

UMESS

Option 9
PCM

Operating Instructions



This manual must not be circulated or copied, or its contents utilized and disseminated, without our express written permission. Persons misusing this manual are subject to prosecution.

All rights reserved, especially in cases of granting a patent or registering a utility model.

Subject to modifications. All rights pertaining to changes in the measuring machine version, scope of delivery, the software packages and the pertaining documentation reserved.

Carl Zeiss does not grant any warranty on this material and no implied warranties on commercial quality and the suitability for a certain purpose.

Carl Zeiss cannot be held liable for errors included in this document, accidental damage or damage resulting from the provision, function or use of this manual.

All product names are registered trademarks or trademarks of the corresponding proprietors.

Carl Zeiss
Corporate Division
Industrielle Meßtechnik
D-73446 Oberkochen

Document type: . . . Operating instructions
Version: 8.x
Datum: 06/01
Order No.: 61212-1100102

Preface

It is here assumed that the user is familiar with the coordinate measuring machine and its components. Please keep all printed materials delivered with the measuring machine ready to hand at all times.

Principles in this operating manual

Before starting to work with this manual, the user has to familiarize himself with the applied principles.

In the following, you will find information on the used font types, signs and symbols.

Typographic principles

The font types and font schemes used in this manual have the following meaning:

- **bold face**
 - Dialog element on the screen
Example: „... the button **<TERMIN>**“
 - Term
Example: "During calculation, the location of a **measuring element** in relation to a **reference element** is determined."
 - File and directory names
Example: **/home/zeiss/UB**
- *italic*
 - Highlighted text of which the contents is very important
Example: "Click with the *right* mouse button"
 - Cross reference
Example: "....., see also ► „Activating the PCM edit mode“ on page 1-10"
- Courier
Program code, file contents
- **Courier bold face**
Text in dialog windows and records

Signs and symbols

Special signs and symbols are used in this manual.

Symbols for warnings and information



Danger!

In this case, special care is called for. The warning triangle indicates risk of injury. Non-observance of this warning may cause personal injury.



Attention!

This symbol warns against situations which may lead to loss of data, measuring errors, errors in the measuring run, collisions or damage to the machine and workpiece.



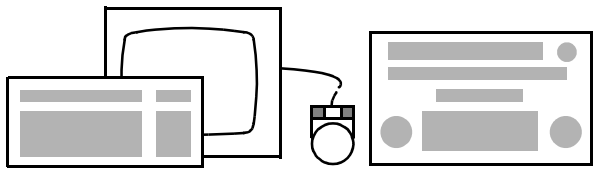
The **Note** symbol is shown next to important text and helpful additional information.

Symbol for function call

There are several possibilities:

- Direct input by means of the DI number
- Function selection by means of the pull-down menu
- Selection by means of icons

Example:



DI	Softkey	FFK	Menu
	<>	<>	<CNC>
	<>		<PCM>
1666			<PCM-Edit on/off>



Symbol for softkey

Reference to softkeys in dialogs.

Overview of chapters

This Manual describes the UMESS Opt. 9 PCM development system.

The following subjects are described:

- „The PCM development system“ on page 1-1
- „PCM functions“ on page 2-1
- „Application example: Bore plate“ on page 3-1

Direct input functions

DI no.	Input abbrev.	Function	Page
1666		PCM edit mode	➤ page 1-10
1642		Correct control data	➤ page 1-15
1671		PCM edit mode	➤ page 1-10
1646		PCM test run	➤ page 1-11
1647		PCM generating run	➤ page 1-13 ➤ page 2-7
1693		Address offset mode even without parameterized NP TERMIN - line	➤ page 2-17
1459		Nominal-actual comparison, enter nominal dimension	➤ page 2-15

Table of contents

Principles in this operating manual	3
Typographic principles	3
Signs and symbols	4
Overview of chapters	5
Direct input functions	5

Chapter 1 The PCM development system

General	1-2
Programming for parts families (Variant programming) .	1-3
Terms PCM file / PCM run	1-3
Principle of the variant programming	1-4
CNC measuring runs for parts families (PCM file/PCM run)	1-5
Part runs	1-7
Parameterization of your own control data, creating your own measurement modules	1-8
Introduction	1-8
Procedure for programming	1-8
Value input for your own measurement modules	1-9
Activating the PCM edit mode	1-10
Activating the PCM run mode	1-11
PCM test run <DI 1646>	1-11
PCM generating run <DI 1647>	1-13
Structure of a control data line	1-15
Value assignment for P parameters	1-15
Parameterizing loops and branchings	1-16
Parameters	1-17
Numerical parameters	1-17
Text parameters	1-18
Measurement result parameters	1-20

Permissible functions and operations	1-21
VPII parameter	1-22

Chapter 2 PCM functions

Definition of a new coordinate system	2-2
Coordinate transformations for parameters	2-4
Modi for coordinate transformation and other control data manipulations	2-7
FOCUS PCM functions	2-11
Dialog functions (parameter input)	2-13
Reading ASCII files with PCM syntax	2-15
Reading ASCII files without PCM syntax	2-17
Reading/editing/outputting ASCII files for string editing ...	2-20
Other help functions	2-24
Vector functions	2-26

Chapter 3 Application example: Bore plate

Technical data of the bore plate	3-2
Bore plate parameters	3-3
Structure of the CNC program	3-4
Control data list	3-5
Explanation of the control data	3-7

Chapter



The PCM development system

This chapter contains:

General	1-2
Programming for parts families (Variant programming).	1-3
Part runs	1-7
Parameterization of your own control data, creating your own measurement modules.	1-8
Structure of a control data line.	1-15
Parameters.	1-17

General

The PCM development system (PCM = Parameter Control Manager) is used for rationalizing and simplified generation of measuring runs, in the following cases:

- Workpieces of a parts family (variant programming)
- Similar recurring runs on the workpiece (geometric elements, combined measuring tasks)
- Flexibility of the measurement and the result output whereby you can switch between the complete run and partial runs of the measuring program. Control by the user, measurement or external computer possible.

In addition to the program library, the PCM development system comprises the following performance features:

- Extended variant programming. This deals with measuring runs for workpieces principally with the same measuring run but different dimensions.
- Programming of branchings by assigning parameters via PCM files or by entering parameters when starting the CNC run. In this way partial runs are possible.
- Programming of loops, e.g. for measuring bore patterns.
- Creation of your own PCM program library.

NOTE

Use of the PCM measurement module library, i.e. the use of PCM measurement modules created by Zeiss, is described in the UMESS manual, Option 10.

Programming for parts families (Variant programming)

Terms PCM file / PCM run

The variant programming is used for similar workpieces for which the principle measuring run is the same, but the dimensions different.

Only one generally valid measuring run (PCM run) has to be created per parts family for which the fixed numerical values of the measurement module calls are kept variable by parameters (➤ „*Principle of the variant programming*“ on page 1-4).

The actual dimensions are in parts-specific files (PCM files) with which the CNC run can also be started. This is possible as apart from the value assignments for the geometry data of the individual elements the call (**EXCALL**) of the PCM run is also contained and PCM files, formally seen, are normal control data.

With a new part of a parts family, the values of an existing PCM file of the parts family only need to be changed and stored as a new file.

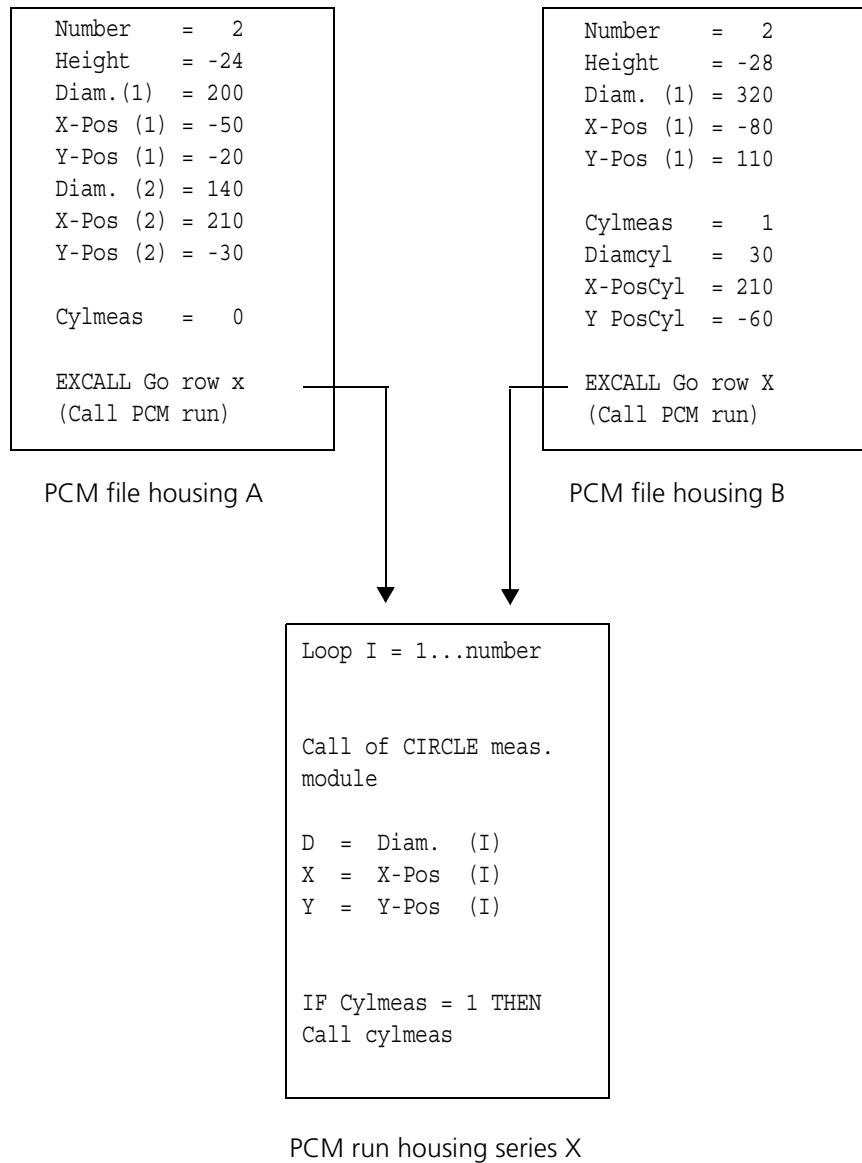
With the UMESS Opt. 9 , the number of elements can be varied by parameterizing loops.

Missing components for a part can be bracketed out by IF/ENDIF without a new PCM run having to be created.

Advantage:

- Considerable reduction of the time needed for programming and testing.
- Considerable increase of effectivity, particularly together with the measurement module library (UMESS Opt. 10).
- Measuring runs for a new variant are available more quickly.
- Improved overview of parts family by avoiding redundancy.

Principle of the variant programming



CNC measuring runs for parts families (PCM file/PCM run)

Generating a PCM run

- Create a control data run by self-teach programming for which all variable elements (within the parts family) are measured using the PCM measurement module library (see using the PCM measurement module library).
- Check the function capability of the run by CNC run.
- Edit this workpiece (PCM run) with the control data editor by changing the fixed numerical values of a measurement module call in parameters. Only those value assignments have to be changed which vary within the parts family.

Example control data line before change

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	50.75% Diam 3	ASSIGNMENT	25	0	9979	0	

Example control data line after change

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	P127% Diam 3	ASSIGNMENT	25	0	9979	0	

NOTE

The values right of the **ASSIGNMENT** function text must not be changed here as they are needed for the assignment of the value within the measurement module.

Generating the first PCM file

- Self-teach programming of a workpiece which only contains the record header.
- Insert assignment lines (see value assignment) using the control data editor:
 - „Enter=" in Function column
 - Input of the index for the P parameter in the SC1 field (e.g. input of 127 when assigning the P127 parameter)
 - Input of the fixed numerical value and comment in the left part of the line.
- In the PCM run an assignment in the PCM file must exist for each parameterized value of a PCM measurement module if the parameterized value cannot be calculated any other way (e.g. with loops) or is used several times (e.g. with tolerances).

Generating more PCM files

- Call existing PCM file with the control data editor.
- Change the values of the assignments (left control data part).
- When ending the control data correction, store workpiece under a new name.

Part runs

By bracketing individual sections of a run with IF/ENDIF this can be directly masked/unmasked in the measuring run.

A section can be preselected for the respective run by assigning parameters in PCM files or by input at the CNC start (INPUT START, ASSIGNMENT , INPUT END). For more details, see ➤ „Parameterizing loops and branchings" on page 1-16.

Before each part run, the address counter should be set to a defined value in order to avoid shifts during variable part runs or incorrect recalls (particularly when using the EXCALL address correction). When using intermediate positions make sure that part runs – independent of the grouping made – can be carried out without collision.

```

INPUT START
Part meas. A = 0 (ASSIGN)
Part meas. B = 0
Part meas. C = 0
INPUT END
Set addr. to 10
IF part meas. A = 1 THEN
..
ENDIF

Set addr. to 20
IF part meas. B = 1 THEN
..
ENDIF

Set addr. to 50
IF part meas. C = 1 THEN
..
ENDIF

```

```

Part meas. A = 1
Part meas. B = 0
Part meas. C = 1

```

Selection in table page
with CNC start

Control data workpiece X

Parameterization of your own control data, creating your own measurement modules

Introduction

If you have to create your own PCM measurement modules or want to create flexible measuring runs, fixed numerical values in the control data must be replaced by variables (parameterizing). Instead of individual parameters, linkings of parameters with various operations and functions can also be used.

Using the PCM test run **<DI 1646>** the conversion of the parameters into numerical values can be checked. For use in systems which do not enable PCM mode, conventional control data can be generated with **<DI 1647>**.

All parameters used in the control data must generally be assigned with previous value assignment. The values of the corresponding parameters replace the parameterized control data during the CNC run (probing, intermediate positions, etc.).

This also applies to PCM measurement modules. Here the input values for the measurement module are not calculated until during the CNC measuring run by further value assignments. In this case the values for the assignments are themselves parameters or linkings of several parameters.

Procedure for programming

- Think carefully beforehand which values are to be variable. The structure of the program remains clear if there is a sufficient number of reserved variables. Reserve a few variables more in order to be able to carry out supplements later.
- Program in a modular structure. You will then have short programs which can be changed easily.
- Avoid data input if parameters can be calculated from values already entered.
- Program the workpiece as usual (intermediate positions, probing, macros, library, EXCALL).
- Activate PCM edit mode with **<DI 1666>** and using the control data editor (**<DI 1642>**) replace the fixed values by variables.
- Execute PCM test run

We recommend the program be checked theoretically before running on the coordinate measuring machine. This is of particular importance if the workpiece could be destroyed as a result of a collision.

Value input for your own measurement modules

With the PCM development system, UMESS Option 9, you can create your own PCM measurement modules.

You create your own PCM measurement modules by parameterizing existing control data blocks and including PCM functions within the control data editor. (► „Parameterization of your own control data, creating your own measurement modules" on page 1-8)

You can then call your own PCM measurement modules as subprograms during the self-teach programming of a CNC program. To do this, you have to enter the current values in a table (dialog file), the measurement module is processed in the CNC program and then the PROG mode is returned to.

You have to deposit your own PCM measurement modules in the control data catalog in the measurement module library; for more information, see UMESS Opt. 8 "Managing control data catalogs".

In order to be able to call your own measurement module within a CNC program, you need a corresponding dialog file. Usually a corresponding dialog file belongs to each PCM measurement module. You also have to put these dialog files in the control data catalog (possibly in its own separate dialog catalog); use and terms, see UMESS Opt. 10 "PCM library".

In the dialog file, you have to assign all the parameter values which are used in the measurement module and are not calculated there internally from other values. The parameter contents of the dialog file are used for preassigning during the self-teaching programming. If the same measurement module is called repeatedly, then the values entered last are offered.

The dialog file consists of assignment lines, the EXCALL call of the measurement module and two lines for labelling the dialog lines (INPUT START and INPUT END).

The assignment lines form the dialog during the value input in the PROG mode.

NOTE

A library with various geometric elements and special elements has been created. This can be found in the UMESS Opt. 10 "PCM library" so that in a standard case you do not have to create your own measurement modules.

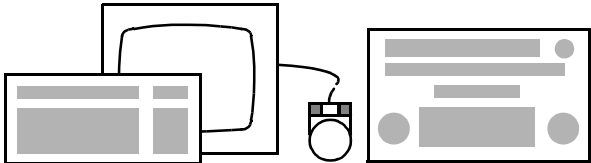
Activating the PCM edit mode

With the PCM edit mode you can generate parameterized control data.

Before you call the control data editor to parameterize your own CNC runs, you have to activate the PCM EDIT MODE <DI 1666> if this has not already been set in the long term mode.

In fields which have contained numerical values till now (e.g. coordinates, loop values,...) you can now also enter parameters or PCM terms (exception: fixed control data part). Control data lines which contain texts and numerical input fields in the variable part (on the left) are only converted if at least one numerical parameter occurs.

Function call



DI	Softkey	FFK	Menu
	<>	<>	<CNC>
	<>		<PCM>
1666			<PCM-Edit on/off>

PCM - EDIT - MODE

J

activated

*

* YES

NO

*

TERMIN

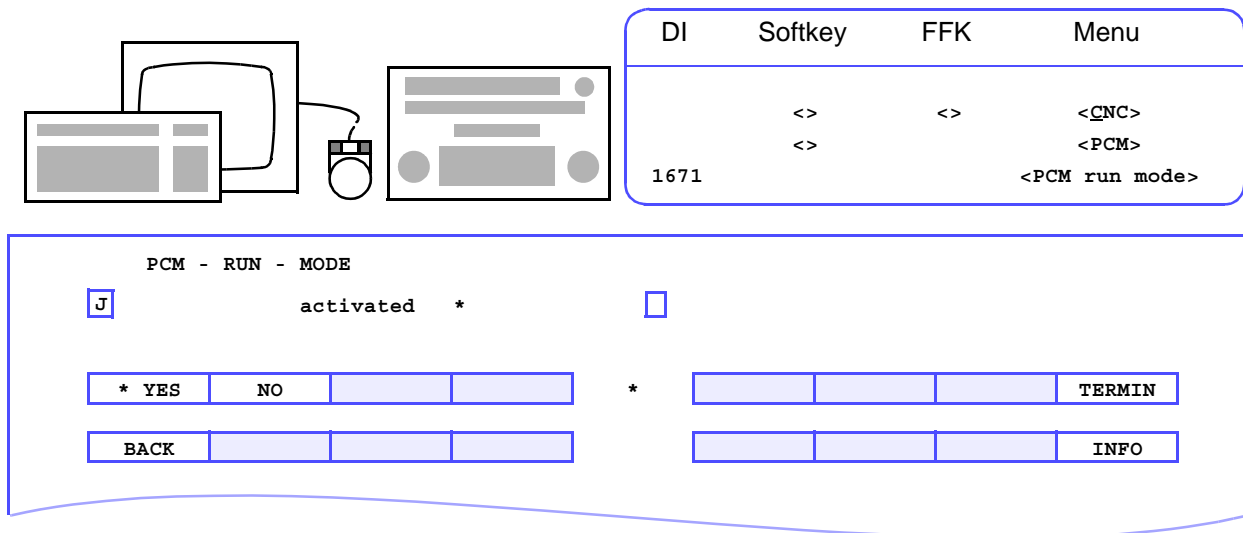
BACK

INFO

Activating the PCM run mode

So that you can self-teach the PCM macros in the PROG mode, you have to activate the PCM run mode. (PROG mode, see UMESS Main operating instructions)

Function call



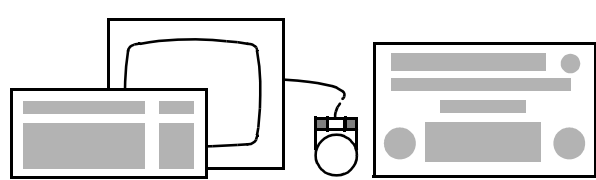
PCM test run <DI 1646>

With the PCM test run you can control the parameterized control data of your PCM runs during the conversion in a CNC run. The parameterized control data lines are output before (with parameters) and after (with calculated numerical value) the conversion in the UMESS record. The function text is output for lines not parameterized.

If a PCM term cannot be converted due to a parameterization error, an error message is displayed. If a coordinate measuring machine is connected, the run is canceled. This also applies to the normal CNC run.

With PCM functions, the actions executed are recorded with the respective relevant values. By assigning 1.000.000 to the parameter #999, the PCM test list can also be activated. (Application: partial output as test list)

Function call



DI	Softkey	FFK	Menu
	<>	<>	<CNC>
	<>		<PCM>
1646			<PCM test run>

PCM test run

c

WP code

Start line

A-Pos

Start paper

Execute CNC and calculate results

Cat code:

Workpiece name

Comment

End line

manually or autom.

☐
*

☐
*

Measurement library dialog German

* YES

NO

*

CATALOG

TERMIN

BACK

INFO

Input field

Execute CNC and calculate results

- <YES>
Run in CNC mode (CMM travels, output of results in record)
- <NO>
CMM does not travel, no result output, however loops, branchings, EXCALL jumps, parameter assignments and PCM functions are executed.

PCM generating run <DI 1647>

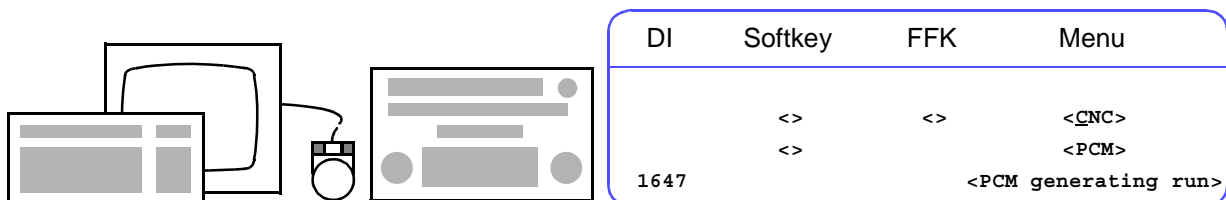
With the PCM generating run, you can generate parameter-free control data in order to transfer PCM runs to program systems not capable of PCM or to coordinate measuring machines with CMS program systems. (PCM-UX control data cannot yet be converted to CMS; travel path plots are only possible at the moment with generated PCM runs.)

The CNC run created is parameter-free and can be converted to CMS control data.

During the PCM generating run, all parameterized control data lines are converted to parameter-free lines, for this all PCM functions, assignments, loops and branchings are executed and canceled.

You can prevent the conversion of parameterized control data lines to parameter-free control datalines using the PCM function `GEN_FILT_OFF(X)`, for more information ➤ „Modi for coordinate transformation and other control data manipulations" on page 2-7.

Function call



1. Input mask

Enter the parameter-free workpiece to be generated.

CNC admin: PCM - Generating run
Cat code:
Converted catalog

Input of the (parameter-free) workpiece to be generated

☒ Restart
☐

WP code
Workpiece name

Comment

* YES
NO

*
CATALOG
TERMIN

BACK

INFO

TERMIN

A second input mask follows

2. Input mask

Enter the parameterized workpiece to be converted.

PCM - Generating run		Cat code: ZA Library Dialog English	
Input of the (parameterized) workpiece to be converted			
<input type="checkbox"/> C	WP code	<input type="text"/>	Workpiece name
			Comment
	A pos.	<input type="text"/>	
	Convert parameter lines (NO: only execute EXCALL)	<input type="checkbox"/>	
	Execute CNC and calculate results	<input type="checkbox"/>	
	Convert masked lines	<input type="checkbox"/>	
<input type="button" value="YES"/> <input type="button" value="NO"/>		<input type="button" value="CATALOG"/> <input type="button" value="TERMIN"/>	
<input type="button" value="BACK"/>		<input type="button" value="INFO"/>	

Input fields

Convert parameter lines

<YES>

Parameterized control data lines are converted to parameter-free control data lines

<NO>

No conversion of parameterized control data lines, only inclusion of EXCALL calls into the main program

Execute CNC and calculate results

<YES>

Run in CNC mode (CMM travels, output of results in the record)

<NO>

CMM does not travel, no result output, but loops, branchings, EXCALL jumps, parameter assignments and PCM functions are executed

Convert masked lines

<YES>

Parameterized lines which are masked are converted to parameter-free lines

<NO>

Parameterized lines which are masked are not converted

Structure of a control data line

Variable part			Fixed part				
No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	PCM	0	1	9972	0	

In each field of the **variable part** of a control data line, a number, a parameter or a PCM term can be entered. Separators (;), which are output in the list during the PCM test run must not be entered.

If the variable control data part contains at least 1 parameter, then YES/NO fields must be assigned with 0 for NO or 1 for YES. If the field is to be variable, a parameter must be entered instead.

NOTE

The sum of the characters in the variable part + the number of variable fields - 1 may reach a maximum of 24. Right justified blanks are truncated. A message is output if the permissible value is exceeded. If a term exceeds the length of 24 characters, then this must be divided into several variables and over several lines.

In the **fixed part** of the control data line (code numbers part) SC1, SC2 and ADR can be parameterized. The parameters P1 ... P950 are permitted.

Value assignment for P parameters

A value assignment line assigns a P parameter with a numerical value during the CNC run.

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	(3) 120%Diam. hole	"=" (1) ASSIGN.	(2) 122	0	9979	0	

Sequence in which the columns are assigned:

Fixed part, columns:

(1) Function

Input of "=" assigns column with ASSIGNMENT

(2) SC2

Index of the parameter (e.g. 122 for P122)

ASC1/PCN/CCN

Automatic entry by editor = 0/9979/0

Variable part, column:

(3) Dialog

Any term as PCM syntax. Here assignment term for assigning the variables with a number or another variable. Comment after assignment term possible "%", example: 120%Diam. hole

P parameters keep their value until they are overwritten by a renewed assignment or a PCM function (TOREC, PCLEAR, ...).

The PCM function PLIST can be used for helping to test during PCM runs, this lists all parameters currently assigned.

Parameterizing loops and branchings

Parameterization of the DO LOOP loops as other control data lines.

Parameterization of branchings with any character format (IF, UNTIL, REPEAT, DO WHILE) with P (X) . (X = Index of the P parameter)
No linkings! Other syntax during branchings as usual.

Example

```
P(20)13      IF      ...
```

The subsequent control data part is processed if P20 is smaller than 3.

Parameters

Numerical parameters

Numerical parameters are variables which are entered in the control data instead of fixed numbers.

Global parameters	P1 ... P990, P1000 ... 5000
Actual probe radius:	PT
Constants π (Pi):	PI
Address counter:	PRC
mm/Inch factor:	MIF
Random number between 0 ... 1:	PRND
Current status of the (current) loop counters:	I

NOTE

Loop counter: The branchings `IF` and `REPEAT` also count as part of the nesting depth, therefore at the start of the loop with subsequent `IF` the loop counter `I` must be assigned another parameter `P`.

With PCM measurement modules the following areas named are assigned with P parameters. Therefore, they should not be used globally:

P1 ... P100 , P600 ... P820 , P900 ... P910 , P950 ... P1000

Global P parameters also apply after a jump to another workpiece with `EXCALL`.

In order, for example, to access different parameters independent of the status of the loop counter, parameters can also be indexed and linked to each other with mathematical operations (+, -, *, /, ^).

`P(I)% DiamASSIGNMENT ...`

`P(I+1) % A1ASSIGNMENT ...`

➤ „Principle of the variant programming" on page 1-4

Text parameters

Text parameters are variables which are contained in the control data instead of fixed character strings (texts, file names, nominal identifications, UMESS formulas, etc.)

Global text parameters @1 ... @950
 Length of text parameters: 30 characters
 Part number from record header: @994

Instead of fixed text enter @xxx (@xxx = no. of the text parameter used, example: @126)

Value assignment to text parameters

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	(3) BO_23%Nom value idf	"==" (1) ASSIGN TEXT	(2) 120	0	9980	0	

Sequence in which the columns are assigned:

Fixed part, columns:

(1) Function

Input of "=" assigns column with ASSIGN TEXT

(2) SC2

Index of the parameter (e.g. 120 for @120)

SC1/PCN/CCN

Automatic entry by editor = 0/9980/0

Variable part:

(3) (Dialog column)

Any term after PCM syntax. Here assignment term for assigning the variables. Comment after assignment term possible by "%", example:
 BO_23%Nom value idf.

NOTE

The text parameter is replaced by the text, this is why you must keep enough space free (blanks). This also applies to UMESS formulas, see examples.

Transfer numerical parameters to text parameters

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	(3) STR(I)%Loop String	"==" (1) ASSIGN TEXT	(2) 126	0	9980	0	

Numerical parameters can be assigned a text parameter with the *STR* conversion function - and thus displayed:

STR (X) : X = convertible value (left justified output without decimal places)

Example

Display of the loop counter in a text line:

```
STR (I) ASSIGN TEXT 126
Borehole @126      Text ...
```

in a loop from 1 ...3 results in the text lines

Borehole 1

Borehole 2

Borehole 3

Select decimal places by format information:

STR (X,S,K)

X	value to be converted
S.K	format
S	mantissa of the value string
K	decimal places of the value string (right justified placing within the converted string with S places)
With s < 0	leading zeros preceding.
With s = -101	special KUM code

Examples

- Display of numerical value with decimal point :
 -12.6943 Assignment 45 and -853.2796 assignment 78
 STR (P45) results in -13
 STR (P45,8.1) results in -12.7
 STR (P78,8.1) results in -853.3
- UMESS formula with PCM:
 - Converting variable numerical values to text parameters
 @5 = STR(P250,8.1)
 If P250 = -853.3, then __-853.3 is entered
 If P250 = -10000.125, then -10000.1 is entered
 - Entering text parameter in formula, **keep space free !**
Wrong: @5+x(30)
Right: @5-----+x(30)

UMESS Option 9 is required when you want to parameterize your own texts. UMESS Option 10 suffices for the assignment of text parameters for Zeiss macros.

Measurement result parameters

Measurement result parameters enable access to measurement results. The options for linking and converting and the scope of application correspond to the P parameter. If a parameter is requested from a geometric element which does not exist (e.g. MD with a surface), this is assigned 99999.

Possible parameters:

MX, MY, MZ, MD, MD1, MD2, MW1, MW2, MWK

MS (dispersion)

MT (deviation T-act with DIN)

Example for the assignment of the cone angle of address 275 to P25:

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	MWK(275)	ASSIGNMENT	125	0	9979	0	

Example for the output of all diameters from address 55 to 58:

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	55 ¼ 58 ¼ 1 ¼ 0	DO LOOP	3	1	9941	1941	
..	STR(I)	ASSIGN TEXT	4	0	9980	0	
..	STR(MD(I),8,2))	ASSIGN TEXT	5	0	9980	0	
..	Diam @4 = @5	TEXT	0	1	1676	0	
..		END LOOP	1003	1	9949	1949	

```
Diam55 = 99.23
Diam56 = 101.05
Diam57 = 100.62
Diam58 = 98.45
```

Permissible functions and operations

Functions / Operations	Remarks
+, -, *, /, ^	
SIN, COS, TAN, ASN, ATN	Information in degrees, e.g. SIN(60)
SQR, LN, LOG, EXP	
ABS, INT, SGN, MOD (A,B)	MOD only possible without linking within MOD bracket
OR, AND, NOT	NOT always with subsequent bracket NOT(..)
>, >=, <, <=, #, =	
Numerical parameter P ...	With direct index e.g. P476 or with term as index, e.g. P(45+I)
Text parameter @(XX)	xx = Number of the text parameter

- Linkings of functions, operations, parameters and numbers result in a PCM term, e.g.: SIN(P34)+P(23+5)/2
- Functions always with subsequent bracket, e.g.: SIN(60)
- No direct sequence of functions or operations without bracket, e.g.: P20/(-5) permitted P20/-5 not permitted
- Nesting depth for brackets: 10
- Text parameter @(XX)
Autom. value function, i.e. numerical access to text parameters is also possible if the text parameter contains a number.

VPII parameter

VPII = variable record header II,

VPII parameters are special text parameters which can only be accessed.

Syntax

@S (XXX) XXX = parameter as per description for VPII

Example

@S (H2) gives data

For possible parameters

see VPII description

Application

Parameter input via record header.

Chapter

2

PCM functions

This chapter contains:

Definition of a new coordinate system	2-2
Definition of a new coordinate system	2-2
Coordinate transformations for parameters	2-4
Modi for coordinate transformation and other control data manipulations	2-7
FOCUS PCM functions	2-11
Dialog functions (parameter input).	2-13
Reading ASCII files with PCM syntax	2-15
Reading ASCII files without PCM syntax	2-17
Reading/editing/outputting ASCII files for string editing	2-20
Other help functions	2-24
Vector functions.	2-26

Definition of a new coordinate system

The new coordinate system applies automatically to the transformation functions `TOOLDCO`, `TONEWCO` and when the transformation mode is activated for all the following parameterized intermediate positions and probings.

Tilt of the coordinate system about 2 projected angles (ROTATE SPACE)

The new axis defined in this way always lies in the Z-axis independent of the PCN

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	ROT(A1, A2, PCN)	PCM	0	1	9972	0	

A1 Parameter number (PN) angle 1
 A2 Parameter number (PN) angle 2
 PCN Parameter number plane code number

Input example

The P variables are P502, P503, P504

Input ROT(502, 503, 504)

ROTATE PLANE according to point input about the axis defined by `ROT`.

Point lies after rotation in ZX plane.

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	ROTPT(X,Y,Z)	PCM	0	1	9972	0	

Range of rotation +90 ... -89.99999 degrees

X,Y,Z Parameter number coordinates of the point

Definition of the new coordinate origin (ZERO POINT)

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TRXYZ (X, Y, Z)	PCM	0	1	9972	0	

X, Y, Z Parameter number of the new coordinate origin.

Application example

Measurement module for line with rough position in X, probing direction vertically to probing surface

ROT with A1 and A2, PCN = 1

ROTP with any point on the probing surface (not on line)

TRXYZ with 1st probing point

Loop over length of line with start value = 0, end value = length of line, step size = pitch

Z-POS with 0, PT(PrbRad) + backmove path, I (loop)

PRB with 0, PT(PrbRad) I (loop)

By entering the PCN (axis) with the ROT function, this measurement module can also be used for axes with Y and Z orientation without this leading to a case differentiation for the I-POS and PRB.

Coordinate transformations for parameters

Transformation of a parameter triple *from the new to the old coordinate system (A position of the control data)*

Definition of the coordinate system see above.

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TOOLDCO (FROM, TO)	PCM	0	1	9972	0	

VON PN of the X value in the transformed system (input value)

TO PN of the X value in the original system (output value)

The respective Y, Z values are in the following parameters.

Transformation of a parameter triple *from the old to the new coordinate system (A position of the control data)*

Definition of the coordinate system see above. Syntax as for TOOLDCO.

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TONEWCO (FROM, TO)	PCM	0	1	9972	0	

Conversion of polar coordinates to right-angled coordinates

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TOREC (A,B,C,D)	PCM	0	1	9972	0	

Input values:

A PN radius

B PN angle in degrees

Output values:

C PN X value

D PN Y value

Conversion of right-angled coordinates to polar coordinates

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TOPOL(A,B,C,D)	PCM	0	1	9972	0	

Input values:

A PN X value

B PN Y value

Output values:

C PN radius

D PN angle in degrees

Conversion of sphere coordinates to cartesian coordinates

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TOXYZ(A,B)	PCM	0	1	9972	0	

Input values:

A PN radius

PN+1 tilt angle in degrees from Z

PN+2 rotation angle in degrees from X

Output values:

B PN X value

PN+1 Y value

PN+2 Z value

Conversion of cartesian coordinates to sphere coordinates

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TOSPH(A,B)	PCM	0	1	9972	0	

Input values:

- A PN X value
- PN+1 Y value
- PN+2 Z value

Output values:

- B PN radius
- PN+1 tilt angle in degrees from Z
- PN+2 rotation angle in degrees from X

Modi for coordinate transformation and other control data manipulations

Transformation mode ON/OFF

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TRMOD(0/1)	PCM	0	1	9972	0	

The transformation mode activated causes a conversion of all subsequent parameterized intermediate positions and probings in the control data from the new to the old coordinate system. This command is activated automatically by the commands ROT, ROTPT and TRXYZ.

Applications

- With reference to both coordinate systems the transformations for parameters are used (TOOLDCO) to execute a correction of the coordinates in the old system. For positions calculated in this way, the transformation mode must be deactivated.
- Deactivation of the mode at the end of the measurement mode.

Polar mode OFF/PCN

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	POLMOD(0/1/2/3)	PCM	0	1	9972	0	

Conversion of 2 coordinates of the control data line of radius and angle to right-angled coordinates with intermediate position and probings.

POLMOD(0)	OFF
POLMOD(1)	Conversion of Y, Z (source and target)
POLMOD(2)	Conversion of Z, X
POLMOD(3)	Conversion of X,Y

Application

Simple handling of rotational elements for which the radius remains constant (circle, cylinder) and for which the input of probings often takes place with angle steps.

Mode for probing direction ON/OFF

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	PRBDIR(0/1)	PCM	0	1	9972	0	

If the mode is active, the probing direction is redetermined with param. probings. The new probing direction results from the largest components of the last probe route (parameterized intermediate position or parameterized probings!) in the original coordinate system. The other part of the probing code number remains unchanged.

Application

Simple handling of elements inclined in space and of round elements (use of POLMOD). The mode is practically obligatory for all general PCM measurement modules.

Mode influences address offset with <DI 1693>.

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	IN_WORKPIECE(0/1)	PCM	0	1	9972	0	

Overwrites the code of the record header for the reset of the address offset.

Do not append an index to the result name within loops!

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	NAM_MOD(0/1)	PCM	0	1	9972	0	

Application

If the result name is not to be changed, e.g. when entering from ASCII files.

Mode EXCALL address correction ON/OFF

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	EXCCOR(0/1)	PCM	0	1	9972	0	

If the mode is active, with an EXCALL run (not in the main program) the current address of the previous run is added to the address from the control data (with measurement mode of geometric elements always 1).

The control data line with the address (N POINT TERMIN) must contain at least one parameter. If the address is parameterized, the parameter value is added. In this case, no address offset takes place.

Application

Activate when using EXCALL as measurement mode call. In the self-teach programming, an address gap appears in the control data of the main program. With a CNC run, the address of the measurement mode element is inserted into the ascending addresses of the main program without a gap. If the mode is not active, the address is taken from the measurement mode (usually 1).

Keeping diverse lines for the run control

These are usually cleared during the generation run <DAW 1647> (branchings, parameter assignments).

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	GEN_FILT_OFF(X)	PCM	0	1	9972	0	

- X = 1 The next line is not cancelled
- X = 2 The next lines are not cancelled until the generation is deactivated
- X = 0 Deactivate

Application

Branchings for which measurement results are referred to should not be cancelled. During the generation run, the IF branching will run independent of the measurement result.

NOTE

Do not use ELSE or ELSIF. Avoid inconsistencies with these parameters, e.g. due to cancelled assignments, but not cancelled branchings.

Example

```
GEN_FILT_OFF(1)
IF X(233)>120.5 THEN
..
..
GEN_FILT_OFF(1)
ENDIF
```

Nominal value filter for filtering individual components of the following nominal block <DI 1459>.

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	SETNOM(TERM)	PCM	0	1	9972	0	

Possible components of the term

X Y Z D \$4 \$5 \$6 R, whereby \$4, \$5, \$6 stand for A1, A2, AC.

Application

SETNOM before the nominal value block within a generally valid macro which contains all possible components. With TERM, the desired components are specified.

Example

SETNOM(XD) Nominal value block with (XYZD) .

Only the nominal values for x and D are output.

Hold function

If the term also has the character "*", then SETNOM applies to all following nominal blocks until deactivated in another term with the character "-".

FOCUS PCM functions

NOTE

The following FOCUS PCM functions listed are part of the UMESS Opt. 17 **FOCUS**. They are described in the corresponding manual.

DIA INP	Manual feature selection during CNC start
READF (XXX)	Reading of selected features as ASCII file
READADR (XXX)	Reading of selected addresses as ASCII file
PATH (XXX)	Input of path (directory) for READ, READF, READFADR
FOC_NEW	Label as FOCUS program (with new probe route definition)
KMG_TYP	Assignment of system variables for FOCUS during ACE
LISTI (1)	Activation of control output of generated probe routes
LISTI (4)	Display of the addresses activated by feature selection and recall (display after feature selection)
FSET_OUT (ADR) or FSET_OUT (FROM, TO)	Manual activation of addresses so that they are measured and output
FSET_MES (ADR) or FSET_MES (FROM, TO)	Manual activation of addresses so that they are measured
NO_CONF (CONF, CP, CP, CP...)	Blocking of clearance planes (CP) for certain probe configurations
RT_MOD (XZ90)	Setting of rotary table behaviour on the FC
ROUT_GEN (0)	Deactivate probe route generation (1 = activate)
ACLEAR (FROM, TO)	Deletion of address range (we recommend that this line is inserted at the start of the run to ensure a defined initial status)

Possible queries (variable part) with IF branchings

MES (0) =1	Execute the following IF brackets, if the next address has been activated (directly by feature selection or by reference)
MES (X) =1	Execute the following IF bracket if the specified address (x) has been activated
MES (X, Y) =1	Execute the following IF bracket if at least 1 address of the address range has been activated (x = from ADR, y = to ADR)
OUT (0) >0	Execute the following function if the output of the next address has been activated (by feature selection)
GRP (0) =1	Bracketing a group
GRP (-1) =1	Bracketing a group which is always processed (even without feature selection)

GRP(-2)=1

Bracketing a group which is always processed completely if at least 1 element in the group has been activated by feature selection or reference

Dialog functions (parameter input)

Start internal macros

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	MACRO START	PCM	1	1	9951	1951	

End internal macros

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	MACRO END	PCM	1	1	9951	1951	

Identification of dialog start

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	INPUT START	PCM	1	1	9951	1951	

Identification of dialog end

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	INPUT END	PCM	1	1	9951	1951	

A dialog page is generated from the assignment lines between `INPUT START` and `INPUT END` for which parameters can be entered in the CNC measuring run / PCM measurement module call. A max. of 50 assignment lines is possible.

The comments in the assignment lines form the identification of the respective input field.

Application

- The input of parameters with CNC start as replacement of PCM files with parts families, for selecting part runs with feature and element related measurement.
- Identification of the dialog part with PCM measurement modules.

Mode for preassignment for parameter input

with INPUT START / END

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	INP_MOD(0/1)	PCM	1	1	9951	1951	

INP_MOD(1) Preassignment from control data

INP_MOD(0) Preassignment from last input

Reading ASCII files with PCM syntax

Reading of PCM parameters as ASCII file

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	READ (XXX)	PCM	0	1	9972	0	

Parameter assignments can be read as ASCII file during a CNC run using a READ command. The name can be kept variable by text parameters.

Application

The assigned parameters can select parts of a complete run (with the IF command) and operate variants of a parts family by variable control data (parameterized probe routes, nominal values, ...). When using READ, the PCM file used with parts families is not usually necessary.

The READ command is used in particular if the part-specific parameters are generated by a host computer and transferred to the UMESS computer with DATACOM. Conversion to a PCM file necessary till now is no longer required.

Form of the assignment files

ASCII file with assignment and comment lines

Numerical parameters

PX = ...

Text parameters

TX = ...

Abbreviated notation

PG = ... for assignment of several parameters

Example of an assignment file

```
%      Turbine blade  T6-32-83 right
%      -----
%      Last modification  10.5.95
%
P6  = 3          % A pos. no.
P21 = 1          % Nominal no.
P41 = 2          % Run code      1 = simple measurement
%                               2 = complete measurement
%                               with KUM
P43 = 5          % Blade type
%
P45 = 20.8        % Width
P46 = 0.1         %      UT
P47 = -0.1        %      LT
%
PG33 = 14 , 76 , 55 % collar width 2 , UTOL ,LTOL
(P33 , P34 and P35 are assigned)
.
T45 = KR_68       % Feature identification
```

NOTE

Note on PG33: P33 = 14, P34 = 76, P35 = 55

Input of a different path to udir for reading in

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	PATH (XXX)	PCM	0	1	9972	0	

Applies to READ, READF, READFADR, AREAD, AWRITE .

Example

```
/users/zeiss/DIR1/      ASSIGN TXT      5
PATH(@5)                PCM
READ(FILE_X)            PCM
```

-> File FILE_X on **/home/zeiss/DIR1** is read.

Reading ASCII files without PCM syntax

Reading of an ASCII file with special format

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	FREAD (NO, FILE)	PCM	0	1	9972	0	

The FREAD function is recommended for large amounts of data in tabular form.

Advantage

Existing ASCII files no longer have to be converted to PCM form. For application, see READ function

NO Number of the format file on ../udir

FILE File name of the ASCII file

Specification of the numbers of the start parameters to which the values from the ASCII file are assigned.

This specification applies to a block.

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	STARTPAR (BL, SP, SPIF)	PCM	0	1	9972	0	

BL Block no.

SP Start parameter in which the 1st value of the block is stored. With index storage, the value is stored displaced about the index.

SPIF Start parameter of the index sequence. Without INDEX sequence, assign 1 to SPIF parameter.

Examples

for FREAD (standard case)

ASCII file to be read in **FR_FILE**, blank or comma possible as separator, no division according to columns required.

```
10 20 30 40
50 60 70
80          90
100         110
```

or

```
10,20, 30, 40
50,60, 70
80    ,    90
100,110
```

Format file (ASCII) to udir FORMAT2 (2 = number for FREAD)

```
BL_STA=NEXT
BL_DEF=(FREE)
BL_END=NEXT
```

Control data

1	PCLEAR	PCM	0	1	9972	0
2	STARTPAR(1,2501,1)	PCM	0	1	9972	0
3	FREAD(2,FR_FILE)	PCM	0	1	9972	0
4	PLIST	PCM	0	1	9972	0

PCLEAR / PLIST for testing the assigned parameters results in:

```
P2501 = 10.00000
P2502 = 20.00000
P2503 = 30.00000
P2504 = 40.00000
P2505 = 50.00000
P2506 = 30.00000
P2507 = 40.00000
P2508 = 80.00000
P2509 = 70.00000
P2510 = 100.00000
P2511 = 110.00000
```

Example

of the value assignment for FREAD with index sequence STARTPAR (special case)

```
STARTPAR(2,301,101)
```

Contents of 2nd block of the ASCII file Resulting param. assignment

Index				Index sequence			
1	68	64	65	P101	P301	P302	P303
2	68	63	64	P102	P304	P305	P306
5	35	54	47	P103	P313	P314	P315
9	33	52	72	P104	P325	P326	P327

Application

Branched access to data

Calculation formula

of the assigned parameter number with INDEX storage

ParNo = (Index - 1) * (Parameter per line - 1) + Start parameter

Reading/editing/outputting ASCII files for string editing

Reading an ASCII file into internal buffer

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	AREAD (XXX)	PCM	0	1	9972	0	

XXX = File name of the ASCII file on **/users/zeiss/udir** or from another directory (see PATH)

The file can have a maximum of 3000 lines each with 80 characters

The line number of the file read is in P990

The internal buffer can be recopied to text parameter by ATCOPY.

Writing the internal buffer as ASCII file

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	AWRITE (XXX)	PCM	0	1	9972	0	

The line number of the file has to be in P991, otherwise as for AREAD.

Output of the internal buffer in the record

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	AOUT (FROM, TO)	PCM	0	1	9972	0	

FROM From line ...

TO To line ...

Line by line copying of the ASCII buffer to text parameters for further editing

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	ATCOPY (SOURCE, FROM, TO, TARGET, FROM, NO)	PCM	0	1	9972	0	

SOURCE	Line number of the ASCII buffer
FROM	Position of the 1st character read within the line
TO	Position of the last character read within the line
TARGET	Number of the text parameter which is to be assigned
FROM	Position within the text parameter where writing is to start
NO.	Number of lines to be copied (input optional)

Line by line copying of text parameters to the ASCII buffer

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TACOPY (SOURCE, FROM, TO, TARGET, FROM, NO)	PCM	0	1	9972	0	

Syntax as for ATCOPY, however SOURCE is the text parameter and TARGET is the ASCII buffer.

Character string processing

I. e. recopying of parts of a text parameter to another text parameter:

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TCOPY (SOURCE, FROM, TO, TARGET, FROM, NO)	PCM	0	1	9972	0	

SOURCE	Number of the source text parameter
FROM	Position of the 1st character read
TO	Position of the last character read
TARGET	Number of the target text parameter which will be assigned
FROM	Position within the text parameter where writing is to start
NO.	Number of the text parameters to be copied (input optional)

Example

```
@20 = ABCDEFGH
@30 = xyz

TCOPY(30,1,2,20,5)

gives 'ABCDxyGH' in @20
```

Comparison of 2 text parameters

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	EQU (TPN1, TPN2)	ASSIGNMENT	200	0	9979	0	

If equal, 1 is returned otherwise 0.

TPN1, 2

Number of the 1st / 2nd text parameter respectively

NOTE

EQU is not a PCM function (9972) but available in numerical linking terms, e.g. with assignments (9979); ► "Permissible functions and operations" on page 1-21

Position of the 2nd text parameter in the 1st text parameter

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	POS (TPN1, TPN2)	ASSIGNMENT	200	0	9979	0	

TPN1, 2

Number of the 1st / 2nd text parameter respectively

NOTE

POS is not a PCM function (9972) but available in numerical linking terms, e.g. with assignments (9979); ► "Permissible functions and operations" on page 1-21

Example

```
@20 = ABCDEFGH
@30 = EF

EQU(20,30)    gives 0 as function value
POS(20,30)    gives 5 as function value
```

Division of a long text parameter into 4 text parameters for output with text data lines

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TSPLIT(TPN1,TPN2)	PCM	0	1	9979	0	

TPN1 Number of the parameter to be divided

TPN2 Number of the 1st target parameter

From the 1st target parameter, 4 text parameters are assigned in ascending order.

```
@15 is to be output with 80 characters
TSPLIT(15,61)          PCM
@61                    TEXT
@62                    DL TEXT
@63                    DL TEXT
@64                    LDL TEXT
```

Other help functions

Initializing the P parameters

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	PCLEAR	PCM	0	1	9972	0	

Listing of all P parameters currently assigned during the PCM test run

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	PLIST	PCM	0	1	9972	0	

For help during error search in PCM runs and measurement modules.

Example

Call before the PCLEAR measurement module and after the PLIST measurement module. All parameters are output which have been assigned during the run of the measurement module.

NOTE

Parameters in program parts which have been factored out with IF or which are not assigned for other reasons are not listed.



Attention!

Only include PCLEAR and PLIST in runs for testing!

Value assignment of a parameter range

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	PSET (FROM, TO, PNR)	PCM	0	1	9972	0	

FROM No. of the start parameter

TO No. of the end parameter

PNR Parameter no. of the value to be assigned

Application

Simple preassignment of several parameters with the same value.

Example for PSET

```
P20 = 1 % value
PSET(300,302,20)
Result
P300 = 1
P300 = 1
P302 = 1
```

Moving parameters block by block

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	PMOVE (QU1, ZI1, ANZ)	PCM	0	1	9972	0	

QU1 No. of the 1st source parameter
 ZI1 No. of the 1st target parameter
 ANZ Number of parameters to be moved

Moving text parameters block by block

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	TMOVE (QU1, ZI1, ANZ)	PCM	0	1	9972	0	

TMOVE (QU1, ZI1, ANZ) corresponding to PMOVE

Deletion of an address range (from... to...)

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	ACLEAR (FROM, TO)	PCM	0	1	9972	0	

Used for avoiding **contamination effects** and insertion is recommended at the start of the run.

Display of the EXCALL jumps

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	LISTI (7)	PCM	0	1	9972	0	

Vector functions

Standardized vector from 2 points

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	VE_2PTNOR (P1,P2,VE)	PCM	0	1	9972	0	

P1 I SPN 1st point triple

P2 I SPN 2nd point triple

VE0 SPN standardized vector P2 - P1

SPN Index start parameter, 2nd and 3rd coordinate subsequently, see also example for cross product of a vector (VE_CROSS (VE1,VE2,VE0))

I Input parameter

O Output parameter

Standardization of a vector

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	VE_NOR (VEI,VEO)	PCM	0	1	9972	0	

VEI I SPN input vector

VEO O SPN standardized vector

Cross product of a vector

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	VE_CROSS (VE1,VE2,VEO)	PCM	0	1	9972	0	

VE1 I SPN 1st input vector

VE2 I SPN 2nd input vector

VEO O SPN cross product vector

Input example:

Input: `VE_CROSS(502,602,702)`

Explanation: P502 is the start parameter NX of the first input vector,
to this belong NY = P503 and NZ = P504
P602 is the start parameter NX of the second input vector,
to this belong NY = P603 and NZ = P604
P702 is the start parameter NX of the cross product vector,
to which belong NY = P703 and NZ = P704

Standardized surface normal from 3 points

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	<code>VE_3PTNOR(P1,P2,P3,VE)</code>	PCM	0	1	9972	0	

P1 I SPN 1st point triple
P2 I SPN 2nd point triple
P3 I SPN 3rd point triple
VE O SPN standardized surface normal

New point from point, direction vector, distance

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	<code>VE_PTVEPT(PI,VE,DI,PO)</code>	PCM	0	1	9972	0	

PI I SPN starting point
P2 I SPN direction vector
DI I SPN distance from starting point to new point
PO O SPN new point

Perpendicular point and standardized perpendicular vector

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	VE_PERP1 (P1,VE1,P2,LP,LVE)	PCM	0	1	9972	0	

PI I SPN starting point

VE1 I SPN starting vector

P2 I SPN point for perpendicular (not on starting vector)

LP O SPN perpendicular point (on starting vector)

LVE O SPN standardized perpendicular vector (points from LP to P)

Total of 2 vectors

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	VE_SUM (VE1,VE2,VES)	PCM	0	1	9972	0	

VE1 I SPN 1st vector

VE2 I SPN 2nd vector

VES O Total of vectors

Difference between 2 vectors

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	VE_DIF (VE1,VE2,VED)	PCM	0	1	9972	0	

VE1 I SPN 1st vector

VE2 I SPN 2nd vector

VED O Differential vector

Mean value of dimensional values

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	VE_MEAN(K1,K2,DIM,ANZ)	PCM	0	1	9972	0	

- K1 I SPN 1st component of the value, the following components subsequently. The components of the next value after the last component of the previous value.
- K2 O SPN 1st component of the mean value
- DIM I Number of dimensions (components). For point with XYZ = 3.
- ANZ I No. of values (points)

Example

Mean value of 4 points

```
P501 P502 P503 XYZ 1st point
P504 P505 P506 XYZ 2nd point
P507 P508 P509 XYZ 3rd point
P510 P511 P512 XYZ 4th point

VE_MEAN(501,201,3,4)

P201 P202 P203 XYZ of the mean value
```

Space distance between 2 points

No.	Dialog	Function	SC2	SC1	PCN	CCN	ADR
..	VE_DIST(P1,P2,DIST)	PCM	0	1	9972	0	

- P1 I SPN 1st point (vector)
- P2 I SPN 2nd point (vector)
- DIST O PN space distance (differential vector)

Chapter 3

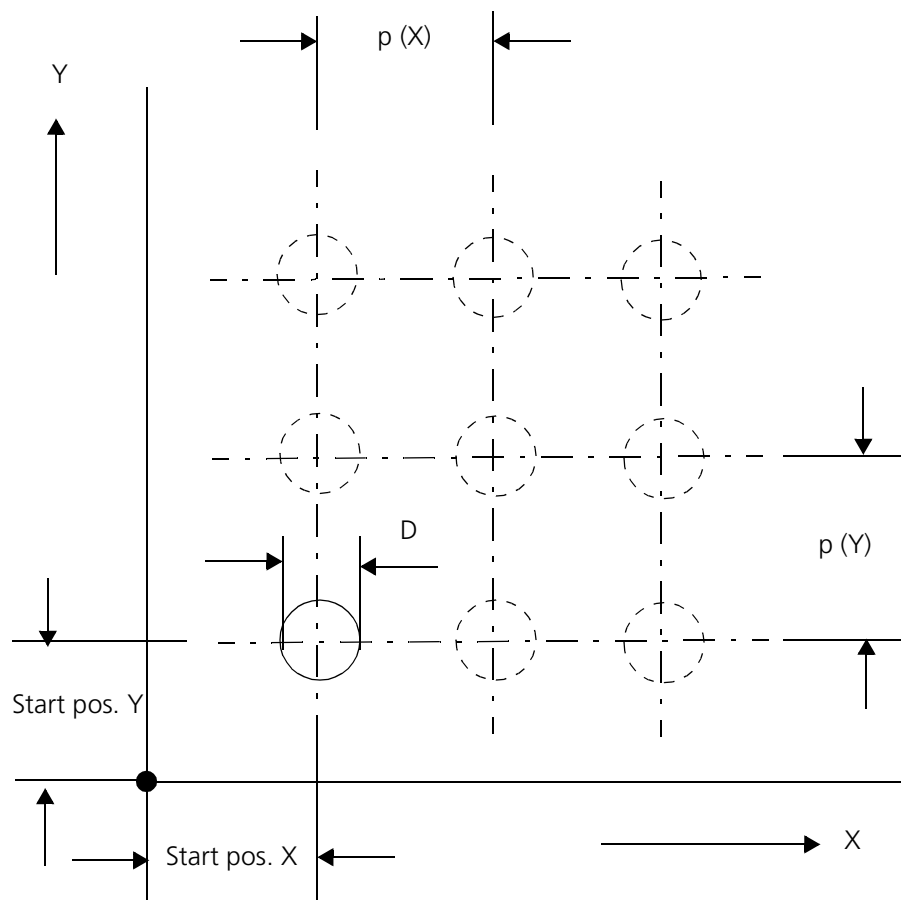
Application example: Bore plate

This chapter contains:

Technical data of the bore plate	3-2
Bore plate parameters	3-3
Structure of the CNC program	3-4
Control data list	3-5
Explanation of the control data	3-7

Technical data of the bore plate

The application example is based on a bore plate whose bore pattern shows a uniform spacing in each direction. The diameter of the boreholes is constant:



Bore plate parameters

Before the start of the measuring run, the parameters of the measuring plate are requested:

Enter parameters for macro

Identification	Value
Bore diameter	10
Start coord. X	20
Start coord. Y	20
Start coord. Z	20
No. of rows Y	5
No. of bores/row X	5
Spacing p direction X	20
Spacing p direction Y	20

* YES

NO

*

CATALOG

TERMIN

BACK

INFO

NOTE

In the example it is assumed that the bore plate variants have the same thickness. Therefore a parameterization of the intersection height is not necessary.

Structure of the CNC program

Alignment, definition of the control and workpiece coordinates
(omitted in the example)

Control data lines

Query for bore plate parameters 6 - 15

Definition of the pos. counter for X and Y 17 - 18

Setting the start positions for X and Y 20 - 21

Outer loops

Inner loops

– Call of CIRCLE PCM macro 28 - 64

– Control of X position counter 66

– Correction of X position counter 70

– Control of Y position counter 71

– Reversal of X travel direction 72

Control data list

NO	X	Y	Z	Function	SC2	SC1	PCN	CCN	ADR
Dialog				Idf	Sy	Nominal	U.tol	L.tol	
Snr	Record type	MEAdr	Idf	Sy	t	(M)	A	(M)	
1				BLANK		0	0	MMMM	MMMM
2				BLANK		0	0	MMMM	MMMM
3				BLANK		0	0	MMMM	MMMM
4				BLANK		0	0	MMMM	MMMM
5				BLANK		0	0	MMMM	MMMM
6	INPUT START			PCM		0	1	9972	0
7	0%Diameter borehole			ASSIGNMENT		10	0	9979	0
8	0%Start coord. X			ASSIGNMENT		11	0	9979	0
9	0%Start coord. Y			ASSIGNMENT		12	0	9979	0
10	0%Start coord. Z			ASSIGNMENT		13	0	9979	0
11	(-1)%No. rows Y			ASSIGNMENT		14	0	9979	0
12	(-1)%No. bore./row X			ASSIGNMENT		15	0	9979	0
13	0%Spacing direction X			ASSIGNMENT		16	0	9979	0
14	0%Spacing direction Y			ZUWEISUNG		17	0	9979	0
15	INPUT END			PCM		0	1	9972	0
16				BLANK		0	0	MMMM	MMMM
17	1 P14 1 0			DO LOOP		1	1	9941	1941
18	I			ASSIGNMENT		6	0	MMMM	MMMM
19	1 P15 1 0			DO LOOP		2	1	9941	1941
20	I			ASSIGNMENT		5	0	MMMM	MMMM
21	P11+(P5-1)*P16			ASSIGNMENT		20	0	9979	0
22	P12+(P6-1)*P17			ASSIGNMENT		21	0	9979	0
23	1			EXCALL-ADR		0	1	1691	0 1
24	MEAS(0)=1			IF		0	1	9951	1951
25	INPUT START			PCM		0	1	MMMM	MMMM
26	CIRCLE MACRO			CTEXT		0	1	1679	0
27	BORE %RESULT NAME			ASSIGN. TEXT		700	0	9980	0
28	3% PLANE CODE NO.			ASSIGNMENT		700	0	9979	0
29	1% PROBE NO.			ASSIGNMENT		750	0	9979	0
30	1% OUTER=0/INNER=1			ASSIGNMENT		764	0	9979	0
31	P20 % NV X-COORDINATE			ASSIGNMENT		701	0	9979	0
32	P21 % NV Y-COORDINATE			ASSIGNMENT		704	0	9979	0
33	P13 % NV Z-COORDINATE			ASSIGNMENT		707	0	9979	0
34	P10 % NV D DIAMETER			ASSIGNMENT		710	0	9979	0
35	-10% START VALUE I-POS			ASSIGNMENT		775	0	9979	0
36	-20% END VALUE I-POS			ASSIGNMENT		776	0	9979	0
37	5% DIST : I-POS/PRB			ASSIGNMENT		749	0	9979	0
38	0% START ANGLE			ASSIGNMENT		795	0	9979	0
39	360% END ANGLE			ASSIGNMENT		796	0	9979	0

Application example: Bore plate

=====											
NO	X	Y	Z	Function			SC2	SC1	PCN	CCN	ADR
	Dialog										
	Snr	Record type		Idf	Sy	Nominal		U.tol	L.tol		
	Snr	MEAdr		Idf	Sy	t		(M)	A	(M)	
=====											
40	4% NO. OF PRB POINTS			ASSIGNMENT			781	0	9979	0	
41	5% INTERSEC. HEIGHT			ASSIGNMENT			719	0	9979	0	
42	X Y Z D	% NV SELECTION		ASSIGN. TEXT			705	0	9980	0	
43		% NV X-IDF		ASSIGN. TEXT			701	0	9980	0	
44		% NV Y-IDF		ASSIGN. TEXT			704	0	9980	0	
45		% NV Z-IDF		ASSIGN. TEXT			707	0	9980	0	
46		% NV D IDF		ASSIGN. TEXT			710	0	9980	0	
47		.1% UT X		ASSIGNMENT			702	0	9979	0	
48		-.1% LT X		ASSIGNMENT			703	0	9979	0	
49		.1% UT Y		ASSIGNMENT			705	0	9979	0	
50		-.1% LT X		ASSIGNMENT			706	0	9979	0	
51		.1% UT Z		ASSIGNMENT			708	0	9979	0	
52		-.1% LT Z		ASSIGNMENT			709	0	9979	0	
53		.05% UT D		ASSIGNMENT			711	0	9979	0	
54		-.05% LT D		ASSIGNMENT			712	0	9979	0	
55	*****			CTEXT			0	1	1679	0	
56	0	% CE-MODE OFF		ASSIGNMENT			690	0	9979	0	
57	*****			CTEXT			0	1	1679	0	
58	INPUT END			PCM			0	1	MMMM	MMMM	
59	CIRCLE.CAL(Z1)			EXCALL			0	1	9971	1971	
60				ENDIF			0	1	9959	1959	
61				END LOOP			1002	1	9949	1949	
62				END LOOP			1001	1	9949	1949	
63				P-END			0	0	9999	1999	

Explanation of the control data

Control data line	Example
6 - 15	Between the INPUT START and INPUT END lines, those parameters are defined which are to be requested of the user at the start of the CNC run. The preassignment of the parameter is specified at the start of the line, the text after the percentage character is displayed in the input mask.
17	Start of the outer loop.
19	Start of the inner loop.
21	Incrementation of the X position counter by 1 spacing.
22	Incrementation of the Y position counter by 1 spacing.
23	By calling <DI 1079> input of the dialog file for the circle measurement.
28 - 41	Control parameters for the circle measurement. If necessary, these parameters can also be entered at the start of the CNC run. See control data lines 31-34.
42 - 54	Evaluation parameters for the circle measurement. If necessary, these parameters can also be entered at the start of the CNC run.
58	End of the dialog file for the circle measurement.
59	Program branching to the CIRCLE measurement module in the Z1 control data catalog.
61	Start of the inner loop.
62	Start of the outer loop.
63	End of the CNC measuring program.

Index

Symbole

<PCM> 4, 1-10

A

Access to measurement results 1-20
ACLEAR(FROM,TO) 2-25
Activating the PCM edit mode 1-10
Activating the PCM run mode 1-11
Actual probe radius 1-17
Address counter 1-17
AOUT(FROM,TO) 2-20
AREAD(XXX) 2-20
ASSIGN TEXT 1-18
ASSIGNMENT 1-7
Assignment file 2-15
Assignment line 1-15
ATCOPY(SOURCE,FROM,TO,TARGET,FROM,NO) 2-21
Avoid data input 1-8
AWRITE(XXX) 2-20

B

Bore plate parameters 3-3
Bracket 1-21
bracketing 1-7
Branchings 1-2, 1-16

C

calling the measurement module 1-9
CCN 1-15
Character string processing 2-21
CNC measuring runs for parts families 1-5
combined measuring tasks 1-2
Comparison of 2 text parameters 2-22
Constants p (Pi) 1-17
Control data list 3-5
Conversion of cartesian coordinates to sphere coordinates 2-6
Conversion of polar coordinates to right-angled coordinates 2-4
Conversion of right-angled coordinates to polar coordinates 2-5
Conversion of sphere coordinates to cartesian coordinates 2-5
Convert masked lines 1-14
Convert parameter lines 1-14

Coordinate transformations for parameters 2-4
Copying of the ASCII buffer to text parameters 2-21
creating flexible measuring runs 1-8
creating PCM measurement modules 1-8
creating your own measurement modules 1-8
creating your own PCM measurement modules 1-9
Cross product of a vector 2-26

D

Definition of a new coordinate system 2-2
Deletion of an address range 2-25
DI 1459 2-10
DI 1642 1-8
DI 1646 1-8, 1-11
DI 1647 1-8, 1-13
DI 1666 1-8, 1-10
DI 1693 2-8
Dialog 1-15
dialog file 1-9
Dialog functions (parameter input) 2-13
Difference between 2 vectors 2-28
Display of the 2-25
Division of a long text parameter into 4 text parameters 2-23
DO LOOP loops 1-16

E

EQU(TPN1,TPN2) 2-22
EXCALL 1-8
EXCALL jumps 2-25
EXCCOR(0/1) 2-8
Execute CNC and calculate results 1-12, 1-14
Explanation of the control data 3-7

F

file names 1-18
Filtering individual components 2-10
Fixed part 1-15
FOCUS PCM functions 2-11
FREAD (NO,FILE) 2-17
Function 1-15

G

GEN_FILT_OFF(X) 2-9
generate parameterized control data 1-10
Generating a PCM run 1-5
Generating more PCM files 1-6
generating run 1-13
Generating the first PCM file 1-5
Geometric elements 1-2, 1-9
Global parameters 1-17
Global text parameters 1-18

H

Help during error search in PCM runs 2-24

I

IF/ENDIF 1-7
IN_WORKPIECE(0/1) 2-8
Initializing the P parameters 2-24
INP_MOD(0/1) 2-14
INPUT END 1-7, 2-13
Input of a different path 2-16
INPUT START 1-7, 2-13

K

Keeping diverse lines for the run control 2-9

L

Length of text parameters 1-18
Line by line copying of text parameters to the ASCII buffer 2-21
Linkings 1-21
LISTI(7) 2-25
Listing of all P parameters currently assigned during the PCM test run 2-24
loop counter 1-17
Loops 1-2, 1-16

M

MACRO END 2-13
MACRO START 2-13
Macros 2-13
Managing control data catalogs 1-9
Mean value of 4 points 2-29
Mean value of dimensional values 2-29
Measurement module library 1-9
Measurement result parameters 1-20
mm/Inch factor 1-17
Mode EXCALL address correction ON/OFF 2-8
Mode for probing direction ON/OFF 2-8
Mode influences address offset 2-8

Modi for coordinate transformation and other control data manipulations 2-7
Moving parameters block by block 2-25
Moving text parameters block by block 2-25

N

NAM_MOD(0/1) 2-8
Nesting depth 1-21
New point from point, direction vector, distance 2-27
No. 1-15
nominal identifications 1-18
Nominal value filter 2-10
Numerical parameters 1-17

O

Option 10 1-9
Option 8 1-9
Option 9 1-9
Other help functions 2-24
Output of the internal buffer in the record 2-20

P

P parameter 1-15
Parameter Control Manager 1-2
parameter-free 1-13
Parameterization of 1-16
Parameterization of your own control data 1-8
Parameters 1-17
Part number from record header 1-18
Part runs 1-7
parts family 1-3
parts-specific files 1-3
PATH (XXX) 2-16
PCLEAR 2-24
PCM development system 1-1
PCM edit mode 1-8
PCM File 1-3
PCM generating run (1-13
PCM library 1-9
PCM measurement module library 1-2
PCM run 1-3
PCM terms 1-10
PCM test run 1-11
PCN 1-15
Permissible functions and operations 1-21
Perpendicular point and standardized perpendicular vector 2-28
PLIST 2-24
PMOVE(QU1,ZI1,ANZ) 2-25
Polar mode OFF/PCN 2-7
POLMOD(0/1/2/3) 2-7

POS(TPN1,TPN2) 2-22
 Position of the 2nd text parameter in the
 1st text parameter 2-22
 PRBDIR(0/1) 2-8
 Principle of the variant programming 1-4
 PROG mode 1-9, 1-11
 Program in a modular structure 1-8
 program systems not capable of PCM 1-
 13
 Programming for parts families 1-3
 PSET(FROM,TO,PNR) 2-24

R

Random number between 0 ... 1 1-17
 READ(XXX) 2-15
 Reading an ASCII file into internal buffer
 2-20
 Reading ASCII files with PCM syntax 2-15
 Reading ASCII files without PCM syntax
 2-17
 Reading of an ASCII file with special
 format 2-17
 Reading of PCM parameters as ASCII file
 2-15
 Reading/editing/outputting ASCII files for
 string editing 2-20
 ROT(A1, A2, PCN) 2-2
 ROTATE PLANE 2-2
 ROTATE SPACE 2-2
 ROTPT(X,Y,Z) 2-2

S

SC1 1-15
 SC2 1-15
 Select decimal places by format
 information:
 1-19
 SETNOM(TERM) 2-10
 similar workpieces 1-3
 Space distance between 2 points 2-29
 Special elements 1-9
 Specification of the numbers of the start
 parameters 2-17
 Standardization of a vector 2-26
 Standardized surface normal from 3
 points 2-27
 Standardized vector from 2 points 2-26
 STARTPAR(BL,SP,SPIF) 2-17
 STR conversion function 1-19
 Structure of a control data line 1-15
 Structure of the CNC program 3-4

T

TACOPY(SOURCE,FROM,TO,TARGET,FR
 OM,NO) 2-21
 TCOPY(SOURCE,FROM,TO,TARGET,FRO
 M,NO) 2-21
 Technical data of the bore plate 3-2
 Test run 1-11
 Text parameters 1-18
 texts 1-18
 theoretically checking the program 1-9
 Tilt of the coordinate system 2-2
 TMOVE(QU1,ZI1,ANZ) 2-25
 TONEWCO(FROM,TO) 2-4
 TOOLDCO(FROM,TO) 2-4
 TOPOL(A,B,C,D) 2-5
 TOREC(A,B,C,D) 2-4
 TOSPH(A,B) 2-6
 Total of 2 vectors 2-28
 TOXYZ(A,B) 2-5
 Transfer numerical parameters to text
 parameters 1-19
 Transformation mode ON/OFF 2-7
 Transformation of a parameter triple 2-4
 TRMOD(0/1) 2-7
 TRXYZ(X,Y,Z) 2-3
 TSPLIT(TPN1,TPN2) 2-23

U

UMESS formulae 1-18
 UMESS Option 10 1-2
 UMESS Option 17 'FOCUS' 2-11
 UMESS Option 9 1-3

V

Value assignment for P parameters 1-15
 Value assignment to text parameters 1-
 18
 Value input for your own measurement
 modules 1-9
 Variable part 1-15
 variable record header II 1-22
 variables instead of fixed character
 strings 1-18
 Variant programming 1-3
 variant programming 1-2
 VE_2PTNOR(P1,P2,VE) 2-26
 VE_3PTNOR(P1,P2,P3,VE) 2-27
 VE_CROSS(VE1,VE2,VEO) 2-26
 VE_DIF(VE1,VE2,VED) 2-28
 VE_DIST(P1,P2,DIST) 2-29
 VE_MEAN(K1,K2,DIM,ANZ) 2-29
 VE_NOR(VEI,VEO) 2-26
 VE_PERP1(P1,VE1,P2,LP,LVE) 2-28
 VE_PTVEPT(PI,VE,DI,PO) 2-27

VE_SUM(VE1,VE2,VES) 2-28

Vector functions 2-26

VPII - parameter 1-22

W

Workpieces of a parts family 1-2

Writing the internal buffer as ASCII file 2-20

Z

ZEROPOINT 2-3