic solutions for the two rotary axes. We will present the most common of these schematically. Machine tool manufacturers are constantly developing new kinematic solutions for different requirements.

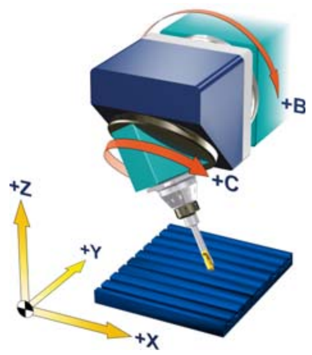
With SINUMERIK controls, even special kinematics can be controlled on the basis of the integrated, kinematic transformation feature. Special cases such as hexapods, etc. will not be explored in further detail here.

Two rotary axes in the head

Fork

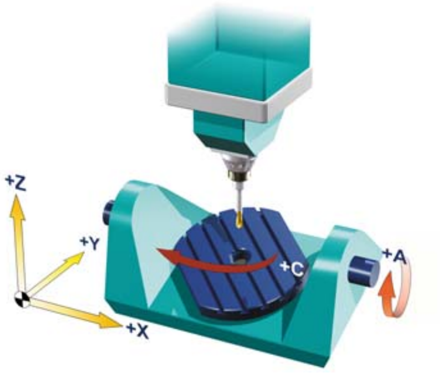


Nutated fork *

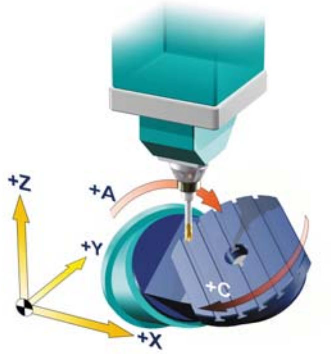


Two rotary axes in the table

Rotate/swivel

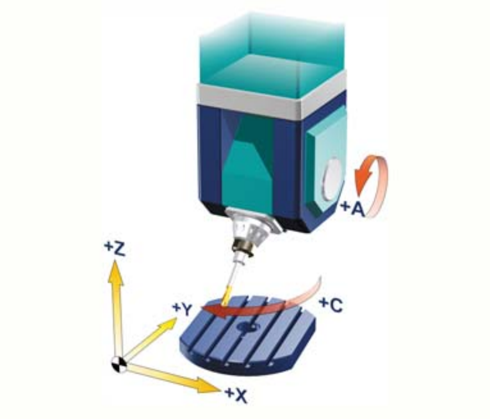


Rotate/swivel nutated *



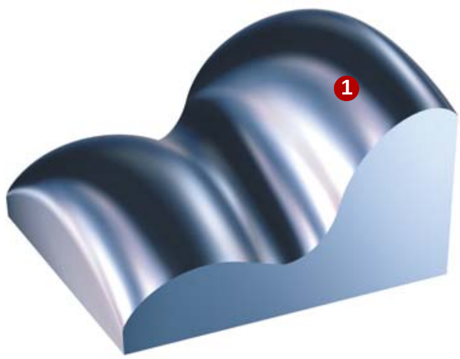
*: If the axis of rotation is not perpendicular to a linear axis, then this is known as a "nutated" axis.

One rotary axis in the head / One rotary axis in the table



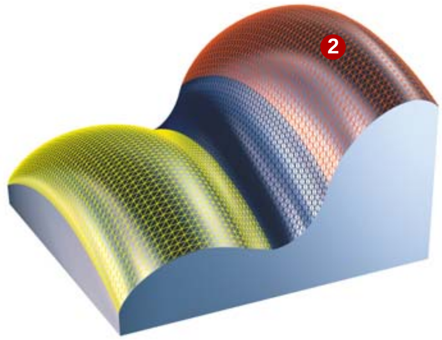
1.4 Surface quality, accuracy, speed

Special attention must be paid to the **CAD -> CAM -> (post processor) -> CNC** process chain when machining three-dimensional geometries, e.g. free-form surfaces. CAM systems generate NC programs for free-form surface machining. The CAM system receives the workpiece geometry from a CAD system. The CNC machine has to process the NC data generated and convert it into axis movements.



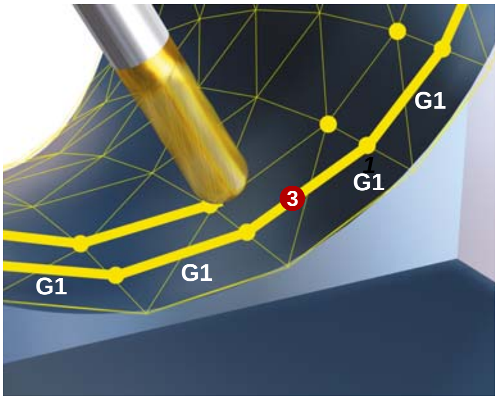
In CAD systems, surfaces (1) of higher orders are constructed (free-form surfaces).

For example, in order to be able to mill an entire surface - or for collision checking - the CAM system generally converts the CAD free-form surface into a polyhedron.



This means that the smooth design surface is approximated by a number of individual small planes (2) .

This produces deviations from the original free-form surface.

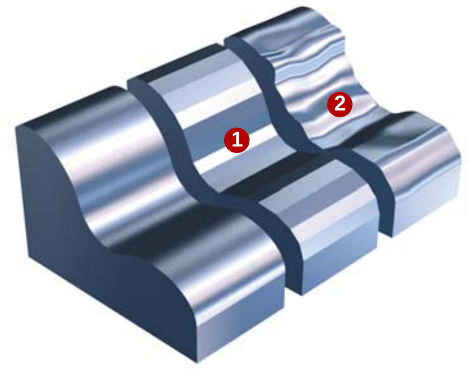


The CAM programmer overlays this polyhedron with tool paths. From these, the post processor generates NC blocks within the specified error tolerances. These usually comprise many short straight line elements, G1 X Y Z. (3)

Thus, the machining result is no longer a free-form surface, but a polyhedron. The small planes of the polyhedron can be visibly mapped on the surface.

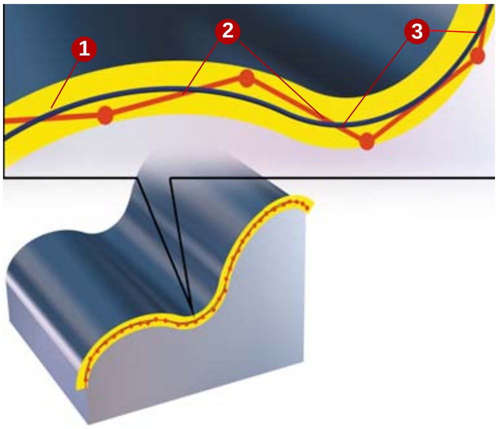
This can result in undesirable remachining.

SINUMERIK controls offer various functions so that remachining can be avoided:



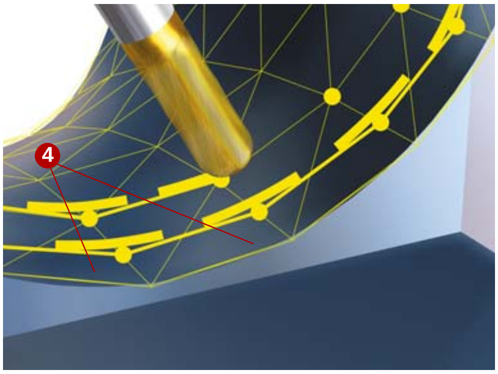
Compressor function (COMPCAD)

At the block transitions, the linear interpolation leads to step changes in velocity in the machine axes, which in turn can cause resonance in the machine elements and can ultimately be detected as a beveled pattern 1 or as the effects of vibrations 2 on the workpiece surface.



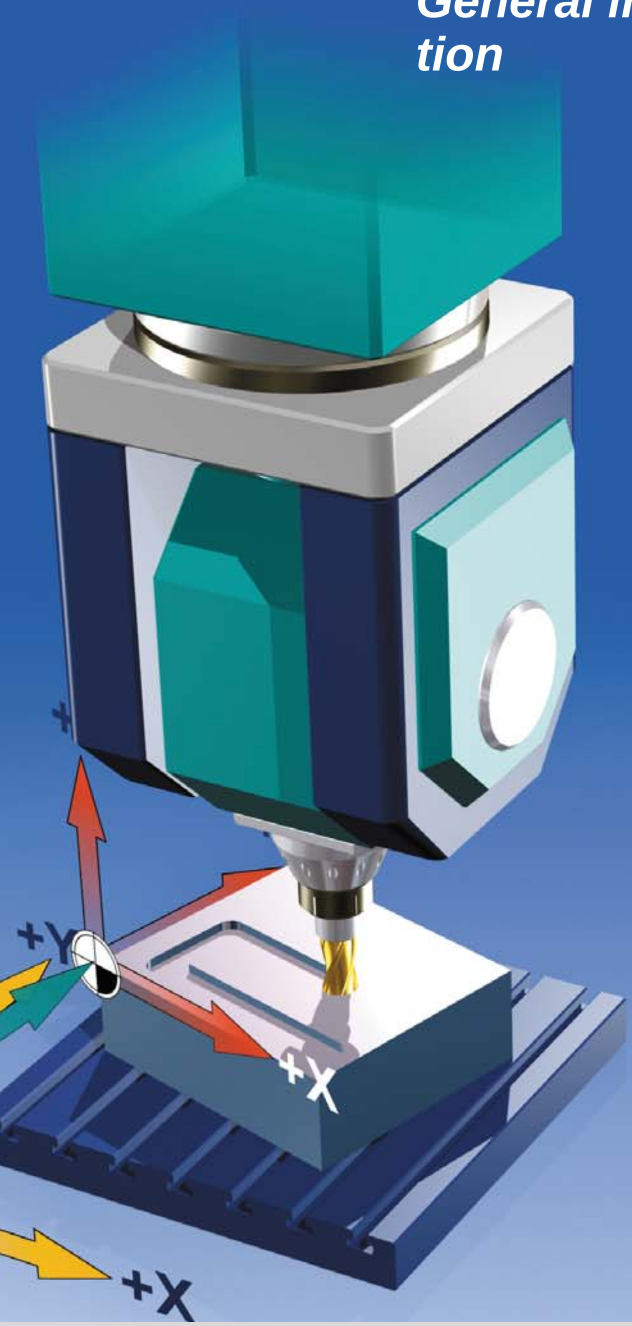
In accordance with the specified tolerance band, 1 the compressor takes a sequence of G1 commands, 2 combines them and compresses them into a spline 3, which can be directly executed by the control.

The compressor generates smooth paths and paths with constant curvature. The constant curvature results in a steady velocity and acceleration characteristic, meaning that the machine can run at higher speeds, thereby increasing productivity.



Programmable smoothing

Discontinuous block transitions can be smoothed with the smoothing function to create steady characteristics. This involves inserting geometrical elements 4 at the corners (block transitions).
The tolerance of these geometrical elements can be adjusted.



General information on workpiece production

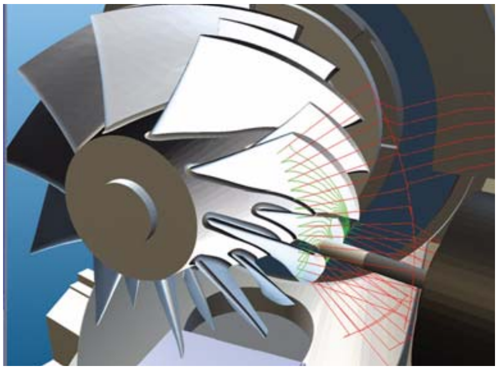
Contents		Page
2.1	Process chain for producing 5-axis workpieces	18
2.2	CAM system	19
2.3	Program structure for 5-axis machining	21
2.4	Introduction - Measuring in JOG and AUTOMATIC	23
2.5	Setting up and measuring workpiece in JOG	24
2.6	Measure tool in JOG	27
2.7	Measure workpiece in AUTOMATIC	30
2.8	Measure tool in AUTOMATIC	33
2.9	Checking/calibrating the machine with the kinematics measuring cycle CYCLE996	35
2.10	Quick View / Fast display	37
2.11	ShopMill - Graphical interface	38

2

2.1 Process chain for producing 5-axis workpieces

The production process chain generally starts with workpiece design. The data generated at this stage provides the basis for further processing and, ultimately, for production.

CAD/CAM



Production

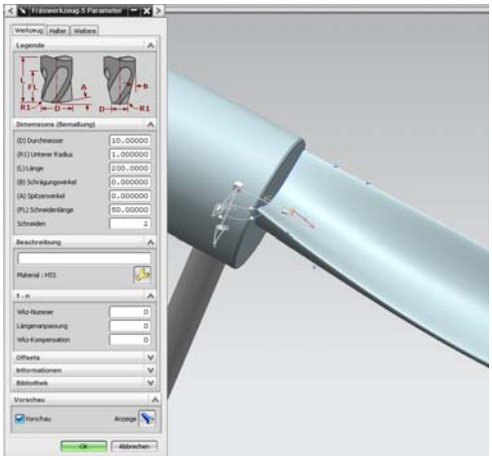
CAD model	Generating a CAD model
CAM strategies Calculation of tool paths	Using the CAM system's internal strategies and methods to generate the individual machining steps and to calculate the paths.
Simulation of tool paths	Simulation of the calculated tool paths with collision checking.
Post processor	Generation of the NC code in the post processor. Generally, it is the path at the tool tip (TCP) that is output in the NC code.
Setting up the machine	Data transmission of NC programs and setting up of tool and workpiece. Possible creation of a main program that will call the generated geometry programs.
Quality	Definition of the high speed settings, e.g. in CYCLE832, if not already taken into account by the post processor. These consist of the following: tolerance, compressor, continuous-path control, smoothing, jerk and speed.
Machining	Production of workpiece on the machine.

2.2 CAM system

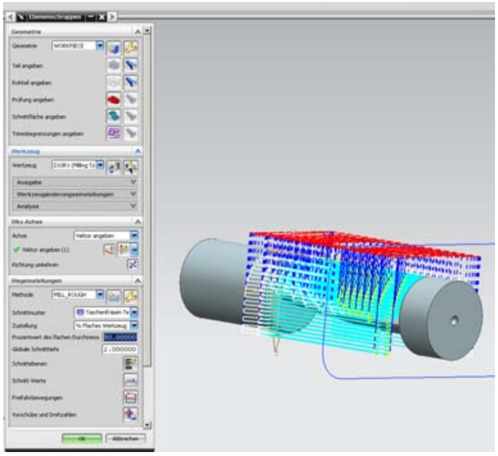
Within the context of the process chain, it is the CAM system that is responsible for the key task of generating the NC programs. The quality of this data plays a decisive role in determining the results of workpiece production.

This section outlines the procedure for generating the NC programs. Given the diverse range of systems available on the market, only a brief summary can be provided.

Tool definition



Plane roughing strategy



Procedure when working with a CAM system:

- **CAD data**
  Read CAD data into the CAM system. When reading the data in, a check should be performed to ensure that the surface geometries are free from defects, i.e. that there are no steps or jumps. Flaws in the data such as these will manifest themselves on the surface of the finished workpiece.
- **Chucking situation**
  Define the chucking situation and the geometry that can be freely machined in a chucking device. Define geometries such as a zero point.
- **Specify tools**
  Define the necessary tools on the basis of the machining task at hand and enter the technology data. As a general rule, CAM systems are able to read the data in from tool databases. Amongst other things, the tools determine which subsequent machining strategies can be used, e.g. whether plunge cutting is possible with the tool.
- **Specify machining strategies**
  Define the machining process for the various workpiece geometries using the appropriate strategies. First of all, roughing strategies are applied, e.g. roughing in the Z plane or equidistant from the surface geometry. CAM systems offer various options from 2 1/2-axis to 5-axis machining.
  The tool paths can be generated automatically or defined manually; these include, for example, approach and retract strategies or special milling strategies such as trochoidal milling.
  The machining strategies are gradually introduced in stages as part of this procedure and are supported by automatic residual material detection, for example.

- **Calculation and simulation**
  When simulating the calculated tool paths/machine movements different levels of quality can be used, from straightforward simulation of the tool paths through to complete simulation of the G and M codes that takes account of all machine-specific and control-specific data. Here, potential collisions can be detected and avoided, for example, and the machine's maximum axis traversing ranges can be taken into account.
- **Output of the NC code with the post processor**
  The post processor converts the sequences into NC programs taking into account the control-specific syntax and the control's special functions. For this purpose, CAM systems make use of universal post processors or special post processors that have been optimized for the SINUMERIK system. Manufacturer-specific functions such as separate coolant strategies must be implemented in the post processor in consultation with the machine manufacturer.

Key parameters

When working with CAD/CAM systems, certain tolerances and levels of accuracy that will have an impact on subsequent machining must be observed.

Tolerance The CAM system uses the CAD surface (spline) to generate a contour consisting of linear traversing blocks (straight line elements). The extent to which the linear contour deviates from the real contour from the CAD system is known as the chord error or chord tolerance. This tolerance depends on the strategy used and is greater in the case of roughing strategies than with finishing strategies. When the NC programs are executed on the machine, the tolerance is specified by the CAM system in CYCLE832 so that optimum results can be achieved in terms of surface quality and contour accuracy.

Accuracy When outputting the NC blocks from the CAM system, you can specify the number of decimal places. The required level of accuracy is dependent on the type of interpolation. In the case of linear axes (X, Y, Z), at least 3 decimal places should be used for 3-axis programs.

If the blocks are to be output as rotary axis positions, with 5-axis programs 5 decimal commas should be used in the linear and rotary axes for optimum surface quality. If they are to be output in the form of a direction vector, we recommend 5 decimal places in the linear axes and 6 for the direction vectors.

2.3 Program structure for 5-axis machining

Recommendation for a practical program structure with CYCLE832

For machining purposes, a main program is generated (1) that includes all technology data. The main program calls one or more subprograms (2), (3) that contain the workpiece's geometry data. The tool change defines how the content is divided into subprograms.

Example

Main program

```
Call.MPF (1)
N10     T1 D1
N20     M6                                  ; Tool change
N30     M3 S15000
N40     G54
N50     CYCLE832 (0.05,112103) (4)          ; High-speed settings
N60     EXTCALL "CAM_Rough" (5)             ; All programs should
                                            ; be located in one
                                            ; directory. If this is not
                                            ; the case, then the
                                            ; paths must be
                                            ; specified.
N70     T2 D1
N80     M6                                  ; Tool change
N90     M3 S20000
N100    CYCLE832 (0.005,112101) (4)
N110    EXTCALL "CAM_Finish" (5)
N120    M30
```

Subprogram

```
CAM_Rough.SPF (2)
N1      G90
N2      G0 X0 Y0 Z10

N3      G1 Z0 F500
N4      G1 X-1.45345 Y0.67878 F10000
N17     G1 X-1.81412 Y0.84245

...

N5046   G1 X-4.11845 Y-11.44212
N5047   G0 Z10
...
N5051   G1 Z-2.13247 A3=0.34202 B3=0 C3=0.93969 F5000 (6)
...
N6582   G1 X7.60978 Y3.55541 A3=0.34202 B3=-0 C3=0.93969
N6583   G0 Z50 A3=0.34202 B3=-0 C3=0.93969
N6584   M17
```

Subprogram

```
CAM_Finish.SPF (3)
N1      G90
N2      G0 X0 Y0 Z10 A3= B3= C3=
....    .......
N7854    M17
```

Main program: The main program includes the two key functions for milling, CYCLE832 ❹ and EXTCALL ❺ .

CYCLE832 ❹ : CYCLE832 has been specifically developed for the program structure shown, where technology data and geometry data are separated. It brings together all the key commands and activates control functions. The machining technology for milling is defined in CYCLE832. For the roughing program "CAM_Rough" using T1, the parameters in CYCLE832 were geared towards achieving a high velocity. For the finishing program "CAM_Finish" the parameters were geared towards achieving a high level of accuracy and surface quality. TRAORI can also be called in CYCLE832. The current work offset is retained. For more information about CYCLE832, see Section See "High speed settings – CYCLE832" on page 63.

EXTCALL ❺ : CAM programs are generally extremely large, which is why they are stored in an external memory. The EXTCALL command is used to call the subprograms from various locations, including external memories.

Subprogram: In the subprogram, G90 for absolute programming is immediately followed by the geometry blocks. In our example, these initially take the form of blocks for 3-axis milling, which are then followed by the blocks for 5-axis simultaneous milling ❻ . These are designated A3, B3, and C3.

2.4 Introduction - Measuring in JOG and AUTOMATIC

Measuring in JOG

When **measuring in JOG** mode (setup), the machine is prepared for machining. This involves determining the dimensions of the workpiece and the tool, which are still unknown.

- Manual measurement is used to prepare the machine for machining.
- Manual measurement is used to determine unknown workpiece or tool geometries.
- The operator interacts with the machine during manual mode in order to perform the measurement.

Measuring in AUTOMATIC

When **measuring in AUTOMATIC** mode (in-process measurement), workpiece tolerances are determined within the production process and the tool parameters are monitored. The nominal dimensions of the tool and workpiece are already known.

- Automatic measurements are performed to check that the workpiece measurements conform to specifications.
- Automatic measurements are performed to correct known workpiece and tool geometries.
- The measurement is performed by calling a measuring cycle in the machining program.

Measuring cycles for all measuring tasks

The SINUMERIK features an extensive pool of practical measuring cycles for measuring tasks in JOG and AUTOMATIC modes. These measuring cycles enable you to measure workpieces and tools using a graphically supported process.

The measuring tasks are carried out with touch trigger or non-switching probes and dynamometers or laser measuring systems.

3D touch trigger probe

Measuring cycle for measuring a hole

2.5 Setting up and measuring workpiece in JOG

Once the machine has been powered up and the reference point approached, the axis positions relate to the machine coordinate system. The work offset signals to the control the position of the workpiece in the machine coordinate system.

2.5.1 Measuring cycles in JOG

Using the semi-automatic "Measuring in JOG", the required measuring function is selected on the control using the appropriate softkeys. The displayed input screens are used for assigning the function parameters. You must bring the tool or probe into a permissible starting position for the measurement task concerned, e.g. using the traversing keys or handwheel (manual traversing).

The **measuring cycles** support the following functions:

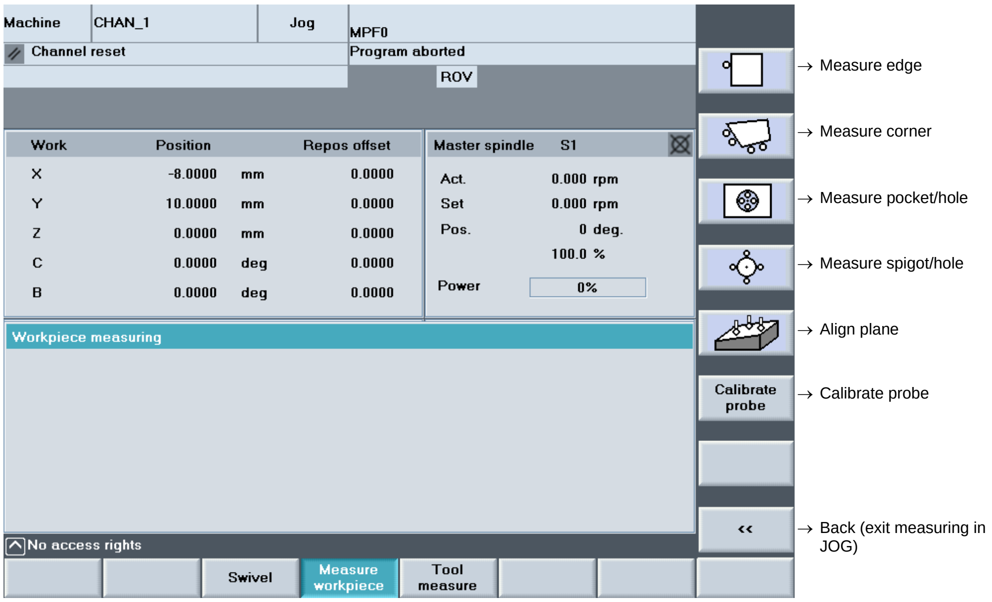
- Calibrating a probe (calibrate)
- Detecting dimensions and position of workpiece geometries, e.g. in order to set up the machine

NOTE

If only one single workpiece is to be machined, then measurements are performed in JOG mode (as described below). If several similar parts are to be machined in the same fixture, the measuring cycles are used in Automatic mode (the approximate zero point must be set up).

Practical measuring cycles are provided to facilitate measurements.

Measuring cycles in JOG for SINUMERIK

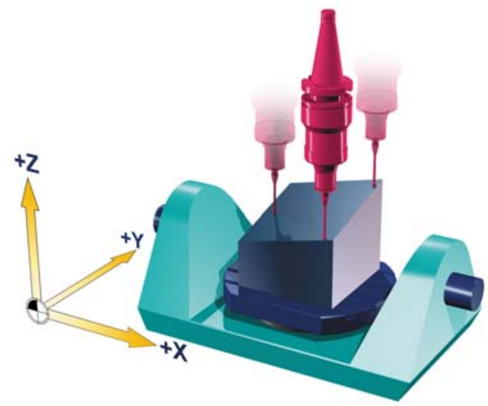


Requirements for using cycles

- The measuring cycles must have been installed
- The workpiece must have been clamped
- The probe must have been calibrated and must be active; the tool offset must have been activated

Examples of measuring cycles

Align plane



Align plane (CYCLE998)

The plane's position within the machine coordinate system is determined using 3-point measurement. If the Swivel/5-axis transformation option has been set up, the table/head will be aligned in accordance with the compensation values when the WO is determined.

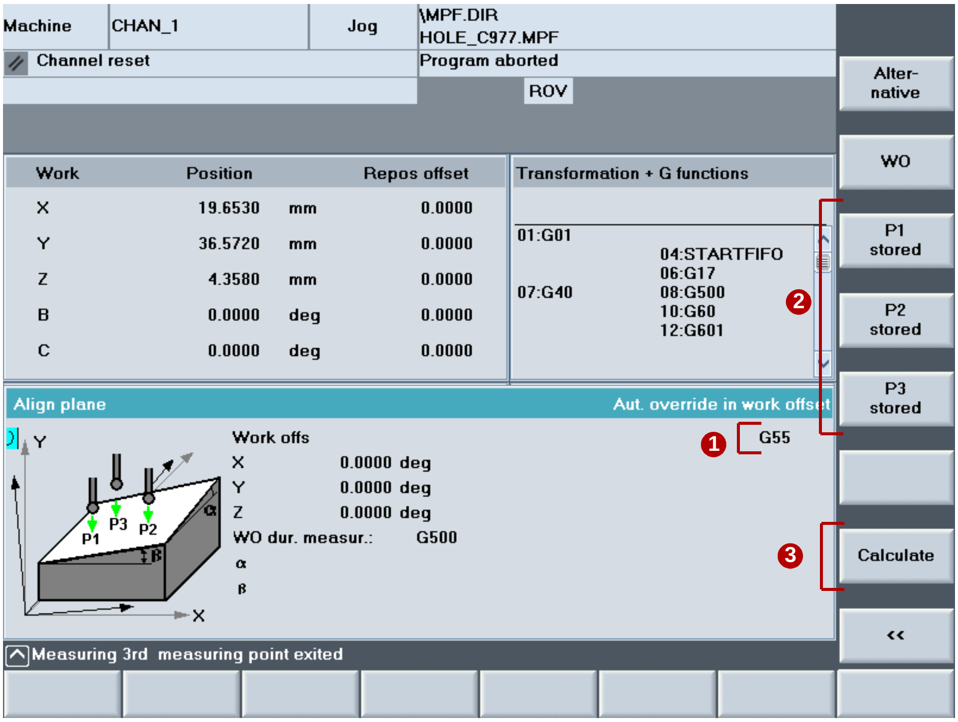
NOTE

The *Tool and Mold Making (3 axes)* manual contains detailed examples of how to set up machines with two rotary axes.

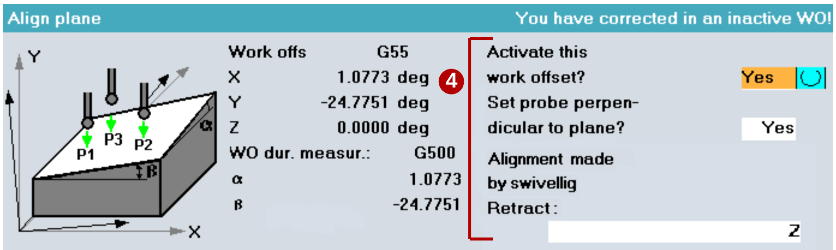
2.5.2 Example measuring process

The **Align plane** measuring cycle is intended for horizontal alignment of the workpiece plane. The compensation should be made in the work offset G54.

- Select the **Align plane** softkey.
- Traverse the probe to measuring point **P1**.
- Select the work offset, e.g. **G54** (1)
- When you press "NC Start", the respective measuring points P1, P2, and P3 (2) are approached automatically, starting from the manually selected preliminary position. This means that the probe approaches the workpiece, is triggered and then retracts to the start position.
- When all measuring points have been approached, press "Calculate" (3).



- The plane is horizontally aligned. If the Swivel/TRAORI option has been set up on the machine, you can align the workpiece with the axes immediately.
  If the Swivel option has not been set up, you can align the probe perpendicular to the plane being measured. The compensation is then only made in the coordinate axes without any visible swiveling of the table or head (4).

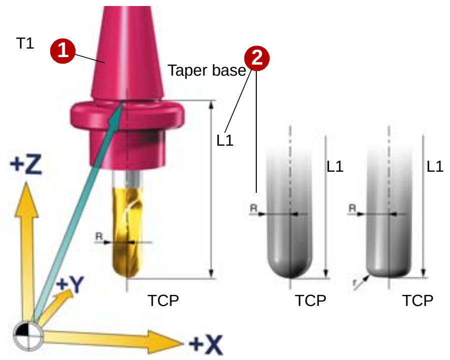


2.6 Measure tool in JOG

When executing a program, the various tool geometries must be taken into account. These are stored as tool offset data in the tool list. When the tool is called, the control considers the tool offset data.

You can determine the tool offset data (i.e. the length and radius or diameter) manually, semi-automatically, or automatically in JOG mode using a dynamometer.

2.6.1 Tool reference point



TCP = Tool Center Point

The tool magazine is loaded as usual, tool numbers T1, T2 etc. (1) are entered into the tool table and the tools are assigned a tool offset D (2) consisting of radius "R" and length "L1".

The CAM system already takes into account the tool diameter when the geometry program is being created. The calculated tool path refers to the miller center point (center point path).

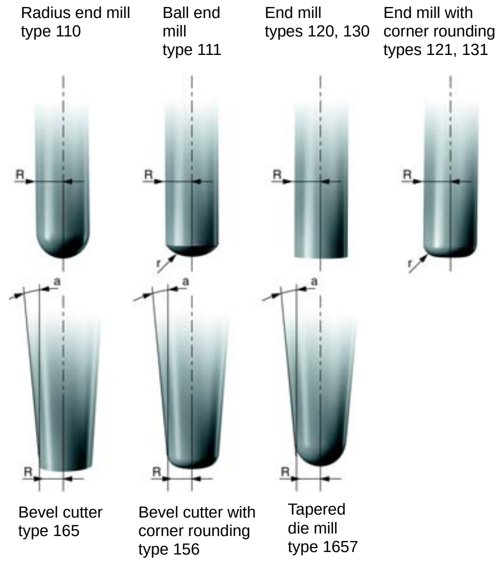
This means that to measure the length of the tool, you must use the same reference point (TCP) as the CAM system. For the purpose of determining the tool length, always remember to check the reference point the CAM programmer used to measure L1. The TCP can either be located at the tool tip or further upwards in the milling tool - e.g. for radius end mills at the center of the radius.

NOTE

CAM systems define the position of the TCP differently depending on the tool shape. For Siemens controls, it is assumed that the TCP is at the tool tip. If the CAM system specifies a different TCP position then this difference must be taken into account when specifying the tool length.

NOTE

Coordinate the following with the CAM programmer: To avoid significant tool deflection, the CAM programmer should keep the tool length as short as possible.



You can specify additional tool data for face milling depending on the tool type.

In an NC program, the control system uses this data and path corrections G41, G42 - defined in the program - to execute the necessary path and length corrections.

2.6.2 Example: Measure tool in JOG

Function

The **"Measure tool"** function permits the following functions:

- Calibrating a dynamometer (calibrate)
- Determining the tool length or the radius of milling tools or the tool length of drills and entering this data into the tool offset memory

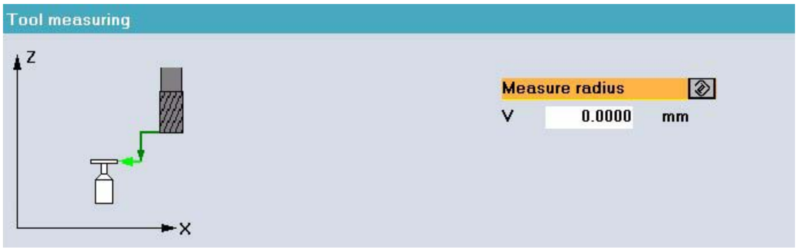
Requirements for using cycles

- The measuring cycles must have been installed
- The tool must have been loaded
- The dynamometer must have been calibrated and must be active

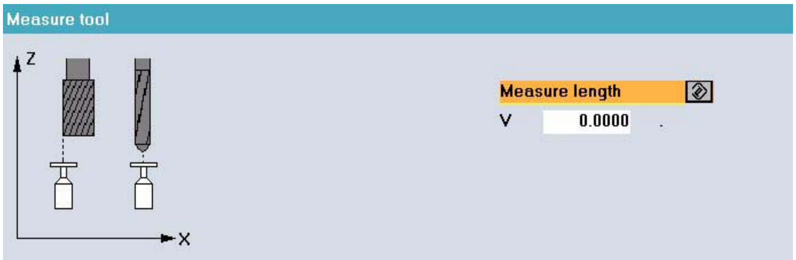
Procedure

- In JOG mode, select the **Measure tool** softkey. Then, you can select the **Radius/Length** or **Calibrate probe** softkey from the horizontal softkey bar.
- Click the relevant softkey (**Radius** or **Length**) and enter the offset in the case of special tools, e.g. those with a rounded cutting edge.
- Click **NC Start** to initiate the measuring process; the tool offsets for radius and length 1 will be entered in the active tool offset data.

Measuring the radius



Measuring the length



2.7 Measure workpiece in AUTOMATIC

With in-process measurements in Automatic mode, NC programs are specifically parameterized for the measuring task (measuring cycles). The input screens of the program editor are used for parameter assignment. The measuring points to be approached and the measuring task are automatically implemented in accordance with the measuring program.

For measuring tasks that are repeated, e.g. setup for multi-part machining, you can simply run the NC programs and the workpieces will be measured plus aligned, and the tools corrected automatically.

The workpiece is measured to determine workpiece tolerances in the production process. Depending on the measuring cycle used, you can select the following options as the result of workpiece measurement:

- Measurement only without offsets (actual value is measured)
- Work offset compensation (setpoint - actual value deviation)
- Tool data offset (setpoint - actual value deviation)

2.7.1 Measuring cycles in AUTOMATIC for SINUMERIK

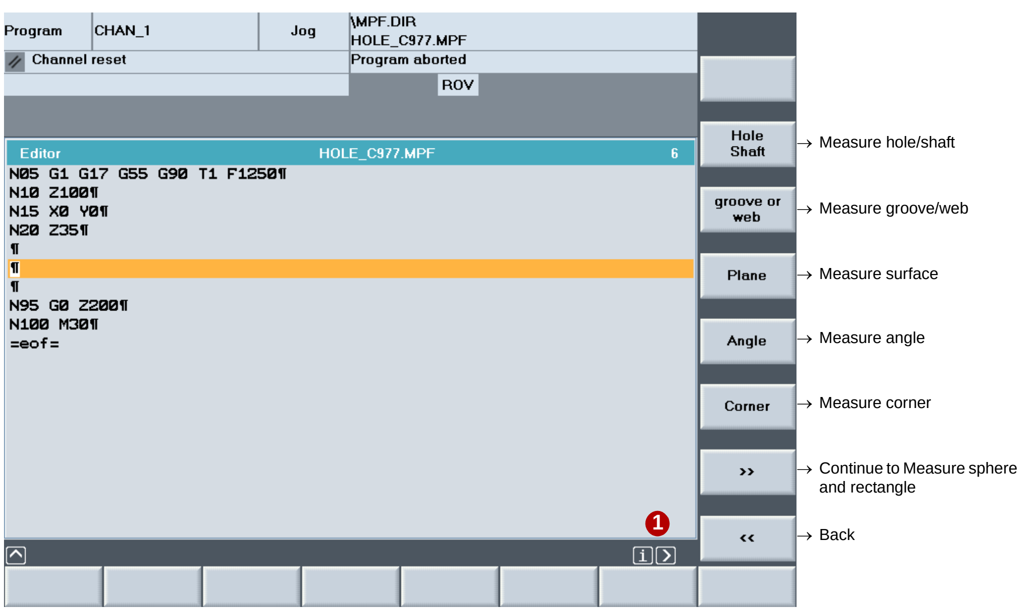
Practical measuring cycles are provided to facilitate in-process measurements.

▶ You can select the measuring cycles within the NC program using the softkeys **Measure Mill > Measure workpiece**.

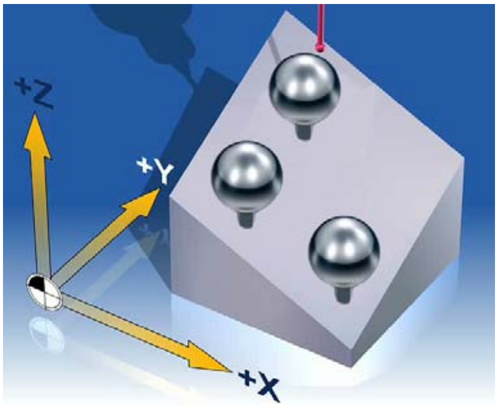
NOTE

The softkeys for in-process measurements can be found on the expanded softkey bar. You can switch to this softkey bar by pressing the expansion arrow > (1)

Measuring cycles in AUTOMATIC



Measuring three spheres



Measure sphere (CYCLE997)

You can use the "Measure sphere" cycle to measure one sphere or three spheres of the same size. With a single sphere, you can determine the position and diameter of the sphere or implement a work offset compensation. If there are three spheres, you can also determine the position of the spheres on the plane.

2.7.2 Example measuring process in Automatic mode

The procedure will be illustrated for you on the basis of the **Measure sphere** and Determine WO (CYCLE997) functions. The cycle can be used to measure one sphere or three spheres of the same size. With **"Measure 3 spheres"**, the angles in space of the sphere grouping can also be determined.

Requirements for using cycles

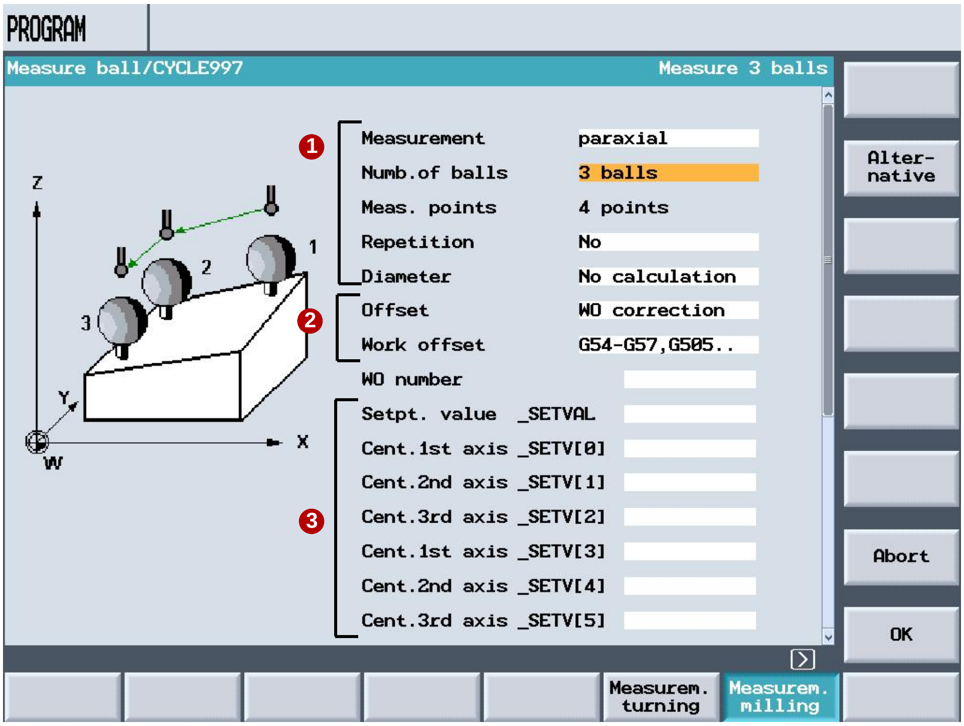
- The measuring cycles must have been installed
- The workpiece must have been clamped
- The probe must have been calibrated and must be active; the tool offset must have been activated

Compensating the work offset with the "Measure 3 spheres" function

- Create a new program for measuring the workpiece or edit an existing one.
- Select the **>>** softkey first, followed by the **Measure sphere** softkey.
- Select the type of measurement and number of spheres; in this example: 3 spheres and axially parallel measurement (1). The diameter of the spheres is determined during the measurement process.
- You can define whether the result of the measurement should be a compensation or merely a measuring process (2).
  - Compensation in the work offset, specifying the WO
  - Compensation in the tool offset data
  - Measurement only

  As you are setting up the workpiece here, the compensation is made in the WO.

▶ Enter the desired values for the spheres such as the diameter and sphere centers and assign additional cycle parameters (3).
At the end of the measuring process, the translatory and rotary components will be corrected in the active work offset frame.



2.8 Measure tool in AUTOMATIC

A practical measuring cycle is provided to facilitate the in-process measurement of tools. The cycle determines the length and the diameter of the tool using a calibrated dynamometer.

▶ You can access the measuring cycles in the NC program by selecting the **Measure Mill > Measure tool** softkeys from the expanded softkey bar.

Requirements

- The measuring cycles must have been installed
- The dynamometer must have been calibrated
- The tool must have been clamped

In Automatic mode, you can automatically measure the tool data or enter it as a tool offset. In the following example, you will generate a program that determines the tool length and the radius and enters this data into the tool offset.

Determining the tool length:

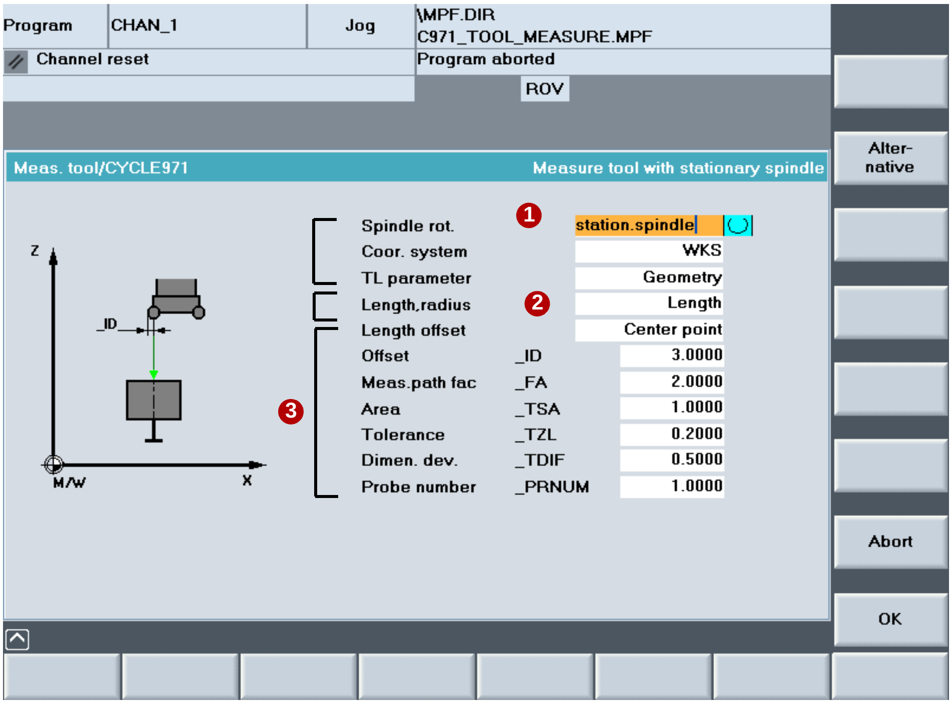
▶ Create a new program to measure the tool.

▶ Select the measuring cycle - **Measure tool**.

▶ The measurement is performed with the spindle stationary and the measured values are entered into the tool geometry component (1).

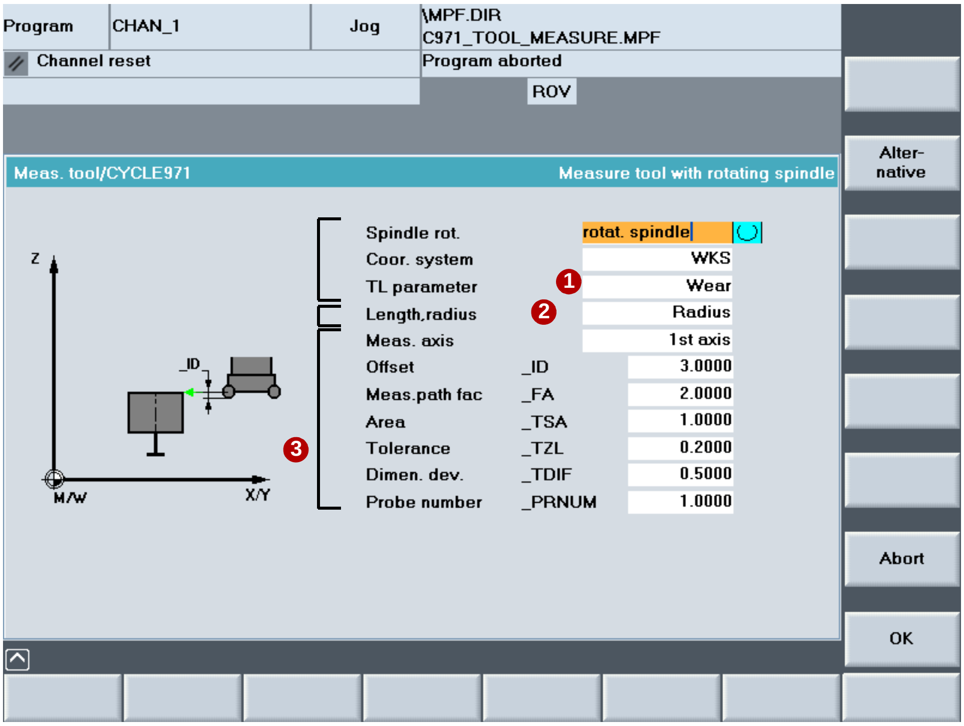
▶ Select the length as measured value (2).

▶ Assign parameters for the measuring process (3).



Determining the tool radius:

- The measurement is performed with the spindle rotating and the setpoint/actual value difference is entered optionally into the radius wear (1).
- Select the radius as measured value (2).
- Assign parameters for the measuring process (3).

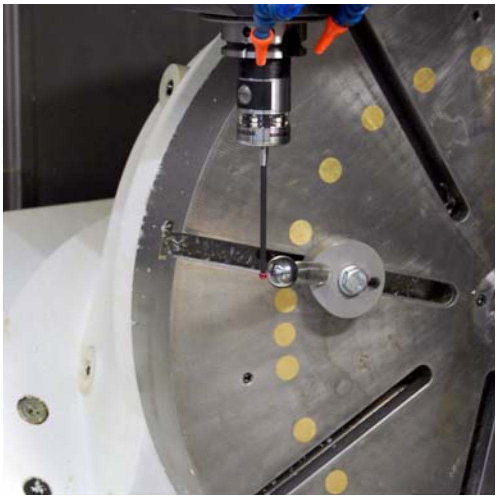


2.9 Checking/calibrating the machine with the kinematics measuring cycle CYCLE996

The requirements associated with 5-axis machining can only be met by means of high-precision machines. As regards kinematic transformation, this means that the vectors for NC-controlled or manually alignable rotary axes must be determined and entered into the control. CYCLE996 is a special measuring cycle that has been designed specifically for this purpose. It uses a calibration sphere and a calibrated probe to measure the rotary axis vectors automatically. In practical applications, the cycle makes control easier and improves the quality of the process, as compensations of the rotary axis vectors (due, for example, to temperature fluctuations, or other influential factors) can be checked automatically.

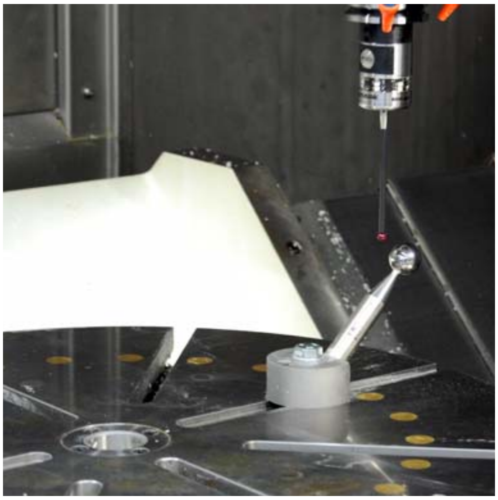
For measurement purposes, a calibration sphere is mounted on the table. The sphere is measured with the probe at three separate swivel positions of the rotary axis and the values obtained are recorded. CYCLE996 must be called three times for each rotary axis using different swivel positions. The kinematics calculation is carried out in full once all the rotary axes have been measured.

A axis, table kinematics



Three sphere positions (offset from one another by approx. 120°) are measured for the A axis.

C axis, table kinematics

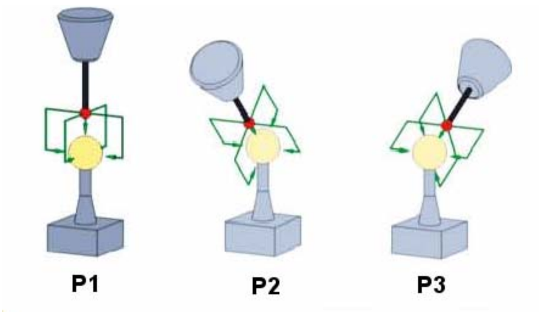


Three sphere positions (offset from one another by approx. 120°) are measured for the C axis.

Checking the machine with CYCLE996:

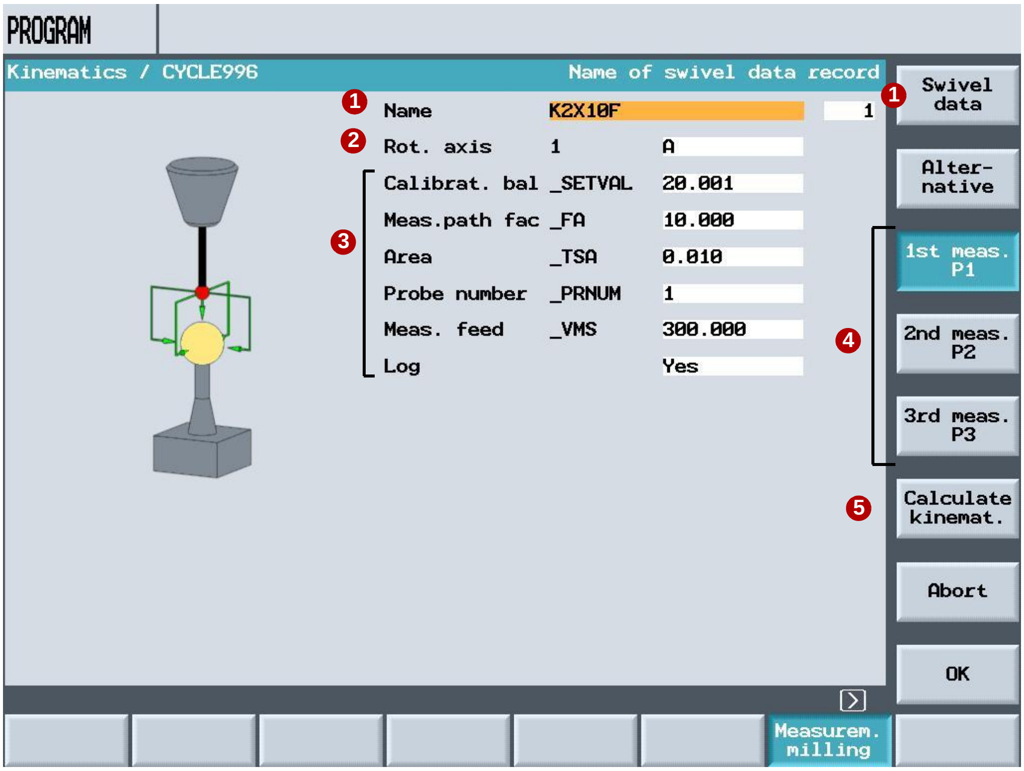
- Create a new program to measure the kinematics.
- In the part program editor, select the **>** > **MEASURE MILL** > **MEASURE WORKPIECE** > **>>** > **KINEMATICS** softkeys in the Programs operating area.
- Generate a new swivel data record (where the rotary axis is swiveled) (**1**). Ideally, the swivel positions should create an equilateral triangle, i.e. each one should involve a swivel of 120°.
- Select the rotary axis you want to measure (**2**).

▶ Assign parameters for the measuring process (3) for measurements 1 to 3 (4).



▶ Press the "Calculate kinematics" softkey once all the rotary axes have been measured (5).

.



As soon as you press the softkey, the **Calculate kinematics** dialog will open. The following options are available in terms of how you then utilize the measurement results:

- Measurement only (measure and calculate vectors)
- Entry of the results (measure, calculate vectors, and enter vectors in swivel data record for correction purposes)

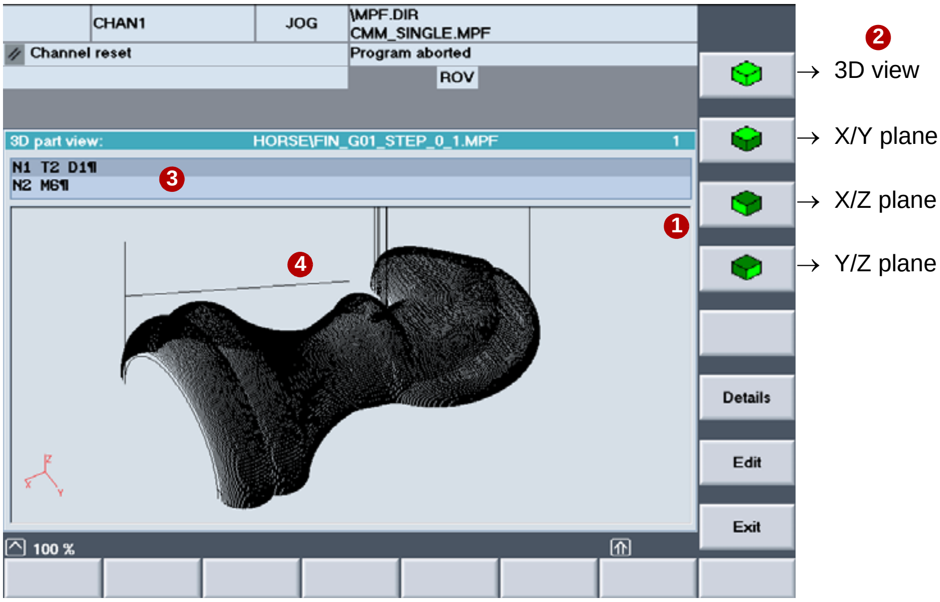
Please take care when modifying the swivel data. This affects the kinematics directly and if an error is made with regard to the correction value, this can result in damage to the machine during operation.

2.10 Quick View / Fast display

Function

The Quick View function on the SINUMERIK with PCU50 makes it possible to visualize part programs that contain G01 blocks. Program loops, polynomials, transformations and G02/03 blocks are not supported. Four views ❶ are available: 3D view ❷, X/Y plane, X/Z plane, Y/Z plane.

The two editor lines ❸ display the block currently highlighted in the graphic. Scrolling through the editor window automatically highlights the position ❹ in the graphic.



The following functions are also available

- Search for a specific block
- "Enlarge/Reduce" the picture detail
- Shift, rotate
- Measure the distance between two points
- Edit the NC part program displayed

Notes You can use Quick View (fast display) for the SINUMERIK with standard interface and for ShopMill. With the standard version, Quick View can be accessed via the **program manager**; with ShopMill, you can open Quick View in the **program editor**.

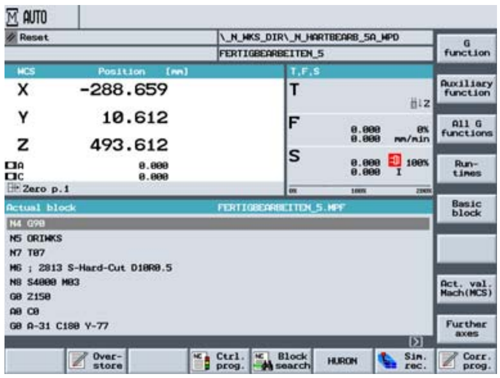
NOTE

Further information regarding Quick View can be found in the *Tool and Mold Making manual (3 axes)*.

2.11 ShopMill - Graphical interface

With SINUMERIK, the user-friendly ShopMill interface provides a viable alternative to the universal SINUMERIK standard DIN/ISO user interface.

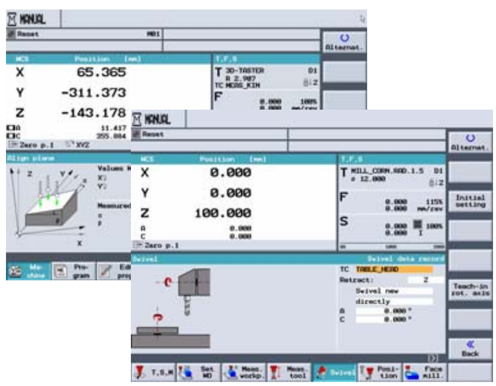
ShopMill in AUTOMATIC



ShopMill has been expanded by means of numerous mold-making and 5-axis functions for maximum operator comfort.

To cater for applications involving programs from external CAM systems or situations where existing programs are to be used, ShopMill features the full range of DIN/ISO programming, including Siemens standard cycles.

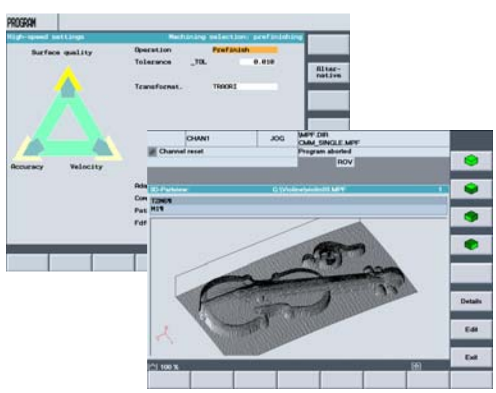
ShopMill, setup and swivel



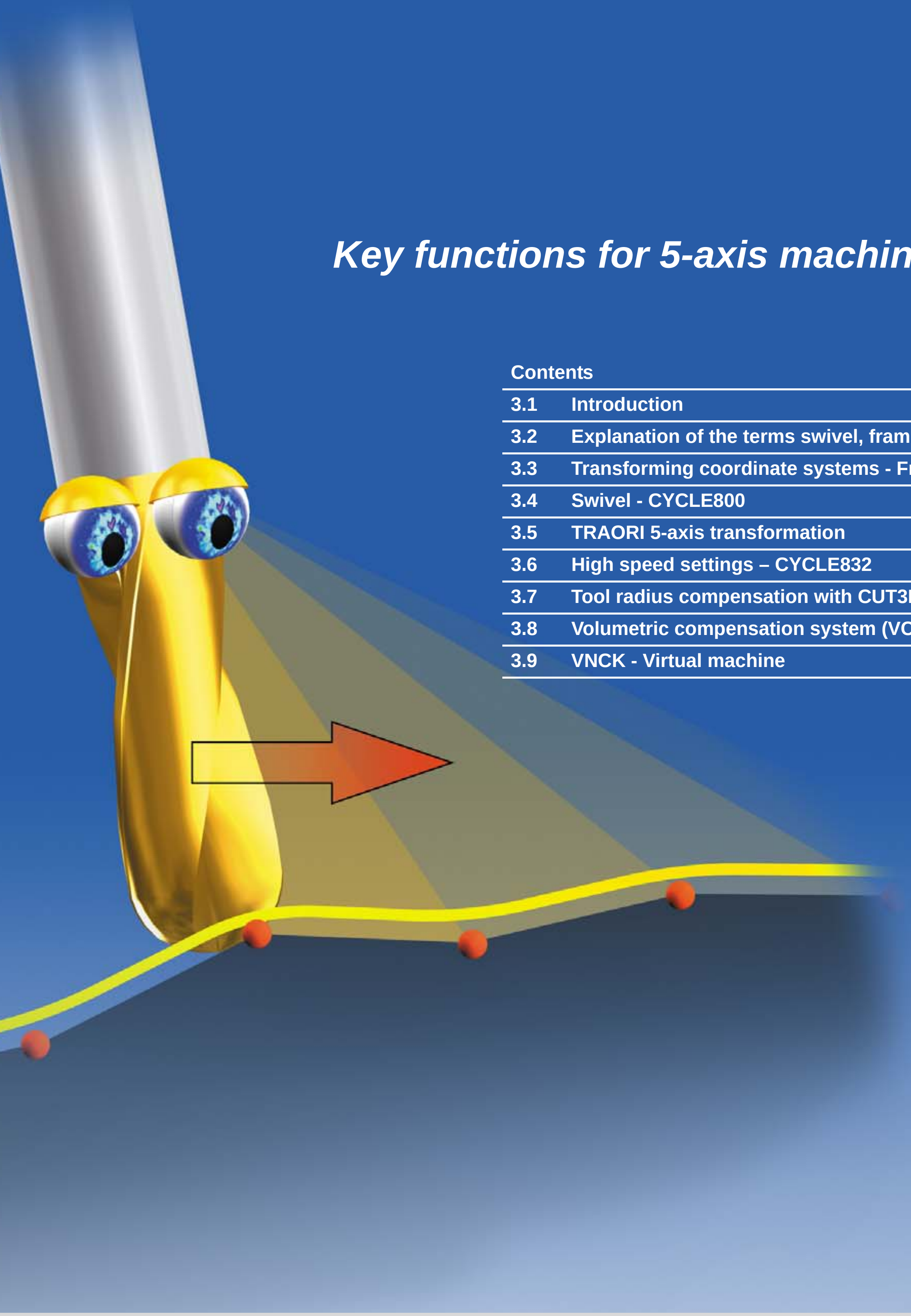
The complex programming normally associated with multi-axis machining is made much easier thanks to special cycles, such as CYCLE800 for swiveling, and straightforward transformation of rotated and swiveled coordinate systems based on the frame concept.

Practical measuring cycles are provided to facilitate setup and in-process measurements. These reduce setup times and make it possible to check dimensional accuracy on the machine.

ShopMill, mold making functions



To enable the perfect balance to be achieved for the respective HSC machining operation (roughing/finishing) in terms of speed, accuracy and surface quality, CYCLE832 brings together the necessary programming commands/G codes to ensure optimum milling results. If you need a fast overview of the programs in G1 blocks, you can use the Quick Viewer, which displays the programs graphically.



Key functions for 5-axis machining

Contents		Page

3

3.1 Introduction

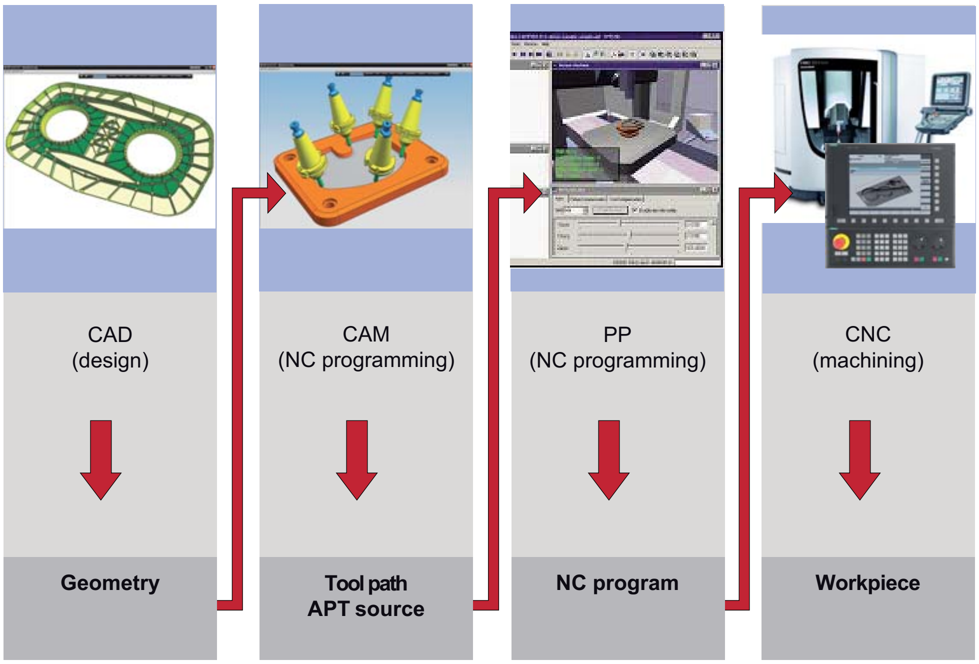
It is within the context of 5-axis programming in particular that the entire CAD/CAM/CNC process chain has a major role to play in terms of ensuring optimum results on the machine.

The CAD system generates the geometry of the desired workpiece. Based on this geometry file, the CAM system generates the corresponding machining strategy with the associated technology information.

The data format output from the CAM system is generally an APT or CL data file. This is converted into an executable NC code in the post processor.

The upstream post processor is of the utmost importance when it comes to using the capabilities and performance of SINUMERIK controls to the full.

The post processor should ensure that the higher-order functions of SINUMERIK controls (as described in this section) are activated in the best possible way. An overview of all the higher-order SINUMERIK functions is provided in the next few sections.



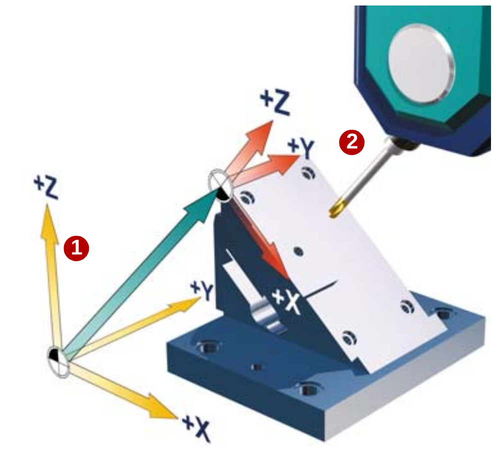
3.2 Explanation of the terms swivel, frames and TRAORI

The swivel (CYCLE800), frames and TRAORI features can be used to control rotary axes and angle the tool in relation to the machining surface. The main difference between these processes lies in whether the tool is oriented statically or dynamically. With the swivel and frames functions, the rotary axes are moved without the tool being active. The tool is oriented from a start position through to an end position and will then only work with the three linear axes. It is aligned in relation to the surface by means of a static process.

By contrast, TRAORI is a dynamic process. The rotary and linear axes can be moved simultaneously during machining. The tool can be continuously aligned with the surface while milling is in progress. All axes (rotary and linear axes) are interpolated at the same time. TRAORI is an option that is available for the 5-axis package.

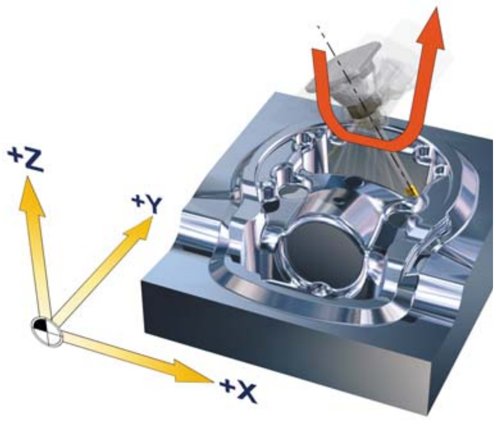
Frames only affect the coordinate system and only the coordinate system is modified. CYCLE800 takes the machine kinematics into account during swiveling, i.e. attention is paid to tool offsets and zero points. In the case of frames (e.g. ROT), these must be taken into account by the user.

Swivel



The tool is aligned with the machining surface by moving the rotary axes. This example involves rotation of rotary axis B ❶ and the tool is angled in relation to the XY plane ❷. Machining then takes place in this plane.

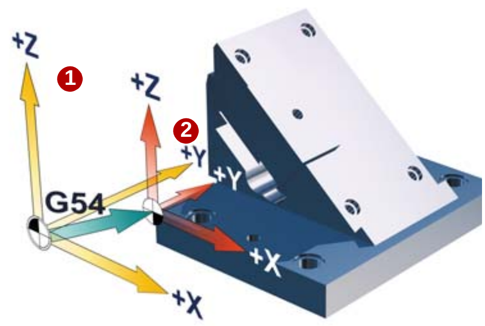
TRAORI



The tool is aligned with the machining surface dynamically during the milling process by means of linear and rotary axis interpolation. The tool length is taken into account and the kinematic compensating movements are initiated by the TRAORI function when the rotary axes are rotated.

3.3 Transforming coordinate systems - Frames

Coordinate systems



Machine coordinate system ❶ with reference point and work offset (G54, G55, etc.) are familiar terms.

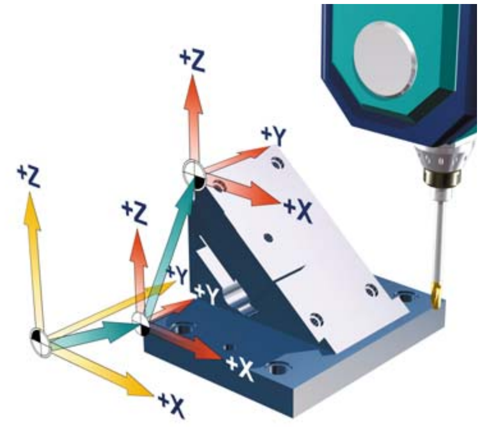
Using frames, the coordinate systems can be shifted, rotated, mirrored and scaled so that they are aligned with the workpiece surface. This allows the programming time and costs to be reduced to a minimum.

With frames, starting from the current workpiece coordinate system ❷, the position of a target coordinate system is defined by specifying coordinates and angles.

Possible frames include

- Basic frame (basic offset, G500)
- Settable frames (G54, G55, etc.)
- Programmable frames (TRANS, ROT, etc.)

Coordinate systems and traversing motion

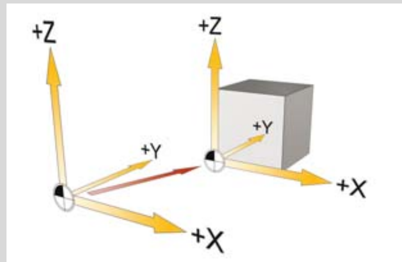
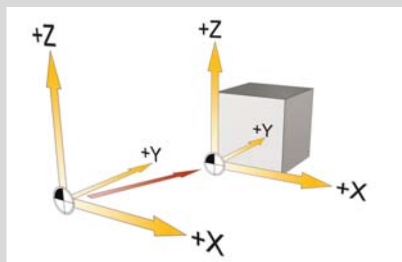
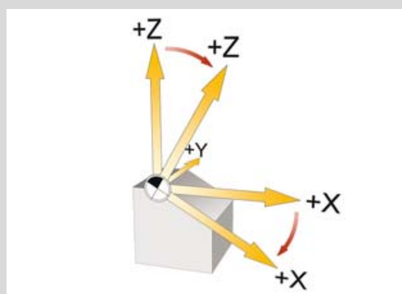
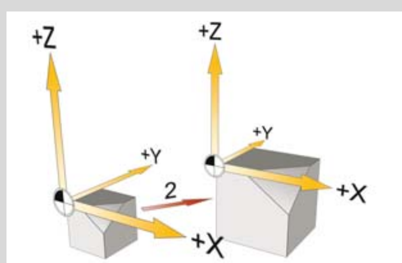
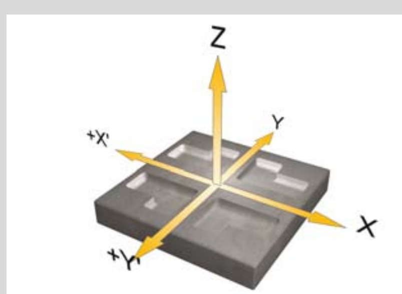


With a 5-axis machine, it is possible to machine surfaces that can be shifted and rotated in space as required.

The workpiece coordinate system merely has to be shifted using frames and then rotated into an inclined plane.

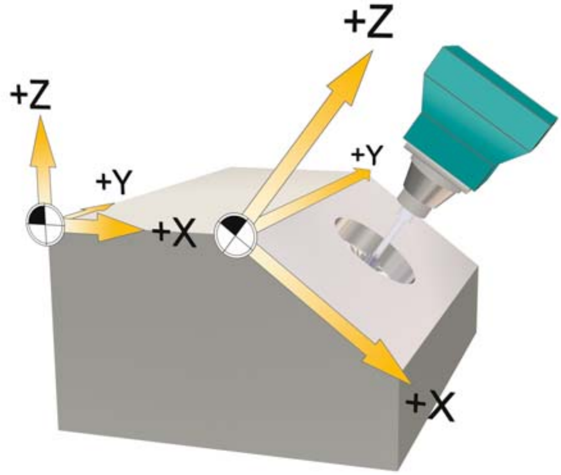
This is precisely why we need **FRAMES**. All subsequent traversing commands now relate to the new workpiece coordinate system shifted using frames.

Frames - Programming components

Offset (coarse)		Programmable with ■ TRANS (absolute offset) ■ ATRANS (incremental offset) ■ CTRANS (absolute offset via frame operators) Can be set with: ■ Frames (G54 to G599)
Offset (fine)		Programmable with ■ C-FINE Can be set with: ■ Frames (G54 to G599)
Rotation		Programmable with ■ ROT/ROTS ■ AROT/AROTS and ■ CROTS
Scaling		Programmable with ■ SCALE ■ ASCALE ■ CSCALE
Mirroring		Programmable with ■ MIRROR ■ AMIRROR ■ CMIRROR

3.4 Swivel - CYCLE800

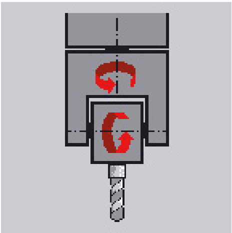
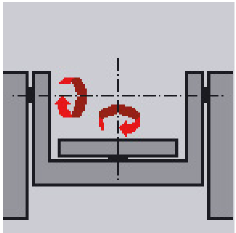
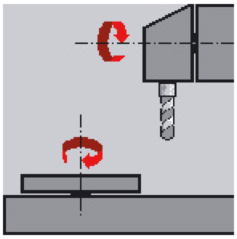
Function



You can use swivel heads or swivel tables to set up and machine inclined planes. The swivel feature is included as standard (i.e. is not an optional add-on). Swiveling is possible in both JOG and AUTOMATIC modes. Swivel operation parameter assignment and programming are facilitated by the clearly laid out graphics. You can either program all the swivel axes directly on the machine (A, B, C) or you can simply specify the rotations around the geometry axes (X, Y, Z) of the workpiece coordinate system as described in the relevant workpiece drawing. The rotation of the workpiece coordinate system in the program is then automatically converted to a rotation of the relevant swivel axis of the machine during machining.

The swivel axes are always rotated in such a way that the machining plane is perpendicular to the tool axis for machining. The machining plane then remains fixed during machining.
When the axes are swiveled, the active zero points and tool offsets are automatically converted for the swiveled state, resulting in a new coordinate system.

Machine kinematics

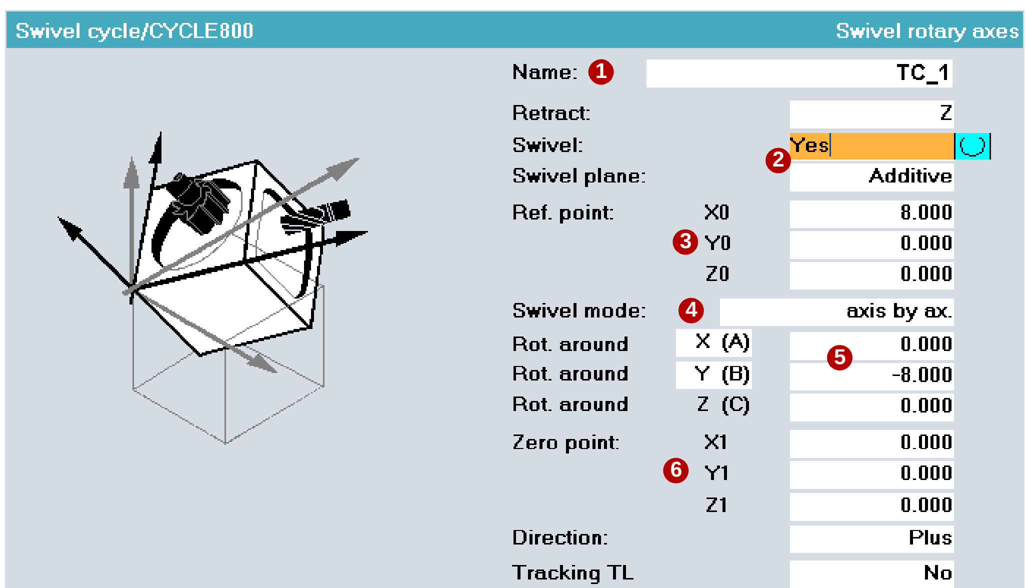
Swivel head (type T)	Swivel table (type P)	Swivel head + swivel table (type M)
Swiveling tool carrier	Swiveling workpiece holder	Mixed kinematics
		

Procedure for programming swivel motion and subsequent machining:

- Swivel the coordinate system into the plane to be machined.
- Program the machining process as usual in the X/Y plane.
- Swivel the coordinate system back to its original position.

Basic procedure for using the swivel cycle

- Call the **Swivel** function in the program.
- Select the name of the swivel data record (1).
- Select "yes" for swivel if you wish to perform a swivel movement. Select "new" as swivel movement if you wish to perform a new swivel movement, or "additive" if you wish to base the movement on a previous swivel movement (2).
- Specify the reference point before rotation (X0, Y0, Z0) (3).
- Select the "axis-by-axis" swivel mode directly using the projection angle or the space angle (4).
- Enter the angle through which the axis should swivel. In the axis-by-axis mode, you can enter the angle for each axis (5).
- Shift the zero point on the swiveled plane (6).



NOTE

You can program several swivel movements one after the other. A subsequent swivel movement can be based on a previous one (additive). This provides you with a really clear means of representing swiveling in the program code.

Note the machine manufacturer’s instructions! You can set the parameters that you want to be available using the CYCLE800 startup menu.

NOTE

For a detailed example and further information on programming the swivel cycle, please see the *Tool and Mold Making (3 axes)* manual. For further information on the swivel function, please refer to the additional documentation (See "Further information/documentation" on page 108.)

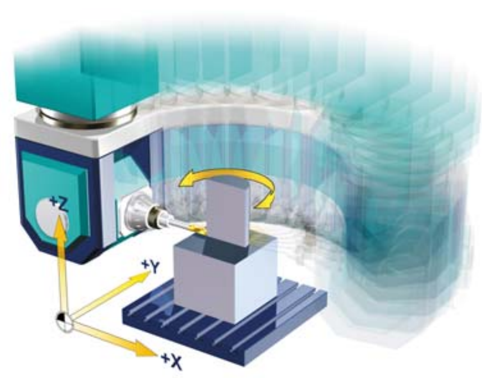
3.5 TRAORI 5-axis transformation

3.5.1 Kinematics and tool orientation

If you were to compare different kinematics with one another, this would very quickly reveal that different machine movements are required for the purpose of machining the same surface and that different NC programs therefore also need to be created. In this example, a lateral cylinder surface is to be machined.

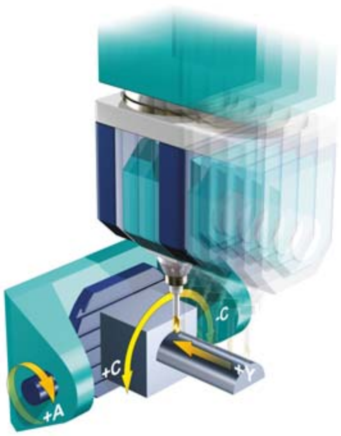
Motion sequence for different kinematics.

Motion sequence for head/head kinematics



For a rotation, a semi-circle must be defined in X/Y and the tool rotated around Z at the same time so that the tool remains perpendicular to the surface at all times.

Motion sequence for table/table



To achieve a rotation, the table swivels 90° around the A axis. In each case, the C axis rotates from +90° to -90° and the tool moves in the Y axis.

NOTE

As a rule, if a machine is being used in conjunction with SINUMERIK, you need not concern yourself with the machine kinematics during programming. All you need to concentrate on is the relative motion between the tool and workpiece. The control system will take care of everything else.

Effect of the tool length on the machine movement

Different tool length

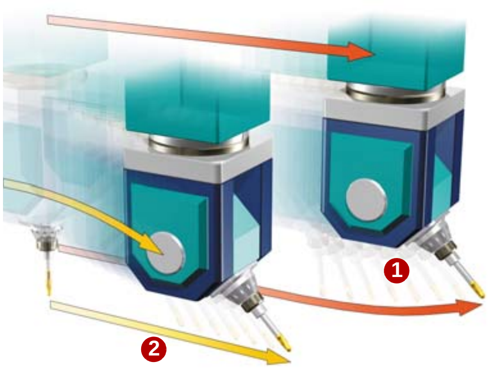
As you can see from the example, a longer tool will result in larger compensating movements for the kinematics.

If the program has been created using a CAM system, it will need to be recalculated whenever there is a change in the tool length.

Consequently, the control system needs to know that a calculated NC program is compatible with any tool length. This active tool length compensation is integrated into the SINUMERIK system and does not need to be given any consideration during programming.

Effect of tool orientation changes

Change in tool orientation



If the tool orientation is changed simultaneously during the traversing motion (e.g. the setting angle), the tool tip will be defined by a complex curve motion rather than a linear one (1).

However, in order to still be able to mill in a straight line, the tool holder must compensate this curve so that the tool tip performs the desired motion (2).

In the example with the yellow traversing motions (2), TRAORI compensation is active.

In order for all these requirements to be taken into consideration, a form of 5-axis transformation is needed that will transform the non-specific programs (in terms of kinematics) for the control and will take into account both the tool offsets and orientation. This is achieved by means of the SINUMERIK's TRAORI function.

3.5.2 5-axis transformation - From workpiece program to machine movement

Normally, NC programs are created in relation to the workpiece, i.e. all the tool positions relate to the workpiece coordinate system (WCS). In order for an NC program to be executed on the machine, the positions must be transformed into axis movements, i.e. converted into the machine coordinate system (MCS). The SINUMERIK is equipped with the TRAORI function to enable this kind of transformation.

What does the TRAORI function do?

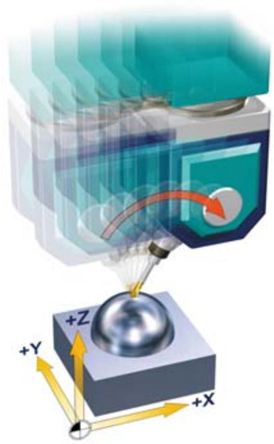
- If there is a change in tool orientation with stationary tool tip (traverse rotary axes only), the necessary compensating movements are calculated in X, Y, and Z
- Tool lengths taken into account
- Programmed feedrate made to relate to tool tip

Example involving TRAORI

Without TRAORI

A change in orientation affecting the B axis is programmed in the NC program without a tool tip traversing motion. The control simply rotates the axis; the tool tip does **not** remain **stationary**.

With TRAORI



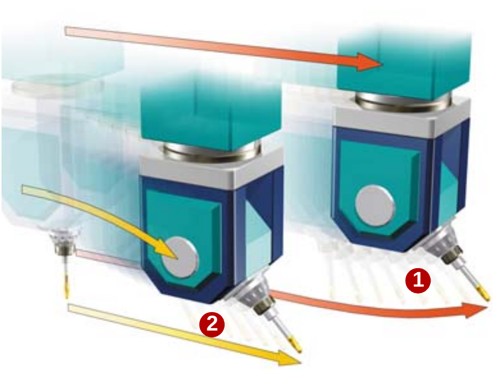
The control recognizes that it is merely a change in orientation that has been programmed; it ensures that the tool tip remains **stationary** and swivels the B axis.

3.5.3 TRAORI programming

There a numerous advantages to TRAORI programming. The program is independent of the tool length and machine kinematics, the feedrate relates to the tool tip and movements for compensating the movements of the rotary axes are performed automatically.

To achieve optimum cutting conditions when machining curved three-dimensional surfaces, the tool's setting angle must be variable. This calls for at least one or two rotary axes in addition to the three linear axes X, Y, and Z. The NC blocks are expanded by means of the orientation information, e.g. A3, B3, C3 or A, B, and C.

TRAORI active



When the transformation is enabled, the positional data (X, Y, Z) always relates to the tip of the tool, TCP. Changing the position of the rotary axes involved in the transformation causes so many compensating movements of the remaining machine axes that the position of the tool tip remains unchanged.

1 Without 5-axis transformation

2 With 5-axis transformation

Programming

TRAORI(n) **; Transformation activated**
TRAFOOF **; Transformation deactivated**

Explanation of the commands

TRAORI	Activates the first configured orientation transformation
TRAORI(n)	Activates the orientation transformation configured with n
n	The number of the transformation (n = 1 or 2), TRAORI(1) corresponds to TRAORI
TRAFOOF	Deactivate transformation

NOTE

TRAORI may reset the active work offset (WO), depending on the specific configuration. Therefore, as a precaution you should program the work offset after the TRAORI command.

3.5.4 Tool orientation

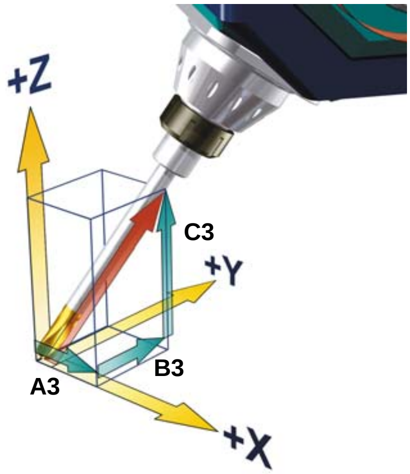
For the purpose of 5-axis simultaneous machining, the orientation of the tool needs to be defined in addition to the position setpoint of the machining point. There are a variety of methods that are commonly used to define the tool orientation. Generally speaking, 5-axis programs are created with the CAM system and the post processor is responsible for defining the type of orientation process used.

Direction vector (A3= B3= C3=)

Explanation of the commands

G1 X Y Z A3= B3= C3=	Programming of direction vector (recommended)

Direction vector programming



The components of the direction vector are programmed with A3, B3, and C3. The vector points towards the tool holder; the length of the vector is of no significance. Non-programmed vector components are set to zero.

N020 TRAORI
N035 G54
N040 G1 X0 Y0 Z0 A3=1 B3=1 C3=1 F10000
...

The example shows the tool in the position (0,0,0) as a diagonal of a cube (35.26° in relation to X-Y plane).

NOTE

The use of the direction vector is recommended. The accuracy level selected should be as high as possible. As far as 5-axis programs are concerned, practical experience has shown that good results can be achieved by using 5 decimal places for the linear axes and 6 for the direction vector.

NOTE

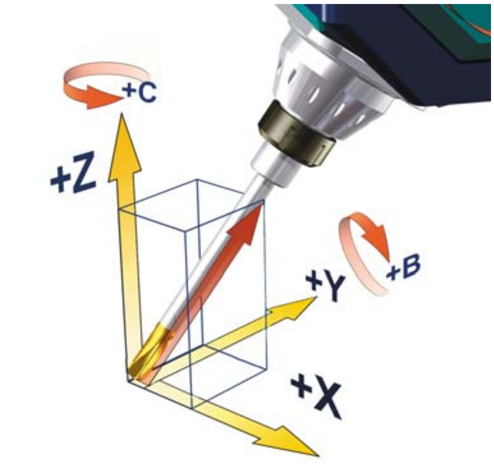
If you program C3=1, the tool will be aligned along the Z axis. This might prove useful, for example, if you need to remove a tool in the Z direction or retract it from a hole.

Rotary axis positions (A= B= C=)

Explanation of the commands

G1 X Y Z A B C	For programming the movements of rotary axes A, B, or C directly. The rotary axes are moved in synchronism with the tool path.

Rotary axis programming



The same position can be achieved as with tool orientation by specifying it on the basis of the rotary axis positions.

The position in the above example would be expressed as follows:

N020 TRAORI
N035 G54
N040 G1 X0 Y0 Z0 B=54.73561 C=45 F10000
...

The example shows the tool in the position (0,0,0) as a diagonal of a cube (35.26° in relation to X-Y plane).

NOTE

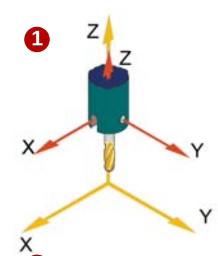
As regards the accuracy of the rotary axis positions, the same resolution can be used as for the linear axes. It is not necessary to increase the number of decimal places.

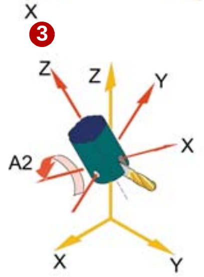
ORIEULER/ORIRPY (A2 = B2= C2=)

Explanation of the commands

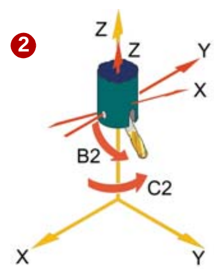
ORIEULER **ORIRPY**	Orientation programming on the basis of Euler angles (default) Orientation programming on the basis of RPY angles
G1 X Y Z A2= B2= C2=	Programming on the basis of Euler or RPY (Roll Pitch Yaw) angles, or G codes. Interpretation is defined by means of machine data. Programming in Euler or RPY angles using A2, B2, C2, or programming of the direction vector. The direction vector points from the tool tip toward the tool holder.

Programming in RPY angles





With A2=30°: rotated around



With C2=90°: rotated around Z axis; with B2=45°: rotated around

The values programmed with A2, B2, and C2 during orientation programming are interpreted as RPY angles (in degrees).

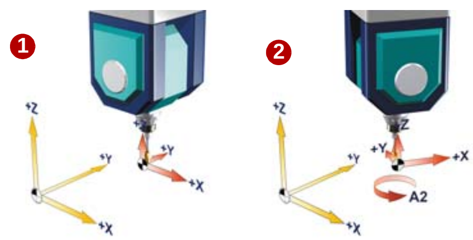
Starting from the normal ❶ position: The orientation vector is obtained by first rotating a vector in the Z direction around the Z axis with C2 ❷, then rotating it around the new Y axis with B2, ❸ and finally rotating it around the new X axis with A2 (**Z, Y', X**").

In contrast to Euler angle programming, in this case all three values have an impact on the orientation vector.

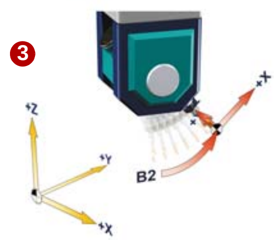
Example:

N020 TRAORI
N030 G54
N040 G1 X0 Y0 Z0 F10000
N050 A2=0 B2=0 C2=0
N060 A2=30 B2=45 C2=90
N070 ...
...

Programming in Euler angles



With A2=90°: rotated around Z axis



With B2=45°: rotated around
Y axis, which is rotated simultaneously

The values programmed with A2, B2, and C2 during orientation programming are interpreted as Euler angles (in degrees).

Starting from the normal ❶ position:
The orientation vector is obtained by first rotating a vector in the Z direction around the Z axis with A2 ❷, then rotating it around the new X axis with B2, ❸ and finally rotating it around the new Z axis with C2 (**Z, X', Z**'').

Example:

```
N020 TRAORI
N030 G54
N040 G1 X0 Y0 Z0 F10000
N050 A2=0 B2=0 C2=0
N060 A2=90 B2=45
N070 ...
```

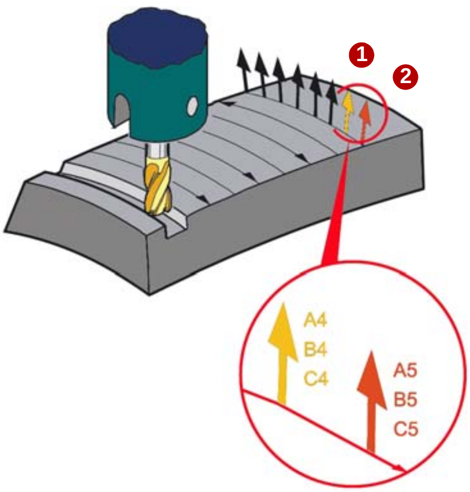
In this case, the value of C2 (rotation around Z axis) is irrelevant and does not need to be programmed.

Surface normal vector (A4= B4= C4=) (A5= B5= C5=)

Explanation of the commands

G1 X Y Z A4= B4= C4=	Programming the surface normal vector at start of block. This information is used by CUT3DF, for example, for the purpose of 5-axis machining. When used in conjunction with LEAD and TILT, this provides a further option for tool orientation programming. In this context, the LEAD and TILT angles relate to normal vectors A4 B4 C4. With ORIPATH, the orientation vectors are programmed relative to the normal vectors with LEAD and TILT.
G1 X Y Z A5= B5= C5=	Programming the surface normal vector at end of block.

Surface normal vector



The surface normal vector is perpendicular to the machining surface. It defines the path curvature. It is necessary for tool orientation with ORIPATH (LEAD, TILT) as well as for face radius correction with CUT3DF.
If only the start vector is programmed in a block (A4, B4, C4) ❶, this means that the programmed surface normal vector will remain constant throughout the entire block.

If only the end vector is programmed (A5, B5, C5) ❷, then large circle interpolation is performed between the end value of the previous block and the programmed end value.

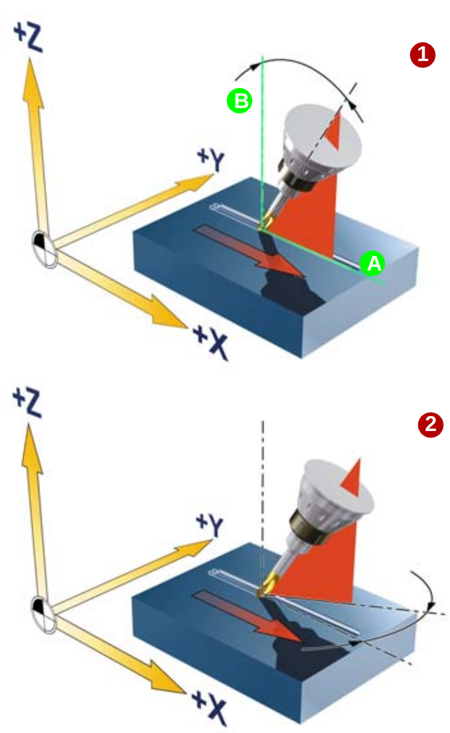
If both start and end vectors are programmed, interpolation according to the large circle principle is also performed between the two directions.

LEAD/TILT

Explanation of the commands

LEAD	**Lead angle** for programming tool orientation. Angle relative to the surface normal vector in the plane that arises from the path tangent and the surface normal vector.
TILT	**Tilt angle** for programming tool orientation. The TILT angle defines the rotation of the lead angle around the surface normal vector.

LEAD and TILT



Programming the tool orientation with LEAD and TILT in conjunction with ORIPATH.

The resulting tool orientation is determined from:

- Path tangent A
- Surface normal vector B
- Lead angle LEAD 1
- Tilt angle TILT at end of block 2

LEAD defines the angle between the surface normal and the new tool orientation, in the direction of the path tangent. If the tool is also rotated around the surface normal from this position, then this corresponds to the TILT angle.

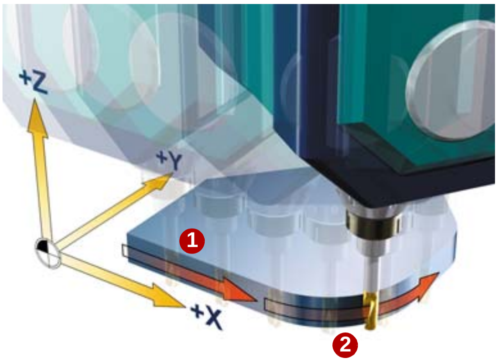
LEAD and TILT are also programmed if the tool is to adopt a fixed setting angle in relation to the machining direction, e.g. so that machining is not performed at the cutter center at cutting rate = 0.

```
N10 TRAORI
N15 G54
N20 ORIWKS
N30 ORIPATH
N40 CUT3DF
N50 START: ROT X=R20
N60 G0 X=260 Y0 A3=1 B3=0 C3=0
N70 G1 Z0 LEAD=5 TILT=10
N80 G41 X240.000 Y0.000 A5=1 B5=0.000
C5=0.000
```

3.5.5 Orientation interpolation and orientation reference

A 5-axis machine can apply any orientation to position the tool in relation to the workpiece. To get from one orientation to another, intermediate positions must be interpolated, as these are not specified in the NC program. These intermediate positions define the path from the start to the end orientation.

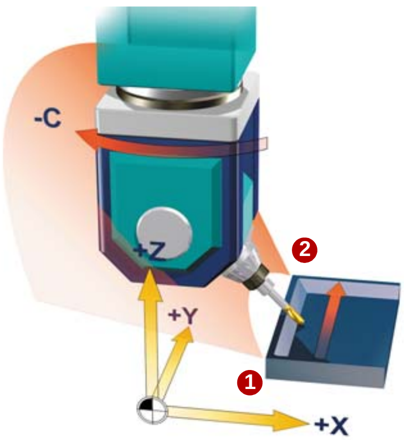
2D orientation



With 2D applications, the usual interpolation types are used to get from one position to another:

- Straight line G1 ❶
- Circle G2, G3 ❷
- Polynomial, B spline (w/o Fig.)

3D orientation



Various types of interpolation are used in 3D applications. In this example, which involves milling a pocket wall with an incline of 45°, the tool moves from position ❶ to position ❷. The A and C axes rotate in harmony during the movement so that the tool can be oriented along the edges of the pocket.

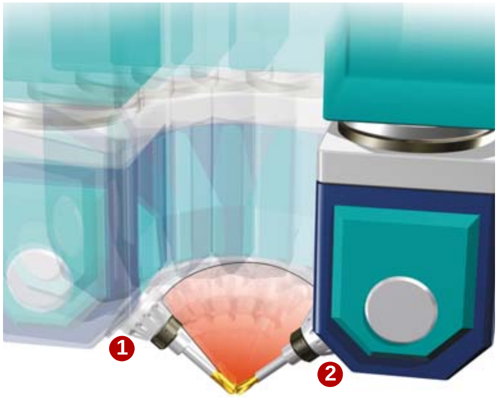
This kind of interpolation is known as large circle interpolation or vector interpolation.

The most common types of interpolation are explored below.

Orientation interpolation of the axes

Linear interpolation (ORIAXES)

Linear interpolation (ORIAXES)



With linear interpolation between a ❶ start and an ❷ end orientation, the necessary rotary axis movements are divided into equidistant sections.

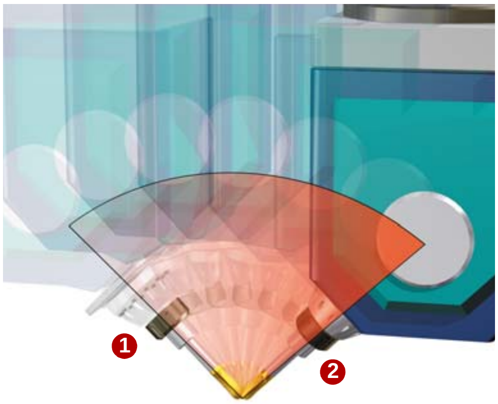
This results in a wall surface that is not flat when using circumferential milling to create inclined walls, for example.

CAM systems attempt to compensate for this effect by using sufficiently small interpolation steps. For optimum results, another type of interpolation (such as vector interpolation) should be used for these kinds of applications.

	Axis/Linear interpolation
ORIAXES	Linear interpolation of the machine axes or interpolation of the rotary axes using polynomials (with active POLY)

Vector interpolation (ORIVECT/ORIPLANE)

Vector interpolation, large circle interpolation (ORIVECT/ORIPLANE)



In the case of vector interpolation between a start and an end orientation, the path is interpolated so that the orientation vector runs in a plane created by the start and end vectors.

The angle between the start and end vectors is divided into equidistant steps at a constant velocity. This kind of orientation interpolation can be used, for example, to enable precise machining of sloping, flat walls in one block.

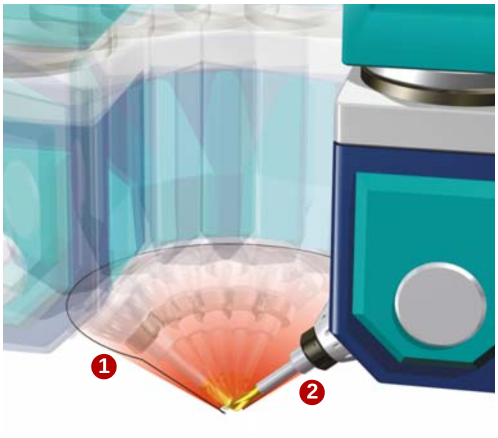
Applications:

- Structural components within the aviation industry
- Face milling of mold making applications

	Vector interpolation
ORIVECT	Interpolation of the orientation vector in a plane (large circle interpolation)
ORIPLANE	Interpolation in a plane (large circle interpolation), identical to ORIVECT

Cone surface interpolation (ORICONxx)

Cone surface interpolation (ORICONCW)

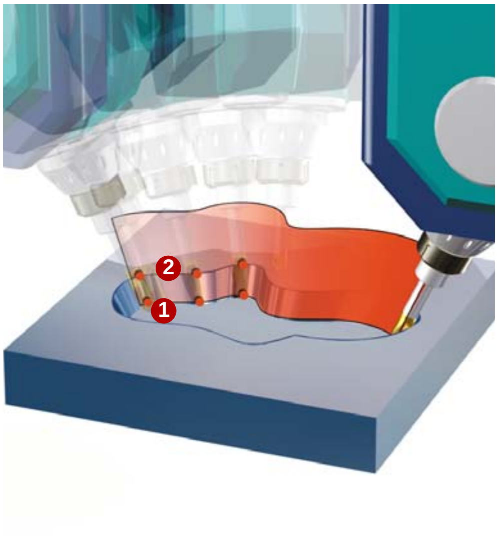


With cone surface interpolation, during reorientation the tool moves on a programmable peripheral surface of a cone located anywhere in space.

	Vector interpolation
ORICONCW	Interpolation on a peripheral surface of a cone in the clockwise direction
ORICONCCW	Interpolation on a peripheral surface of a cone in the counterclockwise direction. Also required in both cases: A3=… B3=… C3=... or XH=…, YH=…, ZH=… end orientation Cone's axis of rotation: A6, B6, C6 Opening angle: NUT=…
ORICONIO	Interpolation on a peripheral surface of a cone with an intermediate orientation specified via A7=… B7=…, C7=…. Also required: A3=… B3=… C3=... or XH=…, YH=…, ZH=… end orientation
ORICONTO	Interpolation on a peripheral surface of a cone with tangential transition. Also required: A3=… B3=… C3=... or XH=…, YH=…, ZH=… end orientation With POLY, PO[PHI] = …, PO[PSI]=… can also be programmed in these cases. This is a more generalized form of large circle interpolation, whereby polynomials are programmed for the lead and tilt angles. With cone interpolation, the polynomials have the same significance as with large circle interpolation for the given start and end orientations. The polynomials can be programmed with ORIVECT, ORIPLANE, ORICONCW, ORICONCCW, ORICONIO, ORICONTO.

Spline interpolation

Spline interpolation, curve interpolation (ORICURVE)



With spline interpolation, the motion of the orientation vector is defined by the tool tip's path 1 and the path of a second point on the tool 2 .

If, for example, you wanted to use circumferential milling to create an inclined surface, you would define the surface on which the milling cutter was to move by means of the two spline curves at the top and bottom ends of the tool (1, 2).

The advantage of this is that it enables you to define a wide variety of surfaces for machining with a high degree of precision.

Spline interpolation is the best type of interpolation available, but it is also the most involved and calls for special syntax in the NC program that must be supported by the CAM system.

	Spline/double spline interpolation
ORICURVE	Orientation interpolation with specification of the tool tip motion and that of a second point on the tool. The path of the second point is defined via XH=… YH=… ZH=…, in conjunction with BSPLINE as a control polygon with POLY as polynomial: PO[XH] = (xe, x2, x3, x4, x5) PO[YH] = (ye, y2, y3, y4, y5) PO[ZH] = (ze, z2, z3, z4, z5) If the BSPLINE or POLY additional information is omitted, straightforward linear interpolation will be performed accordingly between the start and the end orientation.

Orientation reference to the coordinate system (OIRMKS, ORIWKS)

ORIMKS **Tool orientation in the machine coordinate system.**

With **ORIMKS**, the programmed orientation relates to the coordinate system defined by the machine axes. Machine data is responsible for defining how orientation changes should be interpolated.

ORIWKS **Tool orientation in the workpiece coordinate system.**

With **ORIWKS**, the programmed orientation relates to the workpiece coordinate system, which may be rotated in relation to the machine coordinate system using a frame. Machine data is responsible for defining how orientation changes should be interpolated.

In the case of a 5-axis program, if it is not immediately obvious on which machine it is to run, ORIWKS must always be selected. Which movements the machine actually executes depends on the machine kinematics.

Orientation reference	
ORIMKS	The reference system for the orientation vector is the machine coordinate system.
ORIWKS	The reference system for the orientation vector is the workpiece coordinate system. Machine data is used to determine precisely what happens.

Further types of interpolation

	Path-related interpolation
ORIPATH	Tool orientation in relation to the path. A plane is created from the normal vector and path tangent that defines the meaning of LEAD and TILT at the end point. In other words, the path reference only applies to the definition of the end orientation vector. Large circle interpolation is performed between the start and end orientations. LEAD and TILT do not merely provide the lead and tilt angles. They have the following significance: LEAD defines the rotation in the plane created by the normal vector and path tangent. TILT then defines the rotation around the normal vector. In other words, they correspond to theta and phi in a sphere coordinate system, with the normal vector serving as the Z axis and the tangent as the X axis. **TIP:** If the path features corners, the path tangent will inevitably involve bends. These bends are reflected in the orientation on a 1:1 basis!

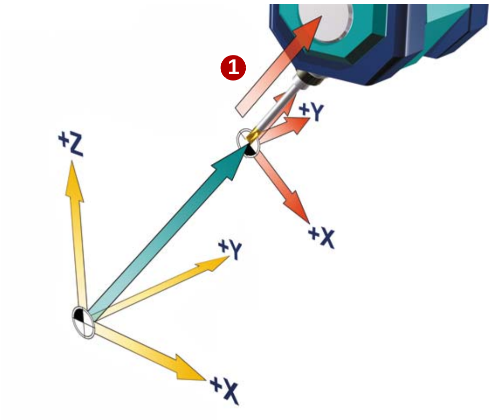
3.5.6 Example application involving TRAORI and frames

Example: TOROT - Retraction from a slanted hole

When 5-axis transformation is active, TOROT generates a frame whose Z axis coincides with the current tool orientation. This function can be used with a 5-axis program, for example, to retract the tool after a tool breakage without collision, simply by retracting the Z axis. Following tool orientation with TOROT, all the geometry axis movements programmed will relate to the frame generated as a result of this function.

Programming TOROT in MDI

N110	TRAORI	; Activate TRAFO
N120	TOROT	; Calculate and select retraction frame
N130	G1 G91 Z50 F500	; Retraction by 50 mm in Z direction in a straight line
N140	M17	; End of subprogram



A frame containing the current tool orientation in the Z direction is generated. 1 . This means that in JOG mode, the tool can be retracted from the workpiece in the Z direction.

As an alternative to traversing incrementally in MDI mode, in JOG mode you can use the direction key to enable retraction in the tool direction.

Notice:
Retraction is only possible in JOG mode if the machine has been configured accordingly (Z axis serves as the geometry axis).

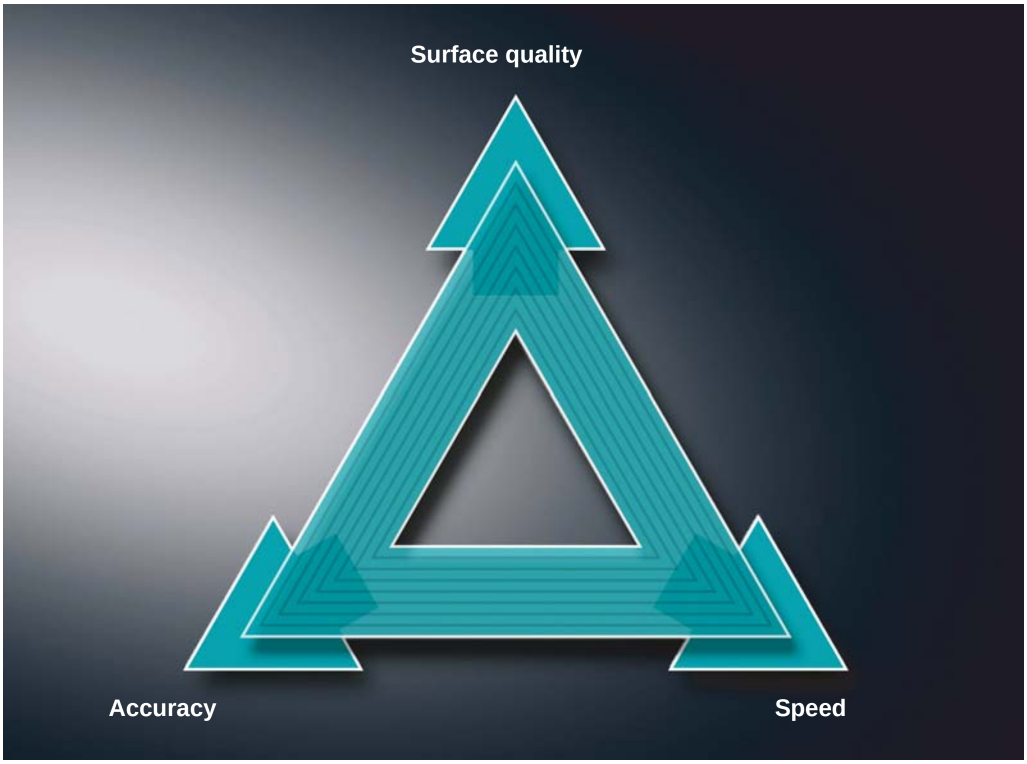
NOTE **TOROT must be deselected prior to the next program start: TOROTOF.**

3.6 High speed settings – CYCLE832

Application

You can influence the sequence of NC programs using SINUMERIK CYCLE832. It provides technological support for 3- and 5-axis machining in the high speed cutting (HSC) range. CYCLE832 brings together the key programming codes, or G codes, required for HSC.

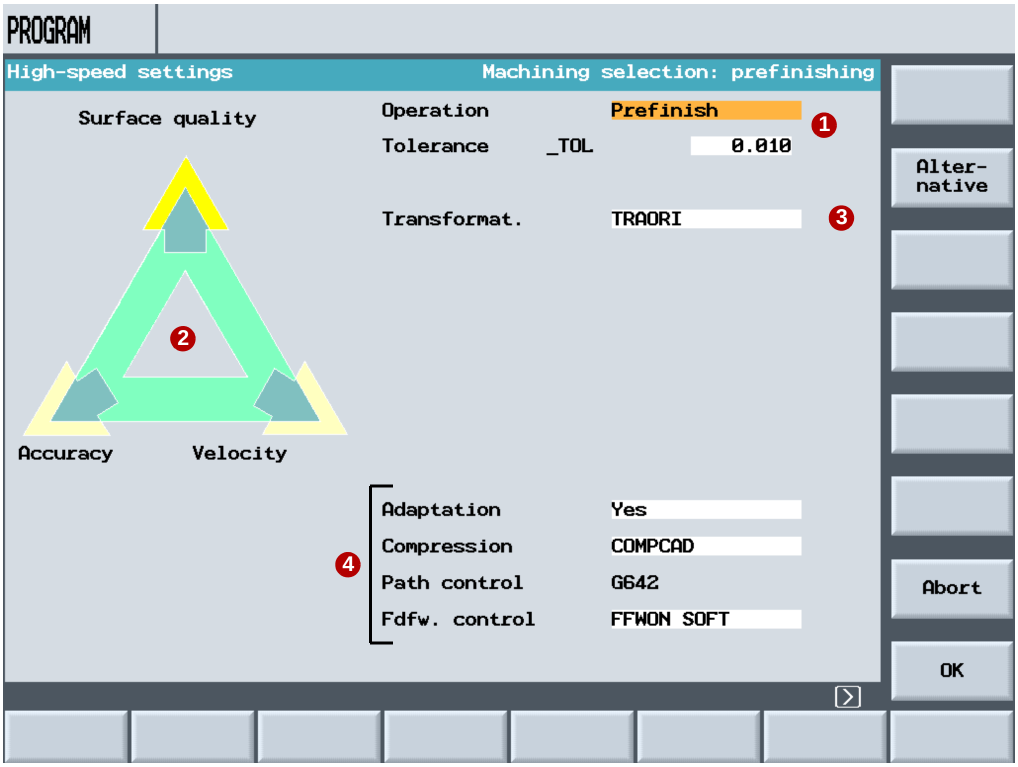
CYCLE832 can be set by the machine operator or, within the context of NC program generation, by the post processor or programmer. At machine level, the operator can also control how it is executed by changing certain parameters.



When executing CAM programs with very short NC blocks in the HSC range, the control needs to achieve high machining feedrates of >10 m/min. By applying different machining strategies, you can use CYCLE832 to fine-tune the program.

- When **roughing**, the emphasis is on speed due to the smoothing of the contour.
- When **finishing,** the emphasis is on surface quality and accuracy.

In both cases, specifying a tolerance ensures that the correct machining contour is achieved in order to obtain the desired surface quality and accuracy. Generally, a higher tolerance is selected for roughing than for finishing.

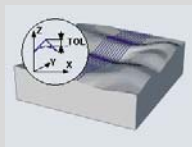
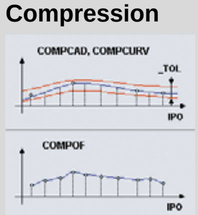
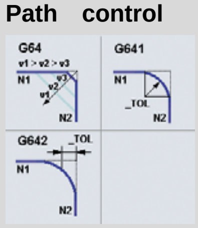
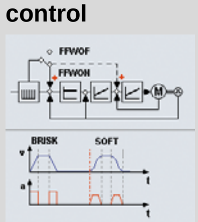


Depending on parameter selection ❶ the yellow arrows ❷ will point towards "Speed", "Surface quality" or in the direction "Accuracy".
TRAORI can be activated with simultaneous machining ❸, unless the CAM program has been calculated with resolved rotary axis positions.
The other options ❹ are activated by the machine manufacturer and are password-protected.

Parameters for the high speed setting cycle

In the **Machining** field, all the user has to do is choose between finishing, pre-finishing and roughing and enter a value in the **Tolerance** field. The values in all the other fields will have already been entered by the machine manufacturer. The machine manufacturer can enable the other fields using the **Adaptation** field (password-protected).

CYCLE832 parameters

Machining	■ Finishing (default) ■ Pre-finishing ■ Roughing ■ Deselection	Calling "Deselection" resets the modified machine/setting data to the value generated by the machine manufacturer.
Tolerance_tol. 	■ Chord tolerance (chord tolerance should be taken from the CAM system or weighted with a factor of 1.0 to 1.5)	Tolerance of linear/rotary axes, default settings: → 0.01 mm/ 0.08° (finishing) → 0.05 mm/ 0.4° (pre-finishing) → 0.1 mm/ 0.8° (roughing) → 0.1 mm/ 0.1° (deselection)
Transformation 	■ TRAORI ■ TRAORI(2) ■ No (default)	→ Activate 5-axis transformation → Activate machine-manufacturer-specific 5-axis transformation (optional) → Deactivate transformation. CAM programs with resolved rotary axis positions are supported.
Adaptation	■ Yes ■ No (default)	→ Subsequent fields can be modified → Subsequent fields are not visible. Activated by machine manufacturer
Compression 	■ COMPOF (default) ■ COMPCAD ■ COMPCURV ■ B SPLINE	→ Compressor off → Compressor on, constant acceleration rate for free-form surfaces → Jerk-free for circumferential milling → Spline interpolation
Path control 	■ G642 (default) ■ G641 ■ G64	→ Smoothing with single-axis tolerances → Programmable smoothing distance → Continuous-path mode In the case of an NC block compressor with COMPCAD, G642 is permanently selected.
Feedforward control 	■ FFWOF SOFT (default) ■ FFWON-SOFT ■ FFWOF-BRISK	→ Without feedforward control, with jerk limitation → With feedforward control, with jerk limitation → Without feedforward control, without jerk limitation In order for feedforward control (FFWON) and jerk limitation (SOFT) to be selected, the machine manufacturer must have optimized the control or the machining axes.

Notes

- CYCLE832 is based on the use of G1 blocks.
- In the event of changes, you should use the tolerance value specified in the CAM program as a guide. Tolerances that are lower than the one specified there are not practical.

CYCLE832 programming

Ideally, you should program CYCLE832 in the higher-level NC master program that then calls the geometry program. This means that you can apply the cycle to the complete geometry or - depending on the transparency of the CAM program - to individual program sections or free-form surfaces.

CYCLE832(0.01,112103)	Programming of the cycle with a tolerance such as 0.01 and transformation parameters.
CYCLE832()	Abbreviated program call. Corresponds to selecting the "Machining" "Deselection" input screen.
CYCLE832(0.01)	Abbreviated program call. Entry of the tolerance value. The active G commands are not changed in the cycle.

Programming example for CYCLE832

N10	T1 D1	; Activate TRAFO
N20	G54	; Select tool zero
N30	M3 S1200	; Clockwise spindle rotation and speed
N40	CYCLE832(0.05,112103)	; **Tolerance value 0.05**
		; From right to left
		; (Tol,decimal places 76543210)
		; 0 [3] = roughing, 1 [0]= no function
		; 2 [1] = TRAORI, 3 [2] = G642
		; 4 [1] = FFWON SOFT, 5 [1] = COMPCAD
		, 6 and 7 not used
N50	EXTCALL "CAM_ROUGH"	; Call subprogram CAM_ROUGH
N60	CYCLE832(0.005,112101)	; **Tolerance value 0.005**
		; From right to left
		; (Tol,decimal places 76543210)
		; 0 [1] = finishing, 1 [0]= no function
		; 2 [1] = TRAORI, 3 [2] = G642
		; 4 [1] = FFWON SOFT, 5 [1] = COMPCAD
		, 6 and 7 not used
N70	EXTCALL "CAM_FINISH"	; Call subprogram CAM_FINISH
N80	M03	

NOTE

Before the functions listed here can be used, the machine manufacturer must have optimized the CNC machine correctly.

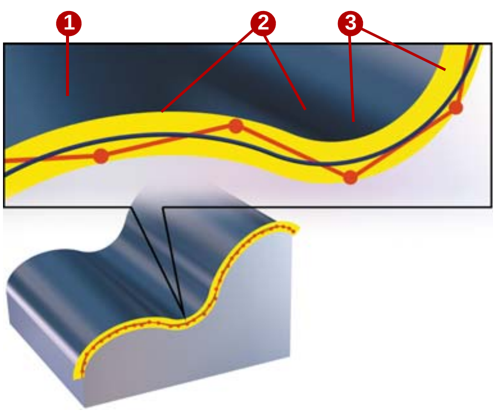
3.6.1 Compressor – COMPCAD

The compressor is ideally called in CYCLE832. If it is to be programmed separately, then proceed as described below.

Explanation of the commands

COMPOF	Compressor off
COMPCAD	Compressor on - surface quality and speed are further optimized. COMPCAD smoothes the points along the characteristic before approximation (B spline) and offers, at a high path velocity, the highest degree of accuracy with transitions that have a constant acceleration rate. Preferably used to mill **free-form surfaces (recommended).**
COMPCURV	Compressor on G1 blocks are approached using a polynomial. Block transitions are jerk-free. Preferably used for **circumferential milling**.

Operating principle of the compressor



In accordance with the specified tolerance band, ❶ the compressor takes a sequence of G1 commands, ❷ combines them and compresses them into a spline ❸, which can be directly executed by the control. A new contour is created whose characteristic lies within the specified tolerance range.

The compressor generates smooth paths and paths with constant curvature. The constant curvature results in a steady velocity and acceleration characteristic, meaning that the machine can run at higher speeds, thereby increasing productivity.

Programming

COMPOF
COMPCAD
COMPCURV

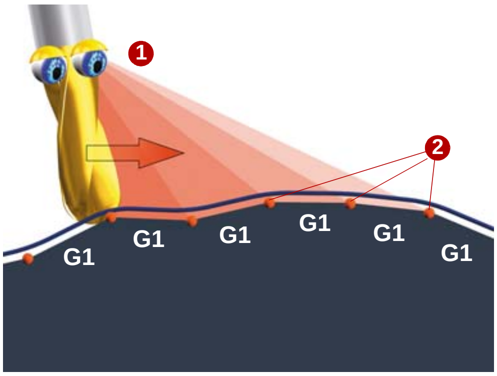
3.6.2 Continuous-path mode, Look ahead – G64, G641, G642

If you call continuous-path mode within CYCLE832, in the case of G642 for example, the tolerance value from CYCLE832 will correspond to the smoothing value for G642.

Explanation of the commands

G64	Continuous-path mode – **Look ahead** where the axis only brakes at corners
G642	Smoothing with axial tolerance **(recommended)** **Look ahead** with additional **corner rounding** corresponding to MD 33100 (machine data) The following applies to G642: There are 2 ways of specifying the tolerance: 1. Specifying individual axes or 2. Programming the retraction distance via ADIS Preferably used to mill **free-form surfaces**
G641	In the case of G641, the control inserts transition elements at contour transitions. You can use ADIS=… and ADISPOS=… to specify the extent to which the corners are rounded.
ADIS=	Smoothing distance for path functions G1, G2, G3. If no ADIS value is specified, the control will use the tolerance value from CYCLE832.
ADISPOS=	Smoothing distance for rapid traverse G0 (not suitable for free-form surfaces)

Using G64, G642



The objective of continuous-path mode is to increase the speed and harmonize the traversing behavior. For the continuous-path functions G64 etc. this is implemented by means of two functions.

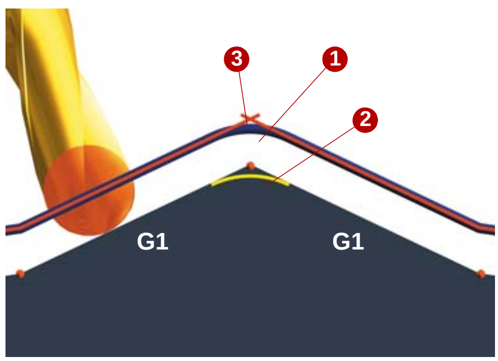
Look ahead - anticipatory velocity control 1

The control calculates several NC blocks in advance and determines a modal velocity profile. The way in which this velocity control is calculated can be set using the functions G64 etc.

Corner rounding ❷

The Look ahead function also means that the control system is able to round the corners that it detects. The programmed corner points are, therefore, not approached exactly. Sharp corners can be rounded.

These two functions mean that the contour is created with a uniform path velocity profile. This results in improved cutting conditions, increases the surface quality and reduces the machining time.



To round sharp corners ❸ , the continuous-path command **G642** forms transition elements ❶ , ❷ at the block boundaries. The continuous-path commands differ in terms of how they form these transition elements.

G642 inserts transition polynomials with constant curvature. These avoid step changes in acceleration at the block boundaries. We recommend G642 for free-form surface applications.

NOTE **We recommend G642 for free-form surface applications.**

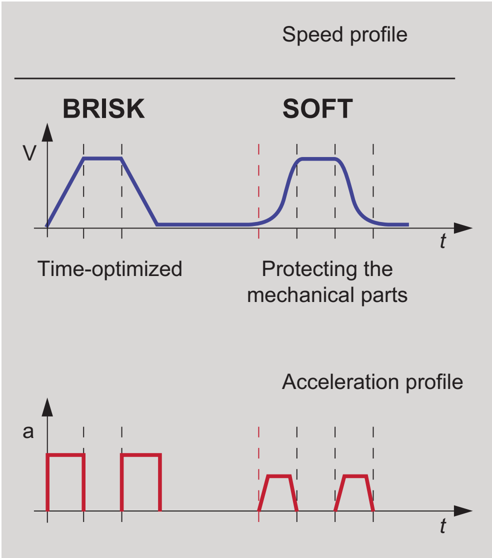
3.6.3 Feedforward control and jerk limitation – FFWON, SOFT, ...

Feedforward control and jerk limitation can only be called in CYCLE832 on a combined basis. This is because this combination offers ideal conditions for free-form surface milling. Of course, both of these functions can also be programmed separately.

Explanation of the commands

FFWON	Feedforward control "on"
FFWOF	Feedforward control "off"
BRISK	**Without jerk limitation** Path axes accelerate abruptly
SOFT	**With jerk limitation** Jerk-limited acceleration of the path axes Axial jerk limitation (maximum jerk in the machine data JOG_AND_POS_MAX_JERK (jog and positioning) and MAX_AX_JERK (continuous-path mode)

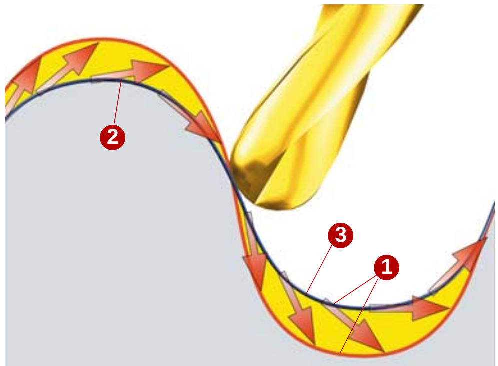
Jerk limitation function



To make acceleration as gentle on the machine as possible, the acceleration profile of the axes can be influenced using the commands **Soft** and **Brisk**. If **Soft** is activated, the acceleration behavior does not change abruptly but is increased in the form of a linear characteristic. This helps to protect the machine and improves the surface quality of workpieces, as much less machine resonance is generated.

BRISK:
Acceleration behavior: Path axes accelerate abruptly in accordance with the set machine data.
The axis slides accelerate at the maximum rate until the feedrate is achieved. BRISK enables time-optimized machining, but with jumps in the acceleration curve..

SOFT:
Acceleration behavior: Jerk-limited acceleration of the path axes.
The axis slides accelerate at a constant rate until the feedrate is achieved. As a result of the jerk-free acceleration characteristic, **SOFT** permits a higher path accuracy and less stress on the machine.

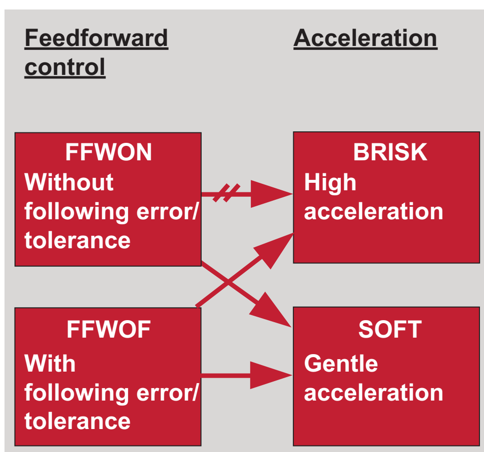


Feedforward control.

In cases where axes are not feedforward-controlled, the following error results in a contour error whose severity is determined by the velocity. ❶ . Generally, this will manifest itself in the form of a narrowing of the radius ❸ on curved contours. The following error depends on the servo gain factor that is set (dependent on mechanics) and the axis velocity.
The **FFWON** feedforward control function brings the velocity-dependent following error down towards zero during path traversal. Traversing with feedforward control permits higher path accuracy and thus improved machining results.

Recommendations

CYCLE832 includes the following combinations:



FFWON SOFT
The emphasis is on a high path accuracy. This is achieved using a soft form of velocity control that is essentially free from following errors.

FFWOF SOFT
The priority is not on achieving a high path accuracy.
Additional rounding is achieved by means of following errors. For use with older part programs/machines.

FFWON BRISK
Not recommended

FFWOF BRISK
For use when roughing and when maximum velocity is required.

Programming

FFWON/
FFWOF
BRISK
SOFT

3.6.4 Technology G groups

Using the "Technology" G group, the appropriate dynamic response can be activated on the machine for five varying technological machining operations. The dynamic values and G codes are configured and set by the machine manufacturer.

Five dynamic response settings are available in the Technology G code group:

- DYNORM for standard dynamic response
- DYNPOS for positioning mode, tapping
- DYNROUGH for roughing
- DYNSEMIFIN for finishing
- DYNFINISH for smooth-finishing

The G groups are switched and activated automatically when the machining method (e.g. finishing or roughing with CYCLE832) is selected. If you are not using CYCLE832, the G groups can also be programmed separately in the program.

Example program section without CYCLE832

```
N100   SOFT                           ; Activate jerk limitation
N110   COMPCAD                        ; Compressor on
N120   G642                           ; Smoothing
N130   ; Setting of the compressor tolerance for linear and rotary axes
N140   $MA_COMPRESS_POS_TOL [X] = 0.1
N150   $MA_COMPRESS_POS_TOL [Y] = 0.1
N160   $MA_COMPRESS_POS_TOL [Z] = 0.1
N170   $MA_COMPRESS_POS_TOL [A] = 0.8
N180   $MA_COMPRESS_POS_TOL [B] = 0.8
N190   FIFOCTRL                       ; Preprocessing memory control
N200   DYNROUGH                       ; DYNSEMIFIN or DYNFINISH
N210   TRAORI                         ; TRAORI on
N220   ORIAXES                        ; Orientation interpolation
N230   ORIWKS                         ; Workpiece coordinate system
N240   ...
```

NOTE

The dynamic values are already active in the block, in which the associated G code is programmed. Machining is not stopped.

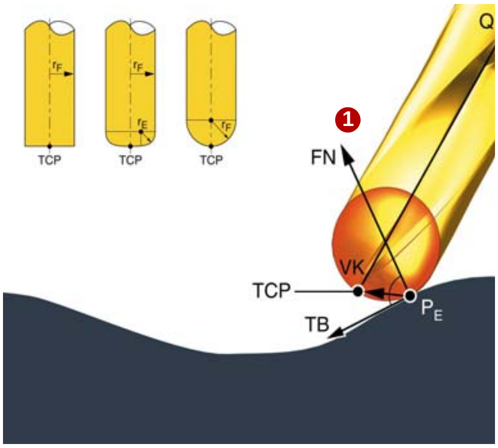
3.7 Tool radius compensation with CUT3D

The tool offset makes a CNC program independent of the tool radius. You will no doubt already be familiar with tool radius compensation in 2 ½ D applications. However, with 3D applications (particularly in the case of 5-axis milling), the situation is considerably more complex.

Influence of the tool radius when face milling with CUT3DF

When face milling with CUT3DF, not only must the milling cutter geometry be specified for radius compensation, but the compensation direction must also be known. The compensation direction is calculated from the surface normal, from the tool direction, and from the tool geometry.

Cherry compensation direction



For a 3D path, compensation must be performed perpendicular to the surface containing the path travelled.

In other words, the compensation direction is defined by the normal vector (FN) 1 , the plane of action. The figure contains the relevant geometry data.

The CAM must provide the surface normal in conjunction with every NC block. This information enables the control to perform radius compensation and to calculate the tool's point of action (P_E).

FN Surface normal
TCP Tool Center Point
P_E Point of action
TB Path tangent
VK Compensation vector

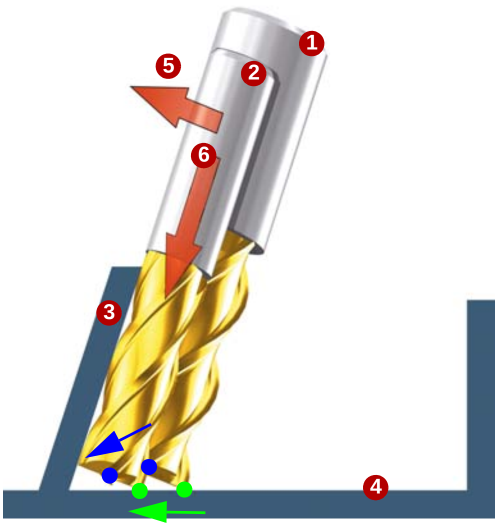
NOTE

Generally speaking, only minor changes in radius compared with the standard tool (the radius that the CAM program used for calculation purposes) can be compensated. A smaller milling cutter radius can be taken into account without any problems, but will result in a different peak-to-valley height. If the radius is larger, there is a risk of the tool colliding with the workpiece contour.

Influence of tool radius compensation with 5-axis circumferential milling, taking into account the limitation surface (CUT3DCC)

Imagine that a pocket needs to be created using a smaller milling cutter. The side wall is not perpendicular to the floor surface. The control supports tool radius compensation with a smaller tool. A typical application involving this function relates to structural components within the aviation industry.

Circumferential milling



1 Standard tool (tool from CAM)

2 Tool with smaller radius

3 Machining surface, inner surface

4 Limitation surface of pocket floor

5 Compensation in relation to machining surface

6 Compensation in relation to limitation surface

The control recognizes the fact that it is not just a question of compensating in the machining surface direction 5 , but also of making an adjustment in the tool direction 6 , so that the point of action (green) is at the same level as the pocket floor. This results in a shift in the TCP (blue) in the direction of the pocket floor.

Explanation of the commands

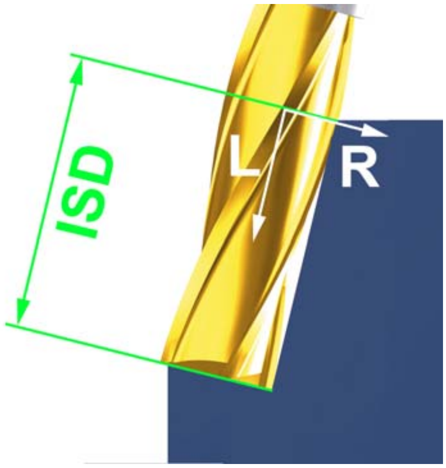
G40	Deactivation of all variants
G41	Activation for circumferential milling, compensation direction left
G42	Activation for circumferential milling, compensation direction right
G450	Circles at external corners (all compensation types)
G451	Intersection method at external corners (all compensation types)
2 ½D circumferential milling	
CUT2D	2 1/2D COMPENSATION with compensation plane determined using G17 – G19
CUT2DF	2 1/2D COMPENSATION with compensation plane determined using a frame
3D circumferential milling	
CUT3DC	Compensation perpendicular to path tangent and tool orientation

ORID	No changes in orientation in inserted circular blocks at external corners. Orientation motion is performed in the linear blocks.
ORIC	Travel path is extended by means of circles. The change in orientation is also performed proportionately in the circle.
Face milling	
CUT3DFS	Constant orientation (3-axis). Tool points in the Z direction of the coordinate system defined via G17 - G19. Frames do not have any effect.
CUT3DFF	Constant orientation (3-axis). Tool in Z direction of the coordinate system currently defined via the frame.
CUT3DF	5-axis with variable tool orientation
3D circumferential milling with limitation surface (combined circumferential/face milling)	
CUT3DCC	CNC program relates to the contour on the machining surface.
CUT3DCCD	The CNC program relates to the tool center point path.

Programming example for circumferential milling

A workpiece contour needs to be milled at the circumference. It is a question of programming from the top edge of the workpiece and the engaged length ISD is taken into account. In the example, compensation is performed to the right based on an ISD of 20.

Example involving CUT3DC



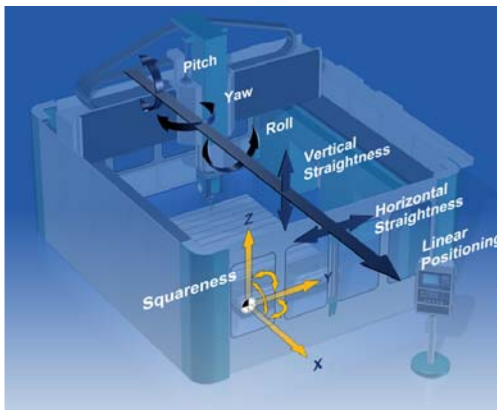

```
N10 A0 B0 C0 X0 Y0 Z0 F5000
N15 T1 D1 ISD=20
N20 ; Call tool and apply tool offset
N25 TRAORI
N30 ; Activate transformation
N35 CUT3DC
N40 ; 3D tool radius compensation
N45 G42 X10 Y10 G1
N50 ; Tool radius compensation and ISD selection
N55 X60
N60 A3=-1 B3=1 C3=1
N65 Y100
N70 ...
N90 G40
N95 ; Tool radius compensation and ISD deselection
N100 ...
```

3.8 Volumetric compensation system (VCS)

As far as the production of large workpieces is concerned, e.g. structural parts on gantry-type milling machines, it is very difficult to achieve the necessary level of machine positioning accuracy due to the large dimensions of the working area. Errors due to sagging, buckling, etc. in particular can only be remedied mechanically with a great deal of effort.

The volumetric compensation system for the SINUMERIK is a tool for correcting geometric distortions of the machine mechanics. It reduces the impact of machine errors on the tool center point (TCP) systematically, thereby increasing the accuracy of the machine.

Types of VCS compensation



Below are some examples of the kinds of error source that can be compensated:

- Linear position deviation
- Straightness of axes
- Unintentional axis rotations
- Roll, pitch, and yaw
- Squareness of axes in relation to one another
- Tool orientation errors involving swivel heads

As part of this process, the machine errors are detected on the basis of measurements and an error file is generated. VCS then implements the compensation values automatically. To increase the level of accuracy, calibration and testing with the compensation values can be performed using an iterative method. VCS automatically compensates the detected errors in conjunction with TRAORI.

NOTE

As regards the VCS commissioning process and machine calibration, please contact your machine manufacturer.

3.9 VNCK - Virtual machine

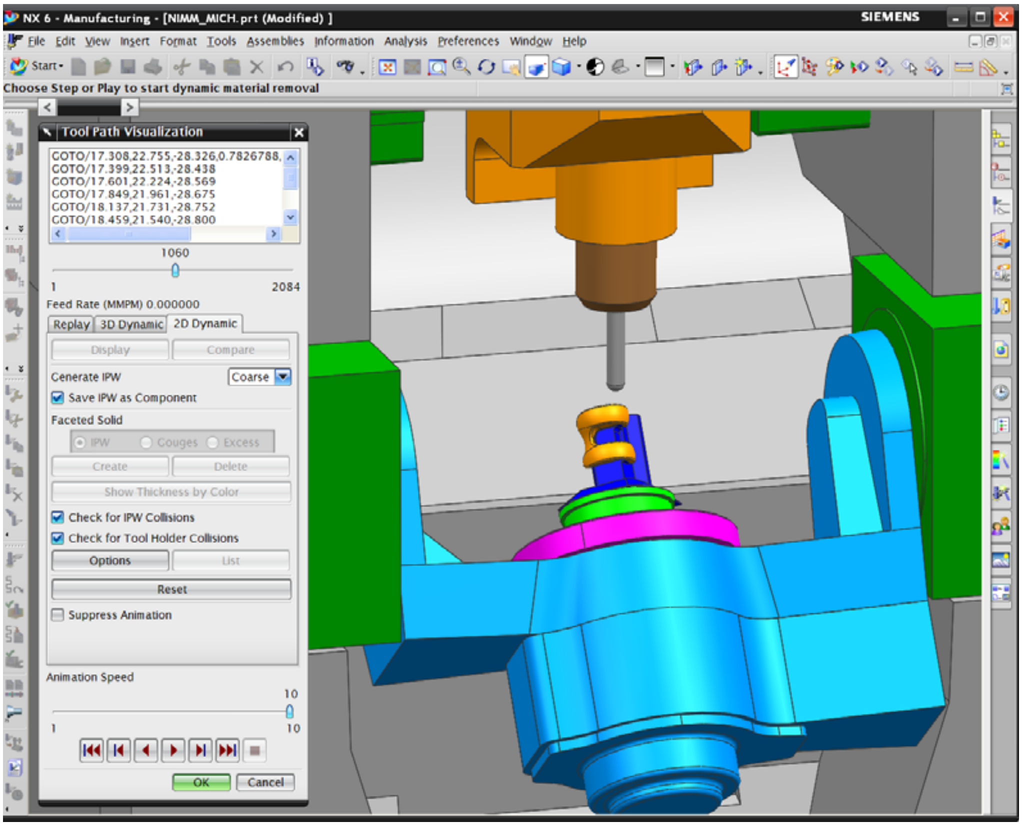
Due to the level of complexity and machining time associated with 5-axis milled workpieces, a deliberate effort is made to check that the programs are free of errors before actual production commences. To ensure that the data obtained is as realistic as possible, virtual models of the machine and control are created and simulated. Siemens provides the following basic module for this purpose:

- the virtual NC kernel (VNCK)

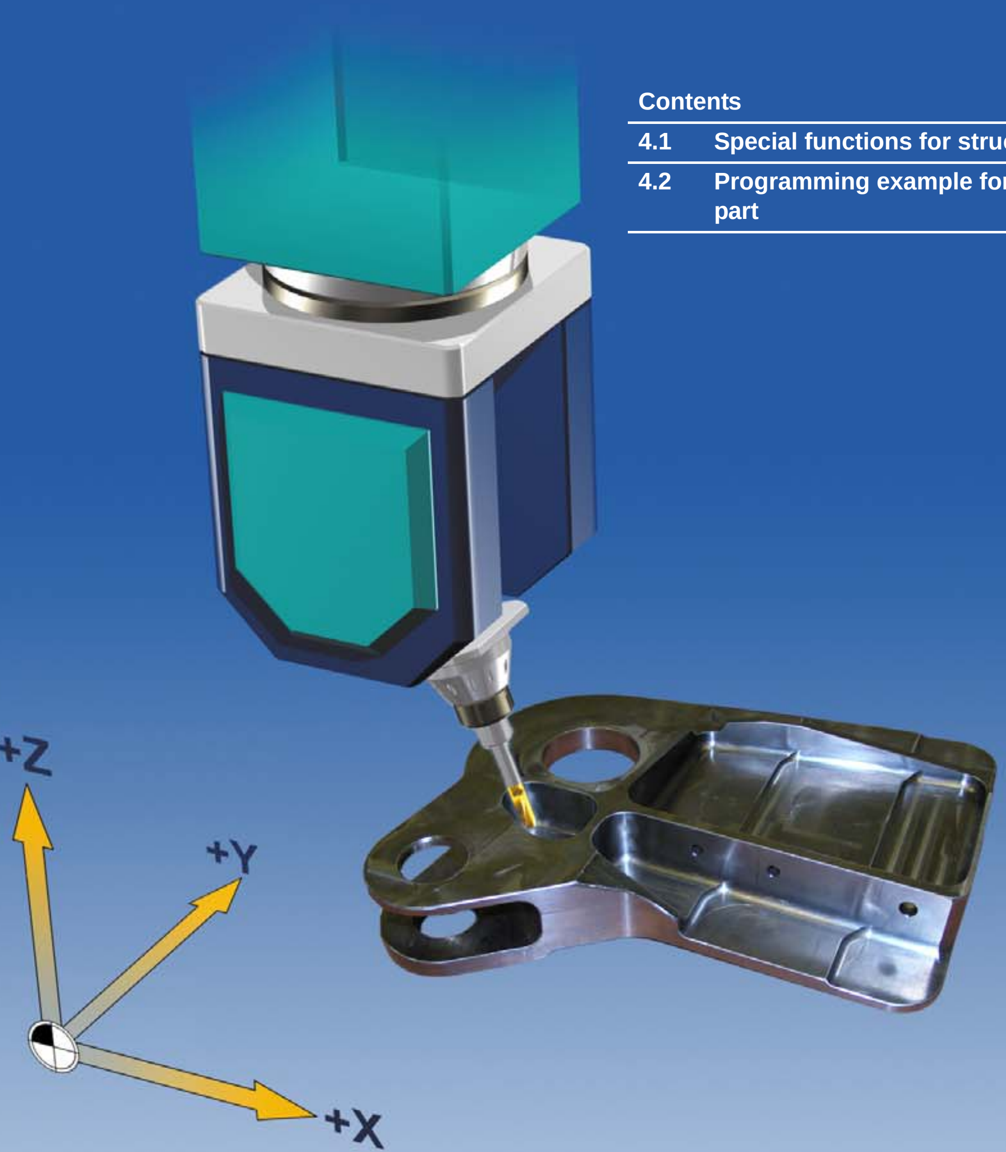
Using these basic modules and other components, such as the CAD data for the real machine, the machine manufacturer or CAM system manufacturer can create a virtual machine that resembles the real machine as closely as possible.

Using the virtual machine in conjunction with the SIEMENS VNCK offers many advantages:

- Programming errors are detected immediately.
- Program simulation with calculation of the actual time so that production effort can be estimated more easily.
- Collision checking with actual tools, chucking devices, and machine geometries.
- While the current production process is underway, the workpiece can be programmed, optimized and then implemented on the machine immediately.
- Shorter setup times.
- Can be used for training and instruction. New machines can be programmed without any risk.



Aerospace, structural parts



Contents

4

4.1 Special functions for structural parts

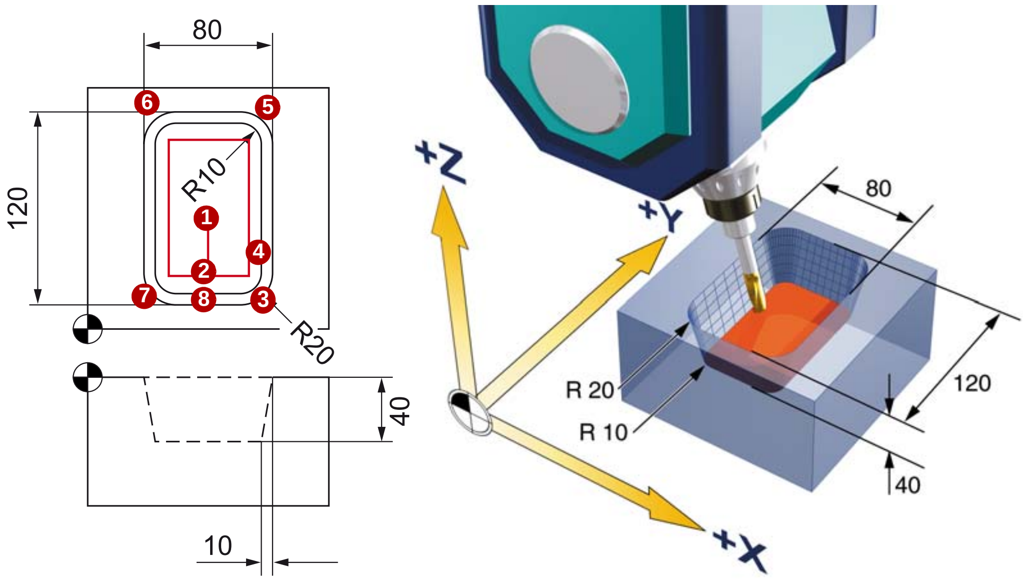
Structural parts are frequently used within the aircraft industry and take the form of load-bearing aircraft parts, e.g. the parts used to assemble the fuselage or wings. One of the key features of structural parts is the way in which they use only a small amount of material due to their geometry, but offer increased load-bearing capacity. For safety reasons, structural parts are milled from solid material and stock removal rates of up to 97% are by no means unusual. As a result, special functions are required throughout the entire machining process.

Key functions for machining structural parts:

- VNCK,
  because the high level of material usage calls for a form of simulation that resembles the control exactly.
- VCS,
  as maximum accuracy is required in conjunction with extremely large workpieces.
- CYCLE832
  Easy preselection of the key machine settings for roughing and finishing.
- ORIVECT,
  as this is the only kind of orientation interpolation that will ensure the necessary precision when creating inclined walls.
- 3D tool radius compensation,
  because this even allows the use of reground tools without having to rebuild the NC program.
- Integrated process chain from generation in CAD through to execution on the CNC.

4.2 Programming example for the pocket on a structural part

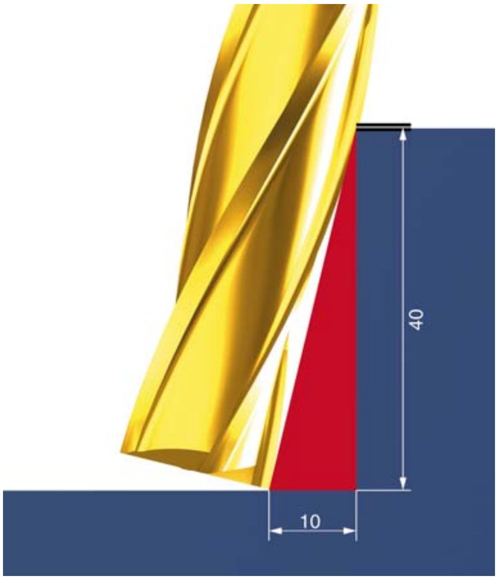
Inclined inner and outer walls are a typical feature of structural parts. The most effective way to create these is to use the circumferential milling technique. The following programming example relates to the milling of a pocket with inclined walls. It illustrates the functions required to achieve this, such as orientation interpolation and tool radius compensation, by showing what actually happens in practice.




```
N110   TRAORI                           ; Activate TRAFO
N120   G54                              ; Select tool zero
N130   TRANS X80 Y80                    ; Shift tool zero to center of pocket!
N140   AROT Z                           ; (Rotate pocket if required)
N150   ORIWKS                           ; Tool orientation in WCS
N160   ORIVECT                          ; Large circle interpolation of orientation
N170   CUT3DC                           ; 3D tool radius compensation (TRC)
N180   ISD=0                            ; Insertion depth of tool = 0
                                        ; The contour has been programmed on the pocket floor,
                                        ; not on the workpiece surface
                                        ; (in this case, ISD = 41.231)
                                        ; (see also note at the end of the program)
N190   G0 X0 Y-40 Z-39                  ; Approach path
N200   G1 G41 X0 Y-50 Z-40 A3=0 B3= - 10 C3=40
```

```
N205                              ; As the contour is being approached,
                                  ; the orientation changes.
                                  ; Selection of TRC and approach 1st machining position
                                  with
                                  ; required orientation.
                                  ; The orientation vector components
                                  ; can be taken directly from the
                                  ; drawing.
N210   X20                        ; 1st machining step. Approach the corner.
N220   ORICONCCW                  ; Selection of cone surface interpolation for
                                  ; the type of orientation interpolation
N230   A6=0 B6=0 C6=1             ; Definition of cone axis (lies parallel to the
                                  ; Z axis of the WCS).
                                  ; Cone defined as perpendicular to the
                                  ; Z axis.
N240   G3 X30 Y-40 CR=10          ; Rounding of the pocket with radius programming
N250   ORIVECT                    ; Large circle interpolation
N260   G1 Y40                     ; Machining steps repeated from this point
N270   ORICONCCW
N280   A6=0 B6=0 C6=1
N290   G3 X20 Y50 CR=10 A3=0 B3=10 C3=40
N300   ORIVECT
N310   G1 X-20
N320   ORICONCCW
N330   A6=0 B6=0 C6=1
N340   G3 X-30 Y40 CR=10 A3= - 10 B3=0 C3=40
N350   ORIVECT
N360   G1 Y-40
N370   ORICONCCW
N380   A6=0 B6=0 C6=1
N390   G3 X-20 Y-50 CR=10 A3=0 B3= - 10 C3=40
N400   ORIVECT
N410   G1 X0
N420   G40 Y-40 Z-39 A3=0 B3=0 C3=1
N425                              ; Deselection of tool radius compensation
N430   G0 Z100                    ; Retraction
N440   TRAFOOF                    ; Deactivate TRAFO (if necessary)
```

ISD based on pocket floor, workpiece surface



The pocket contour can be programmed on the basis of the pocket floor, in which case the ISD is 0.

Alternatively, the contour can be programmed in relation to the workpiece surface and in this case the insertion depth ISD is 41.231 (length of pocket wall). The radii will need to be adjusted.

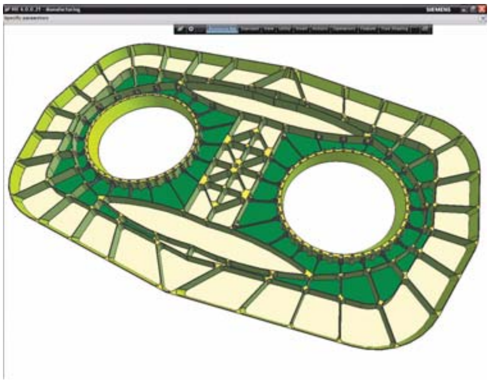
The adjustment can be calculated using Pythagoras' theorem.

$$ISD = \sqrt{40^2 + 10^2} = 41,231$$

Special functions in the CAM system

CAM systems provide support for special workflows. Within the context of 5-axis machining in particular, they provide methods that meet the demands associated with the programming of structural parts, thereby creating the perfect conditions for ensuring optimum results on the machine.

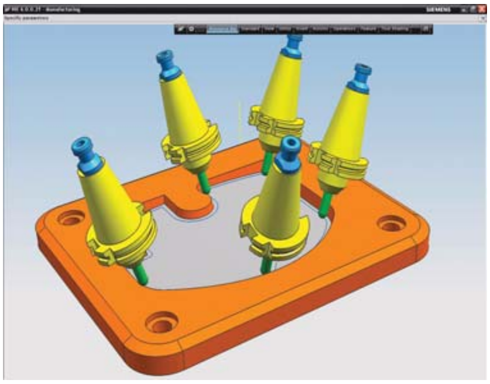
Support for fins and pockets



CAM systems enable roughing and finishing to be performed quickly and precisely on highly complex parts such as those typical of the aircraft industry.

The geometry selection process (e.g. for frequently occurring parts such as pockets and fins) has been highly automated to enable quick and easy programming.

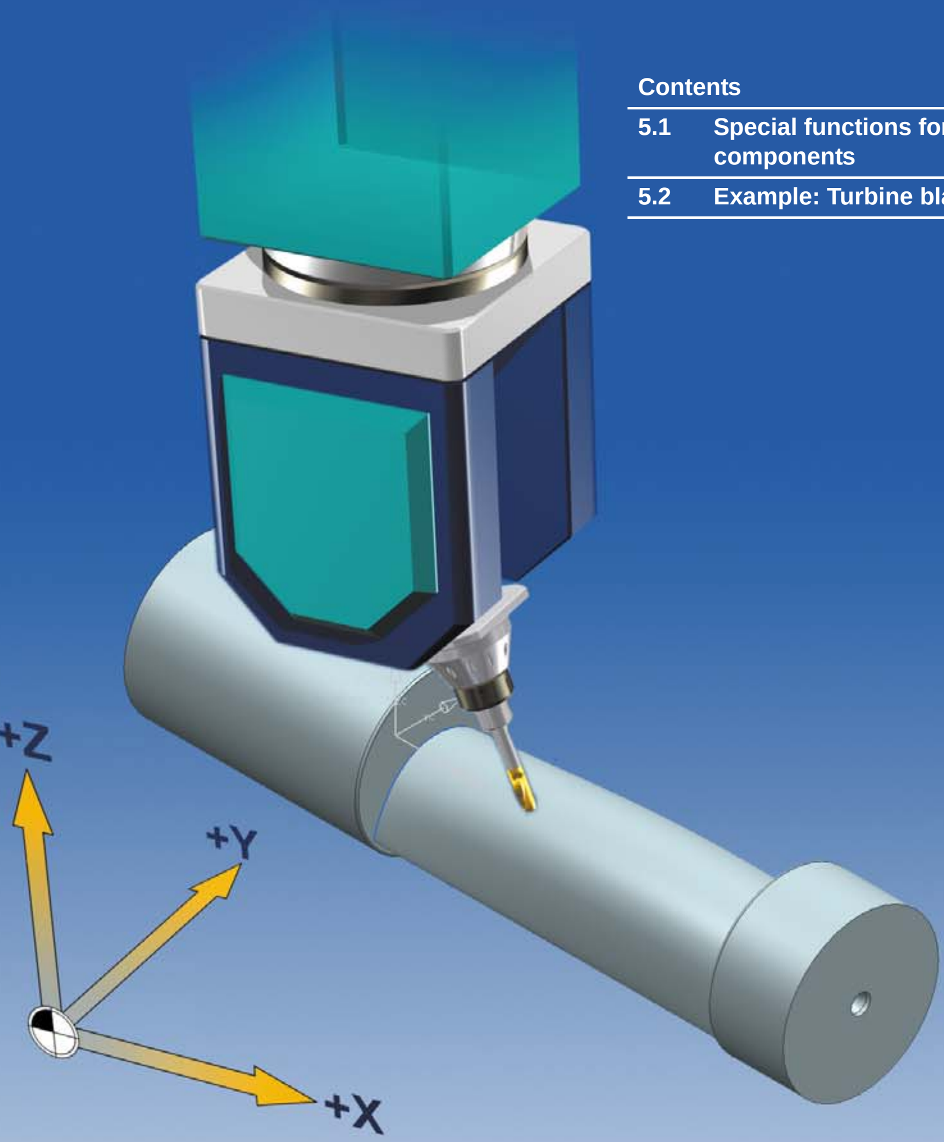
Tools set at an angle during profile milling



Automatic profile milling with variable axes speeds up the programming process. CAM systems offer a range of automatic tool position and tool axis settings for 5-axis milling on walls and other profiles.

The tool paths may involve following pocket floors, the edges of walls or offsets.

Driving gear and turbine components



Contents

5

5.1 Special functions for driving gear and turbine components

Turbine impellers or blades are subject to the toughest of requirements in terms of surface quality and contour accuracy. As a result, the NC is expected to process large quantities of data within a very short space of time. Even the smallest jumps in deceleration and acceleration can result in surface defects (e.g. chatter marks).

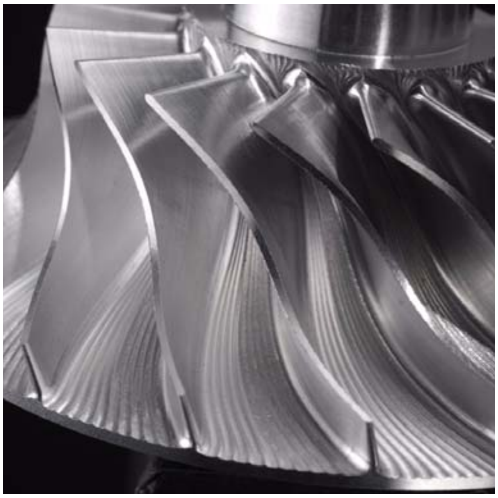
Turbine blades are often made from high-strength nickel alloys or titanium, which means that suitable methods have to be employed. This makes SINUMERIK the perfect solution, as it is a complete package offering a highly dynamic drive combined with a control system.

Key functions for machining parts for driving gear and turbine technology:

- High speed settings CYCLE832,
  as optimum data compression within the tolerance band, combined with feedforward control and jerk limitation, ensures the required surface quality and contour accuracy.
- Spline interpolation for hobbing (face/circumferential milling) impeller blades.
- TRAORI,
  for 5-axis transformation that is independent of the kinematics.
- Integrated process chain from generation in CAD through to execution on the CNC.

.

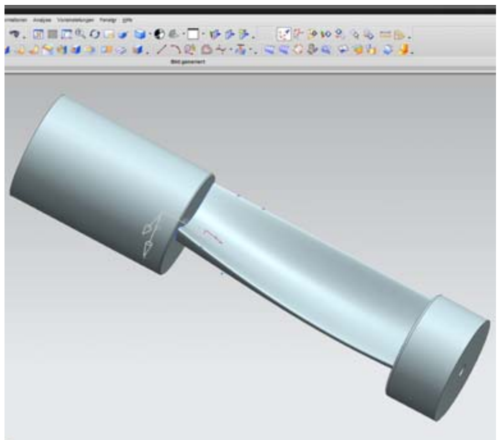
Impeller



5.2 Example: Turbine blade

This example relates to the milling of a turbine blade. The blade is modeled using a CAD/CAM system.

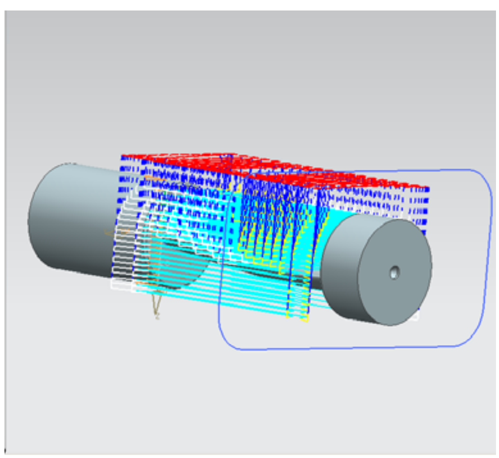
Turbine blade



At the modeling stage, it is essential to ensure that the machining strategies take account of the chucking conditions that will apply during production.

As a general rule, the contours of turbine blades are milled in a helical path, i.e. a full rotation is performed around the Z axis using a suitable chucking device.

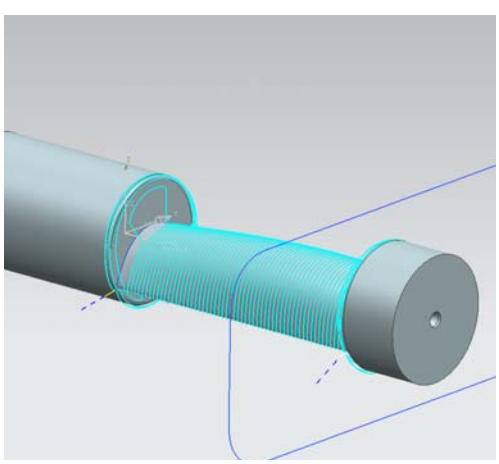
Plane roughing



Roughing was carried out by machining the upper and lower surfaces with the plane roughing method. This figure shows the tool paths involved in machining the upper surface.

From the point of view of ensuring optimum performance and surface quality, constant Z plane roughing is a highly effective approach and allows good control over the level of stress to which the tool is subjected.

5-axis copy milling, face finishing



The 5-axis copy milling method was used for finishing purposes, as this allowed face milling in the form of helical finishing to be performed in accordance with axis selection. The tool is set at a lead angle.

Turbine blade during machining. Face milling on the other side with rotation around the X axis.

Example program code

The key aspects of driving-gear and turbine-component production are illustrated below on the basis of the start program and a finishing program.

Example start program The individual subprograms are called in the start program. All the tool and technology data is stored in the subprogram. The start program controls how the NC programs generated with the CAM system are called.

NOTE **If you have a suitable post processor (e.g. from PostBuilder), both the main programs and the subprograms can be generated automatically.**

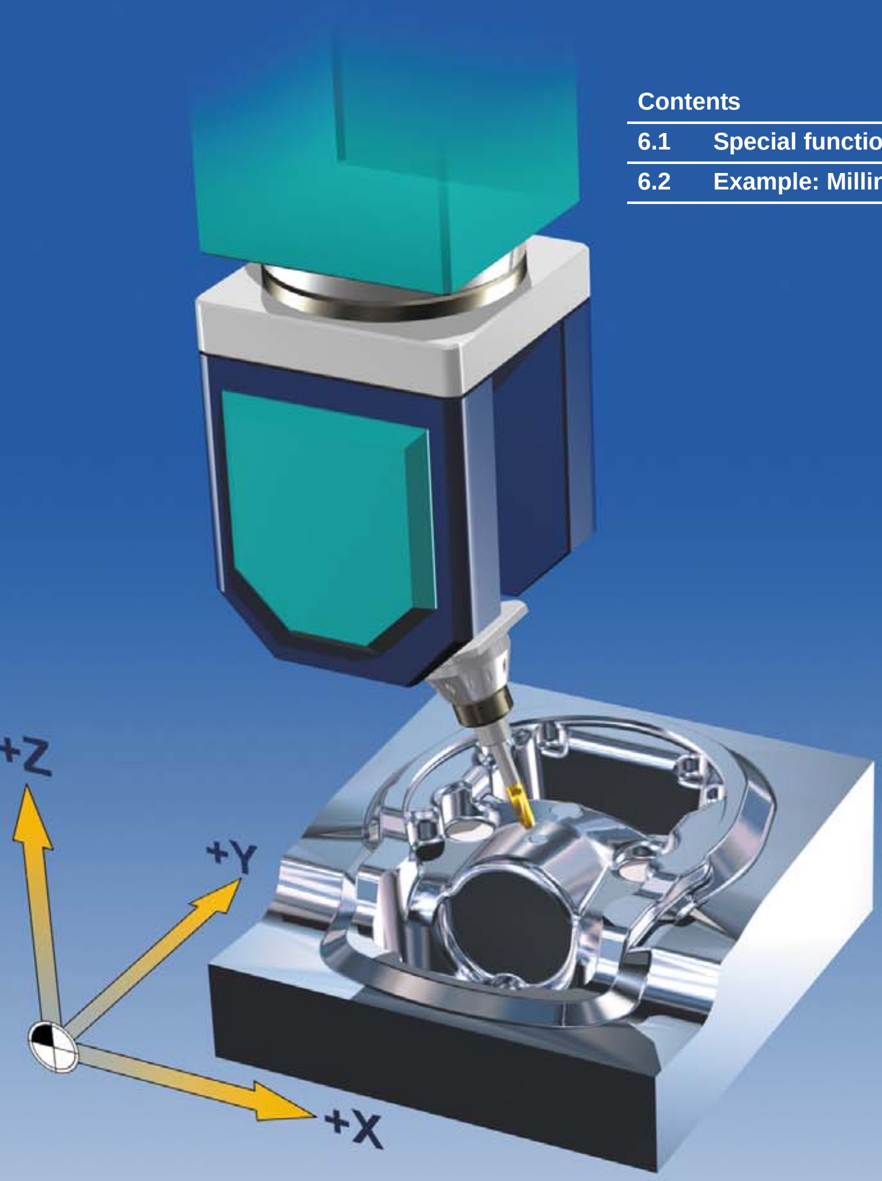
```
N100    ; MILL START PROGRAM   ;
N110    EXTCALL "PROGRAM"      ; Call roughing and finishing subprograms
N120    M01                    ; Continue with NC Start
N130    STOPRE                 ; Preprocessing memory stopped, i.e. the subsequent NC
                               ; blocks will only be read in once all the previous NC
                               ; blocks have been executed.
N140    ...                    ;
N420    EXTCALL "FINISH_04"    ; FINISH_04.MPF finishing program is called. See the
                               ; next page for an explanation of this program.
N220    M01                    ;
N230    STOPRE                 ;
N240    M30                    ; End of program
```

Example finishing subprogram: FINISH_04

The subprogram contains the NC blocks for the geometry and all the data required for production. Assuming that your post processor has been optimized, all this data should be listed in the subprogram. All subprograms are structured in a similar fashion. They only differ in terms of the tool data, technology data, CYCLE832 parameters, and of course the NC blocks.

```
N100   ; TOOL                     ; Tool specification in the form of a comment
N110   ; T1 cherry D8             ; Dimensions of the cherry tool 8 mm
N115   ; Tolerance=0.01           ; Tolerance specification in the form of a comment
N120   G40 G17 G710 G94 G90       ; Tool radius compensation, working plane, metric sys-
                                  ; tem, feedrate in mm/min in relation to spindle, absolute
                                    dimension specification
N125   TRAFOOF                    ; Deactivate all active transformations and frames
N130   CYCLE800(1,"K2X10F",0,57,0,0,0,0,0,0,0,0,0,-1,)
N135                              ; Swivel all axes to the normal position
N150   CYCLE800()                 ; Resetting of the swiveled planes for defined original
                                  ; position
N160   T1                         ; Call tool T1
N170   M6                         ; Change tool in spindle
N180   G54                        ; Work offset
N190   ORIWKS                     ; Workpiece coordinate system is valid
N200   ORIAXES                    ; Axis interpolation
N210   S10000 M3 M8               ; Spindle speed, clockwise rotation, cooling on
N215   CYCLE800(1,"K2X10F",100000,39,0,0,0,90,60,0,0,0,0,-1,)
N220                              ; Pre-positioning of the tool in relation to the workpiece. In
                                  ; each subprogram, a fixed position should first be
                                  ; approached/swiveled into so that there is a defined orig-
                                  ; inal position at the start of machining. This means that if
                                  ; TRAORI is active, the way the workpiece is approached
                                  ; may vary under certain circumstances. Pre-positioning
                                    without TRAORI.
N225   G0 X0 Y0
N230   G0 Z100
N235   CYCLE832(0.01,112001)
N240                              ; Define high speed settings, 0.01 tolerance. From right to
                                  ; left: 1 finishing activated, 0 not assigned, 0 TRAORI
                                  ; deactivated, 2 G642, 1 FFWON SOFT, 1 COMPCAD.
N245   TRAORI                     ; Activate TRAORI
N250   G54                        ; Reactivate work offset after TRAORI
N260   G0 X46.84229 Y48.25858 Z30.5 A3=.89140864 B3=.45320044 C3=0.0 S25000 M3
N270                              ; Rapid traverse to position, define spindle speed and
                                  ; direction of rotation
N275   G1 X21.95965 Y29.38587 A3=.89140864 B3=.45320044 C3=0.0 M8 F6000
N280                              ; Approach first position with feedrate, coolant on
N290   ...                        ; NC blocks for geometry
...    ...
N4580  G0 Z150                    ; Retraction in Z
N4590  TRAFOOF                    ; Deactivate transformation
N4600  CYCLE832(0.02,10000)       ; Set CYCLE832 to default values
N4610  CYCLE800()                 ; Resetting of the swiveled planes
N4620  M5                         ; Spindle stop
N4630  M17                        ; End of subprogram
```

Complex free-form surfaces



6

6.1 Special functions for free-form surfaces

When machining free-form surfaces, surface quality is the top priority. This calls for a correspondingly high level of accuracy for the CAM data with small tolerances and a large number of intermediate points.

The large number of points results in even distribution, thereby ensuring a high level of accuracy and an extremely smooth surface. The integrated high speed setting cycle (CYCLE832) ensures a high machining speed. This cycle is responsible for activating all the functions that are required for milling free-form surfaces. This includes the Look ahead function featuring smoothing and jerk limitation, which anticipates a configurable number of traversing blocks so that the machining speed can be optimized.

In addition, feedforward control has been implemented to ensure that machining is free from following errors. This is supplemented by the COMPCAD online compressor, which is specifically recommended for free-form surface milling applications. This brings together a sequence of G1 commands in accordance with the tolerance set in CYCLE832 and compresses them into a spline that can be executed by the control directly.

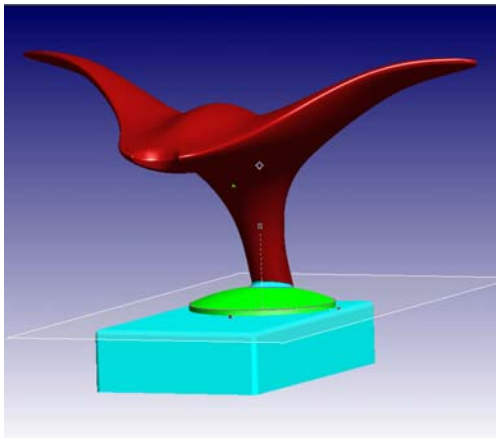
Key functions for machining free-form surfaces:

- High speed settings CYCLE832,
  as optimum data compression within the tolerance band, combined with feedforward control and jerk limitation, can be set for the required surface finish without contour deviation.
- TRAORI,
  as it has an integrated 5-axis transformation feature for all kinds of machine kinematics, enabling the tool to be perfectly oriented in relation to the surface so that no contour or surface defects can occur.
- VCS,
  as maximum accuracy is required in conjunction with extremely large workpieces. Particularly applicable when making compression molds and templates in an automotive engineering context.
- Integrated process chain from generation in CAD through to execution on the CNC.

6.2 Example: Milling a manta ray

This example involves milling a manta ray from a free-form surface model. The manta ray is modeled using a CAD/CAM system.

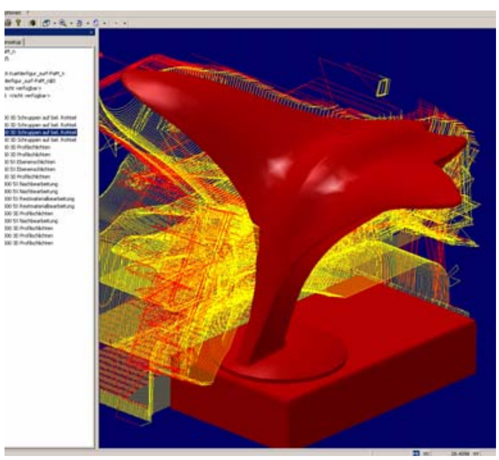
Manta ray in the CAM system



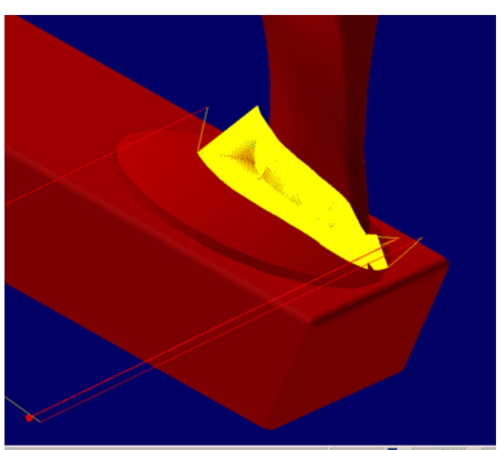
The manta ray is modeled as a free-form surface in the CAM system using a digitized scatterplot. The machining strategies included 3D plane roughing and several 3-axis and 5-axis pre-finishing and finishing operations.

The face was machined, for example, using a line-by-line finishing operation with axes permanently set at an angle. This technique enabled optimum surface quality to be achieved given the extreme level of curvature involved.

Plane roughing with 3 axes



5-axis residual material machining



Strategies for 5-axis residual material machining were used to finish the residual material, e.g. undercutting without taking off the tool.

Example program code

The NC programs for producing the manta ray involve a number of roughing, pre-finishing, and finishing strategies. The key components of the NC programs are illustrated below on the basis of the start program and a roughing program.

Example start program

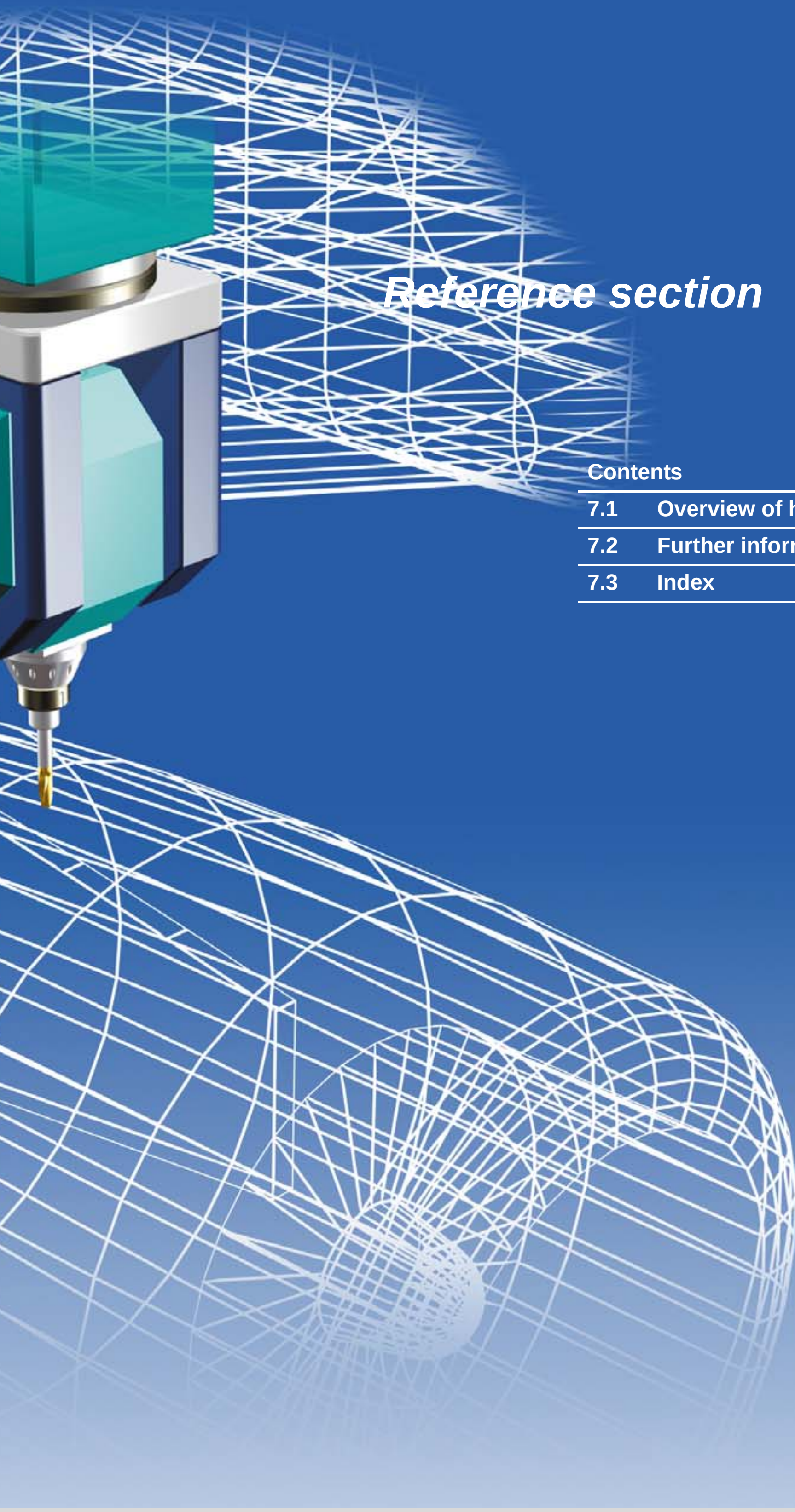
Within the start program, it is sometimes a good idea to make all the settings in the geometry subprograms, particularly as far as individual part production and the test phase are concerned. The start program will only call those subprograms that contain all the specifications such as tool, CYCLE832. This is particularly helpful in cases where the post processor is perfectly attuned to the SINUMERIK system and all higher-order functions have already been integrated. In test mode, it is advisable to execute the subprograms individually; jump labels can be used within the start program, for example, for the purpose of launching the required subprogram.

```
N100   G90 G17 G54             ; Absolute dimension specification, select working plane
                               ; and work offset
N105   ORIWKS ORIAXES          ; Workpiece coordinate system, axis interpolation
N110   GOTOF _ROUGH_01         ; Subprogram jump label for calling roughing with the
                               ; ROUGH_01.MPF program.
                               ; This program is explained in greater detail on the next
                               ; page.
N120   ;GOTOF _ROUGH_02        ; Unused jump labels are
                               ; commented out for the test phase.
       ...                     ;
N210   ;GOTOF _FINISH_05       ;
N220   _ROUGH_01:              ; Jump destination for GOTOF
N230   EXTCALL "ROUGH_01"      ; Call for roughing 01 subprogram
N240   STOPRE                  ; Preprocessing memory stopped, i.e. the subsequent NC
                               ; blocks will only be read in once all the previous NC
                               ; blocks have been executed.
N250   M00                     ; Program stop
N280   ...
N360   _FINISH_05:             ;
N370   EXTCALL "FINISH_05      ; Subprogram call for the last finishing program
N380   STOPRE                  ;
N390   M00                     ;
N400   M30                     ; End of program
```

Example roughing subprogram: ROUGH_01

The subprogram contains the NC blocks for the geometry and all the data required for production. Assuming that your post processor has been optimized, all this data should be listed in the subprogram. All subprograms are structured in a similar fashion. They only differ in terms of the tool data, technology data, CYCLE832 parameters, and of course the NC blocks.

```
N100   ; TOOL                        ; Tool specification in the form of a comment
N110   ; T1 radius milling tool D32  ; Tool dimensions
       R2
N120   G90 G17 G54                   ; Absolute dimension specification, select working plane
                                     ; and work offset
N130   TRAFOOF                       ; Deactivate all active transformations and frames
N140   CYCLE800(1,"K2X10F",0,57,0,0,0,0,0,0,0,0,0,-1,)
N145                                 ; Swivel all axes to the normal position
N150   CYCLE800()                    ; Resetting of the swiveled planes for defined original
                                     ; position
N160   T1                            ; Call tool T1
N170   M6                            ; Change tool in spindle
N180   R2=10000                      ; R2 as parameter for feedrate in XY plane.
                                     ; Feedrate is programmed in NC block as R2. In this way,
                                     ; the feedrate value can be modified quickly for the test
                                     ; phase.
N190   R1=10000                      ; R1 as feedrate in Z direction
N200   R3=4500                       ; Reduced feedrate
N210   S10000 M3 M8                  ; Spindle speed, clockwise rotation, cooling on
N220   CYCLE800(0,"K2X10F",0,57,-36,0,-105,0,0,0,0,0,0,-1)
N225                                 ; Pre-positioning of the tool in relation to the workpiece. In
                                     ; each subprogram, a fixed position should first be
                                     ; approached/swiveled into so that there is a defined orig-
                                     ; inal position at the start of machining. This means that if
                                     ; TRAORI is active, the way the workpiece is approached
                                     ; may vary under certain circumstances. Pre-positioning
                                     ; without TRAORI.
N230   CYCLE832(0.13,112003)         ; Define high speed settings with 0.13 tolerance for
                                     ; roughing. From right to left: 3 roughing, 0 not assigned,
                                     ; 0 no TRAORI, as only 3-axis roughing, 2 G642, 1
                                     ; FFWON SOFT, 1 COMPCAD.
N240   G0 X133.1221 Y1.2413          ;
N250   G0 Z125                       ;
N260   G0 Z108.1501                  ;
N270   G1 Z103.1501 F=R1             ; The programmed feedrate R1 is used here.
N280   X126.5626 Y1.1611 F=R2        ; The programmed feedrate R2 is used here.
N290   ...                           ; NC blocks for geometry
...    ...
N4580  G0 Z150                       ; Retraction in Z
N4590  CYCLE800(1,"K2X10F",0,57,0,0,0,0,0,0,0,0,0,-1,)
N4595                                ; Swivel to original position
N4600  CYCLE832(0.02,10000)          ; Set CYCLE832 to default values
N4610  CYCLE800()                    ; Resetting of the swiveled planes
N4620  M17                           ; End of subprogram
```



Reference section

Contents | Page

7

7.1 Overview of higher-order functions

Higher-order functions of the SINUMERIK control system are summarized on the following pages. This provides you with an overview of the commands that go beyond the requirements laid down in DIN 66025 and that facilitate significant improvements in the area of 5-axis machining.

Motion commands

Language elements with circular interpolation programming

CIP	Circular interpolation using intermediate point CIP X... Y... Z... I1=... J1=... K1=...
CT	Circle with tangential transition CT X... Y... Z...
TURN	Number of full circles to be traversed G3 X… Y… I… J… TURN =
	Additional parameters:
CR=	Circle radius
I1, J1, K1	Intermediate points in Cartesian coordinates (in X, Y, Z direction)
AP=	End point in polar coordinates, polar angle, also in the case of linear interpolation
RP=	End point in polar coordinates, polar radius, also in the case of linear interpolation
AR=	Opening angle

840D spline types

CSPLINE	Activation of cubic interpolating spline
ASPLINE	Activation of Akima spline
	Start and end condition BNAT/ENAT zero curvature BTAN/ETAN tangential transition BAUTO/EAUTO C3-constant at first and last spline segment transition

BSPLINE **SD=...** **PL=...** **PW=...**	Activation of B spline B spline order (max. 3) Interval length (node vector), "Non-uniformity" Weightings, i.e. denominator of rational B spline with polynomial representation **Example** N20 BSPLINE X... Y... SD=... PL=... PW=...

POLY **SD=...** **PL= ...**	Activation of polynomial interpolation, B spline representation in polynomial format B spline order (max. 5!! -> different from BSPLINE) Interval length (node vector), "Non-uniformity" **Syntax** PO[axis] = (block end position, a2 (quadratic coefficient), a3 (cubic coefficient), a4, a5) -> numerator polynomials PO[ ] = (Nblock end, b2, b3, b4, b5) -> denominator polynomial **Example** N10 POLY PO[X] = (0.25,0.5,0) PO[Y] = (0.433,0,0) PO[] = (1,1,0)

Compressor

COMPCAD	Surface-optimized compressor (constant acceleration) With the corresponding single-axis tolerances: **$MA_COMPRESS_POS_TOL[X] = ...** see also CYCLE832
COMPCURV	Transitions with constant acceleration and jerk-free transitions
COMPON	Transitions with constant velocity

UPATH	Additional commands for combining path and synchronized axes. Parameter assignment for synchronized axes corresponds to path axes, i.e. the following applies to the motion of a synchronized axis A: A = f(u), where u denotes the path parameter for the path motion.
SPATH	Parameter assignment of the synchronized axes follows the arc length for the path axes, i.e. the following applies to the motion of a synchronized axis A: A = f(s), where s denotes the arc length for the path motion.

Dynamic response

Technology G groups

DYNNORM	Standard dynamic response, as previously
DYNPOS	Positioning mode, tapping
DYNROUGH	Roughing
DYNSEMIFIN	Finishing
DYNFINISH	Smooth-finishing

Look ahead

G60	**Exact stop** at end of block
G601 **G602** **G603**	Block change on reaching the fine in-position window Block change on reaching the coarse in-position window Block change at the end of interpolation
G64	**Overrun of block end (LOOK AHEAD)**
	Smoothing
G641	ADIS = … smoothing distance ADISPOS =… smoothing distance for G0, constant velocity
G642	Smoothing with single-axis tolerances ($MA_COMPRESS_POS_TOL[X] = …) or ADIS, ADISPOS with intermediate blocks, constant-acceleration **(recommended)**
G60, G64, G641, G642	G code group 10
G601 – G603	Internal G code group (group 12)

Velocity programming

	Conventional block-by-block (non-modal) velocity programming in
G94	Inches/min or mm/min
G93	Inverse time
G95	Inches, mm per spindle revolution
G96	Constant cutting rate

	Programming of velocity/feedrate profiles To permit flexible definition of the feedrate characteristic, feedrate programming according to DIN 66025 has been extended by means of linear and cubic characteristics. The cubic characteristics can be programmed either directly or as interpolating splines. These additional characteristics make it possible to program continuously smooth velocity characteristics depending on the curvature of the workpiece to be machined. As a result, limiting acceleration changes can be programmed, enabling uniform workpiece surfaces to be produced.
FNORM	Basic setting. The feedrate value is defined via the block's path and subsequently applies as a modal value.
FLIN	Linear path velocity profile: The feedrate value is approached linearly using the path, from the current value at the start of the block right through to the end of the block; it then applies as a modal value.
FCUB	Cubic path velocity profile: The F values that have been programmed block by block are combined (in relation to the end of the block) by means of a spline. The spline starts and ends at a tangent to the previous or subsequent feedrate function. If the F address is missing from a block, the most recently programmed F value will be used for this purpose.
F=FPO(..)	Path velocity profile via polynomial: The F address describes the feedrate characteristic on the basis of a polynomial, from the current value right through to the end of the block. The end value subsequently applies as a modal value.
endfeed	Feedrate at end of block
quadf	Quadratic polynomial coefficient
ubf	Cubic polynomial coefficient
	Path reference
FGROUP(X, Y, Z,...)	Defines the path axes in relation to the feedrate, i.e. the overall feedrate relates to the axes defined here. Example: FGROUP(X, Y), so the following then applies: $F = \sqrt{F_x^2 + F_y^2}$

Jerk

SOFT	Jerk limitation
BRISK	Acceleration limitation

Feedforward control

FFWON	Feedforward control on
FWOF	Feedforward control off

5-axis functionality

Transformation

TRAORI	Activation of transformation 1
TRAORI(1)	Activation of transformation 1
TRAORI(2)	Activation of transformation 2
TRAORI(1, ..., ..., ...)	Activation of transformation 1, generic transformation, additional 3 parameters for basic orientation vector
TRAORI(2, ..., ..., ...)	Activation of transformation 2, generic transformation, additional 3 parameters for basic orientation vector
TRAFOOF	Deactivation of transformation

Orientation programming

ORIEULER	Orientation programming on the basis of Euler angles (default)
ORIRPY	Orientation programming on the basis of RPY angles In both cases, the function will only be effective if $MC_ORI_DEF_WITH_G_CODE is set to 1. Otherwise, specified via machine data. On older systems, a distinction can only be made by means of machine data $MC_ORIENTATION_IS_EULER.
A2=... B2=... C2=...	Euler or RPY angles
A3=... B3=... C3=...	Cartesian orientation vector
XH=..., YH=..., ZH=...	In the case of ORIVECT or ORIPLANE: synonymous with A3=... etc. More extensive meaning in conjunction with ORICURVE; in this case either serves as a control polygon with BSPLINE, or has a polynomial definition with POLY, otherwise linear interpolation for the upper straight line, geometric large circle, but not in terms of velocity.
LEAD, TILT	Lead/tilt angle relative to normal vectors and path tangent. The normal vectors at the start of the block and at the end are defined by A4=... B4=... C4=... und A5=... B5=... C5=... Only in conjunction with ORIPATH.

Orientation reference

ORIMKS	The reference system for the orientation vector is the basic coordinate system. When $MC_ORI_IPO_WITH_G_CODE = 0, identical to ORI-AXES at the same time.
ORIWKS	The reference system for the orientation vector is the workpiece coordinate system. When $MC_ORI_IPO_WITH_G_CODE = 0, identical to ORIVECT at the same time.

Orientation interpolation

	The following G codes are only active, when $MC_ORI_IPO_WITH_G_CODE is set to 1:
	Axis interpolation
ORIAXES	Linear interpolation of the machine axes or interpolation of the rotary axes using polynomials (with active POLY)
	Vector interpolation
ORIVECT	Interpolation of the orientation vector in a plane (large circle interpolation)
ORIPLANE	Interpolation in a plane (large circle interpolation), synonymous with ORIVECT
ORIPATH	Tool orientation in relation to the path. A plane is created from the normal vector and path tangent that defines the meaning of LEAD and TILT at the end point. In other words, the path reference only applies to the definition of the end orientation vector. Large circle interpolation is performed between the start and end orientations. LEAD and TILT do not merely provide the lead and tilt angles. They have the following significance: LEAD defines the rotation in the plane created by the normal vector and path tangent. TILT then defines the rotation around the normal vector. In other words, they correspond to theta and phi in a sphere coordinate system, with the normal vector serving as the Z axis and the tangent as the X axis.
ORICONCW	Interpolation on a peripheral surface of a cone in the clockwise direction
ORICONCCW	Interpolation on a peripheral surface of a cone in the counterclockwise direction Also required in both cases: A3=... B3=... C3=... or XH=., YH=... , ZH=... End orientation, cone's axis of rotation: A6, B6, C6 Opening angle: NUT=.
ORICONIO	Interpolation on a peripheral surface of a cone with an intermediate orientation specified via A7=... B7=... C7=.... Also required: A3=... B3=... C3=... or XH=..., YH=..., ZH=... end orientation

ORICONTO	Interpolation on a peripheral surface of a cone with tangential transition Also required: A3=... B3=... C3=... or XH=..., YH=..., ZH=... end orientation With POLY, PO[PHI] = ..., PO[PSI]=... can also be programmed in these cases. This is a more generalized form of large circle interpolation, whereby polynomials are programmed for the lead and tilt angles. With cone interpolation, the polynomials have the same significance as with large circle interpolation for the given start and end orientations. The polynomials can be programmed with ORIVECT, ORIPLANE, ORICONCW, ORICONCCW, ORICONIO, ORICONTO.
ORICURVE	Orientation interpolation with specification of the tool tip motion and that of a second point on the tool. The path of the second point is defined via XH=... YH=... ZH=..., in conjunction with BSPLINE as a control polygon with POLY as polynomial:' PO[XH] = (xe, x2, x3, x4, x5) PO[YH] = (ye, y2, y3, y4, y5) PO[ZH] = (ze, z2, z3, z4, z5) If the BSPLINE or POLY additional information is omitted, straightforward linear interpolation will be performed accordingly between the start and the end orientation.

Tool radius compensation

G40	Deactivation of all variants
G41	Activation for circumferential milling, compensation direction left
G42	Activation for circumferential milling, compensation direction right
G450	Circles at external corners (all compensation types)
G451	Intersection method at external corners (all compensation types)

2½D circumferential milling

CUT2D	2 1/2D COMPENSATION with compensation plane determined using G17 - G19
CUT2DF	2 1/2D COMPENSATION with compensation plane determined using a frame

3D circumferential milling

CUT3DC	Compensation perpendicular to path tangent and tool orientation
ORID	No changes in orientation in inserted circular blocks at external corners. Orientation motion is performed in the linear blocks.
ORIC	Travel path is extended by means of circles. The change in orientation is also performed proportionately in the circle.

Face milling

CUT3DFS	Constant orientation (3-axis). Tool points in the Z direction of the coordinate system defined via G17 - G19. Frames do not have any effect.
CUT3DFF	Constant orientation (3-axis), tool in Z direction of the coordinate system currently defined via the frame.
CUT3DF	5-axis with variable tool orientation

3D circumferential milling with limitation surface - Combined circumferential/face milling

CUT3DCC	NC program relates to the contour on the machining surface.
CUT3DCCD	The NC program relates to the tool center point path.

FRAMES

Programmable frames

TRANS X… Y… Z…	Absolute offset
ATRANS X… Y… Z…	Incremental offset, relative to the currently active frame
ROT X… Y… Z…	Absolute rotation
AROT X… Y… Z…	Incremental rotation, relative to the currently active frame
ROTS X… Y…	Absolute rotation, defined by two angles. The angles are the angles of the lines of intersection between the inclined plane and the main planes with respect to the axes.
AROTS X… Y…	Incremental rotation, relative to the currently active frame, angles as for ROTS
RPL=…	Rotation in the plane
MIRROR X… Y… Z…	Absolute mirroring
AMIRROR X… Y… Z…	Incremental mirroring, relative to the currently active frame
SCALE X… Y… Z…	Absolute scaling
ASCALE X… Y… Z…	Incremental scaling, relative to the currently active frame

Frame operators

	Frame operators can be used to define frame variables as a chain of individual frame types:
CTRANS (X… Y… Z…)	Absolute offset
CROT (X… Y… Z…)	Absolute rotation
CROTS (X… Y… Z…)	Absolute rotation
CMIRROR (X… Y… Z…)	Absolute mirroring
CSCALE (X… Y… Z…)	Absolute scaling

FRAME = CTRANS(…) : CROT (X… Y… Z…) : CMIRROR (X… Y… Z…)

Special frames

TOFRAME	Tool frame, coordinate system with Z axis in tool direction, zero point is the tool tip
TOFRAMEX	Tool frame, coordinate system with X axis in tool direction, zero point is the tool tip
TOFRAMEY	Tool frame, coordinate system with Y axis in tool direction, zero point is the tool tip
TOFRAMEZ	Tool frame, coordinate system with Z axis in tool direction, zero point is the tool tip, identical to TOFRAME
TOROT	Tool frame, coordinate system with Z axis in tool direction, only contains the rotation component from TOFRAME. The zero point remains unchanged.

TOROTX	Tool frame, coordinate system with X axis in tool direction, only contains the rotation component from TOFRAME. The zero point remains unchanged.
TOROTY	Tool frame, coordinate system with Y axis in tool direction, only contains the rotation component from TOFRAME. The zero point remains unchanged.
TOROTZ	Tool frame, coordinate system with Z axis in tool direction, only contains the rotation component from TOFRAME. The zero point remains unchanged.

7.2 Further information/documentation

Information about the SINUMERIK system can be found in a number of sources. User and manufacturer documentation is supplemented by user forums and information on the Internet. An overview of this additional information is provided below.

Doconweb

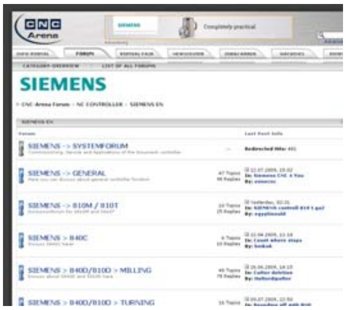
Internet site that enables you to download the SINUMERIK documentation in its entirety. You can search for specific terms online, look things up in the index or download the required manual in PDF format.

www.siemens.com/automation/doconweb

CNC4you - User portal

This portal provides up-to-date information about SINUMERIK controls and real-life examples.

www.siemens.com/cnc4you



SINUMERIK - User forum

The SINUMERIK user forum is a platform that allows you to discuss technical issues with other SINUMERIK users. The forum is moderated by experienced Siemens technicians.

www.siemens.cnc-arena.com

7.3 Index

Further information

More details on SINUMERIK are available under:
www.siemens.com/automation/mc

For detailed technical documentation refer to our Service&Support Portal:
www.siemens.com/automation/support

For contac persons near you refer to:
www.siemens.com/automation/partner

Direct online ordering is possible in our mall:
www.siemens.com/automation/mall

Siemens AG
Industry Sector
Drive Technologies
Motion Control
Postfach 3180
91050 Erlangen
DEUTSCHLAND

Subject to change without prior notice
6FC5095-0AB10-0BP1

www.siemens.com/automation/mc