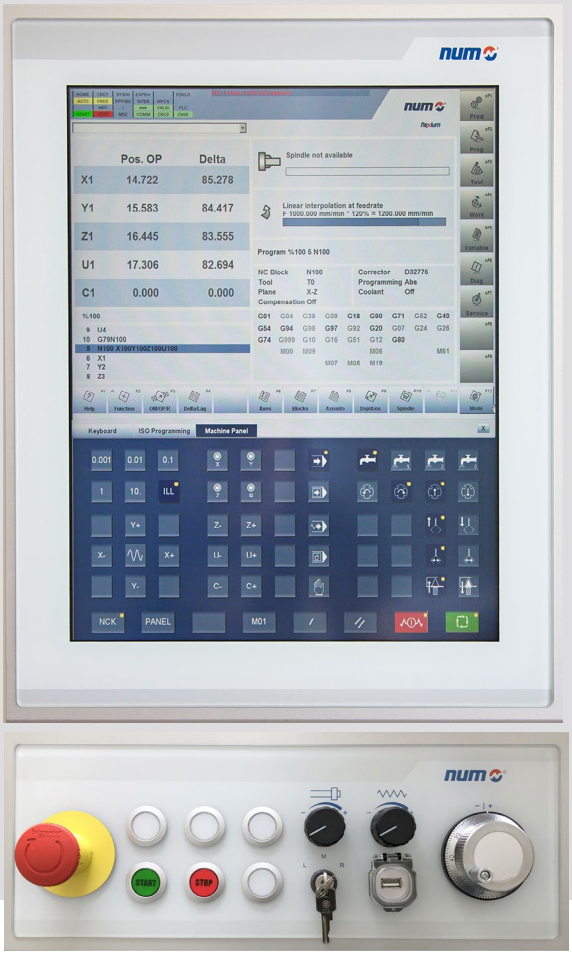
Flexium+

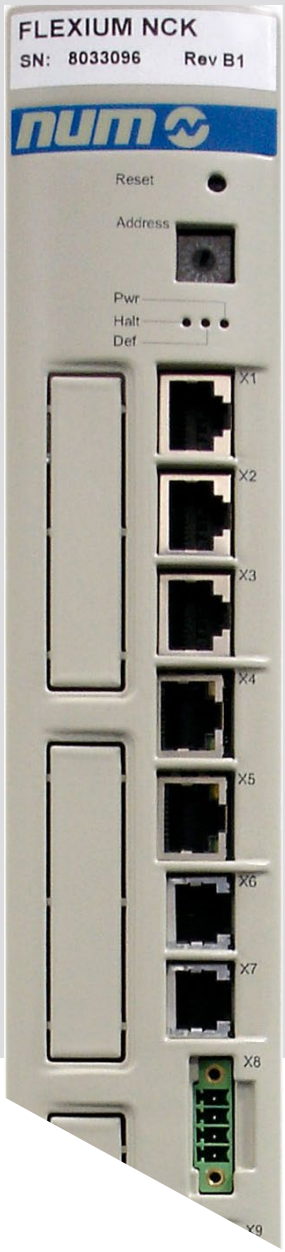
Programming manual

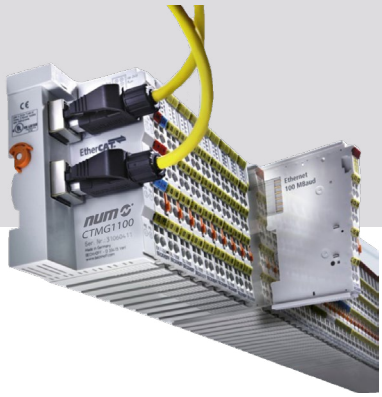
(for software release 4.2.0.0)

M00040EN-07

Edition 10.2022







www.num.com

num®
CNC HighEnd Applications

Despite the care taken in the preparation of this document, NUM cannot guarantee the accuracy of the information it contains and cannot be held responsible for any errors therein, nor for any damage which might result from the use or application of the document.

The physical, technical and functional characteristics of the hardware and software products and the services described in this document are subject to modification and cannot under any circumstances be regarded as contractual.

The programming examples described in this manual are intended for guidance only. They must be specially adapted before they can be used in programs with an industrial application, according to the automated system used and the safety levels required.

All rights reserved. No part of this manual may be copied or reproduced in any form or by any means whatsoever, including photographic or magnetic processes. The transcription on an electronic machine of all or part of the contents is forbidden.

This software is the property of NUM. Each memorized copy of this software sold confers upon the purchaser a non-exclusive license strictly limited to the use of the said copy. No copy or other form of duplication of this product is authorized.

Summary

1 Foreword 13

1.1 Use of the manual 13

1.2 Use of symbols in the manual 13

1.3 Document revisions 14

1.4 How to read the manual 17

1.5 Use of this Programming Manual 18

1.5.1 Function Syntax Entry Conventions 18

1.5.2 NC Operating Modes 18

1.6 NUM Agencies 19

2 Review 23

2.1 System overview 23

2.1.1 Overview of modes 23

2.1.2 Defining a program 23

2.1.3 Preparing a program 24

2.2 Machine overview 25

2.2.1 Review of axis definition and direction 25

2.2.2 Machine coordinates 26

2.2.3 Definition of travels and origins 27

2.2.4 Part settings 29

2.2.4.1 Milling machines equipped with rotary tables 31

2.2.5 Tool dimensions 33

2.2.5.1 Milling tools dimensions 33

2.2.5.2 Turning tools dimensions 34

2.2.6 Definition of dynamic tool corrections 36

2.2.6.1 Dynamic tool corrections for Milling tools 36

2.2.6.2 Dynamic tool corrections for Turning tools 37

3 Structure of a Program 41

3.1 Word format 42

3.1.1 General word format 42

3.1.2 Examples of word formats 44

3.1.2.1 Word defining a dimension, address X (internal unit in µm) 44

3.1.2.2 Word defining a function, address G 44

3.2 Block format 45

3.2.1 Examples of blocks 45

3.2.1.1 A block defining a tool change and calling up the tool correction 45

3.2.1.2 A block defining spindle rotation 46

3.2.1.3 A block defining a move 46

3.3 General structure of a program 47

3.3.1 General 47

3.3.2 Branches and Subroutine Calls 49

3.3.3 Program Numbering 50

3.3.3.1 Inhibiting display of subroutines being executed 50

3.4 Word format - Main summary table 51

3.5 Classification of preparatory and miscellaneous Functions 54

3.5.1 Classification of G Functions 54

3.5.1.1 Modal G Functions 54

3.5.1.2 Nonmodal G Functions 54
3.5.1.3 G Functions not compatible with the program status 54
3.5.1.4 G Functions associated with arguments 55
3.5.2 Classification of M Functions 56
3.5.2.1 Modal M Functions 56
3.5.2.2 Nonmodal M Functions 56
3.5.2.3 «Pre» M Functions 56
3.5.2.4 «Post» M Functions 56
3.5.2.5 Decoded M Functions 57
3.5.2.6 Encoded M Functions 57

4 Axis Programming 61

4.1 General 61
4.2 Programming the independent secondary axes 62
4.3 Programming with reference to Diameter or Radius 63
4.3.1 System programmed with Reference to Diameter 63
4.3.2 System programmed with Reference to Radius 63
4.4 Programming of carrier/carried parallel axis pairs 64
4.5 Programming rotary axes modulo 64
4.5.1 Rotary axis programmed in absolute dimensions (G90) 64
4.5.2 Rotary axis programmed in incremental dimensions (G91) 66
4.6 Programming rotary axes with limited travel 67

5 ISO Programming 75

5.1 Functions and programming addresses 75
5.1.1 G Functions - Main summary table 75
5.1.2 M Functions - Main summary table 91
5.1.3 Other Functions - Main summary table 95
5.2 Initialisation 98
5.2.1 Absolute or Incremental selection ► G90, G91 98
5.2.2 Plane selection ► G17, G18, G19 100
5.3 Spindles control 101
5.3.1 Spindle control selection ► M62 102
5.3.2 Spindle range ► M40, M41, M42, M43, M44, M45 104
5.3.3 Spindle rotation selection ► M03, M04, M05 105
5.3.4 Indexed spindle stop ► M19 107
5.3.5 Spindle release ► M61 109
5.3.6 Spindle measurement selection ► M66 110
5.4 Spindle speed 111
5.4.1 Spindle speed control ► G97 111
5.4.1.1 Review of surface speed 112
5.4.1.2 Using a fixed spindle rotation speed (N) 112
5.4.1.3 Using a constant surface speed (V) 113
5.4.1.4 Spindle speed limit 113
5.4.2 Spindle speed limitation ► G92 S.. 114
5.4.3 Constant surface speed on Turning machine ► G96 116
5.5 Axes Feed rate 117
5.5.1 Feed rate expressed in Millimetres, Inches or Degrees per Minute ►G94 117
5.5.2 Feed rate expressed in millimetres or Inches per revolution ► G95 119
5.5.3 Feed rate in inverse of time (1/min) ► G93 121
5.5.4 Case of rotary axes 125
5.5.4.1 Programming of a modulo rotary axis alone 125
5.5.4.2 Programming of modulo rotary axis and a linear axis 125
5.5.4.3 Programming of two modulo rotary axes 125
5.5.4.4 C axis feed rate for Turning machine 126
5.5.5 Feed rate specific to Fillets EB+ and Chamfers EB- ► EF.. 127
5.5.6 Tangential feed rate ► G92 R.., G92 L.. 130
5.5.7 Setting feed rate per axis set ► G92 F.. 133

5.5.8 POLAR CONSTANT FEEDRATE ► G20 G92 F… P 134
5.5.9 XY CONSTANT CONTACT SPEED ► G20 G92 F… Q 135
5.5.10 CONSTANT C SPEED ► G21 G92 F… C 136
5.5.11 Linear feed variation ► G94/G95F.. EF.. 138

5.6 Axes programming and interpolation 140
5.6.1 Rapid positioning ► G00 140
5.6.1.1 Positioning with respect to the cutter diameter (R-/R+) 142
5.6.2 Linear Interpolation ► G01 144
5.6.2.1 Positioning with respect to the cutter diameter (R+/R-) 145
5.6.2.2 Programming of additional axes and carrier/carried axis pairs 146
5.6.3 Circular interpolation ► G02, G03 148
5.6.3.1 Direction of movement according to the turret position with Turning machine 150
5.6.3.2 Programming a circle by its radius (R..) 151
5.6.3.3 Programming a circle by the coordinates of its center (I and J, plane G17) 151
5.6.3.4 Programming the additional axes and carrier/carried axis pairs 152
5.6.4 Polar programming 153
5.6.4.1 Polar programming of a line 154
5.6.4.2 Polar programming of a circle 156
5.6.4.3 Defining a circle by the arc angle 160
5.6.5 Programming fillets and chamfers 166
5.6.5.1 Fillet between two interpolations ► EB+.. 166
5.6.5.2 Chamfer between two linear interpolations ► EB-.. 167
5.6.6 Helical interpolation 169
5.6.6.1 Determination of the helix depth 170
5.6.6.2 Circular interpolation defined by three points ► G23 173
5.6.7 Cylindrical grinders specificity 175
5.6.7.1 Initial tool positioning before machining on an inclined axis ► G07 177
5.6.7.2 Movement along the inclined axis ► G05 178
5.6.8 Coordinates systems with C Axis with Turning machine 180
5.6.8.1 Programming in Polar Coordinates ► G20 180
5.6.8.2 Definition of the Start X for Interpolation on the C Axis ► G98 181
5.6.8.3 Programming in cartesian coordinates ► G21 182
5.6.8.4 Extended Features of G21 186
5.6.8.5 Programming in cylindrical coordinates ► G22 188

5.7 Program offsets 191
5.7.1 Programming of movements in absolute coordinates referenced to the measurement origin ► G52 191
5.7.2 Datum shift (DAT1 and DAT2) cancel ► G53 194
5.7.3 Selection and activation of datum shift 1 (DAT1) ► G54, G55, G56, G57, G58 195
5.7.4 Selection and activation of datum shift 2 (DAT2) ► G501.. G599 196
5.7.5 Program origin preset ► G92 197
5.7.6 Program origin offset ► G59 199
5.7.6.1 Function G59 programmed in absolute dimensions (G90) 200
5.7.6.2 Function G59 programmed in incremental dimensions (G91) 200
5.7.6.3 Features related to arguments I, J (K) 200
5.7.7 Angular offset ► ED.. 210
5.7.8 Table offset by DAT3 with Milling machine ► DAT3 214

5.8 Tool offsets 216
5.8.1 Tool axis orientation ► G16 216
5.8.2 Tool change ► M06 219
5.8.3 Tool Offset activation ► D.. 222
5.8.3.1 Case with Tool Management function 222
5.8.3.2 Tool length correction (L) 224
5.8.3.3 Tool dimensions corrections 224
5.8.3.4 Tool radius corrections 224
5.8.3.5 Tool tip orientation (C) according to the turret position 226
5.8.3.6 Use of tool corrections (X, Z, R and C) on a lathe with two mechanically coupled turrets 226

5.9 Tool correction 228
5.9.1 Tool radius compensation ► G41, G42, G40 228
5.9.1.1 Tool positioning 231

5.9.1.2 Radius correction according to the turret position 232
5.9.1.3 Tool outside the profile (line/line or circle/circle) 233
5.9.1.4 Tool outside the profile (line/circle) 235
5.9.1.5 Tool inside the profile 235
5.9.1.6 Tool inside the profile, special cases 236
5.9.1.7 Tool disengagement 237
5.9.2 Enhanced tool radius correction 247
5.9.3 3D Tool compensation 249
5.9.3.1 Concept of surface and normal vector 249
5.9.3.2 Toroidal and spherical tools 250
5.9.3.3 Double twist Machine Head 250
5.9.3.4 Geometric transformations with 3-Axis Correction 251
5.9.3.5 Geometric transformations with 5-Axis Correction 253
5.9.3.6 Tridimensional tool correction with toroidal or spherical tool ▶ G29 254
5.9.3.7 Tridimensional tool correction with cylindrical Tool and RTCP ▶ G43+/- 256

5.10 Canned cycles 257
5.10.1 Cycle overview ▶ G8x 257
5.10.2 Cancellation of a canned cycle ▶ G80 259
5.10.3 Center drilling cycle ▶ G81 260
5.10.4 Counterboring cycle ▶ G82 262
5.10.5 Peck drilling cycle ▶ G83 264
5.10.6 Tapping cycles ▶ G84 268
5.10.7 Tapping with chip breaking ▶ G84 271
5.10.8 Rigid tapping cycle ▶ G84 274
5.10.9 Rigid tap removal ▶ G39+ 278
5.10.9.1 Cycle execution 279
5.10.10 Reaming cycle ▶ G85 280
5.10.11 Boring cycle with indexed spindle Stop at the bottom of the hole ▶ G86 282
5.10.12 Drilling cycle with chip breaking ▶ G87 284
5.10.13 Boring and Facing cycle ▶ G88 288
5.10.14 Boring cycle with Dwell at the bottom of the hole ▶ G89 290
5.10.15 Thread chasing cycle ▶ G31 295
5.10.15.1 Cycle repeat to achieve the thread precision 297
5.10.16 Cycle summary 298
5.10.17 Examples of G81 to G89 cycles 299

5.11 Milling cycles 302
5.11.1 Simple pocket cycle ▶ G45 302
5.11.1.1 Pass setting possibilities 304
5.11.1.2 Roughing cycle with a single tool 305
5.11.1.3 Roughing Cycle Steps 305
5.11.1.4 Roughing and finishing cycle with a single tool 306
5.11.1.5 Finishing cycle alone with a new tool 306
5.11.1.6 Pass setting in the plane (Q) 307
5.11.1.7 Pass setting along the tool axis (P) 307
5.11.2 Spline interpolation ▶ G48, G06, G49 310
5.11.2.1 Spline curve definition ▶ G48 311
5.11.2.2 Spline curve execution ▶ G06 313
5.11.2.3 Spline curve deletion ▶ G49 318
5.11.3 Tridimensional curve smoothing ▶ G104 319
5.11.4 Corner rounding function ▶ G164 321
5.11.5 Trochoidal milling ▶ G725 323
5.11.6 Engraving ▶ G730 325
5.11.7 Tangential Control ▶ G748 327

5.12 Turning specific canned cycles 329
5.12.1 Constant pitch thread chasing cycle ▶ G33 329
5.12.1.1 Tool retraction slope at the end the pass defined by R 331
5.12.1.2 Penetration defined by EB 332
5.12.1.3 Tapered thread cutting, angle defined by EA 332
5.12.1.4 Multiple start thread cutting 333
5.12.1.5 Multi cut threading cycle 335
5.12.2 Sequenced thread cutting ▶ G38 337

5.12.3 Resume thread cutting ► G38+ 339
5.12.3.1 Inclusion of function G38+ in Cycle G33 340
5.12.3.2 Associating a Teach-in cycle with resume thread cutting cycle G33 341
5.12.4 Rough Turning/Facing cycle ► G64 342
5.12.4.1 Roughing cycle P or R 345
5.12.5 Groove roughing cycle ► G65 350
5.12.6 Plunging cycle ► G66 355
5.12.7 Roughing cycle with groove ► G63 359
5.12.7.1 Definition of EB and EC 363
5.12.7.2 Cycle differences between turning and facing 364

5.13 Jumps and branches 370
5.13.1 Restrictions related to drip feed mode 370

5.13.2 Unconditional branch to a subroutine or sequence blocks with return ► G77 371
5.13.2.1 Call to subroutine return block ► G77 -i 377
5.13.3 Branch to a sequence without return ► G79 378
5.13.4 Subroutine call by M function ► M.. 380
5.13.4.1 Subroutine call by M function 380
5.13.5 Subroutine call by PLC 381
5.13.6 Emergency retract ► G75 382
5.13.7 Block interrupt / Probing ► G10 385
5.13.7.1 Analysis of the interrupt 387
5.13.7.2 Positions in programmed frame 389
5.13.7.3 Special use of sequence interrupt 390

5.14 Standard functions 392
5.14.1 Dwell ► G04 392
5.14.2 Accurate end of block Stop ► G09 393
5.14.3 Feed increase ► G12 395
5.14.4 Mirror function ► G51 396
5.14.4.1 Influence of the mirror function on the A, B or C rotary axes 397
5.14.4.2 Influence of the mirror function on circular interpolations 397
5.14.5 Programming in Inches or Metric data ► G70, G71 399
5.14.6 Scaling factor ► G74, G73 401
5.14.7 Program Stop ► M00 405
5.14.8 Optional Stop ► M01 406
5.14.9 Coolant ► M08, M07, M09 407
5.14.10 Axis clamping and unclamping ► M10, M11 408
5.14.11 Overrides inhibit ► M49, M48 409
5.14.12 Block Skip ► / N 410

5.15 Advanced Miscellaneous functions 411
5.15.1 Automatic homing 411
5.15.2 Subroutine call on Reset %11000 412
5.15.3 Programmed feed stop for intervention ► M12 413
5.15.4 Acceleration reduction ► EG.. 414
5.15.5 Spindle acceleration reduction ► EGS 415
5.15.6 Temporary suspension of next block preparation ► G79+/- 416
5.15.7 Calling a subroutine before executing a block 418
5.15.8 Temporary inhibition of MDI access ► M999, M998 419
5.15.9 Force consecutive execution of blocks ► M997 421
5.15.10 Processing of blocks and programmed G and M functions ► G999, G998, G997 422

5.16 Dialog 424
5.16.1 Message transmission ► $0 424
5.16.1.1 Message transmission to the display 424
5.16.2 Message display with request for an operator response ► L../E..,$ 426
5.16.3 Display of messages with parametric value ► $ 428

5.17 Storage of values 429
5.17.1 Transfer of the current values of L Variables and E Parameters into the part program ► G76 429
5.17.2 Format of the numerical values 431

5.17.3 Saving symbolic variables in a dedicated memory zone ► G76+ S, G76- S 432
5.17.4 Restoring symbolic variables from dedicated memory zone ► G76+ [Sym 1] ... [Sym n] S** 432
5.17.5 ISO program or block creation/deletion ► G76+/- 433
5.17.6 Checking the existence of a file ► G76=H 434
5.17.7 Creating a program ► G76+ H 435
5.17.8 Deleting a program ► G76- H 436
5.17.9 Inserting a block ► G76+ 437
5.17.10 Deleting a block ► G76- 438
5.18 Electronic Gear box 439
5.18.1 MLEGB Spindle continuous rotation ► G770 439
5.18.2 MLEGB Synchro Definition / Cancellation ► G771 440
5.18.3 MLEGB Activation / Deactivation ► G775 441
5.18.4 MLEGB Waiting for synchronization completion before proceeding ► G776 442
5.18.5 MLEGB Synchro Homing / Gear alignment ► G779 443
5.19 Special programming for Multiple Channels 444
5.19.1 Program declaration 444
5.19.2 Programming notes 444
5.19.3 Channel synchronization with markers ► G78 447
5.19.3.1 Programming a meeting point 449
5.19.3.2 Programming by function G78 449
5.19.3.3 Programming with M functions 450
5.20 Special programming of Auxiliary channels 451
5.20.1 Program declaration and storage 451
5.20.2 Programming of the Auxiliary channels 452
5.20.2.1 Emergency retraction on an Auxiliary Channel 452
5.20.2.2 Exchanging axes between channels 453
5.21 Special features of mixed Machines (MX) 454
5.21.1 Features related to the Machine Axes 454
5.21.1.1 Boring machine 455
5.21.1.2 Mixed type machine 455
5.21.2 Assignment of tool dimensions to the Axes 456
5.21.3 Preselection of the program Origin (with function G92) 458
5.21.4 Notes on programming the ISO Functions 459

6 Profile Geometry Programming 463

6.1 Profile Geometry 463
6.1.1 Definition of the addresses characterizing PGP 464
6.1.1.1 Addresses with values ► X../Y.. , Z.., EA, X.. Y.., I.. , K.., J.., R.., EB+, EB- 464
6.1.1.2 Addresses without values ► ET, ES, E+/E- 467
6.1.2 Programming of Geometric Elements 469
6.1.2.1 Programming of fully defined Geometric Elements 469
6.1.2.2 Programming of Geometric Elements not completely defined 471
6.1.3 Programming chamfers and fillets between two elements 483

7 Parametric Programming 495

7.1 Operations and functions 495
7.1.1 Arithmetic Operations 495
7.1.2 Arithmetic Functions 496
7.1.2.1 Functions available since first versions of Flexium+ 496
7.1.2.2 Functions available since Flexium+ revision 4.1.10 497
7.1.3 Logic Operations 500
7.1.4 Assignment 500
7.1.5 Comparison Symbols Used with L Variables 501
7.1.6 Conversion in Internal Unit 501
7.1.7 Curly Brackets 501
7.1.8 Priorities 502
7.2 Programming syntax for L variables 503
7.2.1 Assignment of a Variable to an NC function 504

7.2.2 Test of a variable for a conditional branch 506
7.2.3 Notes on programming variables L100 to L199 and L900 to L959 507
7.2.4 Equivalence of variables L900 to L925 507
7.2.5 Symbolic addressing of variables L900 to L925 and L926 to L951 507
7.3 Programming syntax for E variables 516
7.4 Address equivalences 517
7.5 Errors and Warnings 518

8 External E Parameters 523
8.1 E Parameters description 523
8.1.1 External E Parameter programming syntax 525
8.1.2 Assignment of an External Parameter to an NC Function 525
8.1.3 Declaration of an External Parameter in the Program 526
8.1.4 Test on an External Parameter for a conditional branch 527
8.2 How to read an E parameters table description 528
8.2.1 Parameter format and values 529
8.3 E Parameters - List of E parameters sorted by function 530
8.4 List of detailed E parameters sorted numerically 542

9 List of Errors and Warnings 607
9.1 Errors and error extensions summary table 607
9.2 CNC Warning (HMI messages) 679
9.3 Drives warning 682

Page deliberately empty

Summary

1 Foreword 13
1.1 Use of the manual 13
1.2 Use of symbols in the manual 13
1.3 Document revisions 14
1.4 How to read the manual 17
1.5 Use of this Programming Manual 18
1.5.1 Function Syntax Entry Conventions 18
1.5.2 NC Operating Modes 18
1.6 NUM Agencies 19

Page deliberately empty

1 Foreword

1.1 Use of the manual

This manual is for use by qualified personnel only.

To reduce the risk due to an improper use of the product, do not perform any servicing other than that contained in this manual unless you are qualified to do so.

To ensure proper use of these products, please always read this manual carefully and retain for future reference. Should the unit require maintenance, please contact a NUM authorized service center only.

1.2 Use of symbols in the manual

Symbol	Meaning of the symbol
	Important notes on product use.
	You may find this symbol whenever a wrong operation may damage the product.
	You may find this symbol whenever a dedicated Milling section or chapter is used.
	You may find this symbol whenever a dedicated Turning section or chapter is used.
	Please note that whenever functions in the manual present this Milling / Turning symbol means that such sections or chapters are used upon both cases.
xxxxxxx	Linked key word. (Blue highlight words)

1.3 Document revisions

Date	Index	Reason for revision	See chapter
02 - 2015	00	First version of the manual compliance with software release 4.0.10.00.	-
11 - 2016	01	Document review.	-
06 - 2017	02	- General structure of a program - Program Numbering - Programming rotary axes modulo 360 degrees - New MLEGB functions G770, G771, G775, G776, G779 - Word format (C+034, EC+033) - Spindle programming (Summary table) - Spindle rotation direction M03, M04, M05 - Spindle speed control G97 - Spindle speed limiting G92S - Indexed spindle stop M19 - Spindle release M61 - Spindle measurement selection M66 - Case with Tool management function D.. - Tridimensional tool correction with Cylindrical Tool and RTCP G43+/- - Canned cycles (Remark) - Breaks in sequence G77 (maximum 16 nesting levels) - Subroutine call by M function - Restoring symbolic variables from dedicated memory zone G76+ [Sym 1] ... [Sym n] S** - Spindle acceleration reduction EGS- Message transmission $0 - New MLEGB (Multi Level Electronic Gear Box) G770, G771, G775, G776 and G779. - G76 modifications - "L" Local variables (L99) - Priorities (new chapter) - Programming syntax for E variables (new chapter) - Errors Warnings (new chapter) - New parameters list sorted by function - New E parameters tables presentation - New E parameters: E11001, E13008, E13009, E13010, E13011, E13012, E13013, E13014, E13015, E13016, E32014, E32015, E33000 to 33009, E33010 to 33999, E36000 to E36002, E36003 to E36999, E41103, E43000 to 43009, E4310 to 43999, E58000, (E6x006 and E6x007 modified), E79012, E79013, E7x200, E7x201, E7x202, E7x203, E7x204,Ex205, E7x206, Ex207, E7x208, E7x209, E91400 to E91431, E91432 to E91463, E94400 to 94431, E95400 to E95431, E99000 to E99031, E2000001 to E200xxyy, E2010001 to E201xxyy, E2020001 to E202xxyy, E2030001 to E203xxyy, E2040001 to E204xxyy, E2050001 to E205xxyy, E2060001 to E206xxyy, E2070001 to E207xxyy, E2080001 to E208xxyy, E2090001 to E209xxyy, E2100001 to E210xxyy, E2110001 to E211xxyy, E2120001 to E212xxyy, E2130001 to E213xxyy. - New CNC errors: Error 10, 11, 12, 13, 14, 15, 19, 35, 36, 73, 163, 424, 839 to 853. - New CNC error extensions for alarms: Error 1, 2, 5, 8, 9, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20, 24, 25, 26, 33, 34, 35, 36, 37, 39, 71, 72, 73, 79, 82, 91, 92, 93, 94, 95, 96, 97, 98, 110, 111, 113, 114, 121, 122, 123, 124, 130, 131, 133, 135, 137, 138, 140, 142, 144, 145, 146, 147, 148, 149, 150, 163, 196, 197, 198, 199, 200, 210, 245, 249, 250, 251, 253, 254, 266, 300, 400, 401, 402, 403, 404, 405, 406, 407, 410, 411, 420, 421, 423, 424, 839.	3.3 3.3.3 4.4 5.1.1 5.1.4 5.5 5.5.1 5.5.2 5.5.4 5.5.6 5.5.7 5.5.9 5.11.3 5.11.6.6 5.12 5.15 5.15.4.1 5.15.18 5.19 5.20 5.23 7.1 7.1.6 7.3 7.5 8.1 8.4 8.5

Date	Index	Reason for revision	See chapter
11 - 2018	03	- Milling machines equipped with rotary tables. - General word format. - General structure of a program. - Program Numbering. - G Functions not compatible with the program status. - Classifiation of M Functions. - «Post» M Functions. - Rotary axis programmed in absolute dimensions (G90). - New G76 H Checking existence of a file. - New G750 Stops storing axes coordinates in a file. - New G751 Start storing axes coordinates in a file. - Helical interpolation. - Accurate end of block Stop G09. - New linear feed variation G94/G95F.. EF.. . - New Enhanced tool radius correction. - Cycle with chip breaking G84. - Simple pocket cycle G45. - New Multi cut threading cycle. - Breaks in sequence G77. - Subroutine call by PLC. - Call to subroutine return block. - Checking the existence of a file G76H. - Inserting a block G76. - Datum shift (DAT1 and DAT2) cancel G53. - Corner rounding function G164. - Temporary inhibition of MDI access. - Block Skip N. - Operations and functions. - Functions available since first versions of Flexium⁺. - New Functions available since Flexium⁺ revision 4.1.10. - New Assignment. - Priorities. - Programming syntax for L variables. - Symbolic addressing of variables L900 to L925 and L926 to L951. - New Parameter format and values. - New E11024, E13017, E13018, E13019, E32016, E32017, E32018, E32019, E41104, E41105, E88000 to E88999, E95500 to E95531, E95600 to E95631, E95700 to E95731, E30000000 to E30000031, E30000032 to E30000063. - New CNC alarms: ..Error 83: M Timed and Inter-sample. ..Error 103: PGP:Chamfer or Fillet. ..Error 141: Programming error in G41/G42 with collision avoidance. ..Error 164: Axis out of limits. ..Error 203: Poor signal or complementarity error on handwheel 3. ..Error 246: Emergency Req. ..Error 500: Tilted Planed configuration: Invalid option FXSO000914. ..Error 501: Invalid T argument given the 5 axes kinematic configuration. ..Error 502: T arg.: positive index of setting table of RTCP function : T1 up to T8. ..Error 503: T arg.: index limited to 8 configuration. ..Error 504: Non authorized secondary axis. ..Error 510: RTCP configuration: Invalid option FXSO000154. ..Error 511: Syntax error in the current block G151Sn or G150Sn... ..Error 512: Angular measurement by encoder is required by RTCP control. ..Error 513: Error in the call conditions of RTCP by G77statement... ..Error 514: Programming error of RTCP or iPLANE function... ..Error 515: Programming error of tool retraction. ..Error 517: Programming Error in the cycle248 of 5-axis kinematics calibrating. ..Error 518: Error occurred while execution of kinematic measurement cycle. ..Error 519: Sphere solving error. - New CNC extensions: Error 1.: Extension 11, 18, 49, 53, 109, 110, 111, 112 ,113 , 114, 115, 116, 117, 118, 119. Error 2: Extension 118, 119, 120,121, 122, 123, 124, 125, 126. Error 6: Extension 7, 8, 9. Error 20: Extension 7, 8, 9, 10, 11, 12, 13. Error 22: Extension 13. Error 29: Extension 9, 10. Error 35: Extension 3. Error 92: Extension 16, 61. Error 93: Extension 26, 27, 28.	2.2.4.1 3.1.1 3.3.1 3.3.3 3.4.1.3 3.4.2 3.4.2.4 4.4.1 5.1.1 5.1.1 5.1.1 5.7.4 5.7.9 5.10.1.6 5.11.6 5.12.8 5.13 5.14.1.5 5.15 5.15.4 5.15.10 5.15.13 5.15.16 5.16.2 5.19.4 5.19.10 5.19.1 7.1 7.1.2.1 7.1.2. 7.1.4 7.1.7 7.2 7.2.6 8.2.1 8.3 9.1

		Error 94: Extension 18, 23, 24, 39, 40, 41, 42 , 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53. Error 95: Extension 63. Error 101: Extension 3, 4. Error 150: Extension 31. Error 191: Extension 32, 33, 34, 35, 36, 37. Error 193: Extension 19, 20, 21, 22, 23, 24, 25, 26, 27. Error 196: Extension 22, 23, 24, 26, 27. Error 197: Extension 20, 21. Error 210: Extension 5, 6, 7. Error 300: Extension 4. - New CNC Warning (HMI messages)	9.1 9.2
2 - 2020	04	- Encoded M Functions. - Rotary axis programmed in absolute dimensions (G90). - New G30, G32, G34. - New informations on G90, G91. - New informations on G94. - New informations on G92F. - New informations on G92F "Pass setting along the tool axis (P)". - New Trochoidal milling G725. - New Engraving G730. - New informations on "Subroutine call on Reset %11000". - New informations on G76=H. - New informations on G76+. - New informations on G76+ [Sym 1]...[Sym n] S**. - New informations on G52. - New informations on EGS. - New informations on "Functions available since Flexium+ revision 4.1.10". - New informations on "Priorities". - New or modified E parameters: E11011, E11025, E11026, E11027, E32018, E32019, E32050, E32051, E32052, E32053, E32054, E35300 to E35363, E41010, E6x008, E6x009, E6x012, E79020 to E79022, E79023 to E79025, E79026 to E79028, E7x008, E90500 to E90531, E90964 to E90995, E91164 to E91195, E91500 to E91531, E91600 to E91631, E93232 to E93263, E96400 to E96431, E98200 to E98231, E98300 to E98331, 98600 to E98631, E98632 to E98663. - New Errors or Errors extensions ": Error 1.11, 1.62, 1.75 , 1.120÷135, Error 2.29, 2.133÷139, Error 4.1, 4.2, Error 16.25, Error 17.2, Error 21.1÷8, Error 29.11, 29.12, Error 35.4, Error 36.5, Error 72.2÷12, Error 91.56, 57, 78, 79, Error 92.62÷67, Error 93.7, Error 94.34, Error 94.54÷62, Error 95.50, 64÷69, Error 121.3÷5, Error 143.1÷18, Error 144, Error 145, Error 191.38÷44, Error 193.28÷39, Error 197.13, Error 252.3÷6, Error 267, Error 518.5, Error 530.1÷59, Error 700.1÷3, Error 701.1÷34, Error 702.1÷37, Error 703.1÷10, Error 704.1÷8, Error 705.1÷22, Error 706.1÷4, Error 702.1÷5, Error 751, Error 752, Error 753, Error 754, Error 755, Error 760.1÷5, Error 761, Error 762, Error 763, Error 764. New Warnings: 61, 84, 85, 86, 87, 88, 89, 90 , 91, 92, 165, 166, 2000, 2001, 2002, 2003, 2004.	3.4.2.6 4.4.1 5.1.1 5.3 5.10.1 5.10.5 5.13.6.1 5.14 5.15 5.17.9 5.17.13 5.17.16 5.17.19 5.18.1 5.21.15 7.1.2.2 7.1.7 8.4 9.1 9.2
7 - 2021	05	Chapter 5 reorganization and updates	5
10 - 2022	07	General word format - info integration G92 F.. - info integration Tangential feed rate ► G92 R.., G92 L.. Polar constant feedrate in ► G20 G92 F… P Extended Features of G21 Update list E Parameters	3.1.1 5.1.1 5.5.6 5.5.8 5.6.8.4 8.3



The possibility of using several DAT1 (G54 to G58) and several DAT2 has introduced a change in functionality of the DAT management.
Please refer to chapter 5.18.3 for important information.

1.4 How to read the manual

The machine Flexium+ Programming Manual gives information on how to set the parameters to customize the Flexium+ for a particular Milling and Turning machine-tool.

Chapter	Description
1	**Foreword** This chapter introduces: - The use of the Programming manual. - Function Syntax Entry Conventions.
2	**Review** This chapter introduces: - General description of the NC and its use with the machine tool. - Review of the rules and standards related to the NC/machine-tool combination.
3	**Structure of a Program** This chapter introduces: - Rules for writing a part program by assembling characters into words. - Words into blocks and blocks into a complete program.
4	**Axis Programming** This chapter introduces: - Description of the features related to axes programming.
5	**ISO Programming** This chapter introduces: - Detailed description of functions related to ISO programming for Milling and Turning machine.
6	**Profile Geometry Programming** This chapter introduces: - Detailed description of profile geometry programming (PGP), - Using PGP used to define contours as a sequence of geometric elements, with computation of intermediate points. PGP is an extension of ISO programming.
7	**Parametric Programming** This chapter introduces: - Gives the possibility of assigning variables to NC functions. The values of the variables can be obtained by computation or by reading machine data.
8	**External E parameters** This chapter introduces the detailed information upon description and use of the E parameters.
9	**List of Errors and Warnings** This chapter introduces the NC error and error extension list and warnings.

1.5 Use of this Programming Manual

1.5.1 Function Syntax Entry Conventions

The lines (blocks) of a part program include several functions and arguments.

Special syntax rules apply to each of the functions described herein. These syntax rules specify how the program blocks must be written.

Certain syntax formats are given as a line. The following conventions simplify writing the line:

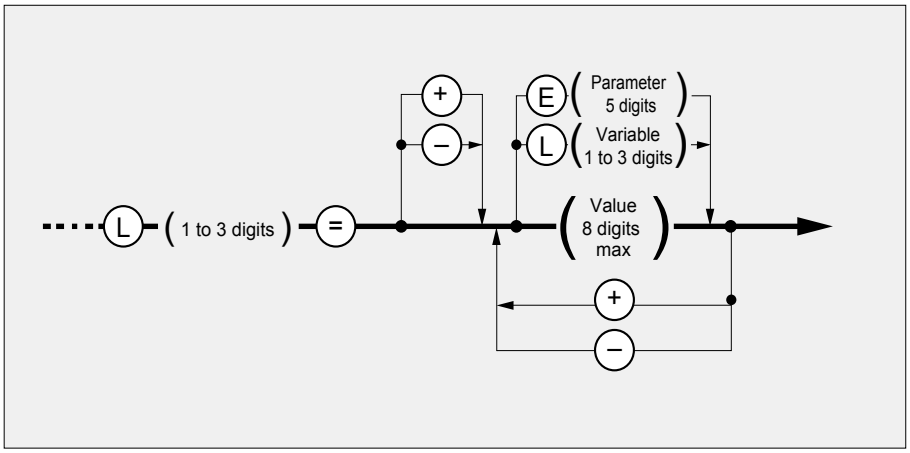
- the function to which the syntax format is related is highlighted by boldface type
- terms between square brackets « **[..]** » are optional functions or arguments in the block (except for symbolic programming where square brackets are part of the syntax)
- « **/** » indicates a choice between several terms (equivalent to « **or** ») (except in parametric programming where " **/** " means divide)
- « **..** » after a letter replaces a numerical value
- « **...** » replaces a character string (for instance a message).

Examples

Syntax of function G12

N.. [G01/G02/G03] **G12** X.. Y.. Z.. [F..] [$0…]

Syntax in the form of a Conway diagram



1.5.2 NC Operating Modes

Certain NC operating modes are mentioned herein when they are directly related to the use of ISO functions. Additional information can be found in the Flexium⁺ HMI help.

1.6 NUM Agencies

The list of NUM agencies is given click here http://www.num.com

Page deliberately empty

Summary

2 Review 23
2.1 System overview 23
2.1.1 Overview of modes 23
2.1.2 Defining a program 23
2.1.3 Preparing a program 24
2.2 Machine overview 25
2.2.1 Review of axis definition and direction 25
2.2.2 Machine coordinates 26
2.2.3 Definition of travels and origins 27
2.2.4 Part settings 29
2.2.4.1 Milling machines equipped with rotary tables 31
2.2.5 Tool dimensions 33
2.2.5.1 Milling tools dimensions 33
2.2.5.2 Turning tools dimensions 34
2.2.6 Definition of dynamic tool corrections 36
2.2.6.1 Dynamic tool corrections for Milling tools 36
2.2.6.2 Dynamic tool corrections for Turning tools 37

2 Review

The aim of this chapter is to introduce concepts that will be detailed in the rest of the manual, rather than to reflect the way an operator works on the machine.

For instance, in the chapter 2.2.4 (Part settings), the aim is to define the offsets and corresponding origins or zero points rather than give a method for measuring the offsets.

This manual covers Turning, Milling and Special machines instructions. When a function is available only on a certain type of machine, it will be indicated precisely.

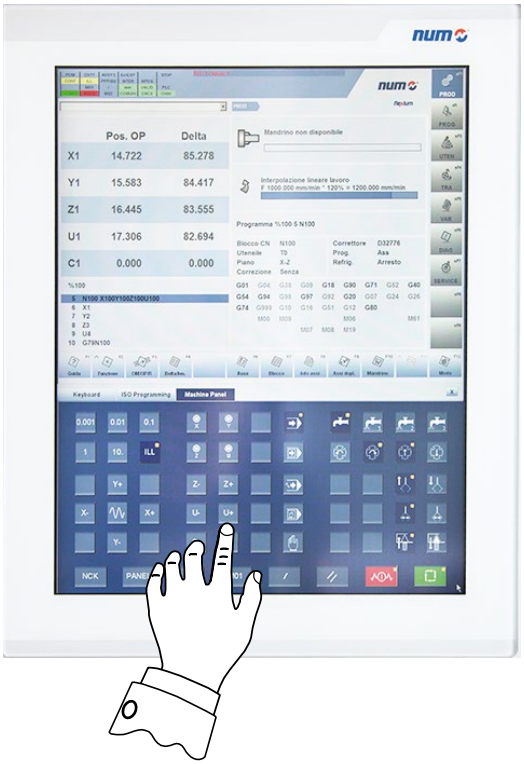
2.1 System overview

2.1.1 Overview of modes

The operator uses the numerical control (NC) in various operating modes accessible from the operator panel.

Each mode corresponds to a particular use of the numerical control (continuous machining, program loading, tool setting, etc.).

For additional information about Modes, please refer to the Flexium+ HMI on line help.



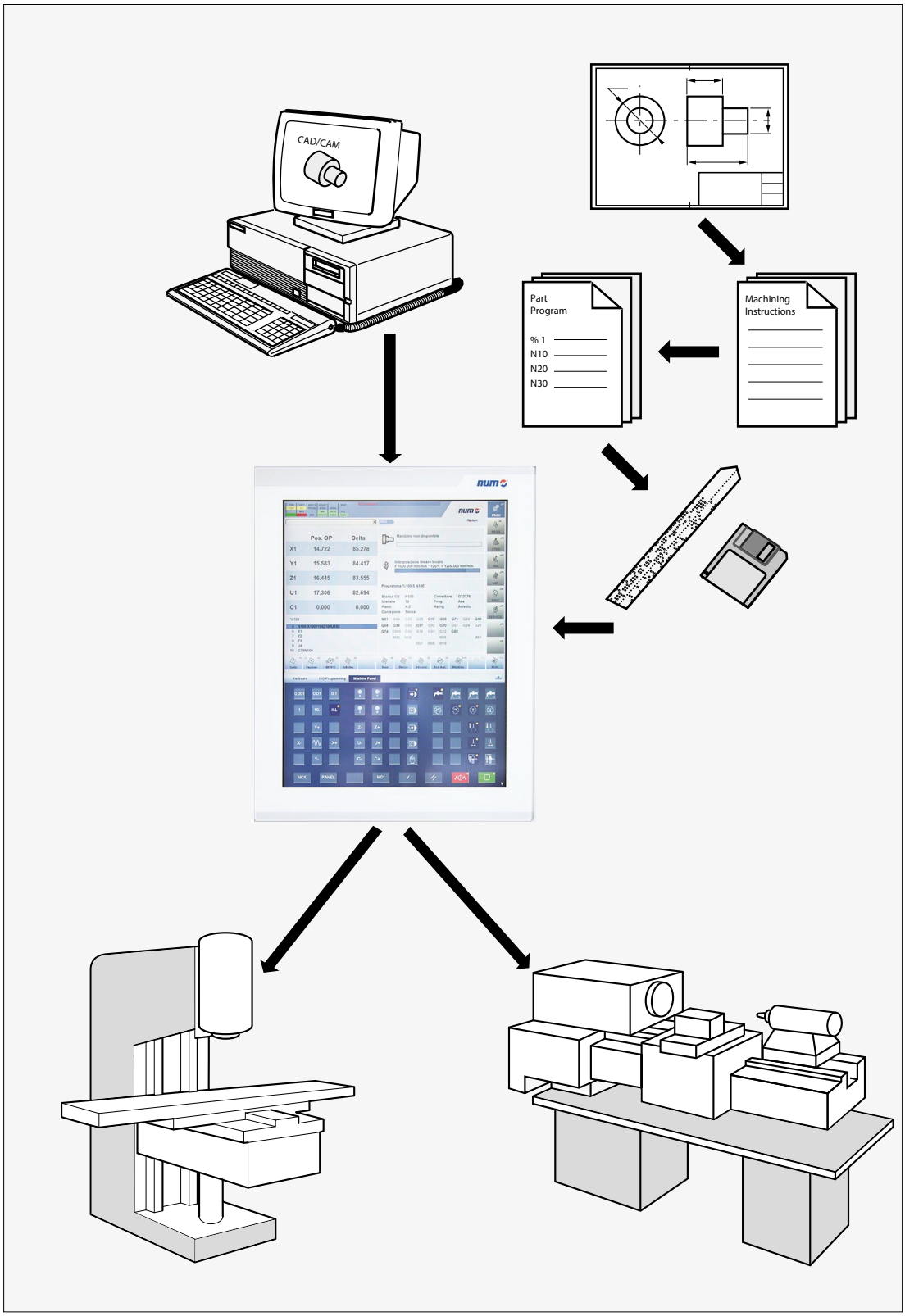
2.1.2 Defining a program

A program is a sequence of instructions written in a programming language specific to the numerical control (the most widely used is ISO code: International Standards Organization).

The numerical control interprets the program to control actions on the machine.

2.1.3 Preparing a program

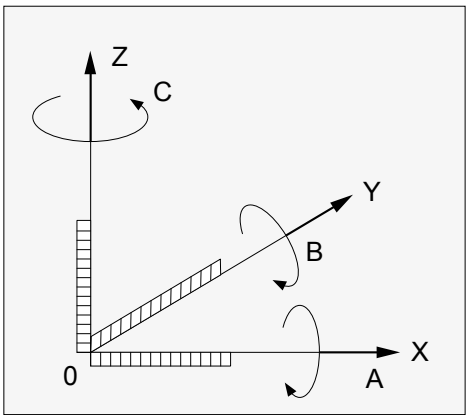
A part program can be created directly on the machine, on a remote editor or using a CAD/CAM system.



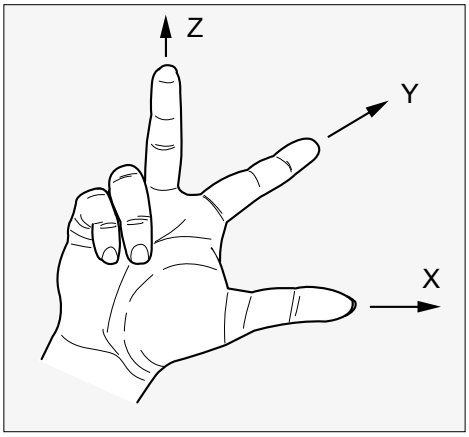
2.2 Machine overview

2.2.1 Review of axis definition and direction

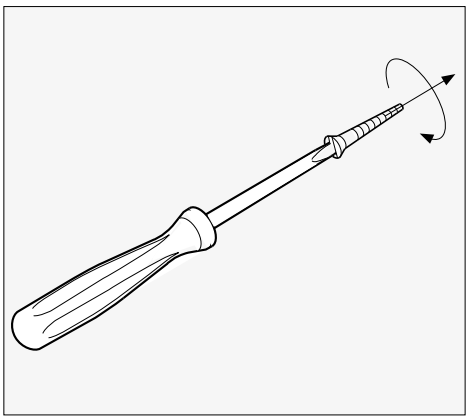
A coordinate system is used to identify the positions and movements of an object with respect to an origin or zero point.
A rectangular Cartesian coordinate system is a direct three-axis system of three linear axes, X, Y and Z, with which are associated three rotary axes, A, B and C.



The direction of axes X, Y and Z is easily remembered by the right-hand rule.



The positive direction of rotation of a rotary axis corresponds to the direction of screwing of a right-hand screw on the associated axis.



2.2.2 Machine coordinates

The manufacturer defines the coordinate system associated with the machine in accordance with standard ISO 841 (or NF Z68-020).

The coordinate system measures tool movements with respect to the part to be machined, assumed fixed.

When it is the part that moves, it may be more convenient to identify its movements. In this case, axes X', Y' and Z', pointing in opposite directions from axes X, Y and Z, are used.

The direction of the axes of a machine depends on the type of machine and the layout of its components.

For a Milling Machine:
- the Z axis is the axis of the main spindle when this axis is parallel to one of the slide ways,
- positive movement along the Z axis increases the distance between the part and tool,
- the X axis is perpendicular to the Z axis and corresponds to the largest excursion,
- the Y axis is perpendicular to the X and Z axes.

Rotary axes A, B and C define rotations around axes parallel to X, Y and Z.

Secondary linear axes U, V and W may or may not be parallel to primary axes X, Y and Z.

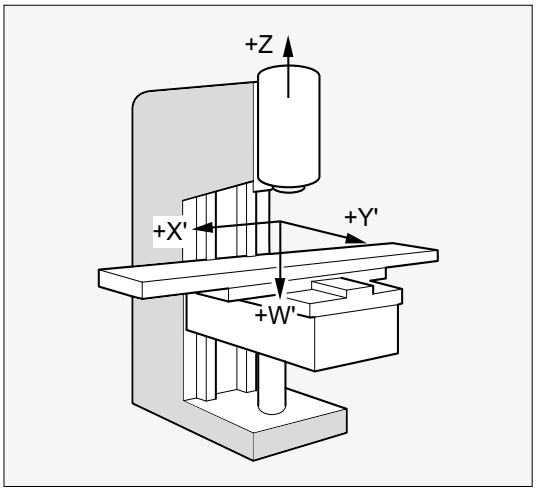
For a Turning Machine:
- the Z axis is the same as the spindle axis,
- the X axis is perpendicular to the Z axis and corresponds to radial movement of the tool-holder turret,
- the Y axis (generally a dummy axis) forms a right-handed coordinate system with the X and axes.

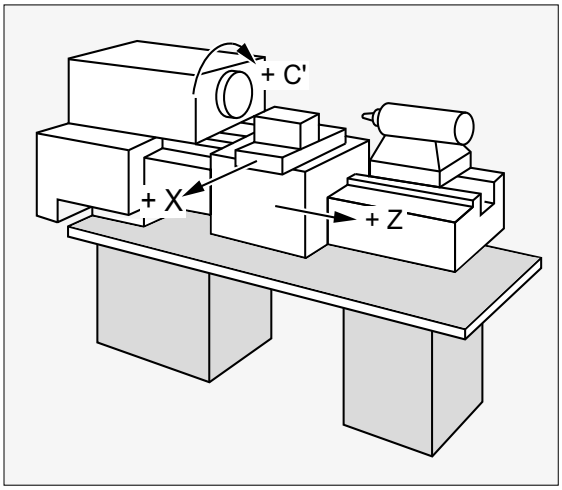
Positive movement along the Z or X axis increases the distance between the part and the tool.

Rotary axes A, B and C define rotations around axes parallel to X, Y and Z.

Secondary linear axes U, V and W may or may not be parallel to primary axes X, Y and Z.

For more details, refer to the above-mentioned standard.





2.2.3 Definition of travels and origins

All programmed movements are eventually related to the Machine Origin ('OM' in the rest of this document). The machine Origin is defined in relation with the Measure Origin ('Om' in the rest of this document). More and more machines are equipped with absolute measurement devices allowing the Measure Origin (Om) to be automatically determined at power up.

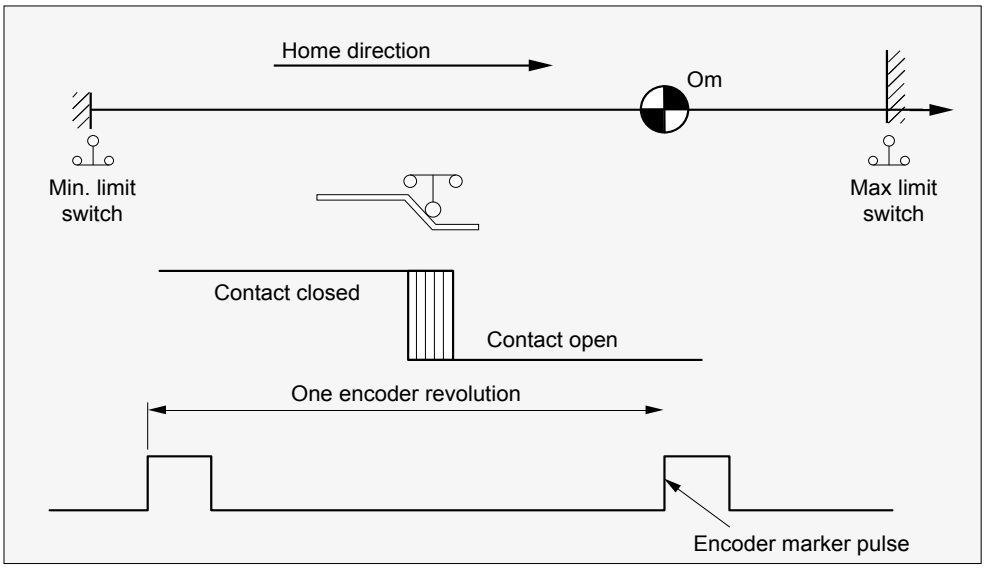
For the machines not equipped with such devices a Homing procedure is mandatory in order for the Control System to know the axes position. Once this is completed the axes travel will be automatically limited to the values defined by the machine manufacturer.

Symbols

Om : 	Measure Origin: reference of the measurement system. It is evaluated either at the completion of the homing procedure or automatically at power up for absolute measurements systems.
OM : 	Machine Origin: reference for all programmed movements.

The homing procedure is complete for each of the axes when:

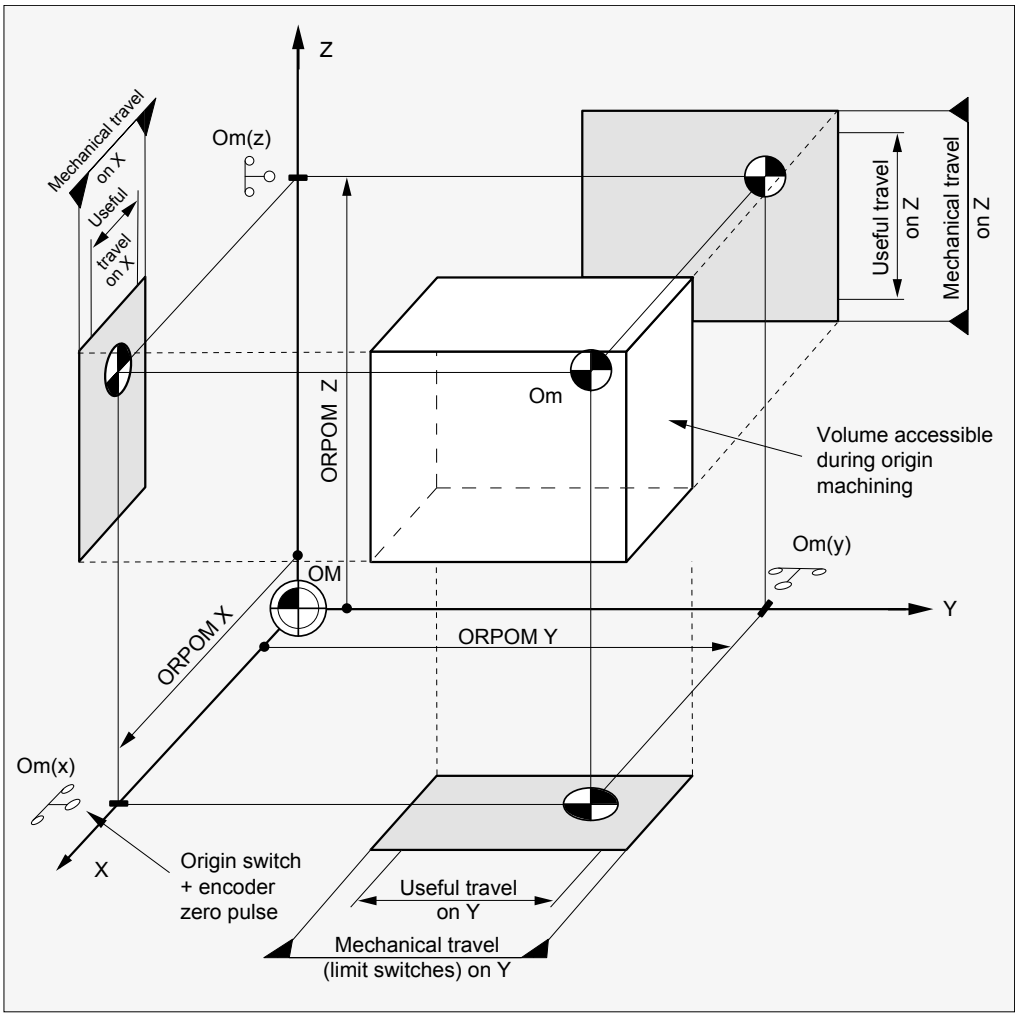
- Automatically if the machine is equipped with an absolute measurement system.
- After a short displacement (~40mm) in Home mode for the machines equipped with semi absolute measurement systems (Coded reference scales).
- After reaching the first marker pulse following the activation of the home switch for machine equipped with non absolute measurement systems.

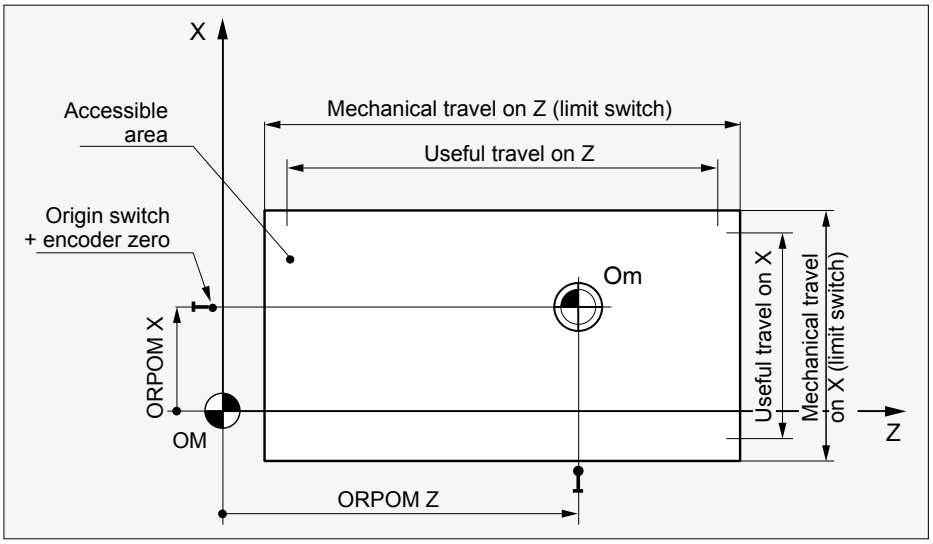


When homing is completed, the system applies the offset defined by the manufacturer to the relevant axis in order to establish the machine origin (OM).
This parameter is often named ORPOM.

Machine origin offset (Om/OM) = ORPOM

The useful travel on each axis is limited by software limits whose values are defined by the manufacturer.





2.2.4 Part settings

The part program is established in reference to a Program Origin (OP).

However, the part is mounted on the machine in reference to a Part Origin (Op).

OP :	Program Origine : Defined from Op for each axis via parameter DAT2:
Op :	Part Origine : Defined from OM for each axis via DAT1.

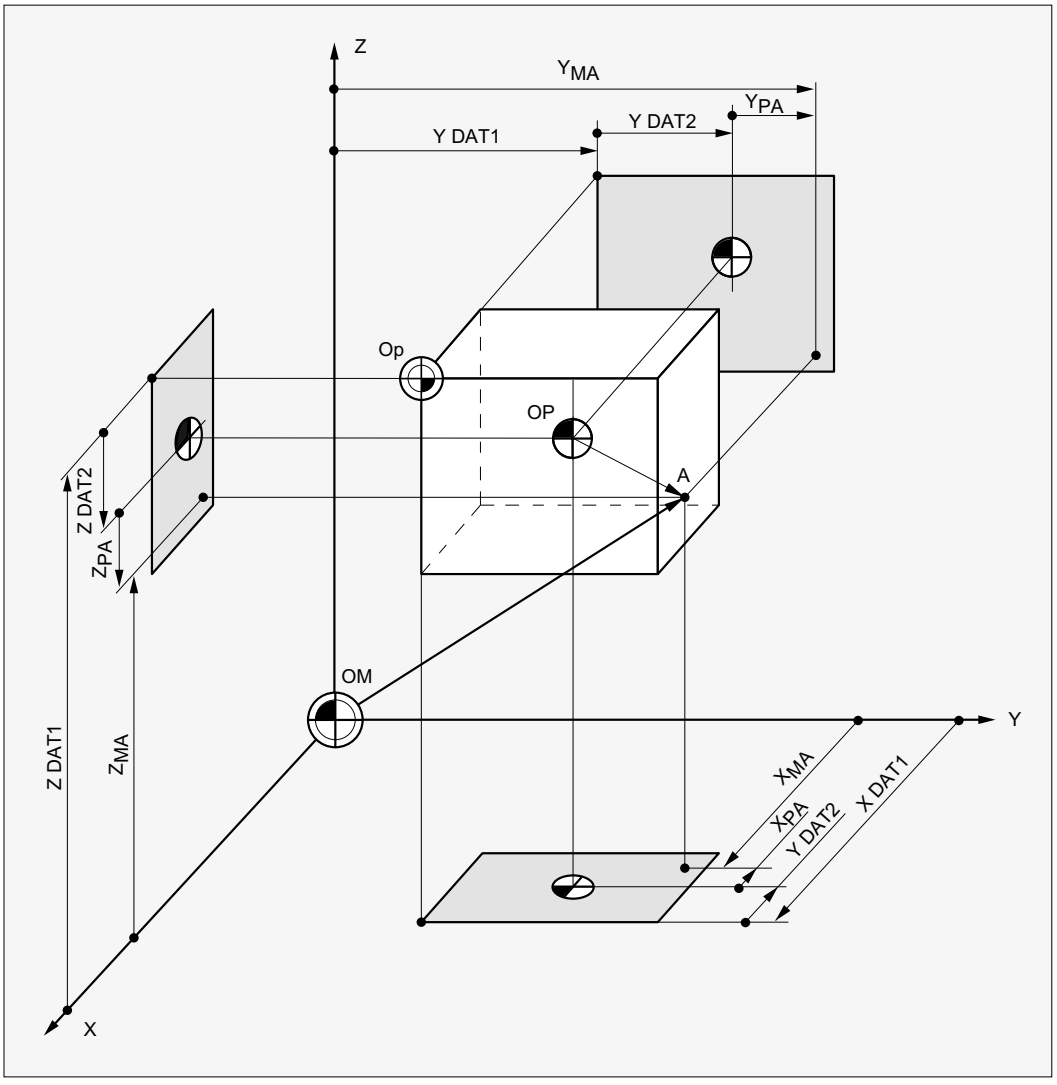
Program origin offset (OP/Op) = DAT2

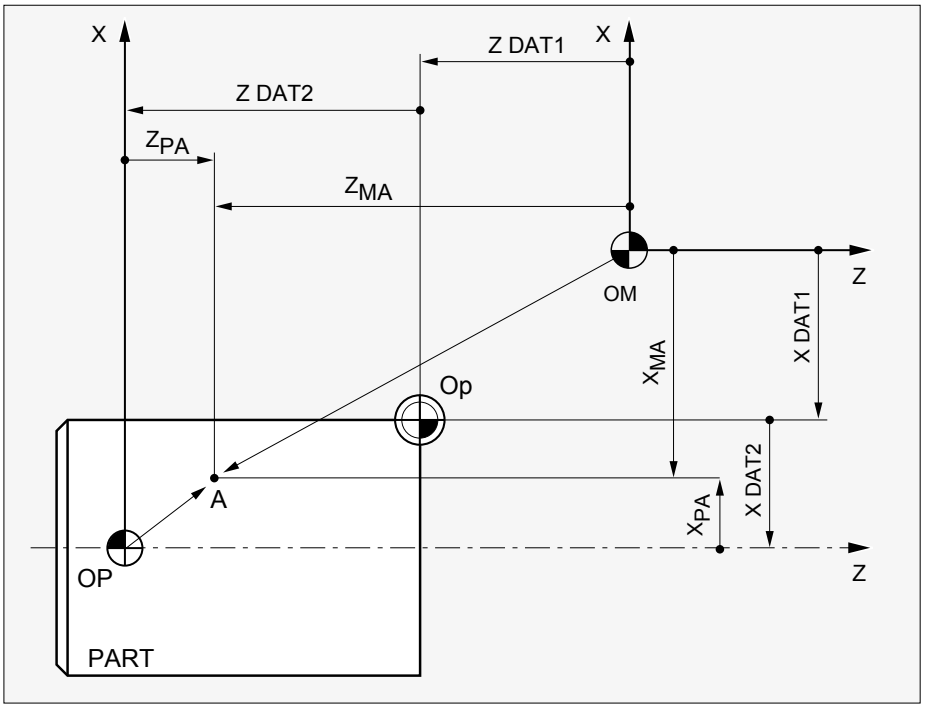


Part origin offset (Op/OM) = DAT1

Flexium⁺ offers 5 sets of DAT1 and up to 99 sets of DAT2 per channel. Both can be selected by G codes or written by parametric programming.

The coordinates of a point (A) defined with respect to the program origin (OP) are converted by the NC to coordinates with respect to the machine origin (OM):





Program dimensions (with respect to OP)	Measurement dimensions (with respect to OM)
X_{PA}	$X_{MA} = X_{PA} + X\ DAT1 + X\ DAT2$
Y_{PA}	$Y_{MA} = Y_{PA} + Y\ DAT1 + Y\ DAT2$
Z_{PA}	$Z_{MA} = Z_{PA} + Z\ DAT1 + Z\ DAT2$

Programmed offsets (G59) can be added to the program dimensions.

2.2.4.1 Milling machines equipped with rotary tables



When a part is mounted on a rotary table, rotating this table will result in a modification of the part origin and therefore the program origin if the part origin is not perfectly aligned with the axis of rotation.

This modification is made along the two axes perpendicular to the axis of rotation

An additional shift (DAT3) is created to take this modification into account.

Considering OC as the center of rotation we have the relations:

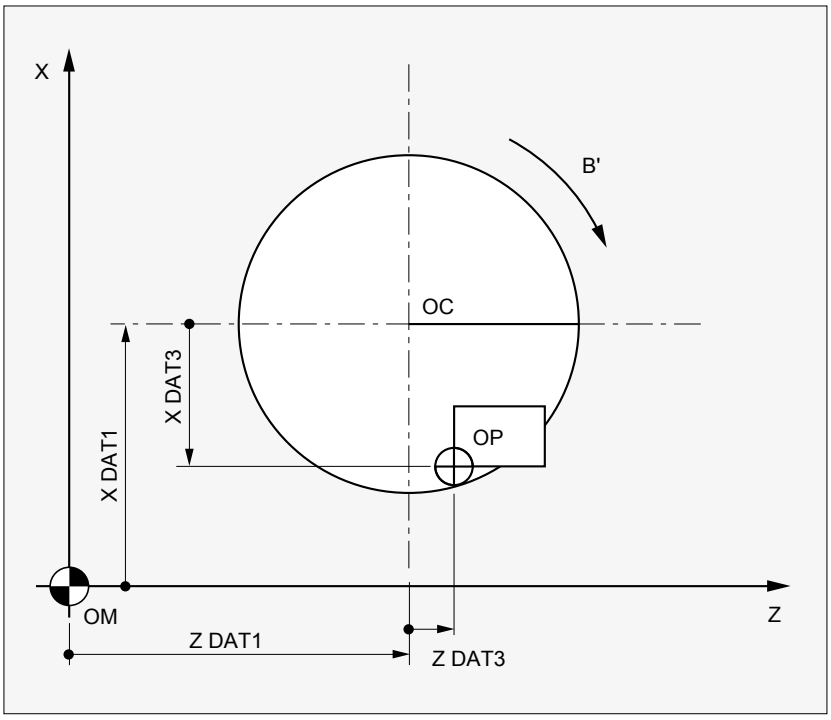
Offset to the center of rotation (OC/OM) = DAT1

Offset to the part origin (OP/OC) = DAT3

For axes other than those affected by rotation, only DAT1 and DAT2 apply.

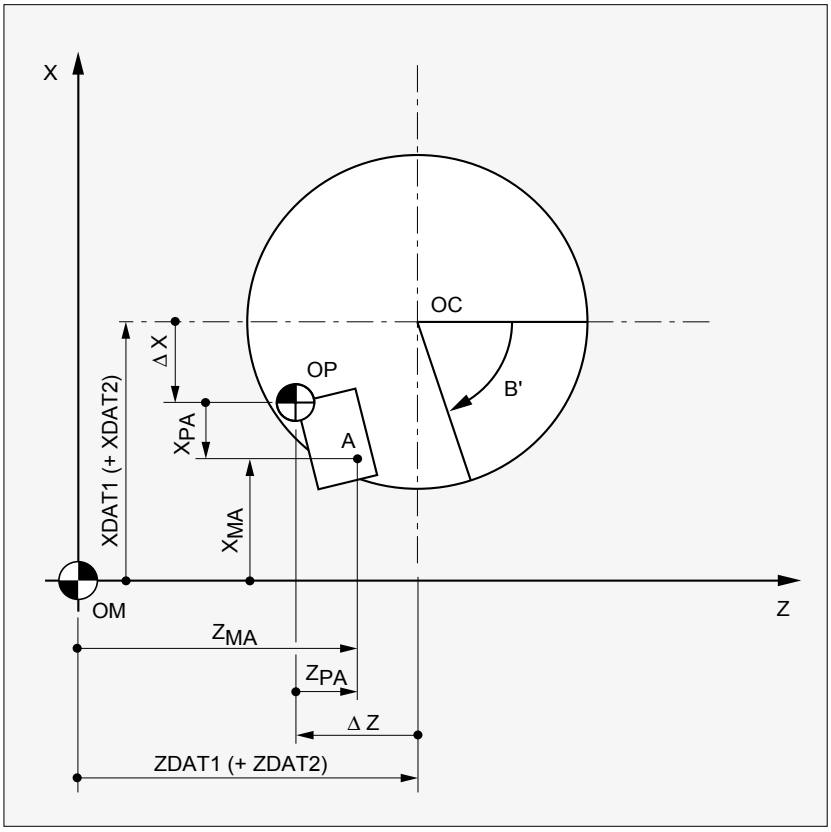
Example: rotary axis B

Rotation is around an axis parallel to the Y axis. Therefore, the axes affected by the rotation are Z and X. For the B axis at Zero/OP DAT3 will be defined as follows:



After a rotation, The NC will apply the corrections ΔX and Δ Z so that the position A is correct.

Remark: The correction will only be taken into account when X and/or Z are programmed. There is no continuous compensation during the table rotation:



Program dimensions (with respect to OP)	Measurement dimensions (with respect to OM)
X_{PA}	$X_{MA} = X_{PA}$ + XDAT1 (+ XDAT2) + ΔX with ΔX = XDAT3 x cos B - ZDAT3 x sin B
Y_{PA}	$Y_{MA} = Y_{PA}$ + YDAT1 + YDAT2
Z_{PA}	$Z_{MA} = Z_{PA}$ + ZDAT1 (+ ZDAT2) + ΔZ with ΔZ = ZDAT3 x cos B + XDAT3 x sin B

2.2.5 Tool dimensions

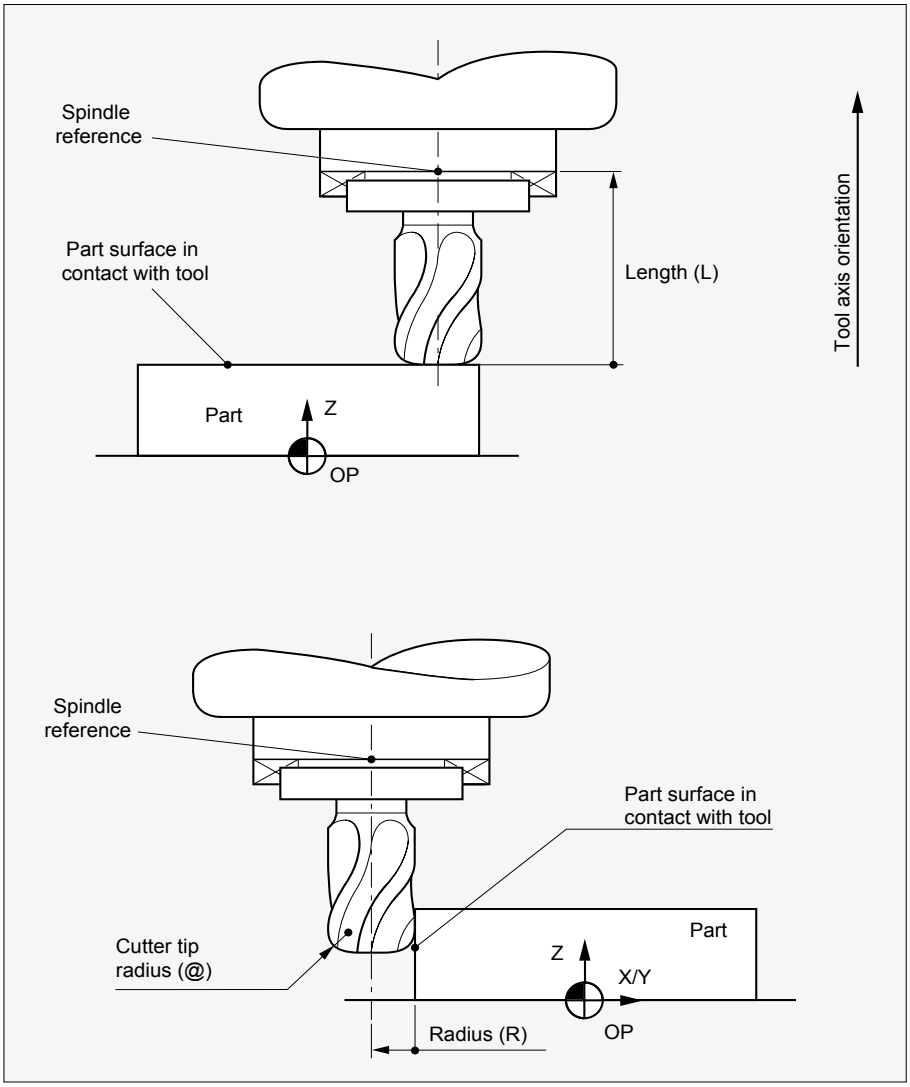
Flexium CNC considers two types of tools :

- Milling tools
- Turning tools

According to the tool type, the tool offset data are different.

2.2.5.1 Milling tools dimensions

Tool dimension = distance from tool cutting edge to spindle reference point

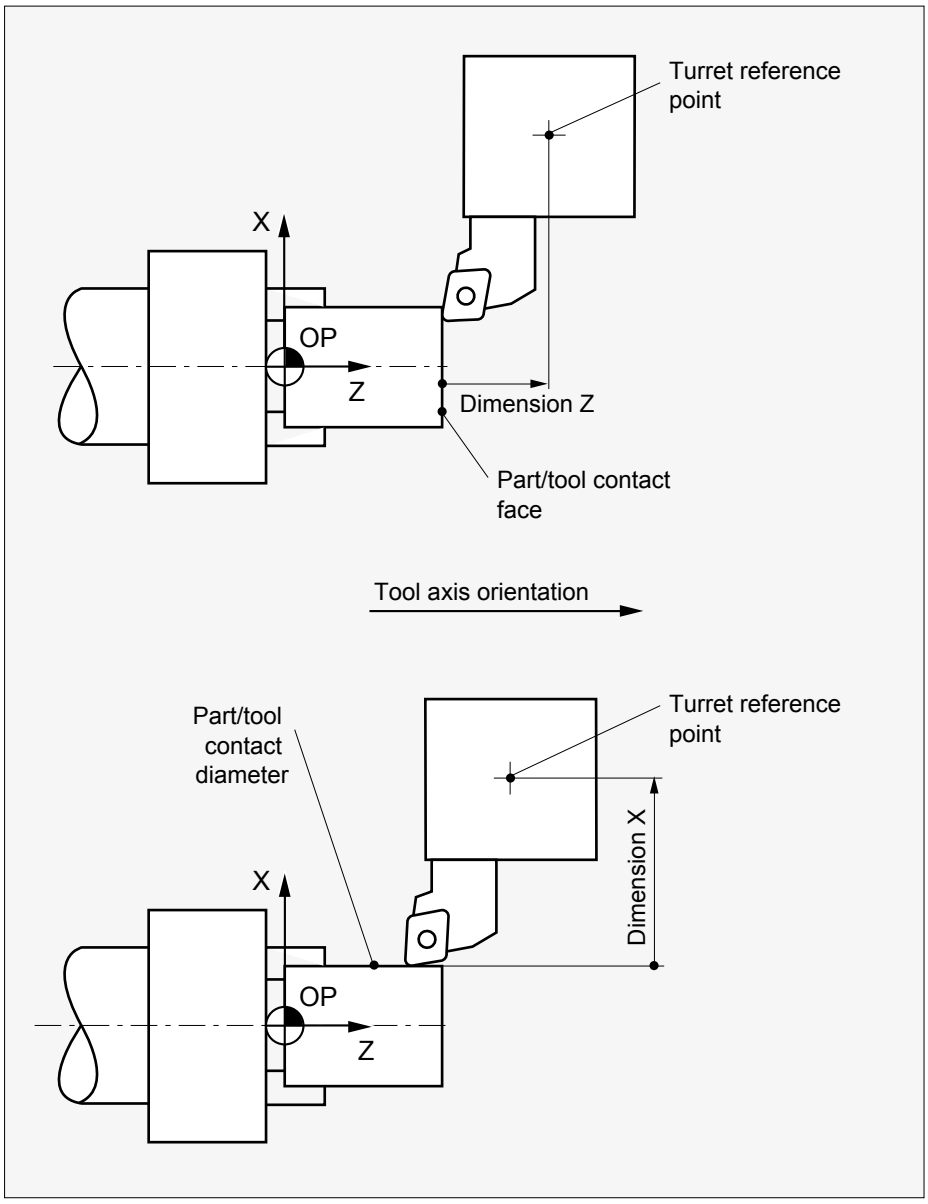


Tool radius = R
Tool length = L

Cutter tip radius = @

2.2.5.2 Turning tools dimensions

Tool dimension = distance from tool cutting edge to turret reference point



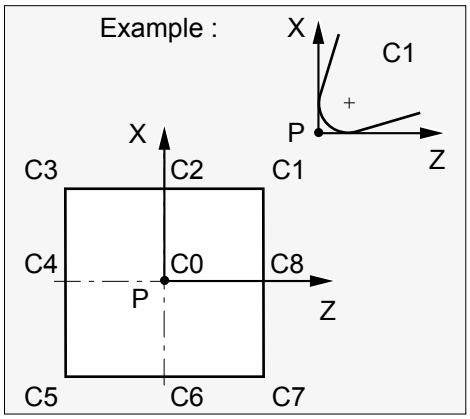
Tool X dimension = X
Tool Z dimension = Z

Definition of tool tip radius and orientation

The description of a tool is shown below:

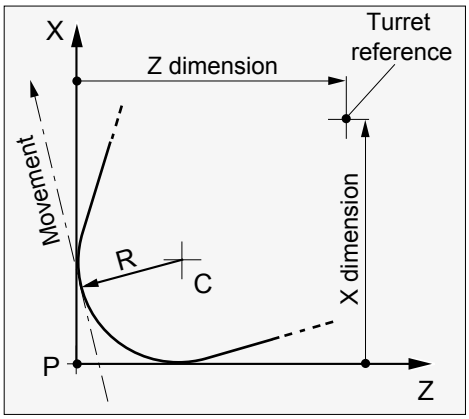
Tool tip orientation = codes C0-C8

The tool tip orientation code allows the system to locate the center (C) of the tool tip from the theoretical cutting point (P).



Tool tip radius = R

The real cutting point of the tool is obtained by applying a vector of length R perpendicular to the direction of movement from C.



2.2.6 Definition of dynamic tool corrections

At any time (even during machining), the operator can enter dynamic tool corrections when he observes a difference between the expected and the actual dimensions on a part.

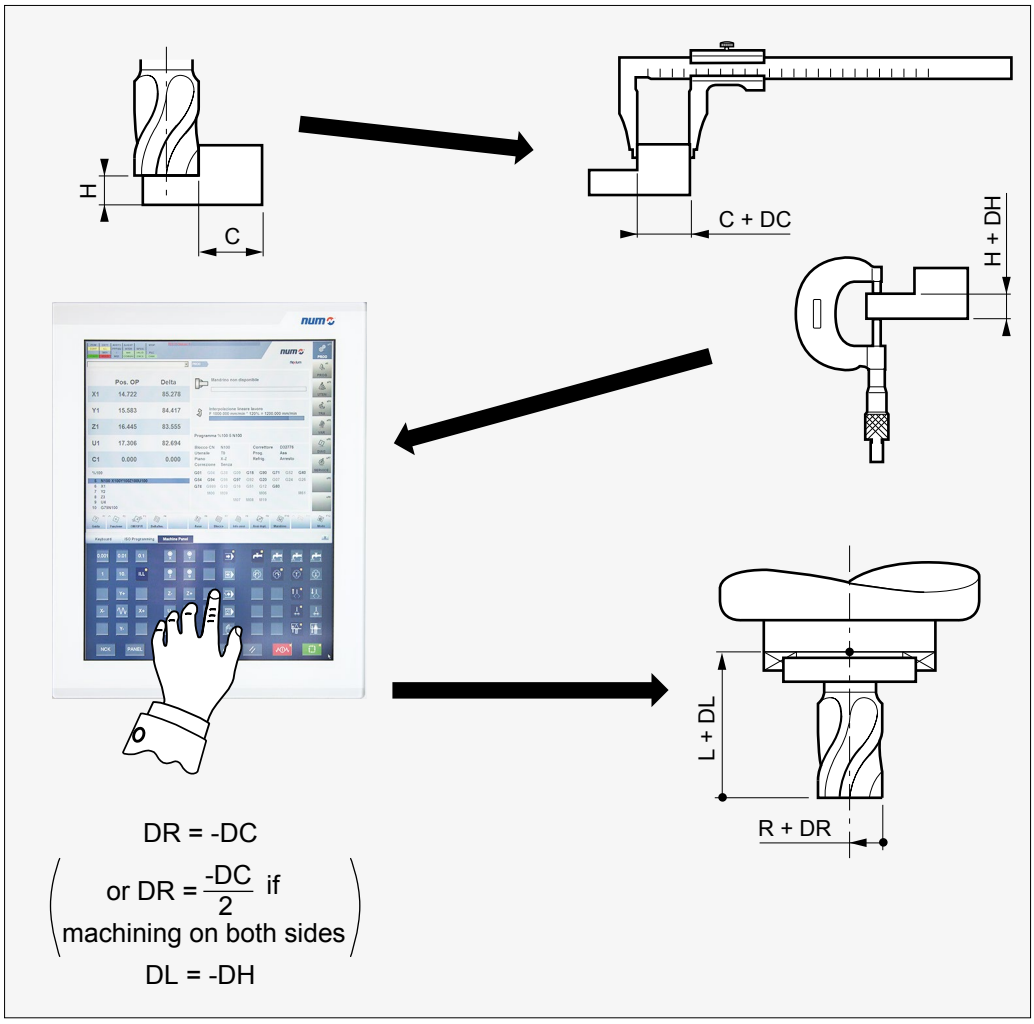
The corrections (positive or negative) compensate for slight variations of the tool or part (wear, expansion). The dynamic tool corrections are taken into account on length and radius for Milling Machines and on X dimension and Z dimension for Turning Machines.

Dynamic tool corrections are taken into account immediately for a change smaller than 0.1mm. For larger amount they will be taken into account at the first block not yet pre-processed.

2.2.6.1 Dynamic tool corrections for Milling tools



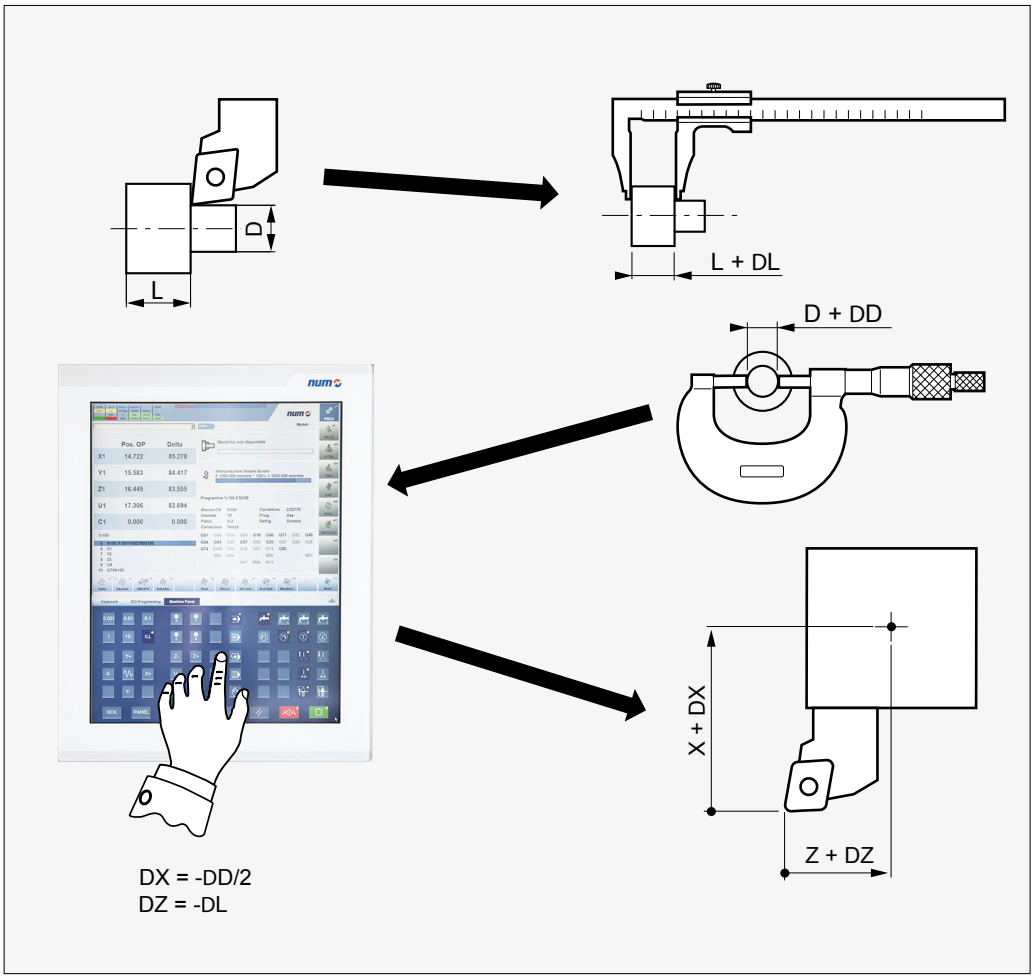
Dynamic tool radius correction = DR
Dynamic tool length correction = DL



Corrected radius = R + DR
Corrected length = L + DL

2.2.6.2 Dynamic tool corrections for Turning tools

Dynamic tool correction on X = DX (diameter)
Dynamic tool correction on Z = DZ



The operator just needs to enter the dimension variations like they are measured (diameter). The system will automatically divide them by 2 to take the right amount at the diameter.

Corrected length on X = X dimension + DX/2
Corrected length on Z = Z dimension + DZ

Page deliberately empty

Summary

3 Structure of a Program 41

3.1 Word format 42
3.1.1 General word format 42
3.1.2 Examples of word formats 44
3.1.2.1 Word defining a dimension, address X (internal unit in µm) 44
3.1.2.2 Word defining a function, address G 44

3.2 Block format 45
3.2.1 Examples of blocks 45
3.2.1.1 A block defining a tool change and calling up the tool correction 45
3.2.1.2 A block defining spindle rotation 46
3.2.1.3 A block defining a move 46

3.3 General structure of a program 47
3.3.1 General 47
3.3.2 Branches and Subroutine Calls 49
3.3.3 Program Numbering 50
3.3.3.1 Inhibiting display of subroutines being executed 50

3.4 Word format - Main summary table 51

3.5 Classification of preparatory and miscellaneous Functions 54
3.5.1 Classification of G Functions 54
3.5.1.1 Modal G Functions 54
3.5.1.2 Nonmodal G Functions 54
3.5.1.3 G Functions not compatible with the program status 54
3.5.1.4 G Functions associated with arguments 55
3.5.2 Classification of M Functions 56
3.5.2.1 Modal M Functions 56
3.5.2.2 Nonmodal M Functions 56
3.5.2.3 «Pre» M Functions 56
3.5.2.4 «Post» M Functions 56
3.5.2.5 Decoded M Functions 57
3.5.2.6 Encoded M Functions 57

Page deliberately empty

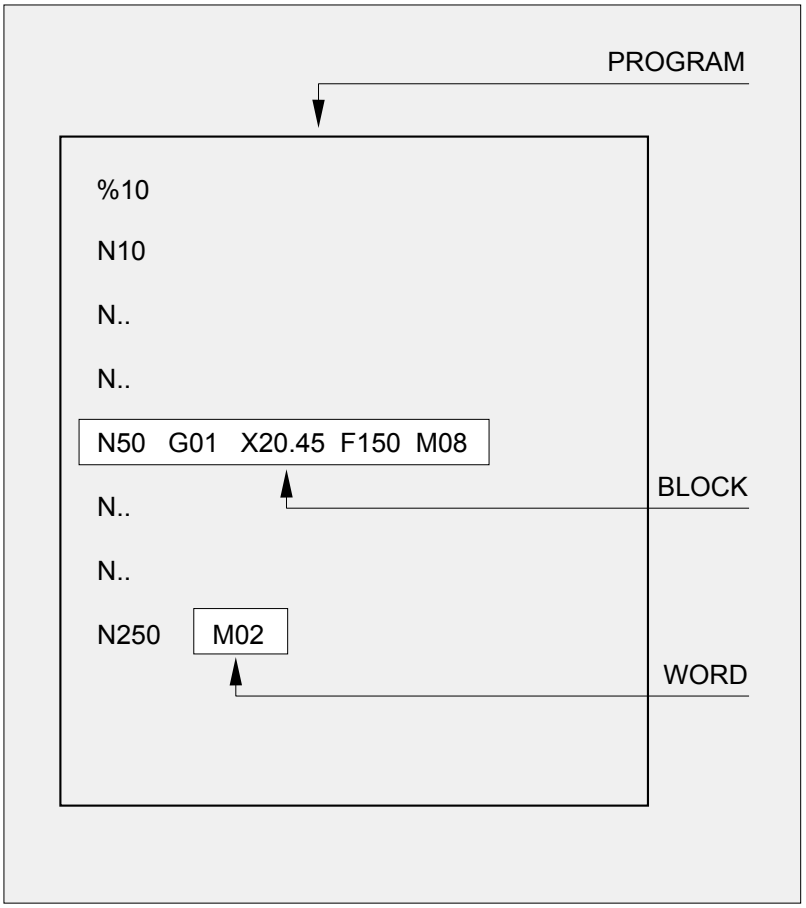
3 Structure of a Program

A CNC part program is a list of instructions and data to be transmitted to the control system.

The creation of a program consisting of blocks and words must obey structure, syntax and format rules.

This structure, syntax and format rules respect the ISO codes and standards:

- ISO (International Standards Organization) 6983-1 (NF Z 68-035), 6983-2 (NF Z 68 036) and 6983-3 (NF Z 68-037).



3.1 Word format

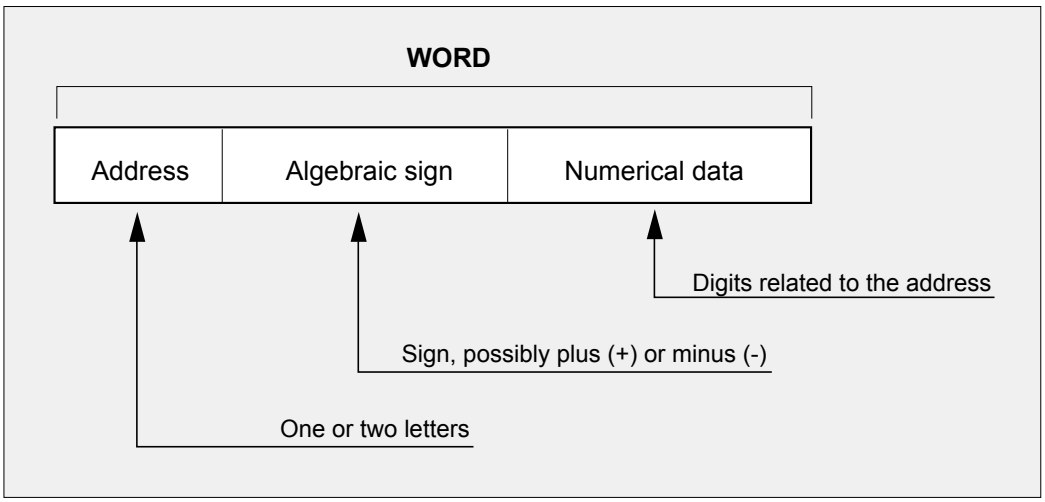
A word contains an instruction or data to be transmitted to the control system.

Word types:

- words defining dimensions
- words defining functions.

The word format defines the specific characteristics of each code word used in programming.

3.1.1 General word format



The number of characters and spaces in a block must not exceed 118.

The numerical data can be expressed in different formats:

- Numerical value
- E parameter (generally in internal units µm)
- L parameter (generally in programming units (mm or inches)
- Symbolic variable [varname]
- Result of a calculation (computed from left to right except if prioritised within {...}.

By default the numerical data is assumed to be 0.

For example it is possible to write

```
GXYZ  \\
```
meaning `G0 X0 Y0 Z0`

The absence of numerical value is accepted but should be avoided as it can lead to parser errors as in the example below

```
G1 X  L1=12  \\
```
Will be interpreted as `G1 XL1`

Move X to position L1

The term =12 generates a parser error.

From version 4.2.0 it is possible to inhibit this particularity (no numerical data after an address assumed to be 0).

This is done by parameter E11030 (1 at power up meaning missing numerical value forbidden.

Error 1 extension 138 will be issue when the condition is trespassed

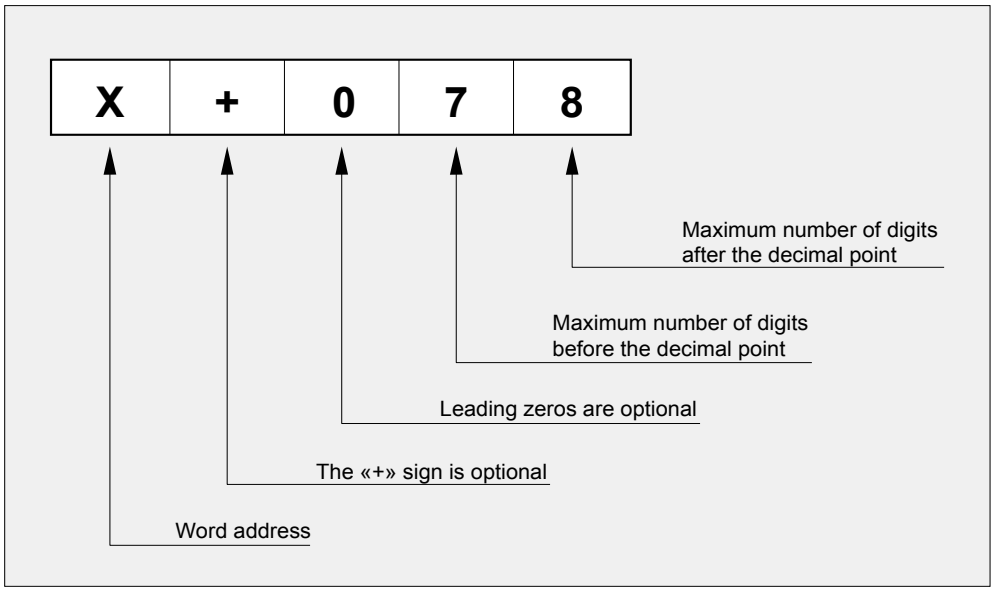
NCERR_1	Programming error
EXT1_138	Ambiguous ISO block (E11030=1)

A warning will be displayed when E11030 is changed

NCWRN_1010	Phantom Zero Check Enabled (E11030 = 1)
NCWRN_1011	Phantom Zero Check Disabled (E11030 = 0)

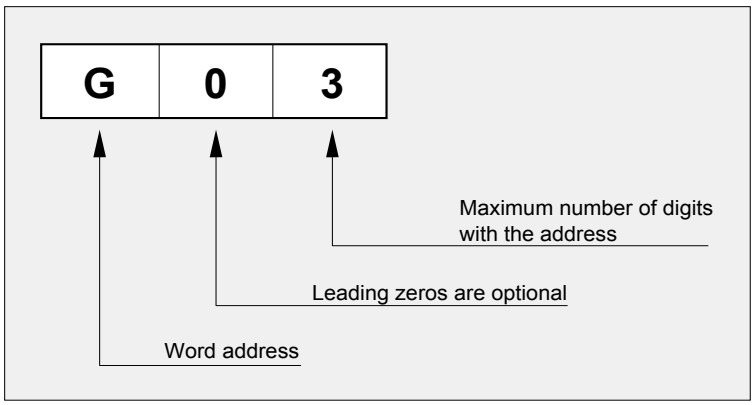
3.1.2 Examples of word formats

3.1.2.1 Word defining a dimension, address X (internal unit in µm)



The dimension 0.450 mm in X+053 format (variable word format), can be written:
X+0.450 or X.45

3.1.2.2 Word defining a function, address G



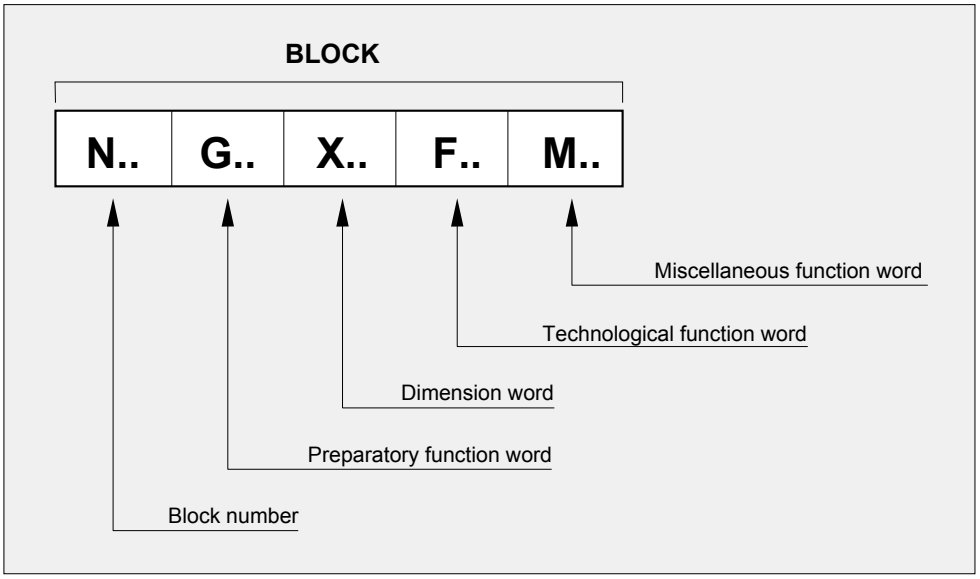
G function words in G02 format (variable word format).

- Word G01 can be written: G1
- Word G04 can be written: G4

3.2 Block format

A block (or sequence) is a list of instructions that will be executed by the control system either simultaneously or in a predefined order.

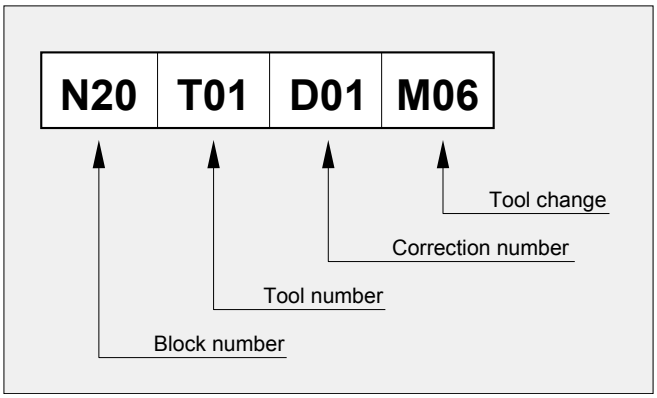
The order of execution and interference between functions will be detailed later in this chapter.



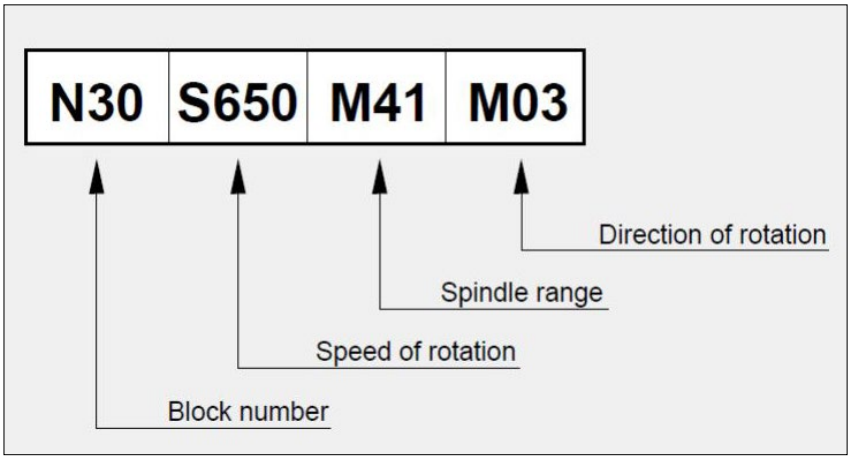
The block number itself (N) can be defined on 10 characters with a maximum value of 2^31-1 = 2147483647.

3.2.1 Examples of blocks

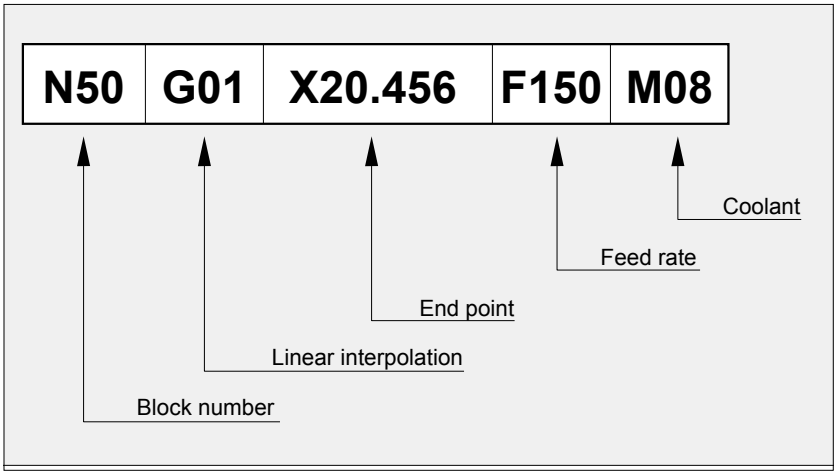
3.2.1.1 A block defining a tool change and calling up the tool correction



3.2.1.2 A block defining spindle rotation



3.2.1.3 A block defining a move



3.3 General structure of a program

3.3.1 General

An NC program always starts with a **"%"** character.

A program is generally executed in the order in which the blocks are written. The different blocks can be numbered or not. The bloc numbers have no effect on the order of execution with one particular exception explained below.

A block can contain Jump and Branch instructions which will change the execution order. The target for a Jump is always searched from the beginning of the program.

In this case if a Jump is programmed to a block number defined twice, always the first occurrence will be found. There is no error detected.



The special character **%** used to identify the start of a program must not be used in a comment.

Structure of an ISO Program

Program start: % character

Program end: code **M02**



M02 marks the end of program execution, it will generate a RESET. Therefore it should not be used in Sub-Programs (Macros).

Comments

Comments are helpful in the program:

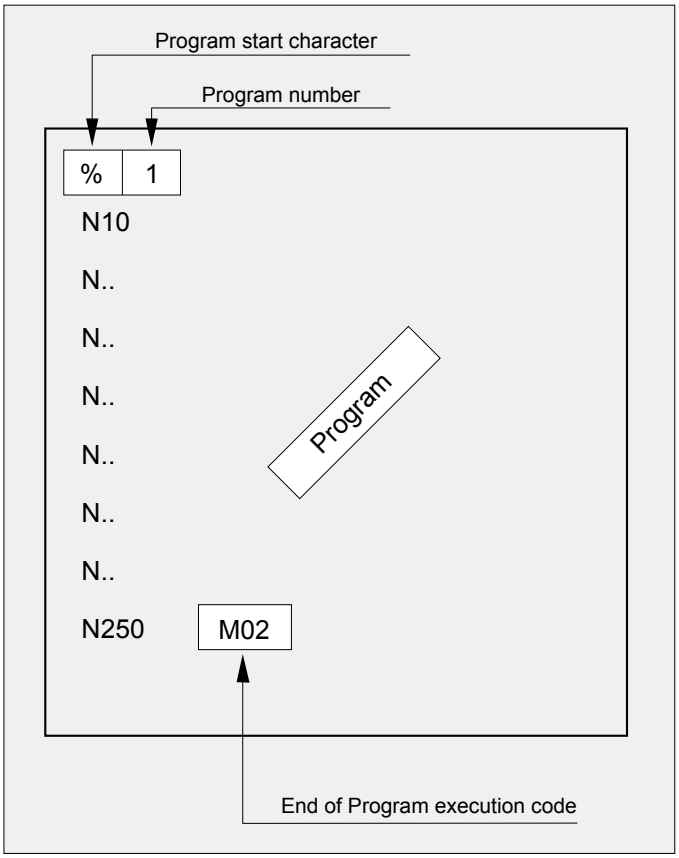
- They can be shown to the operator (see $ messages) for instructions.
- They can also be only included in the program body for explanations.

Two syntaxes of comments are valid:

- Between parentheses (**...**) : Just the part within the parentheses is commented and ignored by the interpolator.
- Introduced by a double back slash **** : The rest of the line will be ignored.

Example :

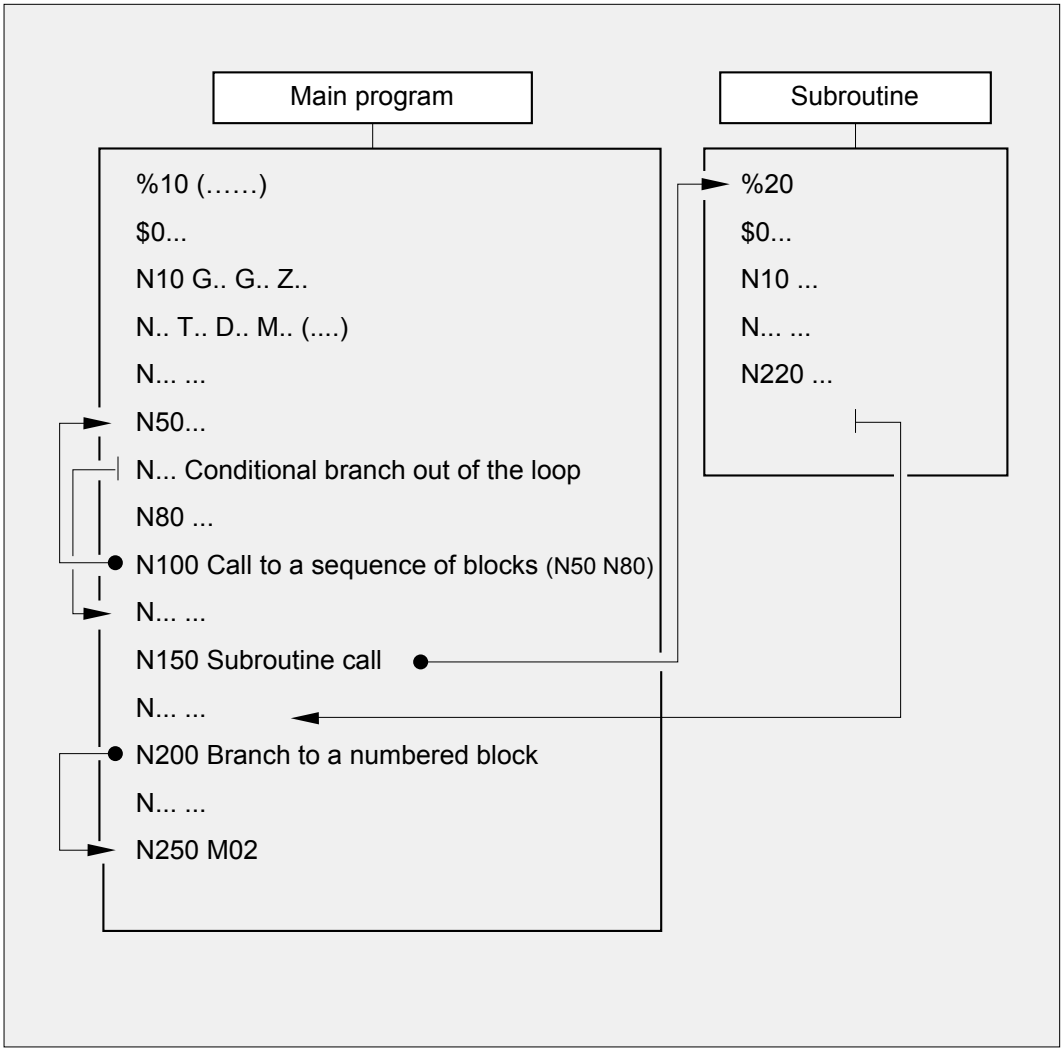
```
X0Y0
/   X10Y10          single slash:  this line is not executed when Block Skip is enabled
\\  X20Y20          double back-slash:  this line is never executed
X30 \\ Y30          only X30 is executed
```



3.3.2 Branches and Subroutine Calls

Particular instructions (branches and subroutine calls) can modify the order in which a program is executed. Branches can be conditional or not.

A program can be structured as follows:



3.3.3 Program Numbering

Program number: The permissible format is %051.

The **%** character is followed by a program number and possibly by a comment in brackets.

It is not possible to program ISO function on the same line as the program number.

The highest possible program number is 29999.9.

However it is not possible to activate a program higher than 9999. Higher number are reserved for subroutine or storage files.

The HMI settings adapted by the OEM may limit the displays of programs whose number is higher than a certain value.

Example

```
%324 (PART No. 72 - PROG 3)
```

A program number can be completed by a single digit index (see rules below).
(for multi channels systems see chapter 5.24)

Example

```
%425.2 (PROG FOR Channel 2)
```

Rules for program index

- In single channel the main program index must always be 0 or absent.
- In multi channel, the main program index is the same as the channel number, whether it's common mode or independent mode.
- In multi channel common mode the program number (excluding the index) must be identical for all channels.
- The subroutines invoked by the main program can have any extension. A not necessary but good practice is to have the same extension as the channel in which they will run or .9 in case of sub routine common for several channels.



Programs with numbers above %9000 may be used by NUM and the OEM integrating the NC on the machine (check with NUM or the OEM for possible use of these numbers).



The NUM or OEM specific programs are located in a different memory area. The search priority is User area first then OEM area and finally NUM area. Therefore if you create a program number already used by the OEM or NUM, the later will never be executed probably causing some machine malfunction. No error is detected as this is a voluntary feature allowing to overwriting for customization pre-existing cycles.

3.3.3.1 Inhibiting display of subroutines being executed

The content of a subroutine and its embedded subroutines can be masked during the execution (E.G. Long calculation sequence which could just fill the screen with difficult to understand data).

To mask the content it is necessary to insert a colon ":" after the program number : e.g. %110: (hidden execution)



The subroutine can still be edited. The data are just masked during execution and replaced by the calling block.

3.4 Word format - Main summary table

The formats of the axis words specified in the table are expressed as follows:

- For linear axes: 5 digits to the left and 3 to the right of the decimal point are allowed when the internal system unit (see chapter 3.1) is µm.
- For rotary axes: 3 digits to the left and 4 to the right of the decimal point are allowed when the internal system unit is 0.0001 degree.

For instance, for linear axes: if the system is set to 0.1µ (internal unit), the formats are expressed with 4 decimal digits; for the X axis, the format is X+044.

For instance, for rotary axes: if the system is set to 0.001 degree (internal unit), the formats are expressed with 3 decimal digits; for the B axis, the format is B+033.

For the words related to the machining feed rate with no assigned formats (F.., EF..), refer to the machine manufacturer's manual for the maximum and minimum feed rates (maximum 8 digits and decimal point).

Syntax convention (numerical part)

- Sign : Default (+ = positive value)
- First digit = 0 : leading zeros are authorized and optional
- First non zero digit : Number of characters before the decimal point
- Last digit : Number of characters after the decimal point.

E.g. B+034 : value positive by default, 3 characters before the decimal points and 4 characters after.

Format	Description
%051	Program number (1 to 99999.9)
N10	Sequence number (1 to 2147483647)
G02 **G03**	Preparatory functions (0 to 99) Preparatory functions (100 to 250 and 997 to 999)
H05	Subroutine number (with G77, G76 and G48)
X+076	X axis coordinate
Y+076	Y axis coordinate
Z+076	Z axis coordinate
I+076 **J+076** **K+076**	For circular/helical interpolation (G02, G03), center of the circle and pitch of the helix With programmed origin offset (G59), center of rotation and angular offset (ED) For thread chasing cycle (G31), thread pitch = I, J or K depending on the interpolation plane
I076	For pocket cycle (G45), axial finishing pass
J076	For pocket cycle (G45), lateral finishing pass
K076	For rigid tapping cycle (G84), tap pitch
U+076	U axis coordinate
V+076	V axis coordinate
W+076	W axis coordinate
A+076	A axis coordinate
B+076	B axis coordinate
C+076	C axis coordinate or spindle indexing position in case of turning channel
E+/E-	For profile geometry programming (PGP), discriminant
EA+033 **EA+033** **EA+033**	For profile geometry programming (PGP), line angle In a boring cycle with indexed spindle stop (G86), angle between EC.. and tool cutting edge In polar programming, angle of the line

EB+053 **EB-053** **EB053**	For contour definition and PGP, radius or fillet between two interpolations For contour definition and PGP, chamfer between two linear interpolations For profile Geometry Programming (PGP), chamfer between 2 lines
EC+033 **EC+033** **EC+033**	For spindle indexing (M19), indexing angle (case of milling channel) For thread chasing cycle (G31), tool orientation in the hole bottom For boring cycles (G86), indexing position
ED+034	Programmed angular offset
EF022 **EF..** **EF..** **EF..**	For cycles (G82, G84 "not rigid tapping", G87, G89 or G31), dwell time In a reaming cycle (G85), retraction rate Feed rate specific to fillets (EB+) or chamfers (EB-) Maximum feed rate after an interrupt (G10)
EG01	Pocket cutting direction (EG02: clockwise; EG03: counter-clockwise)
EG03	For interpolation, acceleration modulation
EH+053	In (G8x) cycles, dimension of the impact plane on the machining axis
EI+053 **EI..**	In polar programming, length of the line (start/circle center) In a pocket cycle (G45), axial finishing feed rate
EJ..	In a pocket cycle (G45), lateral finishing feed rate
EK01	For rigid tapping cycle (G84), retraction/penetration speed ratio
ES+/ES-	For profile geometry programming (PGP), secant element
ET+/ET-	For profile geometry programming (PGP), tangent element
EP053 **EP053** **EP053** **EP..**	In a boring cycle with indexed spindle stop (G86), lateral back-off in hole bottom In a peck drilling cycle (G83), back-off clearance after each peck In a drilling cycle with chip breaking (G87), back-off between two in-feeds In a pocket cycle (G45), axial roughing feed rate
EQ053	For pocket cycle (G45), lateral roughing feed rate
ER+053	In cycles, infeed or retraction dimension on the machining axis
ES02	In cycles G83 and G87, number of in-feeds at constant value
EX+076 **EX076**	In polar programming, angle of the line (start/end) For pocket cycle (G45), dimension along X (or U)
EY076	For pocket cycle (G45), dimension along Y (or V)
EZ076	For pocket cycle (G45), dimension along Z (or W)
P+053 **P+043** **P053** **P043** **P041** **P+ / P-**	For thread chasing cycles (G31), withdrawal of tool For 3D correction (G29), X component of the normal vector For cycle (G83 or G87), value of the first penetration For pocket cycles (G45), axial roughing path With axis channel synchronization (G78), wait for a marker Tool axis orientation (G16), along X (or U)
Q+053 **Q+043** **Q043** **Q04** **Q+ / Q-**	For cycles (G83 or G87), value of the last penetration For correction in space (G29), Y component of the normal vector For pocket cycle (G45), lateral roughing pass With channel synchronization (G78), declaration of a marker Tool axis orientation (G16), along Y or V
R+076 **R053** **R+043** **R+ / R-** **R+ / R-**	For circular/helical interpolation (G02, G03), radius of the circle For tangential feed rate programming (G92), value of the curve radius For space correction (G29), Z component of the normal vector For linear interpolation (G00, G01), positioning at programmed distance Tool axis orientation (G16), along Z or W
F.. **F..** **F..** **F..** **F..** **F022** **F01**	Feed rate in mm/min and degrees/min (G94) Feed rate in inches/min (G94) Feed rate in mm/rev (G95) Feed rate in 1/t (G93) Feed rate in inches/revolution (G95) Dwell in seconds (G04) For thread chasing cycles (G31), number of threads (1 to 9)

M02 **M03**	Miscellaneous functions 0 to 99 Miscellaneous functions 100 to 899
NC04	For spline curve interpolation (with G06, G48, G49), curve number (1 to 9999)
S032 **S05 or** **S06**	Spindle rotation speed in rpm (G97) In a subroutine branch (G77), number of subroutine iterations (1 to 1000000)
T08	Tool number (0 to 99999999)
D03	Correction number (0 to 255)
L03	Program variables L (0 to 99, 100 to 199 and 900 to 959)
E5 / E7	External parameters E

3.5 Classification of preparatory and miscellaneous Functions

3.5.1 Classification of G Functions

The G functions, also named preparatory functions are of two types:

- Modal G functions, stay active as long as they are not cancelled by an other G function or a reset.
- Nonmodal G functions, active in the current block only.

Moreover it is important to notice that:

- Certain G functions can be incompatible with the current state of the execution.
- Certain G functions must be programmed with the associated arguments in the same block (the order of declaration can be important).

3.5.1.1 Modal G Functions

Functions belonging to a family and that cancel one another (e.g. interpolation).

Certain families of G functions include a default function that is initialised at power up or after a reset. (See chapter 5).

These functions remain enabled until cancelled by another function of the same family.

Example for Milling

`N.. G00 X.. Y..`	High-speed linear interpolation
`N.. G01 Z..`	G00 cancelled by linear interpolation at machining feed rate

Example for Turning

`N.. G00 X.. Z..`	High-speed linear interpolation
`N.. G01 G20 Z..`	G00 cancelled by linear interpolation at machining feed rate

3.5.1.2 Nonmodal G Functions

Functions enabled only in the block where they are programmed (cancelled at the end of the block).

Example for Milling and Turning

`N.. G09 X..`	Accurate stop at end of block cancelled at end of block.

3.5.1.3 G Functions not compatible with the program status

Functions whose programming is enabled or not according to the status of the current program.

Example for Milling

`N.. G18 G41 X.. Y..`	Correct syntax, ZX plane selection (G18), followed by cutter compensation (G41)
`N..`	
`N.. G41 G18 X.. Y..`	Incorrect syntax, change of plane prohibited when in cutter compensation.

Example for Turning

`N.. G21 G42 X.. Y.. Z..`	Correct syntax, change of X Y Z coordinate system (G21), followed by radius offset (G42)
`N..`	
`N.. G42 G21 X.. Y.. Z..`	Incorrect syntax, change of coordinate system prohibited with radius offset.

3.5.1.4 G Functions associated with arguments

Some function are associated with specific arguments.The argument (s) must immediately follow the function. The analysis of the arguments of a G function is ended by reading a word that does not belong to the list of arguments of this function.

Example for Milling

```
N.. G04 F2 T03 F200
```
Correct syntax

```
N.. G04 T03 F2 F200
```
Incorrect syntax, argument F2 does not immediately follow G04

Example for Turning

```
N.. G04 F2 T03 F0.2
```
Correct syntax

```
N.. G04 T03 F2 F0.2
```
Incorrect syntax, argument F2 does not immediately follow G04.

When a G functions has several arguments, they can be programmed in any order except for G functions that introduce breaks in the sequencing (G10, G76, G77 and G79, see chapter 5.15). The arguments associated with a function can be:

- compulsory,
- optional.

Compulsory Arguments

The arguments are compulsory if:

- the G function serves only to announce arguments.

Example:

```
N.. G16 P+
```
G function and its argument P+

- the G function cancels a former modal state and characterizes its argument differently.

Example:

```
N.. G94 F100
N..
N.. G95 F0.5
```
Feed in mm/min

The change from feed in mm/min to mm/revolution requires redefining argument F

Optional Arguments

The arguments are optional if the G function allows them to be defined by default.

Example:

```
N.. G96 [X..] S150
```
Case where the X position (with respect to OP) was specified by an earlier block (G96: constant surface speed for a turning machine)

Arguments Programmed Alone

The argument can be programmed alone in a block when the associated G function is still active.

Example for Milling

```
N.. G94 F150 X.. Y..
N..
N.. X.. Y.. F100
```
Feed in mm/min

Example for Turning

```
N.. G94 F150 X.. Z..
N..
N.. X.. Z.. F100
```
Feed in mm/min

Function G94 is not compulsory with its argument because the system is still in state G94.

3.5.2 Classification of M Functions

The M functions, also named miscellaneous functions are also from two types:
- Modal M functions,
- Nonmodal M functions.

M functions can be:
- «*Pre*», «*Post*» or "*On the fly*" functions,
- decoded or encoded functions (non relevant for part programming).

There is also a type of function which is timed with respect to the execution. Being reserved for special applications they are described in the Flexium+ Extended Programming manual M00055.

M functions with arguments:
Some M function can be associated with optional arguments (*: and #*). They are related to spindle programming. For details please refer to the particular chapter '*Spindles on Flexium+*'.

3.5.2.1 Modal M Functions

Functions belonging to a family of M functions that cancel one another (e.g. Spindle rotation).

Certain families of M functions include a default value function that is initialized at power up or after a reset. (See chapter 5).

These functions remain enabled until they are cancelled by another function of the same family.

Example for Milling and Turning:

`N.. S500 M03`	Start of spindle rotation
`N.. M05`	Spindle stop, cancels M03

3.5.2.2 Nonmodal M Functions

Enabled only in the block where they are programmed.

Example for Milling and Turning:

`N.. M00`	Program stop

3.5.2.3 «Pre» M Functions

Functions executed before axis movements programmed in the block.

`N.. X100 Z50 M08`	Coolant function M08 is executed before the movements on X and Z.

3.5.2.4 «Post» M Functions

Functions executed after the axis movements programmed in the block.

`N.. X50 Y100 M09`	The coolant off function (M09) is executed after movements on X and Y

3.5.2.5 Decoded M Functions

The decoded M functions are the basic system functions whose meaning and behaviour are known.

Each one can be either Pre or Post, Modal or non Modal.

Generally activation functions are *"Pre"* whereas deactivation are *"Post"*.

Example for Milling and Turning:

```
N.. T01 M06
```
Tool change function M06

Several decoded M functions can be programmed in the same block.

Example for Milling and Turning:

```
N.. G97 S500 M03 M40 M08
```

3.5.2.6 Encoded M Functions

The encoded functions are defined by the machine manufacturer and are specific to the machine (See manufacturer's technical data). They include all M codes except those already defined as Decoded.

The modality as well as the exclusion is managed by the machine manufacturer. They have two functionalities according to the function number.

Standard encoded M functions (default M100 to M199)

These functions are *"Post"* functions, they require an acknowledgement (by the PLC).

Up to three such M functions can be programmed in one block. They will be executed in the order they are read in the block.

If more than three M functions are programmed then error 1 ext 134 will be issued.

On the fly encoded M functions (default M200 to M899)

These functions are modal *«pre»* functions. The program continues immediately after issuing the function. These functions are called *"On the Fly"* M functions because they are executed at the beginning of the block without interruption of the movement.

They do not require any acknowledgement.

Only one of these functions can be programmed in a block. If more than one is programmed only the last one will be taken into account.

They can however be programmed on the same block that other types of M function (decoded and/or standard encoded).



As these functions do not require acknowledgement use caution when programming several M *"On the fly"* consecutively. Be sure that there is enough time between each (dwell or axes movement) for the machine to take them into account. Refer to the machine operation manual.



The M100 to M 199 and M200 to M899 are the default values. They can be adapted by the machine builder. Please refer to the machine documentation.

Page deliberately empty

Summary

4 Axis Programming 61
4.1 General 61
4.2 Programming the independent secondary axes 62
4.3 Programming with reference to Diameter or Radius 63
4.3.1 System programmed with Reference to Diameter 63
4.3.2 System programmed with Reference to Radius 63
4.4 Programming of carrier/carried parallel axis pairs 64
4.5 Programming rotary axes modulo 64
4.5.1 Rotary axis programmed in absolute dimensions (G90) 64
4.5.2 Rotary axis programmed in incremental dimensions (G91) 66
4.6 Programming rotary axes with limited travel 67

Page deliberately empty

4 Axis Programming

4.1 General

Programmable axes:
- Primary axes X, Y, Z,
- Secondary axes U, V, W,
- Rotary axes A, B, C.

Primary and secondary axes:
- They can be independent or form carrier/carried axes pairs,
- They can be programmed in millimetres (basic unit) or inches.

Rotary axes:
- They can be modulo 360 degrees or have limited travel,
- They are programmed in degrees (basic unit).

Reminder

Definition of the internal system measurement units

The internal dimension unit is defined by the OEM when integrating the CNC. It does not affect the machine travels, but may affect the maximum speed. Please refer to the machine documentation.

The number of decimal digits is also defined by the OEM, it can be different from the internal measurement and determines the word formats (see chapter 1.10 and chapter 3.1).

For linear axes, the internal unit can be from 0.1mm down to the nanometer.

For rotary axes, the internal unit can be from 0.1 degree down to the millionth of degree.

For additional information, refer to the machine operation manual.

4.2 Programming the independent secondary axes

Programming the independent secondary axes U, V, W is unrelated to the programming of the primary axes X, Y, Z.

The secondary axes U, V, W are respectively parallel to the primary axes X, Y, Z.

It should be noted that the tool length correction (L for Milling Machine offsets and X, Z for Turning Machine offsets) is not applied on the secondary axes.

The tool radius correction is applied.

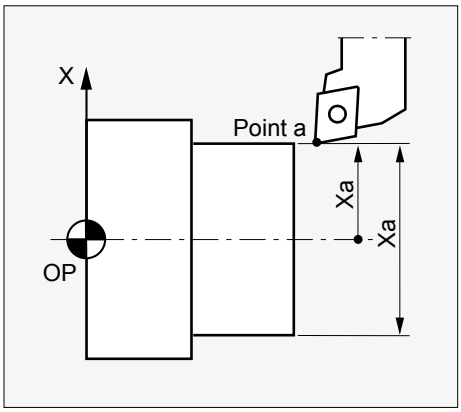
4.3 Programming with reference to Diameter or Radius



The part program and certain data related to machining along the X (or U) axis are directly affected by whether programming is with reference to diameter or radius.

Programming with reference to diameter or radius is selected by the machine builder.

In both cases, certain functions are always expressed with respect to diameter and others with respect to radius.



4.3.1 System programmed with Reference to Diameter

Programmed values expressed with reference to diameter:
- values programmed in absolute dimensions (G90): coordinates of a movement along X.. and position I.. of the center of a circle,
- value of the starting diameter with constant surface speed (G96),
- value programmed with function G98.

Programmed values expressed with reference to radius:
- values programmed in incremental dimensions (G91): value of a movement along X.. and position I.. of the center of a circle,
- circle radius with circular interpolation (R),
- fillet or chamfer (EB+, EB-),
- pass depth for roughing cycle (P or R),
- machining allowance for roughing cycles (I or K), (ER),
- positioning clearance for roughing cycle (Q),
- minimum depth of cut for roughing cycle (EQ),
- thread depth (P) and last pass for thread cutting (Q),
- pass depth for drilling cycle (P and Q),
- offsets programmed with functions G59 and G52.

Tool dimensions:
- Entry of values with respect to radius.

Dynamic tool corrections:
- Entry of values with respect to diameter, but display of radial change on the context "Tools"» page.

Value of offset DAT2:
- Entry of the value with reference to radius.

Movements related to the manual controls:
- Movements on the X axis with reference to radius, but display with reference to diameter on the context "Production" page.

4.3.2 System programmed with Reference to Radius

Values expressed with reference to radius:
- All the programmed movements applied to the X axis and all the values entered related to machining along X.

Values expressed with reference to diameter:
- Only the dynamic tool corrections on X are entered with reference to diameter.

4.4 Programming of carrier/carried parallel axis pairs

Axes pairs (X/U,Y/V, Z/W) can be declared by the OEM as Carrier/Carried.

The tool tip position reflects the cumulated position of both axes. Only one axis of each pair can be programmed in a single block

The tool length compensation is taken on the primary axis of the pair (X, Y, Z).

4.5 Programming rotary axes modulo

4.5.1 Rotary axis programmed in absolute dimensions (G90)



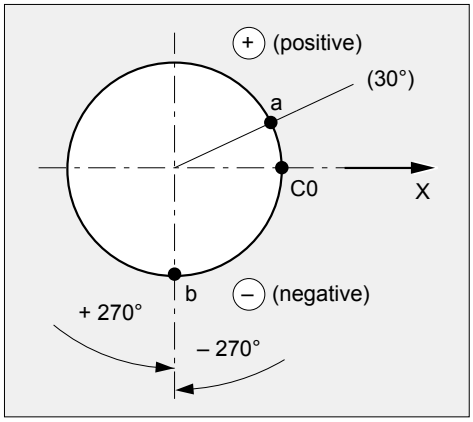
The programmed value assigned to the axis is the end angular position with respect to the program origin, regardless of the sign (value between 0 and the modulo value, maximum one revolution) .

As a general rule, the sign (+ or -) determines the direction of rotation used to reach this point. (See second remark)

Example

a : Start point.

b : End point.



Positive movement

```
N.. ...
N.. G90 B+270
N..
```

Negative movement

```
N..
N.. G90 B-270
N..
```

Both sequence of blocks will reach the same angular position.



Programming the same position results in a complete revolution or not according to the value of parameter E11009. However the distinction was not always clear in particular in case of several turns or a modulo different of 360. Starting with version 4.1.30, the functionnality of E11009 has been adjusted 1 --> Enable or 0 --> Disable the full revolution in all cases.

This could lead to some discrepancies in existing programs. Please check in particular when E11009 is used in incremental programming.position. Refer to E parameter E11010.

Please note the following particularity in case the position is given by a parameter. The sign of the parameter will define the position to reach.
To define the negative direction it is necessary to precede this parameter by a - sign.

Example E80000 = -30
CE80000 will move the C axis to the position 330° in POSITIVE direction.
To move in the negative direction it is necessary to program C-E80000 and here again to final position will be 330°

The settings made by the machine builder can override the direction of rotation convention. In such a case, the direction will be the defined by the shortest path to reach the target position. Refer to E parameter E11010.

Starting with version 4.1.20 the modulo value can be freely defined.
The modulo value is defined by machine parameters and can be read in E93232 to E93263.

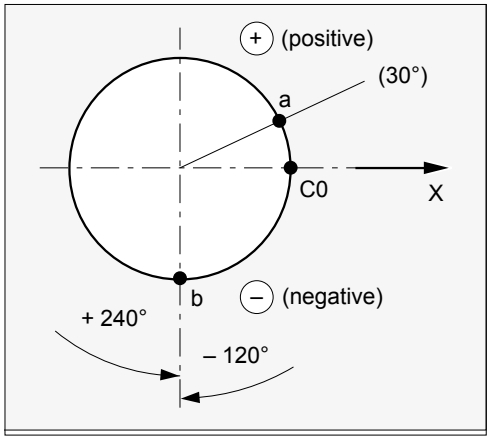
4.5.2 Rotary axis programmed in incremental dimensions (G91)

The programmed value assigned to the axis indicates the amplitude of rotation with respect to the previous position (see chapter 5.3 for function G91).

Example

a : Start point.

b : End point.



Positive rotation

```
N.. ...
N.. G91 B+240
N..
```

Negative rotation

```
N.. ...
N.. G91 B-120
N..
```

With incremental programming G91, a movement of more than one revolution is allowed on modulo rotary axes A, B or C.

The maximum amplitude is 16383 revolutions (5897880 degrees).
If this value is exceeded, the system returns error message 1.

Caution: remember that the first movement in a program should always be absolute (G90) to correctly set the position.
If the first movement is programmed in incremental (G91), it will be considered as absolute anyway. No error message will be generated.

In Turning Machine configuration only (G20) please note that the speed on the C axis is related to the position of the X axis. The programmed speed will be applied in mm/min at the tool contact point.

4.6 Programming rotary axes with limited travel

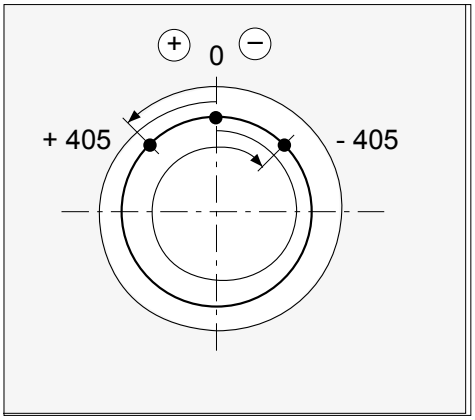
Rotary axes A, B or C with limited travel are defined by the OEM like linear axes and therefore follow the same programming rules, in particular the sign defines the final coordinate and not the direction of travel.

The travel for such an axis can be from a fraction of revolution to several revolutions. The maximum limit being +/- 99 999 999 degrees. For the current limits see the machine operation manuals.

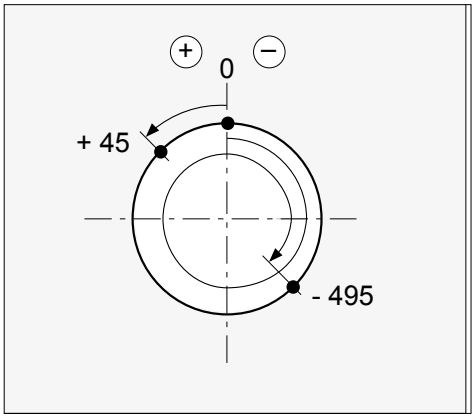


Example

Rotation by more than one revolution in absolute dimensions (G90).



Rotation by more than one revolution in incremental dimensions (G91).



The speed of movement on axes A, B or C declared as nonrotary is expressed in Units/min (therefore generally degrees/min).

However, if they are programmed in a block together with primary and secondary axes X, Y, Z, U, V or W, the programmed speed is assigned to the main axes.

Page deliberately empty

Summary

5 ISO Programming 75

5.1 Functions and programming addresses 75
5.1.1 G Functions - Main summary table 75
5.1.2 M Functions - Main summary table 91
5.1.3 Other Functions - Main summary table 95

5.2 Initialisation 98
5.2.1 Absolute or Incremental selection ► G90, G91 98
5.2.2 Plane selection ► G17, G18, G19 100

5.3 Spindles control 101
5.3.1 Spindle control selection ► M62 102
5.3.2 Spindle range ► M40, M41, M42, M43, M44, M45 104
5.3.3 Spindle rotation selection ► M03, M04, M05 105
5.3.4 Indexed spindle stop ► M19 107
5.3.5 Spindle release ► M61 109
5.3.6 Spindle measurement selection ► M66 110

5.4 Spindle speed 111
5.4.1 Spindle speed control ► G97 111
5.4.1.1 Review of surface speed 112
5.4.1.2 Using a fixed spindle rotation speed (N) 112
5.4.1.3 Using a constant surface speed (V) 113
5.4.1.4 Spindle speed limit 113
5.4.2 Spindle speed limitation ► G92 S.. 114
5.4.3 Constant surface speed on Turning machine ► G96 116

5.5 Axes Feed rate 117
5.5.1 Feed rate expressed in Millimetres, Inches or Degrees per Minute ►G94 117
5.5.2 Feed rate expressed in millimetres or Inches per revolution ► G95 119
5.5.3 Feed rate in inverse of time (1/min) ► G93 121
5.5.4 Case of rotary axes 125
5.5.4.1 Programming of a modulo rotary axis alone 125
5.5.4.2 Programming of modulo rotary axis and a linear axis 125
5.5.4.3 Programming of two modulo rotary axes 125
5.5.4.4 C axis feed rate for Turning machine 126
5.5.5 Feed rate specific to Fillets EB+ and Chamfers EB- ► EF.. 127
5.5.6 Tangential feed rate ► G92 R.., G92 L.. 130
5.5.7 Setting feed rate per axis set ► G92 F.. 133
5.5.8 POLAR CONSTANT FEEDRATE ► G20 G92 F… P 134
5.5.9 XY CONSTANT CONTACT SPEED ► G20 G92 F… Q 135
5.5.10 CONSTANT C SPEED ► G21 G92 F… C 136
5.5.11 Linear feed variation ► G94/G95F.. EF.. 138

5.6 Axes programming and interpolation 140
5.6.1 Rapid positioning ► G00 140
5.6.1.1 Positioning with respect to the cutter diameter (R-/R+) 142
5.6.2 Linear Interpolation ► G01 144
5.6.2.1 Positioning with respect to the cutter diameter (R+/R-) 145
5.6.2.2 Programming of additional axes and carrier/carried axis pairs 146
5.6.3 Circular interpolation ► G02, G03 148
5.6.3.1 Direction of movement according to the turret position with Turning machine 150
5.6.3.2 Programming a circle by its radius (R..) 151
5.6.3.3 Programming a circle by the coordinates of its center (I and J, plane G17) 151
5.6.3.4 Programming the additional axes and carrier/carried axis pairs 152

5.6.4 Polar programming 153
5.6.4.1 Polar programming of a line 154
5.6.4.2 Polar programming of a circle 156
5.6.4.3 Defining a circle by the arc angle 160
5.6.5 Programming fillets and chamfers 166
5.6.5.1 Fillet between two interpolations ► EB+.. 166
5.6.5.2 Chamfer between two linear interpolations ► EB-.. 167
5.6.6 Helical interpolation 169
5.6.6.1 Determination of the helix depth 170
5.6.6.2 Circular interpolation defined by three points ► G23 173
5.6.7 Cylindrical grinders specificity 175
5.6.7.1 Initial tool positioning before machining on an inclined axis ► G07 177
5.6.7.2 Movement along the inclined axis ► G05 178
5.6.8 Coordinates systems with C Axis with Turning machine 180
5.6.8.1 Programming in Polar Coordinates ► G20 180
5.6.8.2 Definition of the Start X for Interpolation on the C Axis ► G98 181
5.6.8.3 Programming in cartesian coordinates ► G21 182
5.6.8.4 Extended Features of G21 186
5.6.8.5 Programming in cylindrical coordinates ► G22 188

5.7 Program offsets 191
5.7.1 Programming of movements in absolute coordinates referenced to the measurement origin ► G52 191
5.7.2 Datum shift (DAT1 and DAT2) cancel ► G53 194
5.7.3 Selection and activation of datum shift 1 (DAT1) ► G54, G55, G56, G57, G58 195
5.7.4 Selection and activation of datum shift 2 (DAT2) ► G501.. G599 196
5.7.5 Program origin preset ► G92 197
5.7.6 Program origin offset ► G59 199
5.7.6.1 Function G59 programmed in absolute dimensions (G90) 200
5.7.6.2 Function G59 programmed in incremental dimensions (G91) 200
5.7.6.3 Features related to arguments I, J (K) 200
5.7.7 Angular offset ► ED.. 210
5.7.8 Table offset by DAT3 with Milling machine ► DAT3 214

5.8 Tool offsets 216
5.8.1 Tool axis orientation ► G16 216
5.8.2 Tool change ► M06 219
5.8.3 Tool Offset activation ► D.. 222
5.8.3.1 Case with Tool Management function 222
5.8.3.2 Tool length correction (L) 224
5.8.3.3 Tool dimensions corrections 224
5.8.3.4 Tool radius corrections 224
5.8.3.5 Tool tip orientation (C) according to the turret position 226
5.8.3.6 Use of tool corrections (X, Z, R and C) on a lathe with two mechanically coupled turrets 226

5.9 Tool correction 228
5.9.1 Tool radius compensation ► G41, G42, G40 228
5.9.1.1 Tool positioning 231
5.9.1.2 Radius correction according to the turret position 232
5.9.1.3 Tool outside the profile (line/line or circle/circle) 233
5.9.1.4 Tool outside the profile (line/circle) 235
5.9.1.5 Tool inside the profile 235
5.9.1.6 Tool inside the profile, special cases 236
5.9.1.7 Tool disengagement 237
5.9.2 Enhanced tool radius correction 247
5.9.3 3D Tool compensation 249
5.9.3.1 Concept of surface and normal vector 249
5.9.3.2 Toroidal and spherical tools 250
5.9.3.3 Double twist Machine Head 250
5.9.3.4 Geometric transformations with 3-Axis Correction 251
5.9.3.5 Geometric transformations with 5-Axis Correction 253
5.9.3.6 Tridimensional tool correction with toroidal or spherical tool ► G29 254
5.9.3.7 Tridimensional tool correction with cylindrical Tool and RTCP ► G43+/- 256

5.10 Canned cycles 257
5.10.1 Cycle overview ▶ G8x 257
5.10.2 Cancellation of a canned cycle ▶ G80 259
5.10.3 Center drilling cycle ▶ G81 260
5.10.4 Counterboring cycle ▶ G82 262
5.10.5 Peck drilling cycle ▶ G83 264
5.10.6 Tapping cycles ▶ G84 268
5.10.7 Tapping with chip breaking ▶ G84 271
5.10.8 Rigid tapping cycle ▶ G84 274
5.10.9 Rigid tap removal ▶ G39+ 278
5.10.9.1 Cycle execution....279
5.10.10 Reaming cycle ▶ G85 280
5.10.11 Boring cycle with indexed spindle Stop at the bottom of the hole ▶ G86 282
5.10.12 Drilling cycle with chip breaking ▶ G87 284
5.10.13 Boring and Facing cycle ▶ G88 288
5.10.14 Boring cycle with Dwell at the bottom of the hole ▶ G89 290
5.10.15 Thread chasing cycle ▶ G31 295
5.10.15.1 Cycle repeat to achieve the thread precision 297
5.10.16 Cycle summary 298
5.10.17 Examples of G81 to G89 cycles 299

5.11 Milling cycles....302
5.11.1 Simple pocket cycle ▶ G45 302
5.11.1.1 Pass setting possibilities....304
5.11.1.2 Roughing cycle with a single tool....305
5.11.1.3 Roughing Cycle Steps 305
5.11.1.4 Roughing and finishing cycle with a single tool 306
5.11.1.5 Finishing cycle alone with a new tool....306
5.11.1.6 Pass setting in the plane (Q)....307
5.11.1.7 Pass setting along the tool axis (P) 307
5.11.2 Spline interpolation ▶ G48, G06, G49 310
5.11.2.1 Spline curve definition ▶ G48 311
5.11.2.2 Spline curve execution ▶ G06 313
5.11.2.3 Spline curve deletion ▶ G49 318
5.11.3 Tridimensional curve smoothing ▶ G104 319
5.11.4 Corner rounding function ▶ G164 321
5.11.5 Trochoidal milling ▶ G725 323
5.11.6 Engraving ▶ G730 325
5.11.7 Tangential Control ▶ G748 327

5.12 Turning specific canned cycles....329
5.12.1 Constant pitch thread chasing cycle ▶ G33 329
5.12.1.1 Tool retraction slope at the end the pass defined by R 331
5.12.1.2 Penetration defined by EB 332
5.12.1.3 Tapered thread cutting, angle defined by EA 332
5.12.1.4 Multiple start thread cutting 333
5.12.1.5 Multi cut threading cycle 335
5.12.2 Sequenced thread cutting ▶ G38 337
5.12.3 Resume thread cutting ▶ G38+ 339
5.12.3.1 Inclusion of function G38+ in Cycle G33 340
5.12.3.2 Associating a Teach-in cycle with resume thread cutting cycle G33 341
5.12.4 Rough Turning/Facing cycle ▶ G64 342
5.12.4.1 Roughing cycle P or R 345
5.12.5 Groove roughing cycle ▶ G65 350
5.12.6 Plunging cycle ▶ G66 355
5.12.7 Roughing cycle with groove ▶ G63 359
5.12.7.1 Definition of EB and EC 363
5.12.7.2 Cycle differences between turning and facing....364

5.13 Jumps and branches 370
5.13.1 Restrictions related to drip feed mode....370
5.13.2 Unconditional branch to a subroutine or sequence blocks with return ▶ G77 371
5.13.2.1 Call to subroutine return block ▶ G77 -i 377
5.13.3 Branch to a sequence without return ▶ G79 378

5.13.4 Subroutine call by M function ► M.. 380
5.13.4.1 Subroutine call by M function..... 380
5.13.5 Subroutine call by PLC 381
5.13.6 Emergency retract ► G75 382
5.13.7 Block interrupt / Probing ► G10 385
5.13.7.1 Analysis of the interrupt 387
5.13.7.2 Positions in programmed frame..... 389
5.13.7.3 Special use of sequence interrupt 390

5.14 Standard functions 392
5.14.1 Dwell ► G04 392
5.14.2 Accurate end of block Stop ► G09 393
5.14.3 Feed increase ► G12 395
5.14.4 Mirror function ► G51 396
5.14.4.1 Influence of the mirror function on the A, B or C rotary axes 397
5.14.4.2 Influence of the mirror function on circular interpolations.....397
5.14.5 Programming in Inches or Metric data ► G70, G71 399
5.14.6 Scaling factor ► G74, G73 401
5.14.7 Program Stop ► M00 405
5.14.8 Optional Stop ► M01 406
5.14.9 Coolant ► M08, M07, M09 407
5.14.10 Axis clamping and unclamping ► M10, M11 408
5.14.11 Overrides inhibit ► M49, M48 409
5.14.12 Block Skip ► / N 410

5.15 Advanced Miscellaneous functions 411
5.15.1 Automatic homing 411
5.15.2 Subroutine call on Reset %11000.....412
5.15.3 Programmed feed stop for intervention ► M12 413
5.15.4 Acceleration reduction ► EG.. 414
5.15.5 Spindle acceleration reduction ► EGS 415
5.15.6 Temporary suspension of next block preparation ► G79+/- 416
5.15.7 Calling a subroutine before executing a block.....418
5.15.8 Temporary inhibition of MDI access ► M999, M998 419
5.15.9 Force consecutive execution of blocks ► M997 421
5.15.10 Processing of blocks and programmed G and M functions
► G999, G998, G997 422

5.16 Dialog.....424
5.16.1 Message transmission ► $0 424
5.16.1.1 Message transmission to the display.....424
5.16.2 Message display with request for an operator response ► L../E..,$ 426
5.16.3 Display of messages with parametric value ► $ 428

5.17 Storage of values 429
5.17.1 Transfer of the current values of L Variables and E Parameters into the part .....
program ► G76 429
5.17.2 Format of the numerical values 431
5.17.3 Saving symbolic variables in a dedicated memory zone ► G76+ S, G76- S 432
5.17.4 Restoring symbolic variables from dedicated memory zone
► G76+ [Sym 1] ... [Sym n] S** 432
5.17.5 ISO program or block creation/deletion ► G76+/- 433
5.17.6 Checking the existence of a file ► G76=H 434
5.17.7 Creating a program ► G76+ H 435
5.17.8 Deleting a program ► G76- H 436
5.17.9 Inserting a block ► G76+ 437
5.17.10 Deleting a block ► G76- 438

5.18 Electronic Gear box 439
5.18.1 MLEGB Spindle continuous rotation ► G770 439
5.18.2 MLEGB Synchro Definition / Cancellation ► G771 440
5.18.3 MLEGB Activation / Deactivation ► G775 441
5.18.4 MLEGB Waiting for synchronization completion before proceeding ► G776 442
5.18.5 MLEGB Synchro Homing / Gear alignment ► G779 443

5.19 Special programming for Multiple Channels..... 444

5.19.1 Program declaration 444
5.19.2 Programming notes 444
5.19.3 Channel synchronization with markers ► G78 447
5.19.3.1 Programming a meeting point 449
5.19.3.2 Programming by function G78 449
5.19.3.3 Programming with M functions 450
5.20 Special programming of Auxiliary channels 451
5.20.1 Program declaration and storage 451
5.20.2 Programming of the Auxiliary channels 452
5.20.2.1 Emergency retraction on an Auxiliary Channel 452
5.20.2.2 Exchanging axes between channels 453
5.21 Special features of mixed Machines (MX) 454
5.21.1 Features related to the Machine Axes 454
5.21.1.1 Boring machine 455
5.21.1.2 Mixed type machine 455
5.21.2 Assignment of tool dimensions to the Axes 456
5.21.3 Preselection of the program Origin (with function G92) 458
5.21.4 Notes on programming the ISO Functions 459

Page deliberately empty

5 ISO Programming

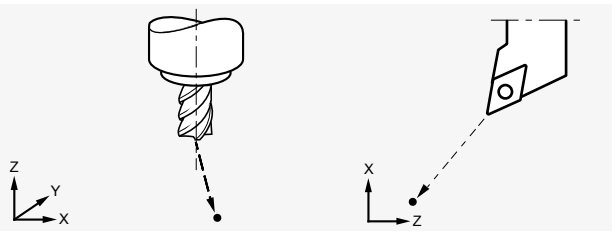
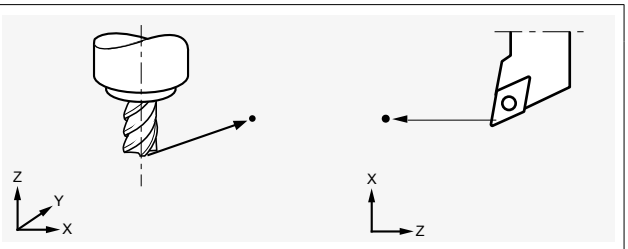
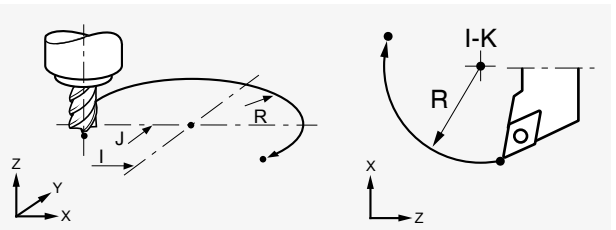
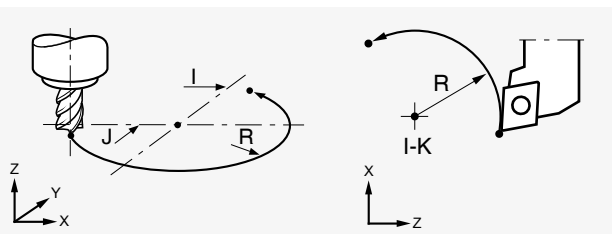
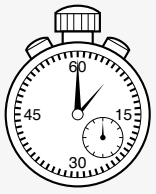
5.1 Functions and programming addresses

5.1.1 G Functions - Main summary table

The functions initialized at power on are identified by " * ".
All G codes, except G24 (inclined plane) are reinitialised at their power on status by a reset.

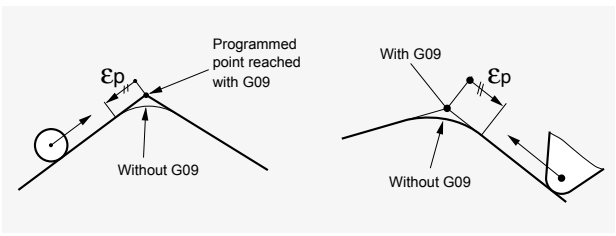
Click on **Gxx** to give the detailed information.

The *"Syntax"* indications in the table are given for Milling application only.
Please refer to the detailed Gxx information to get the *"Syntax"* on Turning application.

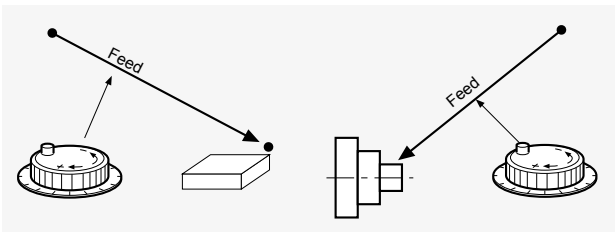
G00	**Linear interpolation at high speed** **Syntax:** N.. [G90/G91] **G00** [R±] X.. Y.. Z.. **Cancellation:** G01/G02/G03	
G01*	**Linear interpolation at programmed feed rate** **Syntax:** N.. [G90/G91] **G01** [R±] X.. Y.. Z.. [F..] **Cancellation:** G00/G02/G03	
G02	**Clockwise circular interpolation at programmed feed rate** **Syntax (XY plane):** N.. [G17] [G90/G91] **G02** X.. Y.. I..J.. or R.. [F..] **Cancellation:** G00/G01/G03	
G03	**Counter-clockwise circular interpolation at programmed feed rate** **Syntax (XY plane):** N.. [G17] [G90/G91] **G03** X.. Y.. I..J.. or R.. [F..] **Cancellation:** G00/G01/G02	
G04	**G04: Programmable dwell** **Syntax:** N.. **G04** F.. **Cancellation:** End of block	
G05	**Programming a movement on the inclined axis** **Syntax:** N.. [G90/G91] [G00/G01] **G05** X.. **Cancellation:** End of block	

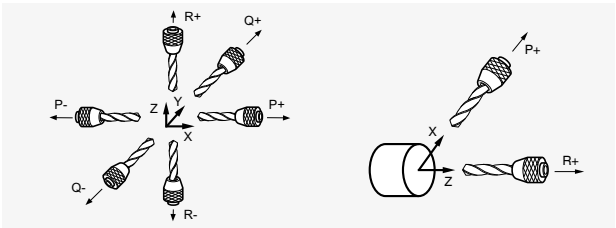
G06	Spline curve execution command
	Syntax: N.. **G06** NC..
	Cancellation End of block

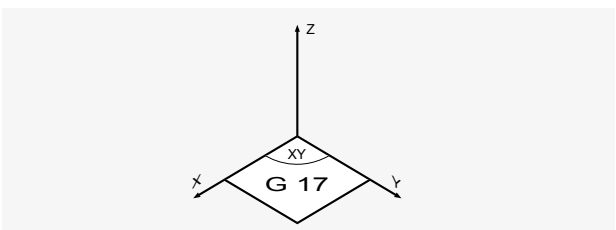
G07	Initial tool positioning before machining along an inclined axis
	Syntax: N.. [G90] [G00/G01] **G07** X.. Z..
	Cancellation End of block

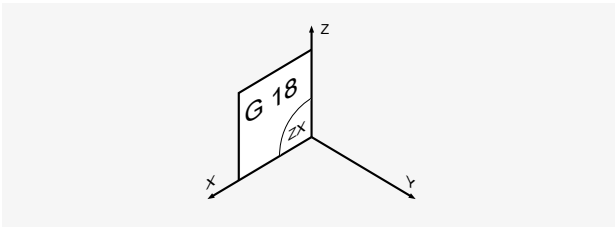
G09	Deceleration at end of block before continuation on next block	
	Syntax: N.. **G09** [G00/G01/G02/G03] X.. Y.. Z.. [F..]	
	Cancellation End of block	

G10	Interruptible block / Probing
	Syntax: N.. [G40] [G04] [G00/G01/G02/G03] X.. Y.. Z.. **G10** [:n] [+X.. or F..] [@n < > Value] N.. [+ Number] [EF..]
	Cancellation: End of block

G12	Overspeed by handwheel	
	Syntax: N.. [G01/G02/G03] **G12** X.. Y.. Z.. [F..] [$0 ...]	
	Cancellation: End of block	

G16*	Definition of the tool axis orientation with addresses P, Q, R	
	Syntax: N.. **G16** P±/Q±/R±	
	Cancellation: G16 P±/Q±/R±.	

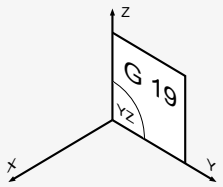
G17*	XY plane selection	
	Syntax: N.. **G17**	
	Cancellation: G18/G19	

G18	ZX plane selection	
	Syntax: N.. **G18**	
	Cancellation: G17/G19	

G19 — YZ plane selection

Syntax:
N.. **G19**

Cancellation:
G17/G18



G20* — Programming in polar coordinates (X, Z, C)

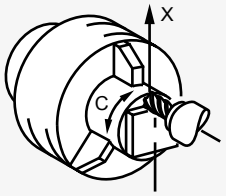
Syntax:
N.. [G40] **G20** [G00/G01] [X.. Z.. C..] [F..]

Cancellation:
G21 and G22

G21 — Programming in Cartesian coordinates (X, Y, Z)

Syntax:
N.. [G40] **G21** [G00/G01] [G41/G42] [X.. Y.. Z..] [G94 F..]

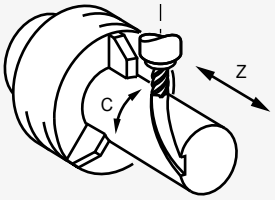
Cancellation:
G20 and G22



G22 — Programming in cylindrical coordinates (X, Y, Z)

Syntax:
N.. [G40] **G22** [G00/G01] [G41/G42] [X.. Y.. Z..] [G94 F..]

Cancellation:
G20 and G21



G23 — Circular interpolation defined by three points

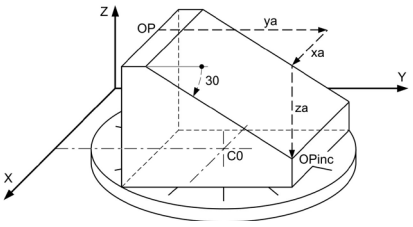
Syntax: (XY plane)
N.. [G17] [G90/G91] **G23** X.. Y.. I.. J.. [F..]

Cancellation:
End of block

G24 — Inclined plane programming

Syntax:
N.. **G24+** X.. Y.. Z.. P(Kxx)/(Kyx)/(Kzx) Q(Kxy)/(Kyy)/(Kzy) R(Kxz)/(Kyz)/(Kzz)

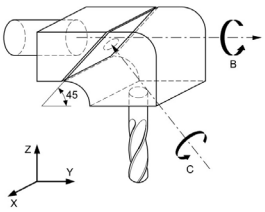
Cancellation: G24-
This function is never programmed directly but embedded in a macro that will calculate the required parameters - Please refer to the Flexium⁺ Extended Programming Manual M00055.



G26 — RTCP Activation

Syntax:
N.. **G26+** I[xl] J[yl] K[zl] TCXc/Yc/Zc TIca/sa TBXb/Yb/Zb

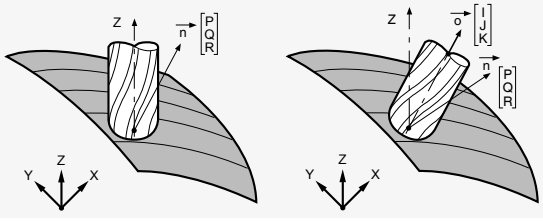
Cancellation: G26-
This function is never programmed directly but embedded in a macro that will calculate the required parameters - Please refer to the Flexium⁺ Extended Programming Manual M00055.

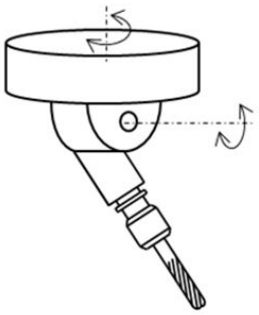
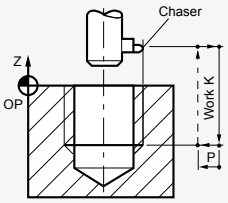
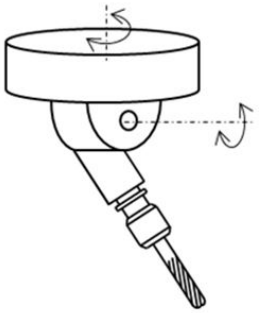
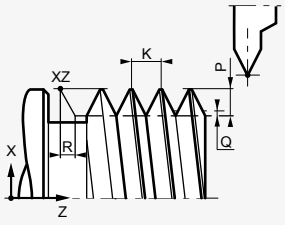
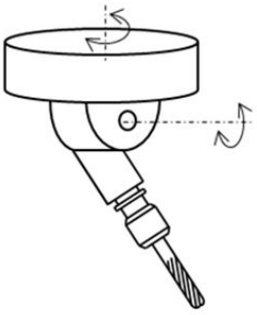


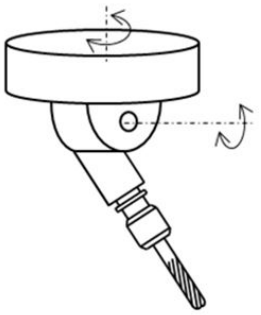
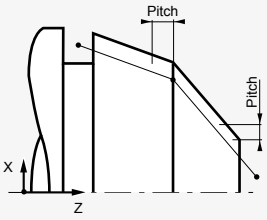
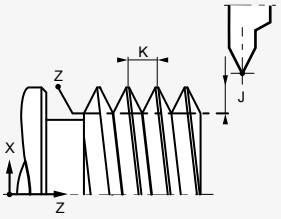
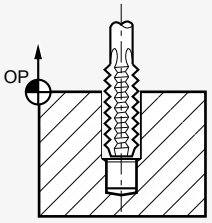
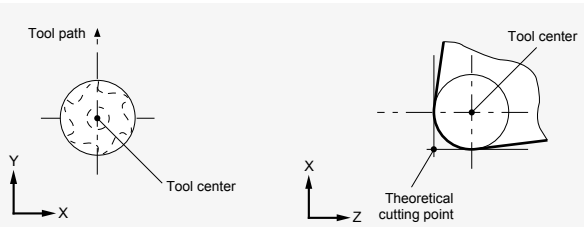
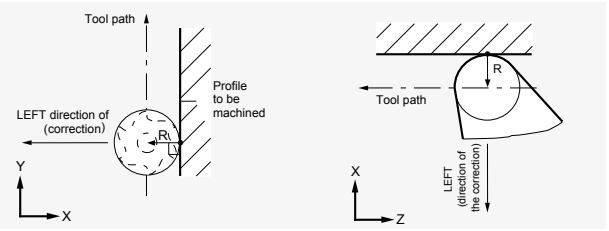
G29 — Tridimensional tool correction with 3 or 5 axes

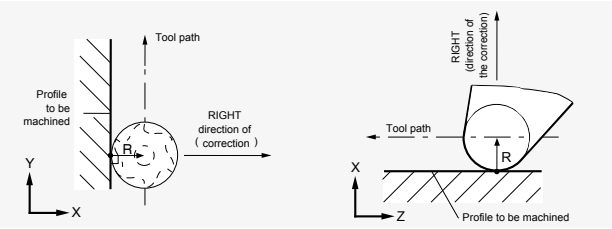
Syntax:
N.. [D..] [G01] **G29** X.. Y.. Z.. P.. Q.. R.. [I.. J.. K..] [A.. /B.. /C..]

Cancellation:
G40/G41/G42



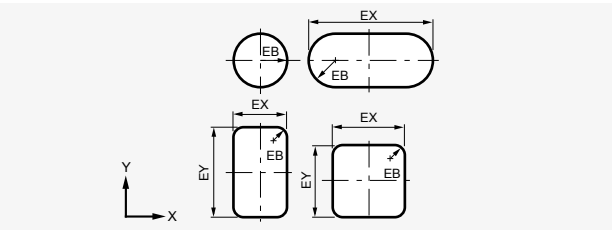
G30*	**RTCP using articular coordinates of rotary axes (default)** **Syntax**: G30 **Cancellation:** Cancellation G32 or G34 or G34 - Standard programming in RTCP the rotary axes are programmed by their dimension -3D workpiece compensation is not available For additional information please refer to the HSC and five axes manual M00052.	
G31	**Thread chasing cycle** **Syntax** (XY plane): N.. [G17] [M03/M04] [S..] **G31** [X.. Y..] Z.. [ER..] [EH..] K.. P.. [F..] [EF..] [EC..] **Cancellation:** G80 to G89	
G32	**RTCP using tool vector coordinates** **Syntax**: G32 **Cancellation:** Cancellation G30, G34 or G34 - RTCP programming via tool vector data For additional information please refer to the HSC and five axes manual M00052.	
G33	**Constant lead thread cutting** **Syntax:** N.. **G33** X.. Z.. K.. [EA..] [EB..] P.. [Q..] [R..] [F..] [S..] **Cancellation:** End of block	
G34 -	**RTCP via articular coordinates and part Balancing without inclined plane** **Syntax:** G34 - **Cancellation:** Cancellation G30 or G32 or G34 RTCP programming via articular coordinates of the rotary axes. The NCK recalculates the tool vector in order to compensate for the part misalignment For additional information please refer to the HCS and five axes manual M00052	

G34	**RTCP via articular coordinates and part balancing, including inclined plane** **Syntax**: G34 **Cancellation:** Cancellation G30, G32 or G34 RTCP programming via articular coordinates of the rotary axes. The NCK recalculates the tool vector in order to compensate for the part misalignment and an inclined plane. For additional information please refer to the HSC and five axes manual M00052.	
G38	**Sequenced thread cutting** **Syntax:** N.. **G38** X.. Z.. K.. **Cancellation:** G00/G01/G02/G03	
G38+	**Resume Thread Cutting** **Syntax:** **G38+** J<delta_posX> EC<posB_appris> K<pas> Z<Xfin> **Cancellation:** End of block	
G39+	**Rigid Tap Removal** **Syntax:** **G39+** M3/4 S... K... Z... / **Cancellation:** End of block	
G40*	**Radius offset cancel** **Syntax:** N.. [G00/G01] **G40** X.. Y.. Z.. **Cancellation:** G29/G41/G42	
G41	**Left radius offset** **Syntax (XY plane):** N.. [G17] [D..] [G00/G01/G02/G03] **G41** X.. Y.. **Cancellation:** G29/G40/G42	

G42	**Right radius offset** **Syntax (XY plane):** N.. [G17] [D..] [G00/G01/G02/G03] **G42** X.. Y.. **Cancellation:** G29/G40/G41	

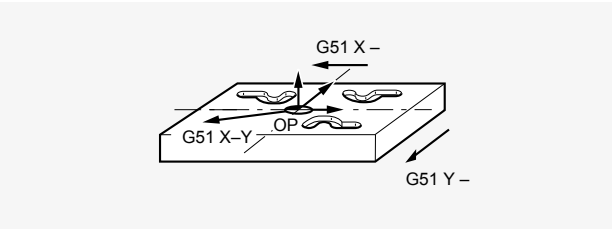
G43	**Tridimensional tool correction with cylindrical tool** **Syntax:** N.. [D..] [G01] **G43** X.. Y.. Z.. P.. Q.. R..[I.. J.. K..] [A.. / B.. / C..] **Cancellation:** G40

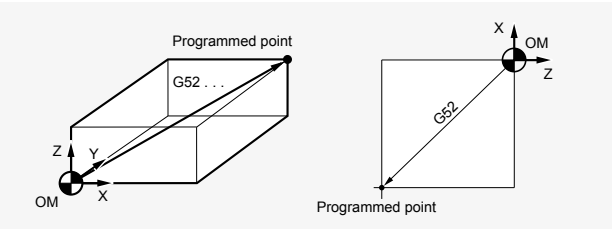
G43 +/-	**Tridimensional tool correction with cylindrical tool in RTCP** **Syntax:** N.. [D..] [G01] **G43** [P.. Q.. R..] **Cancellation:** G40

G45	**Simple pocket cycle** **Syntax:** N.. **G45** X.. Y.. Z.. [ER..] EX.. EY.. [EB..] P.. Q.. [I..] [J..] [EG2/EG3] EP.. EQ.. EI.. EJ.. **Cancellation:** End of block	

G48	**Spline curve interpolation** **Syntax:** N.. **G48** NC.. H.. /N.. N.. **Cancellation:** End of block

G49	**Spline curve deletion** **Syntax:** N.. **G49** NC.. **Cancellation:** End of block

G51	**Mirror function** **Syntax:** N.. **G51** X- Y- Z- **Cancellation:** G51 X- Y- Z- cancelled by G51 X+ Y+ Z+	

G52	**Programming movements in absolute dimensions with reference to the measurement origin** **Syntax:** N.. [G40] [G90] [G00/G01] **G52** X.. Y.. Z.. A.. B.. C.. [F..] **Cancellation:** End of block	

G53	**Cancellation of shifts DAT1 and DAT2** **Syntax:** N.. **G53** **Cancellation:** G54, G55, G56, G57and G58

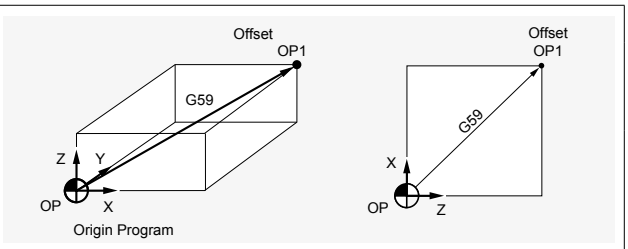
G54	*Enabling of first set of DAT1 origin shift
	Syntax: N.. **G54**
	Cancellation: G53, G55, G56, G57and G58

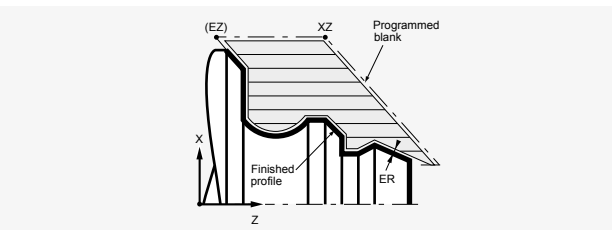
G55	Enabling of second set of DAT1 origin shift
	Syntax: N.. **G55**
	Cancellation: G53, G54, G56, G57, G58

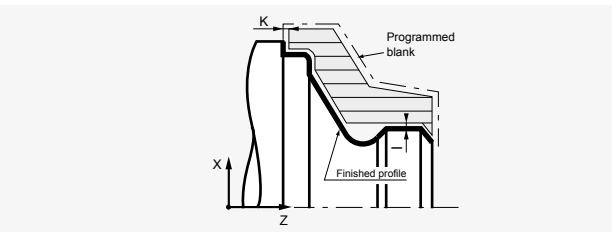
G56	Enabling of third set of DAT1 origin shift
	Syntax: N.. **G56**
	Cancellation: G53, G54, G55, G57, G58

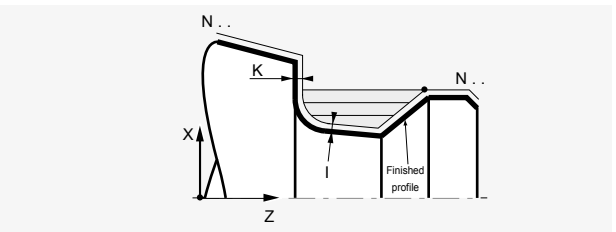
G57	Enabling of fourth set of DAT1 origin shift
	Syntax: N.. **G57**
	Cancellation: G53, G54, G55, G56, G58

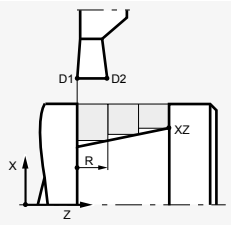
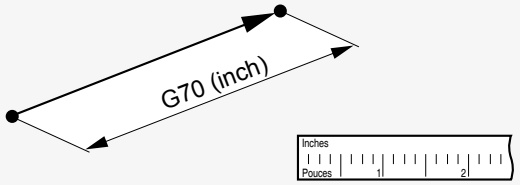
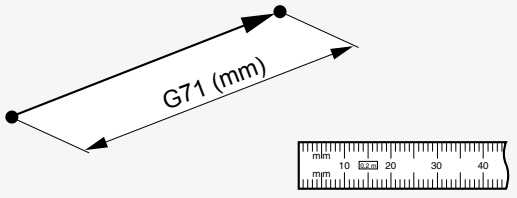
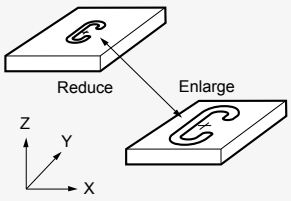
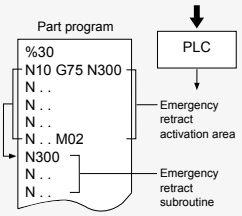
G58	Enabling of fifth set of DAT1 origin shift
	Syntax: N.. **G58**
	Cancellation: G53, G54, G55, G56, G57

G59	Program origin offset	
	Syntax: N.. [G90/G91] **G59** X.. Y.. Z.. U.. V.. W.. A.. B.. C.. [I.. J.. K..] [ED]	
	Cancellation: Cancelled by different G59 X.. Y.. Z..	

G63	Roughing cycle with groove	
	Syntax: N.. **G63** N.. N.. X.. Z.. EX.. / EZ.. P.. / R.. EA.. / EU.. EW.. [EB..] [EC..] [ER..] [Q..] [EQ..] [EF..]	
	Cancellation: End of block	

G64	Turn/Face roughing cycle	
	Syntax: N.. **G64** N.. N.. [I.. K..] P.. / R.. N.. BLANK DEFINITION	
	Cancellation: G80	

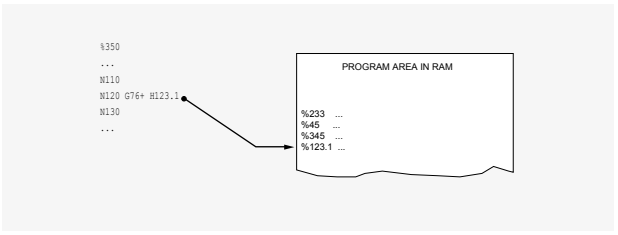
G65	Groove roughing cycle	
	Syntax: N.. **G65** N.. N.. X.. / Z.. [I.. K..] [EA..] P.. / R.. [Q..] [EF..]	
	Cancellation: End of block	

G66	**Plunging cycle** **Syntax:** N.. **G66** D.. X.. Z.. [EA..] P.. / R.. [EF..] **Cancellation:** End of block	
G70	**Inch data input** **Syntax:** N.. **G70** **Cancellation:** G71 according to the machine set-up G70 can be initialised by default instead of G71	
G71*	**Metric data input** **Syntax:** N.. **G71** **Cancellation:** G70	
G73*	**Scaling factor cancel** **Syntax:** N.. [G40] **G73** **Cancellation:** G74	
G74	**Scaling factor enable** **Syntax:** N.. [G40] **G74** **Cancellation:** G73	
G75	**Emergency retraction subroutine declaration** **Syntax:** N.. **G75** N.. **Cancellation:** Cancelled by G75 N0 or different G75 N..	
G76	**Transfer of the current values of L and E parameters into the part program** **Syntax:** N.. **G76** [H..] [N.. N..] **Cancellation :** End of block	
G76+/-	**ISO program or block creation/deletion** The syntax specific to each function is described below **Cancellation :** End of block	

G76+ — Program creation

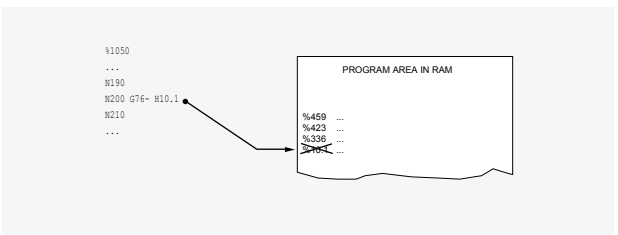
Syntax:
N.. **G76+** H..

Note.
G76+ H<bin> is used to create binary files.
Detailed information can be found in the Flexium⁺ Extended Programming manual M00055.



G76- — Program deletion

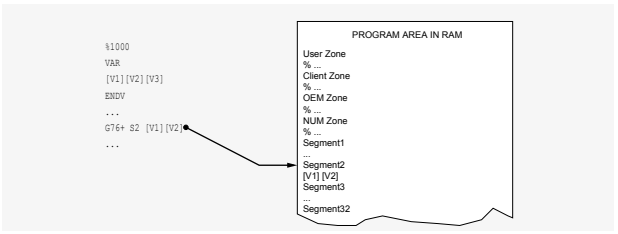
Syntax:
N.. **G76-** H..



G76+ S — Memory reservation and storage

Syntax:
Syntax **G76+ S**.. [list of symbols]
is used to reserve a segment of memory and store symbolic variables inside.
Detailed information can be found in the Flexium⁺ Extended Programming manual M00055.

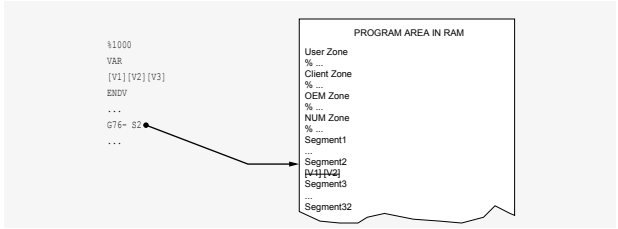
Cancellation :
End of block



G76- S — Cancel memory reservation

Syntax
G79- S..
This function is used to release a segment of memory previously reserved by G76+ .

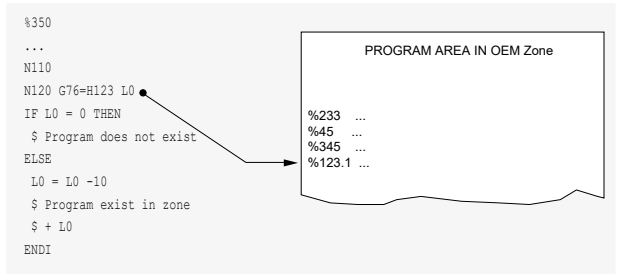
Cancellation :
End of block



G76=H — Checking existence of a file

Syntax
G76= H... L...| E...
This function checks if a file exists and returns the answer in the L or E parameter..

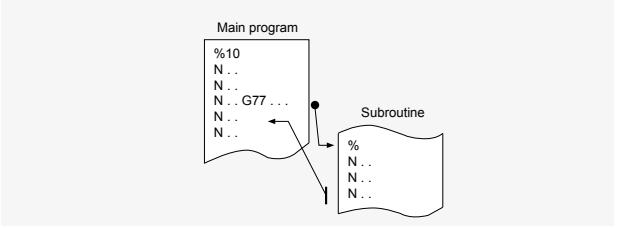
Cancellation :
End of block



G77 — Unconditional branch to a subroutine or sequence of blocks with return

Syntax:
N.. **G77** [H..] [N.. N..] [S..]

Cancellation :
End of block

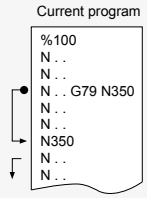


G77 -i — Call of the subroutine return block

Syntax:
N.. **G77 -i**

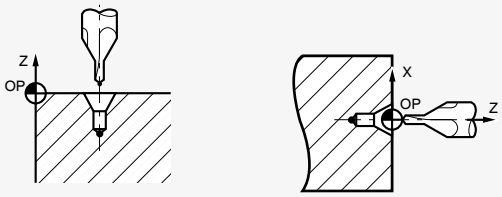
Cancellation :
End of block

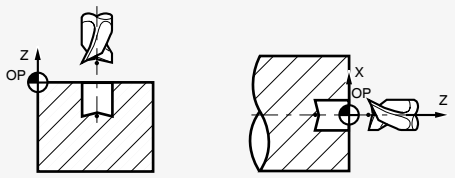
G78	**Channel synchronization with markers** **Syntax:** N.. **G78** Q.. Pj.i Pk.l .... **Cancellation :** Other G78 End of program	Please refer to chapter 5.26.3

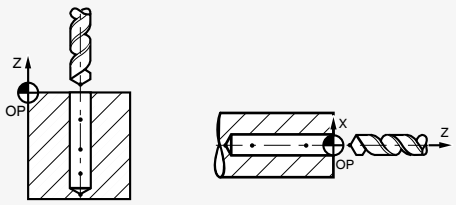
G79	**Conditional or unconditional jump to a sequence without return** **Syntax:** N.. **G79** [L../E.. > = < Number] N..	
	Cancellation : End of block	

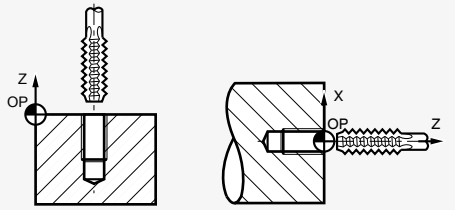
G79 +/-	**Temporary suspension of next block preparation in a sequence with movements**
	Syntax (XY plane) N.. [G00/G01/G02/G03] X.. Y.. Z.. **G79 +/-** X.. /F..
	Cancellation : End of block

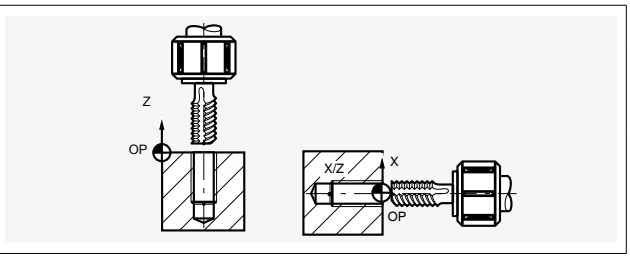
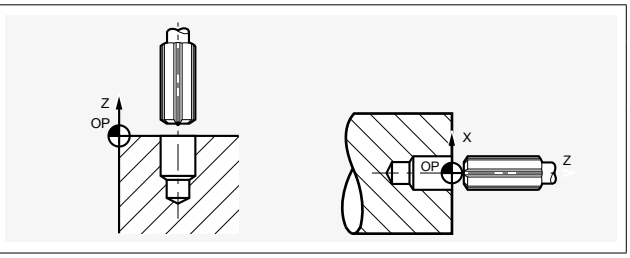
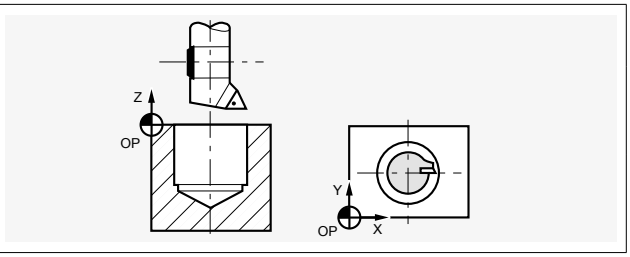
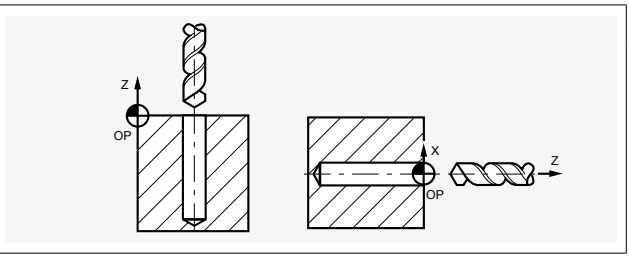
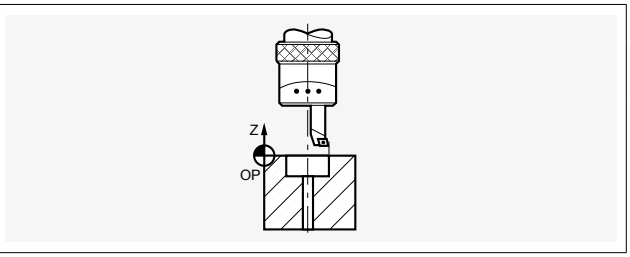
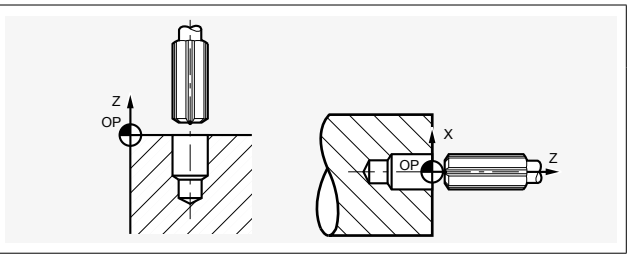
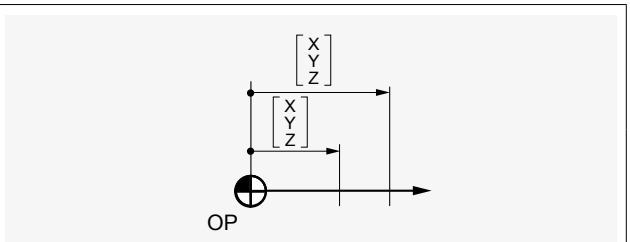
G80*	**Machining cycle cancel**
	Syntax: N**.. G80**
	Cancellation: G31/G81-G89

G81	**Drilling/center drilling cycle** **Syntax (XY plane):** N.. [G17] **G81** [X.. Y..] Z.. [ER..] [EH..] [F..] **Cancellation:** G31/G80/G82-G89	

G82	**Counterboring cycle** **Syntax (XY plane):** N.. [G17] **G82** [X.. Y..] Z.. [ER..] [EH..] EF.. [F..] **Cancellation:** G31/G80/G81/G83-G89	

G83	**Peck drilling cycle** **Syntax (XY plane):** N.. [G17] **G83** [X.. Y..] Z.. [ER..] [EH..] [P..]/[ES..] [Q..] [EP..] [F..] [EF..] **Cancellation:** G31/G80-G82, G84-G89	

G84	**Tapping cycle** **Syntax (XY plane):** N.. [G17] **G84** [X.. Y..] Z.. [ER..] [EH..] EF.. [F..] **Cancellation:** G31/G80-G83, G85-G89	

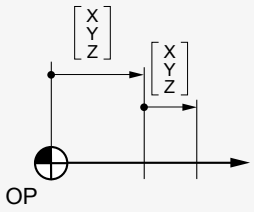
G84	**Rigid tapping cycle** **Syntax (XY Plane):** N.. [G17] [M03/M04] [M04-M05] **G84** [X.. Y..] Z.. [ER..] [EH..] K.. [EK..] **Cancellation** G31/G80-G83, G85-G89	
G85	**Reaming cycle** **Syntax (XY plane):** N.. [G17] **G85** [X.. Y..] Z.. [ER..] [EH..] [F..] [EF..] **Cancellation:** G31/G80/G81-G84, G86-G89	
G86	**Boring cycle with indexed stop and clearance at hole bottom** **Syntax (XY plane):** N.. [G17] **G86** [X.. Y..] Z.. [ER..] [EH..] [EC..] [EA..] [EP..] [F..] **Cancellation:** G31/G80-G85, G87-G89	
G87	**Drilling cycle with chip breaking** **Syntax (XY plane):** N.. [G17] **G87** [X.. Y..] Z.. [ER..] [EH..] [P..]/[ES..] [Q..] [EP..] [EF..] [F..] **Cancellation:** G31/G80-G86, G88/G89	
G88	**Boring and facing cycle** **Syntax (XY plane):** N.. [G17] **G88** [X.. Y..] Z.. [ER..] [EH..] [F..] **Cancellation:** G31/G80-G87/G89	
G89	**Boring cycle with Dwell at the Bottom of the hole** **Syntax (XY plane):** N.. [G17] **G89** [X.. Y..] Z.. [ER..] [EH..] [EF..] [F..] **Cancellation:** G31/G80-G88	
G90*	**Absolute dimensions with respect to the program origin** **Syntax:** N.. **G90** X.. Y.. Z.. A.. B.. C.. **Cancellation:** G91	

G91

Incremental dimensions with respect to the start of the block

Syntax:
N.. **G91** X.. Y.. Z.. A.. B.. C..

Cancellation:
G90

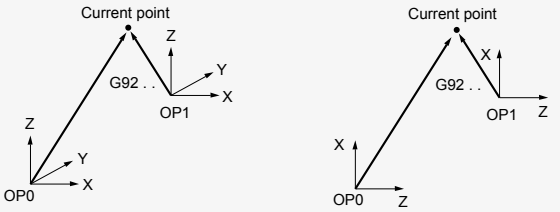


G92

Program origin preset

Syntax:
N.. **G92** X.. Y.. Z..

Cancellation:
End of block

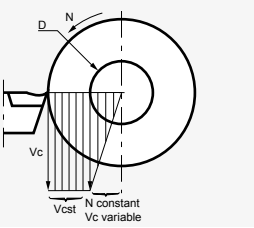


G92 S..

Spindle Speed Limiting

N.. **G92** S..

Cancellation:
Cancelled by G92 S0 or G92 with different S.

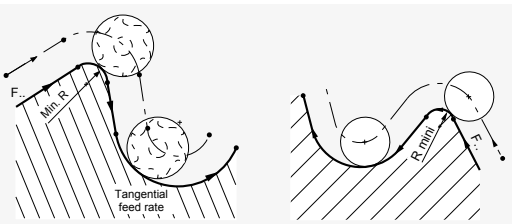


G92 R..
G92 L..

Programming the tangential feed rate

Syntax:
N.. **G92** R.. L..

Cancellation:
Cancelled by G92 R0 or by different G92 R

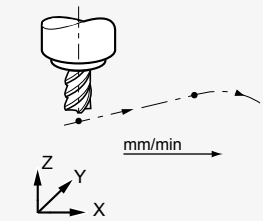


G92 F..

Setting feed rate per axis

Syntax:
G92F... <list of axes> or
G92F... P (MLEGB only)
G92F... Q (MLEGB only)

Cancellation:
G92 F... or M02

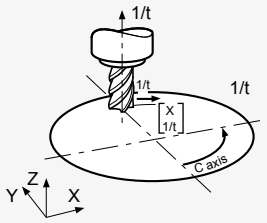


G93

Feed rate in inverted of time (1/t)

Syntax:
N.. **G93** F.. G01 X.. Y.. Z.. A.. B.. C..

Cancellation:
G94/G95

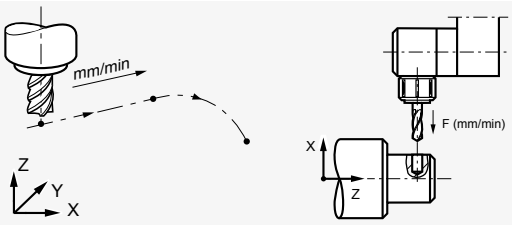


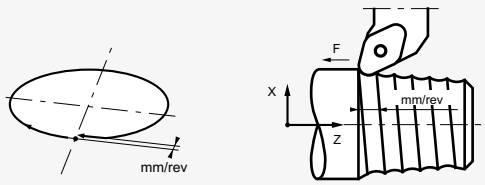
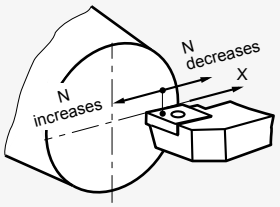
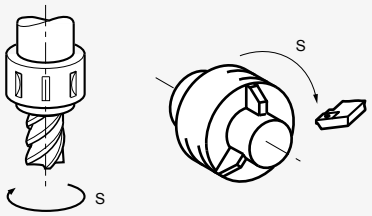
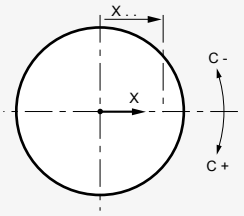
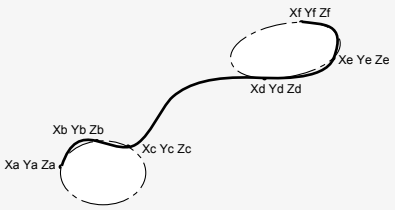
G94*

Feed rate expressed in millimetres, inches or degrees per minute

Syntax:
N.. **G94** F.. G01/G02/G03 X.. Y.. Z.. A.. B.. C..

Cancellation:
G93/G95



G95	**Feed rate expressed in millimetres or inches per revolution** **Syntax:** N.. **G95 F**.. G01/G02/G03 X.. Y.. Z.. **Cancellation:** G93/G94	
G96	**Constant surface speed is expressed in meters per minute** **Syntax:** N.. **G96** [X..] S.. **Cancellation:** G97	
G97*	**Spindle speed Control (RPM)** **Syntax:** N.. **G97 S**.. [M03/M04] **Cancellation:** G96 (combined machine)	
G98	**Definition of the Start X for Interpolation on the C Axis** **Syntax:** N.. **G98** X.. **Cancellation:** End of block	
G104	**Tridimensional curve smoothing** **General syntax:** N.. X. . Y.. Z.. (first point on the curve) N.. [G01] **G104** X..Y.. Z.. [F..] [Intermediate points on the curve] N..G80 X.. Y.. Z.. (last point on the curve) **Cancellation:** G80	
G143 **G146 to G148** **G149** **G158**	**Functions reserved for measuring cycles** These functions are related to measurement cycles. Please refer to Work piece Measuring Cycles M00046 for additional information G143 : Fixed probe calibration G146 to G148 : Measurement cycles for milling G149 : Spindle probe calibration G158 : Diverses Work piece measurement cycles	
G164	**Corner rounding function** **Syntax:** **G164** [Hfile] Nbegin Nend ER.. [EQ..]* [ES..]* [ET..]* **Properties of the function** G164 is not modal (cancelled at the last block of the trajectory defined). This function is intended to smooth the corners via a spline curve instead of relying to the lag value. By defining a smoother trajectory it generally permits to pass the corners at a higher speed.	

G199	Programmed homing
	Syntax: **G199 F... L1 =** ... <list of axes to home>
	Cancellation: End of block L1 represents the maximum retract distance in case the axis is already on the home switch. Additional information is available in the Flexium+ Commissioning manual M00044. Remark This function is parametrised by the machine builder. Refer to the machine user's documentation.

G246	Volumetric errors compensation
	This function is intended to compensate the kinematics errors of a machine by building an error model For detailed information, please refer to VEComp Instruction manual M00045

G247	Volumetric errors compensation for rotary axes
	This function is intended to compensate the kinematics errors of the rotary axes of a machine by building an error model. For detailed information, please refer to VEComp Instruction manual M00045

G248	3D probing of a sphere for RTCP calibration
	This function is intended to estimate and compensate the positioning errors due to an incorrect RTCP setting For detailed information, please refer to HSM and Five axes manual M00052.

G252 to G255	Reserved functions for machine setup
	These functions are reserved for used with the Flexium Tools instruments (e.g. Ballbar). They should not be used directly nor edited or replaced

G501 to G599	Enabling of second set of DAT2 origin shift.. G5xx
	Syntax: N.. **G5xx**
	Cancellation: Other G5xx, G53 **Remark** After cancellation by G53, the programming of G54..G58 will not re-validate the last G5xx programmed. The machine builder can limit the range to a value lower than G599 - The default is G550.

G725	Trochoidal milling
	Trochoidal milling is a method of machining used to create a slot wider than the cutting tool diameter. This is accomplished using a series of circular cuts known as a trochoidal tool path. This cycle is used to machine a path with a trochoidal motion.

G730	Engraving
	This cycle is used to engrave characters. Characters available are the uppercase and lowercase letters (A-Z and a-z) and some special characters (\!$&/()=?'[]+*@#_-:.;,<>).

G732	Surface Quality Performance
	This function is intended to optimise the servo parameters with respect to the machining phase (Rough, Semi-finish, Finish,…) to optimise the machining quality and time. It can be adapted by the machine builder and is further explained in the Five axes and HSM manual (M00052)

G748	Tangential control
	This function is intended to automatically control an axis in order to maintain a tool at a certain angle with respect to the trajectory.

G749	Cancel Tangential control
	This function cancels the tangential control activated by G748

G750	Stops storing axes coordinates in a file
	This function is reserved for setup purposes. Please refer to Flexium⁺ Extended Programming manual M00055.

G751	Start storing axes coordinates in a file
	This function is reserved for setup purposes. Please refer to Flexium⁺ Extended Programming manual M00055.

G770	MLEGB Spindle continuous rotation
	Syntax: N.. **G770**.. M03/M04 S
	Cancellation: G770 M05 This function is specific for Multi Level Electronic gear box (MLEGB). Please refer to Flexium⁺ Extended Programming manual M00055.

G771	MLEGB Synchro Definition / Cancellation
	Syntax: N.. **G771**.. + C A P37 Q41
	Cancellation: G771- This function is specific for Multi Level Electronic gear box (MLEGB). Please refer to Flexium⁺ Extended Programming manual M00055.

G772	Modification of the synchronisation
	Syntax: G772 +: Reactivates the synchronisation G772 = modify the synchronisation ration on the fly
	Cancellation: G772 - This function is specific for Multi Level Electronic gear box (MLEGB). Please refer to Flexium+ Extended Programming manual M00055.

G775	MLEGB Activation / Deactivation
	Syntax: **G775+**
	Cancellation: G775- This function is specific for Multi Level Electronic gear box (MLEGB). Please refer to Flexium⁺ Extended Programming manual M00055.

G776	MLEGB Waiting for synchronization completion before proceeding
	Syntax: **G770**.. C F 4
	Cancellation: Automatic This function is specific for Multi Level Electronic gear box (MLEGB). Please refer to Flexium+ Extended Programming manual M00055.

G777	Early Block Change
	This function is intended to start the execution of a programmed block.even if the previous has not been fully completed. It included a quick dialog with the PLC thanks to 32+32 dedicated bits. The different cases of Early Block Change (EBC) are : • Immediately, the two blocks are therefore executed simultaneously. • Once a certain distance has been travelled. • When the residual distance is lower than a particular value. • When the PLC sets a certain signal. • When the block is normally ended (standard way) This function is further explained in the Extended Programming Manual (M00055)

G779	MLEGB Synchro Homing / Gear alignment
	Syntax: **G779**
	Cancellation: Automatic This function is specific for Multi Level Electronic gear box (MLEGB). Please refer to Flexium+ Extended Programming manual M00055.

G997	Enabling and execution of all functions stored in state G999
	Syntax: N.. **G997**
	Cancellation: G998/G999

G998	Enabling of execution of the blocks and part of the functions processed in state G999
	Syntax: N.. **G998**
	Cancellation: G997/G999

G999	Suspension of execution and forcing of block concatenation
	Syntax: N.. **G999**
	Cancellation: G997/G998

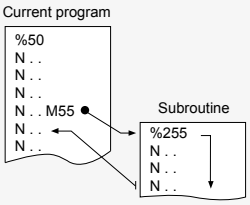
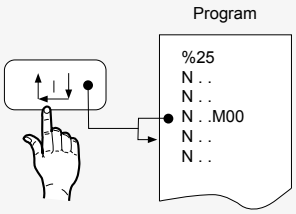
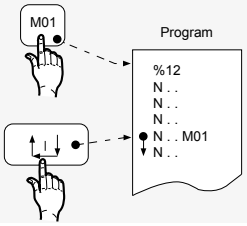
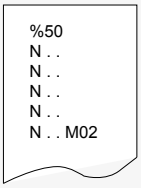
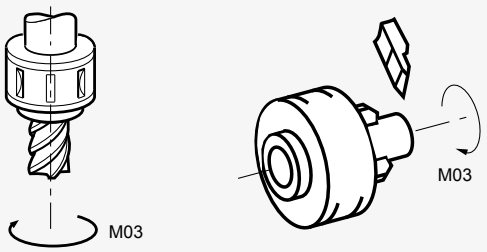
5.1.2 M Functions - Main summary table

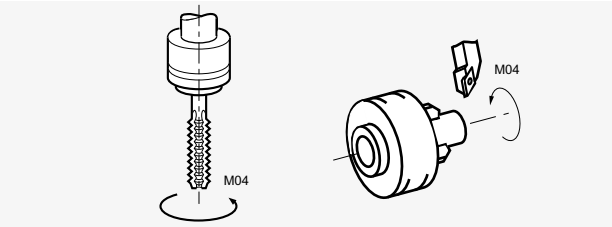
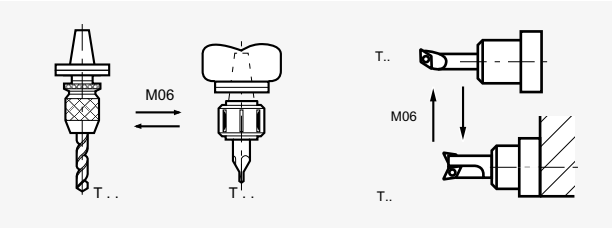
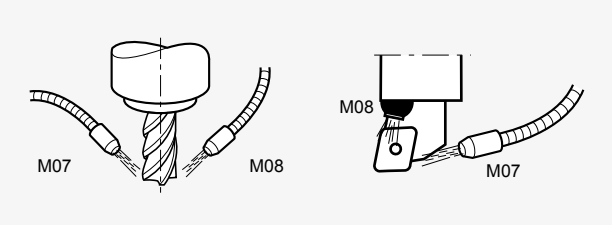
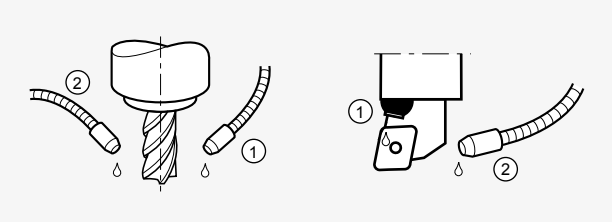
The functions initialized at power on are identified by **" * "**.

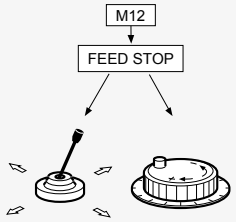
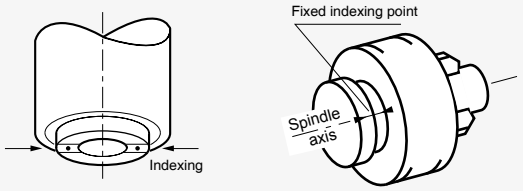
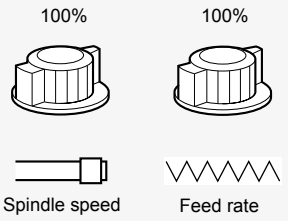
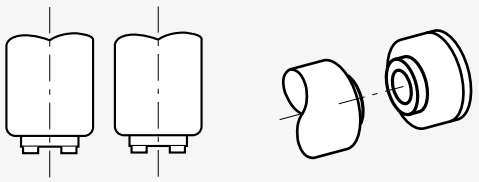
The miscellaneous functions listed in the table are decoded functions. Several decoded M functions can be programmed in the same block, e.g. N.. S100 M03 M40 M08.

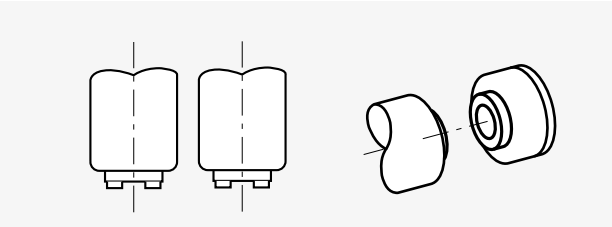
Click on **Mxx** to give the detailed information.

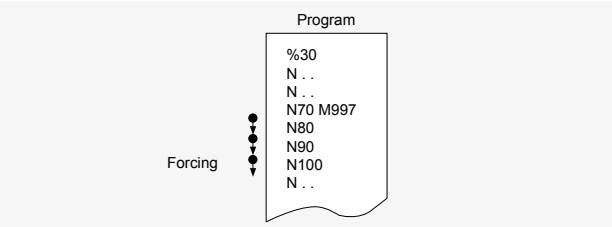
The *"Syntax"* indications in the table is given for Milling application only.
Please refer to the detailed Mxx information to get the *"Syntax"* on Turning application.

M..	**Subroutine Branch by M Function** N.. **M..** Up to 32 M functions can be linked to a particular subroutine each. This list is defined by the machine builder.	
M00	**Program stop** **Syntax**: N.. [G40] **M00** [$0 ...] **Cancellation**: Action on the machine panel CYCLE key	
M01	**Optional stop** **Syntax**: N.. [G40] **M01** [$0 ...] **Cancellation**: Action on the machine panel CYCLE key	
M02	**End of program** **Syntax**: N.. **M02**	
M03	**Spindle clockwise rotation** **Syntax**: N.. **M03** **Cancellation**: M02/M04/M05/M00/M19 M03 as well as the other spindle programming functions can have other parameters according to the spindle to control. ' : ' Refers to the logical number of the spindle. '#' Refers to the rank of the spindle in the channel. Refer to the chapter 5.3 Spindle programming. **Remark.** According to the machine setting this direction can be inverted.	

M04	**Spindle counter-clockwise rotation** **Syntax**: N.. **M04** **Cancellation**: M02/M03/M05/M00/M19. For more detailed information about this function, please refer to the chapter 5.3 Spindle programming. **Remark.** According to the machine setting this direction can be inverted.	
M05*	**Spindle off** **Syntax**: N.. **M05** **Cancellation**: M03/M04/ M00 / M19 For more detailed information about this function, please refer to the chapter 5.3 Spindle programming.	
M06	**Tool change** **Syntax**: N.. T.. [D..] **M06** [$0.. / (...)] **Cancellation**: M function report (CRM).	
M07	**Coolant 2 on** **Syntax**: N.. **M07** **Cancellation**: M09.	
M08	**Coolant 1 on** **Syntax**: N.. **M08** **Cancellation**: M09	See figure above.
M09*	**Coolant off** **Syntax**: N.. **M09** **Cancellation**: M07/M08	
M10	**Clamp** **Syntax**: N.. [G00/G01/ G02/ G03] **M10** X.. Y.. Z.. A.. B..C.. **Cancellation**: M11	
M11	**Un-clamp** **Syntax**: N.. **M11** **Cancellation**: M10	

M12	**Programmed feed stop** **Syntax**: N.. **M12** [$0...] **Cancellation**: Press CYCLE on the machine panel	
M19	**Indexed Spindle Stop** **Syntax**: N.. **M19 [C/EC +- ...]** **Cancellation**: M03/M04/M05. For more detailed information about this function, please refer to the chapter 5.3 Spindle programming.	
M40/ M41/ M42/ M43/ M44/ M45	**Spindle speed ranges** **Syntax**: N.. [S..] [M03/M04] **M40 to M45** **Cancellation**: Cancel one another. For more detailed information about this function, please refer to the chapter 5.3 Spindle programming.	
M48*	**Spindle speed and feed rate potentiometer enable** **Syntax**: N.. **M48** **Cancellation**: M49	
M49	**Spindle speed and feed rate potentiometer enable inhibit** **Syntax**: N.. **M49** **Cancellation**: M48	See figure above.
M61	**Release a spindle** **Syntax**: N.. **M61**: [1..32] or M61 #[0..3] **Cancellation**: M62-M65. Releasing a spindle is necessary to make it available for another channels. For more detailed information about this function, please refer to the chapter 5.3 Spindle programming.	
M62	**Control of a spindle as main spindle** **Syntax**: N.. [S..] **M62**: <1..32> **Cancellation**: Selection of another main spindle. For more detailed information about this function, please refer to the chapter 5.3 Spindle programming.	

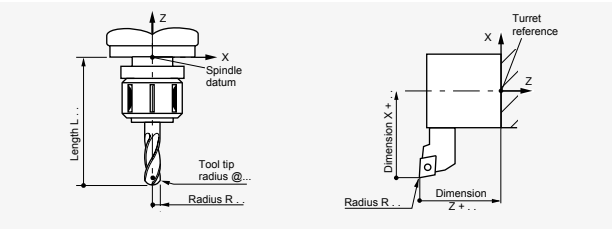
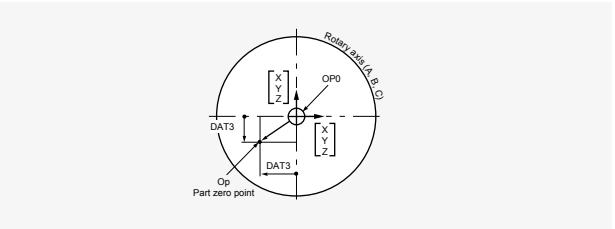
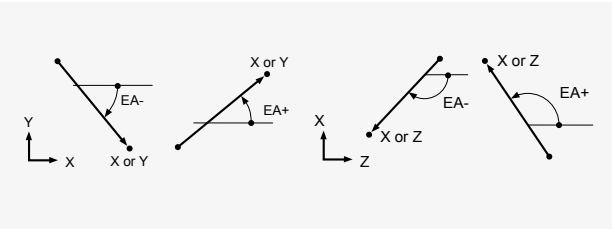
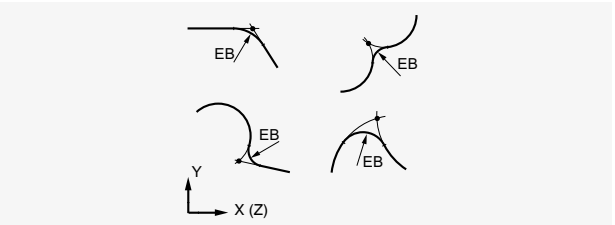
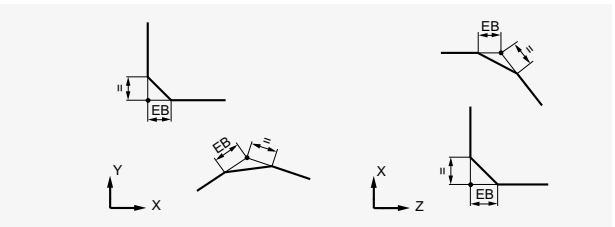
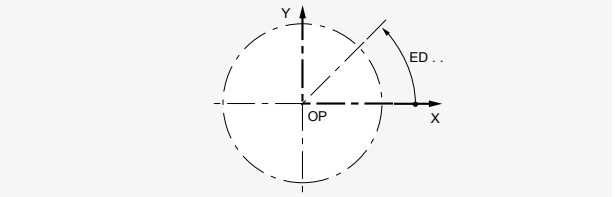
M66	**Definition of the reference spindle (measurement)**	
	Syntax: N.. [S..] **M66**:<1..32> or M66#<0..3>	
	Cancellation: Cancellation M61:<spindle> or selection of another third auxiliary spindle. For more detailed information about this function, please refer to the chapter 5.3 Spindle programming.	

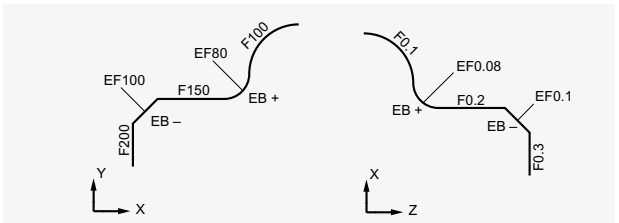
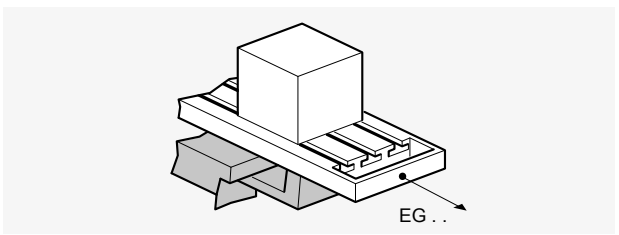
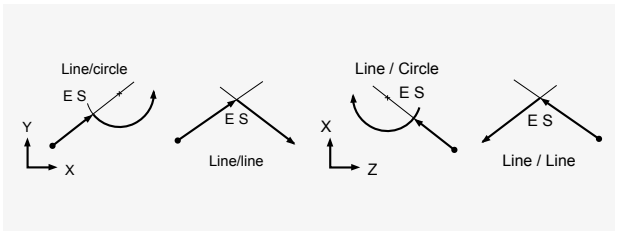
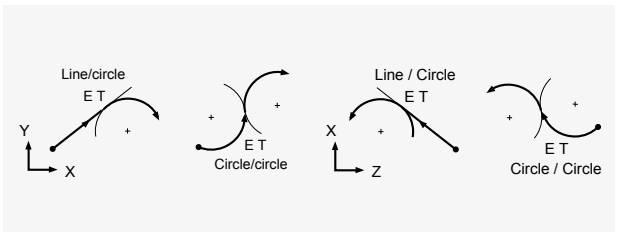
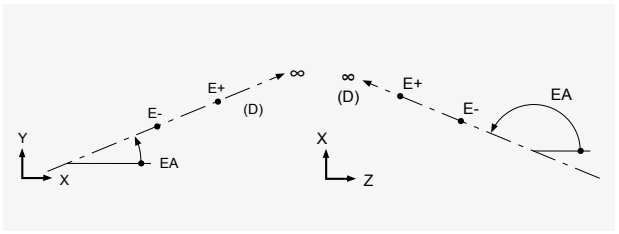
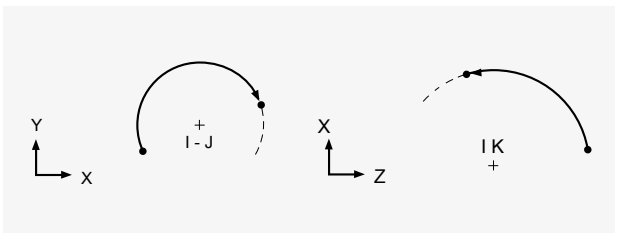
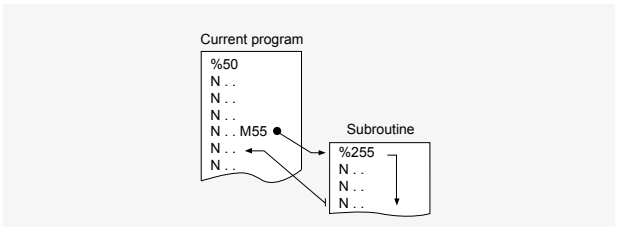
M997	**Forced block consecutive execution**	
	Syntax: N.. **M997**	
	Cancellation: M998/M999/M02.	

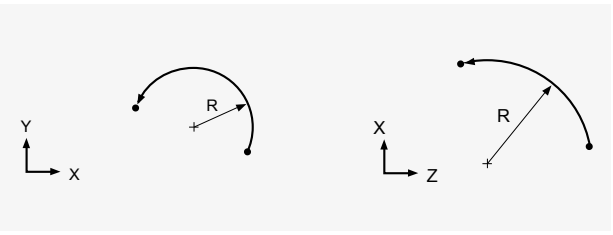
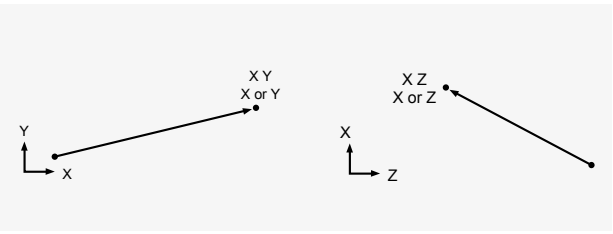
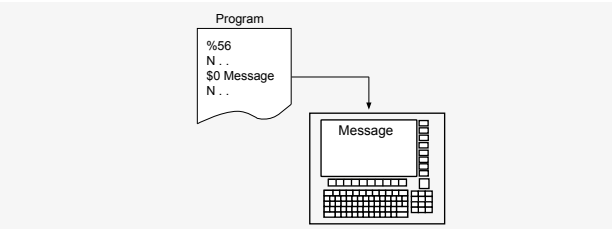
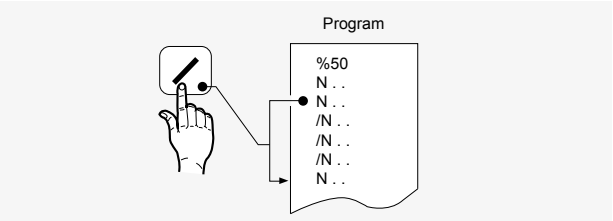
M998*	**Reactivation of MDI mode and subroutine calls by the automatic control function**
	Syntax: N.. **M998**
	Cancellation: M997/M999

M999	**Programmed cancellation of MDI mode and subroutine calls by the automatic control function**
	Syntax: N.. **M999**
	Cancellation: M997/M998/M02

5.1.3 Other Functions - Main summary table

D..	**Tool Offset activation** **Syntax**: N.. [G17] [G16 R+] **D..** [G40/G41/G42] X.. Y.. Z.. **Cancellation**: D0 , M02 or new tool offset. By default up to 255 different tool offsets can be selected. If the tool management option is active the D value represent a particular cutting edge of the current tool. Refer to Tool Management chapter 5.1.1.3.	
DAT3	**Milling Table offset**	
EA..	**Angle element of a line**	
EB+..	**Fillet between two interpolations** **Syntax**: N.. G01/G02/G03 X.. Y.. I.. J.. / R.. [F..] **EB+**.. [EF..] **Cancellation:** End of block	
EB-..	**Chamfer between two linear interpolations** **Syntax**: N.. G01 X.. Y.. [F..] **EB-**.. [EF..] **Cancellation:** End of block	
ED..	**Programmed angular offset** **Syntax**: N.. [G90/G91] **ED..** **Cancellation**: Programming of a new value, M02, reset.	

EF..	**Feed rate specific to fillets EB+ and chamfers EB-.** **Syntax**: N.. Interpolation EB+.. / EB-.. **EF..** **Cancellation**: Programming of a new value, M02, reset.	
EG..	**Programmed acceleration reduction** **Syntax**: N.. **EG..** **Cancellation**: Programming of a new value, M02, reset.	
ES..	**Secant element**	
ET..	**Tangent element**	
E+ / E-	**Discriminant**	
I.. J.. K..	**Coordinates of the center of a circle**	
M..	**Subroutine branch by M function** **Syntax**: N.. **M..**	

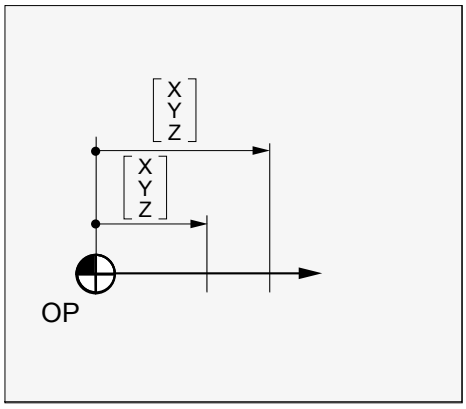
R..	**Radius of a circle**	
X.. Y.., Z..	**Coordinates of the end point of a line**	
$0	**Message Transmission to the display** **Syntax**: N.. **EG..** **Cancellation**: Programming end program, M02, reset.	
/	**Block skip - Multi level block skip** **Syntax**: **/** N.. (Contents unimportant) or /i (skip blocks level " i "), i,j,...=1..32 /i,j (skip blocks level " i, j and... "), /i-j (skip blocks level " i to j ") /i-j, k,... (skip blocks level " i to j and k and ...")	

5.2 Initialisation

5.2.1 Absolute or Incremental selection ► G90, G91

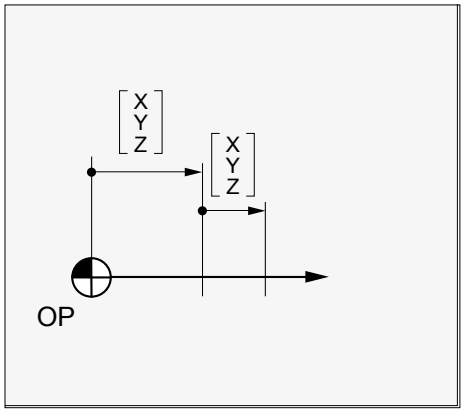
G90	Absolute dimensions with respect to the program origin.

The value programmed on an axis is referenced to the program origin (OP).



G91	Incremental dimensions with respect to the start of the block.

The value programmed on an axis is referenced to the last programmed position. The value is equal to the movement to be performed.



Syntax

N.. **G90/G91** X.. Y.. Z.. A.. B.. C..

G90	Absolute dimensions.
G91	Incremental dimensions.
X.. Y.. Z.. A.. B.. C..	End points.

Properties of the Functions

Functions G90 and G91 are modal.

G90 is the default function.

Cancellation

Functions G90 and G91 cancel one another.

Starting with version 4.1.20 the rules for naming the axes (X,Y, ...,C) have been extended to give the possibility of, for example, several X axes in the same channel with an index to distinguish them. Please refer to the Flexium Extended programming manual M00055 for complement.

Important note

The first movement must be programmed:

- Starting with version 4.1.20 the first movement in a program can be programmed in incremental assuming that the axis position has been previously defined.
- If the conditions are not met then error 72 will be issued.
- Please note that before version 4.1.20 the first move in a program was executed in absolute whatever the selection G90/G91 with no error issued.

Combined programming

Both types of programming (G90/G91) can be included in a program and even in a block.
For instance:

```
N..
N.. G91 X.. Y..
N.. G90 X.. G91 Y..
N.. G90 X.. Y..
N..
```

X absolute, Y incremental

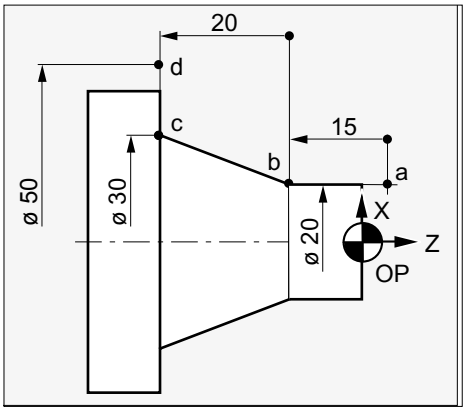
Example



Incremental programming (G91), (System programmed on diameter)

Absolute programming of point a,
incremental movement to points b, c, d.

```
N.. (G90) ...
N.. X20 Z5
N.. G91 Z-15
N.. X5 Z-20
N.. X10
N.. G90
N..
```

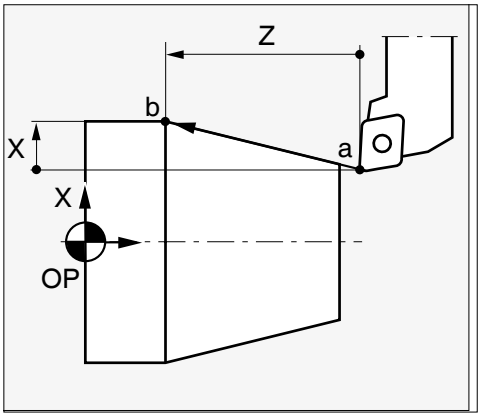




Incremental programming (G91), (System programmed on radius)

Tool located at point a (starting point)
Incremental programming of point b
(amount of movement to reach end point b).

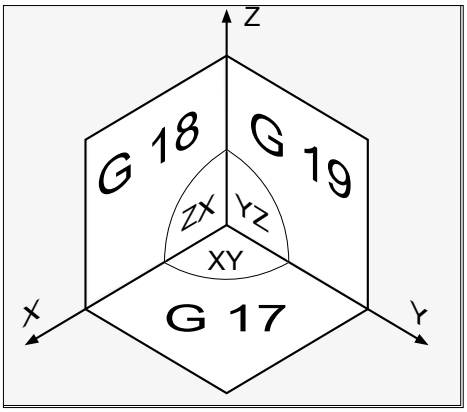
```
N.. (G90) ...
N.. Xa Za
N.. G91 Xb Zb
N..
```



5.2.2 Plane selection ► G17, G18, G19

G17/G18/G19	Plane selection for circular interpolation and radius correction

One of these G functions is programmed to define the radius correction and circular interpolation plane.



Syntax

N.. **G17/G18/G19**

G17	XY plane selection.
G18	ZX plane selection.
G19	YZ plane selection.

Properties of the Functions

Functions G17, G18 and G19 are modal.

G17 is the default function.

Cancellation

Functions G17, G18 and G19 cancel one another.

Notes

A change of plane must be programmed:

- with the system in state G40 (no G41/G42 cutter compensation active) or the system returns error message 138.
- after a sequence completely defined in "PGP" (Profile Geometry Programming, see Chapter 6) or the system returns error message 137.

Example

```
N.. ...
N50 G17...                    XY plane selection
N..
N160 G18 ...                  ZX plane selection
N..
N250 G17 ...                  XY plane selection
N..
```

5.3 Spindles control

Flexium+ is able to handle up to 32 spindles, four in each channel.

For easier programming each spindle receives a unique number from 1 to 32 known as "*Logical Address"*. Each spindle can be identified in the part program via this number, *Spindle1, Spindle2,…*

A machine parameter associates this logical number to a physical drive. Thanks to this mechanism, the user does not have to worry about the physical address of the spindle on his machine.

Each channel can control up to four spindles:

- One main spindle
- Three auxiliary spindles.

The main spindle offers all the possibilities whereas the auxiliary spindles cannot perform the standard canned cycles as well as *Constant Surface Speed*. There is no difference between each of the auxiliary spindles.

The type of spindle in each channel is called the *'Rank'*:

- Main spindle rank is 0
- Auxiliary spindle rank is 1 to 3.

Except some cases described later, the different spindle controlling functions are related to the rank. When no rank (or address) is specified the command relates to the main spindle to ensure compatibility with existing programs.

- The logical addresses are introduced by the character *" : "*
- The rank is introduced by the character *" # "*.

Examples

- **S200#2** : Launch the second auxiliary spindle in M3 (this spindle should belong to the channel)
- **M5** : Stop the main spindle (no suffix implies main spindle).

It is possible to program several spindles in the same block:

- **M40#2 M3#1 S800**

The spindle logical address is defined by the OEM via the machine parameters and cannot be changed by programming.
On the contrary the ranks are predefined but can be changed at any time, thanks to the functions below:

- **M61** : Release a spindle to make it available for the same channel or another. (According to the machine setting a released spindle can keep it speed and direction or must be stopped before being released.
- **M62** : Takes control of a particular spindle either as main (M62) or auxiliary 1, 2 or 3 respectively. The spindle logical address is a mandatory argument.
- **M66** : Used to define the reference spindle. The reference spindle is the spindle whose measure is taken as reference for feed/rev.

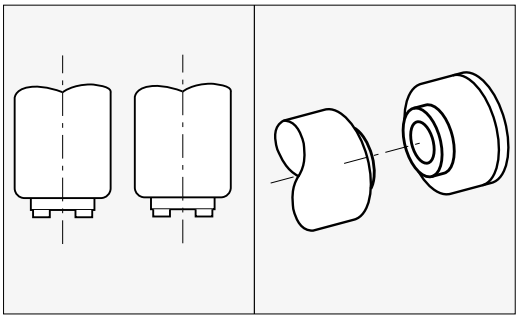
Summary table

Function	M..	M :	M #	M # :
M61	e.g. **M61** *Release main spindle*	e.g. **M61:5** *Release logical 5*	e.g. **M61#2** *Release second auxiliary*	e.g. **M61#1:8** *Error Too much data*
M62	e.g. **M62** *Error*	e.g. **M62:8** *Takes logical 8 as main*	e.g. **M62#1** *Error*	e.g. **M62#2:4** *Takes logical 4 as a second auxiliary*
M66	e.g. **M66** *Use main as reference*	e.g. **M66:3** *Takes logical 3 as reference*	e.g. **M66#2** *Takes second auxiliary as reference*	e.g. **M66#1:7** *Error Too much data*

5.3.1 Spindle control selection ► M62



M62	Spindle control selection



M62 assigns a spindle to a channel and defines its rank in reference to its logical address.

When the machine is equipped with several spindles, these functions set the commands for the spindle servo-drives.

Syntax

M62:<1..32> [#<1..3>]

S..	RPM spindle speed (with G97).
M62	Spindle control selection.
:<1..32>	Logical address of the spindle.
[#<1..3>]	Rank of the spindle : nothing = main, 1..3 = auxiliary 1 to 3.

Please refer to the beginning of this chapter for detailed information about logical addresses and spindle rank.

Properties of the Functions

The function M62 is a modal *"pre"* function.

Spindle selection

Before selecting a spindle, it is mandatory that this spindle be released by the channel to which it belongs. Releasing a spindle is made by programming M61.

Cancellation

A main spindle control selection will cancel a previous one.

Additionally a spindle release (M61) of the main spindle will cancel any previous selection.

Notes

A spindle receives the functions of the channel to which it is assigned:

- Speed G97 S.., Constant surface speed if any (G96 S...)
- Speed limitation if any (G92 S...)
- Direction of rotation or spindle stop (M03, M04, M05),
- Spindle speed ranges (M40-M45),
- Spindle indexing (M19 EC..),
- Spindle speed override enabled or disabled (M48 or M49).

The spindle of a channel is released by:

- Selecting a new spindle (M62:<1..32>),
- Release function M61.

When released from a channel, a spindle keeps all the characteristics it had when released but new functions for the channel are no longer addressed to it. They are addressed to the new spindle assigned to the channel if any.

Example

```
N.. ...
N130 M62:8                    Assignment as main spindle of logical spindle number 8
                              to the channel spindle control
N140 G97 S500 M03 M40         Spindle 2 control selection
N..
```

5.3.2 Spindle range ► M40, M41, M42, M43, M44, M45



M40/M41/M42/M43/M44/M45 Spindle ranges

The system allows the definition of six spindle ranges associated with address S.

Each of these functions can be activated on any spindle of the channel.

Syntax

N.. [S..][M03/M04]**M40-M45 [<:1..31< | <#1..3>]**

S..	RPM spindle speed (with G97).
M03/M04	Spindle direction of rotation.
M40 to M45	Spindle range selection.
<:1..31>	Spindle logical address.
<#1..3>	Spindle rank in the channel.

Please refer to the beginning of this chapter for detailed information about logical addresses and spindle rank.

Without suffix, the functions M40..M45 refer to the main spindle of the channel.

Properties of the Functions

Functions M40 to M45 are modal «pre» functions.

Cancellation

Functions M40 to M45 cancel one another.

Notes

The minimum and maximum speeds are defined for each range by the machine manufacturer.

Example:

- M40 = 50-500 rpm
- M41 = 400-900 rpm
- M42 = 800-4200 rpm

Example

```
N.. ...
N30 G97 S650 M41 M03          Range M41 for the main spindle
M42#2                         Range M42 for the second auxiliary spindle
M40:8                         Range M40 for the logical spindle number 8
N..
```

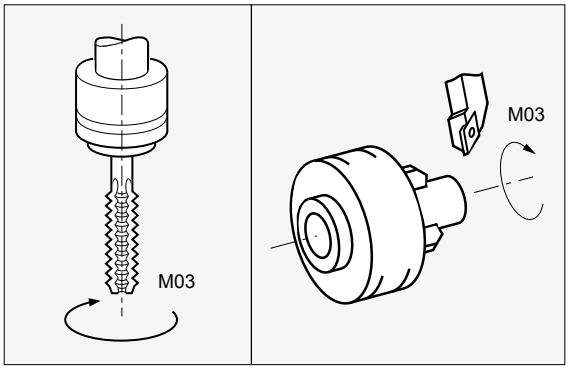
Optionally, the spindle range can be determined automatically by programming the S address and the rpm. In case of ranges overlapping the selection highest range or lowest range is preset by the OEM (see machine operation manual).

Even with automatic range selection it is always possible to force a range as long as it is within the authorized values.

5.3.3 Spindle rotation selection ► M03, M04, M05

M03 Spindle clockwise rotation. [:<1..32> / #<1..3>]

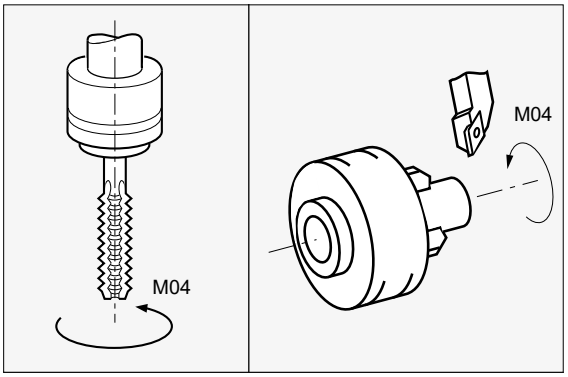
This command starts spindle rotation at the speed programmed.



M04 Spindle counter-clockwise rotation. [:<1..32> / #<1..3>]



This command starts spindle rotation at the speed programmed.



M05 Spindle off. [:<1..32> / #<1..3>]

This command stops spindle rotation.

Without a suffix these functions refer to the main spindle of each channel:

- The suffix **: <1..32>** refers to a particular logical spindle
- The suffix **#<1..3>** refers to a specific spindle rank in the channel
- Referring to another spindle than the main spindle will only be accepted if the said spindle is released (M61) or strictly belongs to the same channel.

Syntax

N.. **M03/M04/M05** [:<1..32> / #<1..3>]

M03	Spindle clockwise rotation.
M04	Spindle counter-clockwise rotation.
M05	Spindle off.
[:<1..32> | #<1..3>]	Address a particular spindle of the channel.

Properties of the Functions

Functions M03 and M04 are decoded modal "pre" functions.

Function M05 is a decoded modal "post" function. It is the default function.

Cancellation

Functions M03, M04 and M05 cancel one another.

Functions M00, M19 and M01 (enabled) cancel functions M03 or M04.

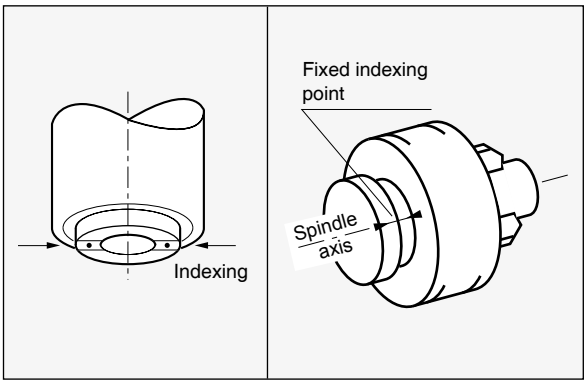
Example

```
N.. ...
N120 ...(MILLING CUTTER WITH R/H HELIX)   Tool call
N130 M03 ...                              Spindle clockwise rotation of the main spindle
N..
N..
N220 M05#1...                             Stop first auxiliary spindle
N..
```

5.3.4 Indexed spindle stop ► M19

M19	Indexed spindle stop.

This function stops the spindle in a position defined with reference to a fixed point.



Syntax

M19 [<:1..31< | <1..3>] [C +- ... | EC +- ...]

S..	RPM spindle speed (with G97).
M03/M04	Spindle direction of rotation.
M40 to M45	Spindle speed ranges.
EC±..	Optional argument defining the indexing angle (in degrees) for a milling spindle.
C+-...	Optional argument defining the indexing angle (in degrees) for a turning spindle.
M19	Indexed spindle stop.
<:1..31>	Spindle logical address.
<#1..3>	Spindle rank in the channel.

Please refer to the beginning of this chapter for detailed information about logical addresses and spindle rank.

M19 alone indexes the main spindle of the channel.

Properties of the Function

Function M19 is a modal "pre" function.

Cancellation

Function M19 is cancelled by one of functions M03, M04 or M05 of the same spindle.

Notes

It is not necessary that the spindle be in rotation before requesting indexation. If the spindle is not rotating, indexing is carried out by positioning along the shortest path.

When the system includes bidirectional orientation capabilities, the stopped position is reached by the shortest path.

It is suggested to have a range selected before requesting an indexing. However if no range is active the system will use the indexing speed defined in machine parameters and select the range accordingly.

Example

Indexed spindle stop at + 90 degrees with reference to the origin.

`N.. ...`	Tool call
`N120 G97 S500 M03 M42`	Spindle rotating
`N130 EC90 M19#1`	Indexed spindle stop for the first auxiliary spindle
`N..`	

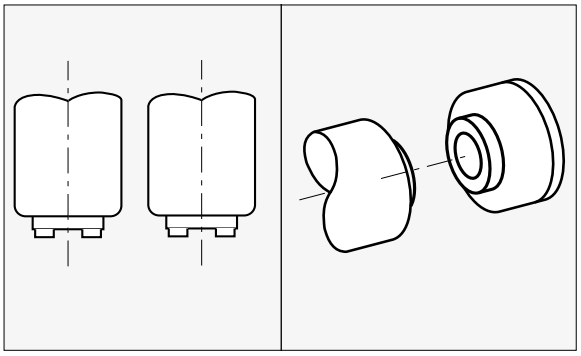
Indexing position

The indexing position can be defined at any value (within 360°). This position is introduced by address C for a turning channel and address EC for a milling channel.

Once M19 is active a new indexing position can be defined. It will be reached by the shortest path.

5.3.5 Spindle release ► M61

M61	Spindle release.



M61 release a spindle to make it available for another channel.

When the machine is equipped with several spindles, these functions set the commands for the spindle servo-drives.

This function can be activated on any spindle of the channel.

Syntax

M61[:<1..32> | #<1..3>]

M61	Spindle release command.
:<1..32>	Logical address of the spindle to release.
#<1..3>	Rank of the spindle to release.

When no spindle is stipulated, the function relates to the main spindle.

Please refer to the beginning of this chapter for detailed information about logical addresses and spindle rank.

M61 alone release the main spindle.

Properties of the Functions

The function M61 is a modal *"pre"* function.

Spindle selection

Before selecting a spindle, it is mandatory that this spindle be released by the channel to which it belongs. Releasing a spindle is made by programming M61.

This function is used to make a spindle available to another channel or rank.

According to the setting of machine parameters:

- A released spindle will keep speed and status (rotation, indexing, stop) "OR".
- A spindle MUST be stopped before being released otherwise an error will be triggered.

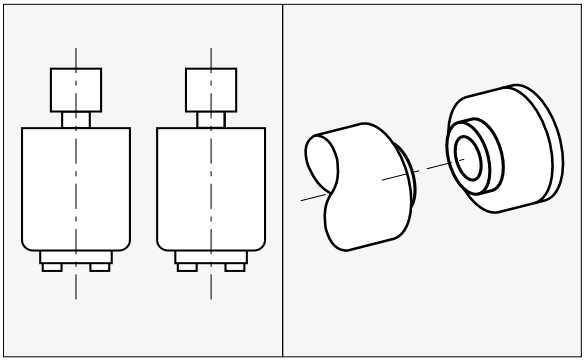
Cancellation

Release will be cancelled by a new selection of that same spindle.

At power-up and after a reset the spindles are associated as defined by machine parameters.

5.3.6 Spindle measurement selection ► M66

M66	Spindle measurement selection.



When the machine is equipped with several spindles, these functions are used to define which spindle will be used as reference for measurement (feed/rev).

For reference for measurement any spindle can be used even if it does not belong to the same channel.

Syntax

N.. **M66 [:<1..32>]**

M66	Spindle measurement.
[:<1..32>	Logical address of a particular spindle..

Properties of the Functions

Function M66 is a decoded modal *«pre»* function.

M66 alone refers to the main spindle of the channel.

Cancellation

Function M66 is cancelled by the selection of a new reference spindle.

At power on, at the end of a program M02 or after a reset, the measurement spindle is assigned to the main spindle of each channel.

Notes

Each channel can use the measurement of any spindle.

Several channels can use the measurement of the same spindle.

When a declared spindle does not have an axis encoder, measurement of this spindle is simulated by the NC.

After a power-up or reset the measurement reference spindle is the main spindle.

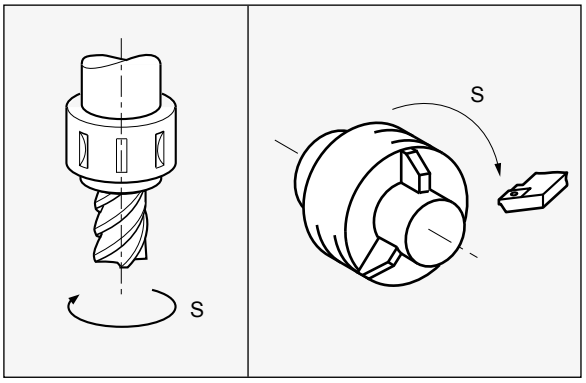
Example

```
N.. ...
N180 M66:2          Assignment of the logical spindle 2 for measurement
N190 G95 F0.15      Feed in mm/rev related to spindle 2
N..
```

5.4 Spindle speed

5.4.1 Spindle speed control ► G97

G97	RPM spindle speed.



This function defines a fixed spindle speed programmed with the S word.

The speed can be defined for any spindle of the channel either with its logical address or its rank.

If neither the logical address or the rank is defined the value refers to the main spindle.

Syntax

N.. **G97** S.. [:<1..32> | #<1..3>]

G97	Spindle speed in rpm.
S..	Mandatory argument associated with the function to define the speed.
M03/M04	Spindle rotation direction.
[:<1..32> | #<1..3>]	Address a particular spindle of the channel.

Properties of the Functions

Function G97 is modal. It is the default function.

Cancellation

On Milling Machine function G97 is the only operating mode.

On Turning this function is cancelled by G96 S...
(constant surface speed).

The spindle speed programmed by G97 is cancelled by S0 or modified by programming S.. followed by a new value.

Notes

Spindle speed format

The speed format may differ according to the machine type:
- Format S06 (1 to 100000rpm - increment 10 rpm),
- Format S05 (1 to 65000 rpm),
- Format S032 (0.01 to 650 rpm).

Example

```
N.. ...
N130 G97 S636 M03                    Spindle rotation
N..
```

Reminder

The spindle rotation speed N is determined from the required cutting speed (V).

The cutting speed V in meters per minute is mainly related to:

- the tool material,
- the part material.
- Cutting speed V = 20 m/min.
- Tool diameter D = 10 mm.

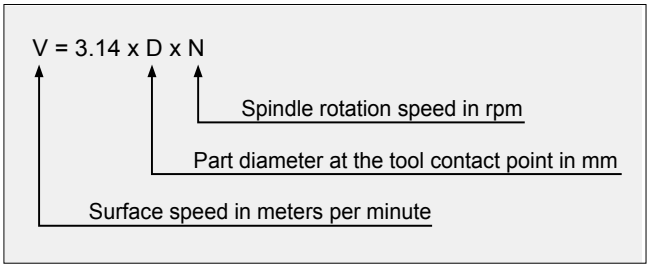
$$N\ (rpm) = \frac{V}{3.14 \times D}$$

$$N = \frac{20000}{3.14 \times 10}$$

N = 636.9 rpm, i.e. S636

5.4.1.1 Review of surface speed

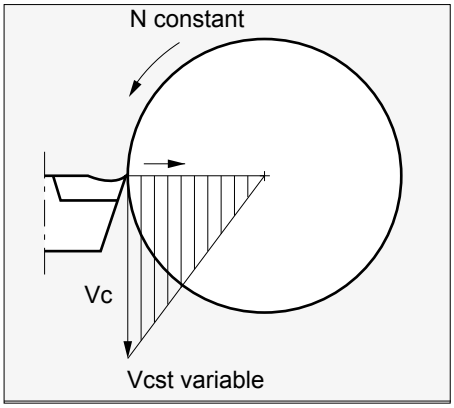
General equation for surface speed V:



5.4.1.2 Using a fixed spindle rotation speed (N)

The fixed spindle speed (N) is expressed in revolutions per minute.

The surface speed V decreases as the tool moves toward the center of the part, (the surface speed is zero at the center).

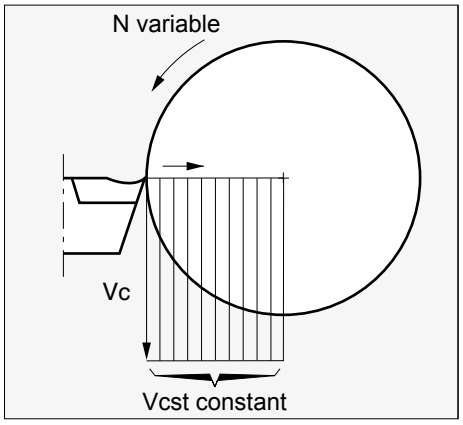


5.4.1.3 Using a constant surface speed (V)

The constant surface speed (Vcst) is expressed in meters per minute.

The surface speed is kept constant as the tool moves toward the center of the part.

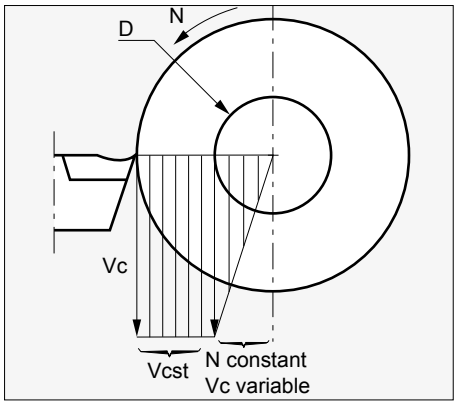
When the tool tip is at the center, diameter D is equal to zero and speed N should be infinite (physically impossible because of the maximum physical rotation speed of the spindle).



5.4.1.4 Spindle speed limit

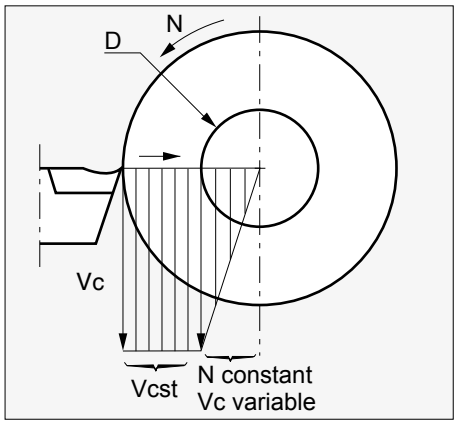
A safety limit can be defined by programming a maximum speed in revolutions per minute.

The limit implies a diameter D beyond which a constant surface speed no longer applies. The system returns to the case of a constant rotation speed and a variable surface speed.



5.4.2 Spindle speed limitation ► G92 S..

G92 S..	Spindle speed limitation.



This function specifies the maximum spindle speed not to be exceeded.

When facing or parting off at constant surface speed (G96), the rotation speed can be limited to prevent problems due to centrifugal force, imbalance, etc.
This function can be activated on any spindle of the channel.

Syntax

N.. **G92** S.. [:<1..32> | #<1..3>]

G92	RPM spindle speed limitation.
S..	Mandatory argument associated with the function to define the maximum spindle speed.
[:<1..32> | #<1..3>]	Address a particular spindle of the channel.

Property of the Function
Function G92 is modal.

Cancellation
Spindle speed limiting is cancelled by:
- cancellation function G92 S0,
- function G92 S.. with assignment of a different speed limit,
- the end of program function (M02),
- a reset.

Notes
Spindle speed limitation:
- must be programmed before selecting constant surface speed (G96),
- must be programmed at a value below the maximum spindle speed,
- is independent of the maximum speed defined in one of the spindle speed ranges (M40-M45).

If the spindle speed programmed by G97 is higher than the speed defined by G92, the spindle rotates at the limited speed.
G92 must be immediately followed by its argument S...

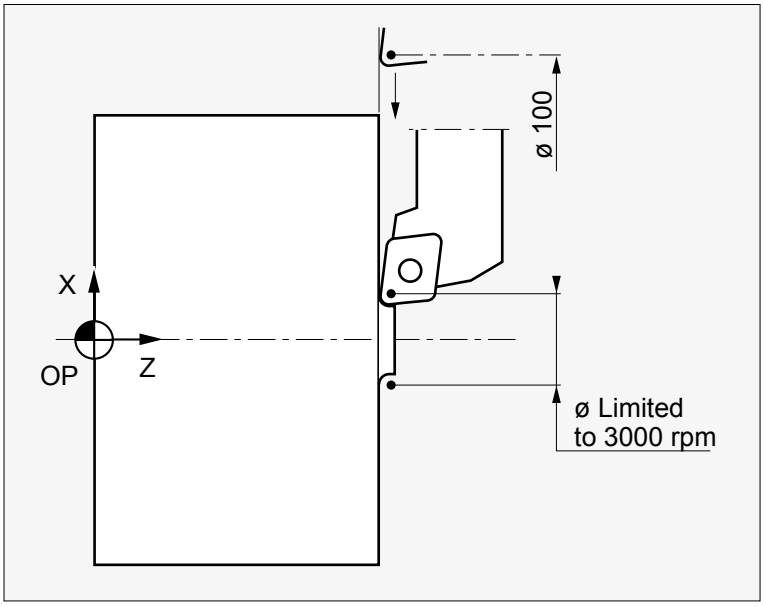
If G92 and S are separated by an axis (X.., Z.., etc.), the system interprets function G92 as preselection of the program origin on the programmed axis (see Sec. 5.7.5) and S as the requested spindle speed.

Example

Programming speed limitation for facing at constant surface speed

On read of block N60, the spindle is initialized at 637 rpm on diameter 100 mm.

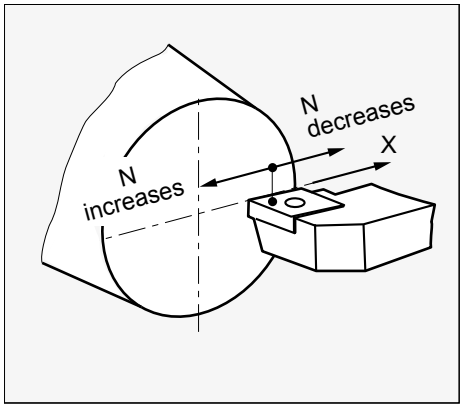
On execution of block N70, the rotation speed is gradually increased up to a maximum of 3000 rpm (approximately 21 mm diameter). The rest of facing up to X0 is carried out at a speed of 3000 rpm.




```
N.. ...
N30 G97 S900 M40 M04          Spindle rotation at 900 rpm
N40 ... X100 Z60              Tool tip positioned on diameter 100
N50 G92 S3000                 Spindle speed limited to 3000 rpm
N60 G96 S200                  Constant surface speed initialized
                              on X100
N70 ... X0                    Facing
N..
N..
N200 G97 S900                 Function G96 cancelled
N..
```

5.4.3 Constant surface speed on Turning machine ► G96

G96	Constant surface speed is expressed in meters per minute.



This function varies the spindle rotation speed (N).
The rotation speed varies according to the position of the tool center with reference to the part diameter.
This function can only be activated on the main spindle of the channel.

Syntax

N.. **G96** [X..] S..

G96	Function forcing a constant surface speed in m/min.
X..	Argument defining the current diameter.
S..	Mandatory argument associated with the function to define the programmed speed.

Property of the Function

Function G96 is modal.

Cancellation

Function G96 is cancelled by function G97 S..

Notes

This function only applies to the main spindle of each channel.

This function can only be programmed when the machine is equipped with a variable speed spindle.

The machine spindle must be rotating when the function is called.

After a retraction with reference to the measurement origin (G52 X..), a new constant surface speed initialization position in X (or U) must be reprogrammed.

When a constant surface speed is programmed:

- the X (or U) axis defining computation of the constant surface speed can be programmed in the same block or one of the previous blocks (between G52 and G96). If X is missing, the system returns error message 28.
- the X (or U) axis must be programmed with reference to the program origin,
- the speed applies to the center of the tool insert radius,
- it can be modified during the program by redefining a new speed by G96 S..,
- enabling of offsets DAT1 and DAT2 does not affect the surface speed.

During machining with constant surface speed, it is recommended to:

- program the feed rate in mm/revolution in order to perform machining with a constant chip thickness,
- cancel the constant surface speed by programming a rotation speed in rpm (G97 S..) before each tool change, then initialize the constant surface speed on the X position of the new tool (and D correction). Program a spindle speed limit to prevent danger when the turned diameters decreased below a certain value.

5.5 Axes Feed rate

5.5.1 Feed rate expressed in Millimetres, Inches or Degrees per Minute ►G94

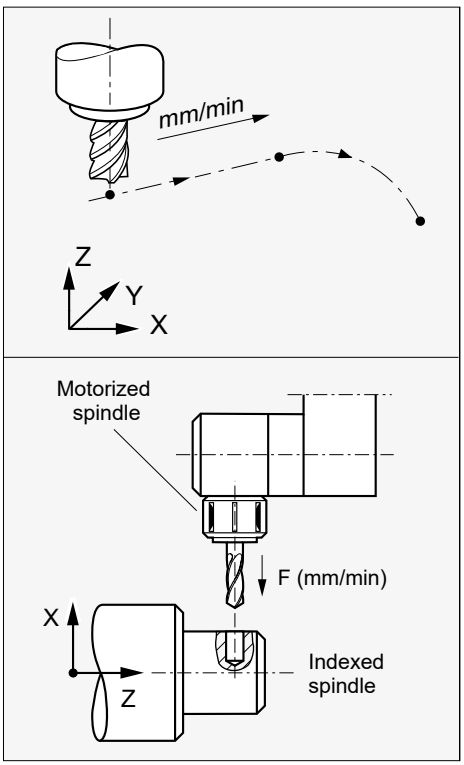
G94	Feed rate expressed in millimetres, inches or degrees per minute.

The feed rate is expressed in millimetres or inches per minute on linear axes and in degrees per minute on programmed rotary axes alone.

The feed rate is expressed on the Cartesian referential defined by the primary axes XYZ.

The other linear axes will be "dragged"; it means their velocity will be adjusted.

However when secondary axes (UVW) are programmed they can become components of this referential.



Rules

- When a secondary axis is defined without the relative primary in a block it becomes the referent.
- When a pair of primary and secondary axes (e.g. YV or ZW) are programmed in the same block the referent is unchanged.
- When a primary axis is defined without the relative secondary in a block it becomes the referent.

Example

`G1 V-1013.537 F5000`	V becomes referent.
`G0 V-430.638 Y1732.447 W0`	V remains referent W becomes referent Y becomes dragged.
`G1 V-439.332 Z819.558 F3000`	Z is referent again.

STOP This notion is important because the feed rate is defined on the main axes of the referential the other axes follow what can lead to a high speed if their travel is bigger that the other axes.
To bypass this effect refer to the function G93 or G92 F...

Syntax



N.. **G94** F.. G01/G02/G03 X.. Y.. Z.. A.. B.. C..

G94	Function setting the feed rate: - in millimetre/min, - in inches/min, - in degrees/min.
F..	Mandatory argument associated with the function and defining the feed rate.
G01/G02/G03	Linear or circular interpolation.
X.. Y.. Z..	End point on linear axes.
A.. B.. C..	Angular end point on rotary axes.

Properties of the Functions

Function G94 is modal. It is the default function.

Reminder

A default value of 1000 mm/min or 100 Inches/min is assigned to address F at power on.

Cancellation

Function G94 is cancelled by one of functions G93 or G95.

Notes

The feed rate limits are defined by the machine manufacturer. When the programmed feed rate exceeds the permissible minimum or maximum rate, the system automatically limits it to the permissible feed rate.

To enter a feed rate in inches/min, the function G70 must be active.

When switching from G94 (unit/min.) to G95 (units /rev.) or back, new G code and F value must be programmed together in the block.

Example

```
N..
N140 G00 X.. Y..
N150 G95 F0.3 G01 Z..
N160 X.. Z.. F0.2
N..
N240 G00 X.. Y.. Z..
N250 G94 F200 G01 Y.. W..

N260 W.. F100
N..
```

Feed rate in mm per revolution

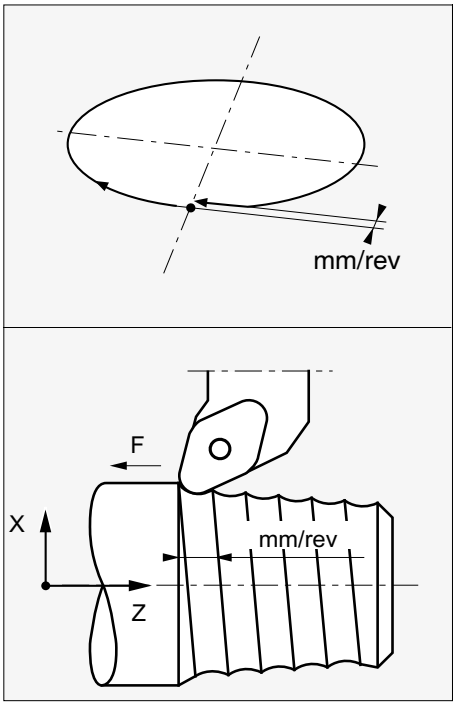
Feed rate in mm/min on a primary axis and a secondary axis

5.5.2 Feed rate expressed in millimetres or Inches per revolution ► G95



G95	Feed rate expressed in millimetres or inches per revolution.

The feed rate is expressed in millimetres or inches per spindle revolution.



Syntax

N.. **G95** F.. G01 / G02 / G03 X.. Y.. Z..

G95	Function setting the feed rate: - in mm/rev, - in in./rev.
F..	Mandatory argument associated with the function and defining the feed rate.
G01 / G02 / G03	Linear or circular interpolation at the programmed feed rate.
X.. Y.. Z..	End point on the linear axes.

Property of the Function

Function G95 is modal.

Reminders

- G94 (units/min) is the default function.
- The initial feed rate is set at 1000 mm/mn or 100 inches/min according to the default programming unit.

Cancellation

Function G95 is cancelled by one of functions G93 and G94.

Notes

The feed rate limits are defined by the machine manufacturer (see manufacturer's technical data). When the programmed feed rate exceeds the permissible values, the system automatically limits it. In addition, the system limits the feed rate to 30 mm/rev anyway.

If a higher value is programmed, the system does not return an error message but simply limits the feed rate to 30 mm/revolution).

When the feed rate unit is changed, the G function defining the new feed rate unit and its programming format must be followed by the argument F.. (if the system is already in G95 mode, address F.. can be programmed alone in a block).

Example

```
N..
N.. G00 X.. Y..
N140 G94 F200 G01 Z..          Feed rate in mm/min
N150 X.. Y.. F100
N..
N240 G00 X..  Y..
N250 G95 F0.3 G01 X.. W..      Feed rate in mm/rev on a primary axis
                               and a secondary axis
N260 W.. F0.2
N..
```



Linear feed variation (see 5.5.8) is also available in G95.

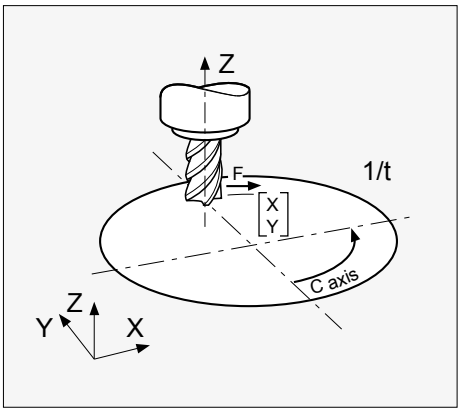
5.5.3 Feed rate in inverse of time (1/min) ► G93



G93	Feed rate in invert of time (1/min.).

The feed rate is programmed in inverse time when the numerical control cannot calculate the length of a path and therefore a feed in inc/min. makes no sense.

Example: A rotary axis programmed alone or with linear axes.



Syntax

N.. **G93** F.. G01 X.. Y.. Z.. A.. B.. C..

G93	Function setting inverse time feed rate coding: min-1.
F..	Mandatory argument associated with the function and defining the feed rate.
G01	Linear interpolation at the programmed feed rate.
X.. Y.. Z..	End point on the linear axes.
A.. B.. C..	Angular end point on the rotary axes.

Property of the Function

Function G93 is modal.

Cancellation

Function G93 is cancelled by one of functions G94 and G95.

The feed rate limits are defined by the machine manufacturer (see manufacturer's technical data). When the programmed feed rate exceeds the permissible minimum or maximum rate, the system automatically limits it to the permissible feed rate.

Reminder

$$\text{Feed rate } F = \frac{V \longleftarrow \text{feed rate in mm/min}}{L \longleftarrow \text{Path length in mm}}$$

STOP Inverse time feed rate should not be used for circular and helical interpolation (no error message).

When the feed rate unit is changed, the G.. function defining the new feed rate unit must be followed by argument F (if the system is already in G93 mode, the F.. address can be programmed alone in a block).

Example

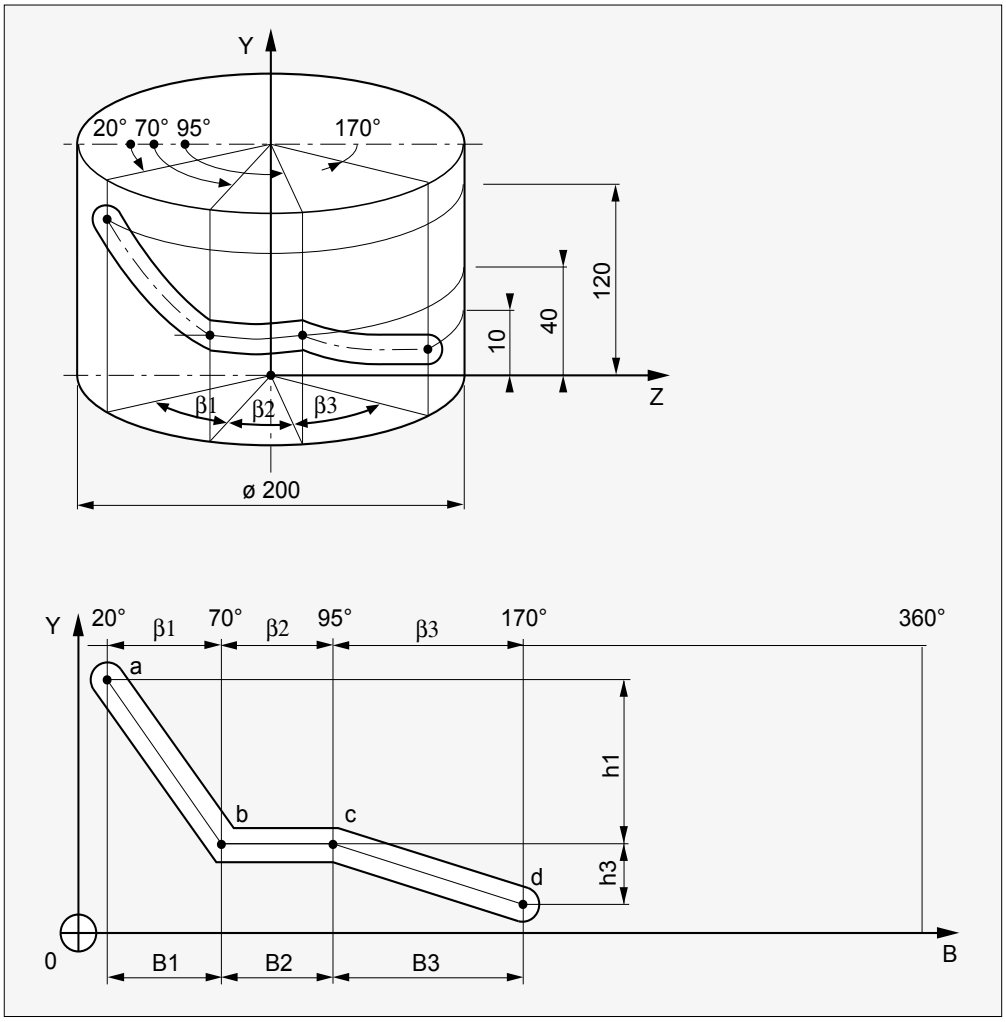
```
N..
N140 G00 X.. Y.. Z..
N150 G94 F200 G01 X.. Y..          Feed rate in mm/min
N160 Y.. F100
N..
N210 G00 Y.. Z..
N220 G93 F5.2 G01 U.. C..          Feed rate in V/L on a secondary axis and a rotary axis
N.. C.. F30
N..
```

Example

F determination in G93

Groove depth = 5 mm. Part diameter = 200 mm

Required cutting speed (V) = 150 mm/min



	Length in mm of the developed arc	Real arc length L in mm	Feed rate in 1/t
	$B = \frac{p \times D \times ß}{360}$	$L = \sqrt{B^2 + h^2}$	F = V/L
a b	B1 = 87.266	L1 = 118.386	F1 = 1.26
b c	B2 = 43.633	L2 = 43.633	F2 = 3.43
c d	B3 = 130.899	L3 = 134.293	F3 = 1.12

```
%50
N10 (G94)                          Feed rate initialized in mm/min
N20 ...                            Tool call
N30 S500 M40 M03
N50 G00 X0 Z150 B0                 Approach position
N60 Y120 Z102 B20                  Point a
N70 G01 Z95 F150                   Penetration in Z
N80 G93 F1.26 Y40 B70              Point b, feed rate in 1/t
N90 B95 F3.43                      Point c, feed rate in 1/t
N100 Y10 B170 F1.12                Point d, feed rate in 1/t
N110 G00 Z300                      Retraction on Z
N120 Y300 B0 M05
N130 M02
```

5.5.4 Case of rotary axes

5.5.4.1 Programming of a modulo rotary axis alone

Example:

```
N.. G91 G94 F40 G01 B30
```

Feed rate F40 is expressed in degrees per minute.

5.5.4.2 Programming of modulo rotary axis and a linear axis

Example:

```
N.. G91 G94 F100 G01 X10 B30
```

The feed rate on the X axis is expressed in millimetres per minute. The feed rate on the B axis depends on the time required to execute the linear path on the X axis.

$t = \Delta X / F = 10 / 100 = 0.1$ minute, i.e. 6 seconds.

The feed rate on the B axis is equal to 30 deg/6 seconds, i.e. 5 deg/s.

5.5.4.3 Programming of two modulo rotary axes

Example:

```
N.. G91 G94 F100 G01 A30 B40
```

The feed rates on A and B axes are expressed in degrees per minute.

Execution time : $t = \sqrt{\Delta A^2 + \Delta B^2} / F$ i.e. t = 30 seconds.

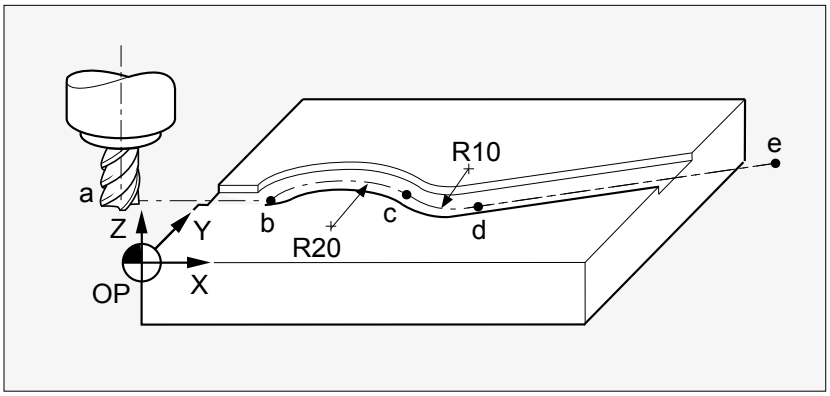
The feed rate on each axis is equal to:

- Feed rate on $A = \Delta A/t$, i.e. 60 deg/min
- Feed rate on $B = \Delta B/t$, i.e. 80 deg/min

5.5.4.4 C axis feed rate for Turning machine

When the C axis is programmed the feed rate represents the relative speed at the tool tip position. Therefore the C rotation speed depends on the position of the X axis.

Example

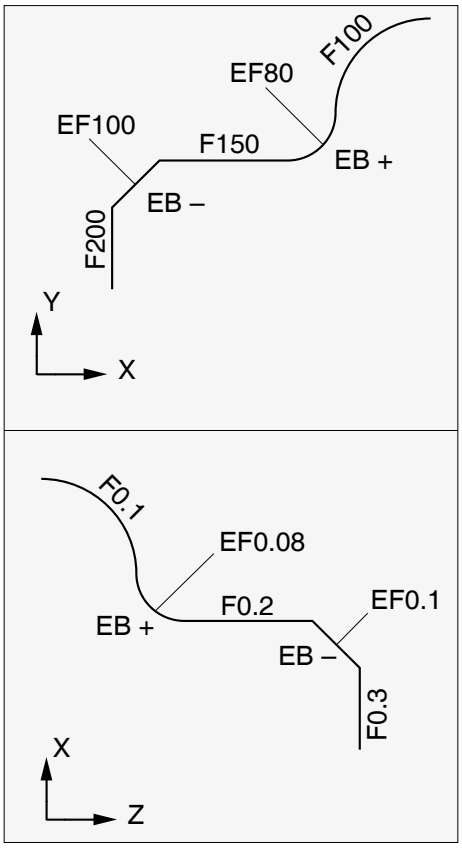



```
%40
N10 G00 G52 Z..                    Rapid to tool change
N20 ...                            Tool call
N30 S800 M40 M03
N40 G94 F100                       Feed rate in mm/min
N50 G00 Xa Ya Z-5
N60 G01 Xb
N70 G02 Xc Yc R20 F160             Change in feed rate
N80 G03 Xd Yd R10 F70              Change in feed rate
N90 G01 Xe Ye F100                 Change in feed rate
N..
```

5.5.5 Feed rate specific to Fillets EB+ and Chamfers EB- ► EF..

EF	Feed rate specific to fillets EB+ and chamfers EB-.

A feed rate different from the modal machining feed rate F can be programmed for fillets and chamfers programmed by EB+ and EB-.



Syntax

N.. Interpolation EB+.. / EB-.. **EF..**

Interpolation	Linear interpolation (G01) or circular interpolation (G02 or G03).
EB+	Fillet dimension.
EB-	Chamfer dimension.
EF..	Feed rate.

Property of the Function

Function EF.. is modal.

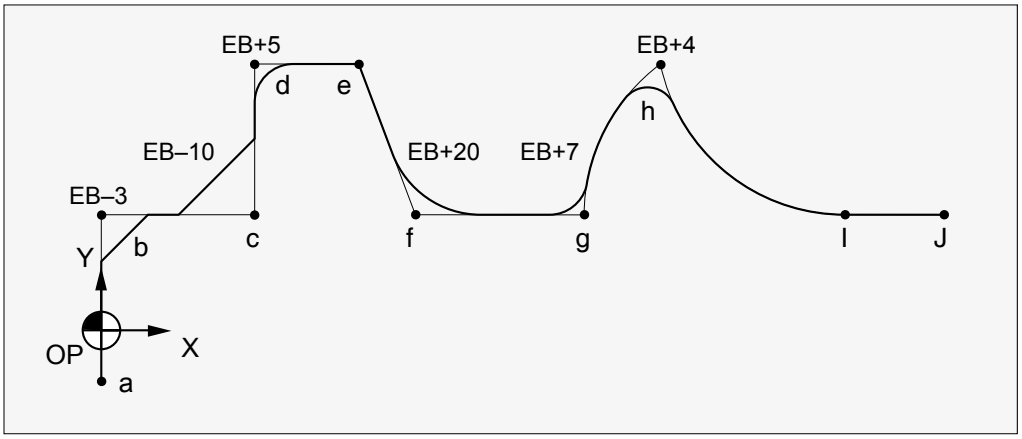
Cancellation

Function EF followed by a value is cancelled by programming:

- function EF followed by a new value, or
- the end of the program (M02).

Feed rate EF is substituted for the modal feed rate F if its value is non-zero and is less than feed rate F.Feed rate EF is in the unit specified by G94 (mm/min) or G95 (mm/rev).
Feed rate F.. in mm/min (G94) or mm/rev (G95) remains modal when executing fillets and/or chamfers

Example



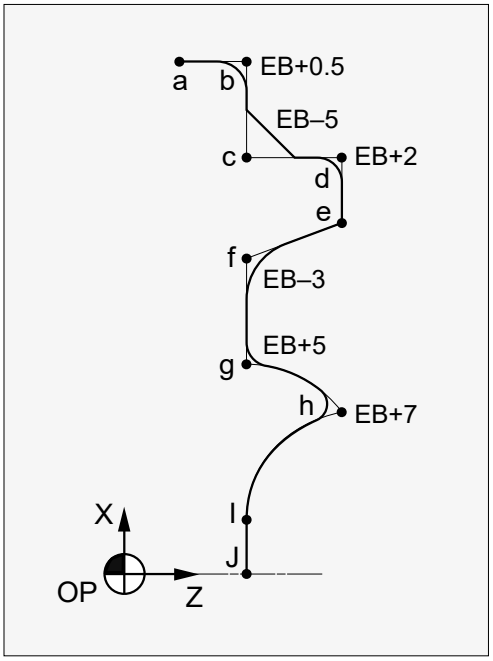
When executing the contour, the linear and circular interpolations are carried out at modal feed rate G94 F120.

```
%37
N10 G00 G52 Z..                    Tool spindle positioning
N20 ...                            Tool call
N30 S800 M40 M03
N40 G92 R1
N50 X0 Y-10                        Point a
N60 Z-5
N70 G94 F120                       Feed rate for G01, G02 or G03 interpolations
N80 G1 Y15 EB-3 EF90               Point b (feed rate  90)
N90 X20 EB-10 EF70                 Point c (feed rate  70)
N100 Y35 EB+5                      Point d (feed rate  70)
N110 X40                           Point e (feed rate  120)
N120 X50 Y15 EB+20 EF90            Point f (feed rate  90)
N130 X70 EB+7 EF70                 Point g (feed rate  70)
N140 G02 X80 Y35 R40 EB+4          Point h (feed rate  70)
N150 G03 X100 Y15 R30              Point i (feed rate  120)
N160 G1 X120                       Point j (feed rate  120)
N170 ...
```

Example

Finishing a contour with feed rate EF in the chamfers and fillet





When executing the contour, the linear and circular interpolations are carried out at modal feed rate G95 F0.4.

```
%39
N.. ...
N110 G00 G52 X.. Z.. G97 S900 M04      Tool spindle positioning
N120 ...                               Tool call
N130 X160 Z30                          Point a
N140 G96 S400
N150 G92 S4000
N160 G95 F0.4                          Feed rate for G1, G2 or  G3
N170 G01 Z50 EB+0.5 EF0.3              Point b (feed rate  0.3)
N180 X140 EB-5 EF0.2                   Point c (feed rate  0.2)
N190 Z60 EB+2                          Point d (feed rate  0.2)
N200 X110                              Point e (feed rate  0.4)
N210 Z60 EB-3 EF0.15                   Point f (feed rate  0.15)
N220 X90 Z50 EB+5                      Point g (feed rate  0.15)
N230 X70 EB+5 EF0.2                    Point h (feed rate  0.2)
N240 G02 X40 Z70 R30 EB+7              Point i (feed rate  0.2)
N250 G03 X10 Z60 R20                   Point j (feed rate  0.4)
N260 G1 X0                             Point k (feed rate  0.4)
N270 ...
```

5.5.6 Tangential feed rate ► G92 R.., G92 L..



G92 R..	Programming the tangential feed rate.

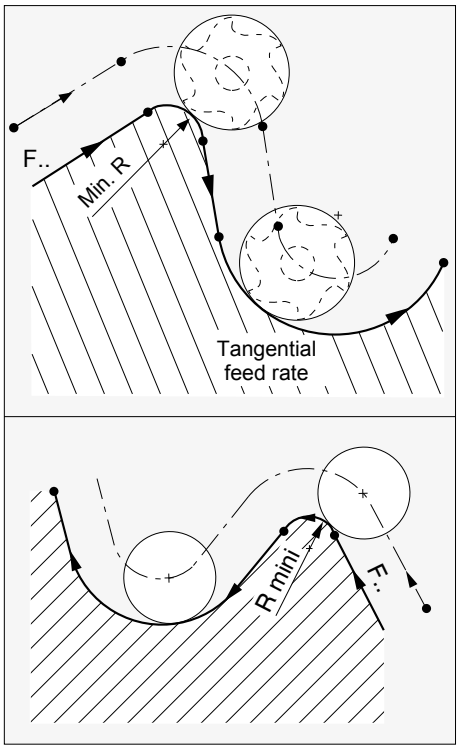
The feed rate is normally applied at the tool center.

With this function it is possible to choose to apply the feed rate along the part when using tool radius correction.

G92 R applies when the curvature is made of circular arcs. If the curvature is described by consecutive segments of G01, then G92 L must be used to obtain a constant tangential feedrate.

G92 L applies to blocks transitions G01->G01 (not on transitions G02/G03 <-> G01) and for angles between the consecutive segments less than 30°.

The programmed speed is corrected as follows: F corrected = F programmed * Length trajectory tool centre point / Length programmed trajectory. For G92 R the same formula applies using the ratio of the radii).



Syntax

N.. **G92** R.. or N.. **G92** L...

G92	Tangential feed rate applied to tool radius correction.
R..	Mandatory argument defining the minimum value of curve radius below which the tangential feed rate calculation (which could otherwise reach a too high value) has to be ignored.
L..	Argument defining the minimum length of segments under which a speed limitation doesn't apply.

Properties of the Function

Function G92 followed by argument R or L or both is modal.

Cancellation

Tangential feed rate G92 R.. and/or L... is cancelled by:
- cancellation function G92 R0,
- function G92 R.. with a different radius,
- the end of program function (M02),
- a reset.

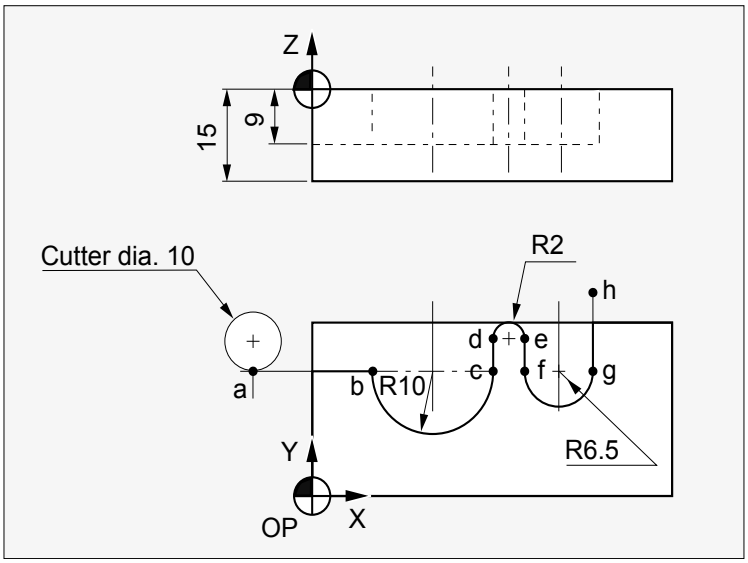


Function G92 is ignored during automatic creation of a connecting circle between two secant elements (lines or circles) with radius correction. The feed rate will then be the same as the feed rate programmed in the previous block.
Function G92 programmed in a block must not be accompanied by axis commands.

Example

In this example, the tangential feed rate is applied to curves whose radius is greater than 3 mm.

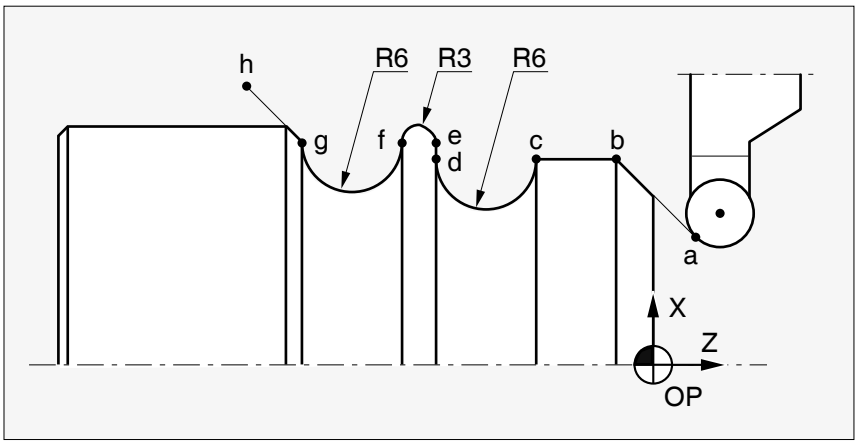





```
%22
N10 G00 G52 Z..                          Retraction for tool change
N20...                                   Tool call
N20 T12 D12 M06 (CUTTER DIA=10)
N30 S600 M40 M03
N40 G00 G41 Xa Ya                        Point a, left radius offset
N50 Z-9
N60 G92 R3                               Tangential feed rate limit
N70 G01 Xb F200
N80 G03 Xc Yc R10                        Feed rate applied to point of tangency
N90 G01 Yd
N100 G02 Xe Ye R2                        Feed rate applied to tool center
N110 G01 Yf
N120 G03 Xg Yg R6.5
N130 G01 Yh
N140 G92 R0                              Tangential feed rate cancelled
N150 G00 G40 G52 Z..
N160 M02
```

Example

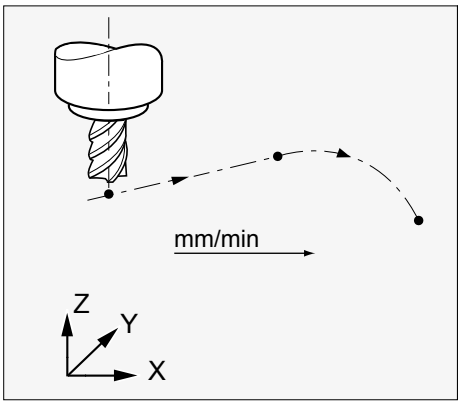
In this example, the tangential feed rate is applied to curves whose radius is greater than 4 mm.




```
%23
N10 G00 G52 X.. Z..                          Tool change position
N20 _                                        Tool call
N30 S900 M40 M04
N40 G95 F0.2
N50 G00 G42 Xa Za                            Right radius offset
N60 G92 S3000
N70 G96 S100
N80 G92 R4                                   Tangential feed rate limit
N90 G01 Xb Zb F0.2
N100 Xc Zc
N110 G02 Xd Zd R6                            Feed rate applied to the point of tangency
N120 G01 Xe, Ze
N130 G03 Xf Zf R3                            Feed rate applied to the tool center
N140 G02 Xg Zg R6
N150 G01 Xh Zh
N160 G92 R0                                  Tangential feed rate cancelled
N170 G00 G40 G52 X.. Z.. G97 S900 M05
N180 M02
```

5.5.7 Setting feed rate per axis set ► G92 F..

G92 F	Setting feed rate per axis.



For a sequence of blocks containing movements on linear and rotary axis G92 F... allows to set the feed rate on one or more axes but not necessarily all axes involved in the trajectory.

This is generally used to maintain a constant speed on rotary axes.

Syntax

G92 F... <list of axes>

G92	Introduces the function feed rate per axes set.
F..	Feed rate to use.
<[X][Y][Z][...][...]>	Set of axes on which combined movement the feed rate will be applied.

Properties of the Function

Function G92 F is modal
Default is Feed rate applied on the primary axes.

Cancellation

- G92 F... <other set of axes>
- G92 F no axes
- Reset

When the argument F is programmed after function G92, the block includes a list of axes on which the feed rate F has to be applied. If no axes are listed, the feed rate is processed as usual i.e. to the primary axes.

Errors detected

The system generates error message 2 in the following cases:

- The axes list includes a value,
- Some axes from the list do not belong to the group,
- The list of axes includes both linear and rotary axes.

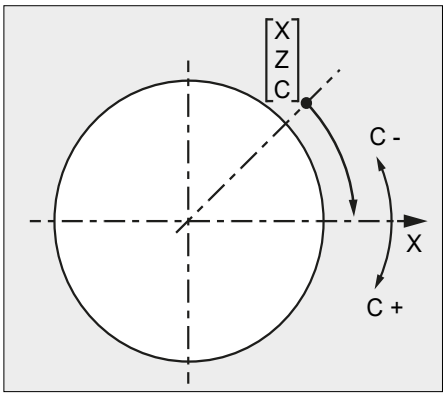
Example

```
G92 F50 A
A+50 X1
A+75 X10
```

Axis A will move at 50 deg/min the speed of X axis will be adapted to follow

5.5.8 POLAR CONSTANT FEEDRATE ► G20 G92 F... P

G92 Fxxx P Polar constant feed on XY.



This feed mode is only available in G20 polar coordinates and if a coupling of the X to the C axis is defined with the MLEGB..

In this mode the NCK will calculate the speed on X and C in order to maintain the programmed feed constant along the programmed path in XY.

Syntax

G92 Fxxx P

G92	Introduces the function
F...	Feed rate to use
P	P code to define polar constant feed rate

Properties of the Function

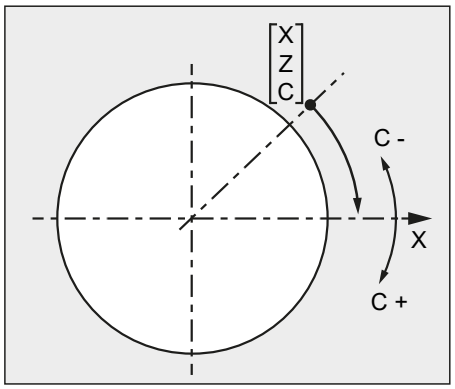
Function G92 F… P is modal.

Cancellation

- Other G92
- G92 F
- Reset

5.5.9 XY CONSTANT CONTACT SPEED ► G20 G92 F... Q

G92 Fxxx Q Constant contact speed on XY.



This feed mode is only available in G20 polar coordinates and if a coupling of the X to the C axis is defined with the MLEGB..

In this mode the NCK will calculate the speed on X and C in order to maintain the programmed feed of the contact point constant along the programmed path in XY.

This feed mode only works on polar contour offset generated by the C.OFF function and not with offset generated by tool radius compensation.

Syntax

G92 Fxxx Q

G92	Introduces the function
F...	Feed rate to use
Q	Q code to define constant contact feed rate on XY.

Properties of the Function

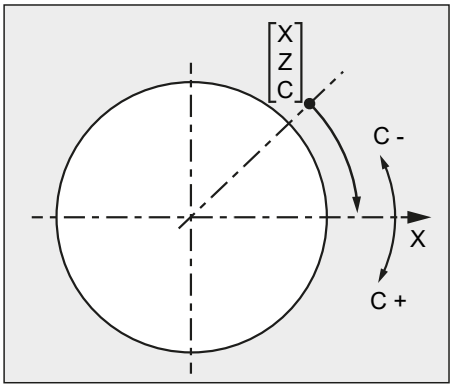
Function G92 F… Q is modal.

Cancellation

- Other G92
- G92 F
- Reset

5.5.10 CONSTANT C SPEED ► G21 G92 F... C

G92 Fxxx C Constant C speed in G21.



This feed mode is only available in G21 cartesian coordinates.

The C speed will remain constant during X-C interpolation. The speed on X will

be adjusted in consequence to respect the condition on C. This may result in abrupt and unavoidable velocity steps.

To remove these steps, it is necessary to renounce to a constant speed on C by enabling and configuring the look ahead in G21. The NCK will in this case reduce when necessary the speed on C in order to respect the speed crossing angle on X.

In G92 Q, the Cartesian velocity could become very high for small circumferences with a big tool radius. To prevent an excessive speed, the parameter E79014 has been created to clamp the maximum feed.

Syntax

G92 Fxxx C

G92	Introduces the function
F...	Feed rate to use
C	C code to define constant C axis feed rate.

Properties of the Function

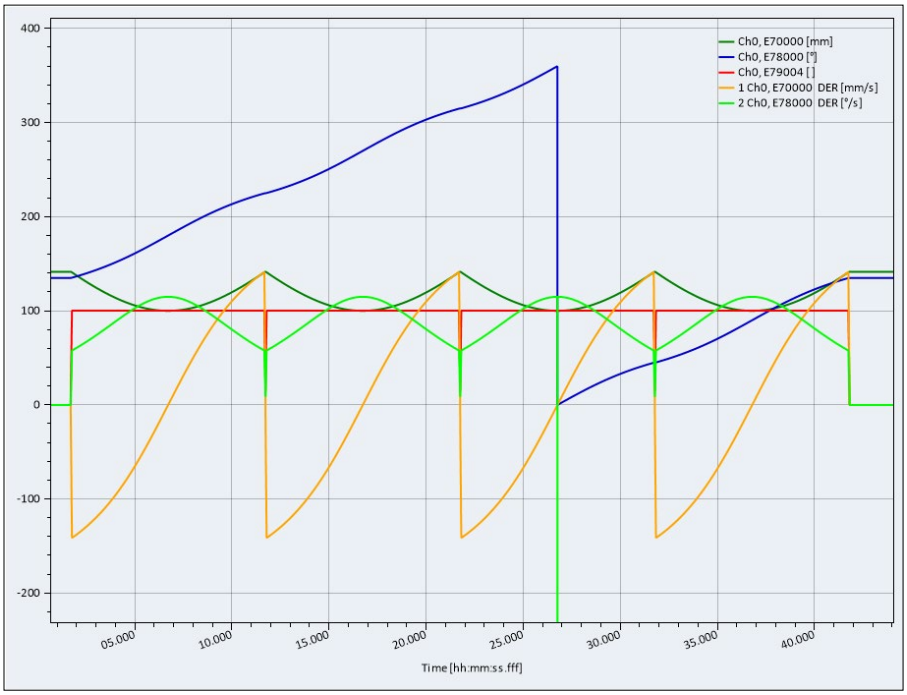
Function G92 F… C is modal.

Cancellation

- Other G92
- Reset

Speed steps

As explained above, the X speed can have abrupt variations to comply with the constant C speed. An example for executing a square is given below:



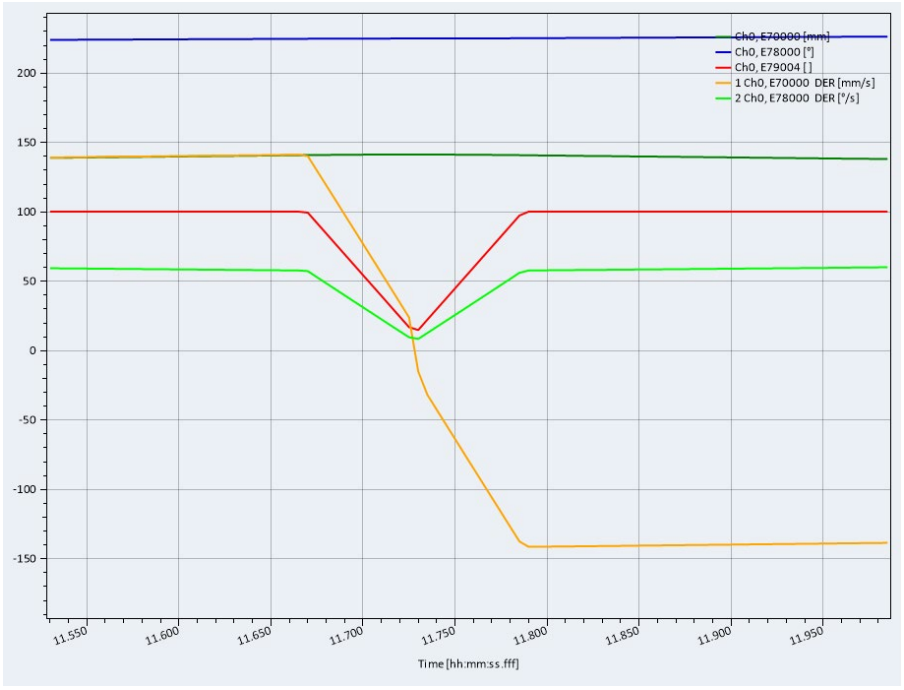
Red trace represents the speed. Dark green and blue traces represent the respective X and C positions

Orange and light green the respective velocities.

One can notice the big and short velocity inversion mostly on X that can be not acceptable.

Enabling the segmentation E11033=1 (default value), the speed crossing angle E11032=1 (default value), setting the segmentation value at 3mm E32020=3 yields the feed reducing on the square vertices according to the feed crossing angle.

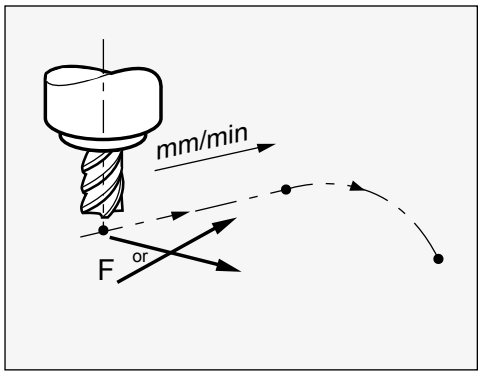
This feed reduction is more evident by zooming the picture above.



5.5.11 Linear feed variation ► G94/G95F.. EF..

G94/G95F.. EF..	Linear feed variation.

For particular application (Jet cutting, grinding, etc.) it is possible for the feed rate to increase or decrease linearly during the complete block execution.



Syntax

G94/G95 F.. EF..

G94/G95	Linear feed variation.
EF0	Constant feed rate (standard functionality).
EF1	Increasing or decreasing feed rate.

The functionality is also available with G95 (feed/rev).

Programming G9x and EFx in the same block is mandatory and they should be alone in this bloc (no movement). The speed variation will only be applied on the following block.

Property of the Function

This function is modal.

Cancellation

G9x EF0 or Reset.

In any case, the speed management makes that the programmed speed in a block will never be exceeded (except by the override).

This means:

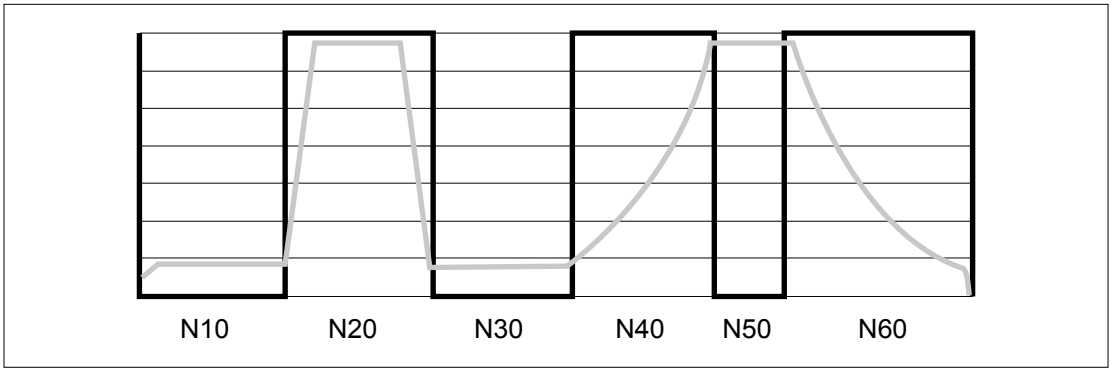
- The system will decelerate before the end of a block if the feed rate in the next block is lower
- The system will accelerate at the beginning of a block if the feed rate is higher than in the previous block.

In the particular case of variable feed rate the system will start the block at the speed of the previous block and modulate it linearly to achieve the programmed feed rate at the end of the block (assuming the acceleration permits).

If the performance of the system does not permit to reach the requested speed, one of the two following situation occur:

- In case of higher feed rate: The system will reach the maximum possible speed according to the limited acceleration capacity. Further increases can take part in the following blocks
- In case of lower feed rate: The deceleration will be initiated on the previous block(s) in order to start the block concerned at a speed permitting to reach the goal of a the correct feed rate at the end. The feed rate variation will be limited by the deceleration capacity.

The above is illustrated on the graphic below: (speed in relation to time).



The black line represents the different blocks.

The grey line represents the speed.

Note 1

The parabolic shape of the curve at N40 and N60 is due to the fact that the recording is in relation to the time. In fact the speed increase/decrease is linear with respect to the distance travelled.

```
N10 F100 X5          \\ Start moving; acceleration up to F500; end of block at F500.
N20 F1000 X55        \\ Acceleration up to F1000; move; deceleration down to F100 before
                        end of block (to start N30 at the correct feed rate).
N30 F100 X60         \\ Move at F100 for the whole block.
G94 EF1              \\ Select variable feed rate.
N40 X80 F1000        \\ Increase speed up to F1000 during the whole block
N50 X100             \\ Maintain F1000
N60 X130 F100        \\ Decrease speed down to F100 during most of the block then stop
                        as usual.
N70 M0
```

Note 2

Variable feed rate is not compatible with triangular acceleration.

5.6 Axes programming and interpolation

5.6.1 Rapid positioning ► G00

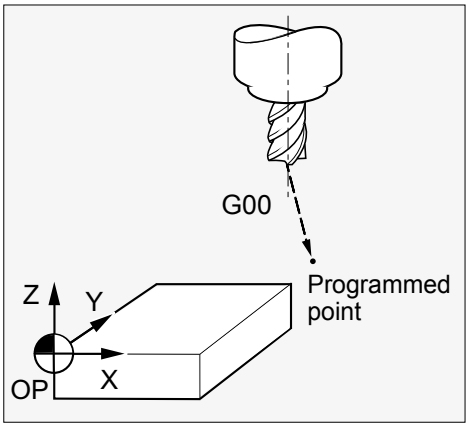


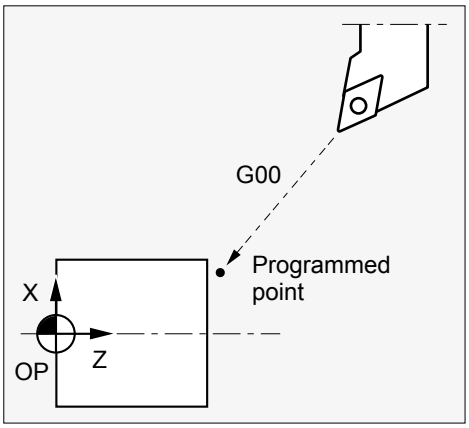
G00	Linear interpolation at high speed.

The programmed point is reached at high speed by a linear path.

The path is a combination of all the axis movements programmed in the block.

Programmable axes:
- X, Y, Z primary axes,
- U, V, W secondary axes,
- A, B, C rotary axes.





Syntax

N.. [G90/G91] **G00** [R-/R+] X.. Y.. Z..

G90/G91	Programming in absolute or incremental dimensions.
G00	Rapid positioning.
R-/R+	Defines a shift of the end point (R- before, R+ after) the value of the shift is equal to the radius of the current tool.
X.. Y.. Z..	End point: - Point coordinates with G90. - Value of the movement with G91.

Property of the Function

Function G00 is modal.

Cancellation

Function G00 is cancelled by one of functions G01, G02 or G03.

The speed along the path programmed in G00 is determined by the most solicited axis (it will run at its max speed).

Optional arguments R+ or R-:

- are active only in the block where they are programmed. The shift is applied in the interpolation plane.
- cannot be programmed in a block with PGP.

Programming of the additional axes and carrier/carried axis pairs

Two axes of carried/carrier axis pairs can be programmed in simultaneously G00 using function G52 (programming with reference to the machine origin).

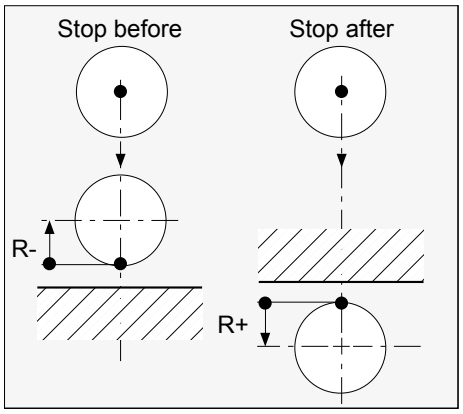
Unlike in G1 programming a rotary axis at the current position will not result in a full turn. The move is considered executed immediately (Positioning function).

5.6.1.1 Positioning with respect to the cutter diameter (R-/R+)



R+ or R- can be programmed to reach the specified approach point regardless of the read tool radius

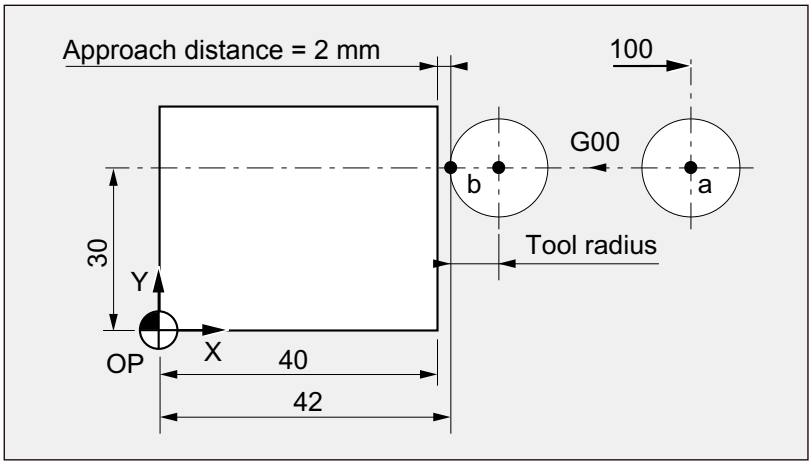
Positioning is only applied to the axes of the plane



Example

Positioning with a 2mm guard



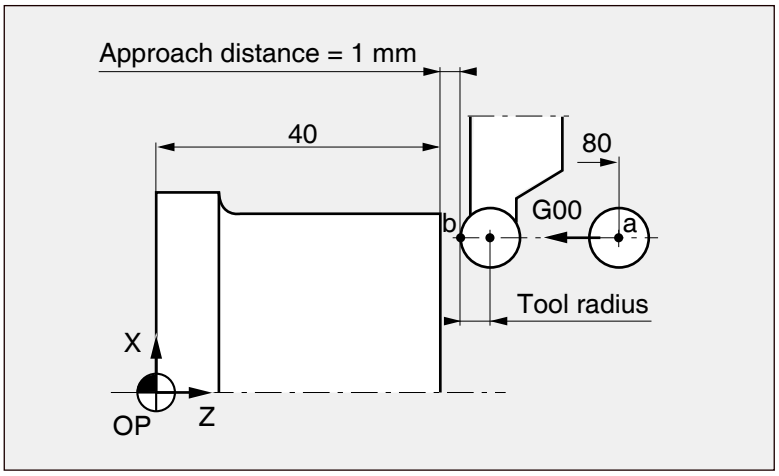



```
N.. ...
N.. D..
N140 G00 X100 Y30 Z-10
N150 R- X42
N160 ...
N..
```

Tool correction (see chapter 5.11)
Point a
Point b, stop before the programmed point

Example

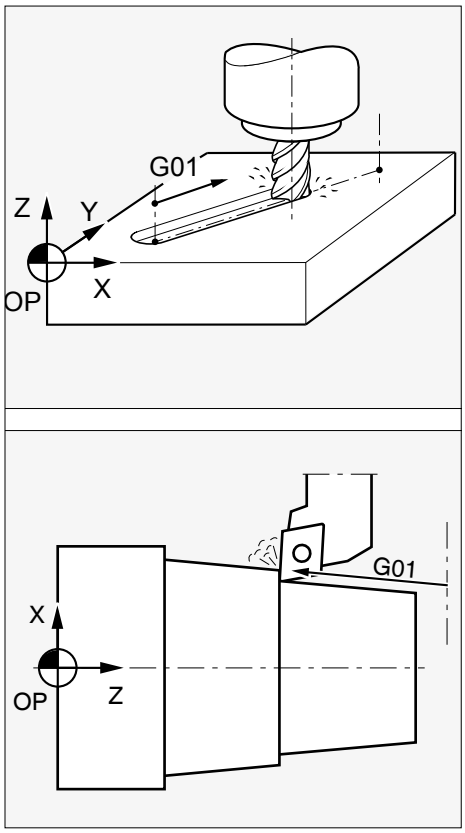
Programming to stop at an approach distance of 1 mm




```
N.. ...
N.. D..                 Tool correction (see chapter 5.11)
N140 G00 X40 Z80        Point a
N150 R- Z41             Point b, stop before the programmed point
N160 ...
N..
```

5.6.2 Linear Interpolation ► G01

G01	Linear interpolation at the programmed feed rate.



The programmed point is reached by a linear path at the programmed feed rate.

The path is a combination of all the axis movements programmed in the block.

Programmable axes:
- X, Y, Z primary axes,
- U, V, W secondary axes,
- A, B, C rotary axes.

Syntax

N.. [G90/G91] **G01** [R+/R-] X.. Y.. Z.. [F..]

G90/G91	Programming in absolute or incremental dimensions.
G01	Linear interpolation at the programmed feed rate.
R-/R+	Shift end point before (R-) or after (R+) the programmed coordinate. The shift value is equal to the active tool radius. The distance is equal to the active tool radius.
X.. Y.. Z..	End point: - Coordinates of the end point with G90. - Value of the movement with G91.
F..	Feed rate.

Properties of the Functions

Function G01 is modal. It is the default function.

Cancellation

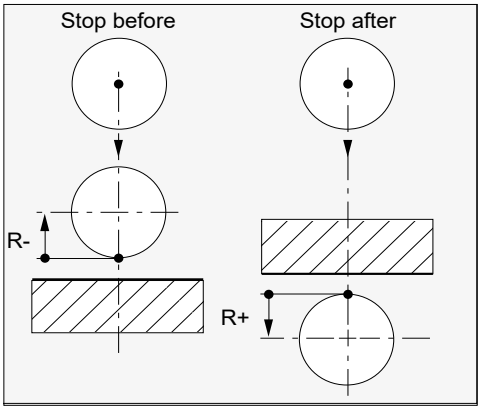
Function G01 is cancelled by one of functions G00, G02 or G03.

Optional arguments R+ or R-:
- are active only in the block where they are programmed,
- cannot be programmed in a block with PGP.

5.6.2.1 Positioning with respect to the cutter diameter (R+/R-)

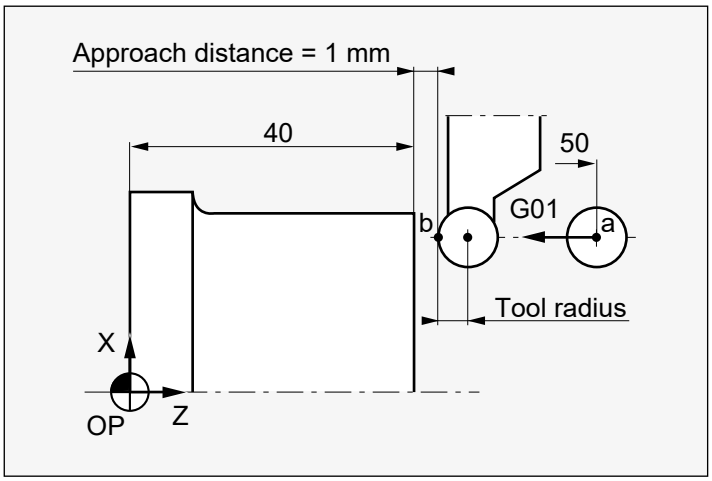
R+ or R- can be programmed to reach the specified approach point regardless of the read tool radius.

Positioning is only applied to the axes of the active plane.



Example

Programming to stop at an approach distance of 1 mm




```
N.. ...
N.. D..                  Tool correction (see chapter 5.11)
N140 G00 X40 Z50         Point a
N150 G01 R- Z41 G95 F0.2 Point b, stop before the programmed point
N..
```

5.6.2.2 Programming of additional axes and carrier/carried axis pairs

Linear interpolations can be executed as combined movements on primary and additional axes.

Example:

Linear interpolation on a primary axis and a secondary axis.

N.. G01 Y.. W.. F..

Linear interpolation on a secondary axis and a rotary axis.

N.. G01 U.. C.. F..

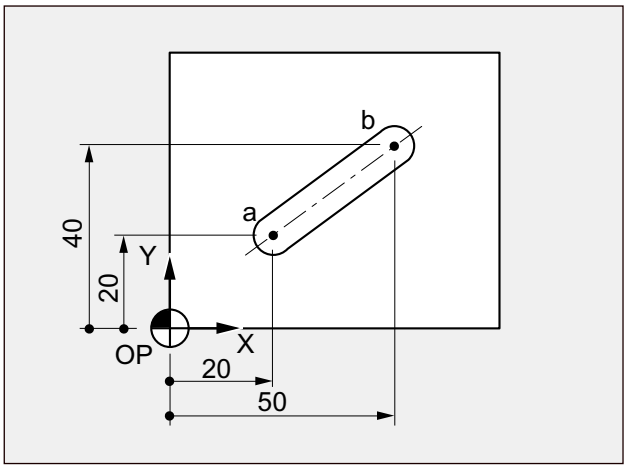
Carrier / carried axis can only be programmed simultaneously in G00 and G52. Any attempt in G01 will result in error 139..

Programming a rotary axis at its current position in G01 will result in a 360° movement, unless parameter E11008 is set to 0.



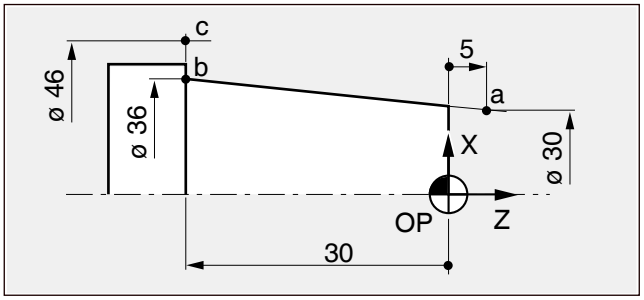
Example

Programming in absolute dimensions (G90)




```
%20
N10 G00 G52 Z..          Rapid to tool change position
N20 ... D8               Tool call offset selection
N30 S600 M40 M03
N40 X20 Y20 Z2           Approach point a
N50 G01 Z-1.5 F50        Penetration in Z
N60 X50 Y40 F120         Point b
N..
```

Turning and facing in absolute programming (G90)

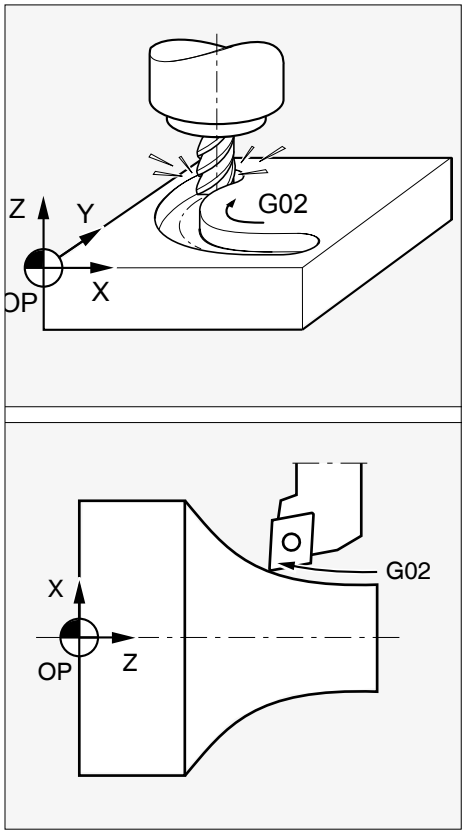



```
%20
N10 G00 G52 X.. Z..        Tool change position
N20 ...                    Tool call
N30 G97 S600 M40 M04
N40 X30 Z5                 Point a, approach
N50 G96 S200
N60 G95 F0.2               Feed rate in mm/rev.
N70 G01 X36 Z-30           Point b, turning
N80 X46 F0.1               Point c, facing
N..
```

5.6.3 Circular interpolation ► G02, G03

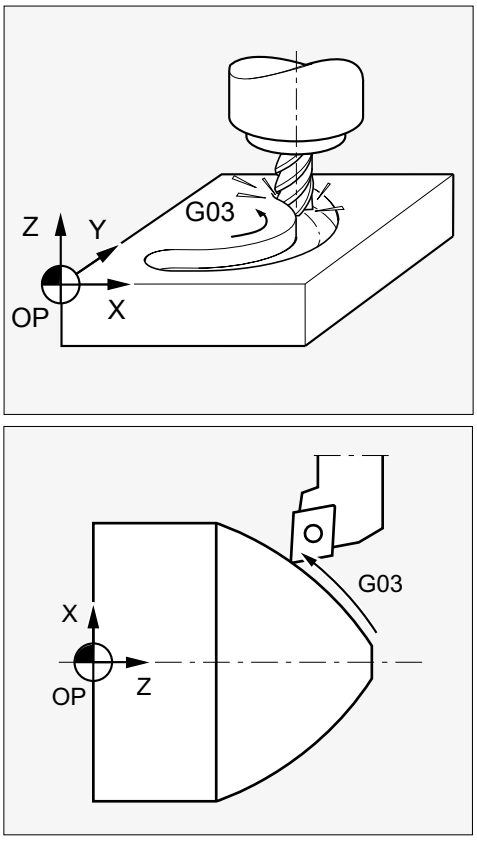


G02	Clockwise circular interpolation at the programmed feed rate.





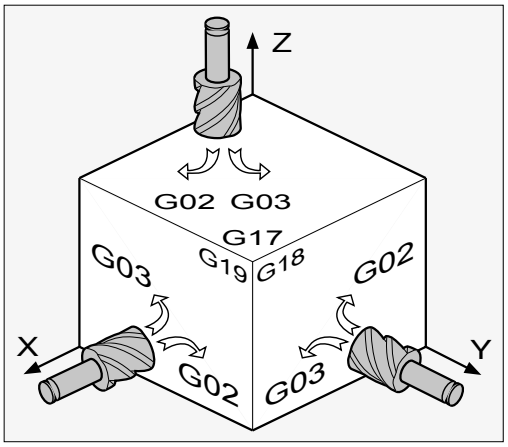
G03	Counter-clockwise circular interpolation at the programmed feed rate.



- The programmed point is reached by a circular path.
- The programmed feed rate is applied at the tool center.
  The two controlled linear axes depend on the choice of the interpolation plane.

For Milling Machine

- X (or U) and Y (or V) axes for G17
- Z (or W) and X (or U) axes for G18
- Y (or V) and Z (or W) axes for G19.



For Turning Machine

- Z (or W) and X (or U) in G20
- X and Y in G21
- Y and Z in G22.

Syntax (XY plane)

N.. [G17] [G90/G91] **G02/G03** X.. Y.. I..J.. / R.. [F..]

G17	XY plane selection.
G90/G91	Programming in absolute or incremental dimensions.
G02	Clockwise circular interpolation.
G03	Counter-clockwise circular interpolation.
X.. Y..	End point. - Coordinates of the end point with G90. - Value of the movement with G91.
I.. J..	Location of the circle center in the XY plane (I along X, J along Y). - With reference to the program origin with G90. - With reference to the interpolation start point with G91.
R..	Radius of the circle.
F..	Feed rate (see chapter 5.10).

Syntax according to the plane selected: G17/G18/G19:

Main interpolation plane	Function	Interpolation	Syntax
XY	G17	G02/G03	X.. Y.. I.. J.. or R..
ZX	G18	G02/G03	X.. Z.. I.. K.. or R..
YZ	G19	G02/G03	Y.. Z.. J.. K.. or R..

Properties of the Functions

Functions G02 and G03 are modal.

Cancellation

Function G02 is cancelled by functions G00, G01 or G03.

Function G03 is cancelled by functions G00, G01 or G02.

Notes

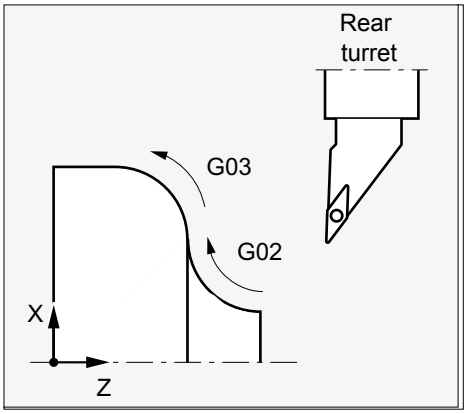
The third axis in the plane if any can be programmed with circular interpolation to give helical interpolation.

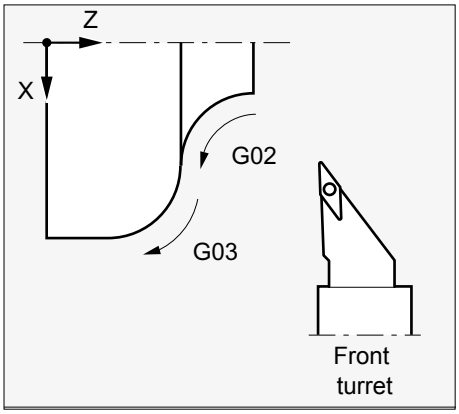
In a block programmed in G02 or G03, all the addresses used for the interpolation must be specified even if they are zero (I0, J0; Plane G17) or unchanged with respect to the previous block (X and Y; Plane G17).

5.6.3.1 Direction of movement according to the turret position with Turning machine

Lathe with one turret





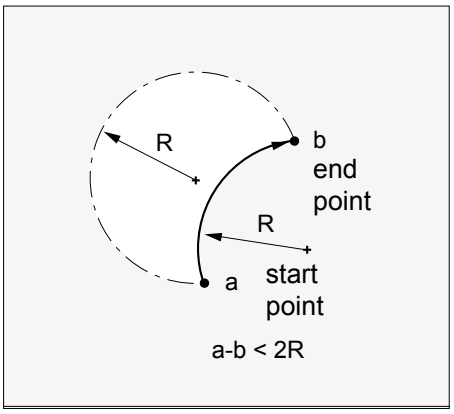


5.6.3.2 Programming a circle by its radius (R..)



The system chooses the path whose angle is less than 180 degrees (a path with an angle greater than 180 degrees can only be obtained by programming the circle by the coordinates of its center or in PGP).

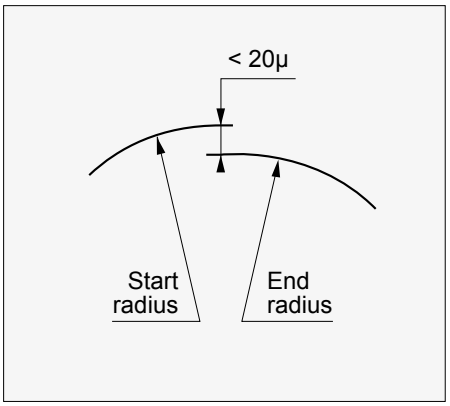
If the distance between the start point and end point is greater than twice the programmed radius, the system generates an error message.



5.6.3.3 Programming a circle by the coordinates of its center (I and J, plane G17)



When the difference of radii between the start point and end point is greater than 20 µ, the system generates an error message.



5.6.3.4 Programming the additional axes and carrier/carried axis pairs

Circular interpolation can be carried out in the plane selected on the following primary and secondary axis pairs:

Function	Main interpolation plane	Axis pairs
G17	XY	XY, UV, UY, XV
G18	ZX	ZX, WU, WX, ZU
G19	YZ	YZ, VW, YW, VZ

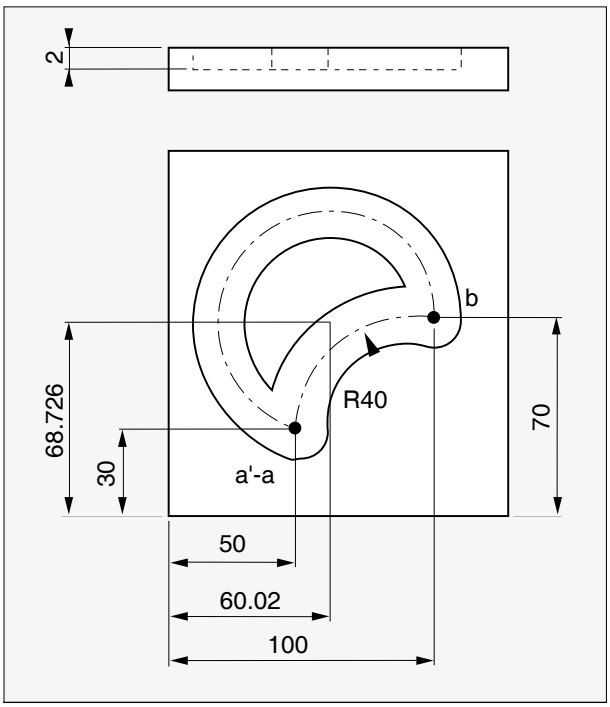
Other axes such as rotary and/or linear axes can be associated in a circular interpolation block, but these axes are interpolated linearly.

A movement on a carrier axis causes an identical movement on the carried axis.

Example

Circular interpolation using two methods

Circular interpolation G02 is carried out by programming the radius (R) and circular interpolation G03 is carried out by programming the circle center (with I and J).




```
%23
N10 G00 G52 Z..                    Rapid to tool change
N20 ...                            Tool call
N30 S300 M40 M03
N40 X50 Y30 Z2                     Approach point a
N50 G01 Z-2 F50                    Penetration in Z
N60 G02 X100 Y70 R40 F150          Point b
N70 G03 X50 Y30 I60.02 J68.726     Point a'
N80 G00 G52 Z.. M05
N90 M02
```

5.6.4 Polar programming



Polar programming is used to define paths or positions when one of the part dimensions is angular.

General

Polar programming is used for line or circle geometric elements defined in the same plane.

Polar programming:
- can coexist with ISO programming and Profile Geometry Programming (PGP)
- can be in absolute dimensions (G90), incremental dimensions (G91) or a combination thereof (G90 and G91, possibly in the same block)
- can be combined with Cartesian programming.

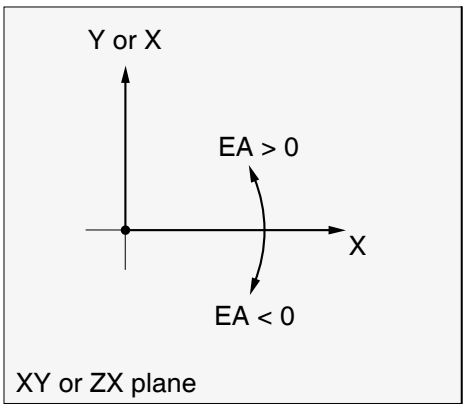
Polar programming can be carried out in one of the interpolation planes selected: XY, ZX or YZ (in all cases, the line or circle start point must be defined in the same plane as the one used in programming).

Reference Axis of the Polar Angle

For polar programming of a line or circle, the polar angle is always programmed by argument EA (whatever the interpolation plane selected).

The polar angle is defined with respect to the reference axis of the interpolation plane:
- X in the XY plane (G17)
- Z in the ZX plane (G18)
- Y in the YZ plane (G19).

The positive or negative direction of the angle is defined in the trigonometric circle.

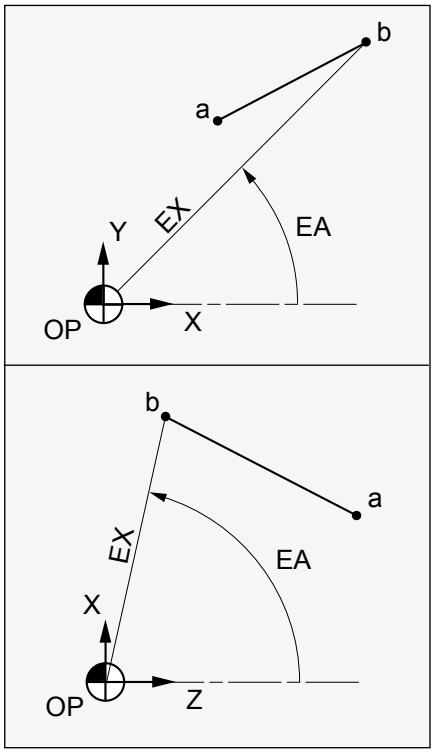


5.6.4.1 Polar programming of a line

A line can be programmed:
- in absolute dimensions with function G90
- in incremental dimensions with function G91.

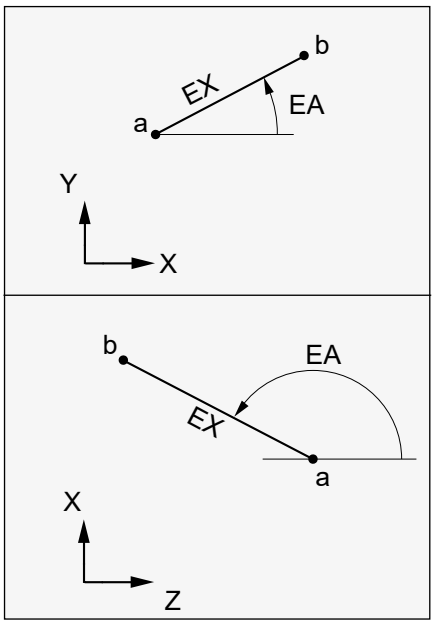
Polar Programming of a Line in Absolute Dimensions (G90)

The line is defined in absolute dimensions by:
- its start point (a) contained in the block preceding the polar programming block
- the polar coordinates of its end point (b) defined with respect to the program origin (OP).



Polar Programming of a Line in Incremental Dimensions (G91)

The line is defined in incremental dimensions by:
- its start point (a) contained in the block preceding the polar programming block
- the polar coordinates of its end point (b) defined incrementally with respect to its start point (or the last programmed point).



Syntax (XY plane)

N.. [G17] [G90/G91] G00/G01 EA.. EX.. [F..]

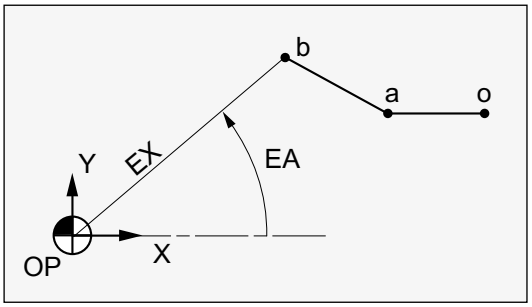
G17	XY plane selected.
G90/G91	Programming in absolute or incremental dimensions.
G00/G01	Linear interpolation.
EA..	Angle of line EX.
EX..	Length of the line. In G90: EX = distance from program origin to end point. In G91: EX = distance from start point to end point.
F..	Feed rate (see chapter 5.10).

In the block, EA must be programmed first, before EX.
Argument EX:
- is always addressed by the same letters (i.e. EX), whatever the interpolation plane;
- must always be programmed as a positive value.

Example

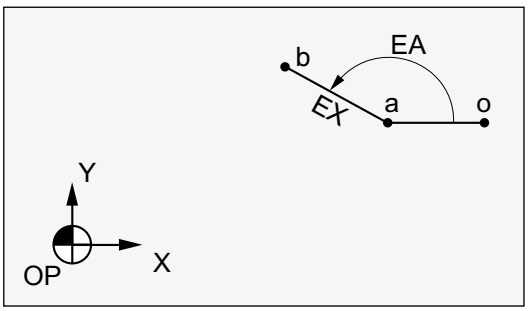
Line programmed in absolute dimensions

```
N..
N.. G90 G17
N200 X60 Y10 (Point o)
N210 G01 X40 Y10
N220 EA30 EX35
N..
```



Line programmed in incremental dimensions example

```
N..
N.. G90 G17
N120 X60 Y10 (Point o)
N130 G01 X40 Y10
N130 G91 EA120 EX15
N..
```

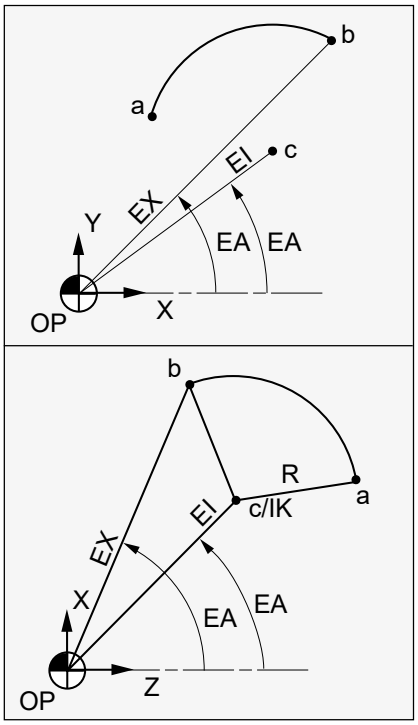


5.6.4.2 Polar programming of a circle

A circle can be programmed:
- in absolute dimensions with function G90 or incremental dimensions with function G91,
- as a combination of Cartesian and polar coordinates,
- as a combination of incremental and absolute dimensions (G90/G91) and Cartesian and polar coordinates,
- by its arc angle and its center defined in Cartesian or polar coordinates.

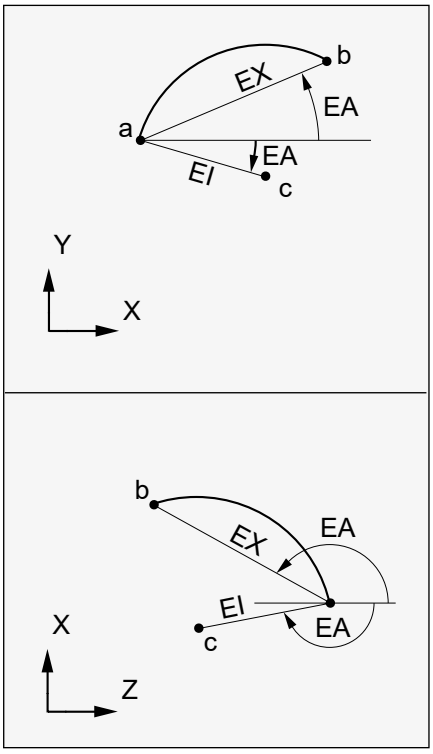
Polar Programming of a Circle in Absolute Dimensions (G90)

The circle is defined in absolute dimensions by:
- its start point (a) contained in the block preceding the polar programming block,
- the polar coordinates of the end point (b) and the center (c) defined in absolute dimensions with respect to the program origin (OP).



Polar Programming of a circle in incremental dimensions (G91)

The circle is defined in incremental dimensions by:
- its start point (a) contained in the block preceding the polar programming block;
- the polar coordinates of the end point (b) and the center (c) defined in incremental dimensions with respect to the circle start point (or last point programmed).



Syntax (XY plane)

N.. [G17] [G90/G91] G02/G03 EA.. EX.. EA.. EI.. [F..]

G17	XY plane selected.
G90/G91	Programming in absolute or incremental dimensions.
G02/G03	Circular interpolation.
EA..	Angle of line EX.
EX..	Length of the line. In G90: distance from program origin to end point. In G91: distance from start point to end point.
EA..	Angle of line EI.
EI..	Length of the line. In G90: EI = distance from program origin to circle center. In G91: EI = distance from start point to circle center.
F..	Feed rate (see chapter 5.10).

The following notes apply to all cases of circles programmed in absolute or incremental dimensions.
In a block, the programming order must be:

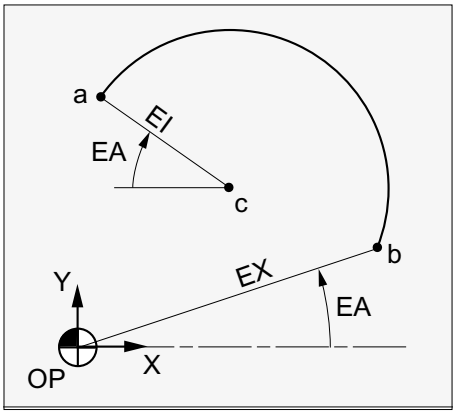
- End point EA then EX
- Center point EA then EI.

Arguments EX and EI must always be programmed in the positive direction.
Arguments EX and EI are addressed by these letters whatever the interpolation plane chosen.

Examples

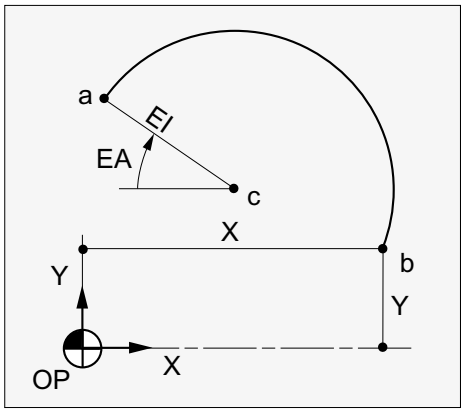
End point in polar absolute center in polar incremental

```
N.. G90 G17 G01 Xa Ya
N.. G02 EAb EXb G91 EAc EIc
```



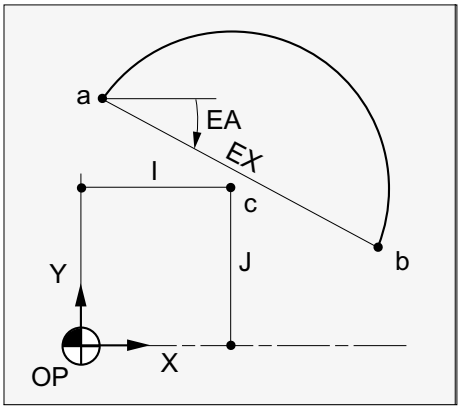
End point in Cartesian absolute center in polar incremental

```
N.. G90 G17 G01 Xa Ya
N.. G02 Xb Yb G91 EAc EIc
```



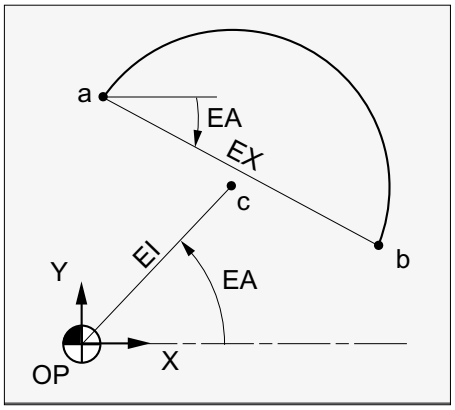
End point in polar Incremental center in Cartesian absolute

```
N.. G90 G17 G01 Xa Ya
N.. G91 G02 EAb EXb G90 Ic Jc
```



End point polar incremental center in polar absolute

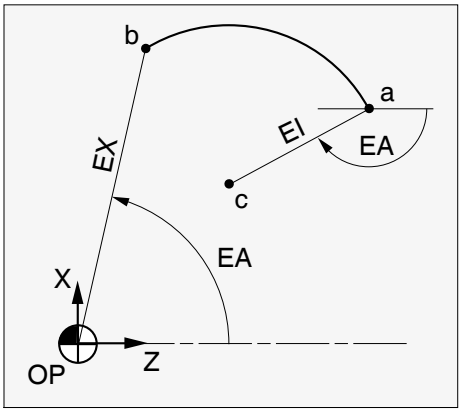
```
N.. G90 G17 G01 Xa Ya
N.. G91 G02 EAb EXb G90 EAc EIc
```



Examples

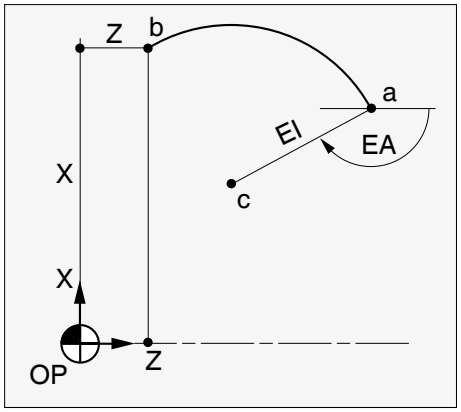
End point in polar absolute center in polar incremental

```
N.. G90 G20 G01 Xa Za
N.. G03 EAb EXb G91 EAc EIc
```



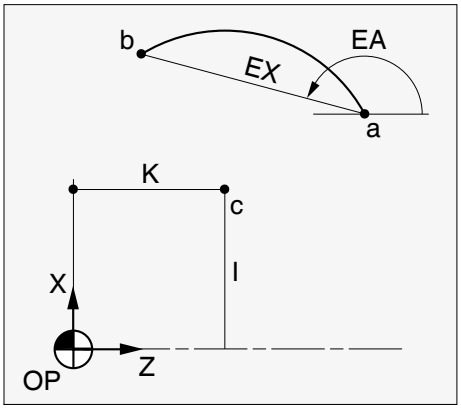
End point in Cartesian absolute center in polar incremental

```
N.. G90 G20 G01 Xa Za
N.. G03 Xb Zb G91 EAc EIc
```



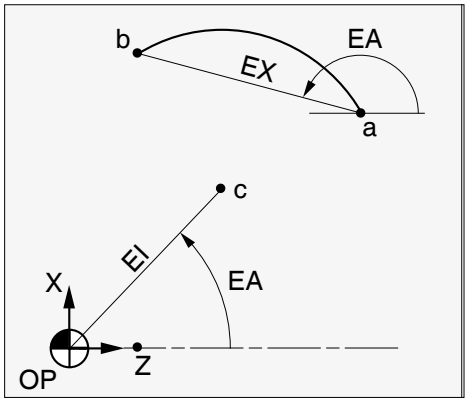
End point in polar incremental center in Cartesian absolute

```
N.. G90 G20 G01 Xa Za
N.. G91 G03 EAb EXb G90 Ic Kc
```



End point polar incremental center in polar absolute

```
N.. G90 G20 G01 Xa Za
N.. G91 G03 EAb EXb G90 EAc EIc
```



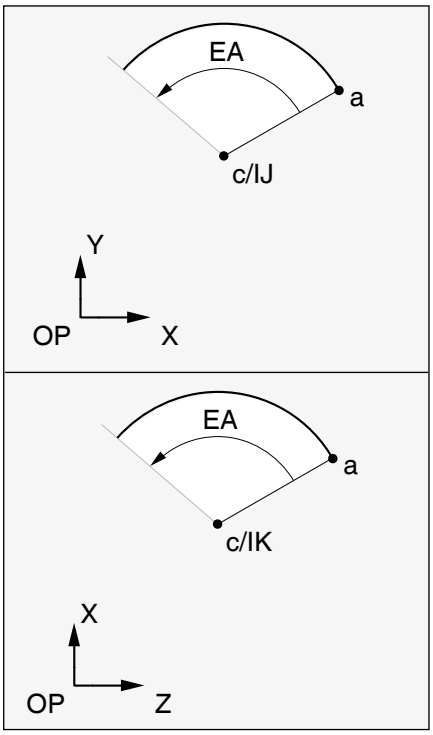
5.6.4.3 Defining a circle by the arc angle

Defining a circle by the arc angle and Cartesian programming of its center defined in absolute or incremental dimensions

The circle is defined by:
- its start point (a) contained in the block preceding the arc angle programming block,
- the Cartesian coordinates of its center (c) and its arc angle.

The center can be programmed in:
- absolute dimensions with G90,
- incremental dimensions with G91.

The arc angle EA is defined in absolute dimensions.



Syntax (XY plane)

N.. [G17] [G90/G91] G02/G03 I.. J.. EA.. [F..]

G17	XY plane selected.
G90/G91	Programming of the circle center in absolute or incremental dimensions.
G02/G03	Circular interpolation.
I.. J..	Cartesian coordinates of the circle center in the XY plane (I along X and J along Y): - in G90 with reference to the program origin. - in G91 with reference to the circle start point.
EA..	Arc angle. Angle of the end point with respect to the start point.
F..	Feed rate (see chapter 5.10).

Notes

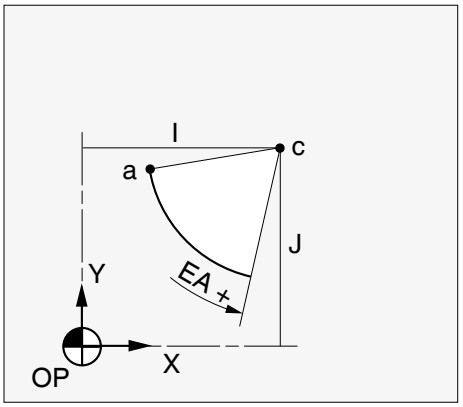
The notes below concern only circles defined by the arc angle (with the center defined in Cartesian coordinates).

When a value of zero is assigned to EA, the system describes a complete circle.
When the circle is defined from X to Y, EA is positive; in the opposite direction, EA is negative.

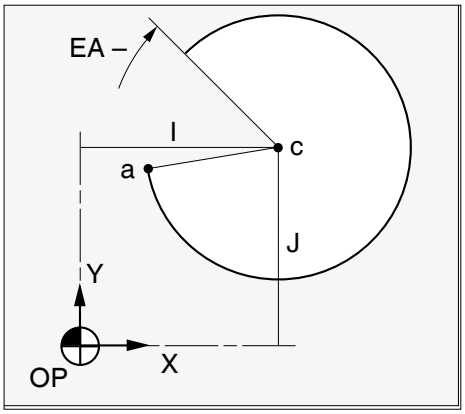
Four types of circles are possible in absolute programming, depending on the direction of the programmed circular interpolation (G02 or G03) and the sign (positive or negative) of arc angle EA.

Examples

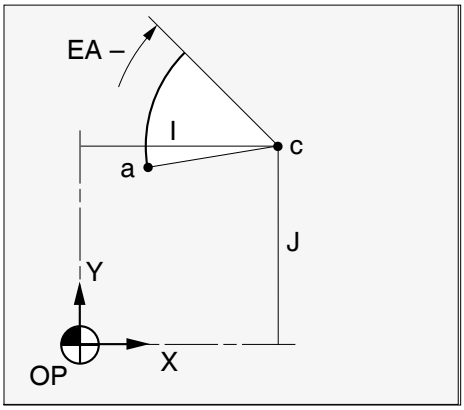
```
N.. G90 G17 Xa Ya
N.. G03 I.. J.. EA+75
```



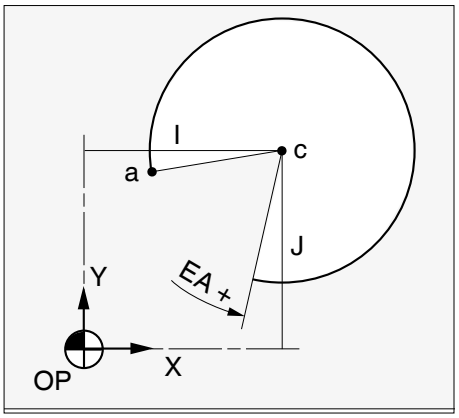

```
N.. G90 G17 Xa Ya
N.. G03 I.. J.. EA-75 (or EA+285)
```




```
N.. G90 G17 Xa Ya
N.. G02 I.. J.. EA-75
```




```
N.. G90 G17 Xa Ya
N.. G02 I.. J.. EA+75 (or EA-285)
```



Syntax (ZY plane)

N.. [G20] [G90/G91] G02/G03 I.. K.. EA.. [F..]

G20	ZX plane selection.
G90/G91	Programming of the circle center in absolute or incremental dimensions.
G02/G03	Circular interpolation.
I.. K..	Cartesian coordinates of the circle center in the ZY plane (I along X and K along Z): - in G90 with reference to the program origin. - in G91 with reference to the circle start point.
EA..	Arc angle. Angle of the end point with respect to the start point.
F..	Feed rate (see chapter 5.10).

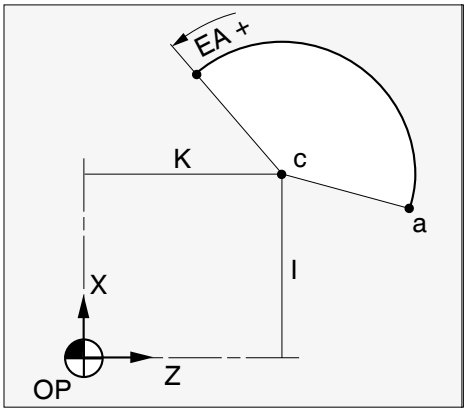
Notes

The notes below concern only circles defined by the arc angle (with the center defined in Cartesian coordinates).
When a value of zero is assigned to EA, the system describes a complete circle.
When the circle is defined from Z to X, EA is positive; in the opposite direction, EA is negative.

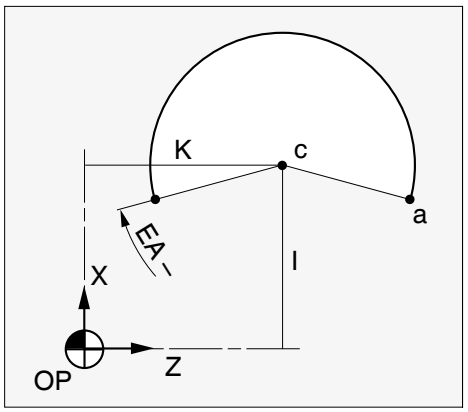
Four types of circles are possible in absolute programming, depending on the direction of the programmed circular interpolation (G02 or G03) and the sign (positive or negative) of arc angle EA.

Examples

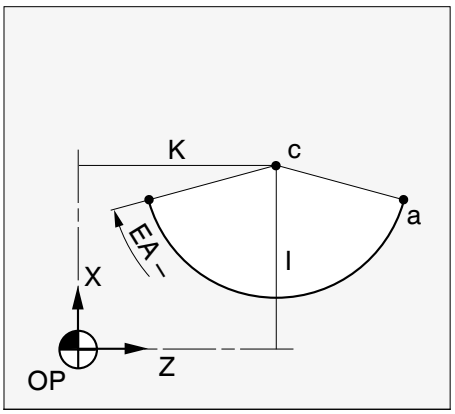
```
N.. G90 G20 Xa Za
N.. G03 I.. K.. EA+160
```



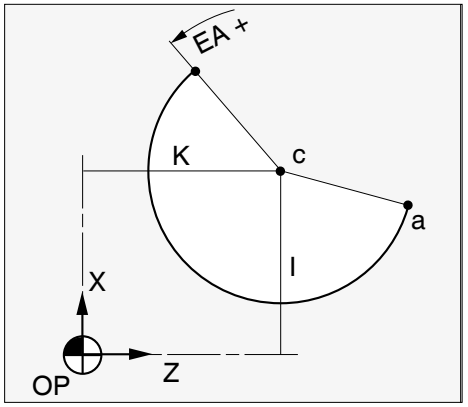

```
N.. G90 G20 Xa Za
N.. G03 I.. K.. EA-160 (or EA+200)
```




```
N.. G90 G20 Xa Za
N.. G02 I.. K.. EA-160
```



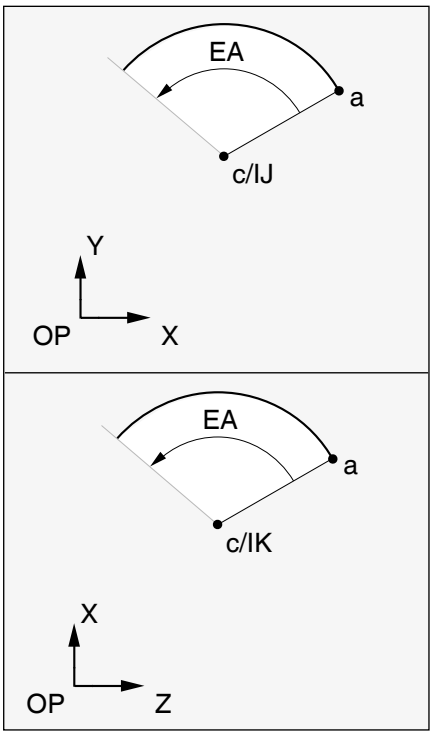

```
N.. G90 G20 Xa Za
N.. G02 I.. K.. EA+160 (ou EA-200)
```



Defining a circle by its arc angle and polar programming of its center in absolute coordinates (G90)

The circle is defined by:
- its start point (a) contained in the block preceding the arc angle programming block
- the polar coordinates of its center (c) and its arc angle.

Arc angle EA is defined in absolute dimensions.



Syntax (XY plane)

N.. [G17] [G90] G02/G03 EA.. EI.. EA.. [F..]

G17	XY plane selection.
G90	Circle center programmed in absolute dimensions.
G02/G03	Circular interpolation.
EA..	Angle of line EI.
EI..	Length of the line. EI = distance from program origin to circle center.
EA..	Arc angle. Angle of the end point with respect to the start point.
F..	Feed rate (see chapter 5.10).

Notes

The notes below concern only circles defined by the arc angle (with the center defined in polar coordinates).

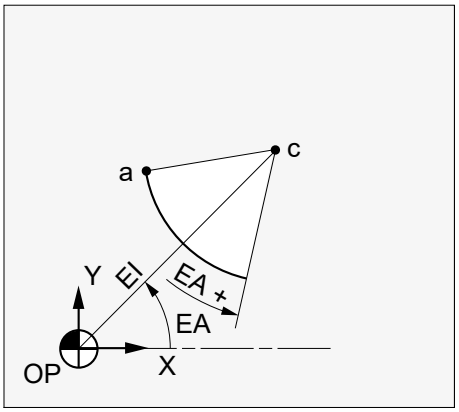
When a value of zero is assigned to EA, the system describes a complete circle.

When the circle is defined from X to Y, EA is positive; in the opposite direction, EA is negative.

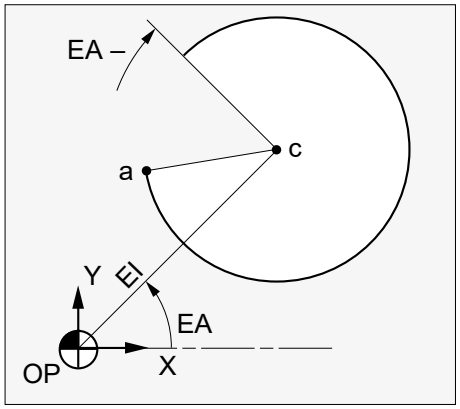
Four types of circles are possible in absolute programming, depending on the direction of the programmed circular interpolation (G02 or G03) and the sign (positive or negative) of arc angle EA.

Example

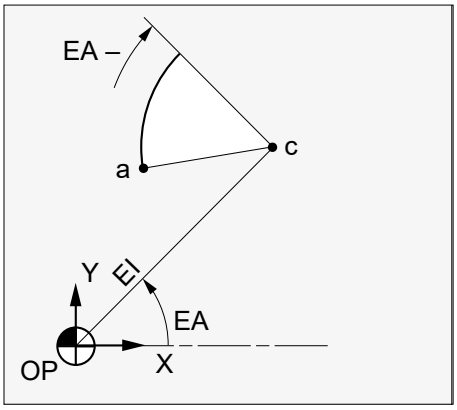
```
N.. G90 G17 Xa Ya
N.. G03 EA.. EI.. EA+70
```



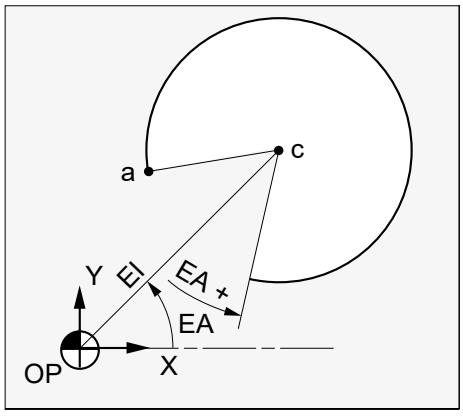

```
N.. G90 G17 Xa Ya
N.. G03 EA.. EI.. EA-70 (or EA+290)
```




```
N.. G90 G17 Xa Ya
N.. G02 EA.. EI.. EA-70
```




```
N.. G90 G17 Xa Ya
N.. G02 EA.. EI.. EA+70 (or EA-290)
```



Syntax

N.. [G20] [G90] G02/G03 EA.. EI.. EA.. [F..]

G20	ZX plane selection.
G90	Circle center programmed in absolute dimensions.
G02/G03	Circular interpolation.
EA..	Angle of line EI. Program origin/circle center.
EI..	Length of the line. EI = distance from program origin to circle center.
EA..	Arc angle. Angle of the end point with respect to the start point.
F..	Feed rate (see chapter 5.10).

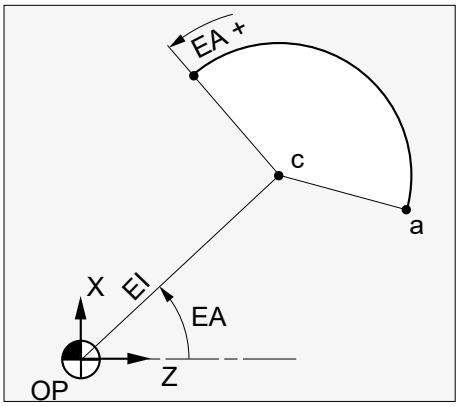
Notes

The notes below concern only circles defined by the arc angle (with the center defined in polar coordinates). When a value of zero is assigned to EA, the system describes a complete circle. When the circle is defined from Z to X, EA is positive; in the opposite direction, EA is negative.

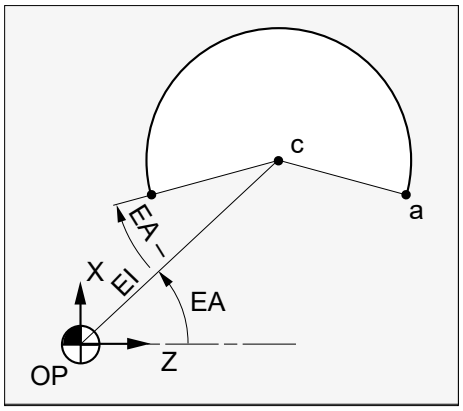
Four types of circles are possible in absolute programming, depending on the direction of the programmed circular interpolation (G02 or G03) and the sign (positive or negative) of arc angle EA.

Example

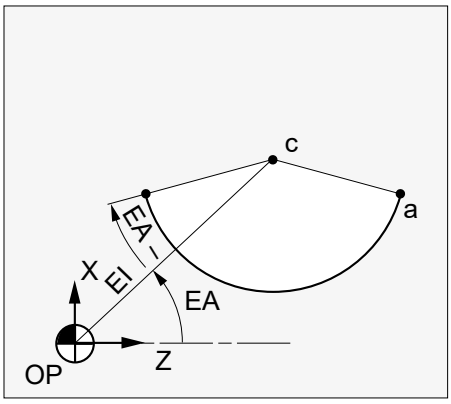
```
N.. G90 G20 Xa Za
N.. G03 EA.. EI.. EA+160
```



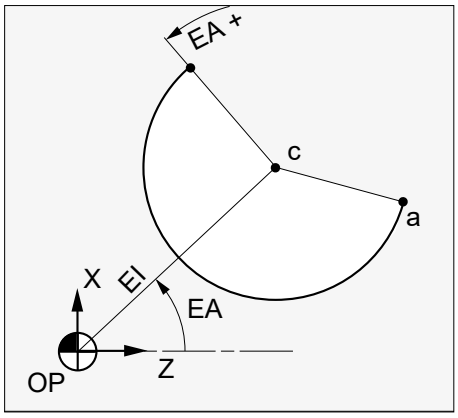

```
N.. G90 G20 Xa Za
N.. G03 EA.. EI.. EA-160 (or EA+200)
```




```
N.. G90 G20 Xa Za
N.. G02 EA.. EI.. EA-160
```




```
N.. G90 G20 Xa Za
N.. G02 EA.. EI.. EA+160 (or EA-200)
```

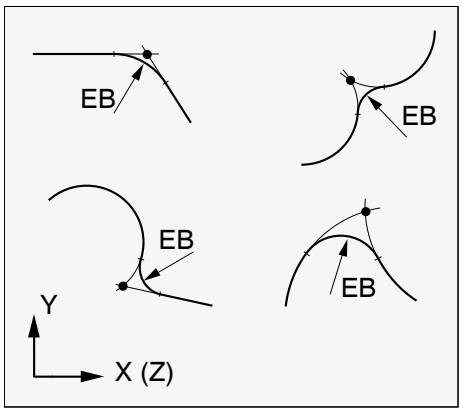


5.6.5 Programming fillets and chamfers

5.6.5.1 Fillet between two interpolations ► EB+..



EB+	Fillet between two interpolations.



This function is used to make a fillet between two linear and/or circular interpolations.

Syntax (XY plane)



N.. G01/G02/G03 X.. Y.. I.. J.. / R.. [F..] **EB+..** [EF..]

G01 / G02 / G03	Linear or circular interpolations.
X.. Y..	Programmed intersection point.
I.. J.. / R..	Circle center or radius in G02 or G03.
F..	Feed rate.
EB+..	Fillet dimension.
EF..	Feed rate specific to the fillet.

Property of the Function

Function EB+.. is nonmodal.

Cancellation

Function EB+ is cancelled at the end of the block.

Syntax (XZ plane)



N.. G01/G02/G03 X.. Z.. I.. K.. / R.. [F..] **EB+..** [EF..]

G01 / G02 / G03	Linear or circular interpolations.
X.. Z..	Programmed intersection point.
I.. K.. / R..	Circle center or radius in G02 or G03.
F..	Feed rate.
EB+..	Fillet dimension.
EF..	Feed rate specific to the fillet.

Property of the Function

Function EB+.. is nonmodal.

Cancellation

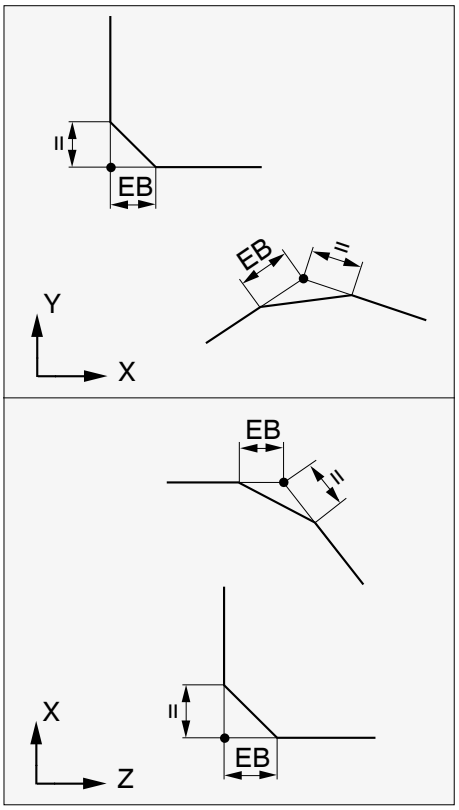
Function EB+ is cancelled at the end of the block.

5.6.5.2 Chamfer between two linear interpolations ► EB-..



EB-	Chamfer between two linear interpolations.

This function is used to make a chamfer between two linear interpolations.



Syntax (XY plane)



N.. G01 X.. Y.. [F..] **EB-**.. [EF..]

G01	Linear interpolations.
X.. Y..	Programmed intersection point.
F..	Feed rate.
EB-..	Chamfer dimension.
EF..	Feed rate specific to the chamfer.

Property of the Function

Function EB-.. is nonmodal.

Cancellation

Function EB- is cancelled at the end of the block.

Syntax (XZ plane)



N.. G01 X.. Z.. [F..] **EB**-.. [EF..]

G01	Linear interpolations.
X.. Z..	Programmed intersection point.
F..	Feed rate.
EB-..	Chamfer dimension.
EF..	Feed rate specific to the chamfer.

Property of the Function

Function EB-.. is nonmodal.

Cancellation

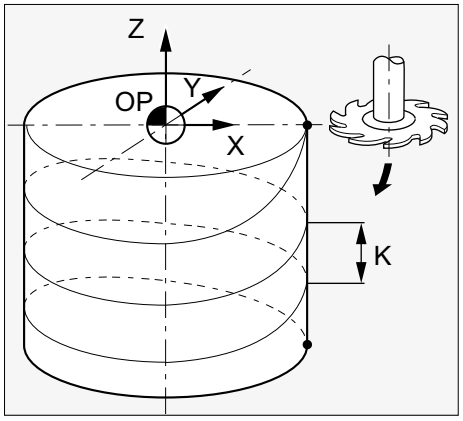
Function EB- is cancelled at the end of the block.

5.6.6 Helical interpolation



Helical interpolation is a combination of circular and linear interpolation. The tool axis describes a helix with a constant pitch.

Helical interpolation can be carried out in all three planes and applies to the primary and secondary axes.



Syntax (XY plane)

N.. [G17] [G90/G91] **G02/G03** X.. Y.. Z.. I.. J.. / R.. K.. [F..] [ER.]

G17	XY plane selection.
G90/G91	Programming in absolute or incremental dimensions.
G02	Clockwise helical interpolation.
G03	Counter-clockwise helical interpolation.
X.. Y..	End point in the XY plane. End point coordinates with G90. Value of the movement with G91.
Z..	Point to be reached on the helix axis (XY plane) with G90 or G91.
I.. J..	Position of the interpolation center in the XY plane (I along X, J along Y), with reference to the program origin with G90 or the interpolation start point with G91.
R..	Radius of the circle to be interpolated.
K..	Approximate helix pitch in Z (unsigned value).
F..	Feed rate in mm/min or inches/min.
ER.	Number of full revolutions in the plane at the end of the helix.

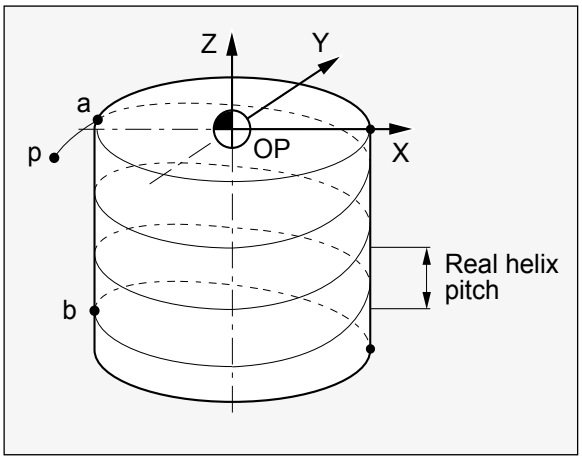
Notes

With helical interpolation:

- The helix pitch applies only to the axis normal to the interpolation plane of the basic reference system.
- Programming of the pitch allows the system to determine the number of revolutions required so that the real pitch is as close as possible to the programmed pitch considering the position of the start and end points.
- K is not required if there is less than one full rotation and must not be programmed.
- Caution if K is programmed the system will check the travels over a complete circle. It may therefore detect a false over travel if the trajectory was less than a full rotation.
- The optional value ER permits to finish the helix by one or more full circles in the interpolation plane.

Example

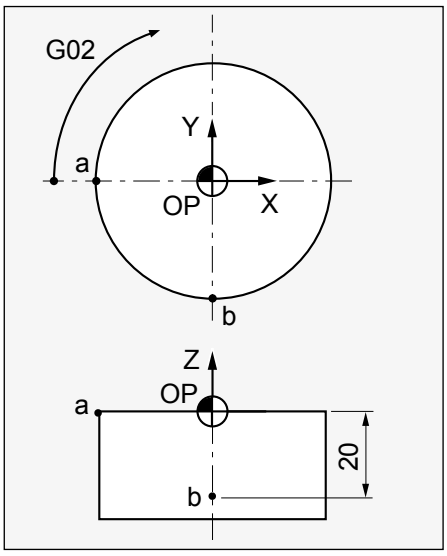
Helical interpolation in the XY plane with programming in absolute dimensions (G90)




```
%55
N10 G00 G52 Z..                          Rapid to tool change
N20 ...                                  Tool call
N30 S200 M40 M03
N40 G00 Xp Yp Zp                         Approach point p
N50 G01 Xa Ya Za                         Start point a
N60 G02 Xb Yb Zb I0 J0 K.. F150          End point
N.. G00 X..
N..
```

5.6.6.1 Determination of the helix depth

Helix depth in G02 with different helix pitches:
Assume difference between a and b in Z: Δ Z = 20 mm.
Δ Z / K = Approximate number of revolutions.



The final X,Y, Z position is always accurate. If the programmed pitch does not match with the axial displacement, the system will recalculate the closest value. No error will be generated.

Syntax according to the plane selected: G17/G18/G19

Interpolation plane	Interpolation	Syntax	Pitch
G17(XY)	G02/G03	X.. Y.. I.. J.. or R..	K
G18 (ZX)	G02/G03	Z.. X.. K.. I.. or R..	J
G19 (YZ)	G02/G03	Y.. Z.. J.. K.. or R..	I

Programming of additional axes and carrier/carried axis pairs

Other axes such as rotary and/or linear axes can be associated in a block, but these axes are interpolated linearly. Carrier/carried axes cannot be programmed simultaneously.

Example

```
N..
N60 G17
N70 G02 X.. V.. W.. A.. I.. J.. K..
N..
```

- X and V linear axes interpolated circularly
- Linear axis W and rotary axis A interpolated linearly
- Helix pitch K applied to the W axis.

Example

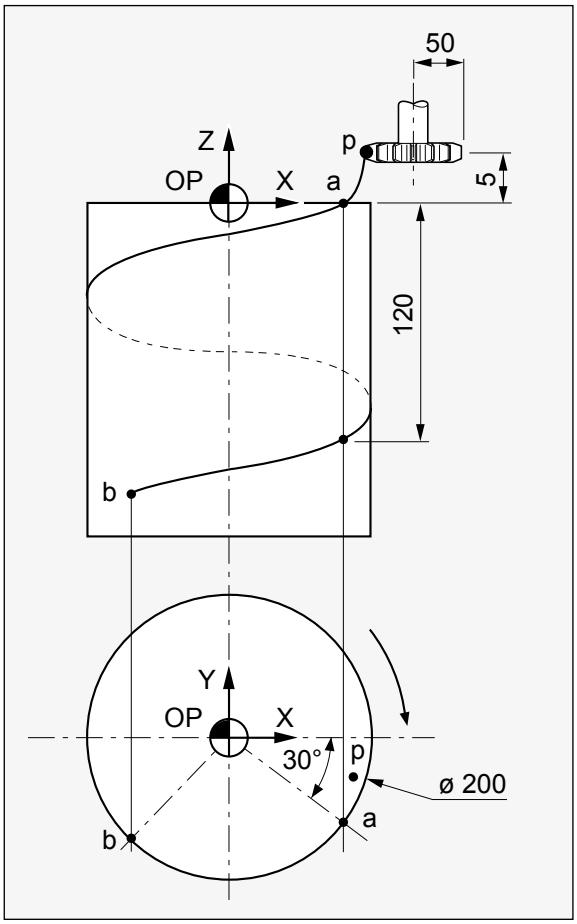
```
N..
N80 G18
N90 G03 X.. Y.. U.. V.. W.. I.. J..
N..
```

- X and W linear axes interpolated circularly
- Y, U, V linear axes interpolated linearly
- Helix pitch J applied to the Y axis.

This type of programming is only possible when the X, U, Y, V axes are independent.

Example

Helical interpolation in the XY plane



Execution of a helix on a cylindrical part with a diameter of 200 mm (radius = 100):

- Helix depth = 2 mm
- Real helix pitch = 120 mm
- Positioning clearance in point p on Z = 5 mm
- Start point a: Position Z = 0, angular position = -30 degrees
- End point b: Position Z = -155

Angular distance from a to b: 360° x 155 / 120 = 465° (i.e. one revolution and 105°)

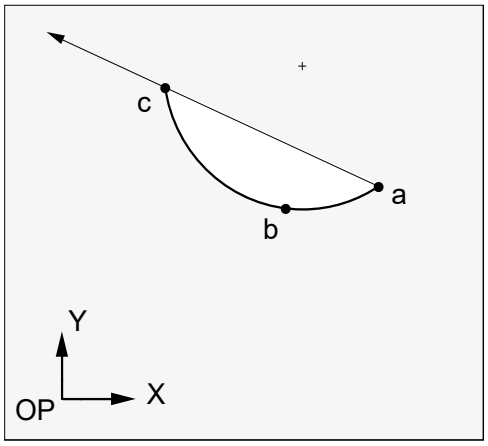
Distance from end point b to point a: -105° + (-30°) = -135°

Position of point p: 360° x 5 / 120 = 15° i.e. p = 30° + 15° = -15°

```
%58
N10 G00 G52 Z..                          Rapid to tool change
N20 ...                                  Tool call
N30 S200 M40 M03
N40 G00 X110 Y110 Z5                     Approach position
N50 G01 G41 X94.661 Y25.364              Point p, (start) in G41 (see chapter 5.11)
N60 G02 X-69.296 Y-69.296 Z-155 I0 J0    Point b, (end)
K120 F300
N70 G01 G40 X-110 Y-110
N..
```

5.6.6.2 Circular interpolation defined by three points ► G23

G23	Circular interpolation defined by three points.



Circular interpolation can be carried out by programming:

- the start point (defined in the block preceding function G23),
- the end point and the intermediate point (defined in the same block as function G23).

The direction of circular interpolation is defined by the position of the intermediate point (b) with respect to the start point (a) and the end point (c), i.e.:

- to the left of line ac: clockwise,
- to the right of line ac: counter-clockwise.

Syntax (XY plane)

N.. [G90/G91] **G23** X.. Y.. I.. J.. [F..]

G90/G91	Absolute or incremental programming.
G23	Clockwise or counter-clockwise circular interpolation.
X.. Y..	End point.
I.. J..	Intermediate point.
F..	Feed rate.

Properties of the Function

Function G23 is nonmodal.

Cancellation

Function G23 is cancelled at the end of the block.

Notes

The arguments of function G23 must not be separated by any other address. Otherwise, the system returns error message 101.

G23 is not available in G21 status (Cartesian to polar interpolation).

Example:

```
N.. G23 X.. Y.. F.. I.. J..
```
Programming incorrect

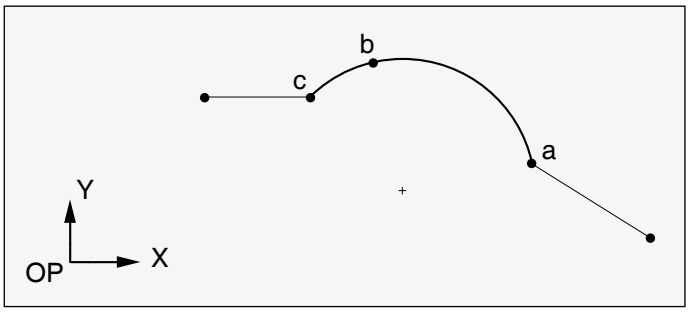
Circular interpolation defined by G23 can be in absolute dimensions (G90) or incremental dimensions (G91).

Syntax according to the plane selected: G17/G18/G19

Main interpolation plane	Function	Intermediate point	Syntax
XY	G17	I J	G23 X.. Y.. I.. J..
ZX	G18	K I	G23 Z.. X.. K.. I..
YZ	G19	J K	G23 Y.. Z.. J.. K..

Example

From A to C through B in G17




```
N..
N130 Z..                          Position on Z
N140 G01 Xa Ya F150               Point a, approach
N150 G23 Xc Yc Ib Jb F100         Circular interpolation
N160 G01 X.. Y.. F150
N..
```

5.6.7 Cylindrical grinders specificity

General



On a cylindrical grinding machine, the "inclined axis" or "inclined wheel" state can be declared by programming the corresponding external parameter E :
- parameter E69001 for the "inclined axis" state,
- parameter E 69002 for the "inclined wheel" state.

In both cases, the angle of inclination declared:
- can be positive or negative,
- must be between -80 and +80 degrees (or error message 94 is returned),
- is expressed in millionths of a degree.

It should be noted that the "inclined axis" and "inclined wheel" states cannot be declared together in a channel (otherwise error message 95 is returned) and that the angle of inclination is saved after a reset.

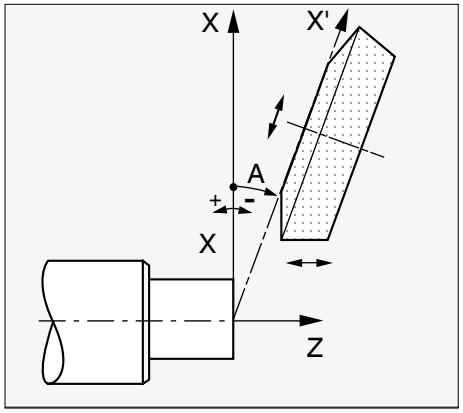
Inclined axis state

Angle Declaration in the "Inclined Axis" State

The inclination angle can be declared:
- by programming external parameter E69001. Example: E69001 = -20000000 (angle = -20 degrees), or,
- Automatically by PLC.
  (See Machine Operation Manual).

The axis inclination is the value of angle A between the penetration axis along X and the normal to the work piece holder spindle.

In the "inclined axis" state, movements programmed on the axis are parallel to the axis inclination angle.



Check of travels

The travels declared in the machine parameters are relative to the physical axes. With linear or circular interpolation (system in state G20) it is checked that the path does not go beyond the soft limits on the X and Z axes.

Axis movement

When an axis inclination is entered, the following elements can be defined in Cartesian coordinates:
- linear and circular movements (the system performs the conversion of these movements for the servo-system)
- machining parameters such as origin offset, machine travels and tool corrections.

The pivot point of the inclined axis must be referenced to the measurement origin.

Inclined wheel state

Angle Declaration in the "Inclined Wheel" State

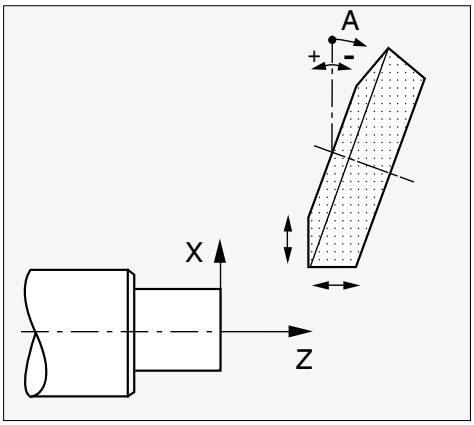
The angle is declared by programming external parameter E69002.

Example:

E69002 = -15000000 (angle = -15 degrees).

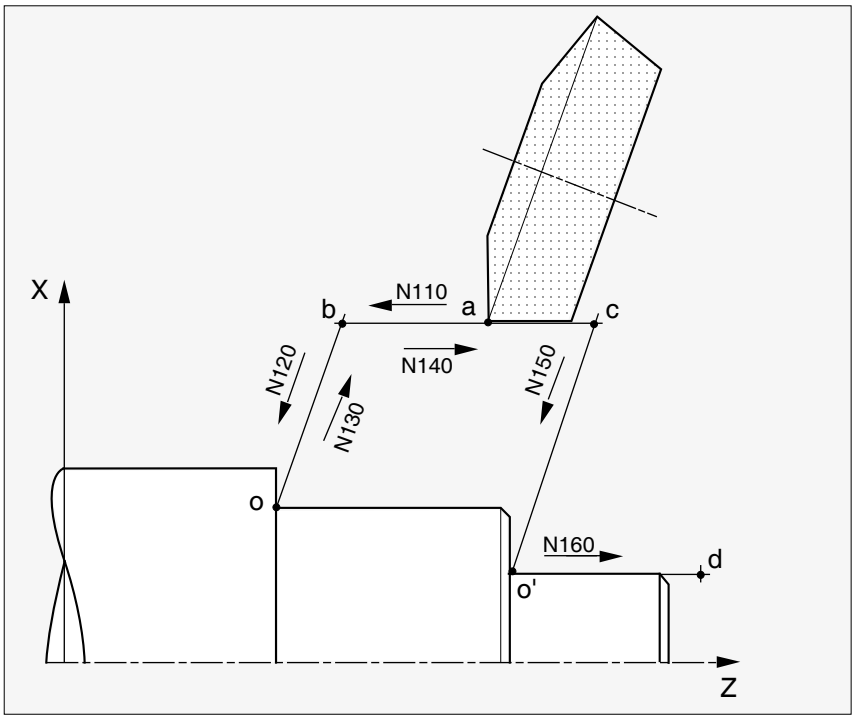
In the "Inclined Wheel" state, wheel movements programmed on X and Z are orthogonal.

Functions G05 and G07 are used in the same way as in the "inclined axis" state (see example below).



Example

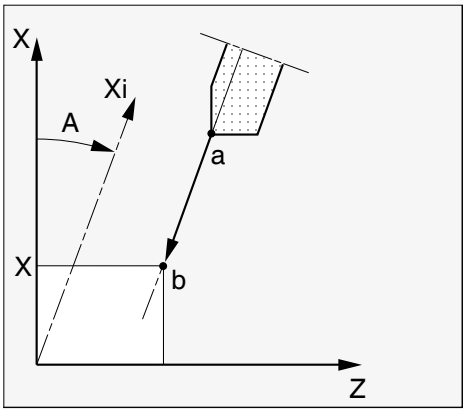
Positioning in Z and X in absolute dimensions (G90) with wheel inclined by -15 degrees.




```
N..
N.. E69002=-15000000          Wheel inclination by parameter E
N.. G97 S300 M03 M40
N.. G95 F0.1
N..
N100 G00 X50 Z50              Point a, position before switching to G07
N110 G07 X20 Z25              Definition of the reference point (o) and positioning in point b
                              on Z
N120 G01 G05 X20F0.2          Point o, approach along X
N130 G05 X50                  Point b, return along X
N140 G07 X10 Z30              Definition of the reference point (o') and positioning in point c
                              on Z
N150 G05 X10                  Point o', approach along X
N160 Z50                      Point d, movement along Z
N..
```

5.6.7.1 Initial tool positioning before machining on an inclined axis ► G07

G07	Initial tool positioning before machining on an inclined axis.



This function defines the location on the Z axis where it intersects with the line parallel to the inclined axis going through the reference point whose position is programmed with the function.

In the figure:
- point a: path start,
- point b: intersection with the line parallel to the inclined axis.

Syntax

N.. [G90] [G00/G01] **G07** X.. Z..

G90	Programming in absolute dimensions (always).
G00/G01	Linear interpolation.
G07	Initial tool positioning before machining on the inclined axis.
X.. Z..	Reference point.

Property of the Function

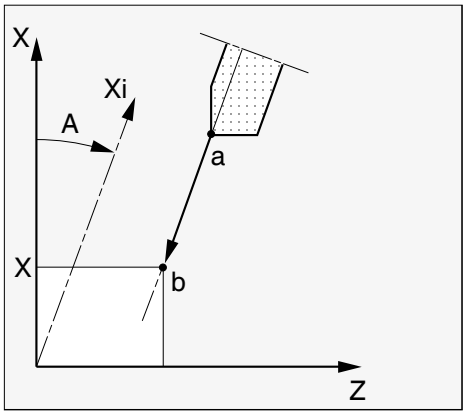
Function G07 is nonmodal.

Cancellation

Function G07 is cancelled at the end of the block.

5.6.7.2 Movement along the inclined axis ► G05

G05	Movement along the inclined axis.



This function defines movement on the X axis parallel to the inclined axis up to the Cartesian end point programmed.

In the figure:
- point a: path start point,
- point b: end point.

Syntax

N.. [G90/G91] [G00/G01] **G05** X..

G90/91	Programming in absolute or incremental dimensions.
G00/G01	Linear interpolation.
G05	Programming of a movement along the inclined axis.
X..	End point.

Property of the Function

Function G05 is nonmodal.

Cancellation

Function G05 is cancelled at the end of the block.

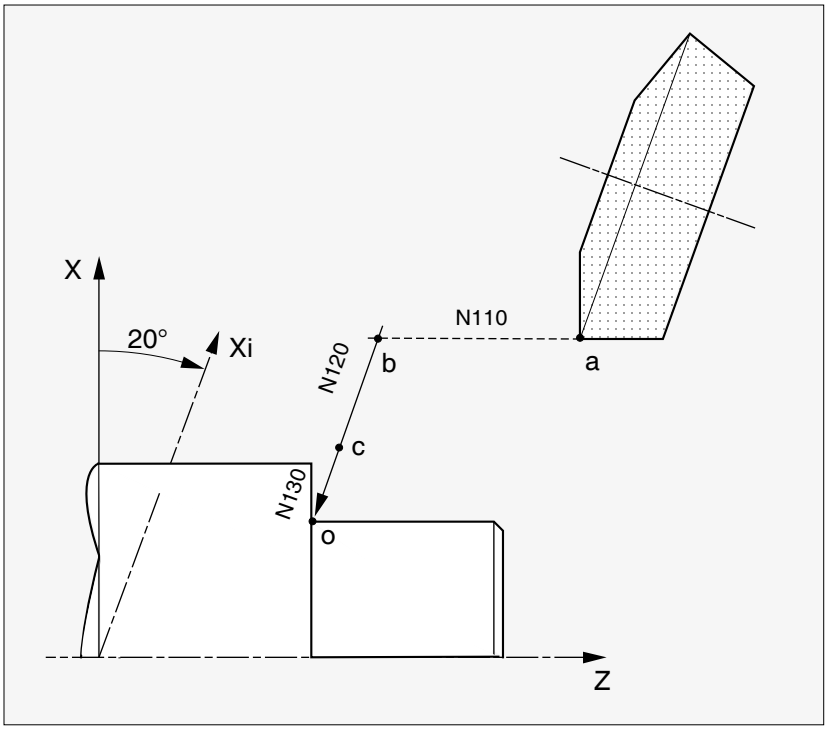
Notes

Function G07 must always be programmed before function G05.

For functions G05 and G07, failure to comply with the following programming rules cause generation of error message 7:
- the functions must be programmed in plane G20,
- the interpolation must be programmed by G00 or G01,
- function G07 must be followed by X and Z,
- function G05 must be followed by X.

Example

Positioning in Z and X in absolute dimensions (G90) with axis inclined by -20 degrees.




```
N..
N.. E69001=-20000000
N.. G97 S300 M03 M40
N.. G95 F0.1
N..
N.. G00 X60 Z70
N110 G07 X10 Z25

N120 G05 X30
N130 G01 G05 X10 F0.2
N..
```

Axis inclination by parameter E

Point a, position before switching to G07
Definition of the reference point (o) and positioning in point b on Z
Point c, positioning on X
Point o, positioning on X

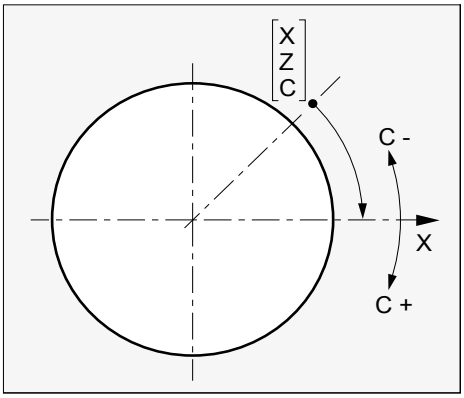
5.6.8 Coordinates systems with C Axis with Turning machine

5.6.8.1 Programming in Polar Coordinates ► G20



G20	Programming in polar coordinates (X, Z, C).

This function is used to program the linear axes X and Z and control a rotary axis C, modulo 360 degrees.



N.. [G40] **G20** [G00/G01][X.. Z.. C..][F..]	
G40	Radius offset cancel.
G20	Programming on axes X, Z and C (angular value C.. with reference to the origin).
G00/G01	Interpolations permitted.
X.. Z.. C..	End point.
F..	Feed rate.

Properties of the Function

Function G20 is modal and is the default function.

Cancellation

Function G20 is cancelled by functions G21 and G22.



Programming of function ER is excluded in G20.

Notes on declaration of the C axis

Control of the C axis requires the use of:
- a servo-drive
- a position encoder
- a special drive arrangement.

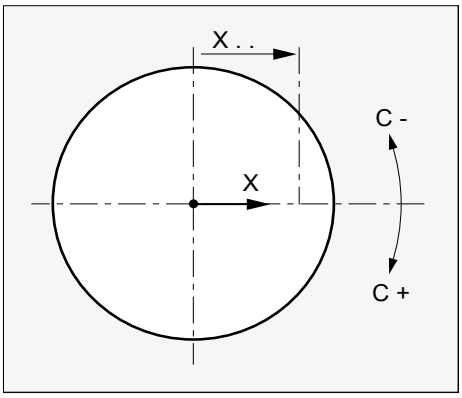
Example

```
N.. ...
N1050 G40 G21 (or G22) ...        Machining in Cartesian or cylindrical coordinates
N..
N1220 G40 G20 G01 X.. Z..         Return to initial state
N..
```

5.6.8.2 Definition of the Start X for Interpolation on the C Axis ► G98

G98	Definition of the start X for interpolation on the C axis.

When the system is in G20 mode, this function is used to compute the feed rate on the C axis in mm/min.



Syntax

N.. **G98** X..

G98	Definition of the start X for interpolation on the C axis.
X..	Start X.

Property of the Function

Function G98 is nonmodal.

Cancellation

Function G98 is cancelled at the end of the block.

Notes

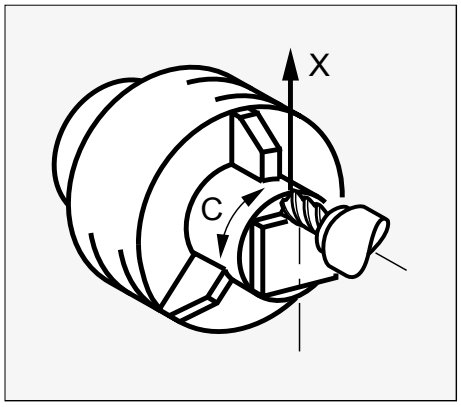
Programming G98 X.. does not cause any movement.

When the function G98 X.. is not programmed, the last value declared with the X axis is used for the computation.

When the C and X axes are interpolated simultaneously, the system performs its feed rate calculation from the mean radius.

5.6.8.3 Programming in cartesian coordinates ► G21

G21	Programming in Cartesian coordinates (X, Y, Z).



The system performs Cartesian/polar coordinate conversion (X-Y converted to X-C).

Interpolation of the X and C axes allows milling in the plane perpendicular to the spindle axis.

The tool is driven by an auxiliary spindle.

Syntax

N.. [G40] **G21** [G00/G01] [G41/G42] [X.. Y.. Z..] [G94 F..]

G40	Radius offset cancel.
G21	Programming in Cartesian coordinates.
G00/G01	Interpolations.
G41/G42	Radius offset to the left or right of the profile.
X.. Y.. Z..	End point.
G94 F..	Feed rate in mm/min.

Property of the Function

Function G21 is modal.

Cancellation

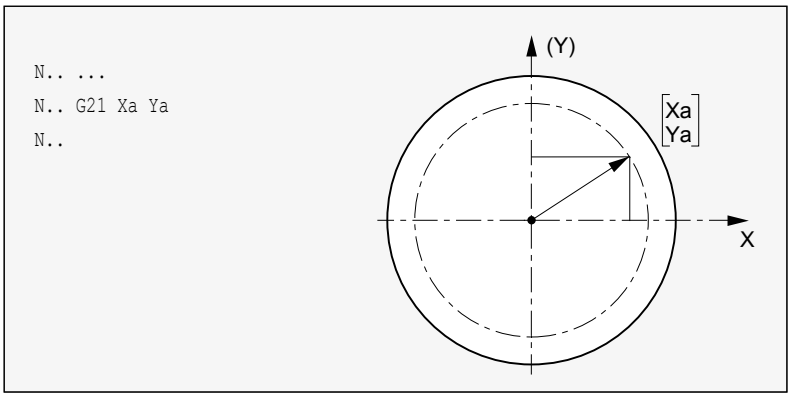
Function G21 is cancelled by function G20.

Programming

Function G21 cannot be programmed during a canned cycle. The use of a cycle causes generation of error message 76.

G21 is used for programming the X, Y and Z addresses by linear interpolation (the Z address is incompatible with circular interpolation G02 and G03).

Programming a point in G21:



During transition from G20 to G21:
- the system must be in G40 mode (radius offset cancel),
- if the last block in G20 includes an X demand, it must have a positive value,
- the system must not be programmed in PGP (Profile Geometry Programming).

During cancellation of G21 by G20, the system must be in G40 mode (radius offset cancel).

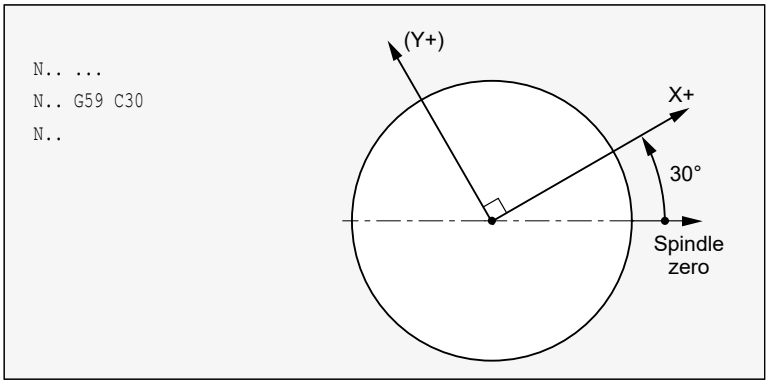
Failure to comply with the transition and cancellation conditions causes generation of error message 75.

Offsets

Programming of offsets is allowed:
- G59 X.. Y.. or G59 C.. : Programmed origin offset,
- ED.. : Programmed angular offset,
- G51.. :Mirroring.

•

An angular offset programmed on the C axis causes a rotation of the X and Y coordinates, e.g.:



Use caution when programming G59 X (programmed offsets)

1. Outside G21, the offset is stored on the real X axis;

2. In G21 the offset is stored on the fictitious X axis

Therefore when programming G59 before enabling G21 and enabling G21 later the offset on the XY fictitious will be zero.

To make sure the programmed offset will be correctly applied on the fictitious axes it is necessary to have G21 enabled before programming G59.

Tool corrections

No change of tool corrections are authorized in G21. The corrections must be programmed and modified in G20.

Feed rate limiting

When the tool is moving toward the center of the part, the constant feed rate along the path is achieved by increasing the angular feed rate on the C axis and decreasing the linear feed rate on the X axis.

The system may therefore have to decrease the tangential machining feed rate so as not to exceed the maximum rotation speed on the C axis when the distance to the center decreases.

However as this feed rate cannot be decreased below 1 inc/scan. the system will issue an error (error34) when the minimum radius is reached.

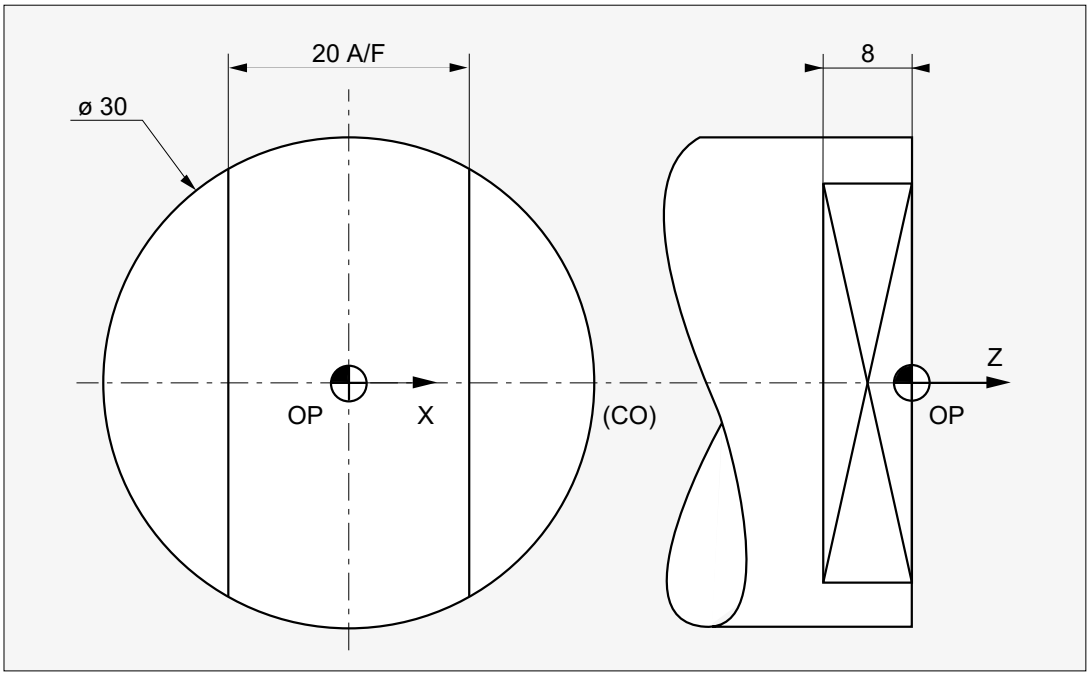
The minimum radius R is calculated by the following equation:

$$R = \frac{3440000}{\text{VC Max} * \text{Sampling period}}$$

Where:
- R is the distance between the cutter center and the C axis center in µm.
- VCmax is the maximum speed on the C axis expressed in degrees/minute (see machine parameter P30).
- Sampling period : Axis scanning period in ms.
- see Flexium+ Parameters Manual M00041 machine parameter P50).

Examples

Execution of two flats

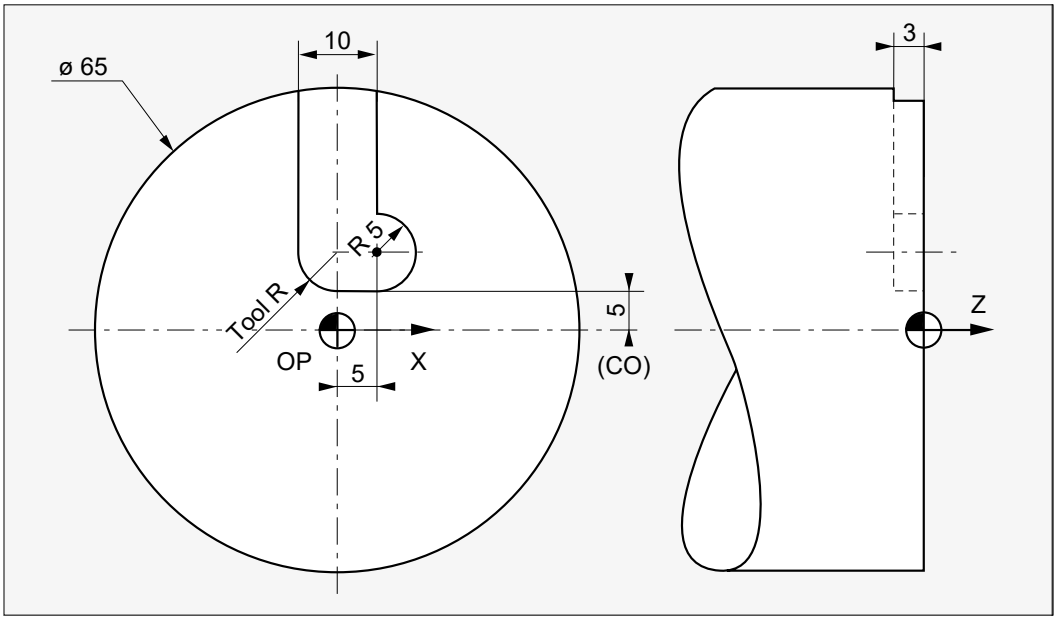



```
N..
N200 G00 G40 G52 X.. Z.. M05
N210 T12 D12 M06 (CUTTER D=10)
N220 G97 S1000 M.. $0 AUXILIARY SPINDLE ROTATION
N230 X50 Z2
N240 G94 F..
```

Tool approach along XZ

```
N250 G21 G01 G42 X10 Y11.18
N260 G00 Z-8                                  Pass setting on Z
N270 G01 Y-11.18                              Execution of flat 1
N280 G00 X-10
N290 G01 Y11.18                               Execution of flat 2
N300 G00 Z3
N310 G40 G20 X100
N320 ...
```

Execution of a shape




```
N..
N200 G00 G40 G52 X.. Z.. M05
N210 T8 D8 M06 (CUTTER D=8)
N220 G97 S1200 M.. $0 AUXILIARY SPINDLE ROTATION
N230 X70 Z2                                   Tool approach along XZ
N240 G94 F..
N250 G92 R1
N260 G21 G00 G41 X-5 Y35
N270 G00 Z-3                                  Pass setting on Z
N280 G01 Y5
N290 X5
N300 G03 X5 Y15 R5
N310 G01 Y35
N320 G00 G40 G20 X200
N330 G52 X0 Z0
N340 ...
```

5.6.8.4 Extended Features of G21

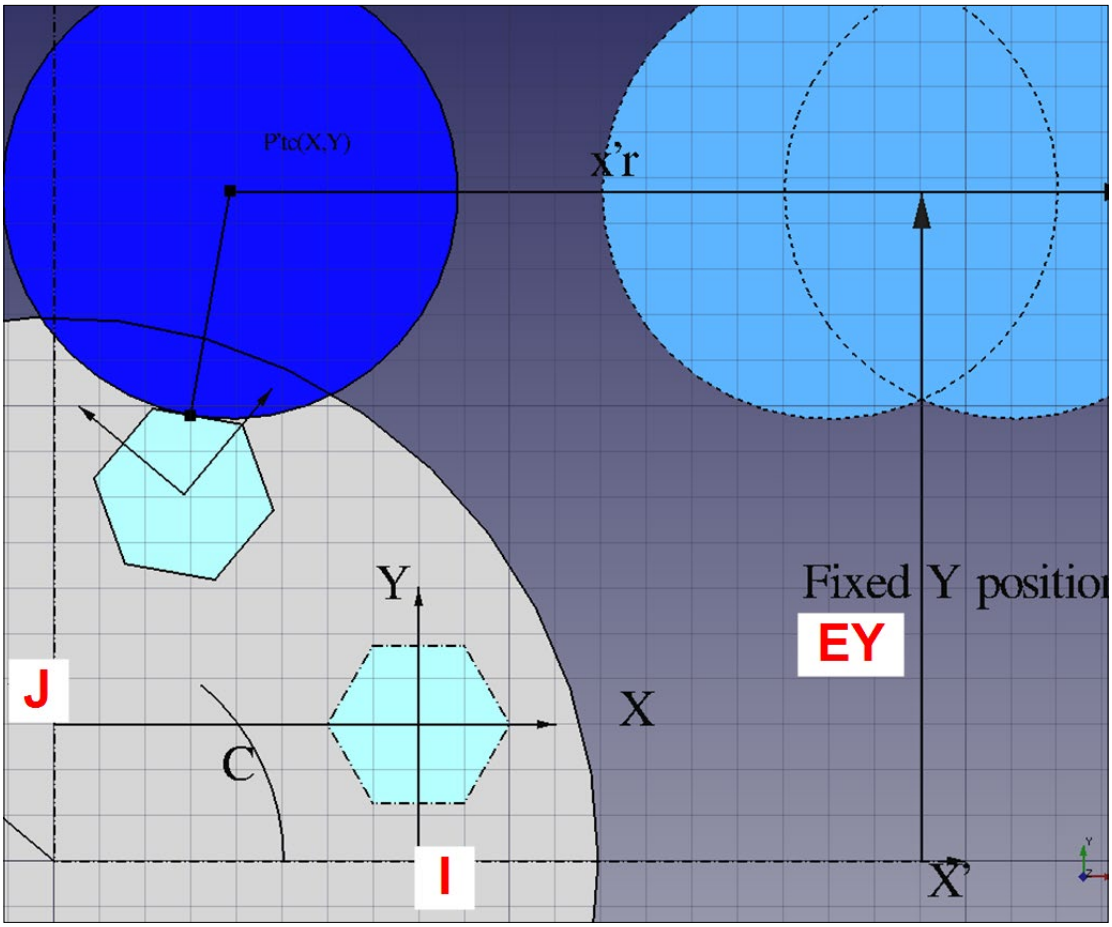
Starting with version 4.2.0 G21 provides the following extended features. For compatibility reasons those features are validated by machine parameter P7 N4 Bit 4.

Non coaxial referential

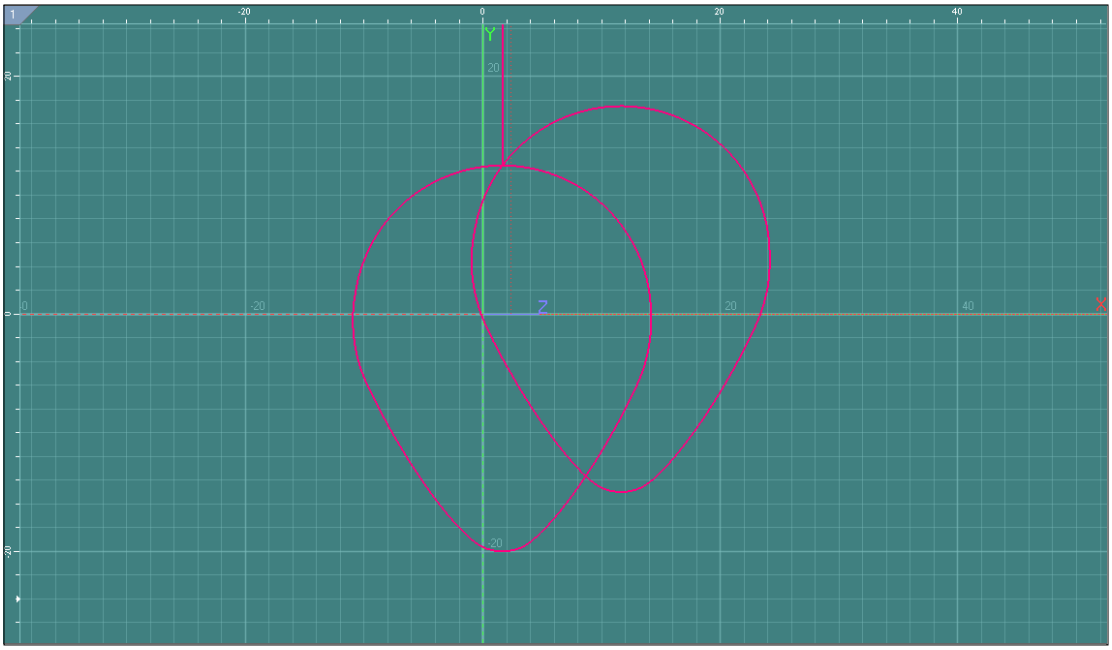
The extend syntax allow programming the Cartesian contour with respect to a local referential instead of the C axis rotation centre. The position of this referential relative to the C axis rotation centre is defined by parameters I and J.

Additionally if the X axis is not aligned with the C axis rotation centre it is possible to define this shift by parameter EY

G21 I… J… EY…



Example xy contour offset using G21



Speed crossing angle

Normally the speed crossing angle is not calculated in G21 therefore the X axis could reverse the speed without deceleration ramp. When extended G21 features are enabled, the speed crossing angle on C and X will be calculated and the block final speed is limited to respect the crossing angle speed.

However, even with extended features enabled the speed crossing angle computation can be disabled by setting the parameter E11032 to 0.

G21 Segmentation

G21 is not a linear transformation therefore some physical principles (like superposition) are not applicable.

G21 Enhanced will split long programmed blocks into small segments so linearity principle can be applied with a negligible error. The segmentation will generate blocks with a Cartesian length (in XY) lower or equal to E32020 (typical value 2 to 3mm).

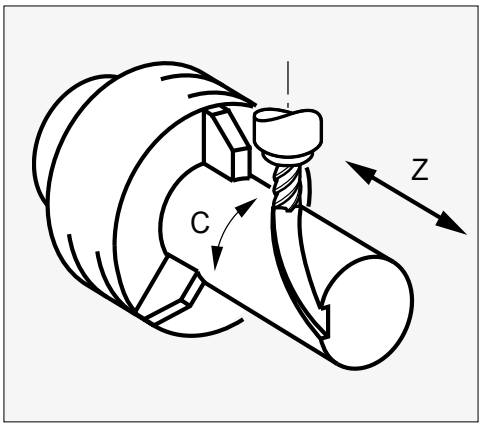
Even with extended features enabled this segmentation can be disabled by setting the parameter E11033 parameter to 0.

Feed depending on radius

In G20 (polar programming) the feed has always been computed by the NCK with respect to the mean radius on C. In order for the G92 feed mode settings to correctly operate, this feature must be inhibited by setting the parameter to 1.

5.6.8.5 Programming in cylindrical coordinates ► G22

G22	Programming in cylindrical coordinates (X, Y, Z).



The system performs cylindrical/polar coordinate conversion (Y-Z converted to Z-C).

Interpolation of the C axis allows milling on the developed cylinder with radius X.

The tool is driven by an auxiliary spindle.

Syntax

N.. [G40] **G22** [G00/G01] [G41/G42] [X.. Y.. Z..] [G94 F..]

G40	Radius offset cancel.
G22	Programming in cylindrical/polar coordinates.
G00/G01	Interpolations.
G41/G42	Radius offset to the left or right of the profile.
X.. Y.. Z..	End point.
G94 F..	Feed rate in mm/min.

Property of the Function

Function G22 is modal.

Cancellation

Function G22 is cancelled by function G20.

Programming

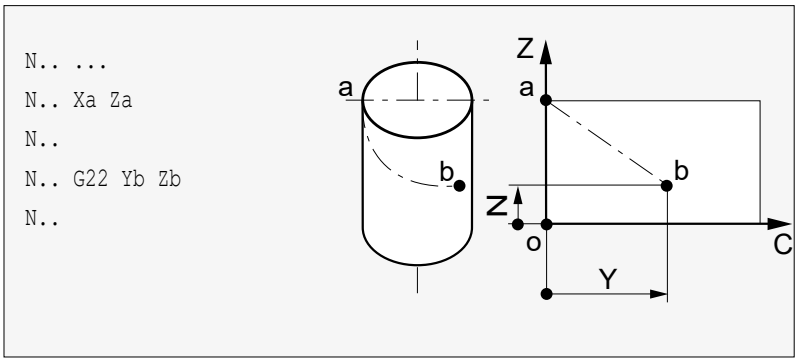
G22 is used for programming addresses:
- X, Y and Z by interpolation G01
- Y, Z, J and K by interpolations G02 and G03.

When the C axis rotates in the positive direction, the Y axis also increases positively.

Position Y0 coincides with position C0 of the C axis.

With G22, canned cycles are not authorized. The use of a canned cycle causes generation of error message 76.

Programming of a point in G22:



During transition from G20 to G22:
- the system must be in G40 mode (radius offset cancel),
- if the last block in G20 included a movement along X, it must be programmed with a positive value,
- the system must not be programmed in PGP (Profile Geometry Programming).

During cancellation of G22 by G20:
- the system must be in G40 mode (radius offset cancel),
- the system must not be programmed in PGP.

Failure to comply with the transition and cancellation conditions causes generation of error message 75.

Offsets

No offsets are allowed in G22 mode.

In G20 mode, origin offsets G59 are possible:

- Before the call to function G22, the start point on the C axis can be modified, e.g. N.. G59 C..
- During the program, the position of the Z axis can be modified, but this requires a return to state G20, e.g. N.. G59 Z..

Tool corrections

No changes of the tool corrections are authorized in G22. The corrections must therefore be programmed and modified in G20.

Feed rate limiting

The feed rate is limited by the maximum feed rate authorized on the C axis and the current value of the positioning radius.

For machining small diameters, the system may have to decrease the feed rate so as not to exceed the maximum rotation speed on the C axis.

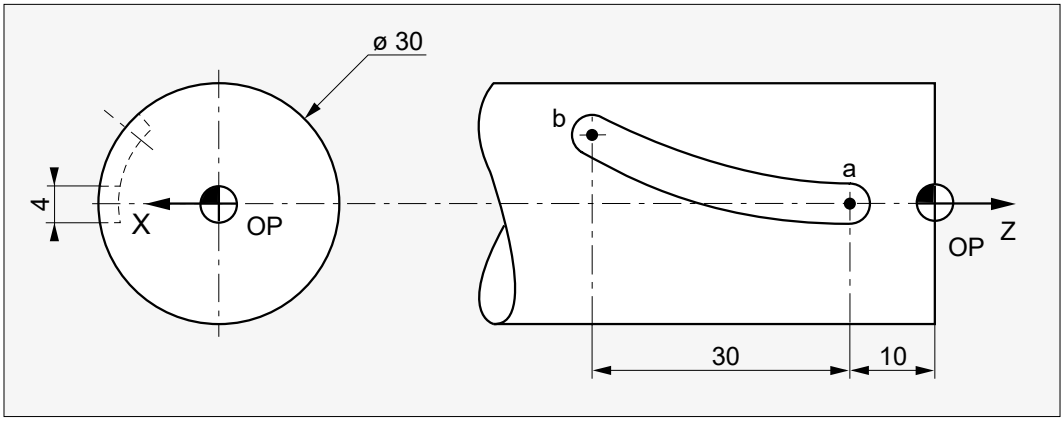
The system automatically limits the minimum feed rate when it is incompatible on the programmed path. When the minimum speed is reached, the system returns error message 34.

Initialization of the Cartesian position on the Y axis

Positioning to the first point programmed in G22 is carried out by computing the function start value on the Y axis from the value on the X axis and on the C axis, interpreted between -180 and +180, giving:
Y = θ . X (θ = angle on the C axis between + and - π).

Example

Application



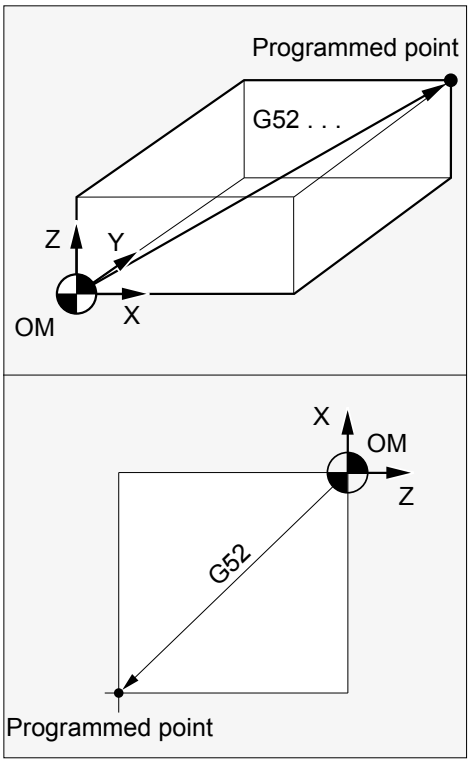

```
N..
N200 G00 G40 G52 X.. Z.. M05
N210 T08 D08 M06 (CUTTER D=4)
N220 G97 S2500 M.. $0 AUXILIARY SPINDLE ROTATION
N230 X32 Z-10                          Point a, tool approach along XZ
N240 C0                                C axis in zero point
N240 G94 F50
N250 G01 X28                           Cut application in X
N260 G22 Y-10 Z-40 F..                 Point b
N270 G20 X32
N280...
```

Homing is required on the C axis.

5.7 Program offsets

5.7.1 Programming of movements in absolute coordinates referenced to the measurement origin ► G52

G52	Programming of movements in absolute coordinates referenced to the measurement origin.



Movements programmed with this function are referenced to the measurement origin (OM). Tool dimensions are not taken into account.

All the axes are programmable with reference to the measurement origin.

Please note that for safety reasons the dynamic overtravels are checked considering tool dimensions.
This means the tool tip remains in the dynamic area defined assuming the mobile also remains in the area defined by the static over travels

Syntax

N.. [G40] [G90] [G00/G01] **G52** X.. Y.. Z.. A.. B.. C.. [F..]

G40	Radius offset cancel.
G90	Programming in absolute dimensions.
G00/G01	Linear interpolations at high or programmed speed.
G52	Programming of movements in absolute coordinates referenced to the measurement origin.
X.. Y.. Z.. **A.. B.. C..**	End point referenced to the measurement origin.
F..	Feed rate.

Property of the Function

Function G52 is nonmodal.

Cancellation

Function G52 is cancelled at the end of the block.

Notes

Programming function G52 in a block suspends the following data:
- tool data,
- DAT1,
- DAT2,
- DAT3,
- program origin offset (G59),
- angular offset (ED..),
- scaling factor (G74).

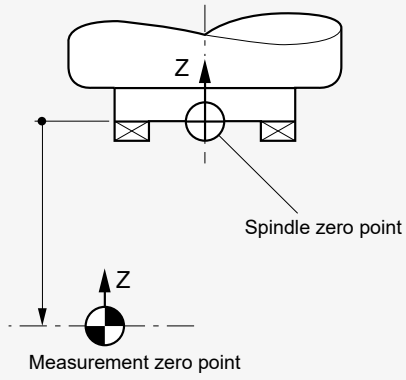
Function G52 must:
- precede programming of the axes in the block,
- be programmed with the system in G40 mode (radius offset cancel) or the system returns error message 27,
- be programmed in absolute coordinates (G90).



Examples

Programming the Z axis with reference to the measurement zero point before a tool change

```
%10
N10 G00 G52 Z0
N20 T03 D03 M06
N..
```



Programming carrier/carried axes pairs

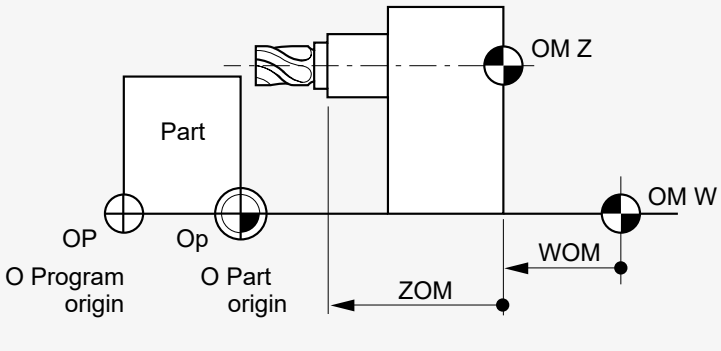
A parallel carrier/carried axis pair can be programmed in G00 and G52.

In all other cases programming a pair of carrier/carried axes simultaneously is prohibited.

Example

DAT1 and DAT2 do not apply to the values programmed with the Z and W axes.

```
N..
N.. G00 G52 Z.. W..
N..
```

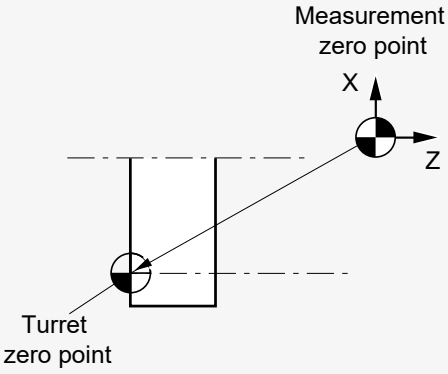




Examples

Programming X and Z axes with reference to the measurement zero point before a tool change

```
%10
N10 G00 G52 X0 Z0
N20 T03 D03 M06
N..
```



Programming of the carrier/carried axes

A parallel carrier/carried axis pair can be programmed in G00 with reference to the measurement origin. In all other cases, programming in G00 is prohibited.

Example

DAT1 and DAT2 do not apply to the values programmed with the Z and W axes.

```
N..
N.. G00 G52 Z.. W..
N..
```

5.7.2 Datum shift (DAT1 and DAT2) cancel ► G53



G53	DAT1 and DAT2 cancel.

This function is used to cancel DAT1 and DAT2 shifts entered on the context *"Work Offset"*.

Syntax

N.. **G53**

G53	DAT1 and DAT2 cancel.

Properties of the Functions

The function G53 is modal.

Cancellation

G53 is cancelled by one of the functions G54.. G58.

Notes

The tool offsets are not affected by function G53.

Programming G53 will also cancel the current G5xx (DAT2). It will be necessary to reprogram the correct DAT2 by G5xx when required.

5.7.3 Selection and activation of datum shift 1 (DAT1) ► G54, G55, G56, G57, G58



G54..G58	DAT1 activated.

These functions are used to select and activate a DAT1 shifts entered on the context *"Work Offset"*.

Syntax

N.. **G54 | G55 | G56 | G57 | G58**

G54.. G58	DAT1 activated.

Properties of the Functions

These functions are modal.

Cancellation

These functions are cancelled by another one of the same range or by G53.



After a cancellation (G53) reprogramming of G54..G58 will NOT reactivate the last DAT2 which was programmed.
It is therefore necessary to program as well the G5xx (501 to 599) required.

5.7.4 Selection and activation of datum shift 2 (DAT2) ► G501.. G599



G501..G599	DAT2 selection.

These functions are used to select and activate a DAT2 shifts entered on the context *"Work Offset"*.

Syntax

N.. **G501 | G502.. G599**

G501 | G502.. G599	DAT2 selection.

Properties of the Functions

These functions are modal.

Cancellation

These functions are cancelled by another one of the same range or by G53.

Notes

After a cancellation (G53), reprogramming G501..G599 will NOT automatically reactivate the last DAT1 which was programmed.

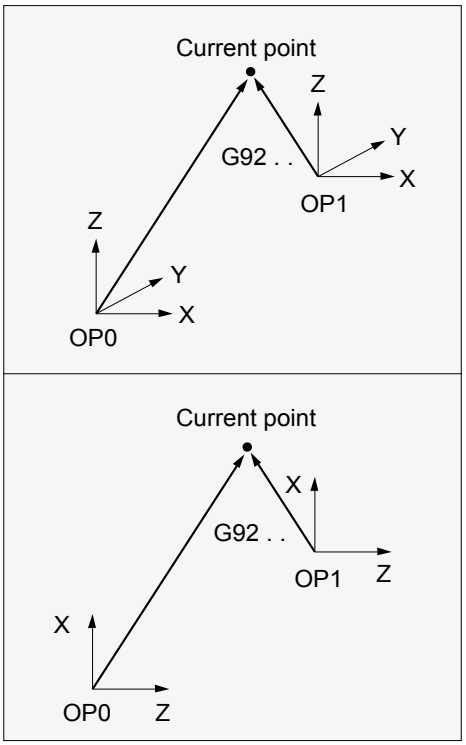
The machine builder may limit the number of DAT2 offsets that can be defined. The default value is 50 therefore G501 to G550. The limit can be different in each channel.

5.7.5 Program origin preset ► G92

G92	Program origin preset.

This function, associated with one or more axes and their values, defines the current point with reference to a new program origin.

DAT1 values are redefined for the programmed axes.



Syntax

N.. **G92** X.. Y.. Z..

G92	Program origin preset.
X.. Y.. Z..	Current position with respect to the new program origin.

Calculation for program origin preset on an axis

New DAT1 = Previous DAT1 + Previous current point/OP - Value programmed with G92

or

New DAT1 = Current point/OM - Value programmed with G92 - Tool length (in the axis) - DAT2

This operation is not performed until completion of the block before that containing function G92.

STOP The new offsets remain active at the end of the program.

Program origin preset function G92:
- applies to all the axes, both carried and independent,
- is rejected if the last movement was programmed with reference to the measurement origin: (error message 2),
- is ignored in test (TEST) and sequence number search (SEARCH) modes,
- suspends analysis of blocks until execution of the previous block is completed,
- cannot be programmed with radius offset; the system must be in state G40,
- cannot be programmed in PGP (Profile Geometry Programming).

Example

- Value entered for Z DAT1 = -300
- Value entered for Z DAT 2 = 20
- Tool length L (correction D9) = 80
- Program origin preset G92 Z60

```
N..
N150 G00 D9 G40 X0 Y0 Z40
N160 G92 Z60                    Preset
N170 G00 Z..
N..
```

Analysis of block N150 yields:

- current point/OM Z = -160

Application of the first equation

New Z DAT1 = -300 + 40 - 60 = -320

Application of the second equation

New Z DAT1 = -160 -60 -80 -20 = -320

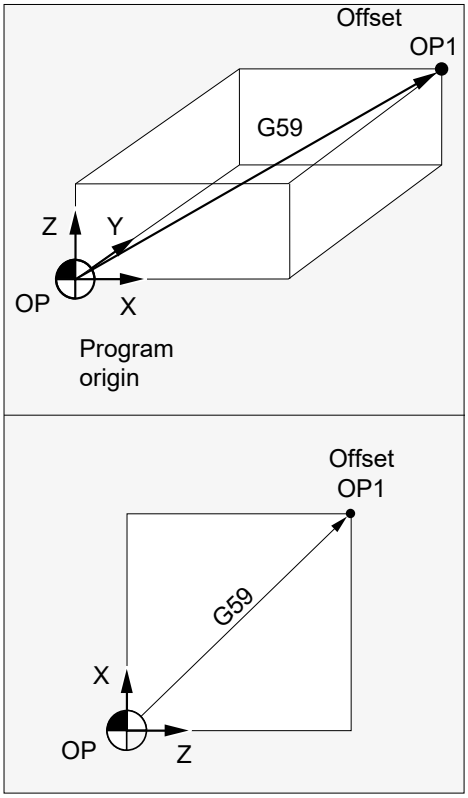
5.7.6 Program origin offset ► G59

G59	Program origin offset.

This function, associated with one or more arguments, axes and values, offsets the program origin (OP).

An origin offset can be assigned to each system axis.

No movement is caused by the function and its arguments.



Syntax

N..[G90/G91] **G59** X.. Y.. Z.. U.. V.. W.. A.. B.. C..[I.. J.. K.. ED..]

G90/G91	Programming in absolute or incremental dimensions.
G59	Program origin offset.
X.. Y.. Z.. U.. V.. W.. A.. B.. C..	The axes programmed are the arguments associated with the function. They must immediately follow the function and at least one axis must be programmed.
I.. J.. K.. ED..	I.. J.. K..: Arguments defining the center of rotation of an angular shift programmed with ED in the plane with respect to the initial program origin). The translation, if any, of the program origin is carried out after the rotation. ED..: Angular shift.

Properties of the Function

Function G59 is nonmodal. The arguments associated with the function are modal.

Cancellation

A programmed offset G59.. is cancelled by:
- programming G59 with axis arguments assigned zero values in absolute dimensions (G90),
- the end of program function (M02),
- a reset.

Notes

In case G59 is used in G21 (polar programming) the results are different according to the situation:

G21 not active when G59 is programmed : The offset is considered on the real axes

G21 active when G59 is programmed : the offset is considered on the virtual axes

For simplification and understanding of the program, it is recommended to be in state G90 (absolute programming) before programming an origin offset.

5.7.6.1 Function G59 programmed in absolute dimensions (G90)

The origin offset G59 ... is referenced to DAT1 + DAT2. A new origin offset G59 ... replaces a previous one.

5.7.6.2 Function G59 programmed in incremental dimensions (G91)

The first movement programmed after G59 ... is translated by the value of the program origin offset. A new origin offset is applied to the next movement, but the absolute position is offset by the sum of all the G59 ... functions programmed earlier.

Remark

Regardless absolute or incremental programming, it is important to respect the following order of declaration:

- ED.. Angular offset,
- G59 ... Program origin offset,
- G51 ... Mirror function,
- Scaling factor.

5.7.6.3 Features related to arguments I, J (K)

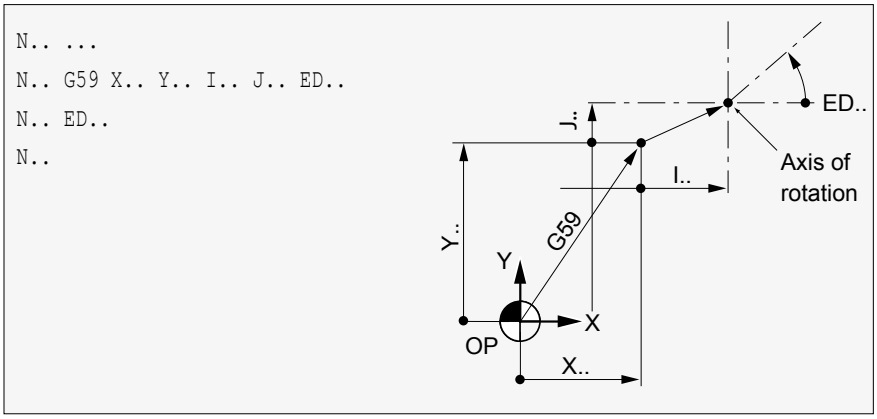
Machining programmed with respect to the program origin (OP) can be translated and oriented according to the angle programmed with ED.

The path is first rotated around a point defined by I, J (case of G17) with an angle defined by ED. The resulting path is then translated by the X, Y values introduced in G59.

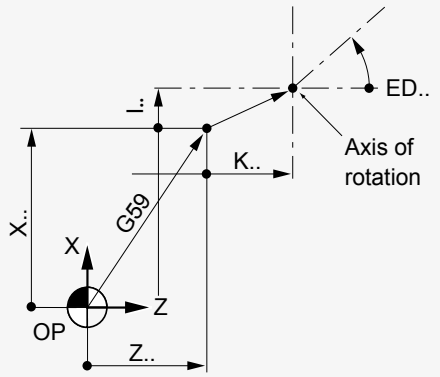
An offset on the X and Y (Z) axes is not mandatory for programming I and J (K).



```
N.. ...
N.. G59 X.. Y.. I.. J.. ED..
N.. ED..
N..
```

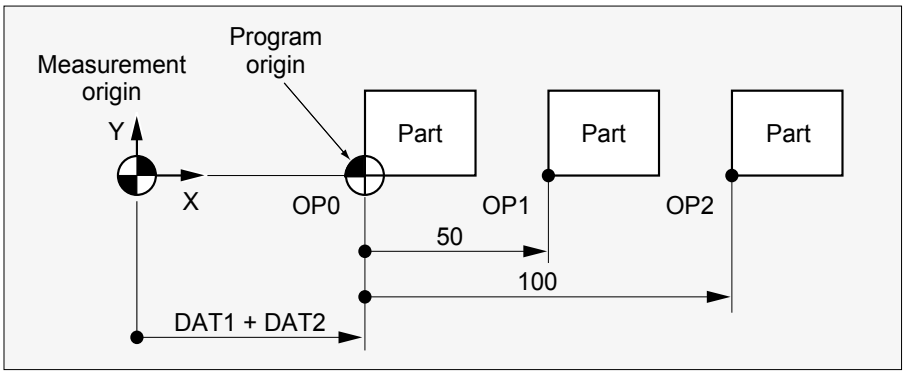


```
N.. ...
N.. G59 X.. Z.. I.. K.. ED..
N.. ED..
N..
```



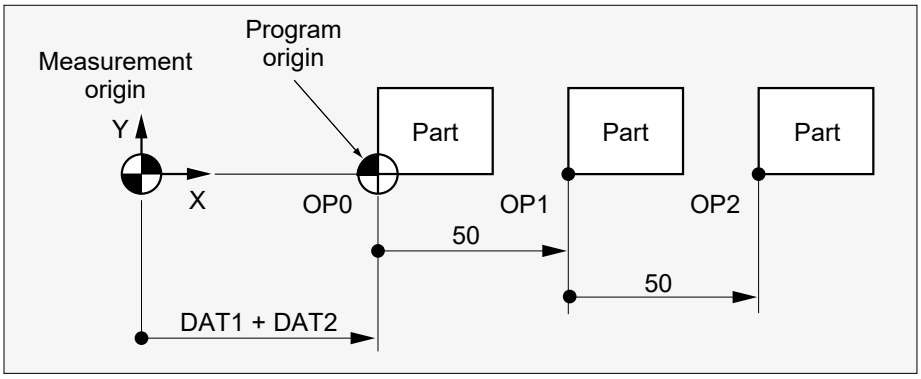
Examples

Origin offsets on the X axis programmed in absolute dimensions



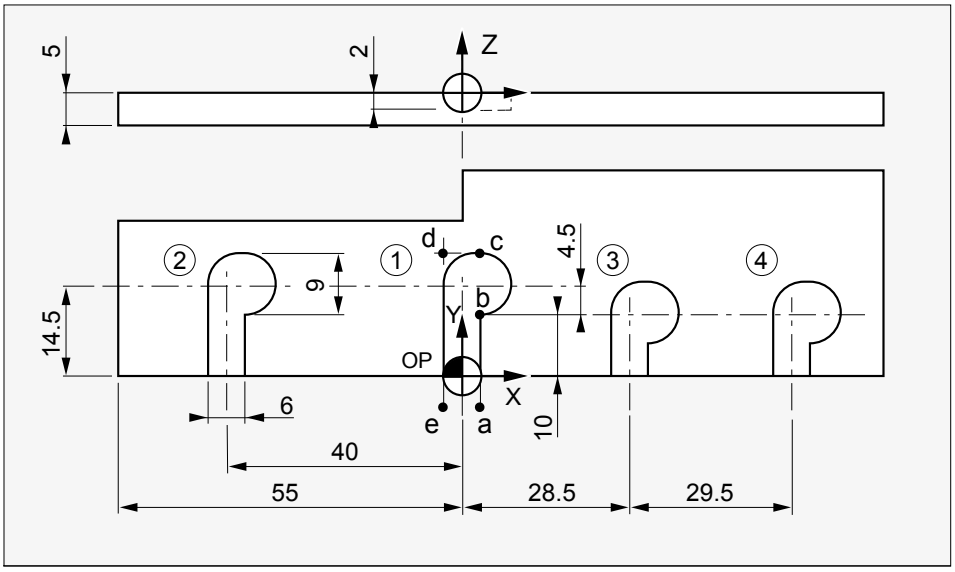

```
%60
N10
N..
N50
N..
N90
N..
N120 G90 G59 X50                    Offset 1
N.. G77 N50 N90                     Machining
N..
N230 G59 X100                       Offset 2
N.. G77 N50 N90                     Machining
N..
N350 G59 X0                         Cancellation
N..
```

Origin offsets on the X axis programmed in incremental dimensions



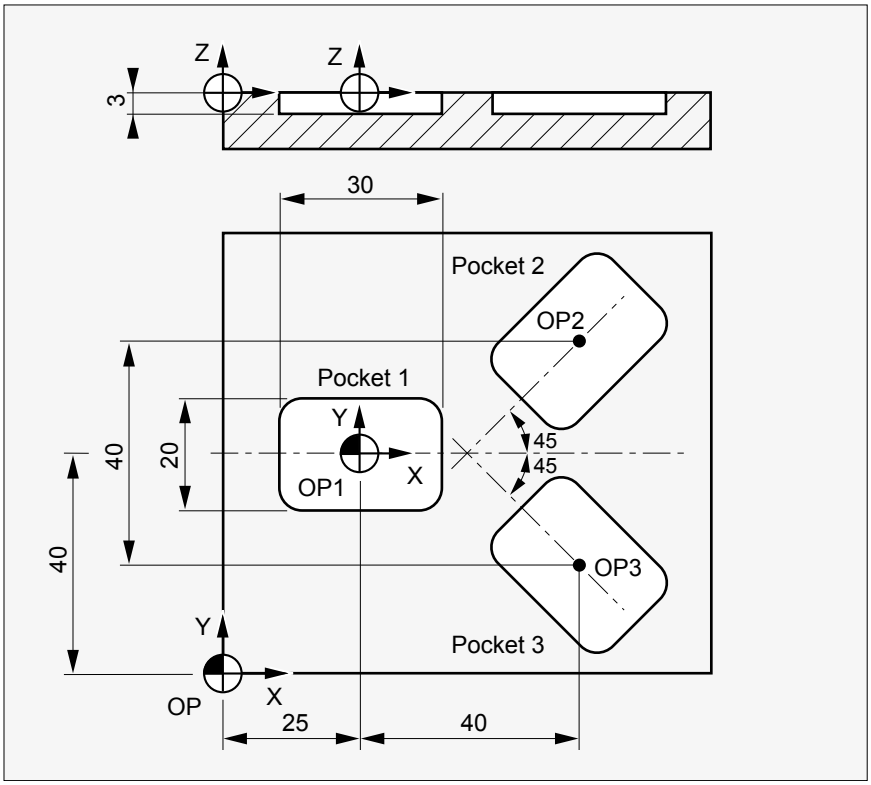

```
%70
N10
N..
N50
N..
N90
N..
N120 G91 G59 X50                    Offset 1
N.. G77 N50 N90                     Machining
N..
N230 G59 X50                        Offset 2
N.. G77 N50 N90                     Machining
N..
N350 G91 G59 X-100 or G90 G59 X0    Cancellation possible in G91 or G90
N..
```

Repetition of a shape with origin offsets, XY plane (G17) example



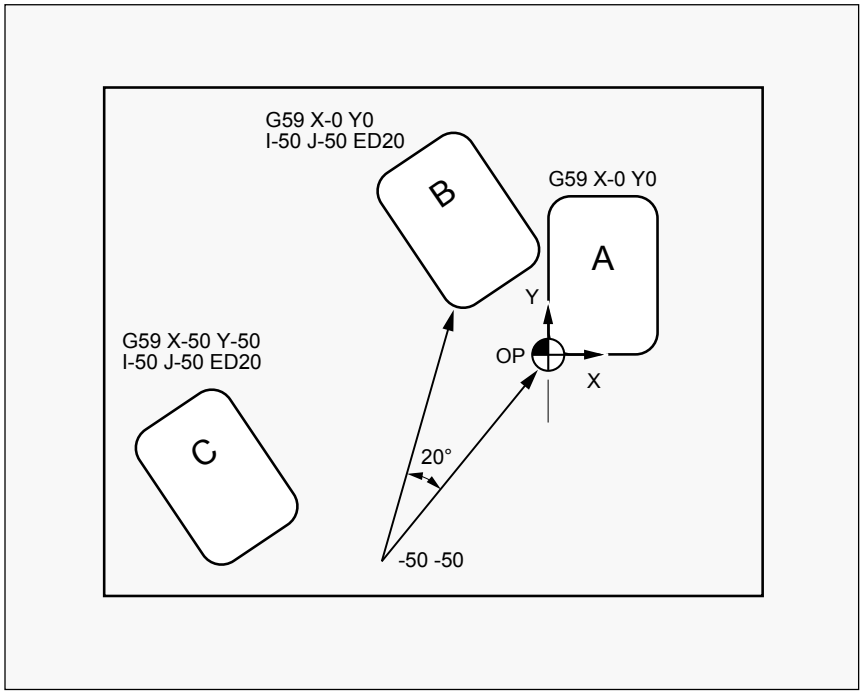

```
%110
N10 G00 G52 Z0
N20 T09 D09 M06 (CUTTER DIAMETER=5)
N30 S2000 M40 M03
$0 SHAPE 1
N40 G00 G41 X3 Y-5                    Point a, approach X, Y
N50 Z-2                               Approach on Z
N60 G01 Y10 F120                      Point b
N70 G03 X3 Y19 R4.5 F80               Point c
N80 G01 X-3 F120                      Point d
N90 Y-5                               Point e
$0 SHAPE 2
N100 G59 X-40                         Absolute offset on X
N110 G77 N40 N90
$0 SHAPE 3
N120 G59 X28.5 Y-4.5                  Absolute offset on X and Y
N130 G77 N40 N90
$0 SHAPE 4
N140 G91 G59 X29.5                    Incremental offset on X
N150 G90 G77 N40 N90
N160 G59 X0 Y0                        Offset cancelled
N170 G00 G40 G52 Z0 M05 M09
N180 M02
```

Repetition of a pocket by origin offsets and angular offsets




```
%61
N10 G00 G52 Z0
N20 T11 D11 M06 (CUTTER DIAMETER=10)
N30 S800 M40 M03 M08
$0 POCKET 1
N40 G59 X25 Y40                          Origin offset
N50 G45 X0 Y0 EX30 EY20 ER1 P3 Q8 Z-3
EP50 EQ200
$0 POCKET 2
N60 G59 X65 Y60                          Pocket center offset
N70 ED45 G77 N50 N50                     Angular offset and machining
$0 POCKET 3
N80 G59 X65 Y20                          Pocket center offset
N90 ED-45 G77 N50 N50                    Angular offset and machining
N100 ED0 G00 G52 Z0 M05 M09
N110 M02
```

Example using I, J and ED




```
%1(Main)()
G0 G17 Z100 T1 D1 M6 M3 S1000                    Pre-positioning
X-50Y-50
G1 F2000 Z00
Z100

G59 X0 Y0
N5 G77 N100 N200                                 Execute figure A
G59 X0 Y0 I-50 J-50 ED20
N20 G77 N100 N200                                Execute figure B

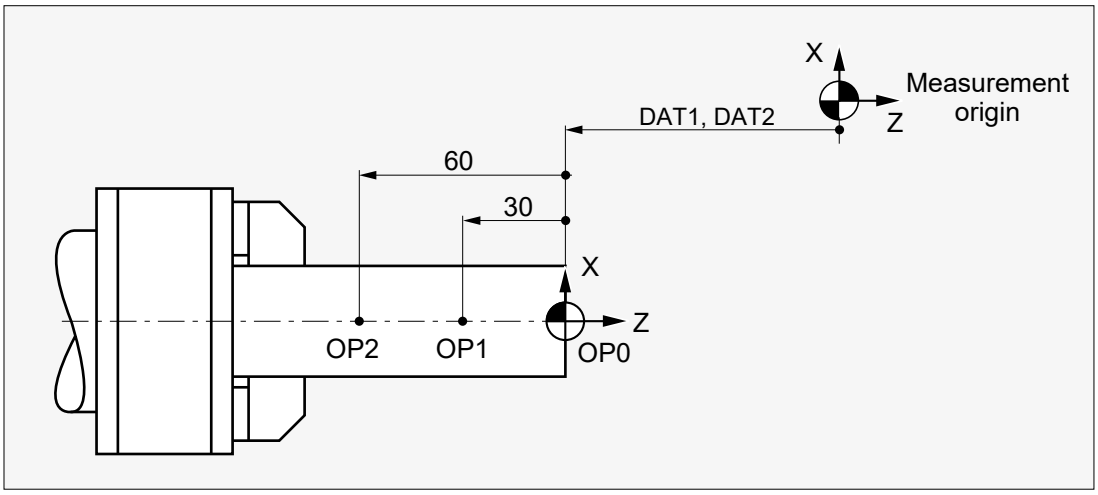
G59 X-50 Y-50
G77 N100 N200                                    Execute figure C

G59 X0 Y0
GXY
M2                                               End of program

N100
G1 F500 X0 Y0
G45 X0 Y0 EX200 EY250 ER100 P10 Q20 EP50 EQ50 Z10
EP50 EQ200                                       Pocket cycle as a subroutine
N200
M2
```

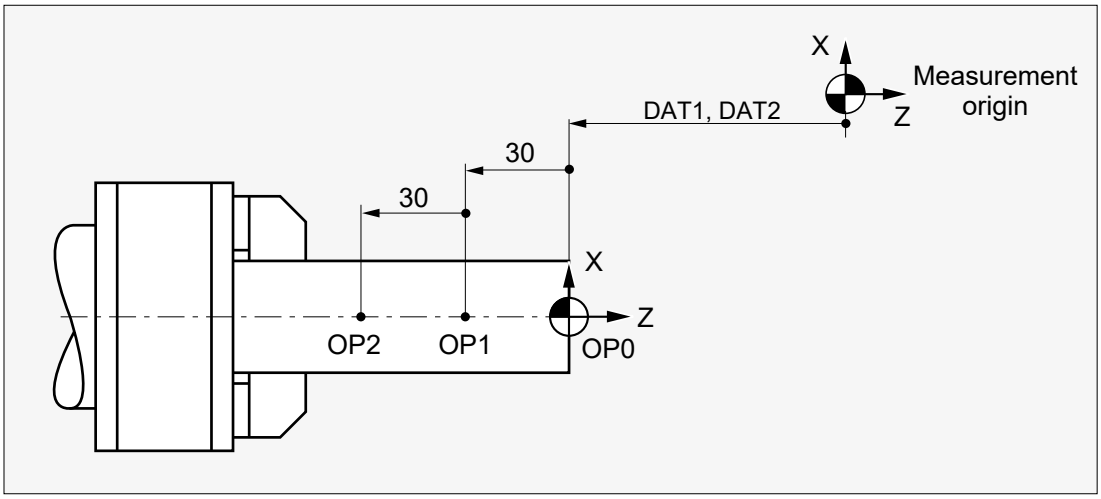
Examples

Origin offsets on the Z axis in absolute programming




```
%60
N10
N..
N50
N..
N90
N..
N120 G90 G59 Z-30          Offset 1
N.. G77 N50 N90            Machining
N..
N230 G59 Z-60              Offset 2
N.. G77 N50 N90            Machining
N..
N350 G59 X0                Offset cancelled
N..
```

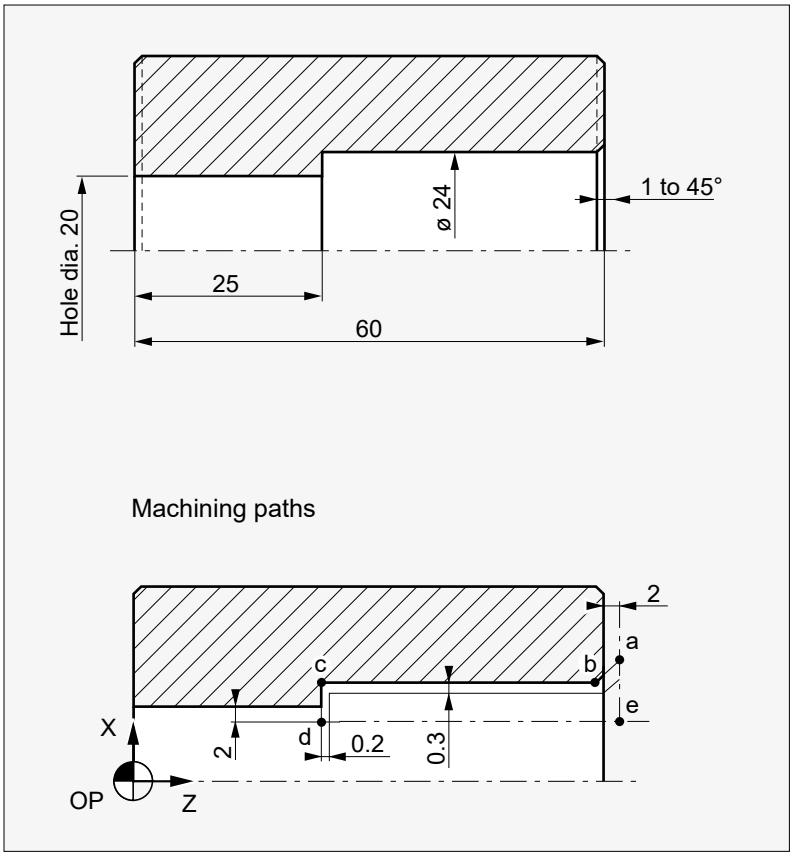
Origin offsets on the Z axis in incremental programming




```
%70
N10
N..
N50
N..
N90
N..
N120 G91 G59 Z-30                    Offset 1
N.. G77 N50 N90                      Machining
N..
N230 G59 Z-30                        Offset 2
N.. G77 N50 N90                      Machining
N..
N350 G91 G59 Z-60 or G90 G59 X0      Cancellation possible with G90 or G91
N..
```

Machining allowance by origin offset

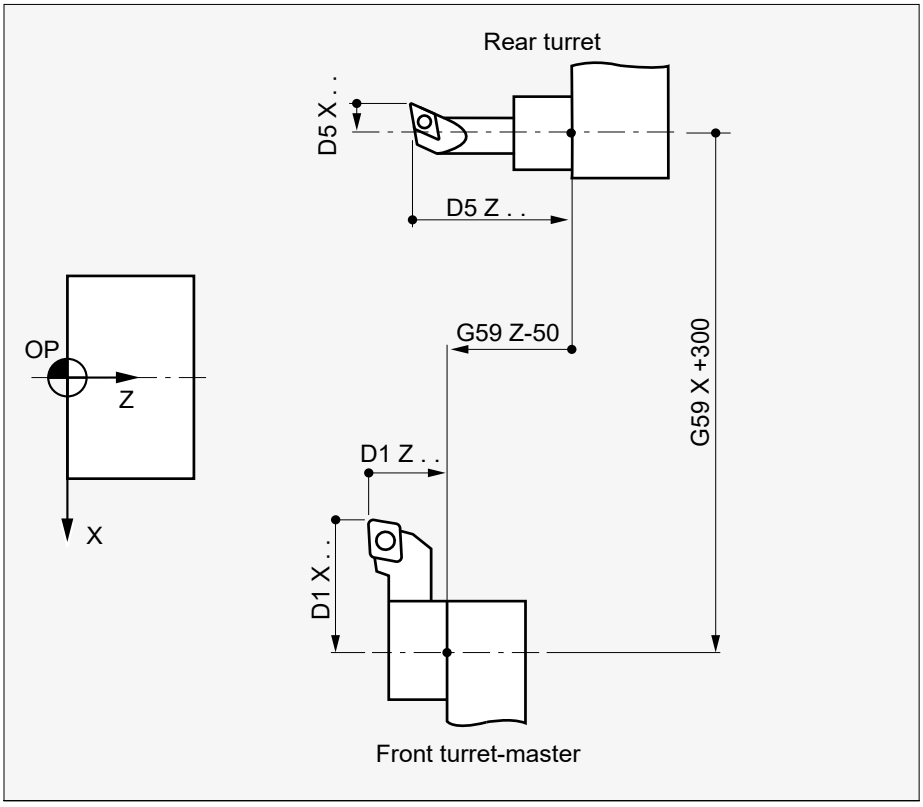
Bore roughing, finishing. Machining allowance: X = 0.3, Z = 0.2




```
N.. ...
N80 G00 G52 X.. Z.. G97 S500
N90 T07 D07 M06 (ROUGHING BORING TOOL)
N100 G95 F0.2
N110 S900 M04 M40
N120 G59 X-0.3 Z0.2                    Origin offsets
N130 G00 G41 X30 Z62                   Point a
N140 G96 S80
N150 G01 X24 Z59                       Point b
N160 Z25                               Point c
N170 X16                               Point d
N180 G00 Z62                           Point e
N190 G59 X0 Z0                         Offsets cancelled
N200 G40 G77 N80 N80
N210 T09 D09 M06 (FINISHING BORING TOOL)
N220 G77 N130 N180                     Finishing
N..
```

Use of origin offset on a lathe with two turrets

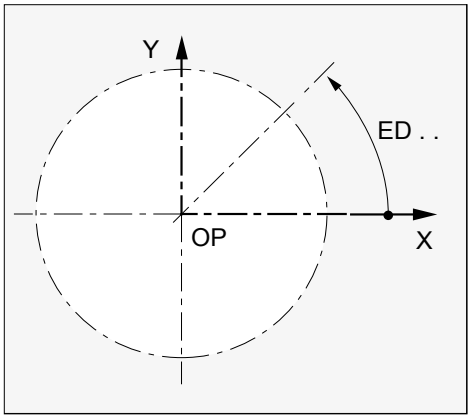
On lathes equipped with two turrets (front and rear), the tool dimensions of the secondary turret can be identified with respect to the reference of the master turret by programming an origin offset.




```
% 88
$0 MACHINING WITH FRONT MAIN TURRET
N10 G00 G52 X0 Z0
N20 T01 D01 M06 (EXTERNAL TOOL)
N..
N..
N340 G00 G52 X0 Z0
$0 MACHINING WITH REAR TURRET
N350 T05 D05 M06 (INTERNAL TOOL)
N360 G90 G59 X300 Z-50
N..
N500 G59 X0 Z0
N..
```

5.7.7 Angular offset ► ED..

ED..	Programmed angular offset.



Function ED associated with a value defines an angular rotation with respect to the program origin.

The angular offset applies to the axes of the plane programmed in the blocks following the function.

Syntax

N.. [G90/G91] **ED..**

G90/G91	Programming in absolute or incremental dimensions.
ED..	Value of the angular offset in degrees and thousandths of a degree (format ED+034).

Property of the Function

Function ED is modal.

Cancellation

Angular offset ED.. is cancelled by:

- reprogramming function ED with a zero value (ED0) in absolute dimensions (G90),
- the end of program function (M02),
- a reset.

Notes

Angular offset ED applies to:

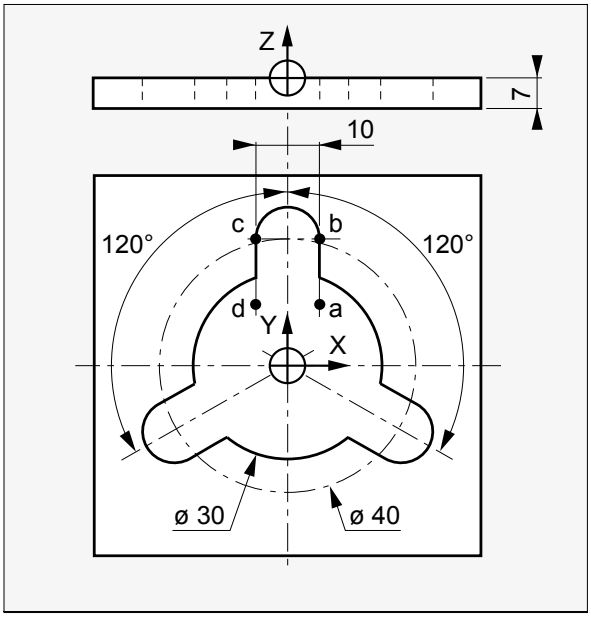
- all the elementary cycles (G81, G45, etc.),
- radius offset (G41, G42),
- PGP (Profile Geometry Programming) unless ED.. is programmed between two blocks that are not completely defined,
- carried or independent secondary axes (U, V, W).

If an angular offset is programmed in a plane (G17, G18 or G19), it is not cancelled by a change of plane. ED.. then applies to the values in the new plane.

Examples

Execution of three grooves in the XY plane

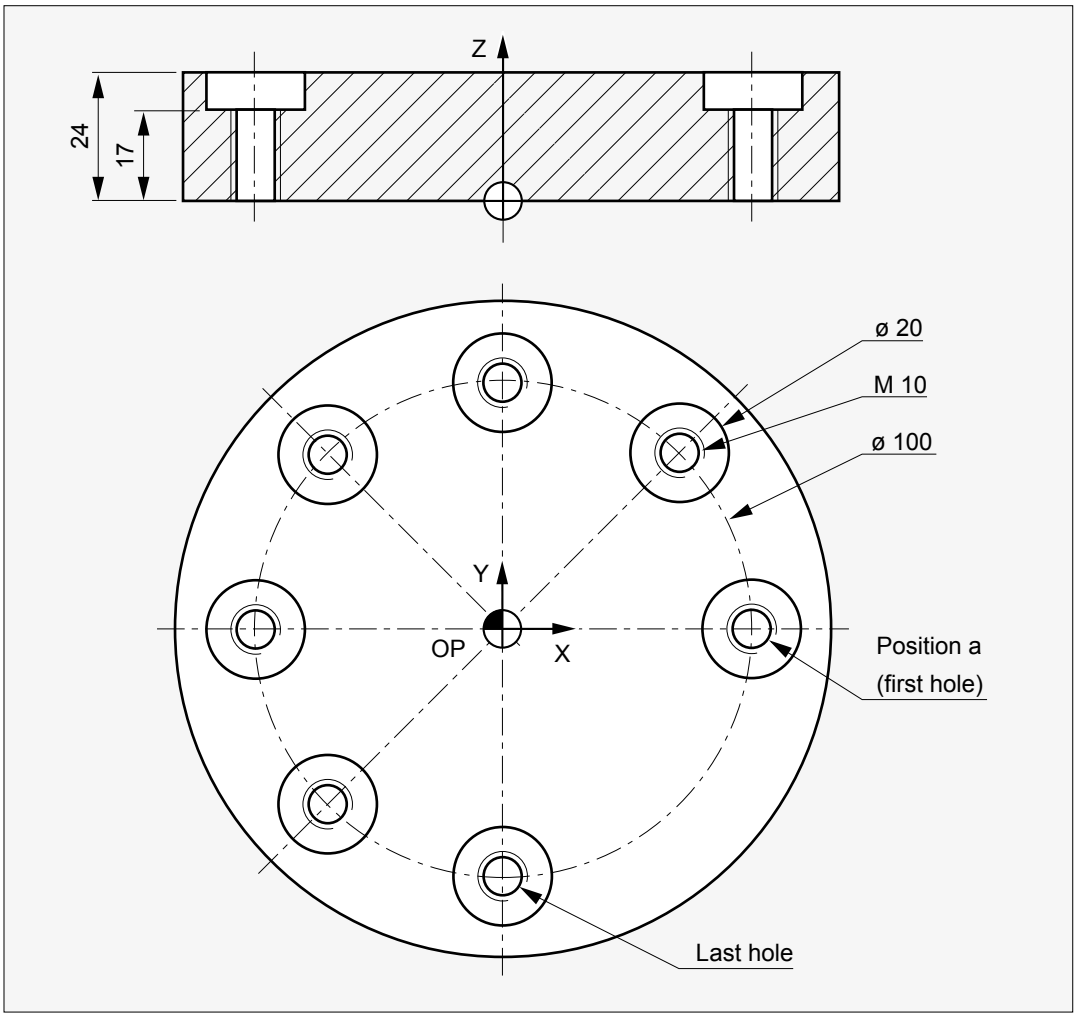
Bore diameter 30 completed.




```
%60
N10 G00 G52 Z-60 M05 M09
$0 3 GROOVES ROUGHING
N20 T06 D06 M06 (CUTTER DIAMETER=8)
N30 S800 M40 M03
N40 G00 G41 X5 Y10                  Point a, groove 1 approach
N50 Z-10                            Approach on Z
N60 G01 Y20 F100 M08                Point b
N70 G03 X-5 Y20 R5 F50              Point c
N80 G01 Y10 F100                    Point d
N90 ED+120                          Angular offset 120 degrees
N100 G77 N40 N80                    Execution of groove 2
N110 ED+240                         Angular offset 240 degrees
N120 G77 N40 N80                    Execution of groove 3
N130 ED0 G77 N10 N10
$0 3 GROOVES FINISHING
N140 T07 D07 M06 (CUTTER DIAMETER=8)
N150 G77 N30 N130
N160 M02
```

Drilling, spot facing and tapping of seven holes at 45° apart

Movement by linear interpolation between holes. Main program




```
%1
$0 MAIN PROGRAM WITH SUBROUTINES %10 %20 %30
N10 G00 G52 Z0 M05 M09
N20 T01 D01 M06 (CUTTER DIAMETER=8.5)
N30 S1000 M40 M03
N40 X50 Y0 Z27                    Point a, approach
N50 G77 H10 S7                    Branch to subroutine %10
N60 G90 ED0                       Offset cancel
N70 G77 N10 N10
N80 T02 D02 M06 (BLADE DIA=20)
N90 S250 M03
N100 X50 Y0 Z27                   Point a, approach
N110 G77 H20 S7                   Branch to subroutine %20
N120 G90 ED0                      Offset cancel
N130 G77 N10 N10
N140 T03 D03 M06 (TAP M10)
N150 S200 M03
N160 X50 Y0 Z27                   Point a, approach
N170 G77 H30 S7                   Branch to subroutine %30
N180 G90 ED0                      Offset cancel
N190 G77 N10 N10
N200 M02
```

Subroutines

```
%10
$0 DRILLING
N10 G90 G81 X50 Y0 Z-5 F90 M08          Drilling cycle
N20 G80 G91 ED45                        Angular offset in incremental dimensions

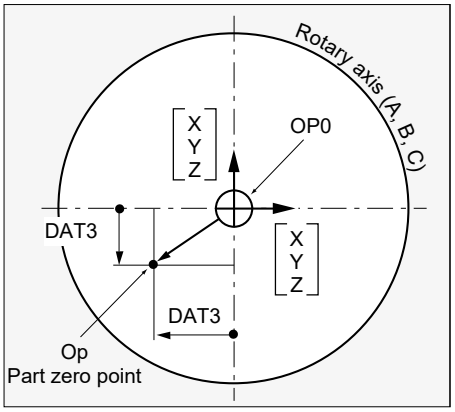
%20
$0 SPOT FACING
N10 G90 G82 X50 Y0 Z17 EF1 F50 M08      Spot facing cycle
N20 G80 G91 ED45                        Angular offset in incremental dimensions

%30
$0 TAPPING
N10 G90 G84 X50 Y0 Z-8 F300 M08         Tapping cycle
N20 G80 G91 ED45                        Angular offset in incremental dimensions
```

5.7.8 Table offset by DAT3 with Milling machine ► DAT3



DAT3 Milling Table offset.



This function applies to rotary axes A, B or C.

The offsets can be applied by entering values:

- on the context "Work Offset”,
- by external parameter E,
- by ENA (from the PLC).



The DAT3 feature takes into account the theoretical shifts caused to a pair of axes when the part zero is not co-incident with the center of rotation.

Rotary axes and axis pairs affected by DAT3

Rotary axis	Axis pairs
A	YZ
B	ZX
C	XY



The axes of the pair to which a DAT3 can be assigned are identified by stars on the context "Work Offset”. A value entered for an axis not marked by a star is ignored.

Entry of DAT3 values on the NC
Refer to the Flexium⁺ HMI Operator Manual M00039.

Entry of DAT3 by external parameters

Parameters E6x004 are use to program the offset (x = logical address of the axis).

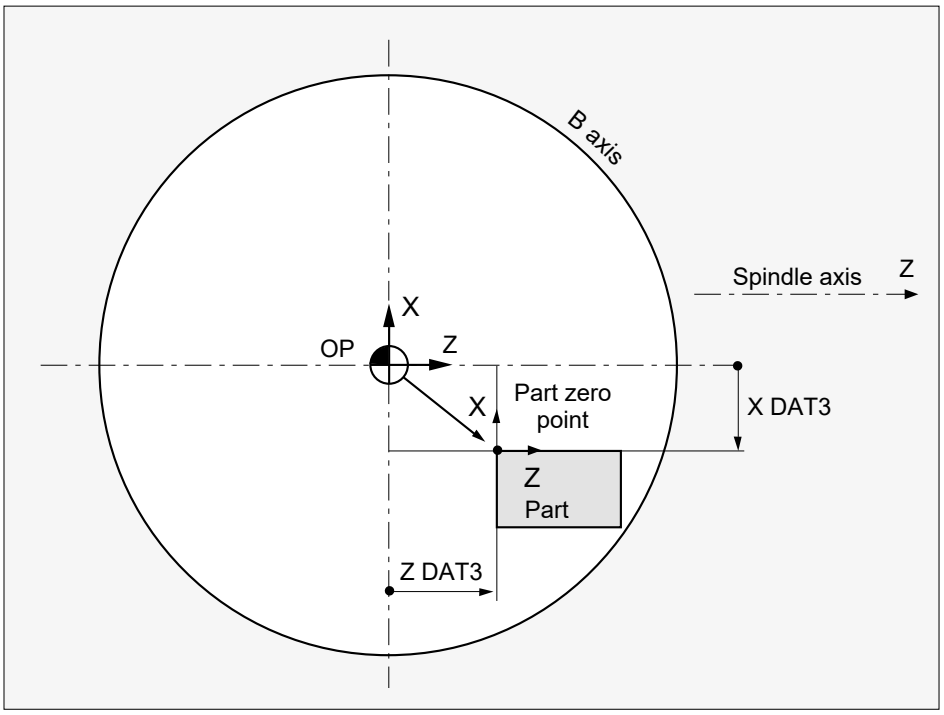
Example

Programming offsets for the B axis affecting the XZ pair.

```
N.. ...
E60004=-50000          Negative 50 mm offset on X
E62004=70000           Positive 70 mm offset on Z
N..
```

The figure below is shown with a slaved B axis (see chapter 4).

The representation is the same for the A and C axes.



5.8 Tool offsets

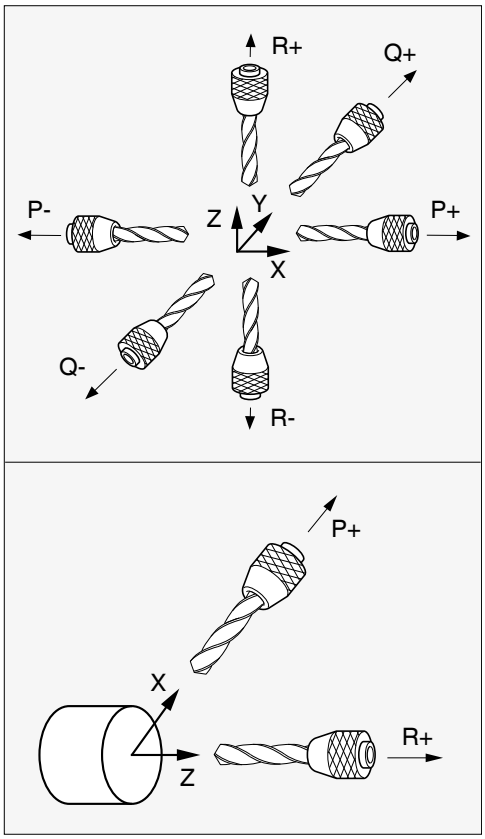
5.8.1 Tool axis orientation ► G16



G16	Definition of the tool axis orientation with addresses P, Q, R.

The tool axis orientation is defined by this function with one of the compulsory arguments P, Q or R followed by a plus or minus sign.

The tool axis can be oriented in six different positions on machines with an interchangeable head or a right angle attachment.



Syntax

N.. **G16** P±/Q±/R±

G16	Definition of the tool axis orientation.
P+	Points along X+
P-	Points along X-
Q+	Points along Y+
Q-	Points along Y-
R+	Points along Z+
R-	Points along Z-

Properties of the Function

Function G16 followed by one of its arguments P, Q, R is modal.

Function G16 followed by R+ is the default function.

Cancellation

Function G16 followed by one of arguments (P, Q or R) different from the one already programmed cancels the former state G16.

By convention, the tool vector points from the tool tip (cutting part) to the tool reference (spindle mounting).

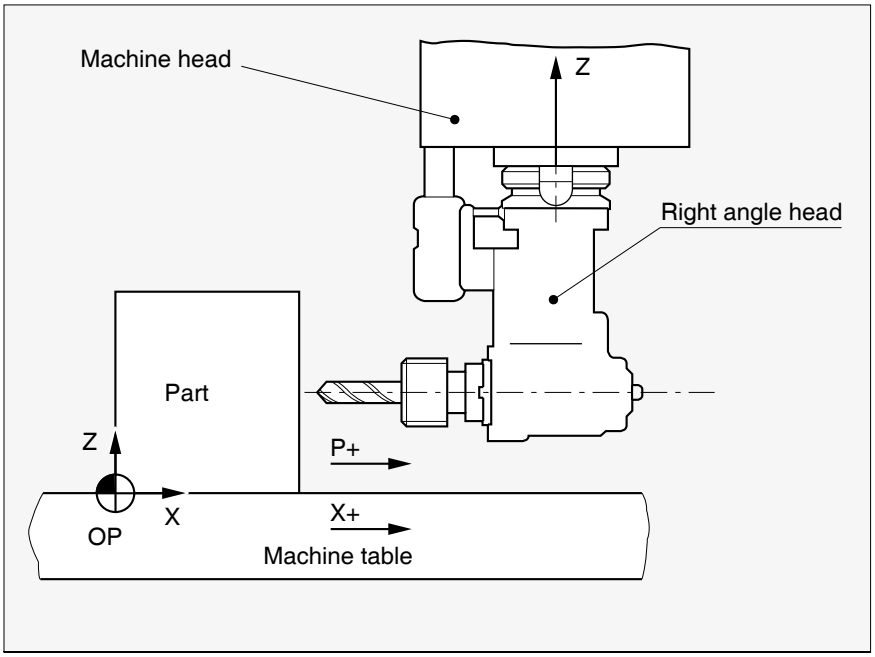
The tool axis cannot be an independent secondary axis.

When defining the tool axis orientation:
- it is recommended to cancel tool radius offset (G40) and canned cycles (G80),
- the block containing G16.. may include movements, miscellaneous M functions and technological S and T functions.

Example

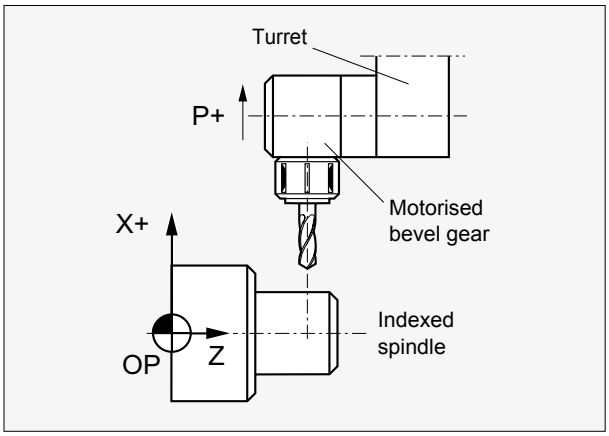
Machine equipped with a right angle head






```
%44
N10 G00 G52 Z.. (G17 G16 R+)        Orientation initialized along Z+
N20 T08 ... M06                     Tool change
N30 S400 M40 M03
N..
N..
N170 G00 G52 X.. Y.. Z..
N180 G16 P+                         Axis orientation along X+
N190 G00 Y.. Z..
N200 G01 X.. F..
N..
```

Example
Machine equipped with a live tool




```
%44
N10 G00 G52 X.. Z.. (G16 R+)          Orientation initialized along Z+
N20 T08 ... M06 (TOOL)
N30 G97 S900 M40 M04
N..
N..
N170 G00 G52 X.. Z..
N180 T11 ... M06 (DRILL)
N190 G97 S200 M03 M40
N200 C0 M19
N210 G16 P+                           Axis orientation along X+
N220 ...
N..
```

5.8.2 Tool change ► M06

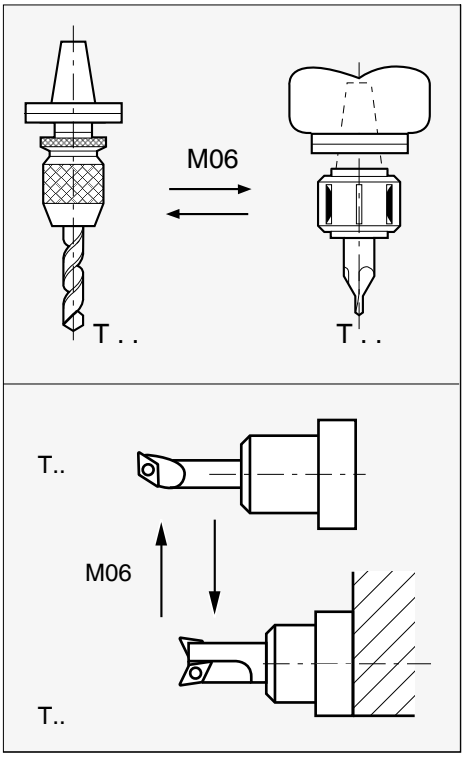


M06 Tool change.

This function calls for a tool to be placed in the spindle.

The tool is placed in the spindle either automatically or manually.

Caution. When the Tool management is active do not forget to reprogram the offset number once the tool change has been completed.



Syntax

N.. T.. **M06** [$0 / (...)]

T..	The tool is selected by function T followed by a number.
M06	Tool change.
$0 or (...)	Possible message or comment concerning the tool characteristics.

Properties of the Function

Function M06 is a nonmodal "post" function.

Cancellation

Function M06 is reset at the end of the block.

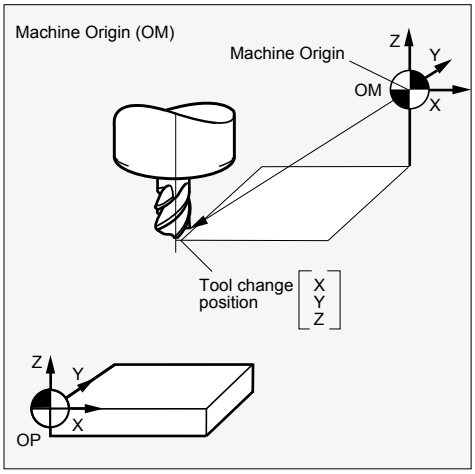
Notes

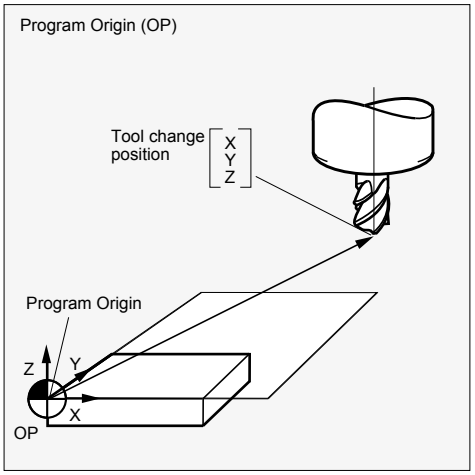
Function T defining the tool number must not be assigned a value greater than 99999999. Above, the system returns error message 1.

Before a tool change, it is recommended to program a safe position for placing the tool in the spindle:
- with reference to the program origin (OP), or
- with reference to the measurement origin programmed with function G52.



Examples






```
N..
N120 G00 G52 Z.. or G00 Z..
N130 T09 M06 (CUTTER DIAMETER=25)
N..
```

Automatic tool change

```
N..
N100 T05 M06 (CUTTER DIAMETER=20)
N..
```

Tool change by subroutine

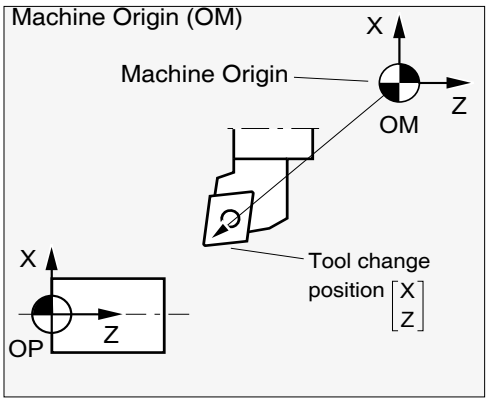
```
N..
N80 T06 (CUTTER DIAMETER=50)
N90 G77 H9000
N..
```

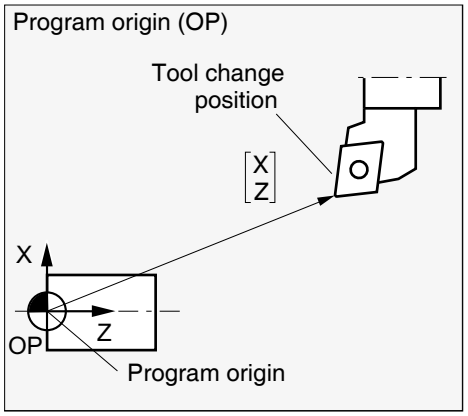
Tool change with spindle preselection and indexing

```
N..
N20 M19
N30 T02 M60 (CUTTER DIAMETER=30)
N40 M06 D02
N..
```

Examples

Automatic tool call



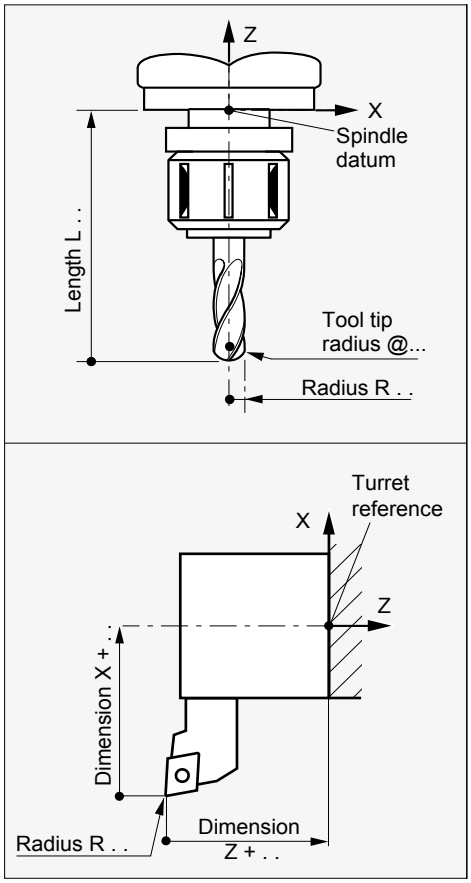



```
N.. ...
N90 G00 X.. Z.. or G00 G52 X.. Z..
N100 T05 M06 (TURN/FACING TOOL R=0.8)
N..
```

5.8.3 Tool Offset activation ▶ D..

D..	Tool Offset activation.

Address D followed by a number selects the tool correction.



5.8.3.1 Case with Tool Management function

According to the tool definition, it is possible to have up to 18 offsets. Each group of offsets corresponds to a particular cutting edge of the tool loaded in spindle (stepped tools). In case of more than 14 spindles per system the number of offsets is limited following the rule Number of offsets = 255 / number of spindles.

The values are displayed in the context ToolMan (sF3), they contain for each cutting stages the usual L, R, @ or X, Z, R, C data as well as tool life data and maximum speed. There is no error detected if those speed and feed rate limits are exceeded in programming but the limitation will be effective.

Each spindle gets its own set of tool offsets. The length and radius correction taken into account are those of the main spindle

Unlike the case without tool management , the D number represents the active cutting edge to select.

STOP With tool management it is MANDATORY to reprogram the D number after each tool change even if this number is unchanged.

Milling Machine data
- L = Tool length
- R = Tool radius
- @ = Tool tip radius.

The dimensions can be entered:
- manually or via a peripheral,
- by parametric programming.

Turning Machine data
- X = Tool dimension along X axis
- Z = Tool dimension along Z axis
- R = Tool radius
- C = Tool tip orientation

The dimensions can be entered:
- manually or via a peripheral,
- by parametric programming.

Syntax (G17 plane)

N.. [G17] [G16 R+] **D..** [G40/G41/G42] X.. Y.. Z..

G17	XY plane selection.
G16R+	Tool axis orientation along Z+.
D..	Correction number (1 to 18 corrections).
G40	Tool radius offset cancel.
G41/G42	Tool radius offset.
X..Y.. Z..	End point.

Properties of the Function

Function D.. is modal. Correction D0 is the default correction.

Cancellation

Function D.. is cancelled by programming a new correction or D0.

Notes

D0 always contains zero.

The system includes between 8 and 18 correctors per tool. Additionally the number of cutting edges is defined as a characteristic of each tool. If the D number is higher than this characteristic, the system returns error message 8 extension 3.

Example

Supposing T100 has two cutting stages - T200 only one

```
%100
T100 M6
D1                  Using the data of the first cutting edge of tool 100
...                 Machining
D2                  Using the data of the second cutting edge of tool 100
T200                Preselecting tool 200 - The offsets of tool 100 remain active
M6                  Change tool in main spindle
D1                  Using the data of the FIRST cutting edge of tool 200
D2                  Error : Tool 200 has only one set of offsets.
```

5.8.3.2 Tool length correction (L)



The tool length correction is assigned to the tool axis orientation defined by G16...

The tool length declared is taken into account by programming:

- a correction number D...,
- a movement on the axis parallel to the tool axis orientation.

During machining, tool length settings are re-applied during:

- correction number changes,
- the use of wear corrections,
- a tool axis orientation change.

The length correction is suspended by programming G52.

The maximum dimension of L corrections is 9999.999 mm.

5.8.3.3 Tool dimensions corrections



The X and Z axes (U and W if carried) are affected by the tool dimensions stored in corrections X and Z.

The declared tool dimensions are taken into account when programming:

- a correction number D..,
- a movement of the axis parallel to the tool axis orientation.

During machining, the tool dimensions are re-applied during:

- correction number changes,
- the use of wear corrections,
- a tool axis orientation change.

The corrections are suspended by programming G52.

The maximum dimension of the X and Z corrections is 9999.999 mm.

5.8.3.4 Tool radius corrections



The tool radius correction is assigned to one of the interpolation planes.

The tool radius declared is taken into account when programming:

- the correction number D..,
- one of functions G41 or G42,
- one of the axes of the interpolation plane.

During machining, a change in the tool radius is made by cancelling the radius offset by G40 then reprogramming the radius offset by G41 or G42 after:

- changing the correction number,
- changing the tool wear compensation.

The maximum dimension of R correction is 9999.999 mm.



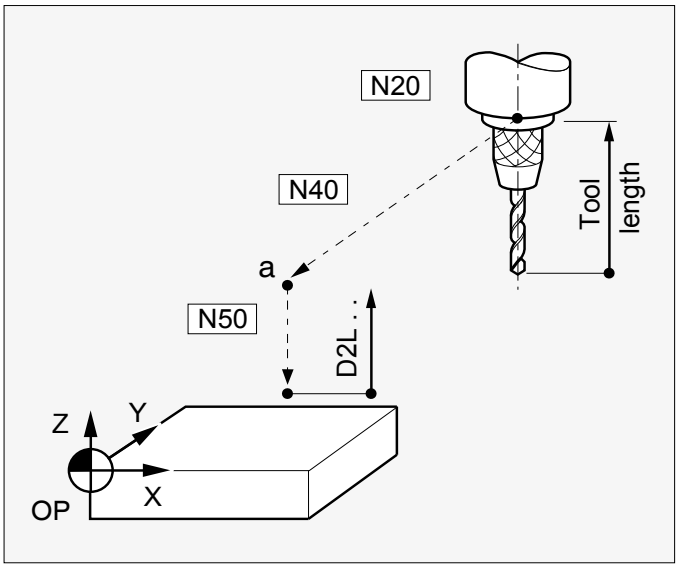
The two axes of the interpolation plane can be primary axes and carried or independent secondary axes.

Example



Multiple length corrections for a single milling tool

Length corrections L.. of tool T02 are taken into account during the first movement on the Z axis programmed after D02 and D12.

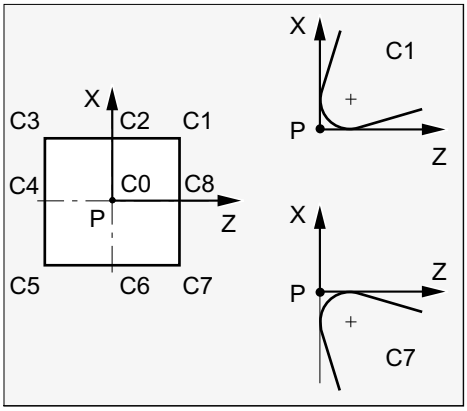



```
%55
N10 G16 R+ G17
N20 T02 D02 M06                    Call of tool T02 and correction D02
N30 S180 M40 M03
N40 G00 X100 Y20
N50 Z30                            Taking into account length L.. of D02
N..
N100 D12 Z30                       Taking into account length L.. of D12
N..
```

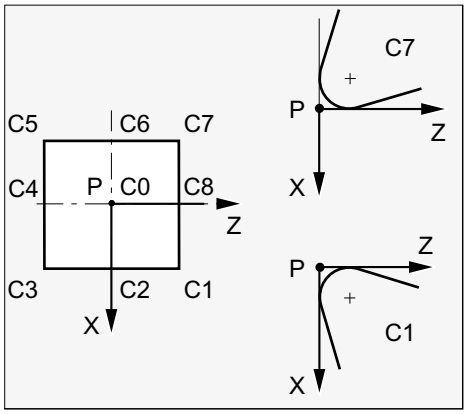
5.8.3.5 Tool tip orientation (C) according to the turret position

The tool tip orientation is defined by codes C0 to C8 (see chapter. 2.2.5).

Rear turret



Front turret



5.8.3.6 Use of tool corrections (X, Z, R and C) on a lathe with two mechanically coupled turrets

Case of a main front turret.

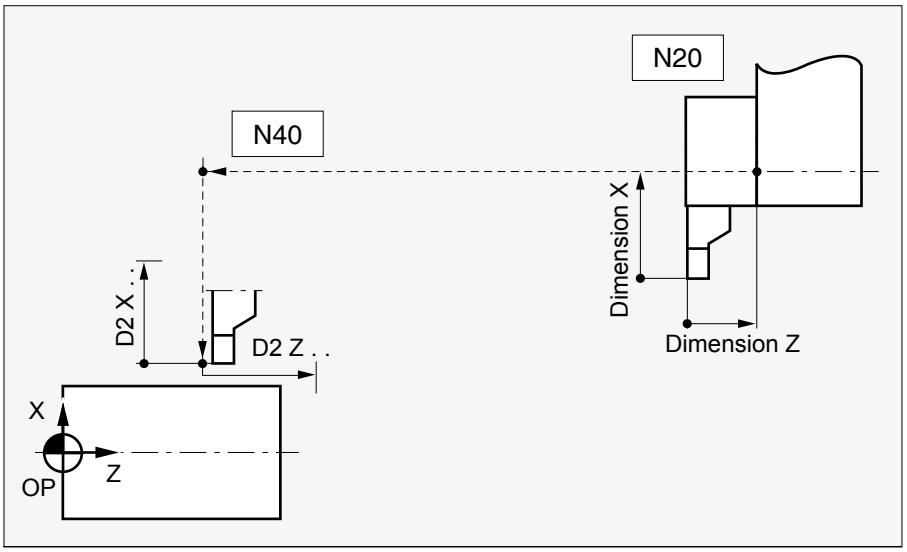
The dimensions of the tools in the secondary turret are defined with respect to the reference point of the main turret.

In this case, the tool tip orientation is defined with reference to the front position of the main turret, i.e. C1.

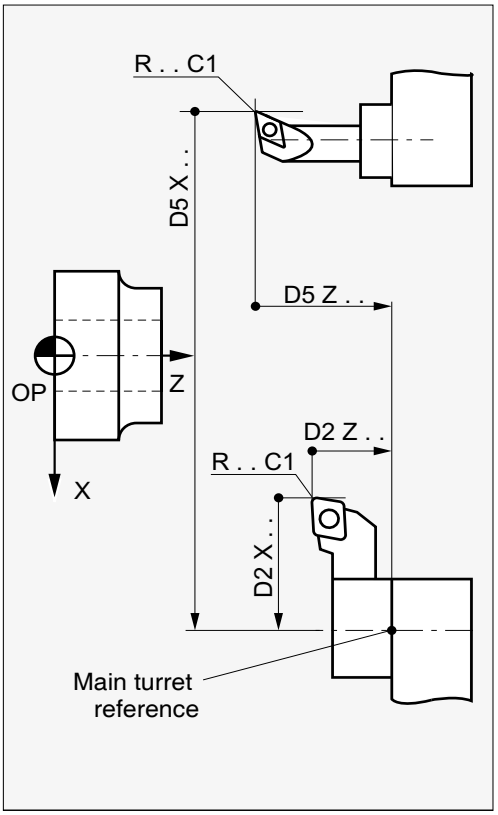
Definition of the tool tip orientation is only important when radius offset G41 and G42 are programmed. With G40 (radius offset cancelled), C0 is defined.

Example

Multiple dimension corrections for a single turning tool




```
%55
N10 G00 G52 X.. Z.. G16 R+
N20 T02 D02 M06                  Call of tool T02 and correction D02
N30 G97 S900 M40 M04
N40 G00 Z100                     Taking into account dimension Z.. of  D02
N50 X60                          Taking into account dimension X.. of  D02
N..
N100 D12 G00 X80 Z100            Taking into account dimensions X.. and
                                 Z.. of D12
N..
```

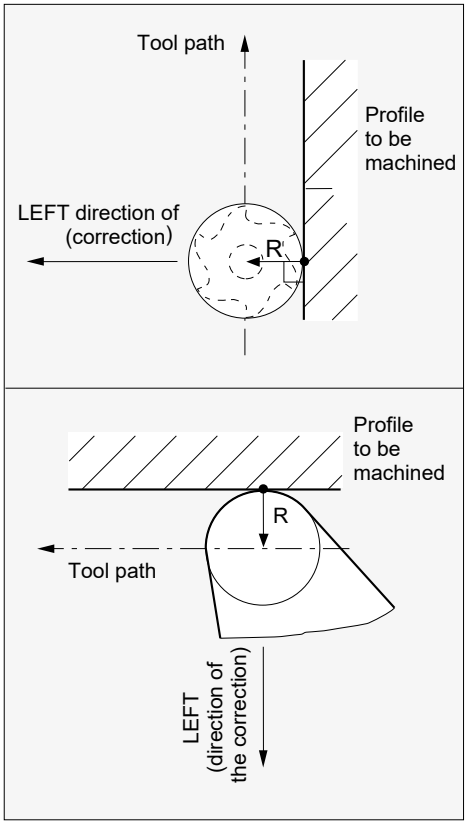


5.9.1 Tool radius compensation ► G41, G42, G40



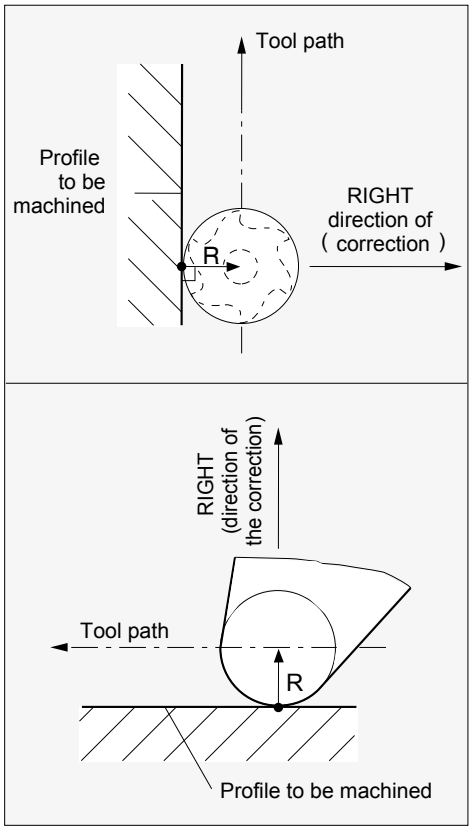
G41 Left radius offset.

The tool paths programmed are corrected (offset to the left) by a value equal to the tool radius (R) declared by corrector D...



G42 Right radius offset.

The tool paths programmed are corrected (offset to the right) by a value equal to the tool radius (R) declared by corrector D...



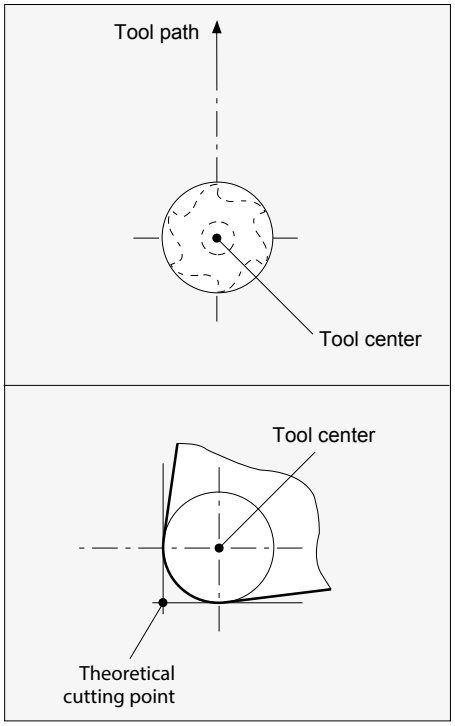
Syntax (XY plane)

N.. [G17] [D..] [G00/G01/G02/G03] **G41/G42** X.. Y..

G17	XY plane selection.
D..	Corrector number containing the tool radius offset.
G00/G01/G02/G03	Linear or circular interpolation.
G41	Radius offset to the left of the profile.
G42	Radius offset to the right of the profile.
X..Y..	End point.

G40	Radius offset cancel.

The programmed paths are applied to the center of the tool.



Syntax

N.. [G00/G01] **G40** X.. Y.. Z..

G00/G01	Linear interpolation.
G40	Tool radius offset cancel.
X.. Y.. Z..	End point.

Properties of the Function

Functions G40, G41 and G42 are modal.

Function G40 is the default function.

Cancellation

Functions G41 and G42 cancel one another.

Function G40 cancels functions G41 and G42 as well as function G29 (3D tool correction).

Notes

Functions G41 or G42 allow programming of a part profile in real profile dimensions without taking the tool radius into account.

With radius correction:
- the paths defining the part profile are followed even if the tool radius used and stored is smaller or larger than that of the intended tool,
- the tool is positioned to the left or right of the profile to be machined with respect to the direction of movement along the path.

Radius correction is carried out along a vector perpendicular to the profile with radius R.. declared in correction D as the vector length.

A change of plane (G17/G18/G19) must always be programmed in G40 mode (tool radius offset cancel).

For example:

```
N.. ...
N100 G17 G40 X..
N..
N320 G18 G41 X.. Z..
N..
```

Change of plane before calling the radius correction

When changing the direction of correction (change from G41 to G42 or vice versa), it is not necessary to cancel the radius offset by G40.

The following functions must be programmed without radius offset (system in state G40), or the system returns error message 140.

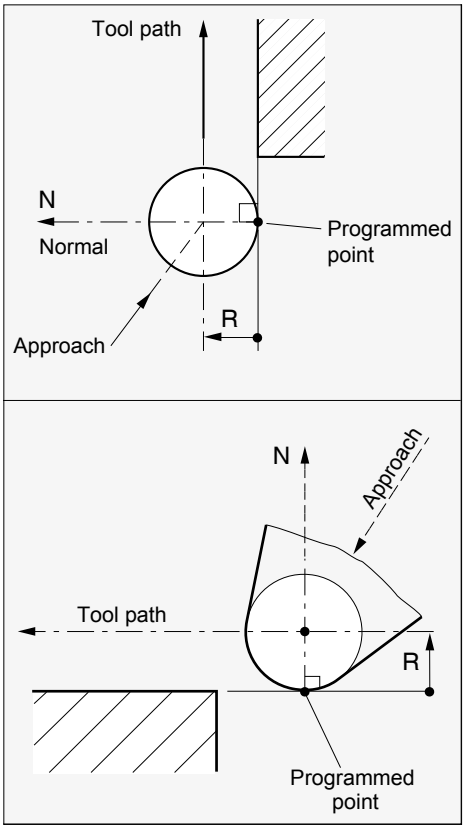
- M00 (program stop),
- M01 (optional program stop),
- M02 (end of program),
- G52 (programming with respect to the measurement origin),
- $0 (message transmission),
- L100 to L199 (program variables, see chapter 7.1),
- Parametric programming using E parameters.

5.9.1.1 Tool positioning



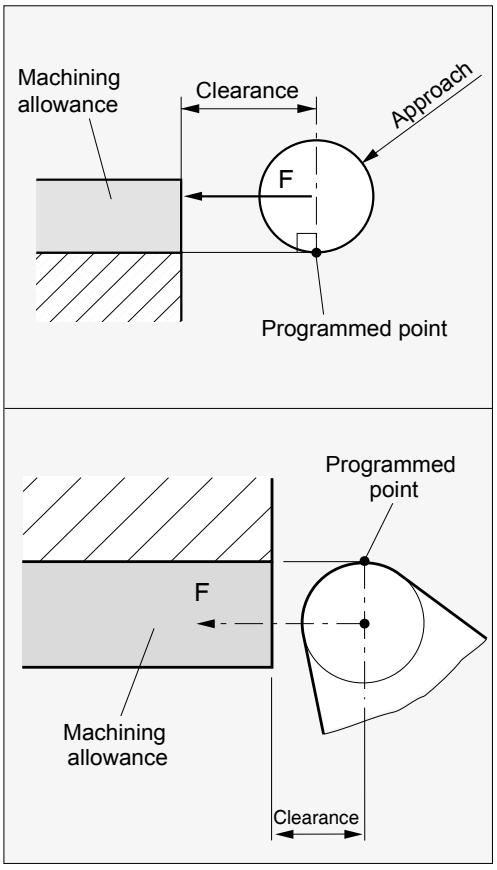
At the end of the first block programmed with radius correction (which must be linear), the tool center is positioned:

- on the normal (N) to the next path,
- offset from the programmed point by the value of the radius correction (R).



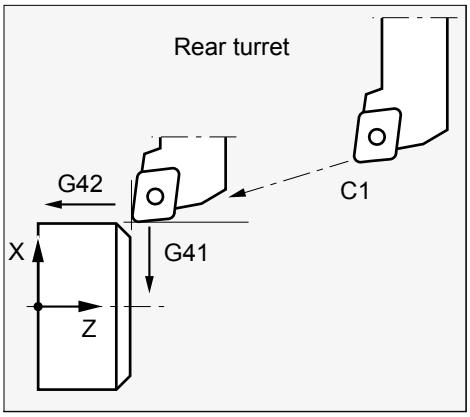
Precaution for positioning the tool

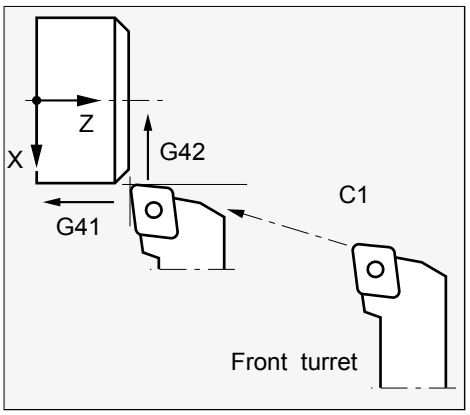
When rapid positioning, provide a clearance with a value higher than the expected tool radius.



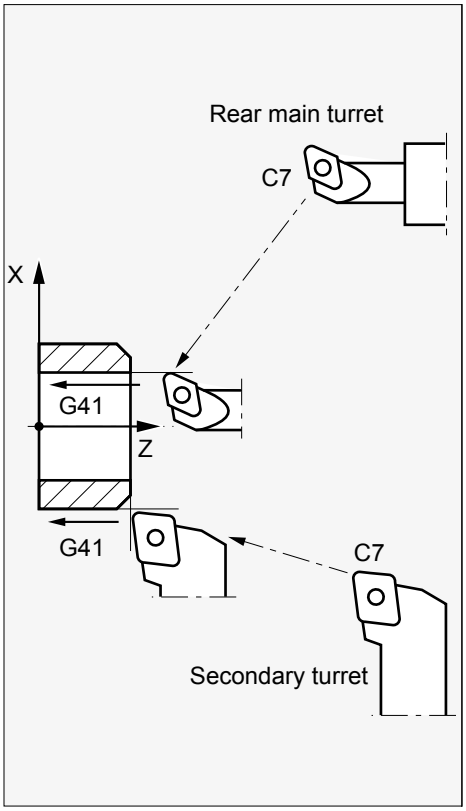
5.9.1.2 Radius correction according to the turret position

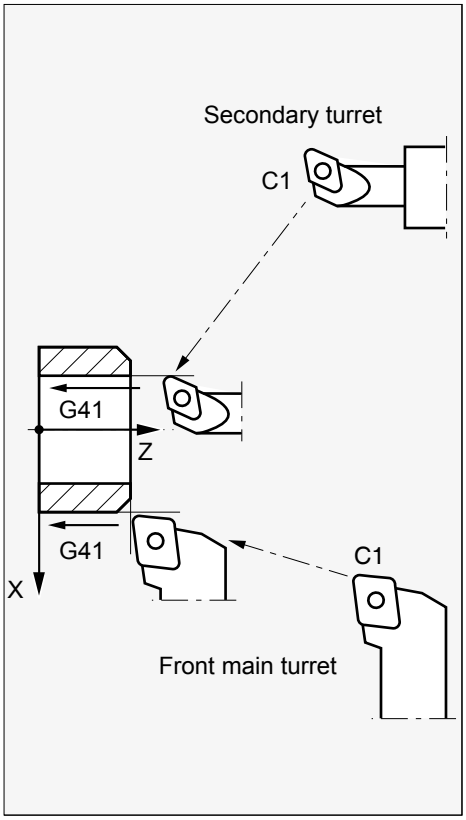
Lathe with one turret





Lathe with two mechanically coupled turrets
The tool tip orientation (C) is defined with reference to the main turret.

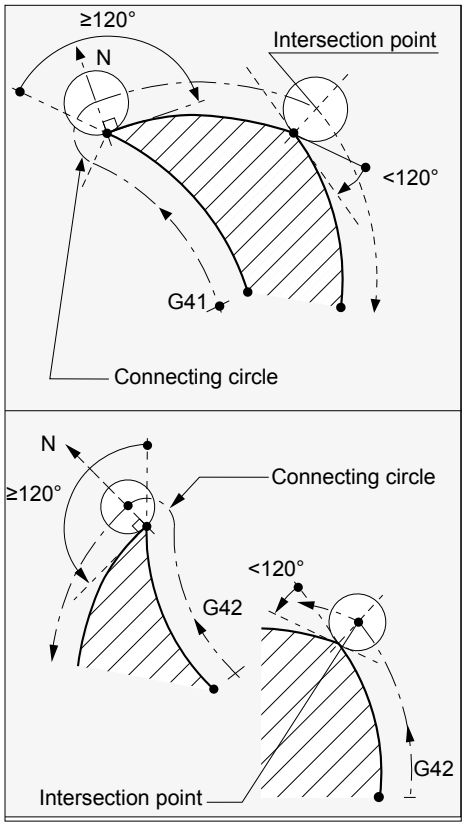


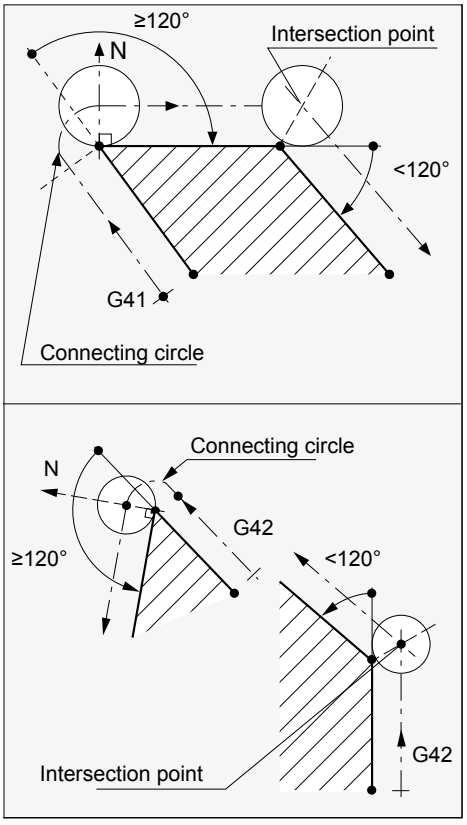


5.9.1.3 Tool outside the profile (line/line or circle/circle)

At the end of the block, the tool center is positioned:

- offset from the programmed point, on the normal to the next path (angle ≥ 120 degrees) after describing a connecting arc,
- on the point of intersection between the current and next path (angle < 120 degrees).



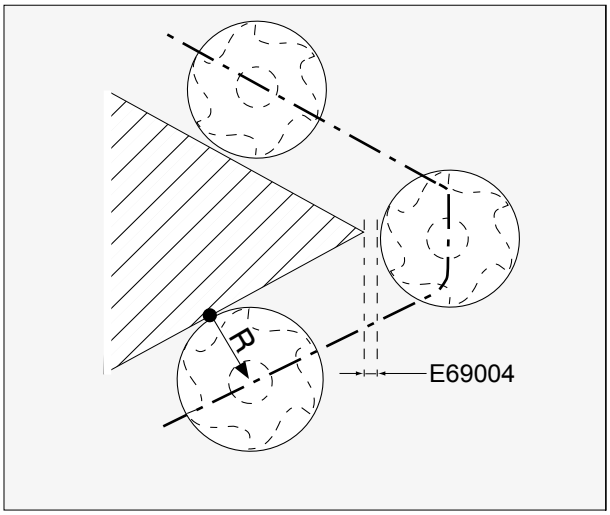


With this approach for an angle larger than 120°, the tool stays describes a connection arc while staying in contact with the part which is fine in most cases as it interferes at minima with the rest of the blank.

However in some cases (e.g. Wood working) this can cause some degradation of the angle. It is therefore possible to define a slight disengagement from the part before the connecting arc.

This disengagement value is programmed via parameter E69004 which should be programmed in G40 status. E69004 is channel specific and is automatically limited to 0.4 R.

This functionality is only available in the case of Line to Line transition.

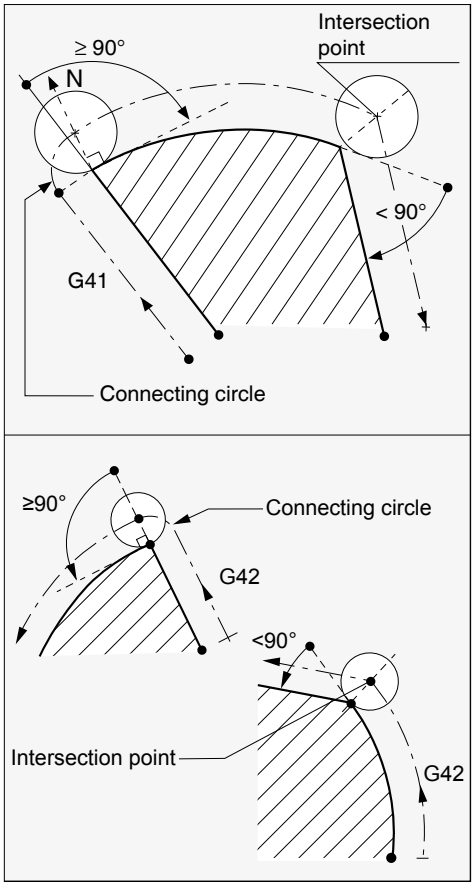


5.9.1.4 Tool outside the profile (line/circle)



At the end of the block, the tool center is positioned:

- offset from the programmed point on the normal to the next path (angle ≥ 90 degrees) after describing an intersecting arc,
- on the point of intersection between the current and next path with offset (angle < 90 degrees).

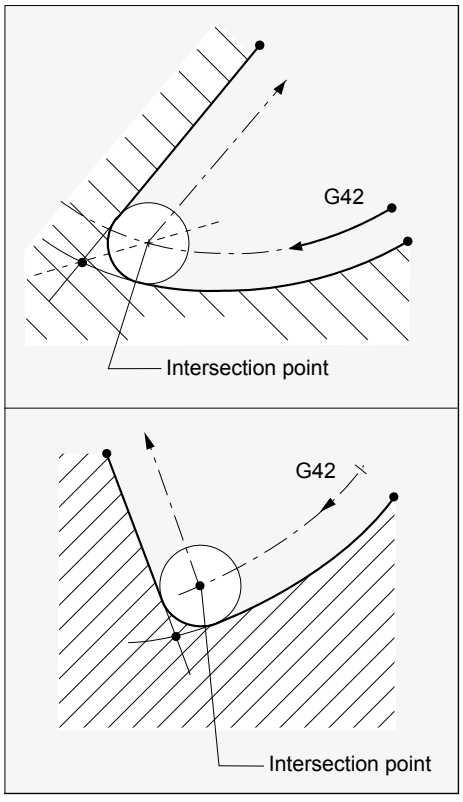


5.9.1.5 Tool inside the profile



The tool follows the path until its center reaches the intersection point between the current and next path with offset.

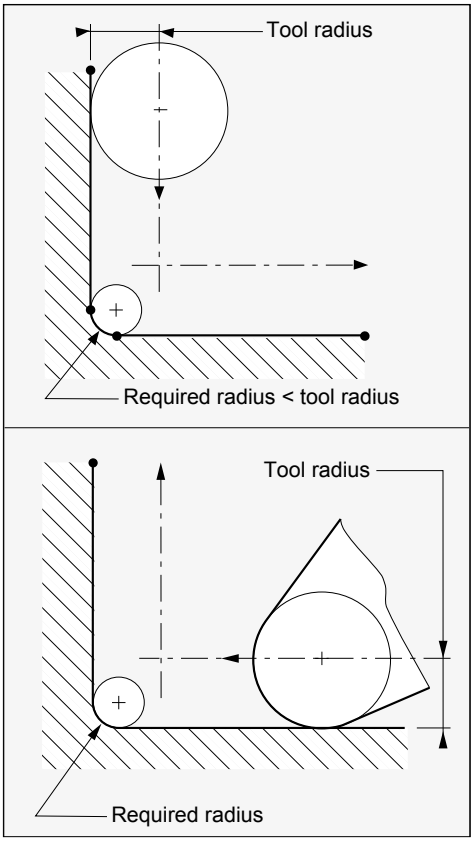
The programmed point is not reached: the radius of the tool leaves a fillet between the two consecutive paths.

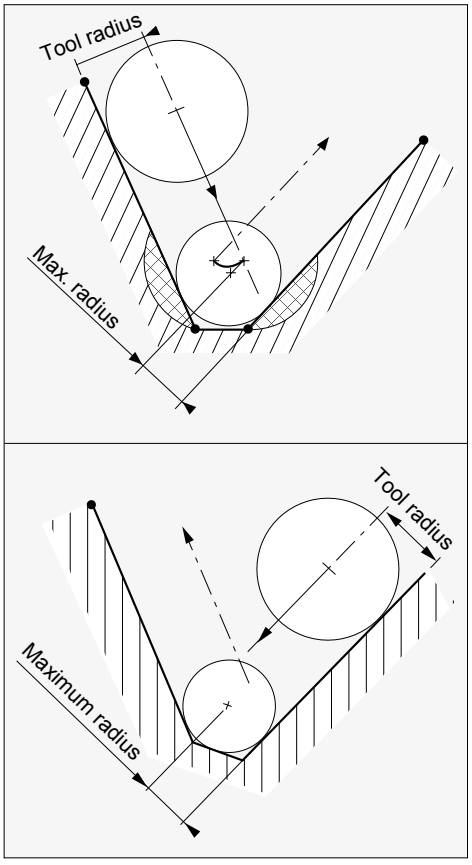


5.9.1.6 Tool inside the profile, special cases



When the tool size is too large to be tangent to one of the programmed paths (path radius less than the tool radius or path inaccessible), the system returns error message 149.





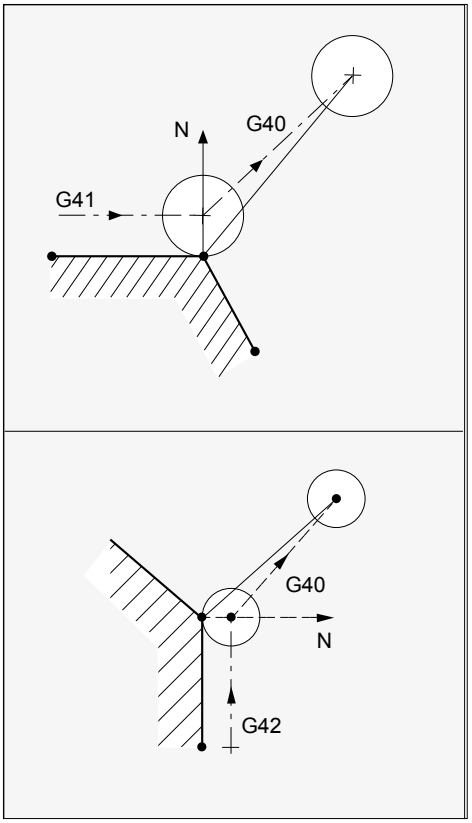
5.9.1.7 Tool disengagement



At the start of the first block programmed with tool radius offset cancelled, (must be a line), the tool center starts:

- from the normal to the previous path,
- offset from the programmed point by the value of the radius correction.

At the end of the move, the tool center coincides with the programmed point end point.

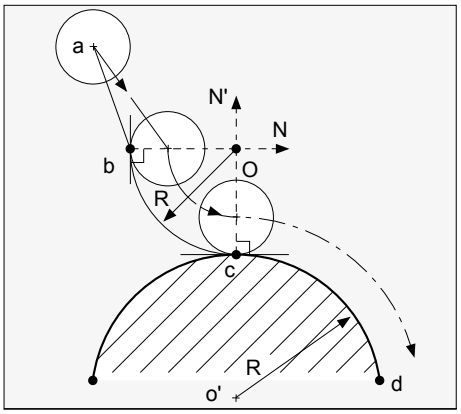




Examples

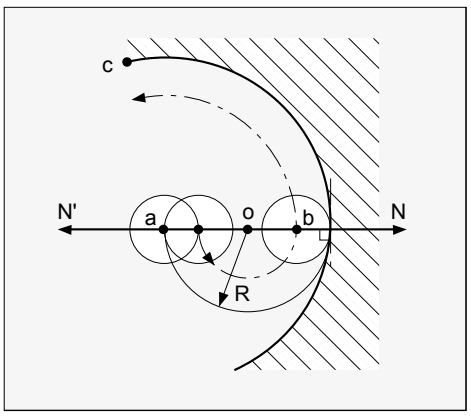
Engagement on an outside circle in G03

```
N.. ...
N.. D04
N40 Xa Ya Za
N50 G01 G41 Xb Yb F..
N60 G03 Xc Yc Io Jo F..
N70 G02 Xd Yd Io' Jo'
N..
```



Engagement on an inside circle in G03

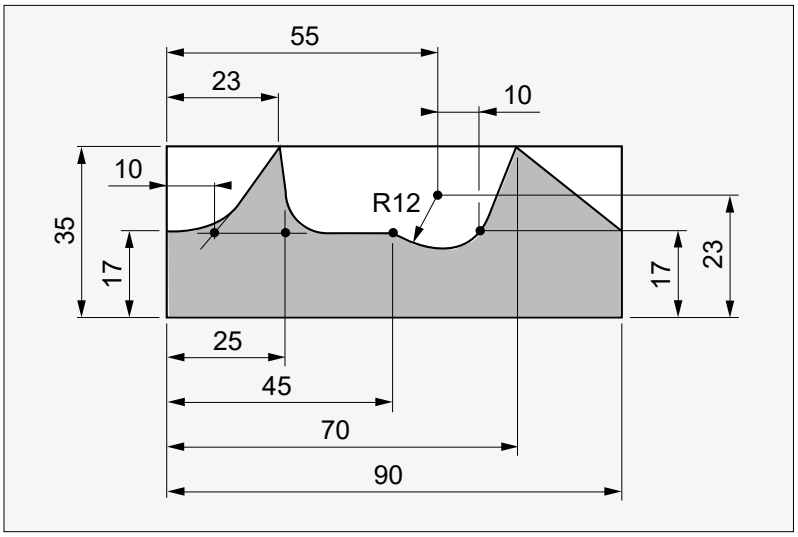
```
N.. ...
N.. D05
N50 G41 Xa Ya Za
N60 G03 Xb Yb Io Jo F..
N70 G03 Xc Yc Ia Ja
N..
```



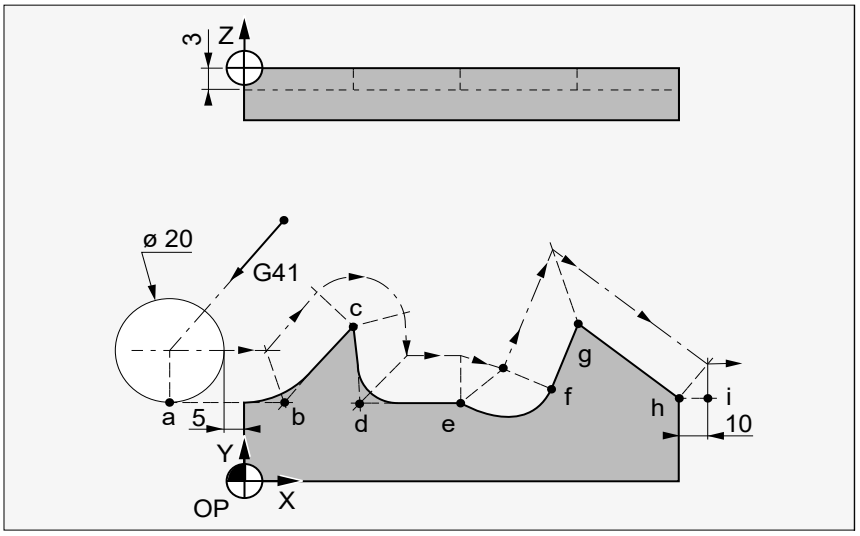


Contouring a profile with radius correction in the XY plane (G17)

Radius without dimension = Tool radius

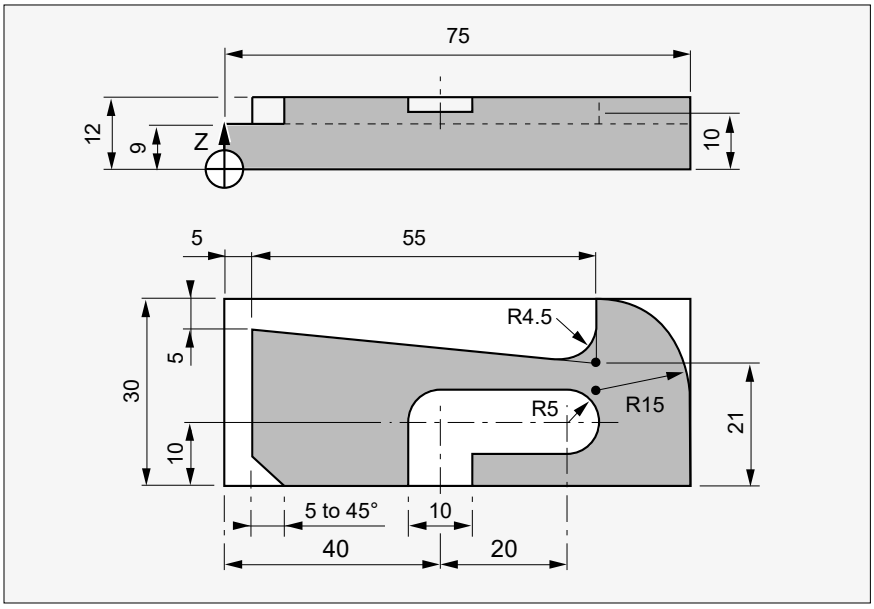


Tool paths (finishing)

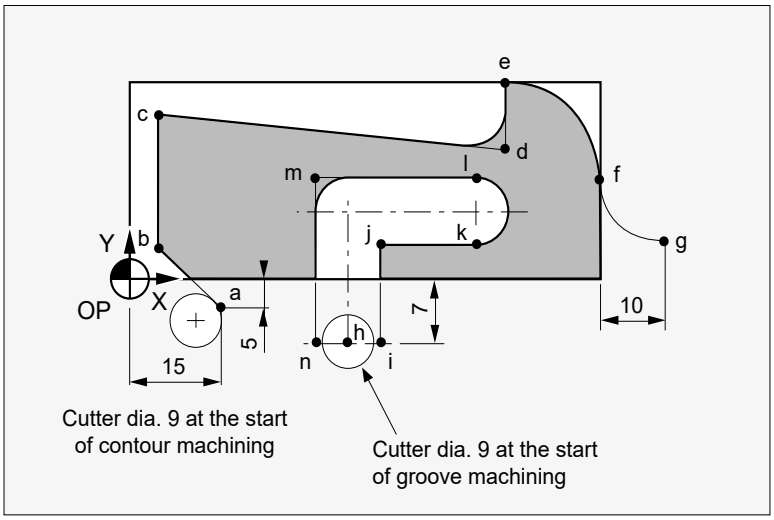



```
%30
N10 G00 G52 Z..
N20 T01 D01 M06 (CUTTER DIAMETER=20)
N30 S300 M03 M40
N40 G00 G41 X-15 Y17                    Point a, left radius offset
N50 Z-3                                 Tool position in Z
N60 G01 X10 F100 M08                    Point b
N70 X23 Y35                             Point c
N80 X25 Y17                             Point d
N90 X45                                 Point e
N100 G03 X65 Y17 R12 F50                Point f
N110 G01 X70 Y35 F100                   Point g
N120 X90 Y17                            Point h
N130 G40 X100                           Point i, radius offset cancel
N140 G00 G52 Z.. M05 M09                Retraction
N150 M02
```

Contouring a profile and a groove with radius correction in the XY plane (G17), with tool retraction and repositioning



Tool paths

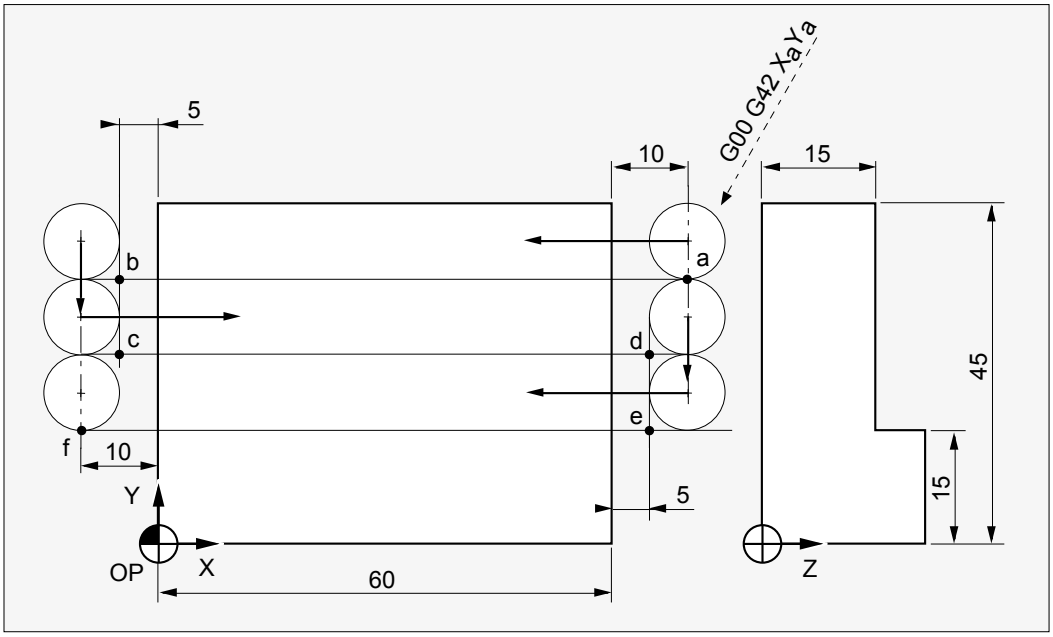



```
%70
N10 G00 G52 Z..
N20 T03 D03 M06 (CUTTER DIAMETER=9)
N30 S700 M40 M03
$0 PROFILE
N40 G00 G41 X15 Y-5                    Point a in XY, left radius offset
N50 Z9 M08                             Tool position in Z
N60 G01 X5 Y5 F120                     Point b
N70 Y25                                Point c
N80 X60 Y21                            Point d
N90 Y30                                Point e
N100 G02 X75 Y15 R15                   Point f
N110 G03 X85 Y5 R10                    Point g
```

```
N120 G00 G40 Z15 M09          Retraction, radius offset cancel
$0 GROOVEN130 X40 Y-7         Point h in XY, left radius offset
N140 Z10 M08                  Tool position in Z
N150 G41 X45                  Point i
N160 G01 Y5                   Point j
N170 X60                      Point k
N180 G03 X60 Y15 R5 F50       Point l
N190 G01 X35 F120             Point m
N200 Y-10                     Point n
N210 G00 G52 Z.. M05 M09      Retraction
N220 G40 G52 X.. Y..          Radius offset cancel
N230 M02
```



Facing with radius correction in the XY plane (G17) with change from G41 to G42



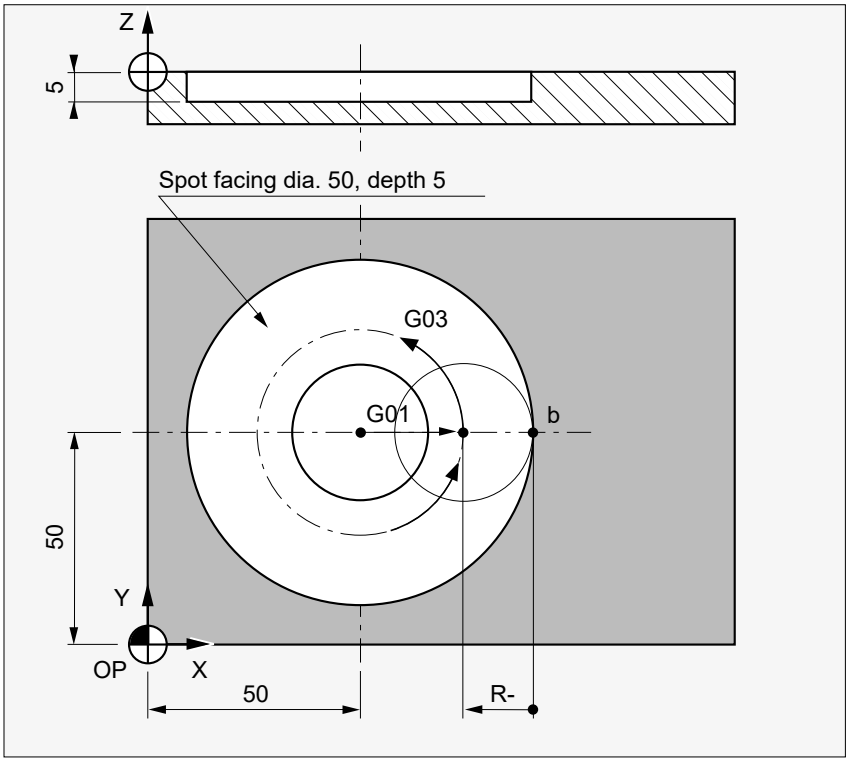

```
%72
N10 G00 G52 Z..
N20 T25 D25 M06
N30 S600 M40 M03
N40 G00 G42 X70 Y35          Point a, right radius offset
N50 Z15                      Tool position in Z
N60 G01 X-5 F200 M08         Point b
N70 Y25                      Point c
N80 G41 X65                  Point d, left radius offset
N90 Y15                      Point e
N100 G42 X-10                Point f, right radius offset
N110 G00 Z250 M09
N120 G40 G52 X.. Y.. M05     Radius offset cancel
N130 M02
```



Spot facing with radius correction applicable to positioning with stop at programmed feed rate (R+/R-)

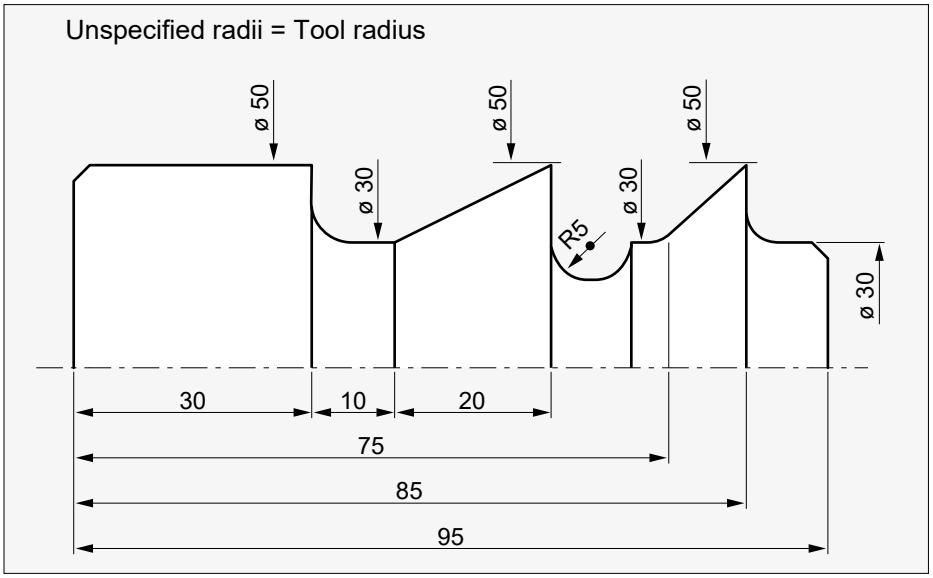
Cutter programmed: diameter = 16.

Cutter declared in correction D12: diameter = 20
(i.e. radius R = 10).

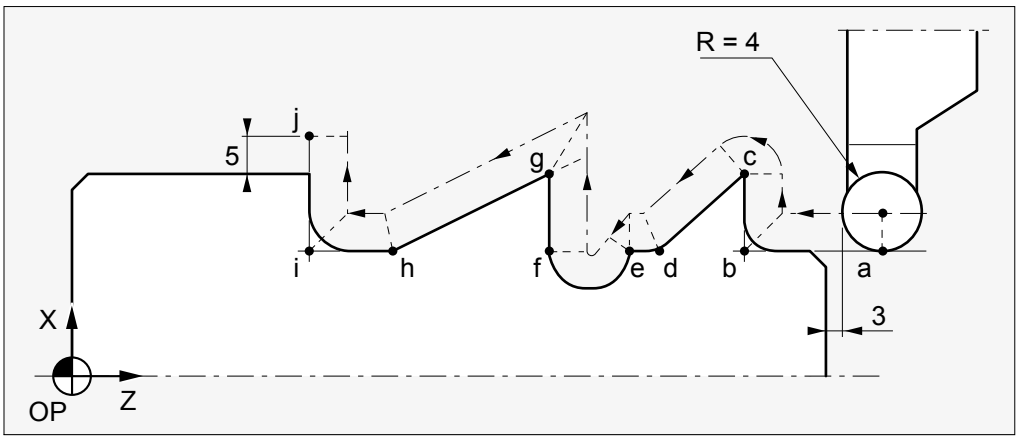



```
%13
N10 G00 G52 Z..
N20 T12 D12 M06 (CUTTER DIAMETER 16)
N30 S300 M40 M03
N40 G00 X50 Y50 Z3                    Point a, approach
N50 G01 Z-5 F50 M08
N60 R- X75 F100                       Stop before programmed point
N70 G41 G03 X75 Y50 I50 J50 F150      Point b
N80 G00 G40 X50
N90 Z50
N..
```

Contouring an external profile with right radius correction

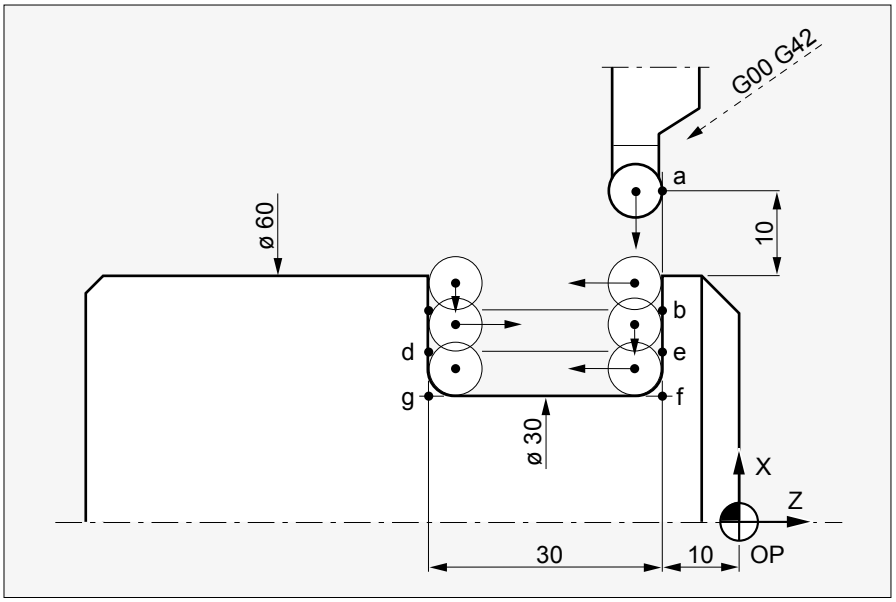


Machining paths (finishing)



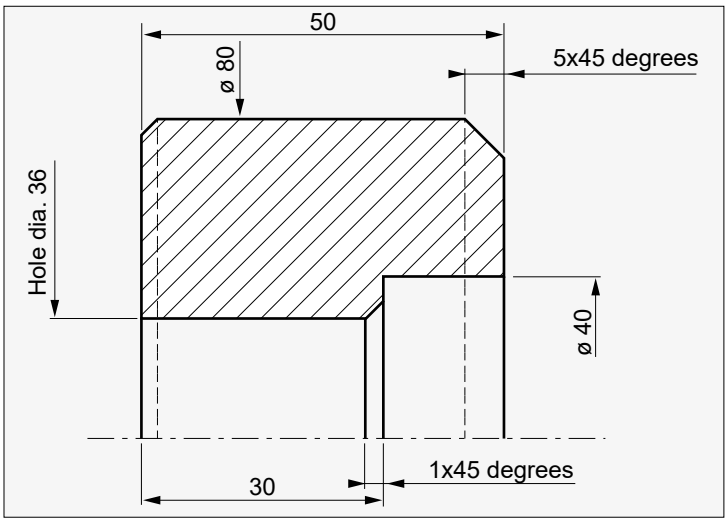

```
%37
N10 G00 G52 X.. Z..                    Tool change position
N20 T03 D03 M06 (COPYING TOOL R=4)
N30 S900 M40 M04
N40 G95 F0.2
N50 G00 X30 Z102                       Point a
N60 G92 S3500
N70 G96 S200
N80 G01 Z85                            Point b
N90 X50                                Point c
N100 X30 Z75                           Point d
N110 Z70                               Point e
N120 G02 X30 Z60 R5 F0.05              Point f
N130 G01 X50 F0.2                      Point g
N140 X30 Z40                           Point h
N150 Z30                               Point i
N160 X60                               Point j
N170 G97 S900
N180 G00 G40 G52 X.. Z.. M05           Radius offset cancel
N190 M02
```

Machining a groove by reciprocating cuts with alternating radius correction (G41-G42)

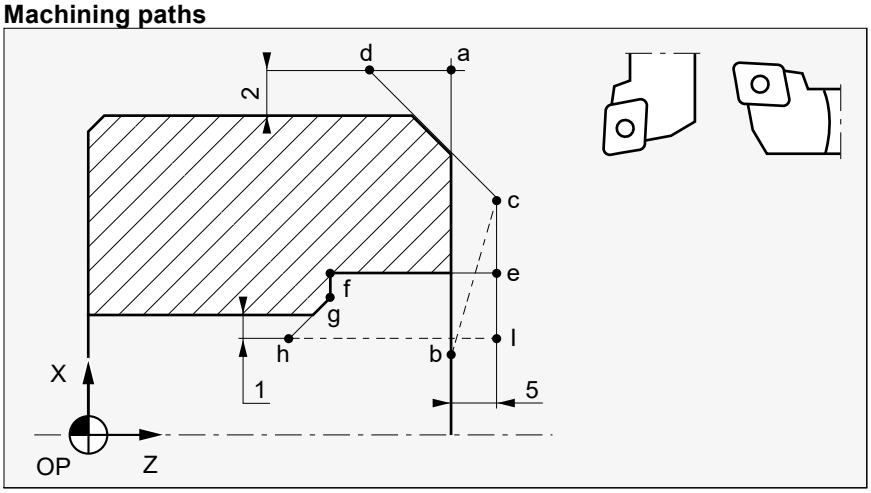



```
%85
N10 G00 G52 X.. Z..                          Tool change position
N20 T01 D01 M06 (COPYING TOOL R=4)
N30 S900 M40 M04
N40 G95 F0.1
N50 G00 G42 X80 Z-10                         Point a, right correction
N60 G92 S3000
N70 G96 S100
N80 G01 X50                                  Point b
N90 Z-40 F0.2                                Point c
N100 G41 X40 F0.1                            Point d, left correction
N110 Z-10                                    Point e
N120 G42 X30                                 Point f, right correction
N130 Z-40 F.2                                Point g
N140 X80
N150 G97 S900
N160 G00 G40 G52 X.. Z.. M05                 Radius offset cancel
N170 M02
```

External machining and internal contouring after tool retraction and positioning



Machining paths




```
%72
N10 G00 G52 X.. Z..                          Tool change position
N20 T01 D01 M06 (OUTSIDE TOOL R=0.8)
N30 S900 M40 M04
N40 G95 F0.12
N50 G00 G41 X84 Z50                          Point a, left correction
N60 G92 S3500
N70 G96 S180
N80 G01 X30                                  Point b
N90 G00 G42 X60 Z55                          Point c, right correction
N100 G01 X84 Z43 F0.08                       Point d
N110 G97 S900
N120 G00 G40 G52 X.. Z..                     Radius offset cancel
N130 T07 D07 M06 (BORING TOOL R=0.4)
N140 G41 X40 Z55                             Point e, left correction
N150 G96 S100
N160 G01 Z30 F0.1                            Point f
N170 X38                                     Point g
N180 X34 Z28                                 Point h
N190 G00 Z55                                 Point i
N200 G97 S900
N210 G40 G52 X.. Z.. M05                     Radius offset cancel
N220 M02
```

5.9.2 Enhanced tool radius correction

In the standard G41/G42, the system can detect interference between the block in execution and the next. In this case the NCK will raise an error and does not provide a tool center path corrected.

The goal of the extended radius correction is to avoid such interference on several blocks ahead by modifying if necessary the tool center path instead of raising an NCK error.

Syntax

G41+|42+ followed by numerical value or parameter

The numerical value either explicit (max 2 digits) or defined by a variable (L, E, symbolic) specifies, from the current block under analysis, the number of blocks used to check a possible interference and to apply (if necessary) correction of the tool center path.

The range allowed for this numerical value is [0, …,15]

0 or 1	NO activation of the correction path. The NCK runs the standard G41/G42 function.
[2, …,15]	Activation of the correction path with the specified number of blocks ahead.

The NCK should be in status G40 before enabling the function.

The function applies after the first block where G41 or G42 is programmed (engagement block to take into account the tool radius offset compensation).

The 2 axes of the interpolation plane on which the function applies are the 2 axes currently selected before or in the block where the function is enabled.

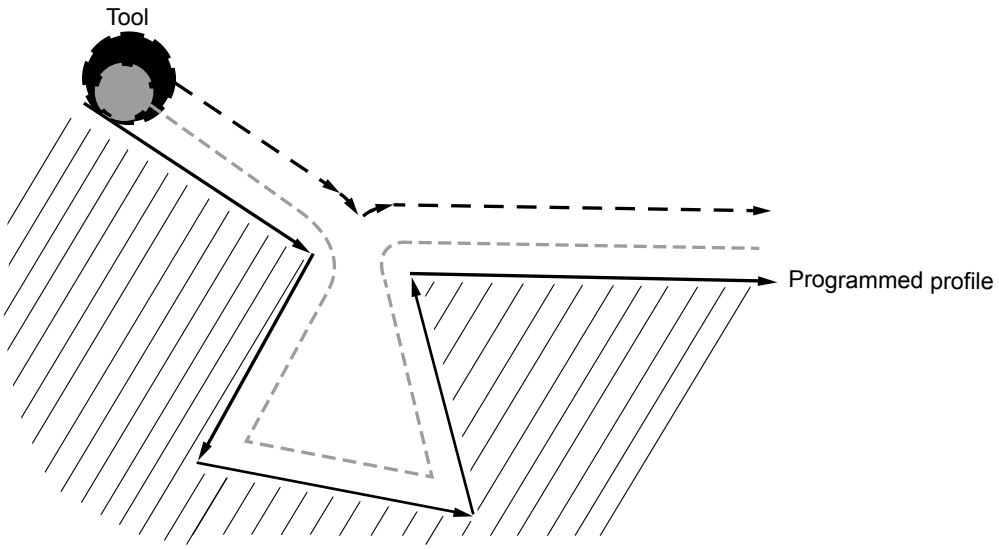
Limitations

Contrary to standard G41/G42, the following possibilities are forbidden when the function is already enabled (otherwise the NCK raises an error):

1) Switch from G41 to G42 or conversely (reason: it doesn't make sense to change the direction of the tool offset compensation among several blocks ahead).
2) Change one axis of the interpolation plane: for instance change from X (primary axis) to U (secondary axis) and conversely (reason: it doesn't make sense to change one axis - or the 2 axes - of the interpolation plane on which applies the tool radius offset compensation among several blocks ahead).
   On the other hand the programming of both primary and secondary axes is allowed: in this case the 2 axes of the interpolation remain unchanged.
3) Modify mirroring of one of the 2 axes of the interpolation plane (the consequence is to change the direction of the tool radius offset compensation – equivalent to switch from G41 to G42 or conversely).
4) For turning operation, modify front / rear turret (equivalent to X axis mirroring).
5) Insert polynomial interpolation.
6) Changing on the fly the number of blocks to look ahead. In this case it is necessary to switch to G40 status (disable the function) and afterwards re-enable the function by specifying the new number of blocks to look ahead.

Of course as for G41/G42 standard the programming of features requiring hold of block preparation are forbidden (e.g. writing E parameters, M0, …).

Principle



Should the tool diameter be small enough (not the case here) to enter the pocket a path similar to the one in grey will be executed otherwise the pocket will be skipped and the trajectory in black followed.

5.9.3 3D Tool compensation

5.9.3.1 Concept of surface and normal vector



The types of surfaces machined using G29 (irregular surfaces) and the programming requirements (normal vector) mean that tridimensional correction can only be used with machining programs in symbolic language.

Machining is executed by scanning "TABCYL" (word of the APT language defining a tabulated cylinder) by consecutive linear interpolations.

The consecutive program blocks contain the coordinates of the points. The unit normal vector in each point always points outwards from the part.

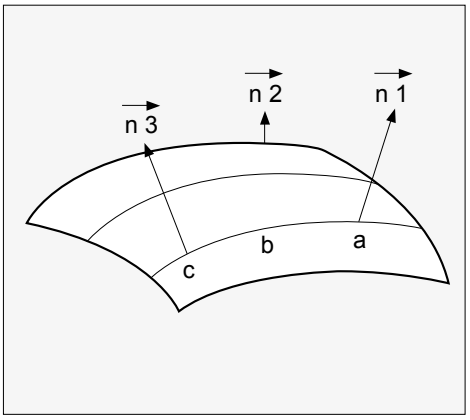
The spacing between the programmed points varies according to the required machining accuracy.

For 3-axis correction, the material vector defined by components P, Q and R must have a length of 100 mm within 1 mm (i.e. +/-0.1%). Otherwise, the system returns error message 145.

The tool correction is applied along the normal vector below:

$$\sqrt{P^2 + Q^2 + R^2} = 1000 \text{ mm}$$

a	X1	b	X2	c	X3
	Y1		Y2		Y3
	Z1		Z2		Z3



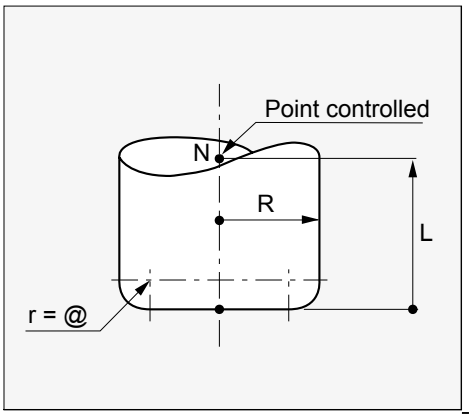
With 5-axis correction, the material vector defined by P, Q and R and the tool direction defined by I, J and K can have any length (the lengths are normalized by the system), but all three components of both vectors must be programmed in each block. Otherwise, the system returns error message 146.

5.9.3.2 Toroidal and spherical tools

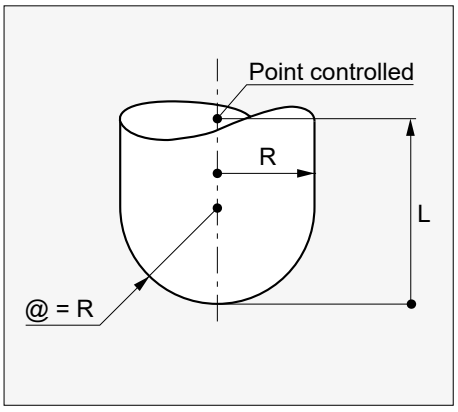
For tridimensional correction, the three corrections to be declared are:
- L.. : tool length,
- R.. : tool radius,
- @..: tool tip radius.

The tool radius "R" must not be confused with the R component of the normal vector.

Toroidal tool



Spherical tool



5.9.3.3 Double twist Machine Head



Head used with 5-axis correction.

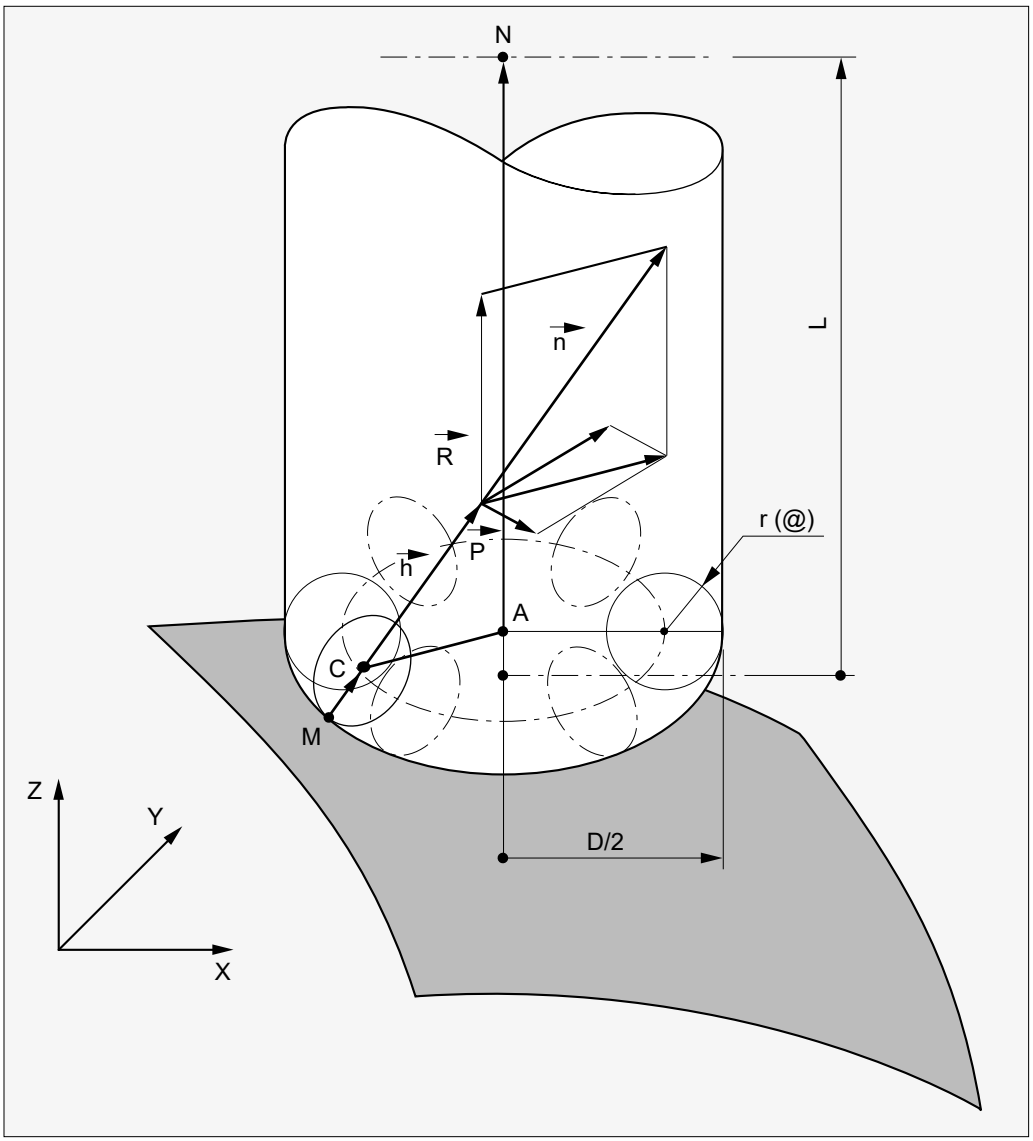
The twist head angles α and β are defined by a pair of rotary axes defined in the NC, i.e.:
- A and B on X and Y,
- A and C on X and Z,
- B and C on Y and Z.

5.9.3.4 Geometric transformations with 3-Axis Correction



The system computes the tool reference position «N» according to the tangency point «M» and the vector normal to the surface « $\vec{n}$ ».

- $\overrightarrow{MN} = \overrightarrow{MC} + \overrightarrow{CA} + \overrightarrow{AN}$
- $\overrightarrow{MC}$: Vector with length r(@) collinear with vector $\vec{n}$.
- $\overrightarrow{CA}$: Vector with length (D/2 - r) collinear with the projection of vector $\vec{n}$ on the interpolation plane (XY in the figure).
- $\overrightarrow{AN}$: Vector with length (L - r) collinear with the tool axis orientation vector (defined by G16..).



Computation mode

$$\overrightarrow{MC}\begin{vmatrix} \frac{r \cdot P}{\sqrt{P^2+Q^2+R^2}} = r \cdot P \\ \frac{r \cdot Q}{\sqrt{P^2+Q^2+R^2}} = r \cdot Q \\ \frac{r \cdot R}{\sqrt{P^2+Q^2+R^2}} = r \cdot R \end{vmatrix}$$

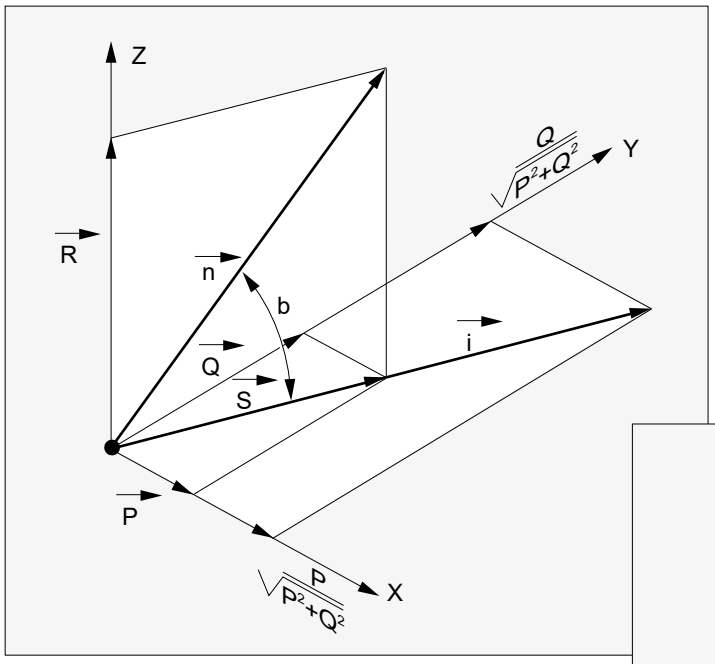
$$XN = XM + \frac{r.P}{1000} + \left(\frac{D}{2} \pm r\right)\frac{P}{\sqrt{P^2+Q^2}}$$

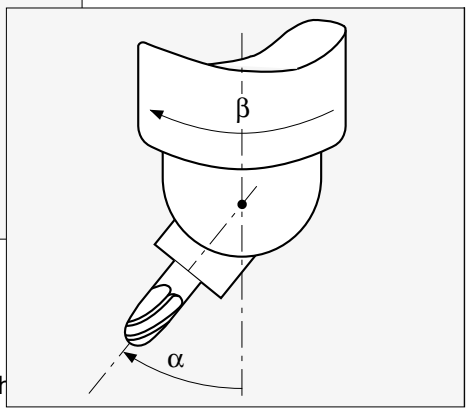
$$YN = YM + \frac{r.Q}{1000} + \left(\frac{D}{2} \pm r\right)\frac{Q}{\sqrt{P^2+Q^2}}$$

$$ZN = ZM + \frac{r.R}{1000} + (L \pm r)$$

$$\overrightarrow{CA}\begin{vmatrix} \left(\frac{D}{2} - r\right)\frac{P}{\sqrt{P^2+Q^2}} \\ \left(\frac{D}{2} - r\right)\frac{Q}{\sqrt{P^2+Q^2}} \\ 0 \end{vmatrix}$$

$$\overrightarrow{AN}\begin{vmatrix} 0 \\ 0 \\ (L - r) \end{vmatrix}$$





Reminder

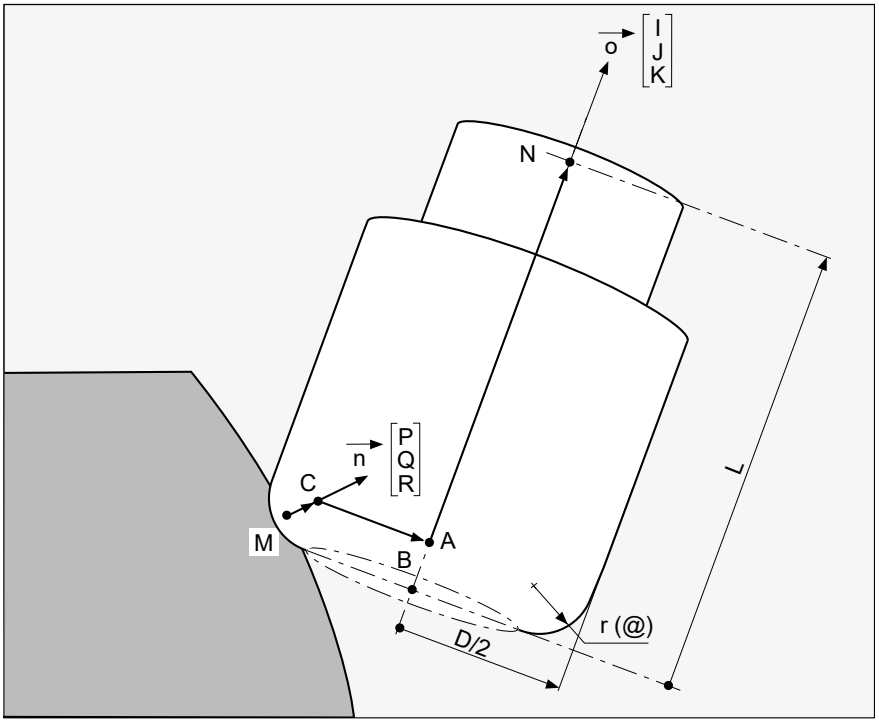
In the above equations, P, Q and R are the components of th

5.9.3.5 Geometric transformations with 5-Axis Correction

The system computes the tool reference position «N» according to the tangency point «M», the vector normal to the surface " $\vec{n}$ " and the tool orientation vector "$\vec{o}$ " (same computation principle as for 3-axis correction).

- M: programmed point
- B: point displayed on the current position coordinate page (AXIS) with respect to OP
- N: point controlled by the system.





Examples

3-axis correction

```
N..
N.. G01 ...
N240 D17
N250 G29 X.. Y.. Z.. P.. Q.. R..
N260 X.. Y.. Z.. P.. Q.. R..
N..
N..
N890 G00 G40 X.. Y..
N..
```

5-axis correction

Twist head axes: A and C.

```
N..
N.. G01 ...
N320 D15
N330 G29 X.. Y.. Z.. P.. Q.. R.. I.. J.. K.. A.. C..
N340 X.. Y.. Z.. P.. Q.. R.. I.. J.. K..
N350 X.. Y.. Z.. P.. Q.. R.. I.. J.. K..
N360 X.. Y.. Z.. P.. Q.. R.. I.. J.. K.. A.. C..
N..
N..
N620 G00 G40 X.. Y..
N..
```

5.9.3.6 Tridimensional tool correction with toroidal or spherical tool ► G29



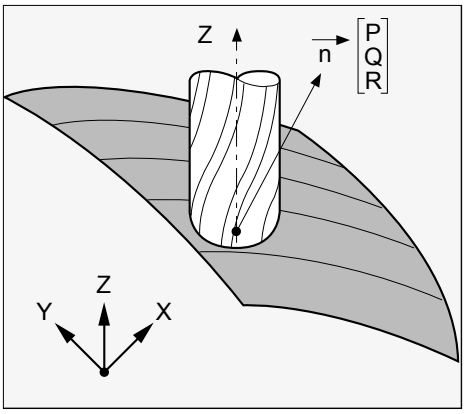
G29	Tridimensional tool correction (3 or 5 axes) with toroid or spherical tool.

Tridimensional tool correction allows machining of tridimensional linear paths taking into account the dimensions of the toroidal or spherical tool used.

3-axis tool correction

For 3-axis correction, the tool axis is parallel to one of the axes of the basic coordinate system defined by the tool axis orientation function G16 ... .

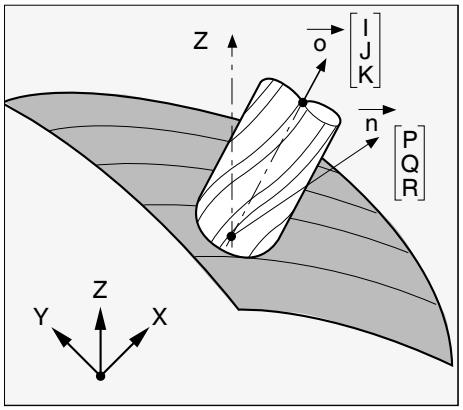
With each programmed point is associated the «material» vector « $\vec{n}$ » normal to the surface to be machined, defined by its P, Q and R coordinates.



5-axis tool correction

For 5-axis correction, the tool axis can be inclined when the machine is equipped with a double twist machining head.

With each programmed point are associated the vector « $\vec{n}$ » normal to the surface to be machined, defined by its P, Q and R coordinates, and the tool orientation vector « $\vec{o}$ », defined by its I, J and K components, plus the twist head angles where required.



Syntax

N.. [D..] [G01] **G29** X.. Y.. Z.. P.. Q.. R.. [I.. J.. K..] [A.. / B.. / C..]

D..	Correction number.
G01	Linear interpolation.
G29	Tridimensional tool correction with toroidal or spherical tool.
X.. Y.. Z..	End point.
P.. Q.. R..	Components of the normal vector $\vec{n}$ (material vector) whose origin is defined in the block by the X, Y, Z coordinates of the end point (mandatory in each block).
I.. J.. K..	Components of the tool orientation vector $\vec{O}$ with 5-axis correction (mandatory in each block).
A.. / B.. / C..	Twist head angles with 5-axis correction: A: angle on X, B: angle on Y, C: angle on Z.

Properties of the Function

Function G29 is modal. None of the arguments associated with the function are modal.

Cancellation

Function G29 is cancelled by function G40 or one of functions G41 or G42.

Notes

The presence or absence of vector I J K in a block allows the distinction to be made between 3-axis and 5-axis tool corrections.

For tridimensional correction, the two axes of the basic three-axis system other than the tool axis may be impacted by tool radii R and @, be these axes primary, secondary, carried or independent.

5.9.3.7 Tridimensional tool correction with cylindrical Tool and RTCP ► G43+/-



G43+/-	Tridimensional tool correction (3 or 5 axes) with cylindrical tool.

Syntax

N.. [D..] [G01] **G43+/-** X.. Y.. Z.. [I.. J.. K..] [A.. B.. C..]

D..	Correction number.
G01	Linear interpolation.
G43	Tridimensional tool correction with cylindrical tool.
X.. Y.. Z..	Point programmed on the surface.
P.. Q.. R..	Components of the material vector (normed at 1000) that positions the center of the tool tip with respect to the programmed point (the offset is obtained by the vector components divided by 1000 and multiplied by the tool radius)..
I.. J.. K..	Components of the tool vector normed at 1 by the system giving the tool axis orientation (see Notes).
A.. / B.. / C..	Twist head angles with 5-axis correction: A: angle on X, B: angle on Y, C: angle on Z.

Properties of the Function

Function G43+/- is modal.

Cancellation

Function G43+/- is cancelled by function G40.

Notes

G43 +/- is intended to simplify the programming by making the arguments I.J.K and P Q R optional.

It can be used when the material vector is assumed perpendicular to the trajectory. The sign + / - defines the direction of the offset with respect to the part.

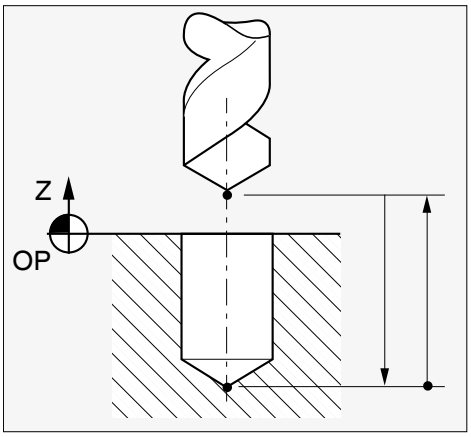
For information on the RTCP function, see the Flexium⁺ Extended Programming Manual M00055.

5.10 Canned cycles

5.10.1 Cycle overview ► G8x



G8x (and G31)	Canned cycles along the tool axis.



Axes programmable with basic cycles:
- X, Y, Z primary axes,
- U, V, W secondary axes,
- rotary axes A, B or C are used only for positioning.

The cycles are currently executed only on the main spindles.

Syntax (XY plane)

N.. [G17] **G8**x [X.. Y..] Z.. [ER..] [EH..] F..

G17	XY plane selection.
G8x	Canned cycle.
X.. Y..	Tool position in the plane.
Z..	End point on the machining axis.
ER..	Dimension of the approach (or retraction) plane in the machining axis.
EH..	Dimension of the impact plane in the machining axis.
F..	Feed rate in the cycle.

Properties of the Functions

Functions G8x are modal.

Cancellation

Function G8x is cancelled by function G31, G80 or another function G8x.

Notes

When a cycle (G8x and G31) is programmed, the system must be in state G40 (G41 or G42 tool radius offset cancelled).



The use of program variables L900 to L959 (see chapter 7) is not recommended in a program including canned cycles, since some of these variables could be overwritten when a cycle is called. The tool axis (G16) must always be perpendicular to the interpolation plane otherwise, error 890 extension 0 will be triggered.

Dimensions ER and EH

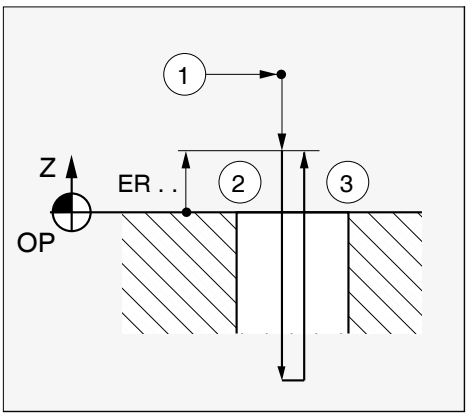
Dimension ER defines the approach (or retraction) plane on the machining axis. It is assigned to the tool axis.

Dimension EH defines the impact plane on the machining axis. It is assigned to the tool axis.

EH must always be programmed in the same block of the cycle as ER.

Cycle steps with ER (without EH)

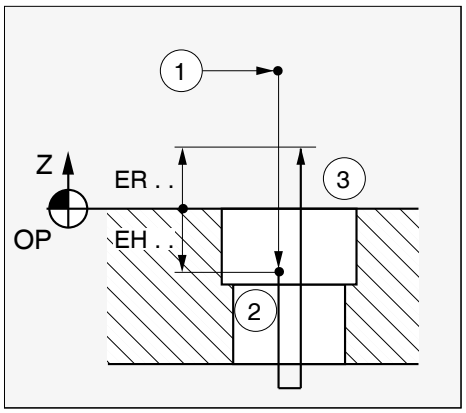
- Step 1: Rapid positioning (linear or circular) In the plane then on ER..
- Step 2: Tool penetration to the value programmed along the tool axis.
- Step 3: Retraction along the tool axis up to ER..



- ER.. not programmed:
  The previous value programmed along the tool axis is used for approach.
- ER.. programmed alone:
  The tool is simply positioned along the tool axis (ie. no cycle).

Cycle steps with EH and ER

- Step 1: Rapid positioning (linear or circular) in the plane, then on EH..
- Step 2: Tool penetration down to the value programmed along the tool axis (Z).
- Step 3: Retraction along the tool axis (Z) up to ER..



- EH and ER.. programmed:
  EH differentiates between the impact plane and the back-off plane.
- EH.. not programmed and ER.. programmed:
  The value of ER is used (ER = EH).

The above descriptions of the cycle steps with ER alone and with ER and EH take into account only the start and end planes. For the detail of these steps, refer to the cycle concerned.

5.10.2 Cancellation of a canned cycle ► G80



G80	Canned cycle cancel.

This function is used to cancel canned cycles.

Syntax

N.. **G80**

G80	Canned cycle cancel.

Property of the Function

Function G80 is modal. It is the default function.

Cancellation

Function G80 is cancelled by one of functions G31, G81-G89.

Notes

Including function G80 in a cycle subroutine makes the cycle nonmodal.

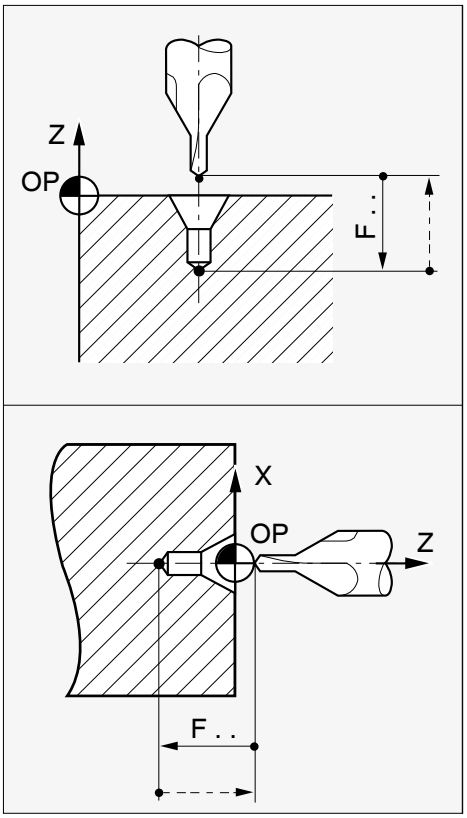
Example

```
N..
N120 G00 X.. Y.. Z..        Tool positioning
N130 G81 Z-10 F100          Drilling cycle
N140 G80 G00 Z200           Cycle cancel
N..
```

5.10.3 Center drilling cycle ► G81



G81	Center drilling cycle.



Syntax (XY plane)

N.. [G17] **G81** [X.. Y..] Z.. [ER..] [EH..] [F..]

G17	XY plane selection.
G81	Center drilling cycle.
X.. Y..	Tool position in the plane.
Z	End point on the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
F..	Feed rate in the cycle.

Property of the Function

Function G81 is modal.

Cancellation

Function G81 is cancelled by one of functions G31, G80, G82-G89.

Cycle Steps

- Step 1: Rapid positioning in the plane.
- Step 2: Infeed at feed rate F..
- Step 3: Rapid retraction along the tool axis.

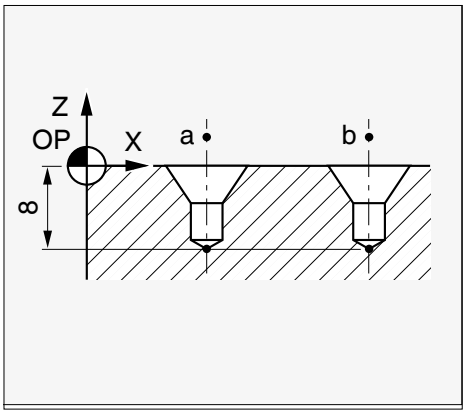


Example

```
N.. ...
N50 G00 Xa Ya Za
N60 G81 Z-8 F80
N70 Xb Yb
N80 G80 G00 Z..
N..
```

or

```
N.. ...
N60 G81 Xa Ya ERa Z-8 F80
N70 Xb Yb
N80 G80 G00 Z..
N..
```





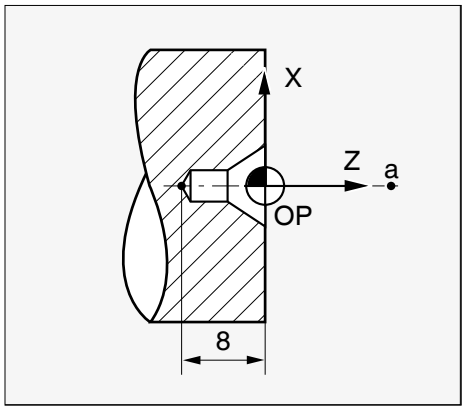
Example

Execution of center drilling

```
N.. ...
N50 G00 Xa Za
N60 G94 (or G95) F..
N70 G81 Z-8
N80 G80 G00 X.. Z..
N..
```

or

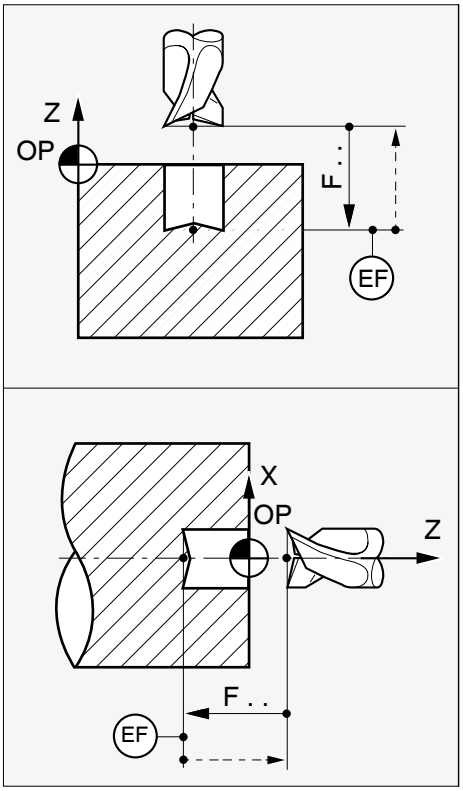
```
N..
N50 G94 (or G95) F..
N60 G81 Xa ERa Z-8
N70 G80 G00 X.. Z..
N..
```



5.10.4 Counterboring cycle ► G82



G82	Counterboring cycle.



Syntax (XY plane)

N.. [G17] **G82** [X.. Y..] Z.. [ER..] [EH..] EF.. [F..]

G17	XY plane selection.
G82	Counter boring cycle.
X.. Y..	Tool position in the plane.
Z..	End point on the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
EF..	Mandatory dwell time in seconds (maximum 99.99 s, format EF022).
F..	Feed rate in the cycle.

Property of the Function

Function G82 is modal.

Cancellation

Function G82 is cancelled by one of functions G31, G80-G81, G83-G89.

Cycle steps

- Step 1: Rapid positioning in the plane.
- Step 2: Infeed at feed rate F..
- Step 3: Dwell at end of drilling or counterboring.
- Step 4: Rapid retraction along the tool axis.

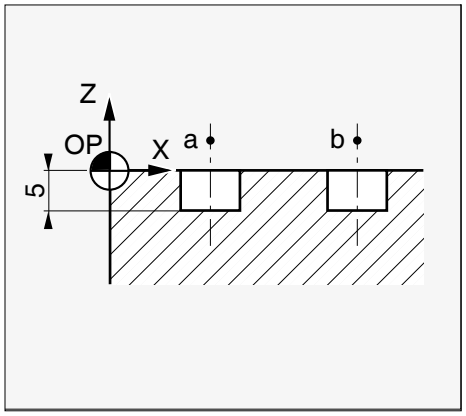


Example

```
N.. ...
N50 G00 Xa Ya Za
N60 G82 Z-5 EF2 F60
N70 Xb Yb
N80 G80 G00 Z..
N..
```

or

```
N.. ...
N60 G82 Xa Ya ERa Z-5 EF2 F60
N70 Xb Yb
N80 G80 G00 Z..
N..
```





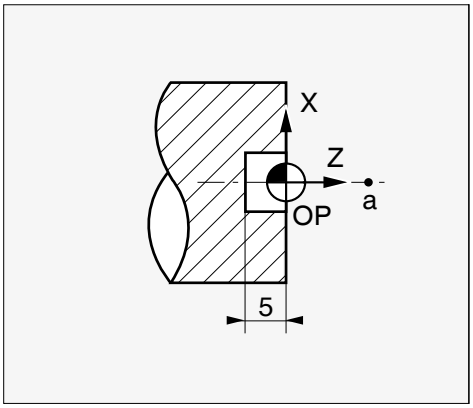
Example

Execution of counterboring

```
N.. ...
N50 G00 Xa Za
N60 G94 (or G95) F..
N70 G82 Z-5 EF2
N80 G80 G00 X.. Z..
N..
```

or

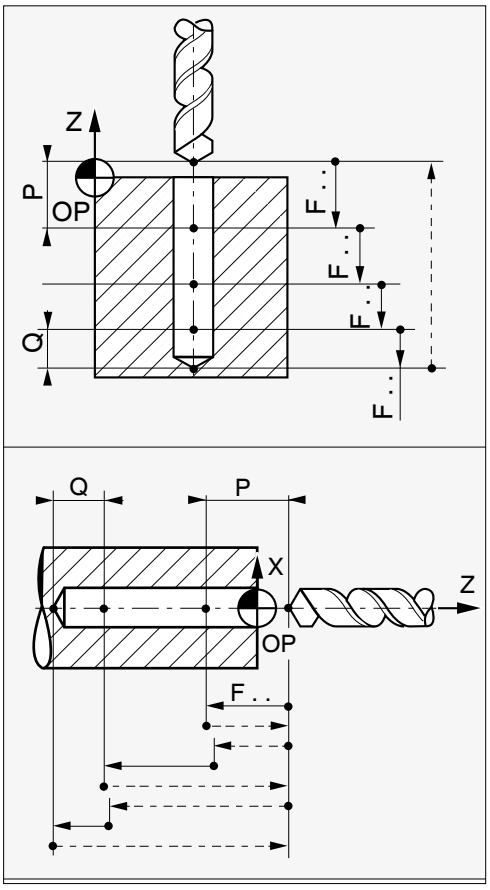
```
N..
N50 G94 (or G95) F..
N60 G82 Xa ERa Z-5 EF2
N70 G80 G00 X.. Z..
N..
```



5.10.5 Peck drilling cycle ► G83



G83	Peck drilling cycle.



Syntax (XY plane)

N.. [G17] **G83** [X.. Y..] Z.. [ER..] [EH..] [P..] / [ES..] [Q..] [EP..] [F..] [EF..]

G17	XY plane selection.
G83	Peck drilling cycle.
X.. Y..	Tool position in the plane.
Z..	End point on the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
P..	Peck drilling cycle.
ES..	Number of in-feeds (pecks) at constant value (see Fig. 1).
Q..	Value of last peck (optional).
EP..	Back-off after each peck (default 1 mm).
F..	Feed rate in the cycle.
EF..	Dwell at the end of each infeed.

Property of the Function

Function G83 is modal.

Cancellation

Function G83 is cancelled by one of functions G31, G80-G82, G84-G89.

Notes

If addresses P and Q are programmed, the consecutive pecks between P and Q get progressively smaller.

At least one of arguments P and ES must be programmed or the system returns error message 889. If the value of P is greater than delta Z, the system returns error message 881.

Notes related to ES (number of constant in-feeds)

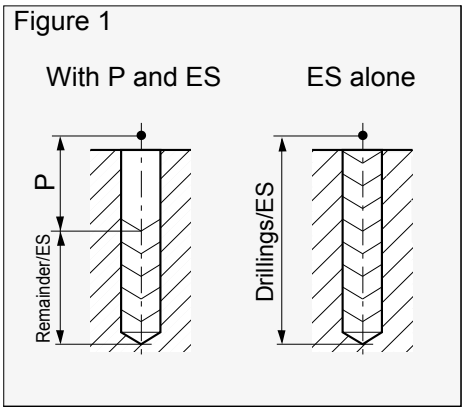
If P and ES are programmed:

The first infeed is equal to P and drilling is continued as a number ES of in-feeds.

If ES is programmed alone (without P):

All the drilling is executed as a number ES of in-feeds.

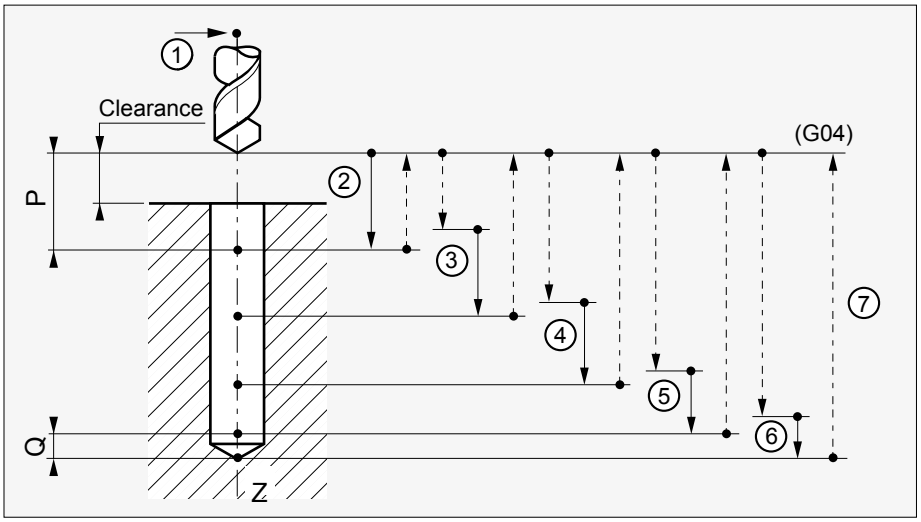
Figure 1



Cycle Steps

The steps below are given as an illustration. The number of steps depends on the values programmed in the cycle.





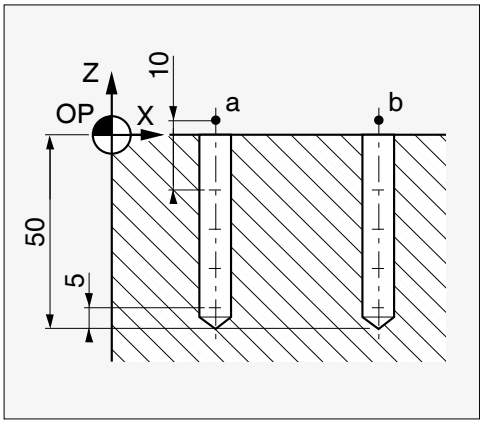
- Step 1: Rapid positioning in the plane.
- Step 2: First peck to depth P.. at feed rate F..
- Rapid retraction to the start point along the tool axis.
- Rapid infeed to 1 mm (or EP..) above depth P..
- Step 3: Second peck at feed rate F..
- Rapid retraction to the start point in the tool axis.
- Rapid infeed to 1 mm (or EP..) above the previous peck.
- Steps 4 and 5: Pecks and retractions same as step 2.
- Step 6: Peck to depth Q.. at feed rate F..
- Step 7: Rapid retraction along the tool axis.
- Possible dwell G04 F.. at the start point.

Example

```
N.. ...
N50 G00 Xa Ya Za
N60 G83 Z-50 P10 Q5 F50
N70 Xb Yb
N80 G80 G00 Z.
N..
```

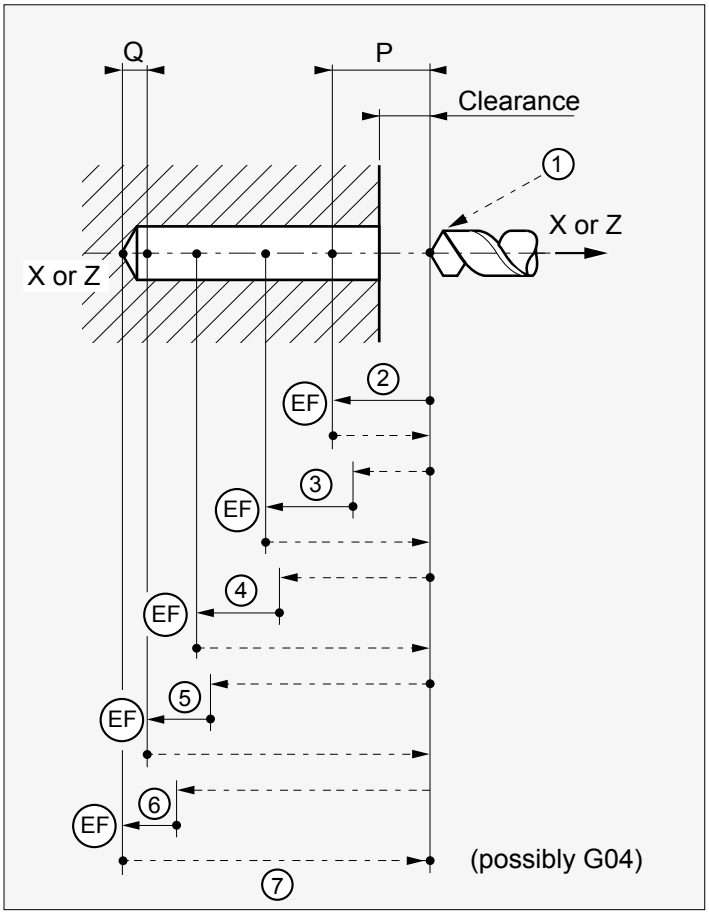
or

```
N.. ...
N60 G83 Xa Ya ERa Z-50 P10 Q5 F50
N70 Xb Yb
N80 G80 G00 Z..
N..
```



Cycle Steps

The steps below are given as an illustration. The number of steps depends on the values programmed in the cycle.



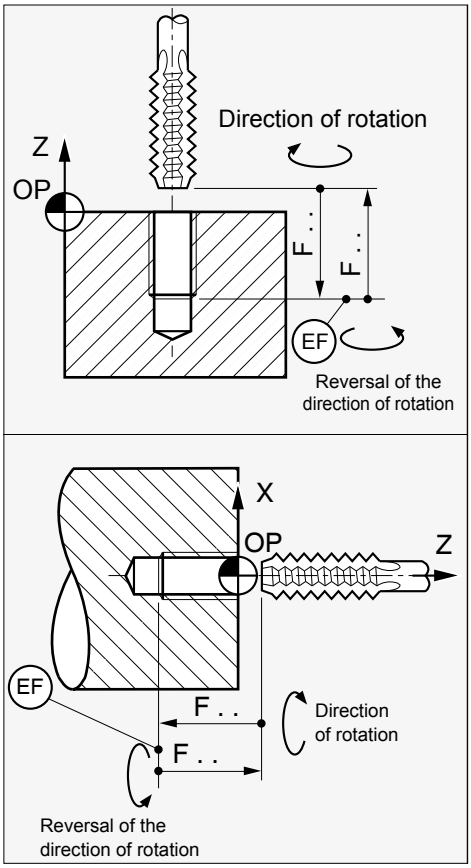
- Step 1: Fast positioning in the machining axis.
- Step 2: First peck to depth P.. along the tool axis at the machining feed rate.
- Possible dwell at the end of the peck.
- Rapid retraction to the start point.
- Rapid infeed to 1 mm (or EP..) above depth P..
- Step 3: Second peck at feed rate.
- Possible dwell at the end of the peck.
- Rapid retraction to the start point.
- Rapid infeed to 1 mm (or EP..) above the previous peck.
- Steps 4 and 5: Pecks and retractions same as step 3.
- Step 6: Peck to depth Q.. at feed rate.
- Step 7: Rapid retraction to the start point.
- Possible dwell G04 F.. at the start point.

5.10.6 Tapping cycles ► G84



G84	Tapping cycle.

This cycle is used to tap a hole with a floating tap-holder.



Syntax (XY plane)

N.. [G17] **G84** [X.. Y..] Z.. [ER..] [EH..]EF.. [F..]

G17	XY plane selection.
G84	Tapping cycle.
X.. Y..	Tool position in the plane.
Z..	End point on the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
EF..	Dwell time in seconds (maximum 99.99 s, format EF022, default 1 second).
F..	Feed rate in the cycle.

Property of the Function

Function G84 is modal.

Cancellation

Function G84 is cancelled by one of functions G31, G80-G83, G84-G89.

Notes

In this tapping cycle, the feed rate is not coupled to the spindle rotation speed. The tap must be allowed to float to compensate for position errors.

During execution of the cycle, feed rate override by potentiometer is inhibited (value forced to 100%).

Calculation of the feed rate in mm/min

F.. = Tap pitch (in mm) x spindle rotation speed (RPM).



Cycle steps

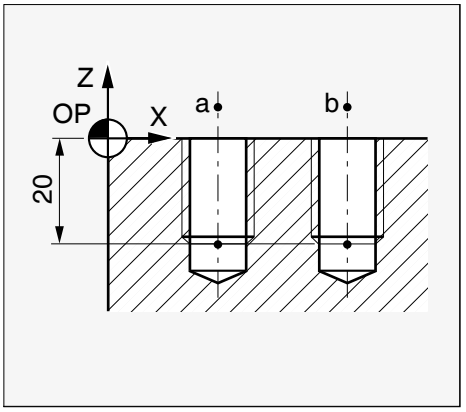
- Step 1: Rapid positioning in the plane.
- Step 2: Infeed at calculated feed rate F..
- Step 3: Reversal of the direction of rotation at depth.
- Step 4: Dwell time at depth.
- Step 5: Retraction at calculated feed rate F.. along the tool axis.

Example

```
N.. ...
N40 S300 M41 M03
N50 G00 Xa Ya Za
N60 G84 Z-20 EF1 F375
N70 Xb Yb
N80 G80 G00 Z..
N..
```

or

```
N.. ...
N60 G84 Xa Ya ERa Z-20 EF1 F375
N70 Xb Yb
N80 G80 G00 Z..
N..
```



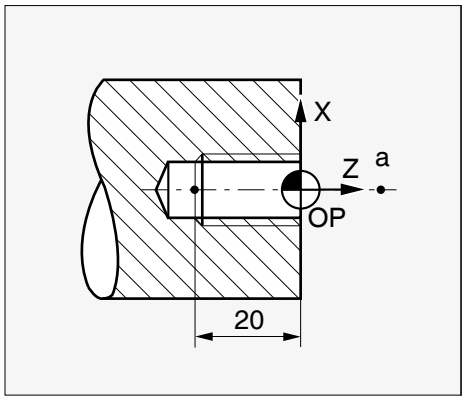


Cycle Steps

- Step 1: Fast positioning in the machining axis.
- Step 2: Infeed at the specified feed rate F..
- Step 3: Reversal of the direction of rotation at the end of tapping.
- Step 4: Dwell at the end of tapping.
- Step 5: Out-feed at the specified feed rate F.. along the tool axis.

Example

Execution of one M8 tapping, pitch 1.25




```
N.. ...
N40 S300 M41 M03
N50 G00 Xa Za
N60 G94 (or G95) F..
N70 G84 Z-20 EF1
N80 G80 G00 X.. Z..
N..

or

N..
N50 G94 F375
N60 G84 Xa ERa Z-20 EF1
N70 G80 G00 X.. Z..
N..
```

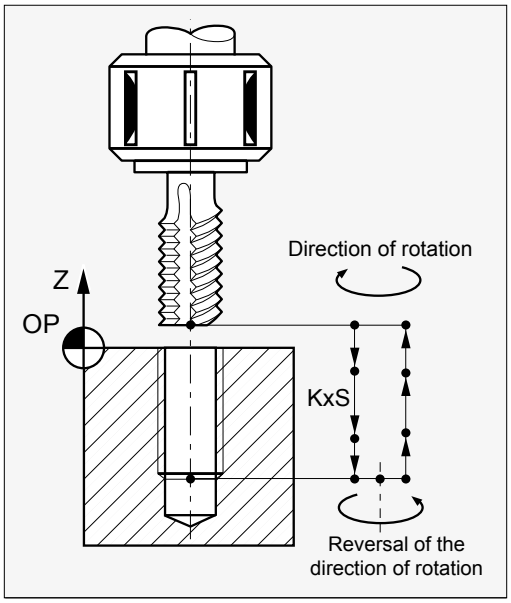
5.10.7 Tapping with chip breaking ► G84



G84 Tapping with chip breaking.

This rigid tapping cycle acts similar to G87 with the difference that the spindle will be managed and inverted at each retraction.

Refer to the rigid tapping cycle for information about spindle management.





Syntax (XY plane)

N.. [G17] **G84** [X.. Y..] Z.. [ER..] [EH..] [P..] / [ES..] [Q..] [EP..] [EF..] [F..]

G17	XY plane selection.
G84	Drilling cycle with chip breaking.
X.. Y..	Tool position in the plane.
Z..	End point in the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
P..	Value of first infeed.
ES..	Number of in-feeds at constant value (see Fig. 1).
Q..	Value of last infeed (optional).
EP..	Back-off between two in-feeds (default no back-off, EP = 0).
EF..	Mandatory dwell time in seconds.
K..	Tap pitch in mm (K specifies rigid tapping).
EK..	Spindle Out-feed/infeed ratio (the default value of EK = 1).

Property of the Function

Function G87 is modal.

Cancellation

Function G87 is cancelled by one of functions G31, G80-G86, G88-G89.

Notes

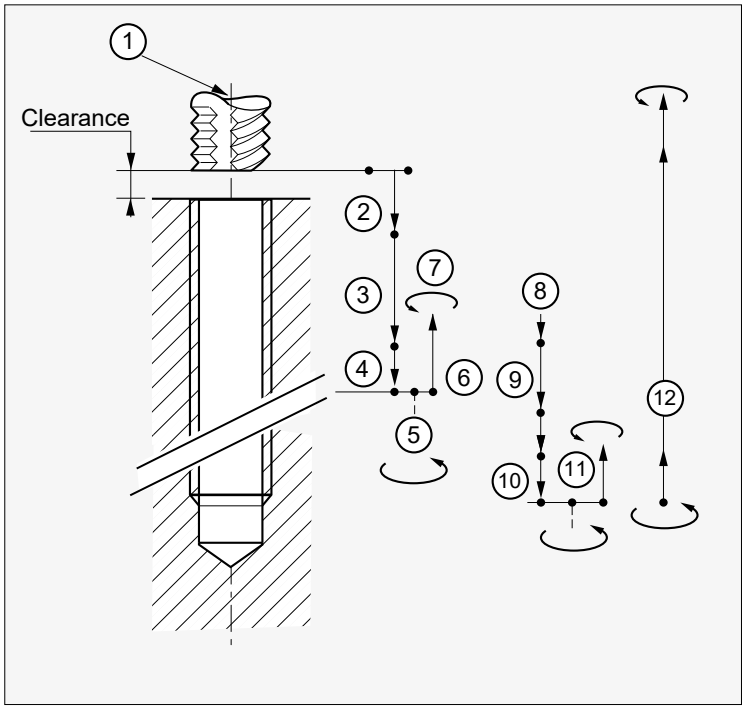
If addresses P and Q are programmed, the consecutive in-feeds between P and Q get progressively smaller. At least one of arguments P and ES must be programmed or the system returns error message 889. If the value of P is greater than delta Z, the system returns error message 881.

Notes related to ES (number of constant in-feeds)

- If P and ES are programmed:
  The first infeed is equal to P and the remaining distance is divided into ES segments.
- If ES is programmed alone (without P):
  The tapping is executed divided in ES segments.

Cycle Steps

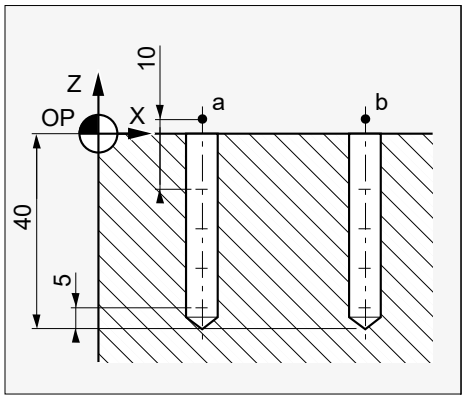




- Step 1: Tool positioning at start point.
- Step 2: Spindle acceleration with synchronized infeed.
- Step 3: Infeed at constant speed.
- Step 4: Spindle deceleration with synchronized infeed; optional dwell; total infeed value = P.
- Step 5: Spindle inversion with synchronized retraction.
- Step 6: Retraction value = EP.
- Step 7: Spindle returns to initial direction.
- Step 8 to 11: Repeat steps 2 to 7 as much as necessary - The last infeed value will be Q (if programmed).
- Step 12: (For the last repetition the spindle remains in unscrew direction (step 7 skipped) - retraction up to EH and then to ER in rapid.

Example

```
N.. ...
N50 G00 Xa Ya Za
N60 G87 Z-40 P10 Q5 EF1 F40
N70 Xb Yb
N80 G80 G00 Z..
N..
```



or

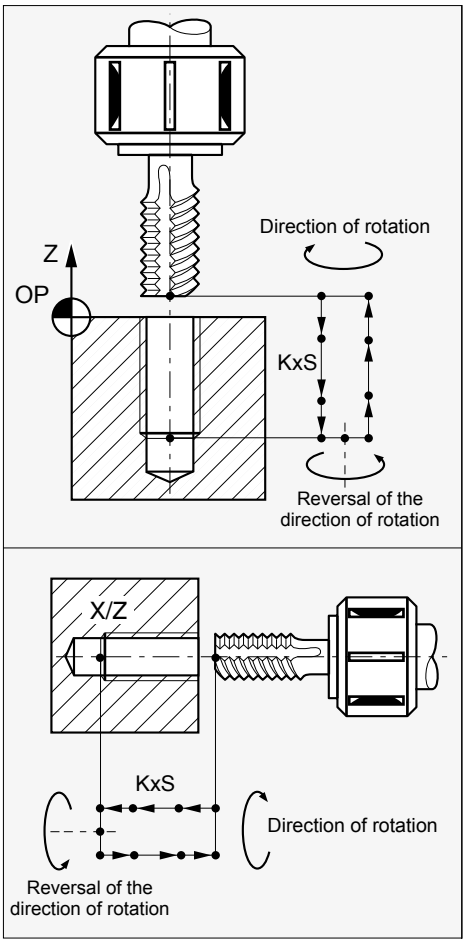
```
N.. ...
N60 G87 Xa Ya ERa Z-40 P10 Q5 EF1 F40
N70 Xb Yb
N80 G80 G00 Z..
N..
```

5.10.8 Rigid tapping cycle ► G84

G84	Rigid tapping cycle.

With this cycle, the spindle speed controls the tap feed. The feed rate is automatically calculated from the actual spindle speed and (K) programmed pitch.

Rigid tapping cycle can only be executed with the main spindle of each active channel.



Syntax (XY plane)

N.. [G17] [M03/M04] [S..] [M40 to M45] **G84** [X.. Y..] Z.. [ER..] [EH..] K.. [EK..]

G17	XY plane selection.
M03/M04	Spindle direction of rotation.
S..	Spindle rotation speed.
M40-M45	Spindle ranges.
G84	Tapping cycle.
X.. Y..	Tool position in the plane.
Z..	End point on the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
K..	Tap pitch in mm (K specifies rigid tapping).
EK..	Spindle Out-feed/in-feed ratio (the default value of EK = 1).

Property of the Function

Function G84 is modal.

Cancellation

Function G84 is cancelled by one of functions G31, G80-G83, G85-G89.

Notes

When the cycle is called, the tool axis is coupled with spindle rotation.
During execution of the cycle:

- the following error of the tool axis is cancelled during feed at constant speed,
- feed rate and spindle speed override by potentiometer is inhibited (value forced to 100%)

In the end-of-tapping region, the spindle is decelerated then rotation is reversed.
At the end of the cycle, the spindle is returned to its initial direction except if M05 is programmed with the cycle, in which case the spindle is stopped at the end when dimension EH is reached.

Rigid tapping can be carried out over several penetrations, but this requires programming several consecutive blocks. With rigid tapping, the system returns error message 899 in the following cases:

- The channel does not control the spindle used or does not provide its measurement.

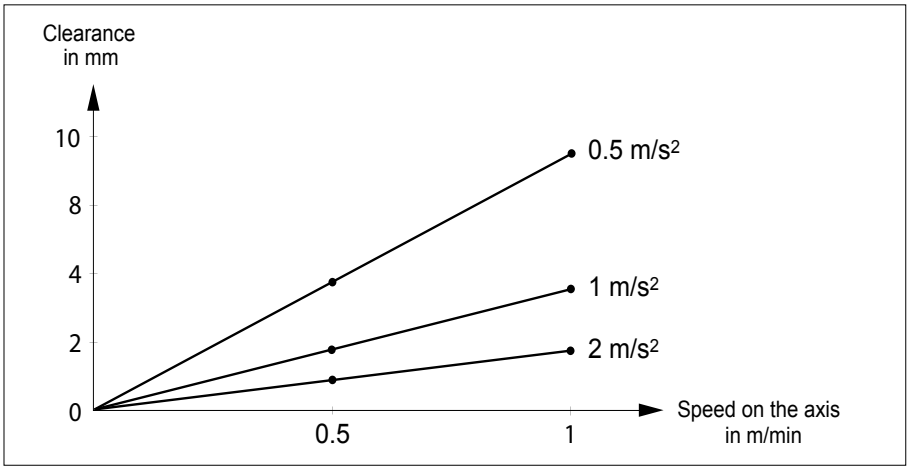
Tapping Clearance

Before starting the cycle, sufficient clearance must be provided to allow the tapping axis to reach correct speed before impacting the material. This clearance depends on the required tapping speed and the acceleration allowed on the axis.

The following chart can be used to approximate the clearance required.

Use of the chart for tapping an M10 hole (pitch = 1.5 mm), for instance:
- Rotation speed = 320 rpm
- Axis feed rate = 480 mm/min or 0.48 m/min
- Acceleration = 0.5 m/s^2.

Required clearance read on the chart: approximately 4 mm

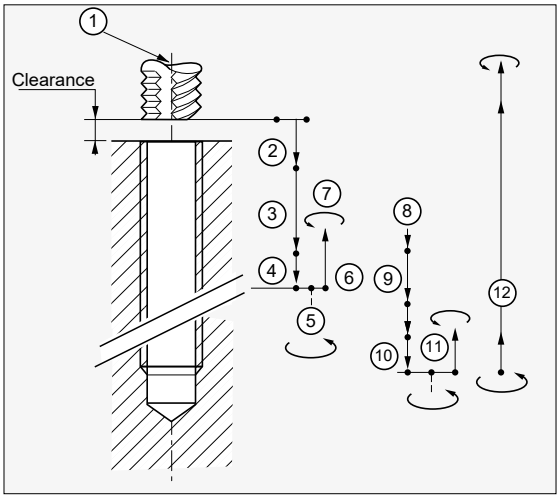


It is possible to stop the spindle during rigid tapping. In such case the tapping axis will also stop in synchronization. The cycle will resume if the spindle is restarted.

A Reset is also possible when the spindle is stopped. In such case the cycle will be cancelled. The tap could then be removed via G39+ function (Rigid tap removal).



Cycle Steps

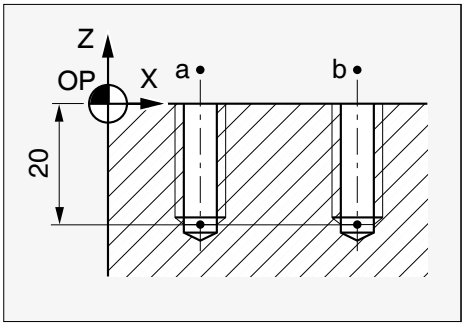


- Step 1: Tool positioning in the hole axis (provide clearance).
- Step 2: Infeed with acceleration of the spindle and feed rate.
- Step 3: Feed at constant speed.
- Step 4: Deceleration before reaching depth.
- Step 5: Reversal of the direction of rotation.
- Step 6: Retraction with acceleration over a distance equal to the deceleration.
- Step 7: Feed at constant speed.
- Step 8: Return of the spindle to the initial direction.

Examples

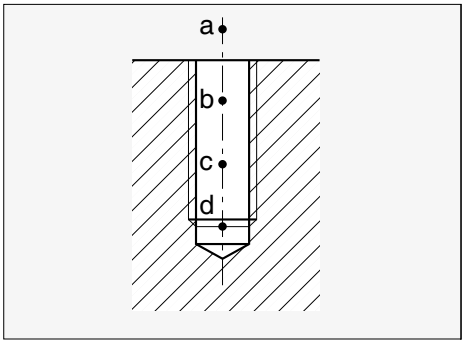
Execution of two rigid tapping operations, pitch 1.50 mm

```
N.. ...
N50 S200 M41 M03
N60 G84 Xa Ya ERa Z-20 K1.5 EK2
N70 Xb Yb
N80 G80 G00 Z..
```

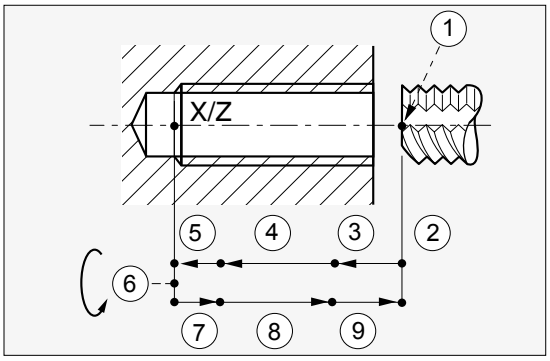


Execution of one rigid tapping operation with several consecutive penetration depths

```
N.. ...
N40 S400 M41 M03
N50 G00 Xa Ya Za
N60 G84 Zb K.. EK..
N70 Zc
N80 Zd
N90 G80 G00 Z..
N..
```



Cycle Steps

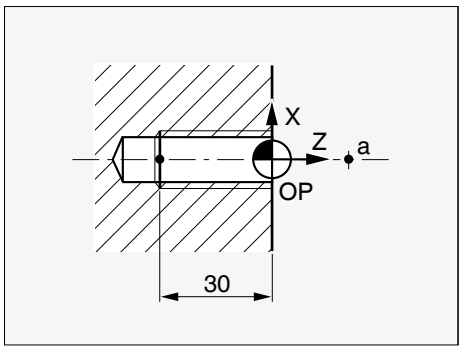


- Step 1: Rapid to start of hole (provide clearance).
- Step 2: Infeed with acceleration of the spindle speed and feed rate.
- Step 3: Feed at constant speed.
- Step 4: Deceleration before reaching depth.
- Step 5: Reversal of the direction of rotation.
- Step 6: Retraction with acceleration over a distance equal to the deceleration.
- Step 7: Feed at constant speed.
- Step 8: Return of the spindle to the initial direction.

Examples

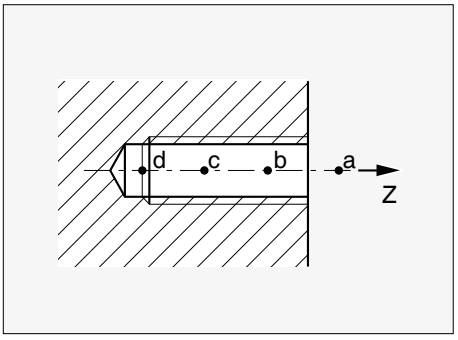
Execution of one M10 hole, pitch 1.50

```
N.. ...
N130 G97 S200 M41 M03
N140 G00 Xa Za
N150 G84 Z-30 K1.5 EK2
N160 G80 G00 G52 X.. Z..
```



Execution of one tapped hole with several consecutive penetrations

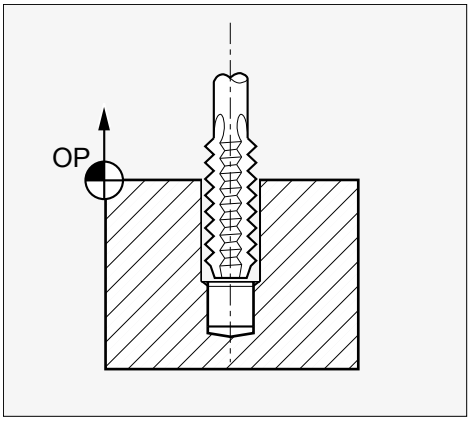
```
N.. ...
N240 G97 S400 M41 M03
N250 G00 Xa Za
N260 G84 Zb K.. EK..
N270 Zc
N280 Zd
N290 G80 G00 G52 X.. Z..
N..
```



5.10.9 Rigid tap removal ► G39+



G39+	Rigid Tap Removal.



This function is used if the rigid tapping has been interrupted while it was in progress, then the tool remains caught in the material.

G39+ allows to easily removing the tool from the material, even if homing is not completed and the inclined plane is enabled.

Syntax

[G17] **G39+** M3/M4 [M40 to M45] S.. K.. Z..

G17	XY plane selection.
G39+	Rigid tap removal cycle which calls 4 mandatory parameters and one optional parameter programmed immediately after the function.
M3 or M4	Direction of spindle rotation.
M40 to M45	Speed range (optional).
S..	Spindle speed.
K..	Pitch in mm.
Z..	Axis and signed retraction distance (mm and incremental). Axes U to Z can be programmed if they exist.

Notes

The arguments of the cycle can be replaced by E parameters E or L variables.

Mirroring and scaling factor are ignored during G39+, then are re-enabled if the program is to be continued.

It should be noted that:

- Cycle G39+ forces use of the spindle in G97
- Since the tool must not be programmed and movement is incremental, the consistency of the tool direction is not checked.

Cycle Steps

- Step 1: Spindle started in the programmed direction.
- Step 2: Tool axis starts to move synchronized with the spindle.
- Step 3: Spindle accelerates up to programmed speed.
- Step 4: When the programmed distance is complete the spindle is stopped and an M02 is automatically generated..

Errors description

Error 1:

- The programmed axis does not exist.
- Several axes are programmed or the same axis is programmed several times.
- M3/M4 or M40-M45 are programmed twice.
- The spindle is not in state M5.

Error 2:

- One of the four mandatory arguments (M3/M4, S, K or Z) is missing
- Another argument was programmed
- An M function other than M3/M4 or M40-M45 -"M02" is programmed
- S is too large (>65536)
- K is negative or zero.

Error 4:

- Rigid tapping option missing.

Error 39:

- S is not compatible with the speed range programmed.

5.10.9.1 Cycle execution

This cycle can be executed even if homing was not completed on the axis and/or spindle.
The inclined plane can be active. The spindle must be controlled by the group in which the cycle is programmed. At the beginning of the cycle, the spindle is in state M5.

The spindle speed range active at the time of the interrupt should be recovered. If necessary, this speed range can be programmed in G39+.

Functions M3/M4 and M40 to M45 are sent to the PLC, but if subroutines are normally called by these functions, they are not executed during a G39+ cycle. At the beginning of the cycle, the spindle is started in the specified direction at the minimum spindle speed.
Then the spindle is accelerated continuously up to the programmed speed S.

From the beginning of the cycle, movement on the Z axis is linked to the spindle and the following error is cancelled to comply with the specified pitch. The rigid tapping parameters set in P63 are used.

At the end of the cycle:

When travel is completed, the spindle is stopped.

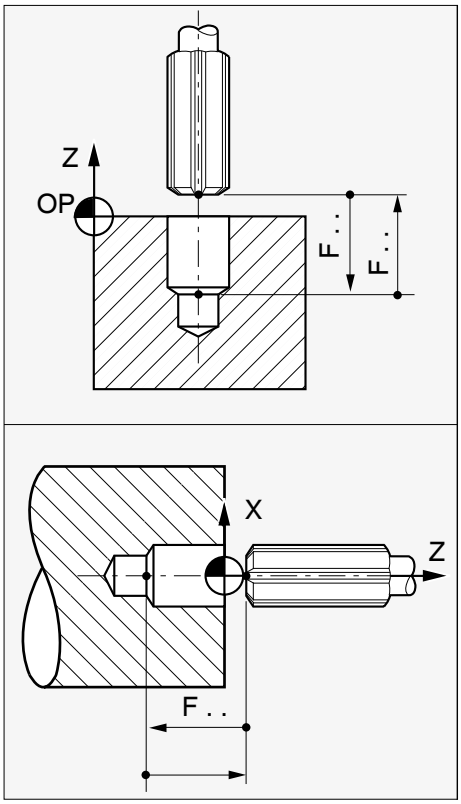
When travel is completed, the HOMING COMPLETED state is cancelled on the spindle and a new SPINDLE HOMING request is generated.

Movement of the tool axis is enabled during G39+, even if the axis is in HOMING NOT COMPLETED state. However, if the program is to continue, axis homing is required.

5.10.10 Reaming cycle ► G85



G85	Reaming cycle.



Syntax (XY plane)



N.. [G17] **G85** [X.. Y..] Z.. [ER..] [EH..] [F..] [EF..]

G17	XY plane selection.
G85	Reaming cycle.
X.. Y..	Tool position in the plane.
Z..	End point on the machining axis.
EH..	Impact plane in the machining axis.
ER..	Retraction plane in the machining axis.
F..	Feed rate in the cycle.
EF..	Retraction rate (default = feed rate F..).

Property of the Function

Function G85 is modal.

Cancellation

Function G85 is cancelled by one of functions G31, G80-G84, G86-G89.

Cycle Steps

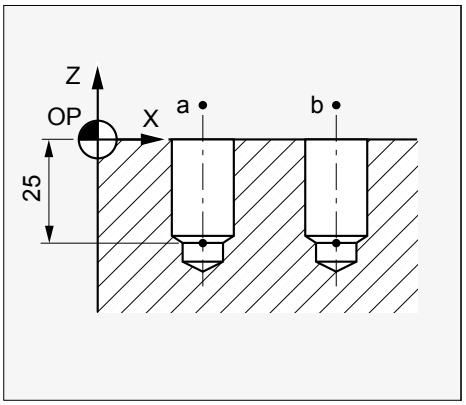
- Step 1: Rapid positioning in the plane.
- Step 2: Infeed at feed rate F..
- Step 3: Retraction at feed rate F.. along the tool axis.

Example

```
N.. ...
N50 G00 Xa Ya Za
N60 G85 Z-25 F80
N70 Xb Yb
N80 G80 G00 Z..
N..
```

or

```
N.. ...
N60 G85 Xa Ya ERa Z-25 F80
N70 Xb Yb
N80 G80 G00 Z..
```



Syntax

N.. **G85** X.. / Z.. [ER..] [EH..] [EF..]

G85	Reaming cycle.
X.. / Z..	End point on the machining axis.
ER..	Dimension of the retraction plane on the machining axis.
EH..	Impact plane in the machining axis.
EF..	Retraction rate (default = feed rate F..).

Property of the Function

Function G85 is modal.

Cancellation

Function G85 is cancelled by one of functions G80-G84, G87 and G89 or functions G64, G65, G66.

Cycle Steps

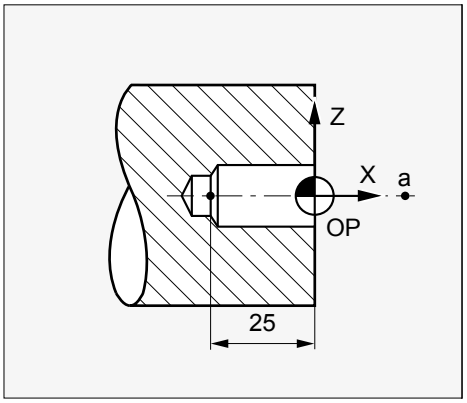
- Step 1: Fast positioning in the machining axis.
- Step 2: Infeed at feed rate F..
- Step 3: Out-feed at feed rate F.. along the tool axis.

Example

```
N.. ...
N50 G00 Xa Za
N60 G94 (or G95) F..
N70 G85 Z-25
N80 G80 G00 X.. Z..
N..

or

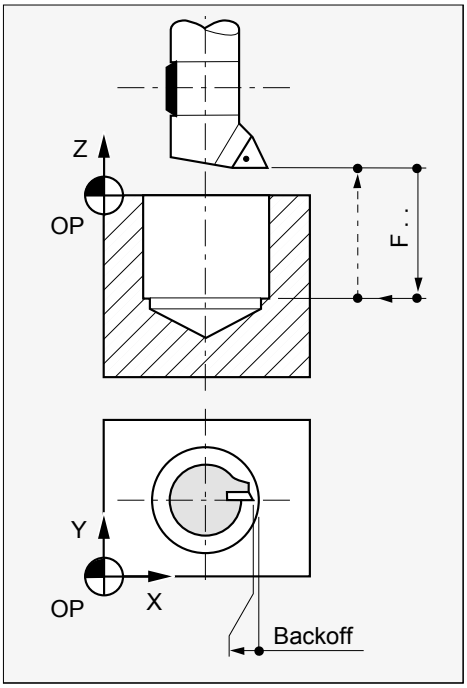
N..
N50 G94 (or G95) F..
N60 G85 Xa ERa Z-25
N70 G80 G00 X.. Z..
```



5.10.11 Boring cycle with indexed spindle Stop at the bottom of the hole ► G86



G86	Boring cycle with indexed spindle stop at the bottom of the hole.



Syntax (XY plane)

N.. [G17] **G86** [X.. Y..] Z.. [ER..] [EH..] [EC..] [EA..] [EP..] [F..]

G17	XY plane selection.
G86	Boring cycle with indexed spindle stop at the bottom of the hole.
X.. Y..	Tool position in the plane.
Z..	Machining end point.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
EC..	Value of the indexing position (the default value of EC is the last indexing position programmed).
EA..	Angle between EC.. programmed and the physical angular position of the cutting edge.
EP..	Lateral back-off in the hole bottom (default 2 mm).
F..	Feed rate in the cycle.

Property of the Function

Function G86 is modal.

Cancellation

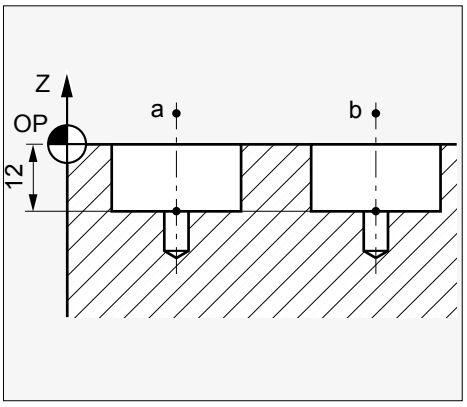
Function G86 is cancelled by one of functions G31, G80-G85, G87-G89.

Cycle Steps

- Step 1: Rapid positioning in the plane.
- Step 2: Infeed at feed rate F..
- Step 3: Indexed spindle stop at depth.
- Step 4: Lateral back-off 2 mm (or EP..) on the indexing axis.
- Step 5: Rapid retraction along the tool axis.

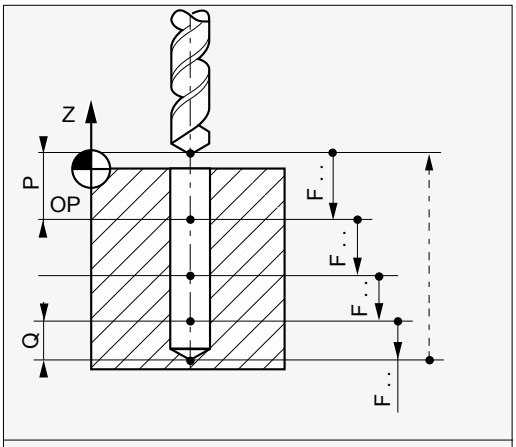
Example

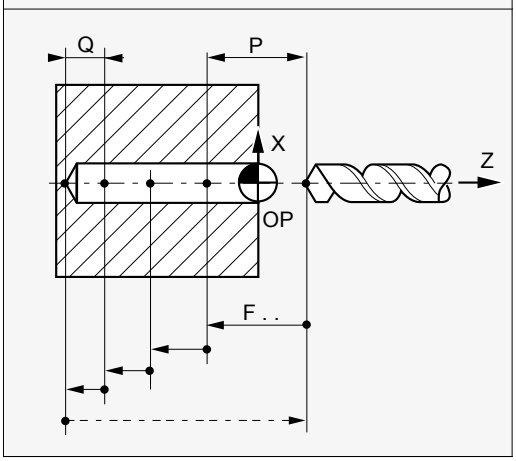
```
N.. ..
N50 G00 X.. Y.. Za
N60 G86 Xa Ya Z-10 EC.. F30
N70 Xb Yb
N80 G80 G00 Z..
N..
```



5.10.12 Drilling cycle with chip breaking ► G87

G87	Drilling cycle with chip breaking.





Syntax (XY plane)

N.. [G17] **G87** [X.. Y..] Z.. [ER..] [EH..] [P..] / [ES..] [Q..] [EP..] [EF..] [F..]

G17	XY plane selection.
G87	Drilling cycle with chip breaking.
X.. Y..	Tool position in the plane.
Z..	End point in the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
P..	Value of first infeed.
ES..	Number of in-feeds at constant value (see Fig. 1).
Q..	Value of last infeed (optional).
EP..	Back-off between two in-feeds (default no back-off, EP = 0).
EF..	Mandatory dwell time in seconds.
F..	Feed rate in the cycle.

Property of the Function

Function G87 is modal.

Cancellation

Function G87 is cancelled by one of functions G31, G80-G86, G88-G89.

Notes

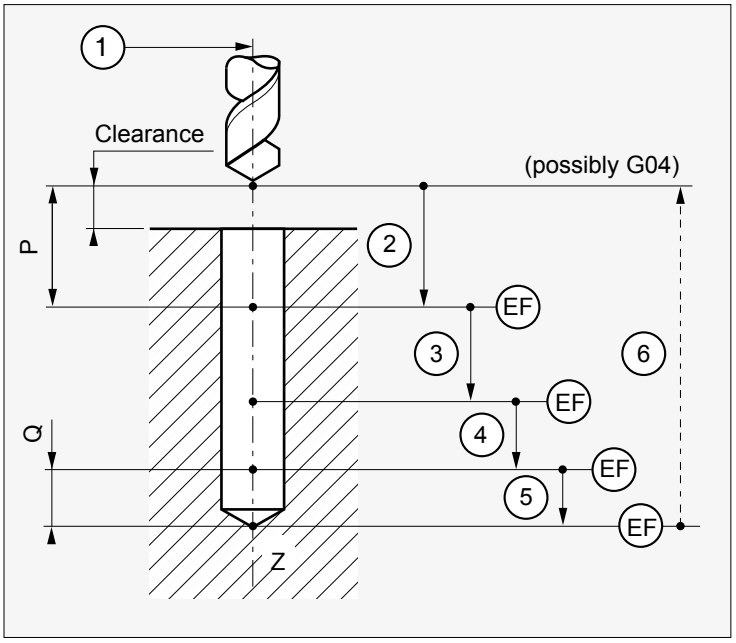
If addresses P and Q are programmed, the consecutive in-feeds between P and Q get progressively smaller. At least one of arguments P and ES must be programmed or the system returns error message 889. If the value of P is greater than delta Z, the system returns error message 881.

Notes related to ES (number of constant in-feeds)

- If P and ES are programmed:
  The first infeed is equal to P and drilling is continued as a number ES of in-feeds.
- If ES is programmed alone (without P):
  All the drilling is executed as a number ES of
  in-feeds.

Cycle Steps

The steps below are given for reference. The number of steps depends on the values programmed with the cycle.



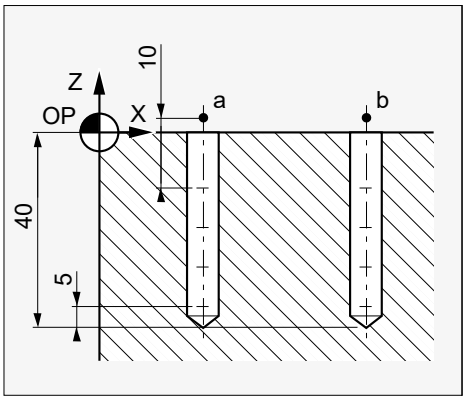
- Step 1: Rapid positioning in the plane.
- Step 2: Infeed to depth P.. at feed rate F..
- Dwell at the end of each infeed (possible back-off by value EP..).
- Steps 3 and 4: Successive in-feeds and dwells (with possible back-offs by value EP..) same as step 2.
- Step 5: Infeed to depth Q.. at feed rate F..
- Dwell at depth.
- Step 6: Rapid retraction along the tool axis.
- Possible dwell G04 F.. at the start point.

Example

```
N.. ...
N50 G00 Xa Ya Za
N60 G87 Z-40 P10 Q5 EF1 F40
N70 Xb Yb
N80 G80 G00 Z..
N..
```

or

```
N.. ...
N60 G87 Xa Ya ERa Z-40 P10 Q5 EF1 F40
N70 Xb Yb
N80 G80 G00 Z..
N..
```



Syntax



N.. **G87** X.. / Z.. [ER..] [EH..] [P..] / [ES..] [Q..] [EP..] [EF..]

G87	Drilling cycle with chip breaking.
X.. / Z..	End point on the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
P..	Value of first infeed.
ES..	Number of in-feeds at constant value (see Fig. 1).
Q..	Value of last infeed (optional).
EP..	Back-off between two in-feeds (default no back-off, EP = 0).
EF..	Mandatory dwell time in seconds.

Property of the Function

Function G87 is modal.

Cancellation

Function G87 is cancelled by one of functions G80-G85 and G89 or functions G64, G65, G66.

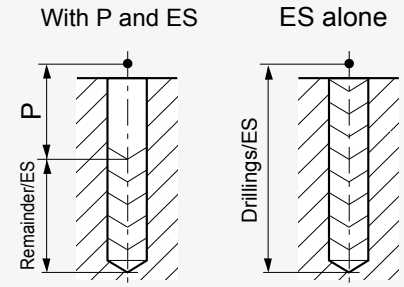
Notes

If addresses P and Q are programmed, the consecutive in-feeds between P and Q get progressively smaller. At least one of arguments P and ES must be programmed or the system returns error message 889. If the value of P is greater than delta Z, the system returns error message 881.

Notes related to ES (number of constant in-feeds)

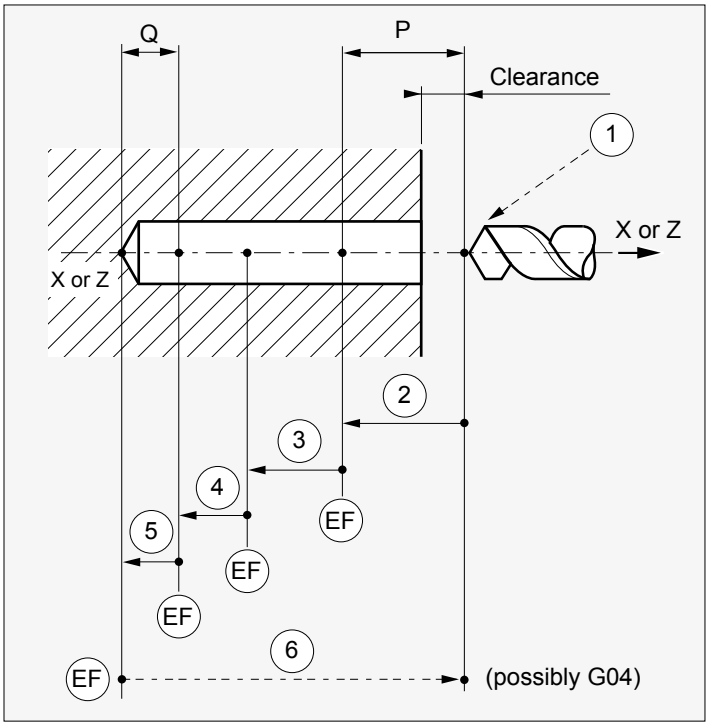
- If P and ES are programmed:
  The first infeed is equal to P and drilling is continued as a number ES of in-feeds.
- If ES is programmed alone (without P):
  All the drilling is executed as a number ES of in-feeds.

Fig.1



Cycle Steps

The steps below are given as an illustration. The number of steps depends on the values programmed with the cycle.

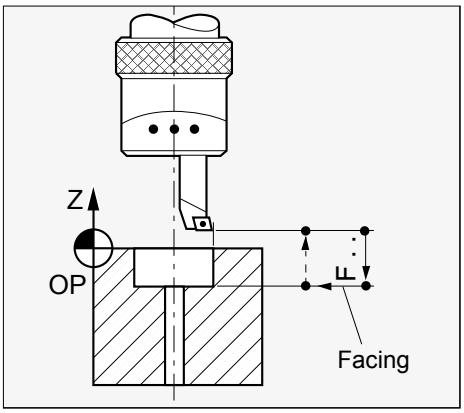


- Step 1: Fast positioning in the machining axis.
- Step 2: First infeed of depth P.. at programmed feed rate.
- Possible dwell at the end of the infeed.
- Steps 3 and 4: Consecutive in-feeds and dwells as step 2.
- Step 5: Infeed depth Q.. at the feed rate.
- Possible dwell at the end of the peck.
- Step 6: Rapid retraction along the tool axis.
- Possible dwell G04 F.. at the start point.

5.10.13 Boring and Facing cycle ► G88



G88	Boring and facing cycle.



Syntax (XY plane)

N.. [G17] **G88** [X.. Y..] Z.. [ER..] [EH..] [F..]

G17	XY plane selection.
G88	Boring and facing cycle.
X.. Y..	Tool position in the plane.
Z..	End point in the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
F..	Feed rate in the cycle.

Property of the Function

Function G88 is modal.

Cancellation

Function G88 is cancelled by one of functions G31, G80-G87, G89.

Cycle Steps

- Step 1: Rapid positioning in the plane.
- Step 2: Infeed at feed rate F..
- Step 3: Feed stop at depth, spindle still rotating.
- Display of the message: "FACING COMPLETED? (Y):"
- Step 4: Operator action for facing.
- The operator answers the message by pressing Y (yes) to enable continuation to the next step (5).
- Step 5: Retraction at rapid along the tool axis.



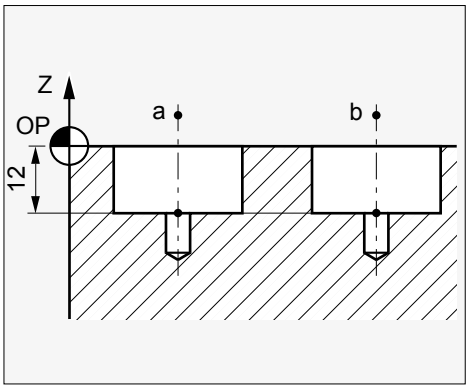
The message "FACING COMPLETED? (Y):" included in the cycle deletes the currently displayed message.

Example

```
N.. ...
N50 G00 Xa Ya Za
N60 G88 Z-12 F30
N70 Xb Yb
N80 G80 G00 Z..
N.
```

or

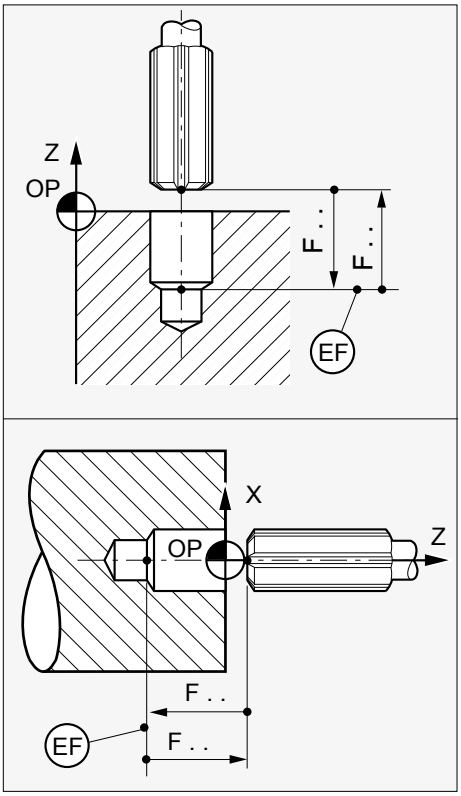
```
N.. ...
N60 G88 Xa Ya ERa Z-12 F30
N70 Xb Yb
N80 G80 G00 Z..
N..
```



5.10.14 Boring cycle with Dwell at the bottom of the hole ► G89



G89	Boring cycle with dwell at the bottom of the hole.





Syntax (XY plane)

N.. [G17] **G89** [X.. Y..] Z.. [ER..] [EH..] [EF..] [F..]

G17	XY plane selection.
G89	Boring cycle with dwell in the bottom of the hole.
X.. Y..	Tool position in the plane.
Z..	End point in the machining axis.
ER..	Retraction plane in the machining axis.
EH..	Impact plane in the machining axis.
EF..	Dwell expressed in seconds (maximum 99.99 s, format EF022, default EF = 1 second).
F..	Feed rate in the cycle.

Property of the Function

Function G89 is modal.

Cancellation

Function G89 is cancelled by one of functions G31, G80-G88.

Cycle Steps

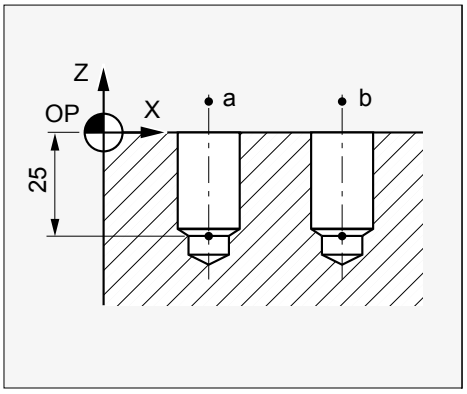
- Step 1: Rapid positioning in the plane.
- Step 2: Infeed at feed rate F..
- Step 3: Dwell at the end of boring.
- Step 4: Rapid retraction along the tool axis.

Example

```
N.. ...
N50 G00 Xa Ya Za
N60 G89 Z-25 EF1 F80
N70 Xb Yb
N80 G80 G00 Z..
N..
```

or

```
N.. ...
N60 G89 Xa Ya ERa Z-25 EF1 F80
N70 Xb Yb
N80 G80 G00 Z..
N..
```



Syntax

N.. **G89** X.. / Z.. [ER..] [EH..] EF..

G89	Boring cycle with dwell in the bottom of the hole.
X.. / Z..	End point on the machining axis.
ER..	Dimension of the retraction plane on the machining axis.
EH..	Impact plane in the machining axis.
EF..	Dwell expressed in seconds (maximum 99.99 s, format EF022, default EF = 1 second).

Property of the Function

Function G89 is modal.

Cancellation

Function G89 is cancelled by one of functions G80-G85, G87 or functions G64, G65, G66.

Cycle Steps

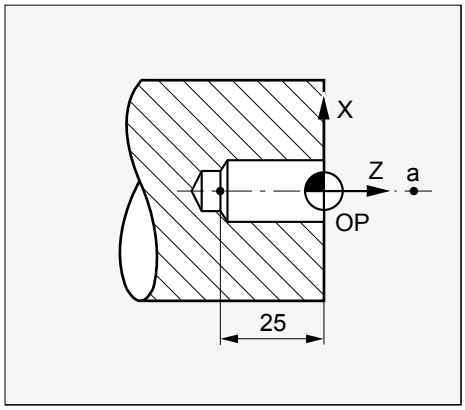
- Step 1: Fast positioning in the machining axis.
- Step 2: Infeed at the feed rate F..
- Step 3: Dwell at the end of boring
- Step 4: Out-feed at the feed rate F.. along the tool axis.

Example

```
N.. ...
N50 G00 Xa Za
N60 G94 (or G95) F..
N70 G89 Z-25 EF1
N80 G80 G00 X.. Z..
N..

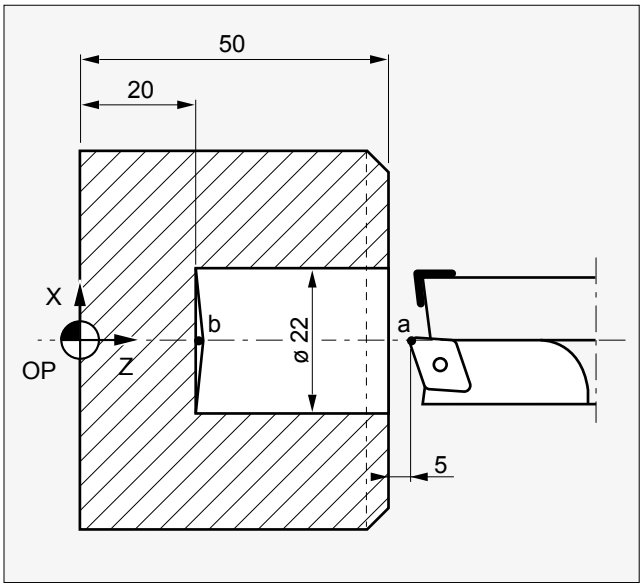
or

N..
N50 G94 (or G95) F..
N60 G89 Xa ERa Z-25 EF1
N70 G80 G00 X.. Z..
N..
```



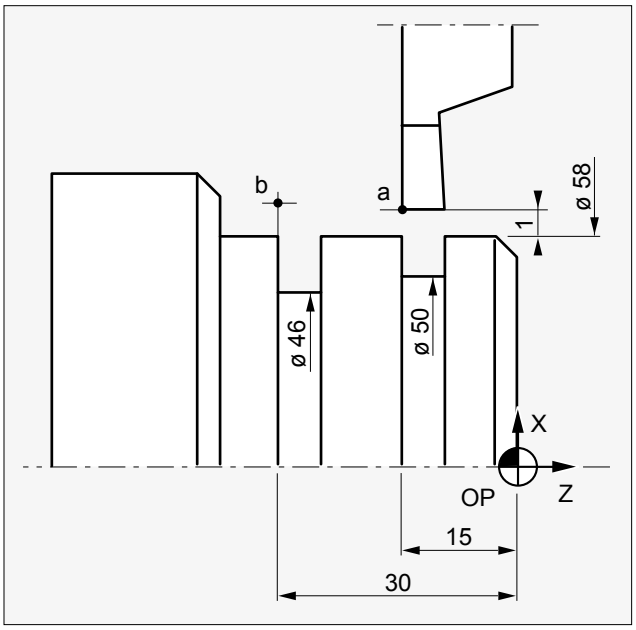
Examples of Programming Cycles

Drilling in the Z axis (cycle G83 or G87)




```
%62
N10...
N.. G92 S3500
N.. ...
N120 T06 D06 M06 (CARBIDE DRILL D=22)
N130 G97 S900 M40 M03
N140 G00 X0 Z55                          Point a
N150 G96 X22 S100
N160 G95 F0.15
N170 G83 (or G87) Z20 P10 Q5             Point b, cycle
N180 G97 S900
N190 G80 G00 G52 X.. Z..                 Cycle cancellation
N..
```

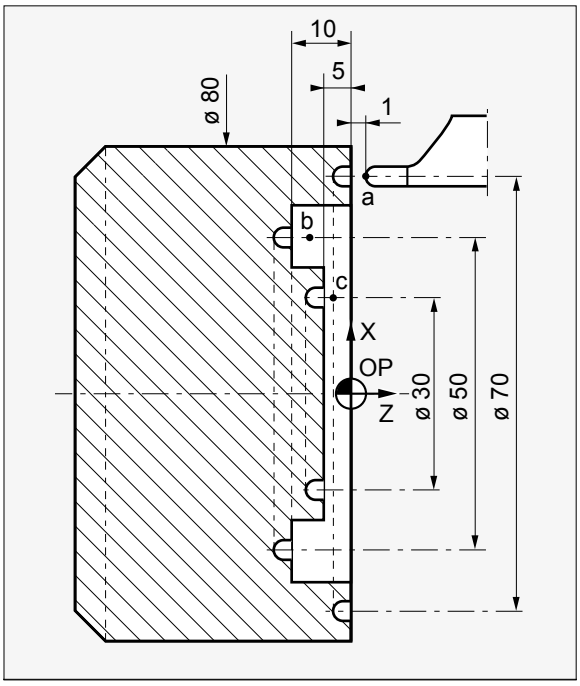
Execution of two grooves in the X axis (cycle G83 or G87)




```
%56
N10...
N.. G92 S3000
N..
N.. ...
N220 T08 D08 M06 (GROOVING TOOL L=4)
N230 G97 S900 M40 M04
N240 G16 P+                          Tool axis orientation on X+
N250 G00 X60 Z-15                    Point a
N260 G96 S50
N270 G95 F0.1
N280 G83 (or G87) X50 P2 Q1 G04 F1   Cycle
N290 X46 Z-30                        Point b, cycle repeat
N300 G97 S900
N310 G80 G16 R+ G00 G52 X.. Z..      Cycle cancellation
N..
```

Execution of three front grooves along the Z axis (cycle G82) with positioning on different levels using ER

Grooves with radius 3, depth 3.

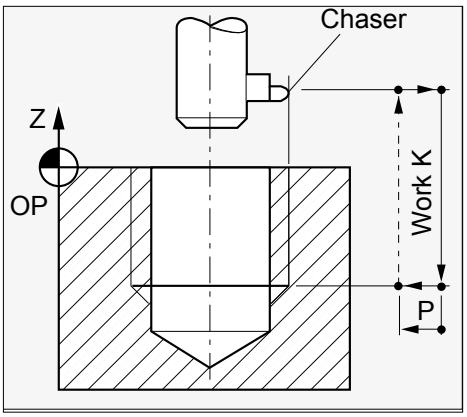



```
%72
N10...
N.. G92 S200
N..
N..
N190 T06 D06 M06 (GROOVE CUTTER R3)
N200 G97 S900 M04 M40
N210 G00 Z10
N220 G96 X80 S40
N230 G95 F0.08
N240 G82 X70 Z-3 ER1 EF2                Point a, cycle
N250 X50 Z-13 ER-9                      Point b, cycle
N260 ER-4
N270 X30 Z-8 EF1                        Point c, cycle
N280 G97 S900
N290 G80 G00 G52 X.. Z..                Cycle cancellation
N..
```

5.10.15 Thread chasing cycle ► G31



G31 Thread chasing cycle.



This cycle couples the tool feed rate to the spindle rotation.

Syntax (XY plane)

N.. [G17] [M03/M04] [S..] **G31** [X.. Y..] Z.. [ER..] [EH..] K.. P.. [F..] [EF..] [EC..]

G17	XY plane selection.
M03/M04	Spindle rotation.
S..	Spindle rotation speed.
G31	Thread chasing cycle.
X.. Y..	Tool position in the plane.
Z..	End point in the machining axis.
ER..	Approach or retraction dimension on the machining axis.
EH..	Impact plane in the machining axis.
K..	Thread pitch in mm. K = pitch with XY plane (G17) J = pitch with ZX plane (G18) I = pitch with YZ plane (G19).
P..	Absolute tool retraction dimension at the end of thread cutting.
F..	Number of thread starts (1 to 9, default 1).
EF..	Dwell time in seconds (maximum 99.99 s, format EF022, default value equal to two spindle rotations).
EC..	Value of the indexing position (the default value of EC is the last indexing position programmed).

Property of the Function

Function G31 is modal.

Cancellation

Function G31 is cancelled by one of functions G80-G89.

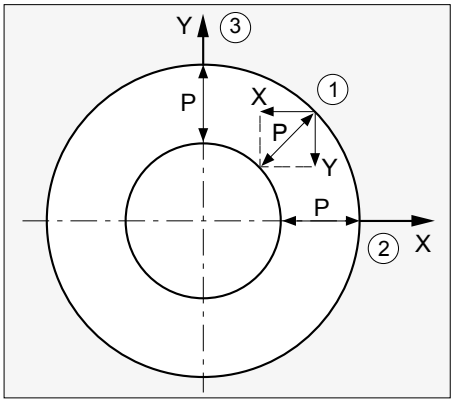
Notes

G31 needs encoder feedback from the spindle. The definition of this device (number of lines per revolution) is established by the machine manufacturer.

The crossing of an angular position (monitored by the system) starts movements for thread cutting. With block continuation, it is possible to change the plane and machining axis.

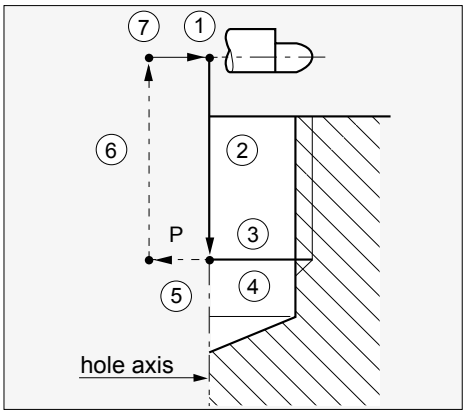
Clearance at the end of thread cutting

- Position 1 in the figure:
  Clearance along the two axes at 45 degrees, in this case X = Y = P/√ 2
- Position 2 on the figure:
  Clearance along the X axis. In this case, X = P.
- Position 3 in the figure:
  Clearance along the Y axis. In this case, Y = P.



Cycle Steps

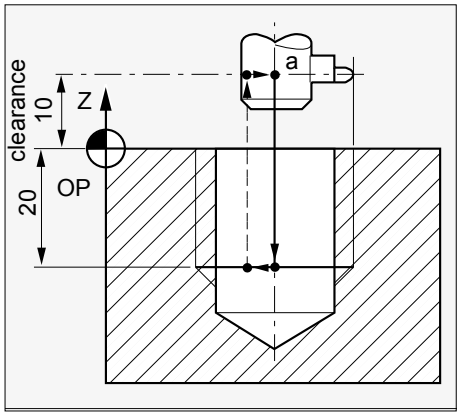
- Step 1: Rapid positioning of the chaser-holder to the center of the hole then dwell for 1.2 seconds to start spindle rotation.
- Step 2: Infeed at the pitch programmed.
- Step 3: Dwell or two spindle rotations at the end of thread cutting.
- Step 4: Indexed spindle stop at position zero of the position encoder.
- Step 5: Rapid clearance by the value P in the axis of the plane and direction defined by the machine manufacturer.
- Step 6: Rapid retraction along the tool axis.
- Step 7: Rapid repositioning of the tool to the hole center and restart of spindle rotation.



For thread cutting with several starts, the cycle is repeated after changing the angular position for each start.

Example

```
N..
N60 T09 D09 M06 (CHASER R2)
N50 G00 X.. Y.. Za
N60 G31 Xa Ya Z-20 K3 F2 P4 EF1
N70 G80 G00 Z..
N..
```



5.10.15.1 Cycle repeat to achieve the thread precision

```
N..
N.. G00 X.. Y.. Z..                          Chaser-holder approach
N100 S300 M40 M03 G31 X.. Y.. Z.. K.. P..    Cycle
N110 G00 Z200 M00 $0 CONTROLE                Retraction and spindle stop and display of a
                                             message (see chapter 5.20)
/N120 G79 N100 G04 F0.5
N130 ...
N..
```

Retraction and spindle stop in N110

Check of thread cutting precision.

If thread cutting is correct:

- Confirm by pressing the block skip key " / " on the NC and restart the cycle.
- The system skips block N120 and goes to block N130 (do no forget to inhibit block skip " / " for the rest of the program).

If thread cutting is not correct:

- Adjust the chaser. Do not press the block skip key " / ".
- Restart the cycle. When reaching block N110, the system jumps to block N100 and repeats the cycle.
- At the end of block N100, select one of the two procedures after checking thread cutting precision again.



It is necessary to program a dwell in N120 so that the cycle can be repeated.

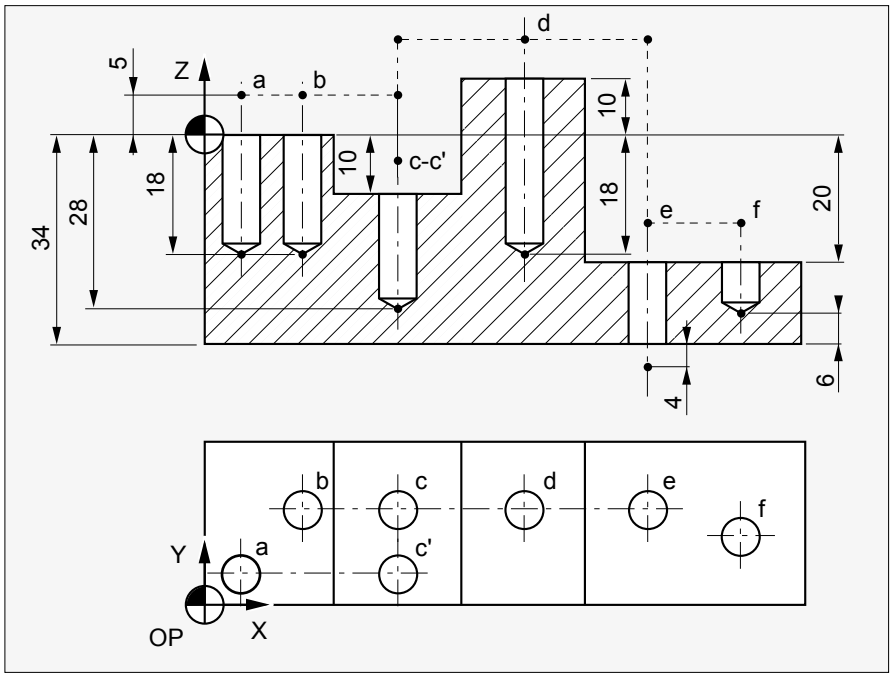
5.10.16 Cycle summary

CYCLES / BREAKDOWN OF MOVEMENTS	G31 Thread chasing	G45 Pocket cycle	G81 Center drilling	G82 Counter boring	G83 Peck drilling	G84 Tapping	G85 Reaming	G86 Boring with indexed spindle stop	G87 Drilling with chip breaking	G88 Boring and facing	G89 Boring with dwell in bottom of hole
Downward	Work-infeed synchronized to the spindle		Work	Work	Rapid then work with n consecutive penetrations (P Q)	Work	Work	Work	Work with in consecutive penetrations (P Q)	Work	Work
Number of constant in-feeds		Prog by EP			Prog by EP				Prog by EP		
Retraction after infeed					Rapid						
Clearance after pecking					Prog by EP						
Back-off between 2 in-feeds									Prog by EP		
Dwell on each infeed					Prog by EP				Prog by EP		
HOLE BOTTOM		See Cycle details	Dwell	Prog by EF	Prog by EF on combined machine used for turning	Prog by EF					Prog by EF
			Spindle			Rotation reversal		Indexed stop programmed by EC			
			Retraction in the plane					Fixed or programmed by EP			
Upward Rapid			Rapid	Rapid	Rapid	Work	Work rate or other feed rate if EF present	Rapid	Rapid at the end of penetration	Rapid after confirmation by the operator	Work
End of upward travel						Spindle rotation reversal		Rapid return of the X and/ or Y axes in the boring axis and reset of the spindle to the initial state			

5.10.17 Examples of G81 to G89 cycles

Sequence of drilled holes in the XY plane

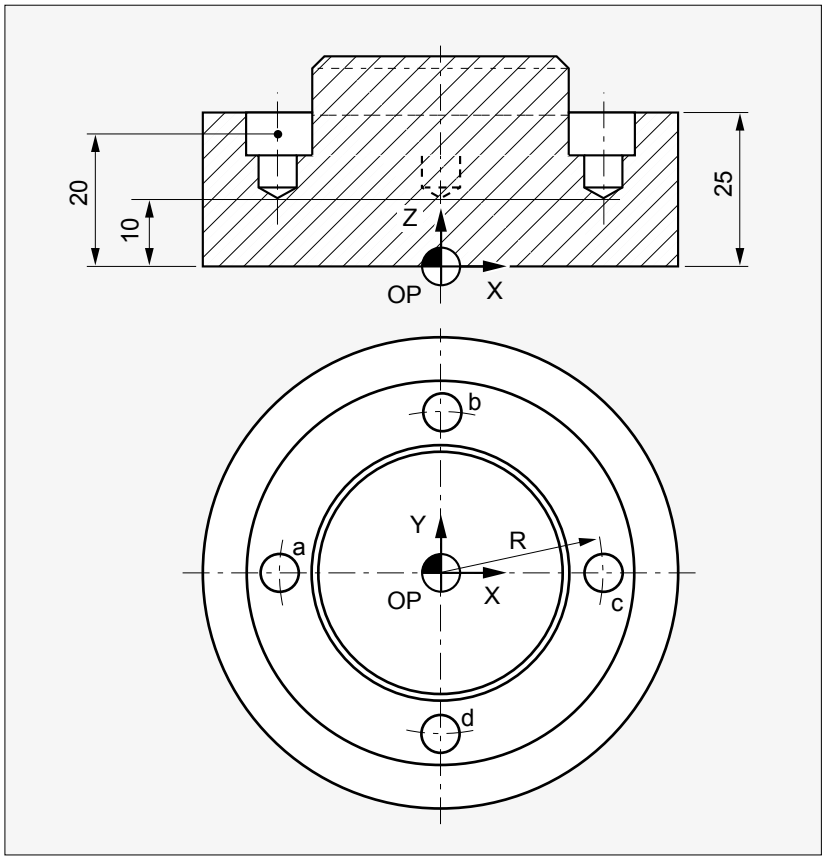
Drilling to different depths and positioning of the tool on different levels with ER.




```
%56
N10 G00 G52 Z..
N20 T12 D12 M06 (DRILL)
N30 S1000 M40 M03
N40 G00 Xa Ya Z50                    Tool approach
N50 G81 Z-18 ER5 F80                 Drilling cycle
N60 Xb Yb                            Cycle
N70 Xc ER-7 Z-28                     Cycle
N80 Yc'                              Cycle
N90 ER12
N100 G83 Xd Yd Z-18 P7 F100          Peck drilling cycle
N110 G81 Xe ER-17 Z-38               Drilling cycle
N120 Xf Yf Z-28                      Cycle
N130 G80 G00 G52 Z..
N..
```

Sequence of drilled holes in the XY plane

Drilling four holes with positioning by circular interpolation inside a groove.

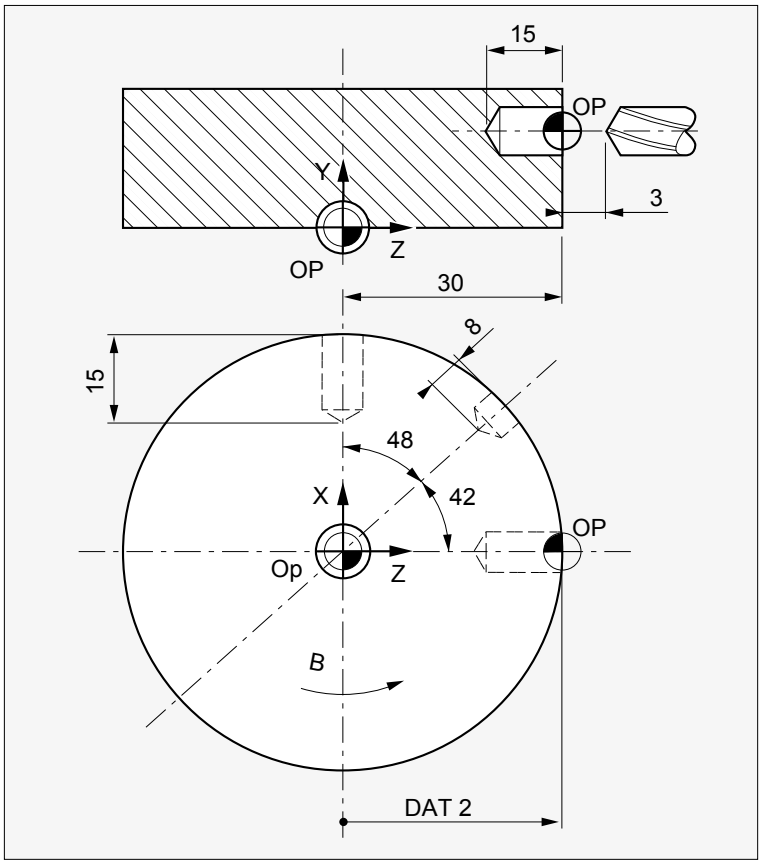



```
%45
N10 G00 G52 Z..
N20 T02 D02 M06 (DRILL)
N30 S1000 M40 M03
N40 G81 Xa Ya ER20 Z10 F80          Drilling cycle
N50 G02 Xb Yb I0 J0 (or R)          Cycle
N60 Xc Yc I0 J0 (or R)              Cycle   (Note the repetition of I0 and J0)
N70 Xd Yd I0 J0 (or R)              Cycle
N80 G00 G80 G52 Z..
N..
```

Sequence of drilled holes in the XY plane on a machining center with rotary axis B

Origin offset on Z.

Drilling to different depths, positioning by rotating the B axis.



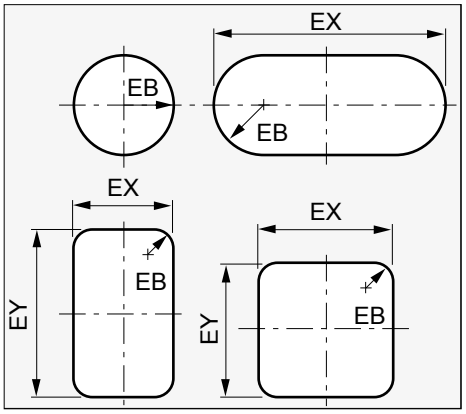

```
%89
N10 G00 G52 Z..
N20 T08 D08 M06 (DRILL)
N30 S900 M40 M03
N40 G00 X0 Y0 Z3 B0
N50 G82 Z-15 EF1.5 F70          Drilling cycle
N60 B42 Z-8                     Cycle
N70 B90 Z-15                    Cycle
N80 G80 G00 G52 Z..
N..
```

5.11 Milling cycles

5.11.1 Simple pocket cycle ► G45



G45	Simple pocket cycle.



This cycle is used to machine circular, oblong, rectangular and square pockets.

The primary and secondary axes are programmable in absolute dimensions and define the pocket center in the plane and the pocket depth along the tool axis.

Syntax (XY plane)

N.. [G17] **G45** X.. Y.. Z.. [ER..] EX.. EY.. [EB..] P.. Q.. [I..] [J..] [EG2/EG3] EP.. EQ.. EI.. EJ..

G17	XY plane selection.
G45	Pocket cycle.
X.. Y..	Position of the pocket center.
Z..	Pocket bottom end point.
ER..	Retraction plane along the tool axis.
EX..	Pocket dimension on the X or U axis.
EY..	Pocket dimension on the Y or V axis.
EB..	Radius of a circular pocket if EB is programmed alone. Radius of an oblong pocket. Fillet for other pockets.
P..	Axial roughing path.
Q..	Lateral roughing path.
I..	Axial finishing path.
J..	Lateral finishing path.
EG2/EG3.	Pocket cutting direction (default EG3): - EG2: Clockwise (Conventional milling). - EG3: Counter-clockwise (Climb milling).
EP..	Value of the axial roughing feed rate.
EQ..	Value of the lateral roughing feed rate.
EI..	Value of the axial finishing feed rate.
EJ..	Value of the lateral finishing feed rate.

The coordinates can be expressed in absolute (G90) or incremental (G91) except of course EX and EY which represent the length and width of the pocket.

Properties of the Function

Function G45 is nonmodal. None of the arguments in the cycle are modal.

Cancellation

Function G45 is cancelled at the end of the block.

Notes

When the start of spindle rotation is programmed in the block, it must be placed before function G45 and its arguments.

Example: N.. S1000 M03 M40 G45...

If tool correction D.. is missing when cycle G45 is called, the system returns error message 898.

When the cycle is programmed, the system must be in state G40 (G41 or G42 radius offset cancelled).

When executing a cycle programmed with axial or lateral roughing and/or finishing passes, if only one feed rate (EP, EQ, EI or EJ) is programmed, this feed rate is used as default.

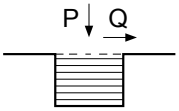
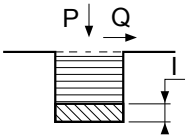
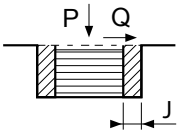
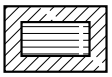
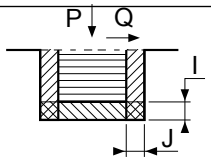
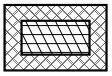
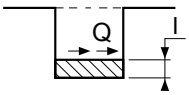
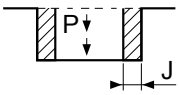
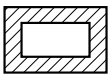
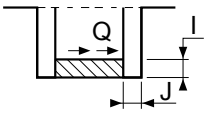
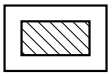
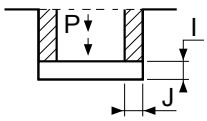
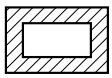
If no:
- axial feed rate (roughing or finishing) is programmed, the system returns error message 892,
- lateral feed rate (roughing or finishing) is programmed, the system returns error message 893.

The cycle calls the axis address equivalence table (see chapter 7.4). Use of the table is incompatible with the cycle call (it is initialized at the beginning of the cycle and restored initialized at the end of the cycle).

In the ZX and YZ planes, the pocket dimension in the Z (or W) axis is programmed by EZ..

During machining:
- The cycle cannot be interrupted until the pocket cycle is completed on a depth level (no possibility of mode change).
- The cycle cannot be modified until it has been completely executed.

5.11.1.1 Pass setting possibilities

Lateral, axial roughing	Lateral finishing	Axial finishing	SIDE VIEW	TOP VIEW
P . . Q . . Axial and lateral roughing				
P . . Q . . I . . Axial and lateral roughing and axial finishing of the bottom				
P . . Q . . J . . Axial and lateral roughing and lateral finishing of the sides				
P . . Q . . I . . J . . Sequenced axial and lateral roughing and lateral finishing (for each axial pass setting)				
Q . . I . . Axial finishing of the bottom by value I				
P . . J . . Lateral finishing of the sides by value J				
Q . . I . . J . . Axial finishing of the bottom up to lateral value J				
P . . I . . J . . Lateral finishing of the sides up to axial value I				

5.11.1.2 Roughing cycle with a single tool

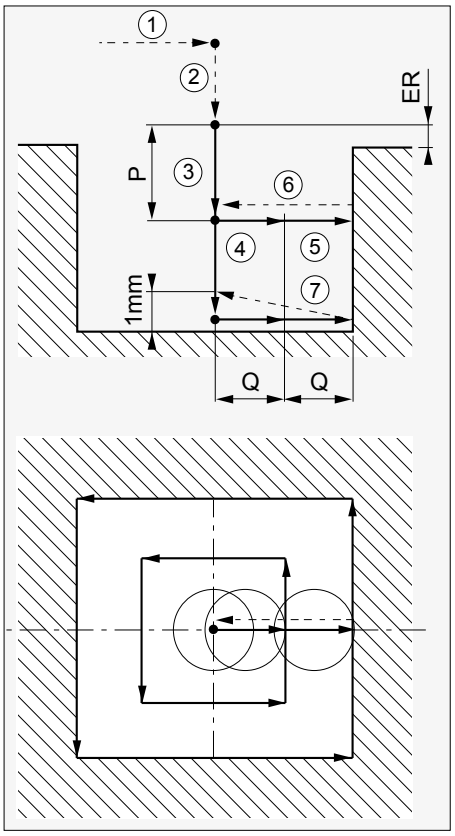
Cycle with axial (P) and lateral (Q) roughing parameters.

Example:

```
G45 X.. Y.. Z.. ER.. EX.. EY.. P.. Q.. EP.. EQ..
```

5.11.1.3 Roughing Cycle Steps

- Step 1: Rapid positioning of the tool to the center of the pocket.
- Step 2: Rapid axial positioning along the tool axis.
- Step 3: Axial penetration at the programmed feed rate to depth P.
- Step 4: Lateral positioning at the programmed feed rate to value Q (along the small side).
- Execution of the first pocket contouring (and subsequent contouring if any).
- Step 5: Lateral positioning on the final contour by value Q.
- Execution of the last contouring to the external pocket dimensions.
- Step 6: Rapid repositioning to the pocket center for execution of a new plunge and a new contouring to depth P following the same steps as 3, 4 and 5.
- Step 7: After execution of the last contouring to depth Z, the tool is rapidly re-positioned to the pocket center with a clearance of 1 mm along the tool axis then retracted to programmed position ER.

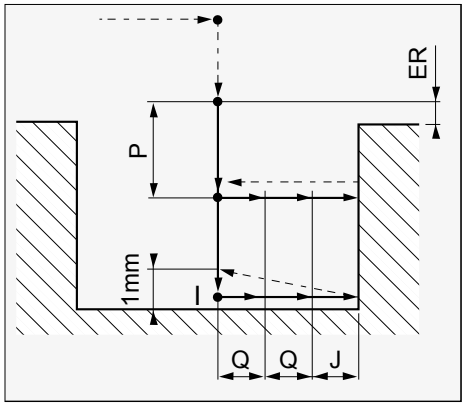


5.11.1.4 Roughing and finishing cycle with a single tool

Cycle with axial (P) and lateral (Q) roughing and axial (I) and lateral (J) finishing parameters.

Example:

```
G45 X.. Y.. Z.. ER.. EX.. EY..
P.. Q.. I.. J.. EP.. EQ.. EI.. EJ..
```

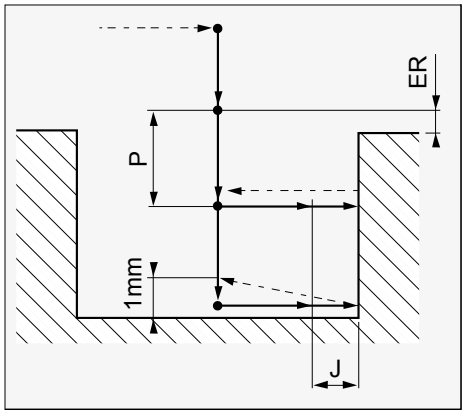


5.11.1.5 Finishing cycle alone with a new tool

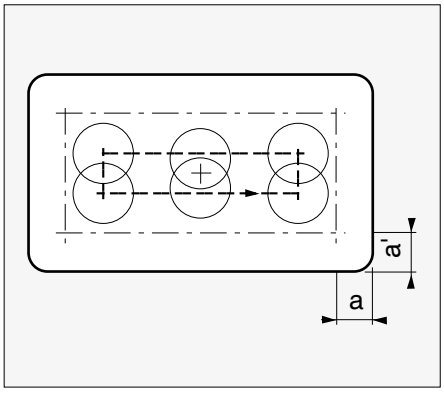
Cycle with depth (P) and lateral finishing (J).

Example:

```
G45 X.. Y.. Z.. ER.. EX.. EY..
P.. J.. EP.. EJ..
```



Oblong, square and rectangular pockets are roughed in such a way that the values of the lateral passes in the two axes are equal (a = a').



5.11.1.6 Pass setting in the plane (Q)

R = Tool radius, L = dimension of the small side

- If Q > 2R, the system returns error message 887.
- If Q = or < R and the result of division of the half-distance to be roughed by the pass set is not an integer, the system computes a new setting for pass Q.

The number of passes is equal to (L/2-R)/Q = m (rounded to the integer above if necessary).

Determination of the new pass setting: Q' = (L/2-R)/n

The value of Q' is rounded off to the next higher micrometer (µm).

In this case, the value of the last pass is:

- L/2-R-(n-1)Q' (value less than Q').
- If R < Q < 2R, the value of the first pass is R/2 (except for circular pockets) and the value of the subsequent passes is Q'=L/2-R/n.

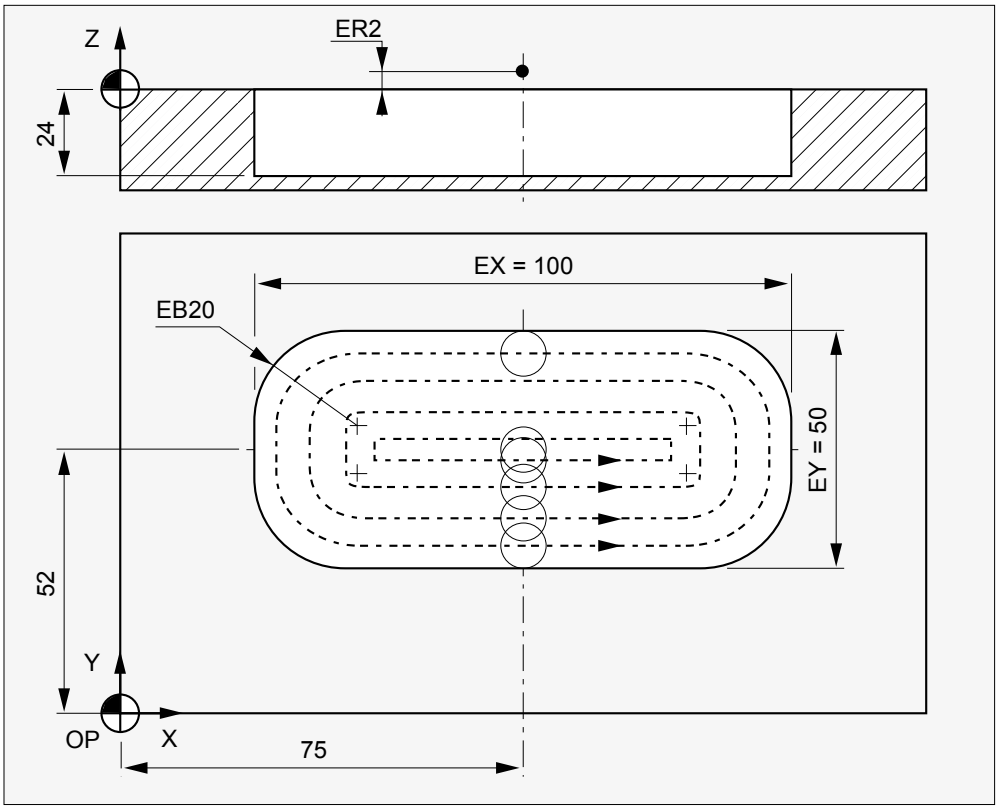
If the cutter radius does not allow at least one roughing pass or the finishing pass to be cut at the programmed value, the system returns error message 896.

5.11.1.7 Pass setting along the tool axis (P)

The system computes a new pass setting P' using the same formula as for Q' if the result of division of the total depth by the pass set is not an integer.

Examples

Execution of a rectangular pocket in the XY plane (G17)



Roughing alone without finishing.

```
%55
N10 G00 G52 Z..
N20 T04 D04 M06 (CUTTER DIAMETER=8 CENTER CUT)
N30 S3000 M03 M40
N40 G45 X75 Y52 Z-24 ER2 EX100 EY50 EB20 P5 Q7 EP100 EQ500
N50 G00 G52 Z..
N..
```

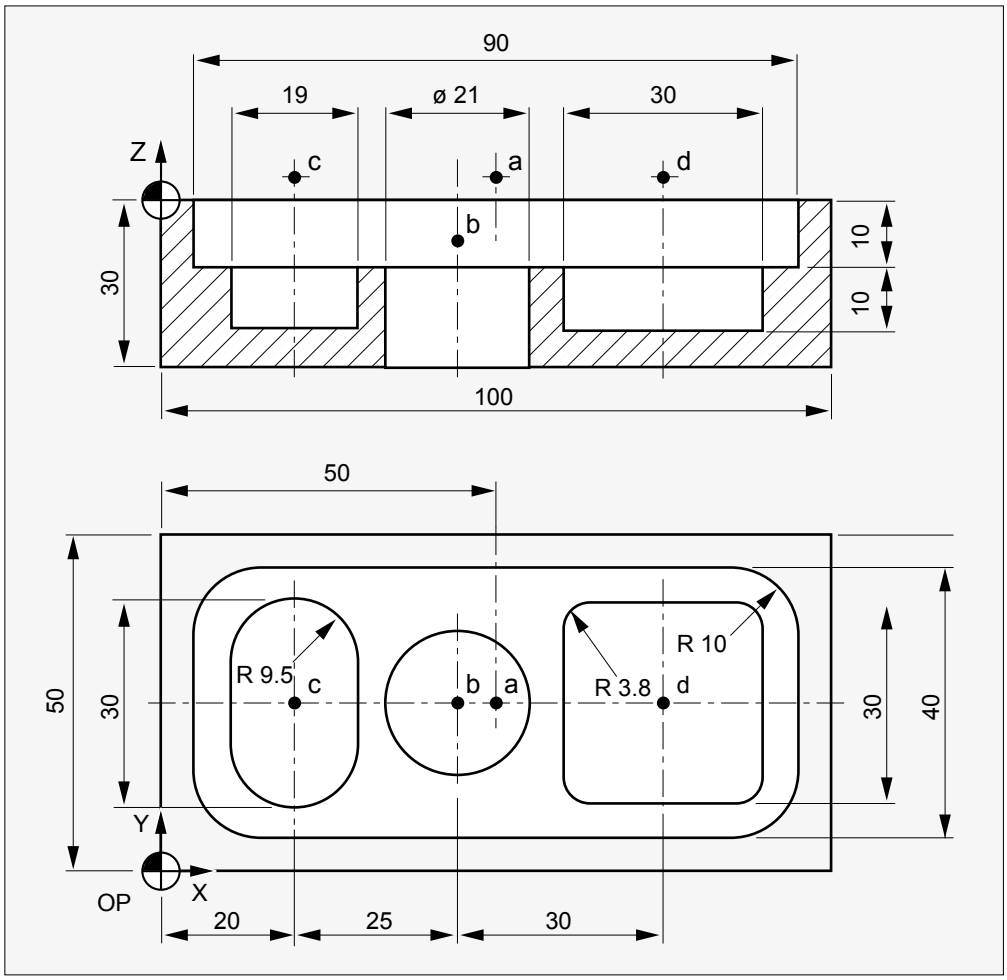
Roughing and finishing with the same tool.

```
%55
N10 G00 G52 Z..
N20 T04 D04 M06 (CUTTER DIAMETER=8 CENTER CUT)
N30 S3000 M03 M40
N40 G45 X75 Y52 Z-24 ER2 EX100 EY50 EB20 P5 Q7 I0.3 J0.3 EP100 EQ500 EI70 EJ300
N50 G00 G52 Z..
N..
```

Execution of a sequence of pockets in the XY plane

The rectangular, circular and oblong pockets are executed with a single tool.

The square pocket is executed with two tools (roughing + finishing tool).




```
%206
N10 G00 G52 Z..
N20 T01 D01 M06 (CUTTER DIAMETER=10 CENTER CUT)
N30 S600 M40 M03 M08
$0 RECTANGULAR POCKET
N40 G45 X50 Y25 Z-10 ER2 EX90 EY40 EB10 P3 Q8 EP50 EQ150
$0 CIRCULAR POCKET
N50 G45 X45 Y25 Z-26 ER-8 EB10.5 P3 Q8 EP50 EQ150 J0.5 EJ200
$0 OBLONG POCKET
N60 G45 X20 Y25 Z-20 ER-8 EY30 EB9.5 P3 Q8 EP50 EQ150 I0.5 J0.5 EI70 EJ200
$0 SQUARE POCKET ROUGHING
N70 G45 X75 Y25 Z-20 ER-8 EX29.6 EY29.6 P3 Q8 EP50 EQ150
N80 G00 Z200 M05 M09
$0 SQUARE POCKET FINISHING
N90 T02 D02 M06 (CUTTER DIAMETER=6 CENTER CUT)
N100 S1000 M40 M03 M08
N110 G45 X75 Y25 Z-20 ER-8 EX30 EY30 EB3.8 P9.5 I0.5 J0.5 EP50 EI50 EJ100
N120 G00 G52 Z.. M05 M09
N130 M02
```

5.11.2 Spline interpolation ► G48, G06, G49



General

Spline curve interpolation is a mathematical curve fitting method. Spline curves are continuous curves connecting a series of points.

Spline curve interpolation insures continuity of the tangent and a constant acceleration in each of the points specified on the programmed paths.

Machining of a spline curve is programmed by:
- defining the points on the curve,
- specifying a curve execution order.

A spline curve can be inhibited by programming.

Programming

G48	Spline curve definition.

A spline curve definition includes several instructions:
- the definition function,
- the curve number,
- the blocks defining the points on the curve.

G06	Spline curve execution command.

A spline curve execution command is given by a block containing G06 followed by the number of the curve to be executed.

G49	Spline curve deletion.

The system allows memory space to be freed by deleting curves that have already been executed.

A curve is deleted by programming G49 followed by the number of the curve to be deleted.

5.11.2.1 Spline curve definition ► G48

G48	Spline curve definition.

Syntax (XY Plane)

N.. **G48** NC.. H../N.. N..

G48	Spline curve definition function.
NC..	Argument defining the curve number.
H..	Number of the subroutine in which are defined the points on the curve (optional).
N.. N..	Numbers of the first and last block defining the points on the curve.

Property of the Function

Function G48 is nonmodal.

Cancellation

Function G48 is cancelled at the end of the block.

Blocks defining the points of a curve

The first and last block defining a curve should include the starting and ending tangents. If the tangents are not known, these blocks must be empty.

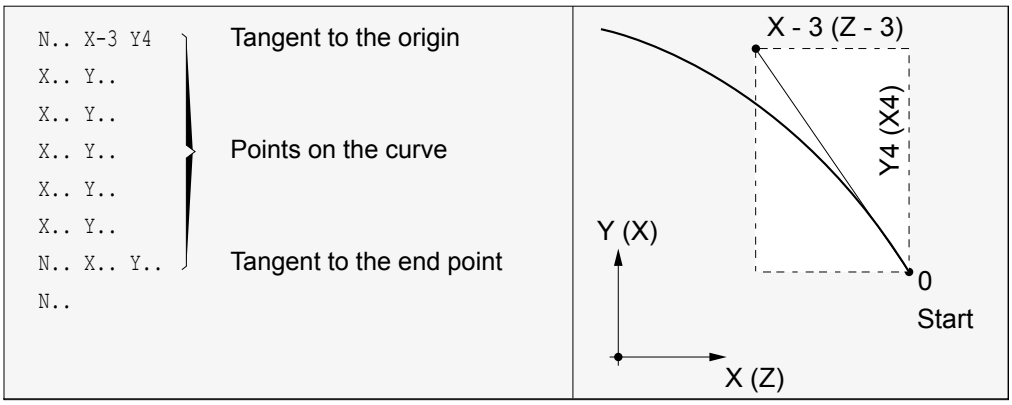
All the definition blocks other than the first and last (tangent to the start and end points) must include points on the curve (no empty lines). Otherwise, the curve would be different from the one desired.

The first point definition block must include all the axes concerned by spline curve interpolation.

If an axis is not programmed in this block, it is not included in spline the curve interpolation even if it is programmed in the following blocks. It is necessary to program the same position in this first block as in the block preceding function G06.

Blocks defining the tangents and points of a spline curve

The projections along the X and Y axes of the tangent to the curve origin have X-3, Y4 as incremental values.



The number of points defining a spline curve is limited to 255. If this value is exceeded, the system returns error message 196.

The points in a spline curve must be defined by ISO programming.

A spline curve must be defined by at least three points. If it is defined by less than three points, the system returns error message 604.

Function G48 must be programmed in state G40, i.e. without radius offset (G41 or G42). Otherwise, the system returns error message 140.

The following functions can be programmed on the curve definition blocks:
- miscellaneous M functions,
- technological F, S or other functions.

Special characters such as $, ( ), etc. should not be programmed in the curve definition blocks.

It is not recommended to program modulo 360 degree axes with spline curves (problem of sign and no movement).

A closed curve is not processed automatically unless the tangents to the start and end points are identical.

Saving spline curve data

The curve coefficients are stored in tables on the program stack. When the program stack is full, the system returns error message 195. In this case, the stack size can be extended (see chapter 8).

The following data are stored for each spline curve defined:
- 1 table with 5 entries,
- 3 tables with P x Q entries, where:
- P: number of axes concerned
- Q: number of points in the profile
- 1 table with P entries.

5.11.2.2 Spline curve execution ► G06

G06	Spline curve execution command.



Syntax

N.. **G06** NC..

G06	Function forcing execution of a spline curve.
NC..	Number of the curve to be executed.

Property of the Function

Function G06 is nonmodal.

Cancellation

Function G06 is cancelled at the end of the block.

Notes

The following functions cannot be programmed in the block containing function G06 NC..:
- F : feed rate,
- S : spindle speed,
- T : tool change.

Error Numbers and Messages for Spline Curve

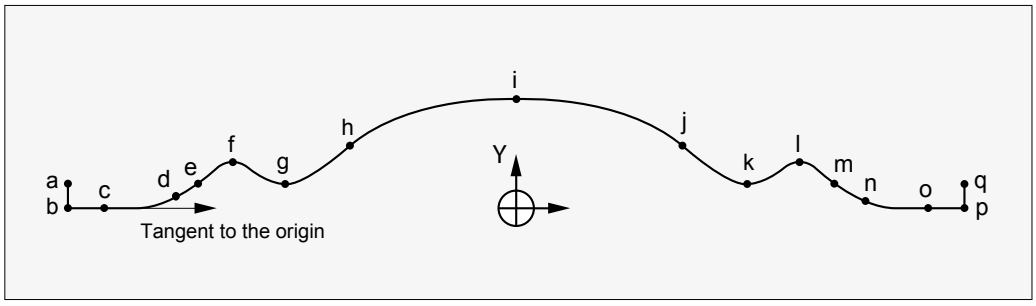
The errors are in error category 600 (see chapter 9).

Examples

Programming of a spline curve defined in the XY plane with a call to definition blocks in the main program

```
%350
G79 N100
N10 X1 Y Z                    Tangent to the origin
X-240 Y0 Z0 F300              Start of the curve and definition of the
X-200 Y10 Z5 (c)              axes to be interpolated
X-180 Y20 Z10
X-160 Y30 Z15
X-140 Y20 Z20
X-80 Y40 Z35
X0 Y60 Z40                    Points on the curve (c to n)
X80 Y40 Z35
X140 Y20 Z20
X160 Y30 Z15
X180 Y20 Z10
X200 Y10 Z5
X240 Y0 Z0 (n)
N20                           No tangent in the end point
N100 T3 D3 M6 (TOOL R5)
N110 S3000 M40 M3
N120 G48 NC5 N10 N20          Definition of the curve
N130 G0 X-300 Y20             Point a
N140 Z0
N150 G1 G41 Y0                Point b
N160 X-240                    Point c
N170 G06 NC5                  Curve execution command
N180 G1 X300                  Point p
N190 G0 G40 Y20               Point q
N200 G0 G52 Z0 M5
N210 M2
```

Representation of machining



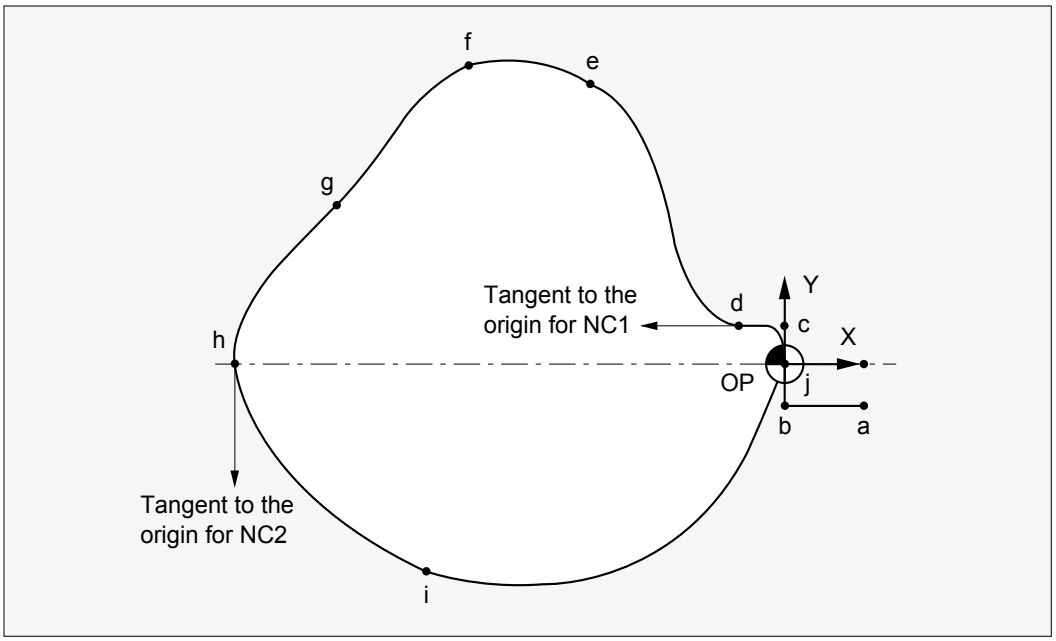
Programming of two consecutive spline curves defined in the XY plane with call to definition blocks in two subroutines

```
%300
N10 G0 G52 Z0
N20 T1 D1 M6 (CUTTER R5)
N30 S2000 M40 M3
N40 G92 R1
N50 G48 NC1 H31 N10 N70          Curve 1 definition
N60 G42 X20 Y-10                 Point a
N70 Z-3
N80 G1 X0 F200                   Point b
N90 EA90 EB5 ES-                 Point c
N100 EA180 X-13 Y10              Point d
N110 G06 NC1                     Curve 1 execution command
N120 G48 NC2 H32 N10 N50         Curve 2 definition
N130 G06 NC2                     Curve execution command
N140 G0 G40 X20
N150 G52 Z0 M05
N160 M02

%31
N10 X-1 Y0                       Tangent to the origin
N20 X-13 Y10
N30 X-50 Y70                     Points on curve 1 (d, e, f, g, h)
N40 X-78 Y75
N50 X-104 Y55
N60 X-140 Y0
N70 X0 Y-1                       Tangent to the end point

%32
N10 X0 Y-1                       Tangent to the origin
N20 X-140 Y0
N30 X-82 Y-60                    Points on curve 2 (h, i, j)
N40 X0 Y0
N50 X1 Y5                        Tangent to the end point
```

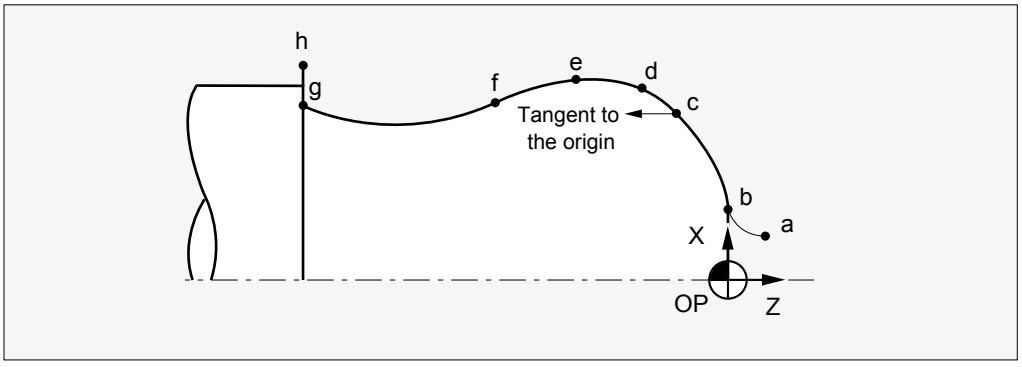
Representation of machining



Programming of a spline curve defined in the ZX plane with a call to definition blocks in the main program

```
%70
N10 ...
N..
N..
$0 FINISHING OF THE SPLINE PROFILE
N180 G0 G52 X0 Z0 G97 S900 M05
N190 G79 N220                          Jump to block N220
N200 X0 Z-1                            Tangent to the origin of the curve
G01 X35 Z-10 F0.15
X37 Z-13
X40 Z-25                               Points in the curve (c, d, e, f, g)
X35 Z-50
X40 Z-80
N210                                   No tangent to the end point
N220 T01 D01 M06 (FINISHING TOOL)
N230 S900 M03 M40
E11005=0                               Forcing programming on radius
N240 G48 NC1 N200 N210                 Spline curve definition
N250 G0 G42 X10 Z5                     Point a
N260 G96 S300
N270 G95 F0.3
N280 G02 X15 Z0 I15 K5                 Point b
N290 G03 X35 Z-10 R30                  Point c
N300 G06 NC1                           Curve execution order
N310 G01 G40 X45                       Point h
E11005=1                               Forcing programming on diameter
N320 G77 N180 N180
N330 M02
```

Representation of machining

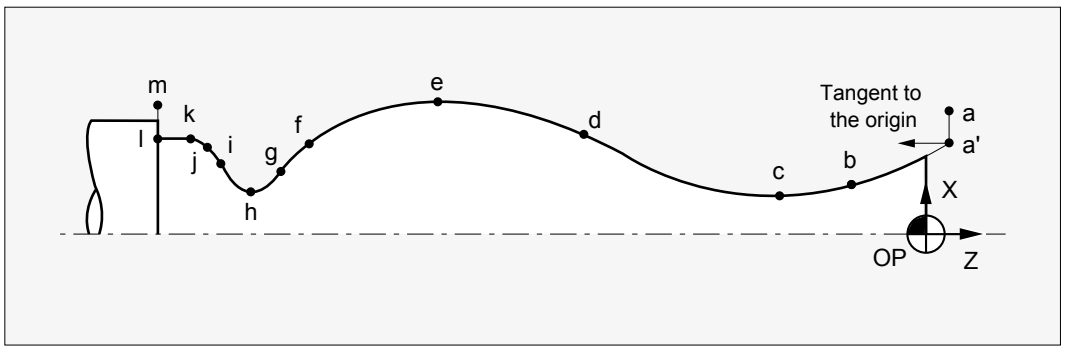


Programming of a spline curve defined in the XZ plane with call to the definition blocks in a subroutine

```
%150
N10
N..
N200 G0 G52 X0 Z0 G97 S500 M05
N210 T03 D03 M06 (FINISHING TOOL)
N220 S2000 M40 M04
E11005=0                    Forcing of programming with reference to the radius
N230 G48 NC1 H151 N10 N130  Curve definition
E11005=1                    Programming with reference to the diameter
N240 G0 X30 Z5              Point a
N250 G92 S3000
N260 G96 S250
E11005=0                    Forcing of programming with reference to the radius
N270 G01 G42 X11 G95 F0.3   Point a'
N280 G06 NC1                Execution order
E11005=1                    Programming with reference to the diameter
N290 G01 Z-105              Point l
N300 G40 X30                Point m
N310 G77 N200 N200
N320 M02

%151
N10 X0 Z-1                  Tangent to the origin
N20 X11 Z5 F0.1             Curve start
N30 X6 Z-10
N40 X5 Z-20
N50 X12 Z-40
N60 X17 Z-60
N70 X12 Z-80                Points on the curve (a to k)
N80 X8 Z-84
N90 X5 Z-88
N100 X9 Z-91
N110 X11 Z-93
N120 X12 Z-95
N130 X0 Z-1                 Tangent to the end point
```

Representation of machining



5.11.2.3 Spline curve deletion ► G49



G49	Spline curve deletion.

This function is used to free the memory space occupied by curves already executed.

Syntax

N.. **G49** NC..

G49	Spline curve deletion.
NC..	Number of the curve to be deleted.

Property of the Function

Function G49 is nonmodal.

Cancellation

Function G49 is cancelled at the end of the block.

Notes

Function G49 must be programmed in state G40, i.e. without radius offset (G41 or G42).

Otherwise the system returns error message 140.

Example

Definition and execution of a spline curve followed by deletion of the curve

```
%300
N10 ...
N.. G79 N200
N110 X.. Y..                    } Points in curve 1
N..
N..
N150 X.. Y..
N..
N200 G48 NC1 N110 N150          Curve 1 definition
N300 G06 NC1                    Curve 1 execution
N..
N..
N450 G49 NC1                    Curve 1 deletion
N..
```

5.11.3 Tridimensional curve smoothing ► G104



G104	Tridimensional curve smoothing.

This function executes tridimensional curves of any shape. It automatically performs polynomial interpolation between a sequence of points defined by their X, Y and Z coordinates.

Curve smoothing must be programmed by a minimum of three blocks.

This feature is provided as an optional macro that can be adapted by the machine builder.

General Syntax

N.. X.. Y.. Z..

N.. [G01] **G104** X.. Y.. Z.. [F..]

[Intermediate points]

N.. G80 X.. Y.. Z..

X.. Y.. Z..	Coordinates of the curve start point.
G104 X.. Y.. Z..	Curve smoothing function and coordinates of the second point on the curve..
Intermediate points	Sequence of blocks each contains the coordinates of the intermediate points on the curve (optional).
G80 X.. Y.. Z..	Cancellation function G80 followed by the curve end point coordinates.

Property of the Function

Function G104 is modal.

Cancellation

Function G104 is cancelled by function G80.

Notes

The block containing the curve smoothing start point must always be programmed before calling function G104. The block containing the end point must always include function G80.

As the function is interpolated directly by the system without prior computations, there is no limit to the number of points in the curve and no stop at the end.

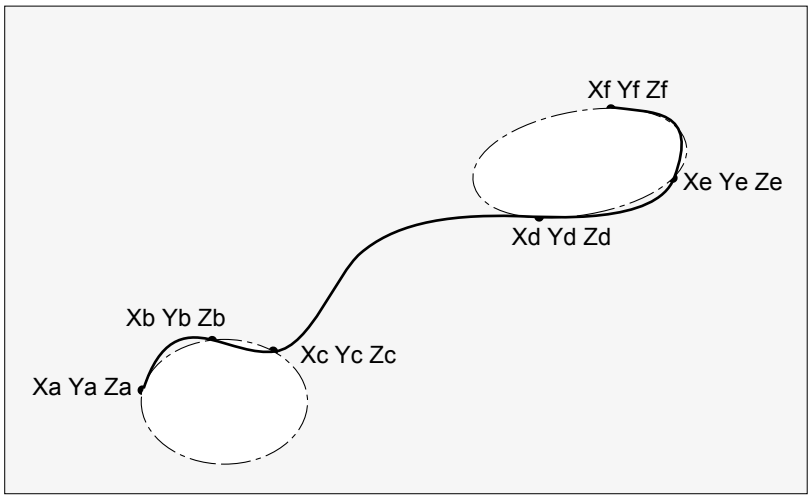
The blocks defining curve smoothing may contain:
- Miscellaneous M functions,
- Program variables L,
- External parameters E.

Curve smoothing can be executed in all the CNC Modes: AUTO, SINGLE, etc. The SEARCH mode can be used for partial execution of a curve.

The location of each point on the curve can be captured empirically by "Teach-in".
(see Flexium⁺ HMI Operator manual M00039)

Example

Programming of curve smoothing to execute polynomial interpolation between points a and f




```
%123
N.. ...
N.. ...
N100 Xa Ya Za                          Curve start point
N110 G01 G104 Xb Yb Zb F200            Smoothing function and second point
N120 Xc Yc Zc
N130 Xd Yd Zd                          Intermediate points
N140 Xe Ye Ze
N150 G80 Xf Yf Zf                      Cancellation and end point
N.. ...
M02
```

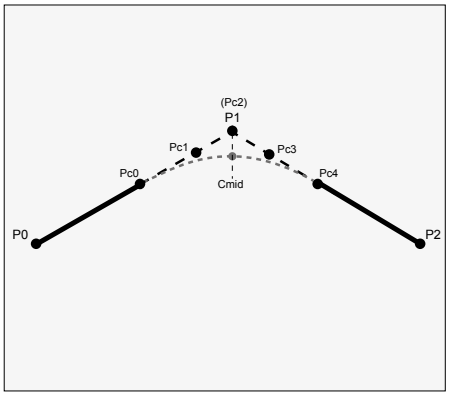
The tangents to the end points (points a and f) are the tangents to the circles going through the first three points (a-b-c) and the last three points (d-e-f) of the curve.

5.11.4 Corner rounding function ► G164



G164 Corner rounding function.

This function is intended to smooth the corners via a spline curve instead of relying to the lag value. By defining a smoother trajectory it generally permits to pass the corners at a higher speed.



Syntax

G164 [Hfile] Nbegin Nend ER.. [EQ..]* [ES..]* [ET..]*

G164	Corner rounding function.
Hfile	Number of the file containing the blocks to process.
Nbegin	First block of the trajectory to process.
Nend	Last block of the trajectory to process.
ERr	**r** specifies the maximum value of cornering error (default value r = 0.033mm) .
EQq	**q=3** cubic polynomial interpolation with conditions of tangent and curvature continuities at ends of smooth curve. **q=4** cubic polynomial interpolation with balanced adjustment between having a continuity in curvature at the ends and reducing the variations of curvature during the transition curve (default value EQ4). **q=5** 5 degrees polynomial interpolation.
ESs	**(s=0/1)** defines a smooth or segmented polynomial interpolation. (default value 0) .
ETt	**(t=0/1)** mode of a tool engagement at the starting of the cycle (default value 0). **1**: (tangential) / 0: (linear interpolation to the entry point).

Properties of the Functions

G164 is not modal (cancelled at the last block of the trajectory defined).

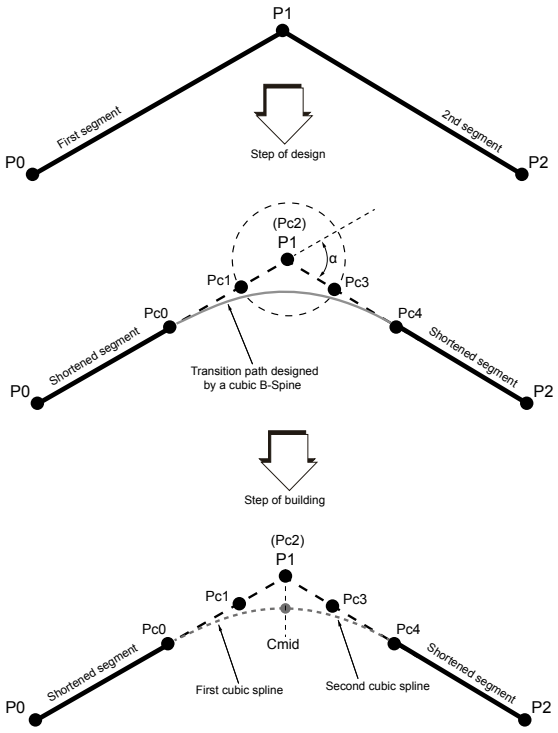
Cancellation

End of block.

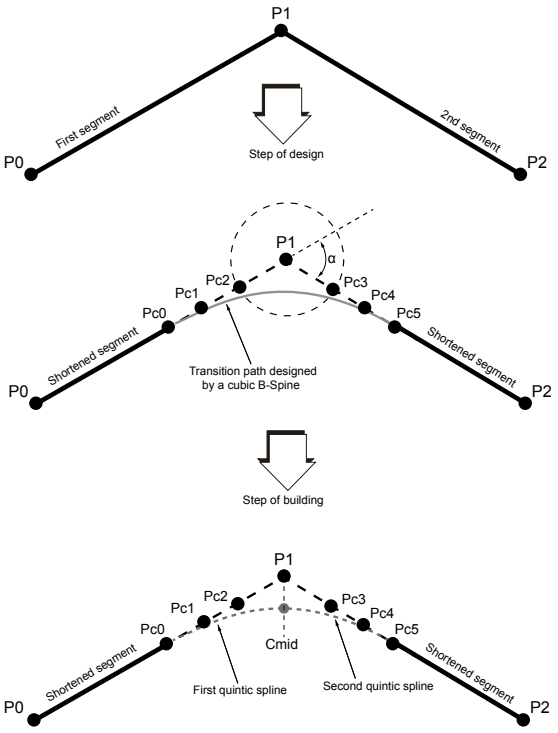
Notes

This function makes sense if the blocks are long enough for a spline to be inserted in between.

Principle of the function



Principle of design of a corner rounding function
Based on cubic B-Spline



Principle of design of a corner rounding function
Based on quintic B-Spline

5.11.5 Trochoidal milling ► G725



G725	Trochoidal milling.

Trochoidal milling is a method of machining used to create a slot wider than the cutting tool diameter.

This is accomplished using a series of circular cuts known as a trochoidal tool path. This cycle is used to machine a path with a trochoïdal motion.

Syntax

N… **G725** EL.. [EF…] ES.. ER.. EH… [EG..] [H..] N.. N.. Q0.9

EL..	Width of the trochoidal machining.
EF..	Machining feed.
ES..	Feed ratio for the back return path.
ER..	Retraction plane in the tool axis.
EH..	Height at which the machining starts.
EG..	Clockwise (2) or counter-clockwise (3) motion.
H..	File where the profile is programmed.
N.. N..	First and last block of the profile (max 128 lines/movements programmed).
Q..	Advancement path.
EE..	If programmed 1, the finishing machining is performed.

The machining feed is programmed with EF, while the Back return speed is calculated as a ratio programmed in ES of this feed (recovery speed = ES*EF).

The machining starts from the first point of the profile (programmed in EI). By default the machining goes counter-clockwise (G3-like) at the left side of the profile. If EG is set to 2 then the machining goes clockwise (G2-like) at the right side of the profile. The path of advancement is programmed in Q as a ratio of the tool radius programmed.

Once the end position is reached, the tool goes to the ER position in G0, then to the XY starting point in G0, and then to EH in G1 G09.

If the feed EF isn't programmed, then it is set as 1000 by default, and ES is set to 1.

It's important that the profile does not intersect itself and that there is enough space to machine without interaction (more formally the path followed by the centres of the circles done by the trochoidal machining does not intersect).

The profile must always be programmed in XY.

If machining involves parallel axes ("U", "V", "W"), then they must be programmed (moved) before entering the macro.

The plan of machining (G17 or G18 or G19) is selected based on tool orientation. So, tool orientation must be properly programmed before entering the macro.

Property of the Function

Function G725 is not modal.

Cancellation

Function G725 is cancelled after execution or by Reset.

Example

```
…
G725 EL50 EF2000 ES3 ER200 EH100 EG3 N1 N3 Q0.9     Execute trochoidal milling
G79 N100                                            Continues the program
N1 G1 X-100 Y0
G1 X0 Y0
G2  I100 J100 EA100
N3 G1 X200 Y-200
…
```

5.11.6 Engraving ► G730

This cycle is used to engrave characters. Characters available are the uppercase and lowercase letters (A-Z and a-z) and some special characters (\!$&/()=?'[]+*@#_-:.;,<>).



G730	Engraving.

The machining starts from the first point of the first letter. From the last position programmed, the tool goes to the ER position in G0, then to the XY of the SE corner of the letter rectangle in G0, and then to EH in G1 G09.

It is possible to write in a straight line with different inclination or on a circle of radius R. If R>0 the text is engraved outside the circle, while if R<0 the text is written inside.

Each letters can be rotated by an additional angle (EX..).

NO parameter is used to pass to the macro the text to be engraved: the macro will fetch the text always from that fixed predetermined variable, called [G730TXT]. Such variable has to be defined in the part-program by the user (see examples).

The dimension of the variable [G730TXT] has to be aligned to the length of the string:

- A string of 7 character fits exactly a simple declaration of variable [G730TXT]
- A string up to 15 characters needs a declaration of the type [G730TXT(2)]
- A string up to 23 characters needs a declaration of the type [G730TXT(3)]
- …
- A string up to ((8*N) – 1) characters needs a declaration of the type [G730TXT(N)].

See examples in the next page.

From the height of the letters (EL..), their width is computed maintaining the same font ratio.

The machining feed is programmed in F, by default it is 1000.

Parameter descriptions

X..	X position of the SE corner of the rectangle including the first letter.
Y..	Y position of the SE corner of the rectangle including the first letter.
Z..	Z position of the engraving end (depth).
EL..	Height of the letters.
F..	Machining feed.
ER..	Retraction plane in the tool axis.
EH..	Retraction plane in the tool axis.
R..	Radius of the circle on top of which the text is engraved.
EA..	Angle of inclination of the text if straight line or angle of start in a circle.
EI..	Space between characters.
EX..	Additional rotation of each character.
EP..	Axial path.

Property of the Function

Function G730 is not modal.

Cancellation

Function G730 is cancelled after execution or by Reset.

Example

Short texts (less or equal to 7 characters)

```
...
VAR
[G730TXT]                                                    Defines string to engrave
ENDV
_ SCPY [G730TXT]/"NUM CNC"
G730 X-130 Y-50 Z90 R0 EL40 EA15 EI20 ER0 EH100 EP2 EX0 F4000
```

Longer texts (less or equal to 15 characters)

```
...
VAR
[G730TXT(2)]                                                 Defines string to engrave
ENDV
_ SCPY [G730TXT]/"demo text !"
G730 X0 Y0 Z90 R200 EL40 EA180 EI20 ER200 EH100 EP2 EX0 F4000
_ SCPY [G730TXT]/"Flexium+"
G730 X0 Y0 Z90 R-200 EL40 EA250 EI20 ER0 EH100 EP2 EX0 F4000
...
```

5.11.7 Tangential Control ► G748



G748	Tangential control.

This function is intended to control a rotary axis in relation with the tangent to a 2D trajectory

This function can be used for example on knife cutting machines

The rotary axis is controlled with respect to the trajectory tangent taking into account a programmable angle. Its position is always reached through the shortest way.

Syntax

G748 axis [EFxxx] [ETxxx] [EAxxx]

G748	Tangential control function
Axis	Name of the rotary axis
EF	Maximum velocity of the rotary axis
ET	Offset angle with respect to the tangent
EA	Trajectory angle above which there is a wait until the rotary axis is in position

EF feed rate

The feed rate at which the axis move to follow the tangency is programmed by the optional argument EF, if EF is not programmed the axis will move at maximum feed.

e.g..: G748 C EF3600 the C axis will follow the path tangency at a maximum feed of 3600 deg/min.

ET tangent offset

The optional argument ET defines an offset in position with respect to the tangent. If ET is not programmed the axis will follow the tangent.

e.g.: G748 C ET90 the C axis will be oriented normal to the profile

e.g.: G748 C ET10 the C axis will be oriented 10 degrees respect to the path tangent.

EA path angle

The optional argument EA defines an angle on the trajectory above which the interpolation will stop until the rotary axis has reached the position required in the second block.

If EA is not programmed the move on the profile never stops which can lead to erroneous positioning on the rotary axes at the beginning of a block forming an angle with respect to the previous.

Property

G748 function is modal

Cancellation

The tangential control is cancelled by the associated G749 function



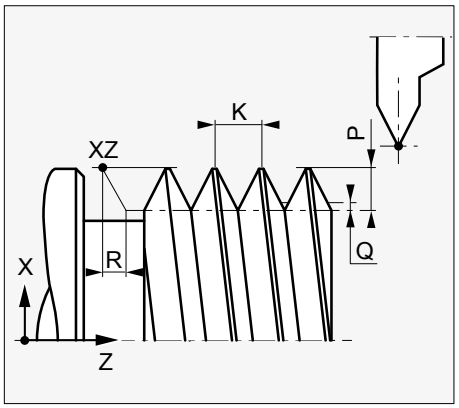
The controlled axis must be full rotary
The axis under control of G748 control can’t be programmed
The axis always follow the tangency, the tangency error will depends on the profile shape and on the tangent axis feed.
The tangent is always reached through the shortest way.

5.12 Turning specific canned cycles

5.12.1 Constant pitch thread chasing cycle ► G33



G33	Constant pitch thread chasing cycle.



This function is used for cutting parallel, tapered or scroll threads.

The threads can be single or multiple starts and can be cut by straight or angular penetration.

The consecutive passes are carried out with reducing depths of cut.

This cycle can only be executed on the main spindle.

Syntax

N.. **G33** X.. Z.. K..[EA..][EB..]P..[Q..][R..][F..][S..] / [ES..] [ER..]

G33	Constant pitch thread chasing cycle.
X.. Z..	Tool position at the end of thread cutting with respect to the program origin.
K..	Pitch in the thread cutting axis (X or Z) in mm.
EA..	Angle for tapered thread (see Fig. 5). - The default value of EA = 0: parallel thread cutting. - EA90: scroll thread cutting.
EB..	Tool penetration angle (see Figs. 3 and 4). - The default value of EB = 0: straight penetration. - EB > 0: penetration down the thread flank in the direction of machining. - EB < 0: penetration down the thread flank opposite the machining direction.
P..	Total thread depth (including Q).
Q..	Depth of the final sizing pass (included in P). - Default no sizing pass (omit Q..). - Q = 0: spring cut required.
R..	Length of the tool retraction slope at the end of a pass (in the thread cutting axis) (see Figs. 1 and 2). The default value of R = 0: retraction perpendicular to the thread cutting axis.
F..	Number of thread starts (format F01) (see Fig. 6). - Default single start.
S.. / ES..	Number of passes (format S02 or ES02) (not including pass Q), default 1 pass. - S..: Passes with reduced depth of cut. - ES..: Constant value passes.
ER..	Approach angle 0<ER<90 This optional parameter indicates threading with multi cut tool (comb).

Property of the Function

Function G33 is nonmodal.

Cancellation

Function G33 is cancelled at the end of the block.

Notes

The modal functions present before the G33 cycle are restored after execution of the cycle.

The thread pitch is applied to the thread cutting axis (axis on which the movement is largest).

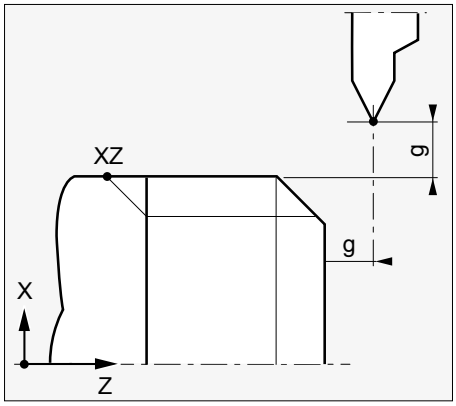
During the cycle, spindle speed override is inhibited (value forced to 100 per cent).

The maximum possible pitch is limited by the feed rate in mm/min acceptable by the machine (feed rate = pitch x rotation speed). Refer to the machine manufacturer's documentation.

Thread cutting start point

The tool must be backed off from the material by a clearance (g).

The clearance in the axis of penetration is mandatory. If the tool positioning and end of thread cutting values in the axis of penetration are the same, the system will generate an error message.



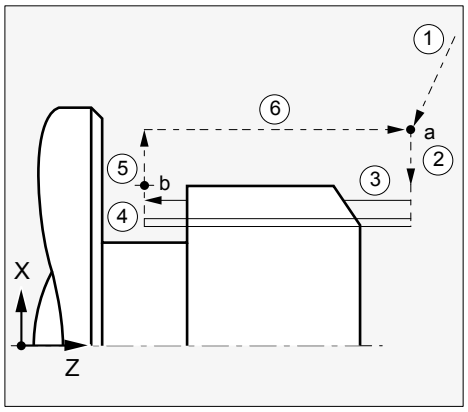
Cycle Steps

The steps below are given for information. The thread cutting cycle contains only the main parameters.

```
N.. ...
N220 G00 Xa Za                    Rapid to start of thread cutting
N230 G33 Xb Zb K.. P.. S..
N..
```

- Step 1: Tool positioning in X and Z to the thread cutting start point (point a).
- Step 2: Rapid advance in the axis of penetration (first pass).
- Step 3: Execution of the first pass in the thread cutting axis Z (point Zb).
- Step 4: Retraction along X (point Xb).
- Step 5: Rapid retraction of the penetration axis.
- Step 6: Rapid return to the thread cutting start point.

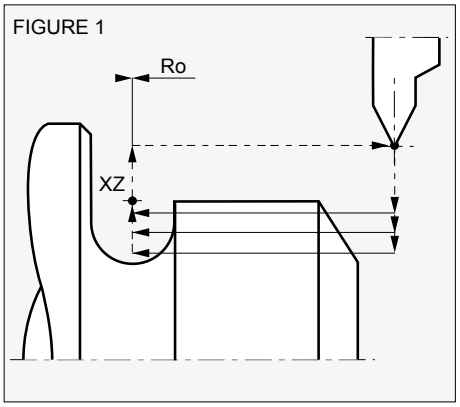
Successive passes are executed in the same way as steps 2 to 6 and may be followed by a spring pass.



5.12.1.1 Tool retraction slope at the end the pass defined by R

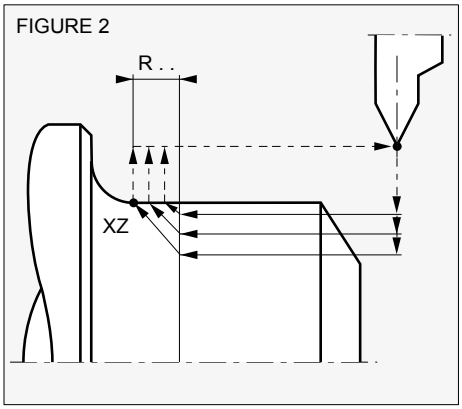
Thread cutting without retraction slope

FIGURE 1



Thread cutting with retraction slope

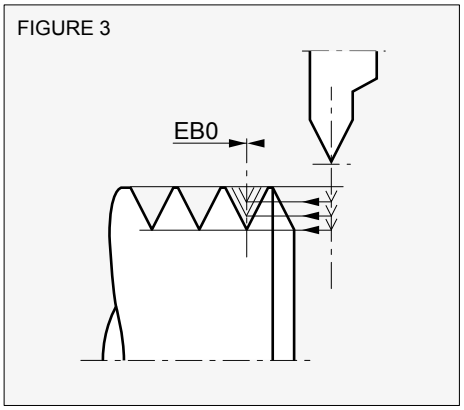
FIGURE 2



5.12.1.2 Penetration defined by EB

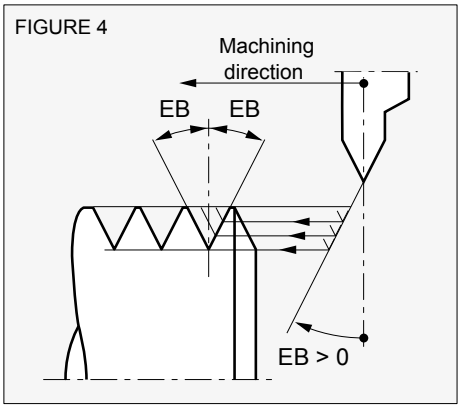
Thread cutting by straight penetration

FIGURE 3



Thread cutting by penetration down the thread flank

FIGURE 4



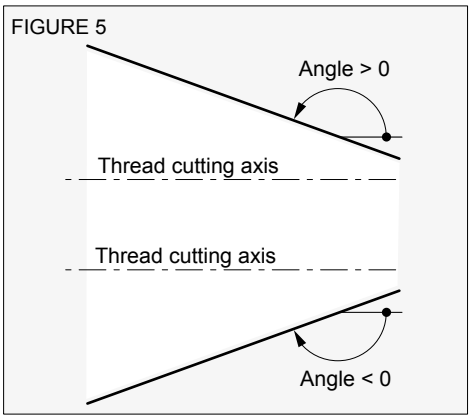
5.12.1.3 Tapered thread cutting, angle defined by EA

The value of EA is positive in the counter-clockwise direction, modulo 180 degrees.

i.e. -45 deg < EA < 45 deg for:
- Z thread cutting axis
- X penetration axis.

i.e. EA > 45 deg or EA < -45 deg for:
- Z axis of penetration
- X thread cutting axis.

FIGURE 5

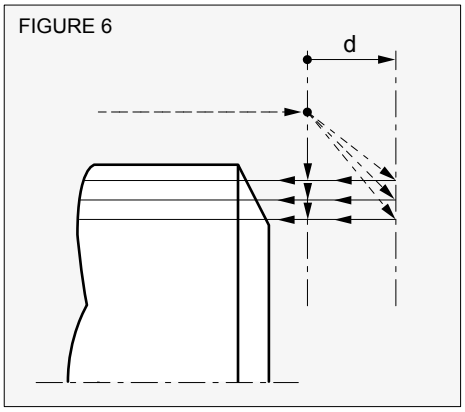


5.12.1.4 Multiple start thread cutting

The offset «d» (pitch) corresponding to each start is in the direction opposite the thread cutting direction.

Each start will be completed at one depth before continuing to the next pass depth.

FIGURE 6



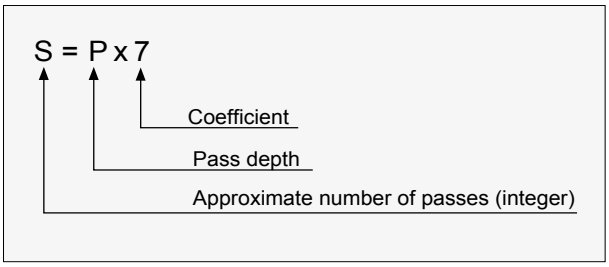
Reminders

Determination of the pass depth (P) for ISO threads.

External thread: 0.613 x pitch

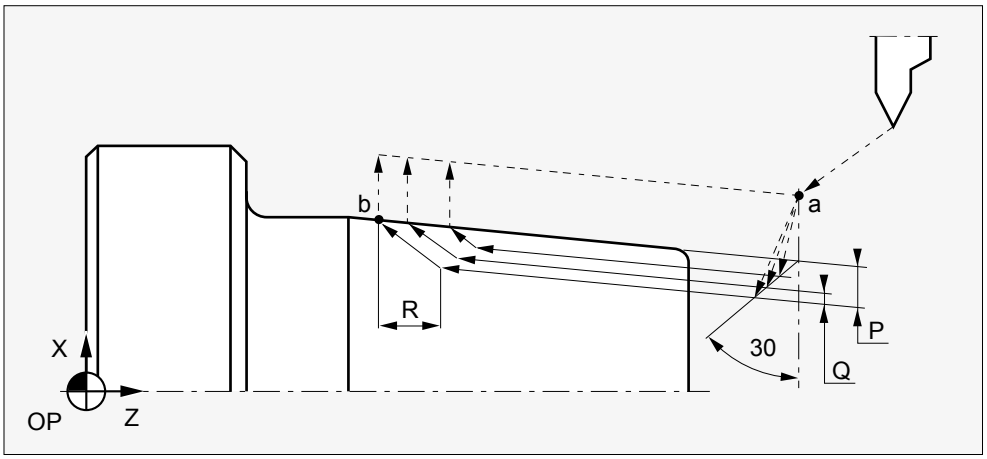
Internal thread: 0.577 x pitch

Determination of the approximate number of passes (S)



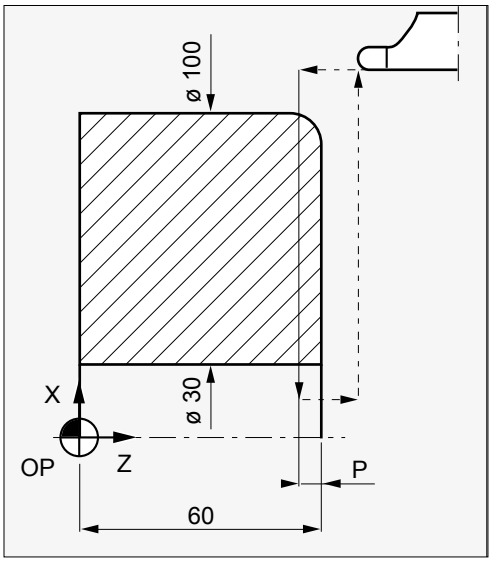
Examples

Cutting of external tapered thread (5 deg)




```
N.. ...
N140 T09 D09 M06 (THREAD CUTTING TOOL, PITCH=1 RH THREAD)
N150 G97 S1000 M40 M03
N160 G00 Xa Za                                    Start of thread cutting
N170 G33 Xb Zb K1 EA175 EB30 P0.61 Q0.02 R4 S5
N.. ...
```

Cutting of front threads

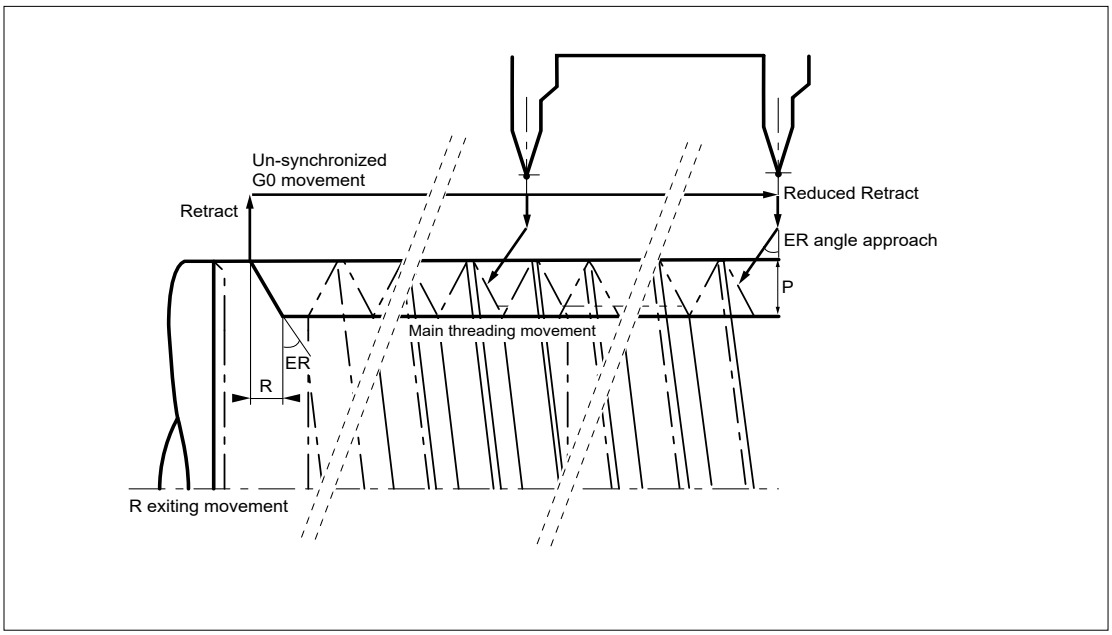



```
N.. ...
$ FRONT THREAD ALONG X
N140 T05 D05 M06 (TOOL R=2)
N150 G97 S200 M40 M03
N160 G00 X110 Z64                          Point a, start of thread cutting
N170 G33 X20 Z60 K4 EA90 P2 Q0 S14
N..
```

5.12.1.5 Multi cut threading cycle

Principe

The left cutter must “penetrate” into the piece with tool feed rate already synchronized with the spindle rotation speed; so the multi-cutters tool has to approach the piece generally quite parallel to the Z axis with an angle, specified by the user via argument “ER” and obviously, the exit from the piece has to be inclined as well.



R and ER parameters options

- If neither R, nor ER arguments are specified, then R will be considered null, the threading will be conventional as previously described

- If argument ER is specified, but argument R is not, then R distance will be computed starting from ER angle, in order to have an exit angle with respect to the approach angle!

- If both R and ER are specified, then we’ll have an entry angle equal to ER argument, and an exit distance given by R argument: the angles can be different!

ER parameter when specified must always be positive. Permitted values are in the range from 0 to 90 degrees, extremes excluded:

- 0 degree **will be not accepted!**
- 90 degrees **will be not accepted!**

The longitudinal end position (Z) is relative to the forward cutter of the tool.

For the start position, the programmer should position the tool taking into account the longitudinal motion due to the approach angle, so that the back cutter is clear from the part.

However as the tool is already synchronized at the beginning of the cut the clearance distance can be minimal.

Possible errors

- **Main thread axis speed limit exceeded**

  Checks based on the comparison between the maximal speed of the main thread axis and the product of pitch times spindle speed.

  - **Error 842** Thread feed X axis exceeds speed limit.
  - **Error 844** Thread feed Z axis exceeds speed limit.

- **Penetration thread axis speed limit exceeded**

  Checks based on the comparison between penetration axis maximal speed and the demanded speed of penetration axis during the phases of retraction (R argument) and approach angle (ER).

  - **Error 843** End of thread feed Z axis exceeds speed limit.
  - **Error 845** End of thread feed X axis exceeds speed limit.
  - **Error 846** Start of thread feed Z axis exceeds speed limit.
  - **Error 847** Start of thread feed X axis exceeds speed limit.

- **Re-threading not allowed in case of multi-cutters tool threading**
  - **Error 840** G38+ NOT allowed with argument ER

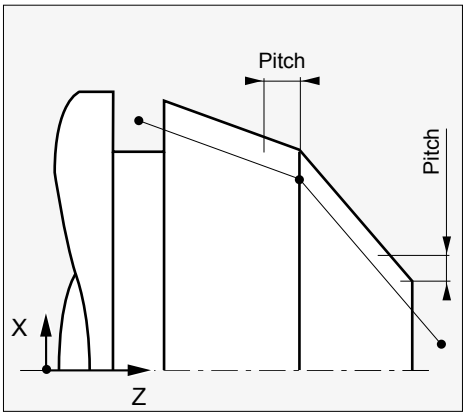
- **Invalid parameter values**
  - **Error 841** ER value: 0 < ER < +90.
    This alarm is issued if a wrong value for ER argument is provided to the macro.
  - **Error 848** End of Thread movement argument R too long.
    This alarm is issued if the R argument is greater than the total threading length.
  - **Error 849** End of Thread movement argument ER too big.
    This alarm is issued if the R value calculated from the ER argument is greater than the total threading length.



G38+ resume thread cutting is not allowed with multi cut threading cycle. Tapered threading is possible with multi cut threading. Obviously a tapered tool must be used.

5.12.2 Sequenced thread cutting ► G38

G38	Sequenced thread cutting.



This function allows the execution of several consecutive thread cutting blocks.

The threads can be cylindrical or tapered.

This cycle can only be executed with the main spindle of any channel.

Syntax

N.. **G38** X.. Z.. K..

G38	Sequenced thread cutting.
X.. Z..	Tool position at the end of thread cutting with respect to the program origin.
K..	Pitch in the thread cutting axis (X or Z) in mm.

Property of the Function

Function G38 is modal.

Cancellation

Function G38 is cancelled by one of functions G00, G01, G02 or G03.

Notes

Different pitches (K) can be programmed in consecutive thread cutting blocks with G38.

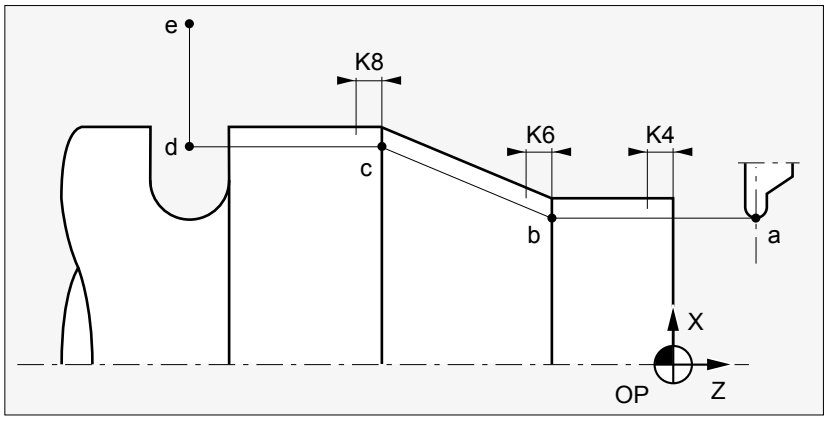
The thread pitch is applied to the thread cutting axis (axis on which the movement is largest).

The consecutive passes, to obtain the required depth, can be programmed by subroutine calls. (During execution of the cycle, spindle speed override by potentiometer is inhibited (value forced to 100 percent).

Function G38 must be programmed:
- without miscellaneous functions (M) in the cycle blocks,
- without use of constant surface speed (nor retract in step for successive passes).

The maximum possible pitch is limited by the feed rate in mm/min acceptable by the machine (feed rate = pitch x rotation speed). Refer to the machine manufacturer's documentation

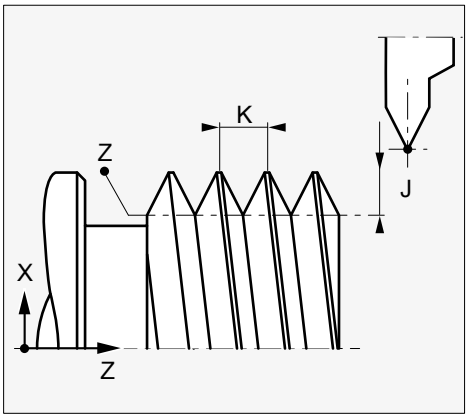
Example




```
N.. ...
N100 T04 D04 M06 (TOOL R=2)
N110 G97 S400 M40 M03
N120 G00 Xa Za                  Approach point
N130 G38 Xb Zb K4               Cycle with 4 mm pitch
N140 Xc Zc K6                   Cycle with 6 mm pitch
N150 Xd Zd K8                   Cycle with 8 mm pitch
N160 G00 Xe Ze                  Cycle cancelled
N..
```

5.12.3 Resume thread cutting ► G38+

G38+	Resume Thread Cutting.



Thread cutting may be interrupted and the part removed from the chuck, for instance to check it. The same part may then be reinstalled in the chuck to continue the thread cutting operation (possibly using a different spindle speed than during roughing).

When reinstalled in the chuck, the part is in a different angular position (and possibly a different axial position) than before.

The resume thread cutting function has been created to teach in the new position of the thread and use it as argument in a resume thread cutting cycle.

Teach-in of the thread position is carried out manually by positioning the tool in the roughed thread and measuring the axial position) (or radial position for thread cutting in X) called Xpos_teachin as well as the angular position of the spindle Spos_ teachin (E90100 for spindle 1).

This cycle can only be executed with the main spindle of any channel.

Syntax

G38+ J<Xpos_delta> EC<Spos_teachin> K.. Z..

J<Xpos_delta> = <Xpos_teachin>- <Xstart>	Difference between the teach-in position of the axis and its start point when thread cutting is resumed, expressed in programming units (mm or inches). For a taper thread, the difference is that of the major axis, i.e., the axis to which the pitch applies.
EC<Spos_teachin> Teach-in	Angular spindle position in degrees.
K..	Thread pitch.
Z..	Thread cutting end point.

Notes

Arguments J and EC must be programmed immediately after function G38+.
When the resumed thread cutting cycle includes several passes with straight or oblique infeed, the values of arguments J and EC remain unchanged.

For sequenced thread cutting, the blocks after the initial block must not include any arguments except the end point and possibly the pitch.

Example

Example of longitudinal thread cutting with tapered retraction

`G0X..Z..`	Positioning at the start point
`G38+ J.. EC.. K.. Z..`	Longitudinal thread cutting
`X.. Z..`	Tapered sequencing of the thread
`G0 X..`	Rapid retraction off the thread.

Constraints

High-speed machining is prohibited for resumed thread cutting to allow on-the-fly exit from the thread. If this function is active when G38+ is analysed, it is automatically cancelled (E11012 = 0).

The gains on the axes must be set consistently with the position loop time constant set in machine parameter P56.

The spindle must be rotating at a steady speed before the block containing G38+ is executed.

5.12.3.1 Inclusion of function G38+ in Cycle G33

Syntax of Resumed Thread Cutting Included in the Call to Cycle G33

The teach-in positions are declared in EC.. for the spindle position and EZ.. or EX.. for the axis, depending on whether thread cutting is axial or radial.

The presence of these two arguments distinguishes between normal and resumed thread cutting.

G33 X.. Z.. P.. EZ.. EC.. K.. [S..] [R..] [EA..] [EB..] [F..]

The difference between the teach-in position of the axis and its initial position is calculated in macro %10033 as well as its assignment to function J after G38+.

Macro %10033 ensures that the syntax is consistent:

It checks that both the teach-in position of the spindle and the teach-in position on the thread axis (major axis) are programmed. For a 45-degree taper thread, either axis is accepted.

In case of an error, it displays the message No 838: $ Restart threading incoherent data.

5.12.3.2 Associating a Teach-in cycle with resume thread cutting cycle G33

An operator may visually note the positions to be taught in and then enter them manually in a G33 block.

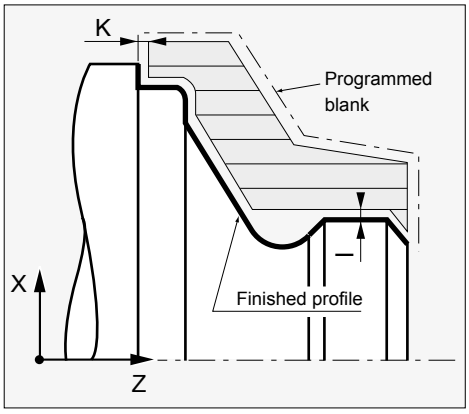
It is also possible to call a subroutine in a program before the G33 cycle to read variables indicating the spindle/axis positions to be assigned to EZ and EC.

Example

```
%1038                                                    Teach-in resume thread cutting
VAR [b] = [.BM69]*3 + [.BM68] + [.BM68] + [.BM67]
[Bpom] = 91124+[b] [Bpos] = 90100+[b]
[ZTeach] [XTeach] [BTeach]
ENDV
M5
WHILE E[Bpom] = 1 DO G4 F.1 $                            Homing on spindle required
ENDW
M12 $                                                    manually set position in thread
                                                         After action on B.CYCLE, read position
[BTeach] = E[Bpos] /10000 [XTeach] = UE70000 [ZTeach] = UE72000
M12 $                                                    Move off the thread
SAVE [BTeach]/[XTeach]/[ZTeach]                          Store position
%10                                                      Main longitudinal thread cutting program
...
G77 H1038                                                Call position teach-in
M4 S...
...
G0 Z... X...                                             Go to start of cycle
G33 Z... X.. K... P.. S… EZ[ZTeach] EC[BTeach]
...
```

5.12.4 Rough Turning/Facing cycle ► G64

G64	Turning/Facing cycle.



This function is used to rough a volume of material located between the specified dimensions of a given blank profile and a finished profile.

This turning or facing cycle can be used for external or internal working.

This cycle can be executed with any spindle. However it generally requires constant surface speed which is only available on the main spindle of each channel.

Syntax

N.. **G64** [N.. N..] / [I.. K..] P.. / R..

N.. BLANK DEFINITION

G64	Turning/Facing cycle.
N.. N..	Numbers of the first and last block defining the finished profile (minimum two blocks, maximum 50 blocks).
I..	Finishing allowance in X: - The default value of I = 0.
K..	Finishing allowance in Z: - The default value of K = 0.
P.. / R..	Cut depth (unsigned): - P: depth of cut in X (turning). - R: width of cut in Z (facing).
N.. BLANK DEFINITION	Sequence of blocks defining the blank dimensions (these blocks are located between G64 and the cycle cancellation code G80).

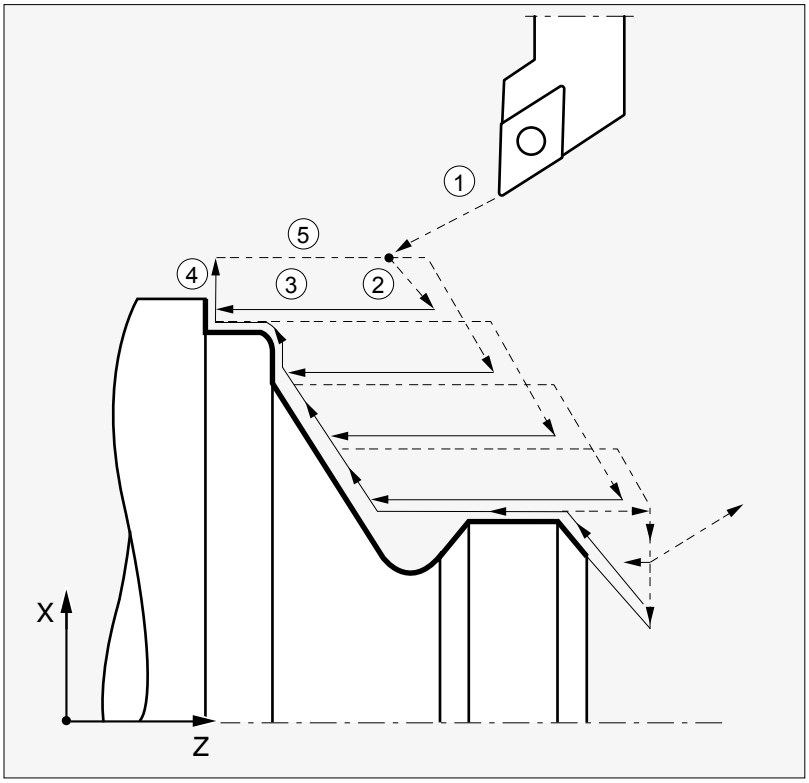
Property of the Function

Function G64 is modal.

Cancellation

Function G64 is cancelled by function G80.

Cycle Steps



- Step 1: Tool positioning in XZ.
- Step 2: Rapid to pass depth (to depth «P» for turning).
- Step 3: Execution of the first pass at the feed rate.
- Step 4: Lift along the profile at the feed rate.
- Step 5: Rapid return to the cycle start point.

Execution of the following passes in the same way as steps 2 to 5, then retraction.

Notes

When the cycle is programmed, the system must be in state G40 (G41 or G42 tool radius offset cancelled).

The feed rate and its argument can be programmed in the cycle block, e.g.:

```
N.. G64 N.. N.. I.. K.. P.. G95 F0.25
```

Notes on definition of the finished profile

Both first and last blocks N.. to N.. defining the finished profile dimensions must include both coordinates X and Z as they may be analysed in reverse order.

Addresses I and K must be signed according the direction of the machining allowance (e.g. «I-..» for a bore).

The cycle will not execute any grooves (in the face or on diameter) included in the definition of the finished profile.

At the end of the cycle, the system is left in G0 mode.

The finished profile definition can be placed before the cycle call, but in this case it is necessary to use branch function G79 (see chapter. 5.15).

The following program variables cannot be used in the finished profile definition. Otherwise, the system returns error message 96:
- L100-L199
- L900-L959 (see chapter 7.1).

(The only case where these variables can be used is when M999 is programmed, see chapter. 5.19.9)

Notes on the blank definition

The blank definition cannot contain:
- PGP (Profile geometry Programming) blocks,
- a block defining a curve.

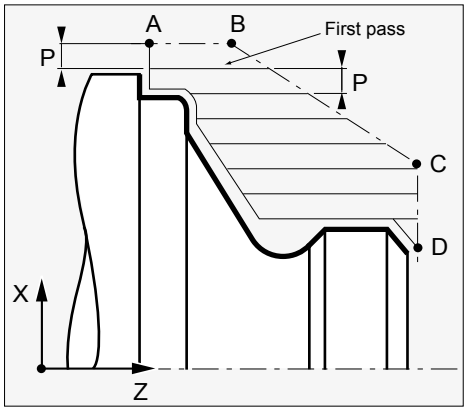
The blank definition blocks can include miscellaneous M functions.

5.12.4.1 Roughing cycle P or R

The roughing cycle uses P or R to determine turning or facing, plus the directional sense of increasing/decreasing dia. from the blank definition, to determine internal/external working.

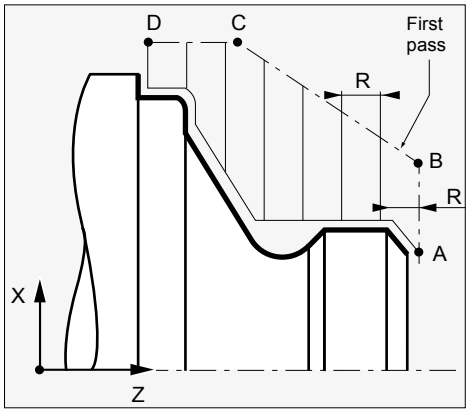
Rough turning

```
N200 ...
N210 G64 N.. N.. I.. K.. P4
N220 X.. Z..                  Point A
N230 Z..                      Point B
N240 X.. Z..                  Point C
N250 X..                      Point D
N260 G80 G52 X.. Z..
N..
```



Rough facing

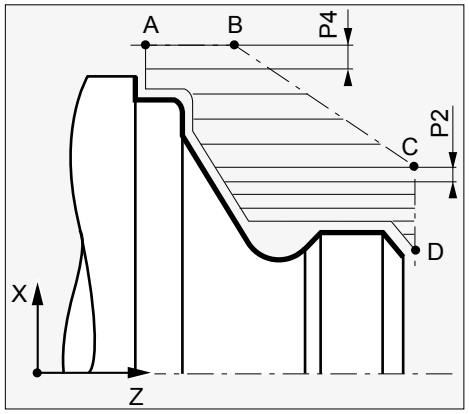
```
N200 ...
N210 G64 N.. N.. I.. K.. R4
N220 X.. Z..                  Point A
N230 X..                      Point B
N240 X.. Z..                  Point C
N250 Z..                      Point D
N260 G80 G52 X.. Z..
N..
```



The pass depth defined in the cycle block (P or R) can be modified during the blank definition blocks.

Example

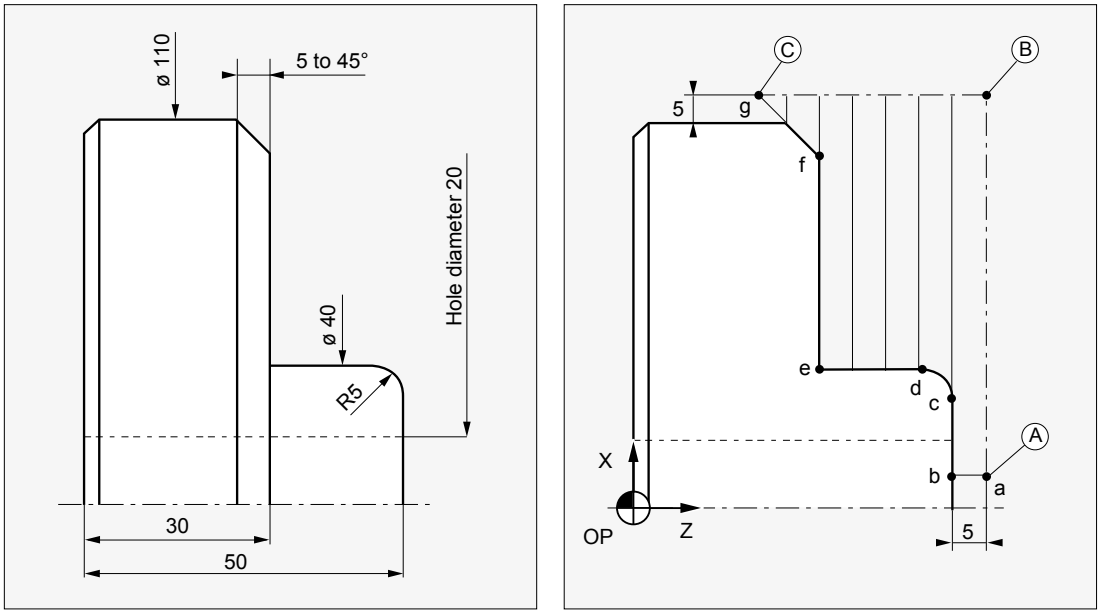
```
N200 ...
N210 G64 N.. N.. I.. K.. P4
N220 X.. Z..                  Point A
N230 Z..                      Point B
N240 X.. Z..                  Point C
N250 X.. P2                   Point D,
                              new P
N260 G80 G52 X.. Z..
N..
```



Examples

Roughing an external profile by facing

Machining paths



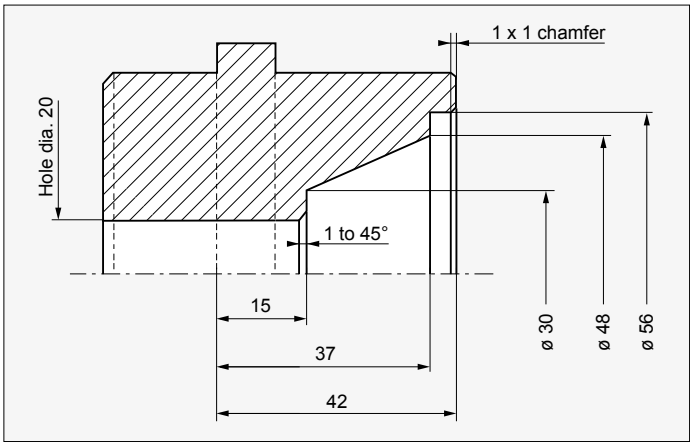

```
%46
N10 G00 G52 X.. Z..                          Tool change position
N20 T01 D01 M06 (FACING TOOL R=0.8)
N30 S900 M40 M04
N40 X120 Z55                                 Point B, tool at the start of roughing
N50 G92 S3500
N60 G96 S200
N70 G95 F0.2
N80 G64 N140 N210 I0.4 K0.2 R3               Cycle
N90 X16 Z55                                  Point (A)
N100 X120                                    Point (B)   } Blank definition
N110 Z20                                     Point (C)
N120 G80 G52 X.. Z.. G97 S900                Cycle cancellation
N130 T03 D03 M06 (COPYING TOOL R=0.4)
N140 G42 X16 Z55                             Point a, tool at the start of finishing
N150 G96 S250
N160 G01 Z50 F0.1                            Point b
N170 X30                                     Point c
N180 G03 X40 Z45 R5                          Point d
N190 G01 Z30                                 Point e
N200 X100                                    Point f
N210 X120 Z20                                Point g
N220 G00 G40 G52 X.. Z.. G97 S900 M05
N230 M02
```

Changes required to program %46 for turning.

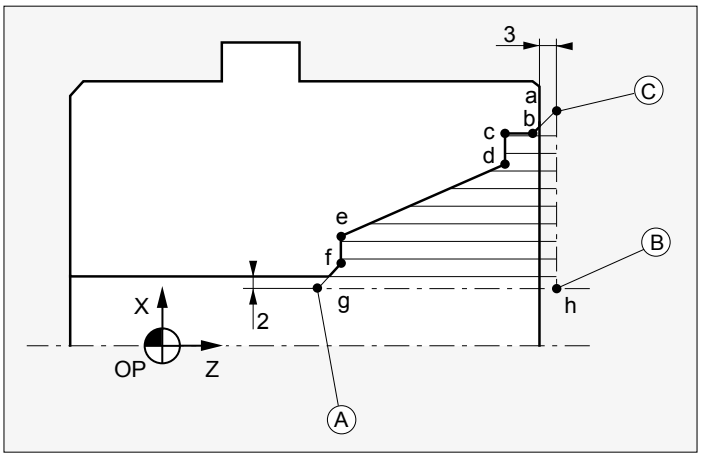
```
N80 G64 N140 N210 I0.4 K0.2 P3               Address R replaced by P

N90 X120 Z20                                 } Blank definition reversed
N100 Z55
N110 X16
```

Roughing of an internal profile by turning

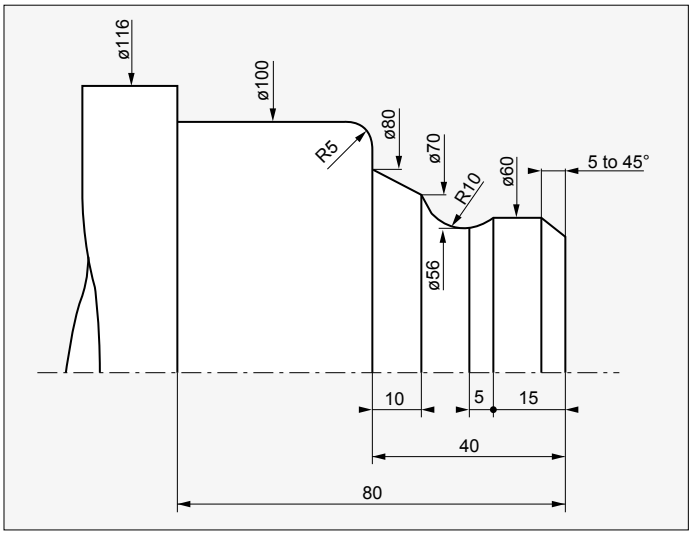


Machining paths

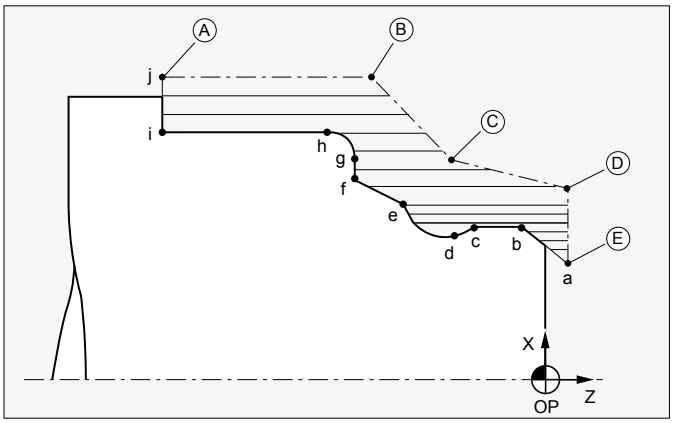



```
%48
N10 G00 G52 X.. Z..                          Tool change position
N20 T07 D07 M06 (BORING TOOL R=0.4)
N30 S900 M40 M04
N40 X18 Z45                                  Point B, tool at the start of roughing
N50 G92 S3500
N60 G96 S100
N70 G95 F0.15
N80 G64 N140 N210 I-0.2 K0.1 P2              Cycle
N90 X16 Z12                                  Point (A)
N100 Z45                                     Point (B)  } Blank definition
N110 X64                                     Point (C)
N120 G80 G52 X.. Z.. G97 S900                Cycle cancellation
N130 T09 D09 M06 (BORING TOOL R=0.2)
N140 G41 X64 Z45                             Point a, tool at the start of finishing
N150 G96 S120
N160 G01 X56 Z41 F0.08                       Point b
N170 Z37                                     Point c
N180 X48                                     Point d
N190 X30 Z15                                 Point e
N200 X22                                     Point f
N210 X16 Z12                                 Point g
N220 G00 Z45 G97 S900                        Point h
N230 G40 G52 X.. Z.. M05
N240 M02
```

Roughing of an external profile by turning (preformed blank profile)

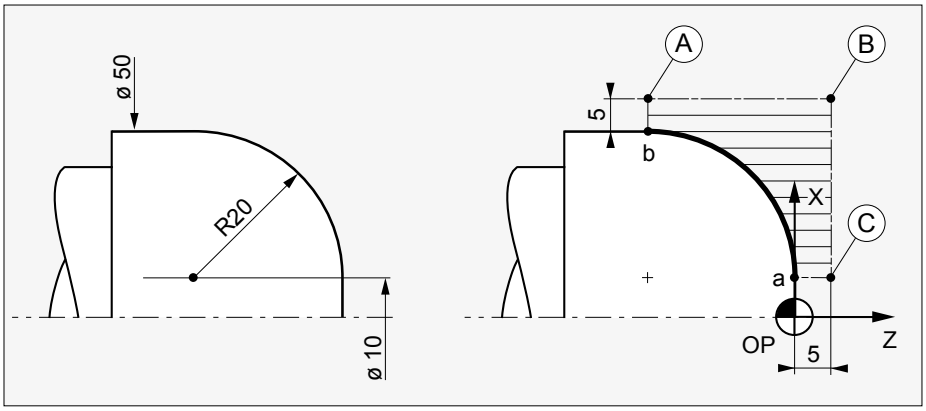


Machining paths




```
%62
N10 G00 G52 X.. Z..                                  Tool change position
N20 T01 D01 M06 (BIDIRECTIONAL TOOL R=0.8)
N30 S900 M40 M04
N40 X120 Z-30                                        Point B, tool at the start of roughing
N50 G92 S3500
N60 G96 S180
N70 G95 F0.25
N80 G64 N170 N270 I0.3 K0.2 P4                       Cycle
N90 X120 Z-80                                        Point (A)
N100 Z-40                                            Point (B)
N110 X80 Z-20                                        Point (C)   } Blank definition
N120 X70 Z5                                          Point (D)
N130 X40 P2                                          Point (E)
N140 G97 S900
N150 G80 G52 X.. Z..                                 Cycle cancellation
N160 T03 D03 M06 (COPYING TOOL R=0.4)
N170 G42 X40 Z5                                      Point a, tool at the start of finishing
N180 G96 S250
N190 G01 X60 Z-5 F0.1                                Point b
N200 Z-15                                            Point c
N210 X56 Z-20                                        Point d
N220 G02 X70 Z-30 R10                                Point e
N230 G01 X80 Z-40                                    Point f
N240 X90                                             Point g
N250 G03 X100 Z-45 R5                                Point H
N260 G01 Z-80                                        Point i
N270 X120                                            Point j
N280 G00 G40 G52 X.. Z.. G97 S900 M05
N230 M02
```

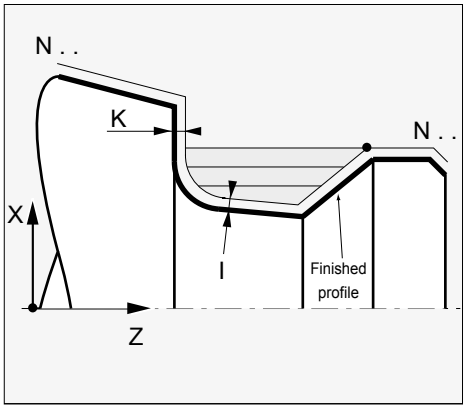
Roughing then finishing by turning of a profile defined by two points (a and b)




```
%64
N10 G00 G52 X.. Z..                    Tool change position
N20 T01 D01 M06 (TOOL R=0.8)
N30 S900 M40 M04
N40 X60 Z5                             Point B, tool at the start of roughing
N50 G96 S300
N60 G92 S3000
N70 G95 F0.3
N80 G64 N150 N160 P3 I0.2 K0.2         Cycle
N90 X60 Z-20                           Point A
N100 Z5                                Point B  } Blank definition
N110 X10                               Point C
N120 G80 G52 X.. Z.. G97 S900          Cycle cancellation
N130 T03 D03 M06 (OUTIL R=0.4)
N140 X10 Z2                            Approach point
N150 G42 G01 X10 Z0                    Point a
N160 G03 X50 Z-20 I10 K-20             Point b
N170 G01 G40 X54 G97 S900              Retraction point
N180 G00 G52 X.. Z.. M05
N190 M02
```

5.12.5 Groove roughing cycle ► G65

G65	Groove roughing cycle.



This function is used to rough a groove whose profile is programmed during definition of a finished profile.

The cycle executes axial grooves by turning and front grooves by facing.

This cycle can be executed with any spindle. However it generally requires constant surface speed which is only available on the main spindle of each channel.

Syntax

N.. **G65** [N.. N..] / X.. / Z.. [I.. K..] [EA..] P.. / R.. [Q..] [EF..]

G65	Groove roughing cycle.
N.. N..	First and last finished profile block numbers. The groove profile must be located between these blocks (maximum 50 blocks).
X.. / Z..	End of cut position in the appropriate axis: - X for front rough facing. - Z for axial rough turning.
I..	Finishing allowance in X: - The default value of I = 0.
K..	Finishing allowance in Z: - The default value of K = 0.
EA..	Cut introduction angle in the groove (see Fig. 1).
P.. / R..	Cut depth (unsigned): - P: depth of cut in X (turning). - R: width of cut in Z (facing).
Q..	Positioning clearance (see Fig. 2): - The default value of Q = 0 (approach distance without clearance at the machining feed rate).
EF..	Material penetration feed rate: - The default value of EF is the same as the modal feed rate F.

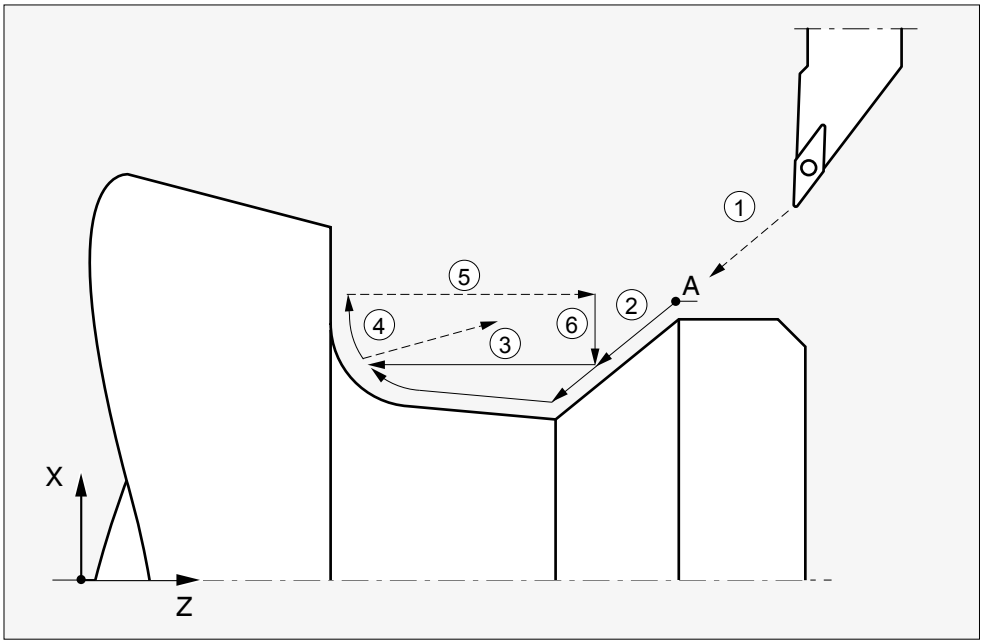
Property of the Function

Function G65 is nonmodal.

Cancellation

Function G65 is cancelled at the end of the block.

Cycle Steps



- Step 1: Tool positioning (point A, cycle start).
- Before the cycle start, the system takes into account the declared tool radius.
- Step 2: Infeed at the introduction angle at the machining feed rate (to depth P for turning).
- Step 3: Execution of the pass at the machining feed rate.
- Step 4: Retraction at the machining feed rate along the groove profile.
- Step 5: Rapid return to the perpendicular over the start point for the next pass.
- Step 6: Positioning at the machining feed rate (if Q with a value not programmed).

The following passes are executed in the same way as steps 2 to 6.

After execution of the last pass, the tool retracts at rapid to the cycle start point.

Notes on argument

Both the finished profile definition blocks N.. to N.. must include both the coordinates X and Z.

Tool positioning to the cycle start must be programmed in G40 (G41 or G42 radius offset cancelled).

The feed rate function and argument can be programmed in the cycle block, e.g.:

```
N.. G65 N.. N.. I.. K.. P.. G95 F0.1.
```

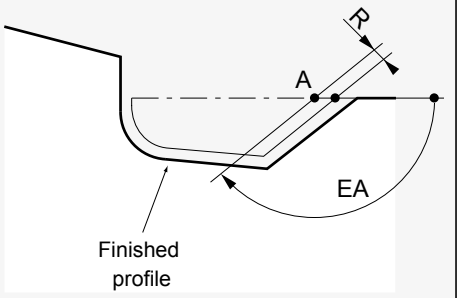
At the end of the cycle, the system is left in G0 mode.

Notes on argument EA

- EA: cut introduction angle defines a line from the cycle start point (point A) to the intersection with the finished profile.

After positioning to the start point, the system reads the declared tool radius (R) before executing the first pass.

FIGURE 1

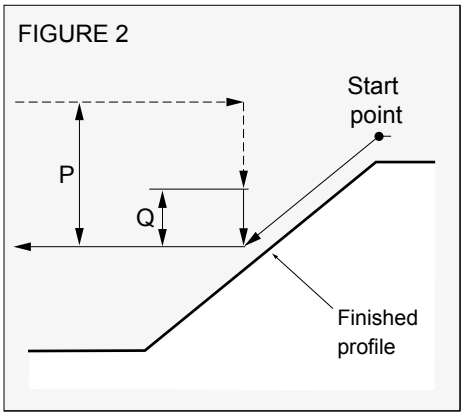


Notes on argument Q

After the first pass, when argument Q is programmed with its value, the approach distance before each new cut is carried out in two steps:

- At high speed up to value Q,
- At the machining feed rate until the start of the next pass.

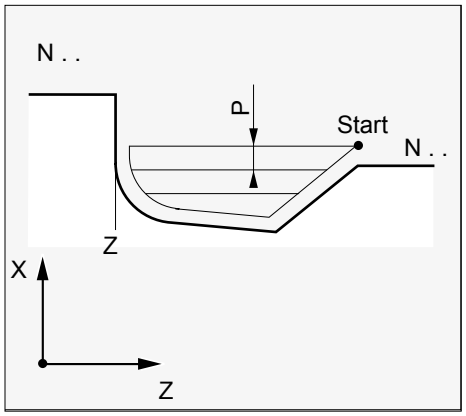
FIGURE 2



Cycle used for diameter or face machining

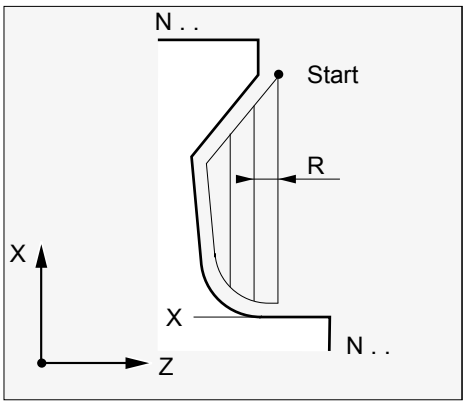
Diameter roughing

```
N70 ...
N80 G65 N.. N.. Z.. I.. K.. EA.. P..
N90 G52 X.. Z..
N..
```



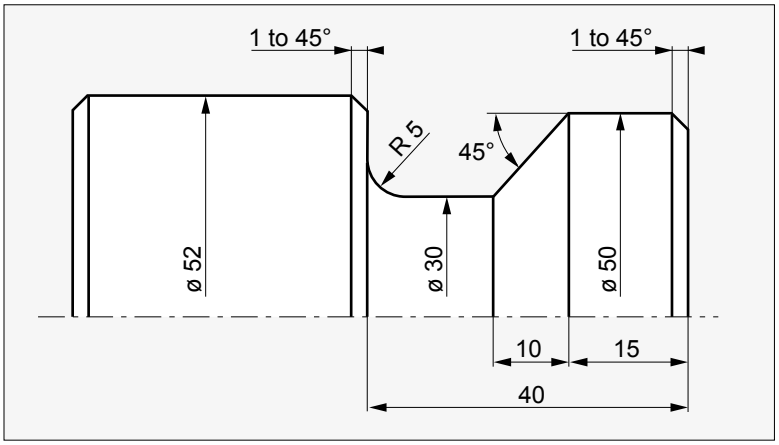
Face roughing

```
N70 ...
N80 G65 N.. N.. X.. I.. K.. EA.. R..
N90 G52 X.. Z..
N..
```

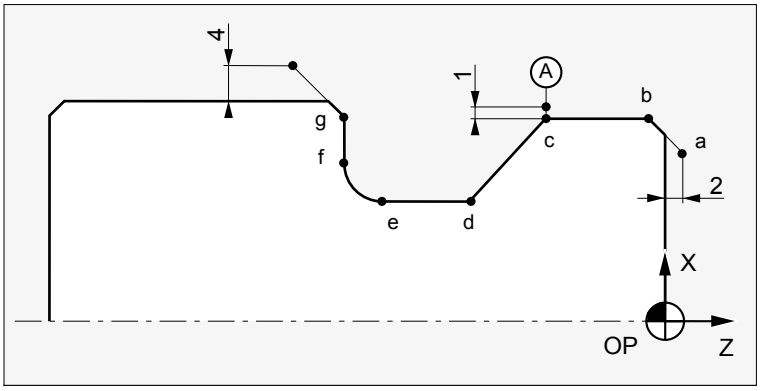


Examples

Roughing an axial groove



Machining paths

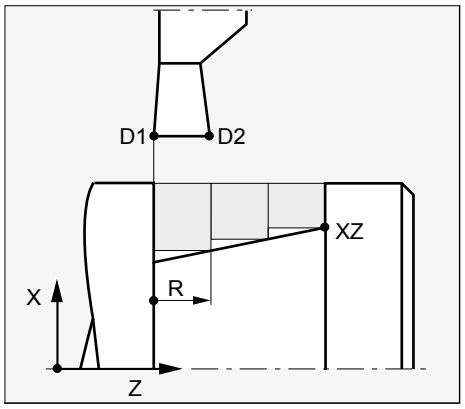



```
%34
N10 G00 G52 X.. Z..                                  Tool change position
N20 T03 D03 M06 (GROOVE ROUGHING TOOL R=0.4)
N30 S900 M03 M40
N40 X52 Z-15                                         Point A, tool at the start of the groove
N45 G92 S3000
N50 G96 S100
N60 G95 F0.2
N70 G65 N100 N180 Z-40 I0.2 K0.1 EA-135 P2
N80 G52 X..Z.. G97 S900
N90 T05 D05 M06 (SLIDING TOOL R=0.8)
N100 G42 X44 Z2                                      Point a, tool at the start of the profile
N110 G96 S250
N120 G01 X50 Z-1 F0.1                                Point b
N130 Z-15                                            Point c
N140 X30 Z-25                                        Point d
N150 Z-35                                            Point e
N160 G02 X40 Z-40 R5                                 Point f
N170 G01 X50                                         Point g
N180 X60 Z-45                                        Point h
N190 G00 G40 G52 X.. Z.. G97 S900 M05
N.. ...
```

5.12.6 Plunging cycle ► G66



G66	Plunging cycle.



This function is used to rough a dia. or frontal groove by consecutive penetrations.

This cycle can be executed with any spindle. However it generally requires constant surface speed which is only available on the main spindle of each channel.

Syntax

N.. **G66** D.. X.. Z.. [EA..] P../R.. [EP..] [EF..]

G66	Plunging cycle.
D..	Number of the second grooving tool correction (the first correction must be programmed in one of the previous blocks, see Fig. 1).
X.. / Z..	Groove machining end point (see Fig. 2).
EA..	Angle defining the slope across the groove bottom (see Fig. 2). - The default value of EA = 0 (no slope).
P.. / R..	Step between each penetration: - P..: value along X (frontal groove). - R..: value along Z (dia. groove).
EP..	Retraction (at 45 degrees) at the end of the pass (see example of special use of the cycle).
EF..	Dwell at the end of each penetration in seconds (Format F022): - The default value of EF = 0.

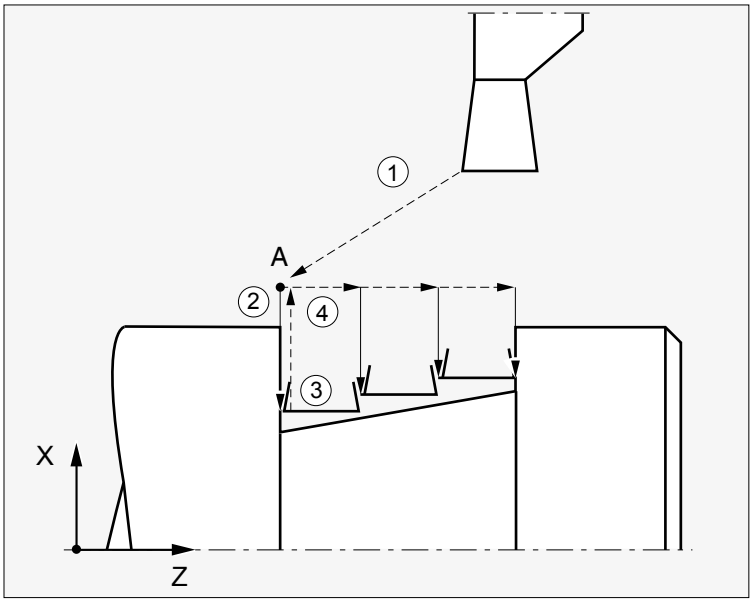
Property of the Function

Function G66 is nonmodal.

Cancellation

Function G66 is cancelled at the end of the block.

Cycle Steps



- Step 1: Tool positioning above the material (point A, cycle start).
  The first tool correction corresponding to the start side of the groove must be programmed in this block or in one of the previous blocks.
- Step 2: Penetration at the machining feed rate
  (to the groove bottom point possibly modified by angle EA).
- Step 3: Retraction at the machining feed rate.
- Step 4: Rapid movement by the value of the step for execution of the next penetration
  (taking into account the correction corresponding to the other side of the groove).

Execution of the following passes in the same way as steps 2 to 4 until reaching the end of the groove.

After execution of the last pass, retraction off of the material.

Notes

When the cycle is programmed, the system must be in state G40 (G41 or G42 tool radius offset cancelled).

The tool penetrations are uniformly distributed over the width of the groove. In certain cases, the system recomputes the programmed step.

The feed rate function and argument can be programmed in the cycle block, e.g.:

```
N.. G66 D.. X.. Z.. R.. EA.. EF.. G95 F0.1.
```

At the end of the cycle, the system is left in G0 mode.

Declaration of the two tool corrections

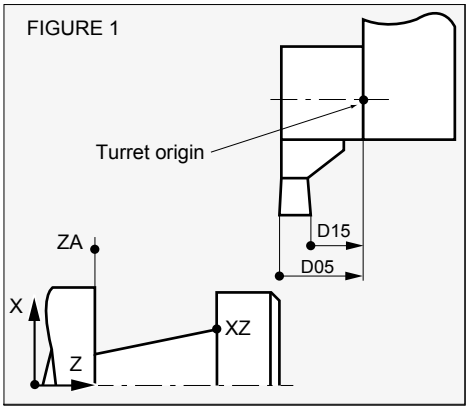
Example:

Correction D05: Groove start side along Z (point A).

Correction D15: Groove end side along Z (point XZ).

The difference in value between the two corrections must be equal to the tool width.

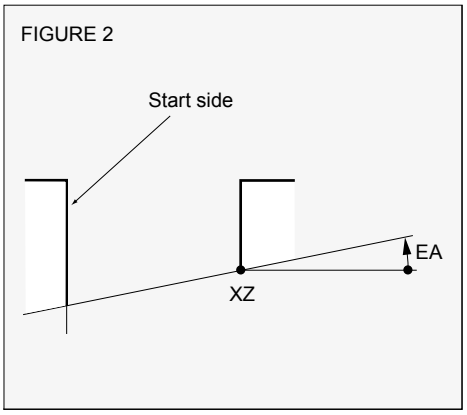
FIGURE 1



Notes on arguments XZ and EA

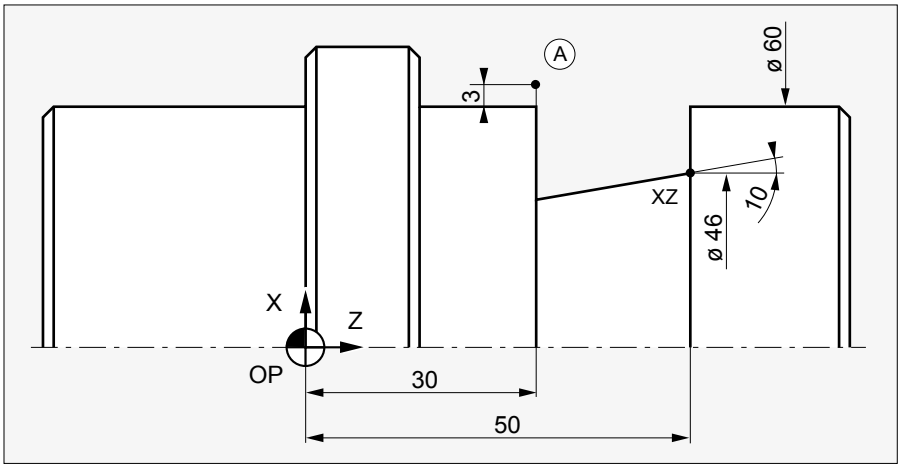
The groove bottom slope is defined from the groove end point (point XZ) to the intersection with the groove start side.

FIGURE 2



Examples

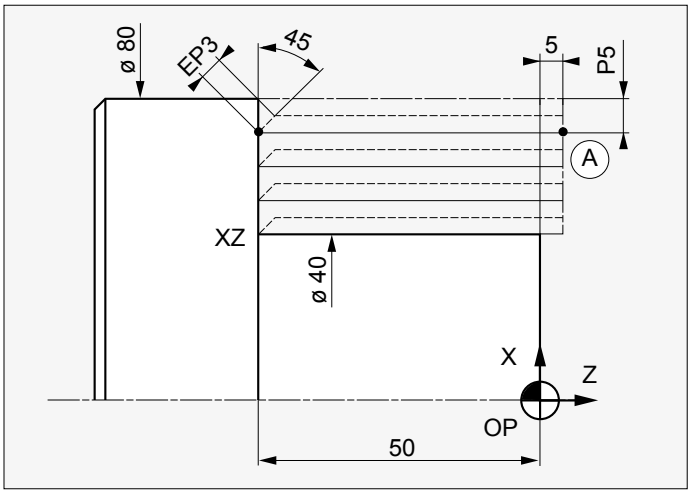
Plunging an axial groove




```
N.. ...
N210 G00 G52 X.. Z..                    Tool change position
N220 T05 D05 M06 (GROOVING TOOL L=6)
N230 G97 S900 M40 M03
N240 X66 Z30                            Point A
N250 G96 S80
N260 G95 F0.1
N270 G66 D15 X46 Z50 EA10 R5 EF0.5      Cycle
N280 G52 X.. Z.. G97 S900 M05
N..
```

Special use of cycle G66 with EP

EP can be added to the cycle syntax to rough a diameter.

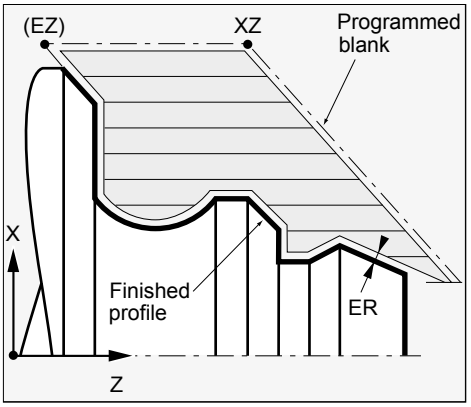



```
N.. ...
N100 G00 G52 X.. Z..                    Tool change position
$0 TURNING, DIAMETER 40
N110 T01 D01 M06 (BIDIRECTIONAL TOOL R=08)
N120 G97 S900 M40 M03
N130 X70 Z5                             Point A
N140 G96 S200
N150 G95 F0.3
N160 G66 X40 Z-50 P5 EP3                Cycle
N170 ...
```

5.12.7 Roughing cycle with groove ► G63



G63 Roughing cycle with.



This function is used for roughing a volume of material located between the definition of a blank and a finished profile.

The cycle will machine a groove whose profile is compatible with the geometry and radius of the tool used. It can be carried out by facing or turning and for external or internal machining.

This cycle can be executed with any spindle. However it generally requires constant surface speed which is only available on the main spindle of each channel.

Syntax

N.. **G63** [N.. N..] / X.. Z.. EX.. / EZ.. P.. / R.. EA.. / EU.. / EW.. [EB..] [EC..] [ER..] [Q..] [EQ..] [EF..]

G63	Roughing cycle with groove.
N.. N..	First and last block numbers defining the finished profile (maximum 95 blocks).
X../Z.. (or U..W..)	Cycle start point).
EZ.. / EX..	Pass end point on the roughing axis: - EZ for rough turning along Z. - EX for rough facing along X. Direction of roughing pass execution: - Z+ if EZ > Z Z- if EZ < Z. - X+ if EX > X X- if EX > Z.
P.. / R..	Pass depth: - P: value along X (rough turning). - R: value along Z (rough facing).
EA../EU../EW..	Roughing pass start positions (see Figs. 1 and 2). These arguments are programmed to define a start angle for cut application and can be combined: EA EU, EA EW or EU EW.
EB..	Maximum groove penetration angle. Angle used to take into account the groove profile geometry and the tool rake angle (α) (see Fig. 3): - By default, EB is parallel to the roughing axis.
EC..	Maximum cut exit angle. Angle used to take into account the groove profile geometry and the tool angle of attack (K) (see Fig. 4): - By default, EC is perpendicular to the roughing axis.
ER..	Finishing allowance. Value applied normal to the profile: - The default value of ER = 0.
Q..	Positioning clearance. Approach distance at feed rate before contact with the material (for rapid to next cut depth): - The default value of Q = 0 (no clearance).
EQ..	Minimum chip value: - Below the programmed value, the pass is not executed.
EF..	Penetration feed rate in the material: - The default value of EF is the same as the modal feed rate F.

Property of the Function

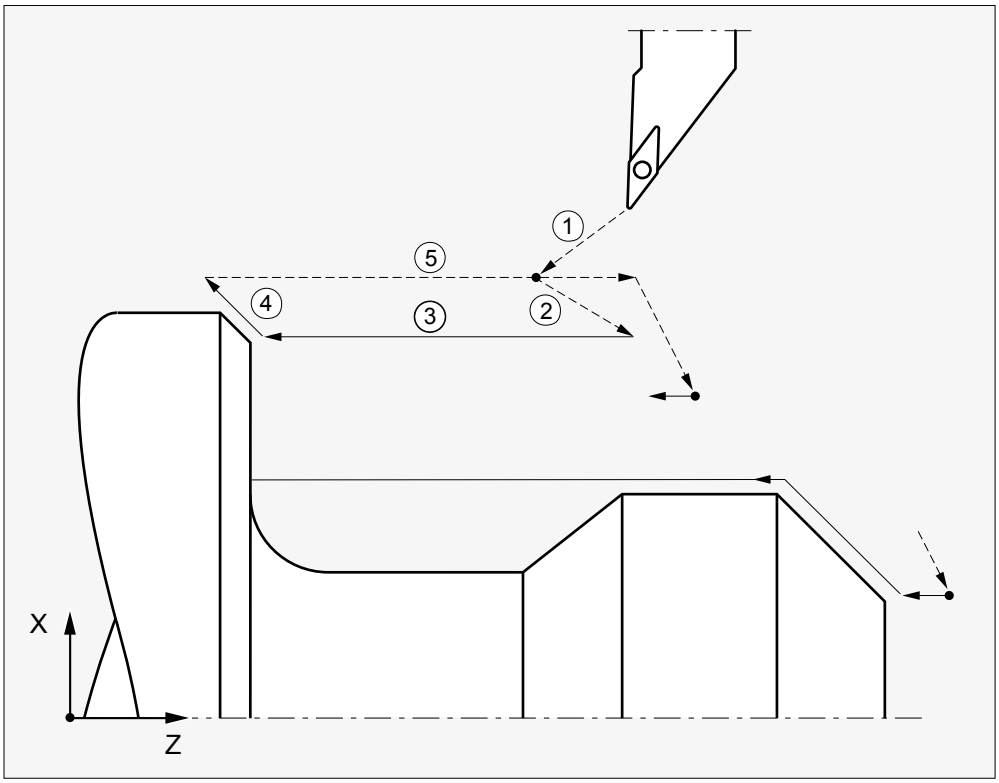
Function G63 is nonmodal.

Cancellation

Function G63 is cancelled at the end of the block.

Cycle Steps

Description of the first rough turning and grooving passes

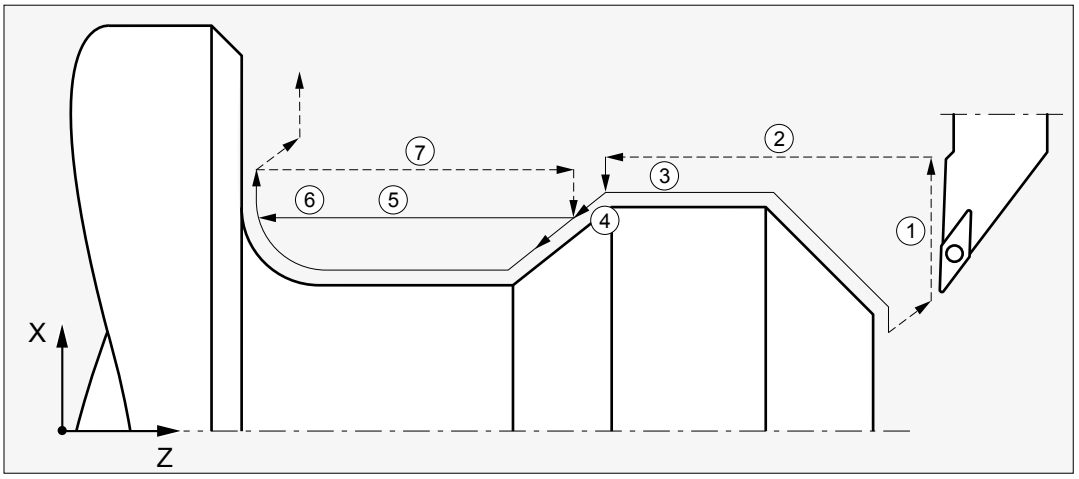


Rough turning pass

- Step 1: Tool positioning to X Z.
- Before the cycle begins, the system reads the declared tool radius.
- Step 2: Rapid to first cut depth P (for turning).
- Step 3: Execution of the first pass at the machining feed rate.
- Step 4: Lift along the profile at the machining feed rate.
- Step 5: Rapid return to the cycle start point.

The following passes are executed in the same way as steps 2 to 5.

After execution of the last roughing pass, the tool is retracted from the surface and positioned in X.



Grooving pass

- Step 1: Rapid positioning in X.
- Step 2: Rapid positioning in Z.
- Step 3: Positioning in X at the machining feed rate.
- Step 4: Cut application at the machining feed rate along the groove profile.
- Step 5: Execution of the pass at the machining feed rate.
- Step 6: Lift at the machining feed rate along the groove profile.
- Step 7: Rapid return to the cycle start point.

The following passes are executed in the same way as steps 3 to 6.

After execution of the last pass, the tool is retracted from the surface and positioned in X.

Notes

The number of blocks from N.. to N.. must not exceed 95. Otherwise, the system returns error message 92.

When the cycle is programmed, the system must be in state G40 (G41 or G42 tool radius offset cancelled).

At the end of the cycle, the system is left in G0 mode.

The finished profile definition can be programmed before the cycle call, but in this case it is necessary to use the branch function G79 (see chapter 5.14.3).

The cycle arguments defining positive angles are specified counter-clockwise.

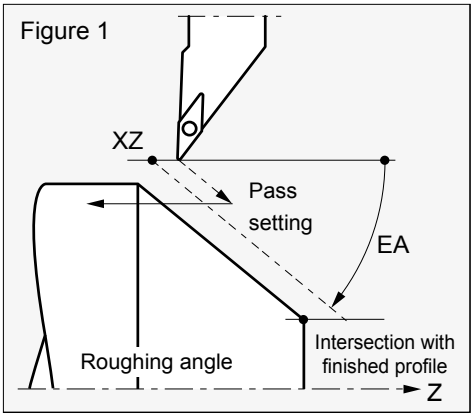
The feed rate function and argument can be programmed in the cycle block, e.g.:

```
N.. G63 N.. N.. X.. Z.. EZ.. P.. EA.. EB.. ER.. Q.. EQ.. EF.. G95 F0.3
```

Notes on arguments EA, EU and EW

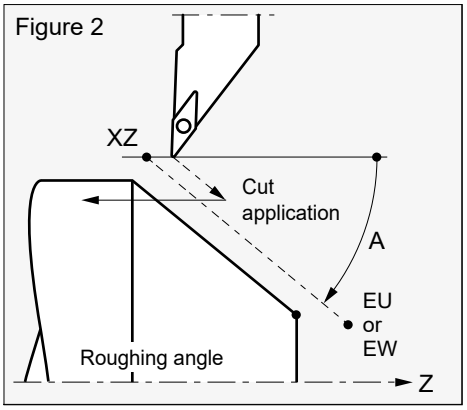
- EA: Angle limiting the start of cut application between point XZ (cycle start) and intersection with the finished profile.
  If there is no intersection, the last pass is limited to the first point of the finished profile).
  After positioning to point XZ, the system reads the declared tool radius before carrying out the first cut application.

Figure 1



- EU or EW: End point of the last roughing pass. If argument EA is not programmed, the values declared with EU and/or EW and XZ (cycle start) are used to define the cut application start angle (A).

Figure 2



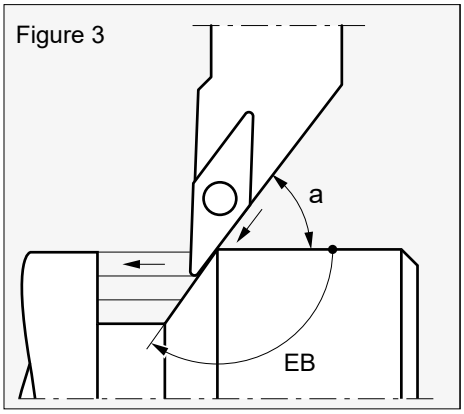
The angle is also used to differentiate between applying the cut at feed rate or rapid for a groove or not, respectively:

- Machining feed rate (in the material) if the angle between the cut application line and the roughing axis is $\geq$ 90 degrees.
- Traversing rate (off the material) if the angle between the cut application line and the roughing axis is < 90 degrees.

5.12.7.1 Definition of EB and EC

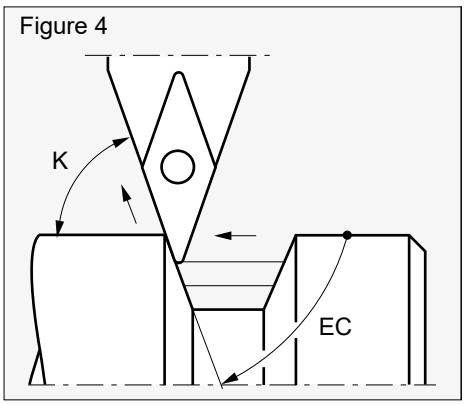
Angle defined by EB

Figure 3



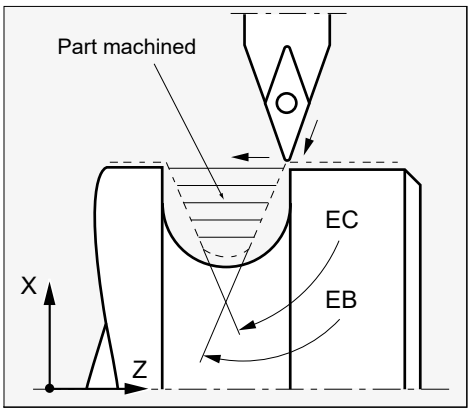
Angle defined by EC

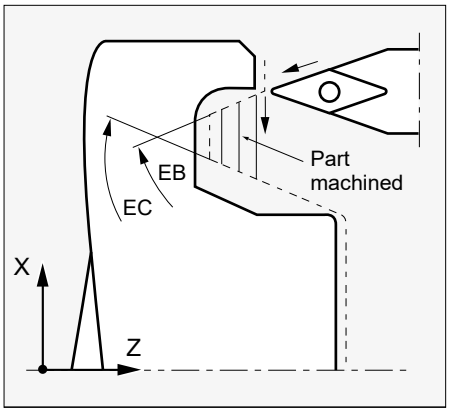
Figure 4



Machining of a groove limited by declaration of angles EB and EC

When the groove profile and tool geometry are completely incompatible, the groove is not executed.

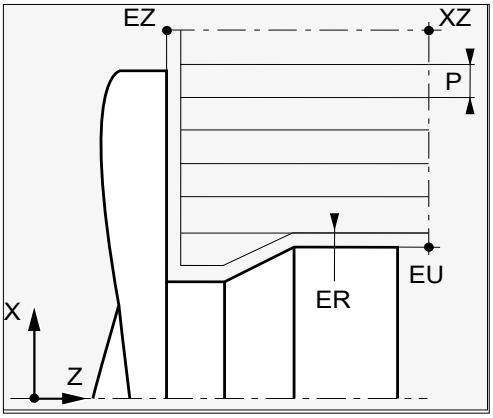




5.12.7.2 Cycle differences between turning and facing

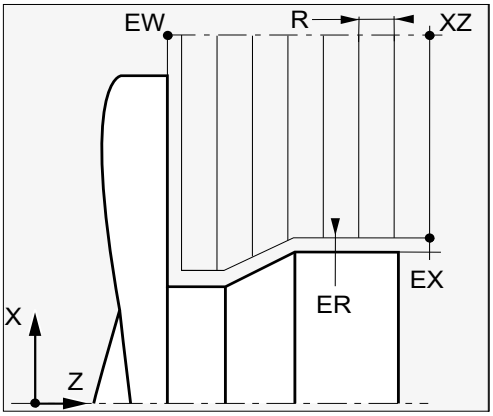
Rough turning

```
N70 ...
N80 G63 N.. N.. X.. Z.. EZ.. P.. EU.. EB.. ER..
N90 G52 X.. Z..
N..
```



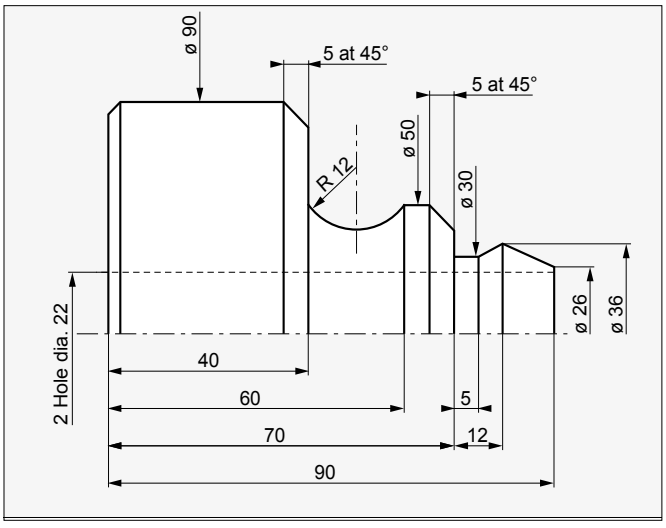
Rough facing

```
N70 ...
N80 G63 N.. N.. X.. Z.. EX.. R.. EW.. EC.. ER..
N90 G52 X.. Z..
N..
```

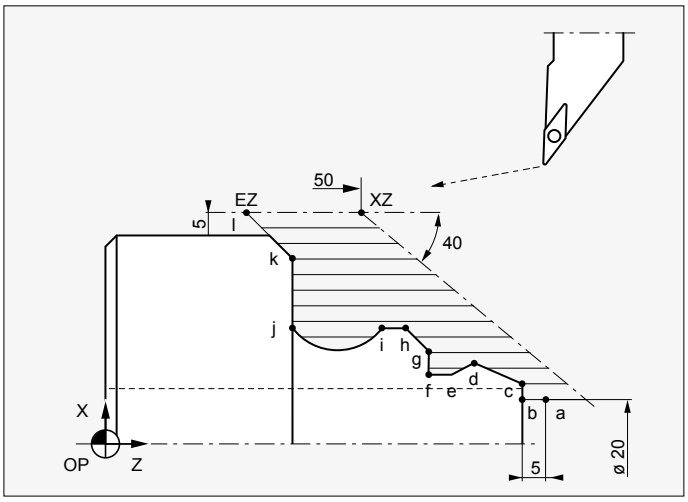


Examples

Rough and finish turning of an external profile

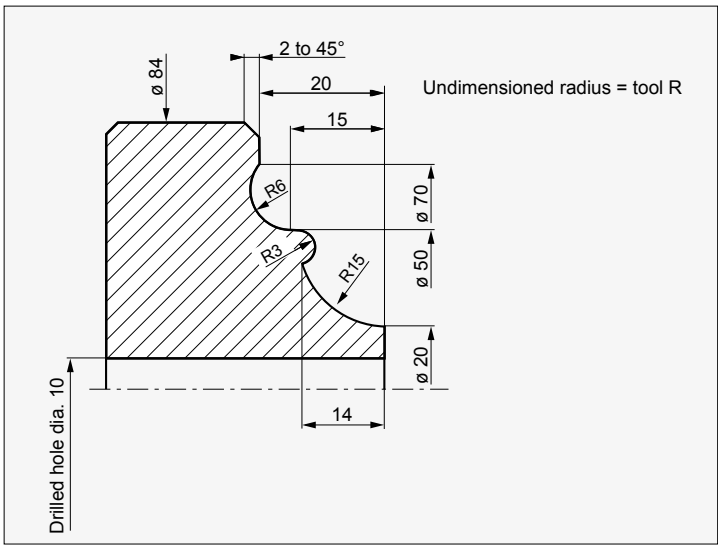


Machining paths

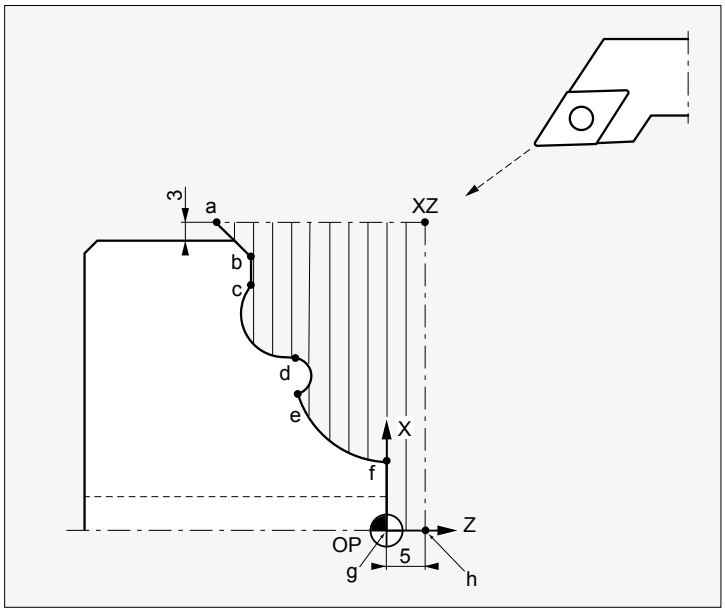



```
%32
N10 G00 G52 X.. Z..                          Tool change position
N20 T01 D01 M06 (SLIDING TOOL R=0.8)
N30 S900 M40 M04
N40 X100 Z50                                 Tool position at the start of roughing
N50 G92 S3500
N60 G96 S200
N70 G95 F0.25
N80 G63 N110 N230 X100 Z50 EZ30 P2 EA-40 EB-145 ER0.4 Q2 EQ1 EF0.1
N90 G52 X.. Z..
N100 T03 D03 M06 (COPYING TOOL R=0.4)
N110 G42 X20 Z95                             Point a, tool at the start of finishing
N120 G96 S250
N130 G01 Z90 F0.4                            Point b
N140 X26 F0.1                                Point c
N150 X36 Z82                                 Point d
N160 X30 Z75                                 Point e
N170 Z70                                     Point f
N180 X40                                     Point g
N190 X50 Z65                                 Point h
N200 Z60                                     Point i
N210 G02 X50 Z40 R12                         Point j
N220 G01 X80                                 Point k
N230 X100 Z30                                Point l
N240 G40 G52 X.. Z.. G97 S900 M05
N250 M02
```

Rough and finish facing of an external profile



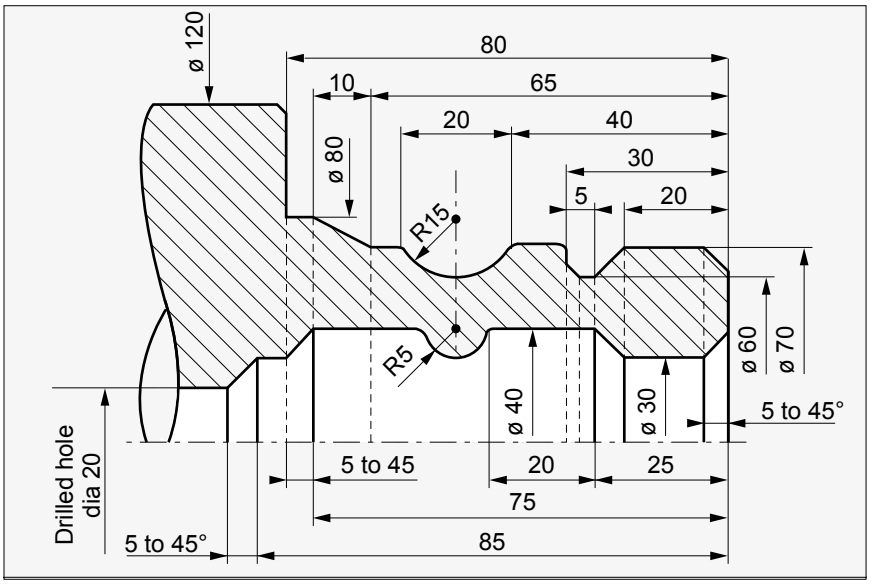
Machining paths



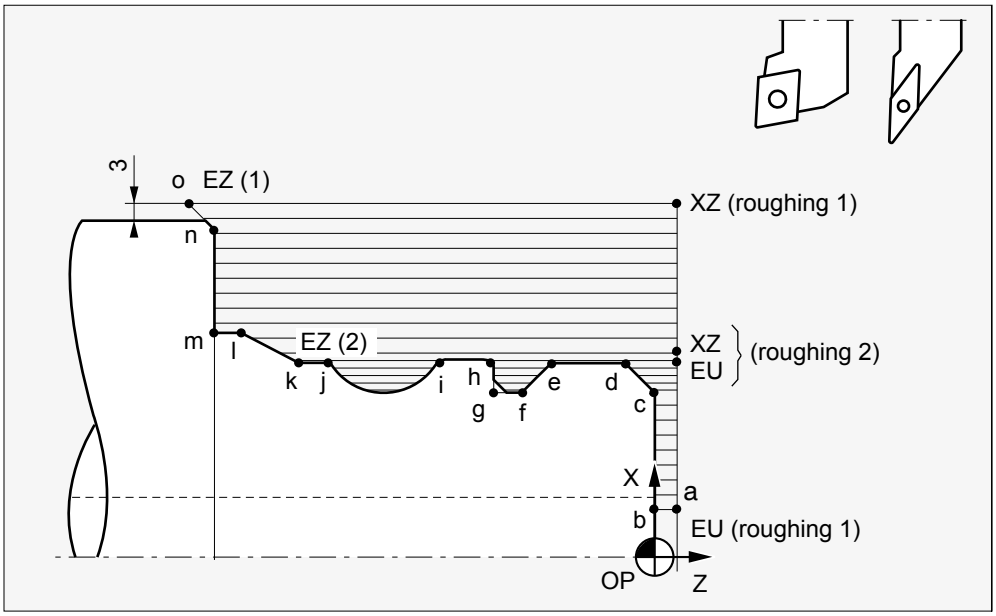

```
%39
N10 G00 G52 X.. Z..                          Tool change position
N20 T07 D07 M06 (FACING TOOL =0.8)
N30 S900 M40 M04
N40 X100 Z10                                 Tool position at the start of roughing
N50 G92 S3500
N60 G96 S200
N70 G95 F0.2
N80 G63 N110 N190 X90 Z5 EX0 R2 EA-180 EB-150 EC-170 ER0.2 Q1 EF0.1
N90 G52 X.. Z..
N100 T09 D09 M06 (COPYING TOOL R=0.4)
N110 G41 X90 Z-25                            Point a, tool at the start of finishing
N120 G96 S250
N130 G01 X80 Z-20 F0.1                       Point b
N140 X70                                     Point c
N150 G03 X50 Z-15 R6                         Point d
N160 G02 X40 Z-14 R3                         Point e
N170 G03 X20 Z0 R15                          Point f
N180 G01 X0                                  Point g
N190 G00 Z5                                  Point h
N200 G40 G52 X.. Z.. G97 S900 M05
N210 ...
N.. M02
```

Execution of a profile by two outside rough turnings then finishing

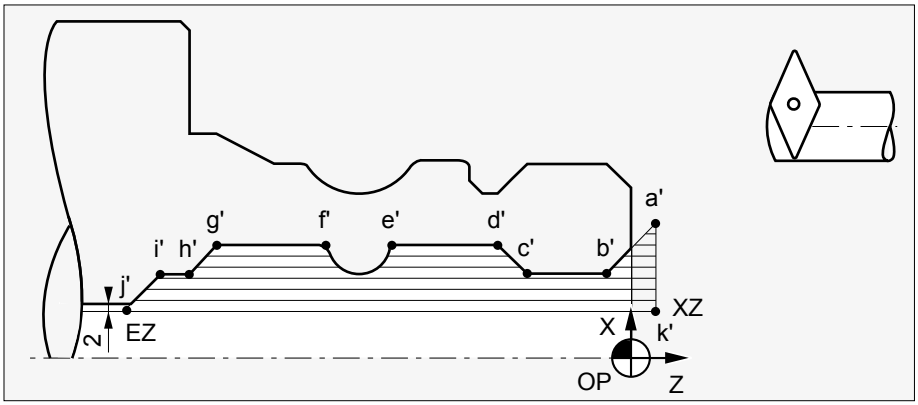
Rough and finish turning an internal profile



External machining paths



Internal machining paths




```
%163
N10 G0 G52 X.. Z..                          Tool change position
$0 EXTERNAL ROUGHING 1
N20 T01 D01 M06 (TOOL R0.8)
N30 S900 M40 M04
N40 X126 Z5                                 Tool position at the start of roughing 1
N50 G92 S4000
N60 G96 S260
N70 G95 F0.3
N80 G63 N150 N300 X126 Z5 P3 EZ-85 EU16 EB-170 EC-93 ER0.2 EQ1 Q1 EF0.2
N90 G0 G52 X.. Z.. G97 S900
$0 EXTERNAL ROUGHING 2
N100 T03 D03 M06 (TOOL R0.8)
N110 X80 Z5                                 Tool position at the start of roughing 2
N120 G63 N150 N300 X72 Z5 P1.5 EZ-65 EU70 EB-125 EC-93 ER0.2 Q1 EF0.15
N130 G0 G52 X.. Z.. G97 S900
$0 EXTERNAL FINISHING
N140 T11 D11 M06 (COPYING TOOL R0.8)
N150 G42 G0 X16 Z5                          Point a
N160 G96 S300
N170 G01 Z0 F0.08                           Point b
N180 X60                                    Point c
N190 X70 Z-5                                Point d
N200 Z-20                                   Point e
N210 X60 Z-25                               Point f
N220 Z-30 EB-2                              Point g
N230 X70 EB2                                Point h
N240 Z-40 EB2                               Point i
N250 G02 X70 Z-60 R15 EB2                   Point j
N260 G01 Z-65                               Point k
N270 X80 Z-75                               Point l
N280 Z-80                                   Point m
N290 X116                                   Point n
N300 X126 Z-85                              Point o
N310 G0 G40 G52 X.. Z.. G97 S900 M05
$0 INTERNAL ROUGHING
N320 T05 D05 M06 (BORING TOOL R0.8)
N330 S900 M40 M03
N340 X10 Z10                                Tool position at the start of roughing
N350 G92 S3000
N360 G96 S70
N370 G95 F0.1
N380 G63 N420 N520 X16 Z5 EU50 EZ-92 P2 EB110 EC70 ER0.2 Q1 EQ0.5 EF0.1
N390 G0 Z5                                  Tool retraction
```

```
N400 G0 G52 X.. Z..
$0 INTERNAL FINISHING
N410 T07 D07 M06 (BORING TOOL R0.4)
N420 G41 X50 Z5                      Point a'
N430 G96 S90
N440 G01 X30 Z-5 F0.07               Point b'
N450 Z-20                            Point c'
N460 X40 Z-25                        Point d'
N470 Z-45 EB2                        Point e'
N480 G02 X40 Z-55 I40 K-50 EB2       Point f'
N490 G01 Z-75                        Point g'
N500 X30 Z-80                        Point h'
N510 Z-85                            Point i'
N520 X16 Z-92                        Point j'
N530 G0 Z5                           Point k', Tool retraction
N540 G40 G52 X.. Z.. G97 S900 M05
N550 M02
```

5.13 Jumps and branches

5.13.1 Restrictions related to drip feed mode

Drip feed mode allows a program to be checked or to be executed while being simultaneously transmitted via the machine panel or the network.

Drip feed mode is used for execution of large programs which cannot be stored in full in the NC.

Restrictions related to drip feed mode

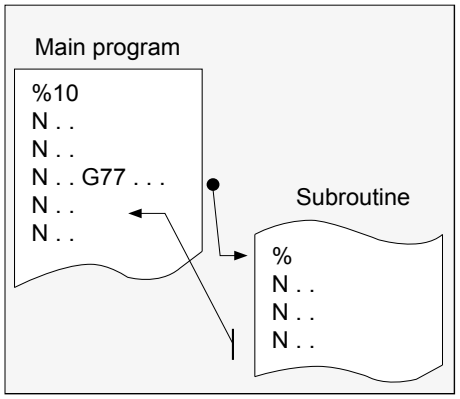
It is impossible to use functions referring to block numbers like :
- jump to a sequence (G79 N..),
- block interrupt (G10 N..),
- branch to sequences of the main program (N.. N..),
- activation of an emergency retract subroutine (G75 N..).

However calls to subroutines **stored in memory** are allowed in drip feed mode (G77 H.. or G77 H.. N.. N..) Additionally Structured programming is not allowed as it requires advanced reading of the program whose required blocks might not be in the buffer yet.

5.13.2 Unconditional branch to a subroutine or sequence blocks with return ► G77



G77	Unconditional branch to a subroutine or sequence of blocks with return.



Subroutines included in the main program or separate from it are called by addresses H.. and/or N.. N.. associated with the function.

Syntax

N.. **G77** [H..] [N.. N../N..] [S..]

G77	Unconditional branch to a subroutine or sequence of blocks with return (maximum 16 nesting levels).
H..	Number of a subroutine external to the main program.
N.. N../N..	Number of the first and last block called (If both N.. have the same number or if only one block is programmed, only one block is called).
S..	Number of times the subroutine or sequence of blocks is to be repeated (default = 1, maximum 1000000 repetitions).

Property of the Function

Function G77 is nonmodal.

Cancellation

Function G77 is cancelled at the end of the block.

Notes

A subroutine called by addresses N.. N.. can be placed after M02.

If argument S is programmed in a block containing other instructions, it must immediately follow the code G77.

Even if the subroutine call is defined by two sequence numbers in reverse order (e.g. G77 N200 N10), the system will executes the program in the normal sequence from N10 to N200 and no error is displayed.

Context

The calling program context can be saved at the beginning of the called program.

It is then restored at the end of execution of the called program. The programmed status access symbols are used for save and restore.

Inhibiting display of subroutines being executed

Display of the programming context of a subroutine and its internal subroutines during execution can be inhibited. Placing the character " : " after the subroutine number inhibits display. Only the subroutine call block is displayed.

Example

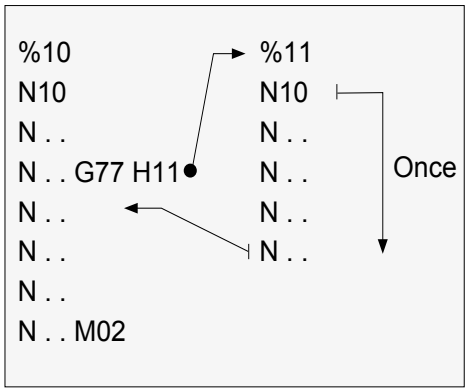
Main program %10 calling subroutine %110: itself containing an internal subroutine %210. Only block N50 of program %10 is displayed during execution of subroutines %110 and %210.

```
%10                  %110:                %210
N10                  N10                  N10
N..                  N..                  N..
N50 G77 H110         N80 G77 H210         N..
N..                  N..                  N..
```

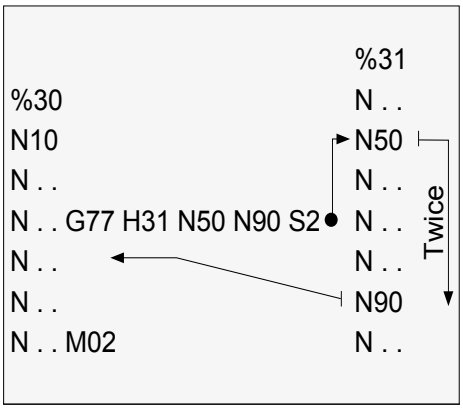
Examples

Call to external subroutines from the main program

Branch to subroutine %11 from main program %10.

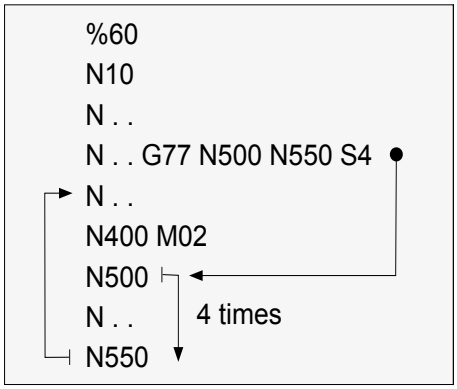


Call to blocks N50 to N90 of subroutine %31 from main program %30 twice.



Call to sequences in the program

Branch to blocks N500-N550 located downstream in program %60, after M02 four times.

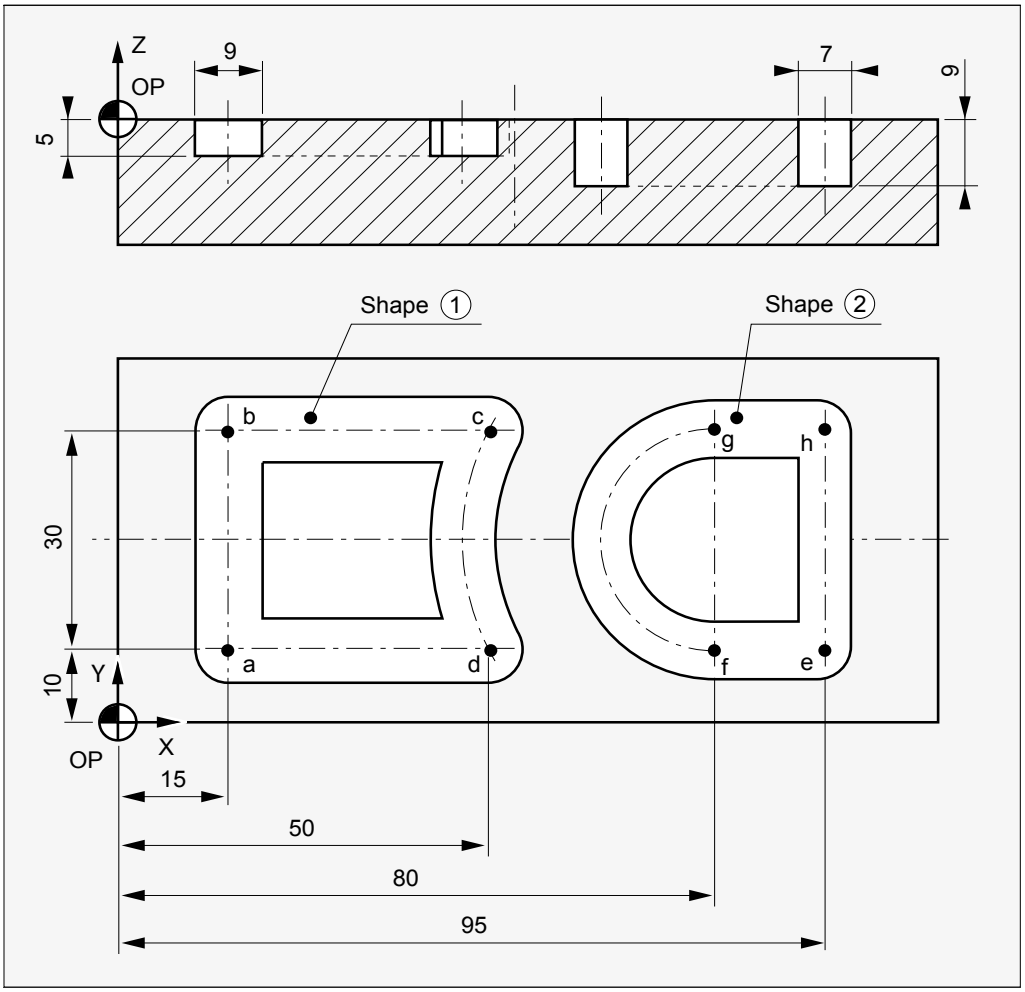




Execution of two grooves by subroutine calls

Shape 1, subroutine H35: two branches for execution of two passes each with depth 2.5 mm.

Shape 2, subroutine H40: three branches for execution of three passes each with depth 3 mm.



Main program and Subroutine for shape 1 and 2

```
%30
N10 G00 G52 Z.. M05 M09
$0 2 SHAPES WITH S/PROG %35 ET %40
N20 T01 D01 M06 (CUTTER DIAMETER=9)
N30 S800 M40 M03
N40 G00 X15 Y10 Z3                    Point a, approach
N50 G01 Z0 F50                        Approach on Z
$0 MACHINING SUBROUTINE %35
N60 G77 H35 S2                        Branch to subroutine %35
N70 G77 N10                           Branch to sequence N10
N80 T02 D02 M06 (CUTTER DIAMETER=7)
N90 S1000 M40 M03
N100 G00 X95 Y10 Z3                   Point e, approach
N110 G01 Z0 F50                       Approach on Z
$0 MACHINING SUBROUTINE %40
N120 G77 H40 S3                       Branch to subroutine %40
N130 G77 N10                          Branch to sequence N10
N140 M02
```

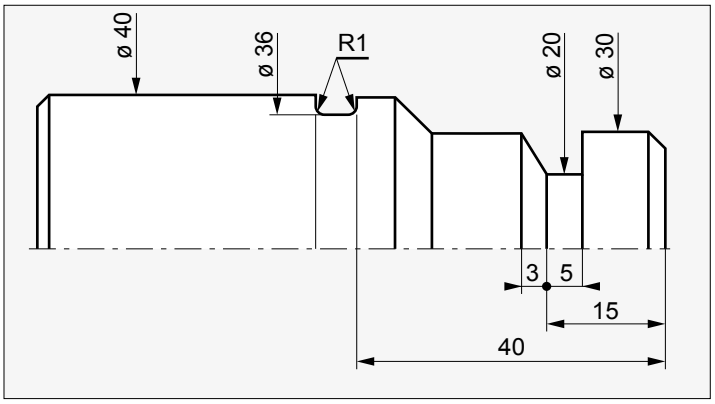
Subroutine for shape 1

```
%35
$0 SUBROUTINE OF MAIN PROGRAM %30
N10 G91 G01 Z-2.5 F40 M08          Point a, incremental infeed
N20 G90 Y40 F80                    Point b
N30 X50                            Point c
N40 G03 X50 Y10 I80 J25            Point d
N50 G01 X15                        Point a
```

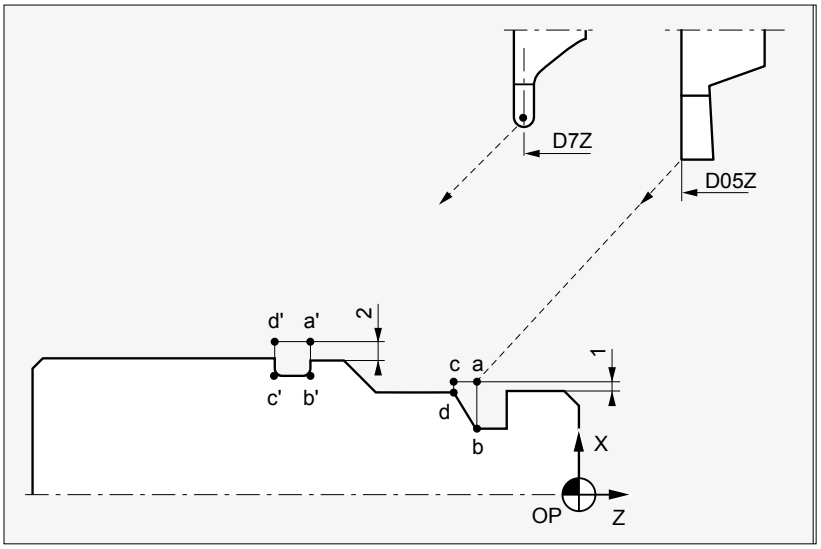
Subroutine for shape 2

```
%40
$0 SUBROUTINE OF MAIN PROGRAM %30
N10 G91 G01 Z-3 F40 M08            Point e, incremental infeed
N20 G90 X80 F70                    Point f
N30 G02 X80 Y40 R15                Point g
N40 G01 X95                        Point h
N50 Y10                            Point e
```

Cutting grooves by calling two subroutines



Machining paths



Main program

```
%30
N10 G0 G52 X0 Z0
$0 EXTERNAL MACHINING
N..
N..
N110 G0 G40 G52 X.. Z.. G97 S900
$0 VEE-GROOVE
N120 T05 D05 M06 (TOOL WIDTH 5)
N130 X32 Z-15                      Point a, approach
N140 G96 S80
N150 G77 H35                       Call to subroutine %35
N160 G77 N110                      Call to sequence N110
$0 GROOVE WITH RADIUS
N170 T07 D07 M06 (TOOL RADIUS 1)
N180 G0 G42 X44 Z-40               Point a', approach
N190 G96 S50
N200 G77 H40                       Call to subroutine %40
N210 G77 N110 M05                  Call to sequence N110
N220 M02
```

Subroutine 1

```
%35
N10 G91 G01 X-6 G95 F0.08          Point b, incremental dimensions
N20 X6                             Out-feed to point a
N30 G0 Z-3                         Point c
N40 G01 X-1 F0.3                   Point d
N50 X-5 Z3 F0.05                   Point b
N60 X6 F0.08                       Point a
N70 G90
```

Subroutine 2

```
%40
N10 G91 G01 X-4 G95 F0.06          Point b', incremental dimensions
N20 Z-5 F0.08                      Point c'
N30 X4 F0.3                        Point d'
N40 G90
```

5.13.2.1 Call to subroutine return block ► G77 -i



G77 -i	Call to subroutine return block.

This function allows a subroutine to call and execute instructions held in the block following the block calling the subroutine. Execution of the subroutines resumes after this call.

Syntax

N.. **G77 -** i

G77	Call to subroutine return block.
-i	Immediate value (or variable) giving the relative nesting level of the program to return to.

Properties of the Function

Function G77 -i is nonmodal.

Cancellation

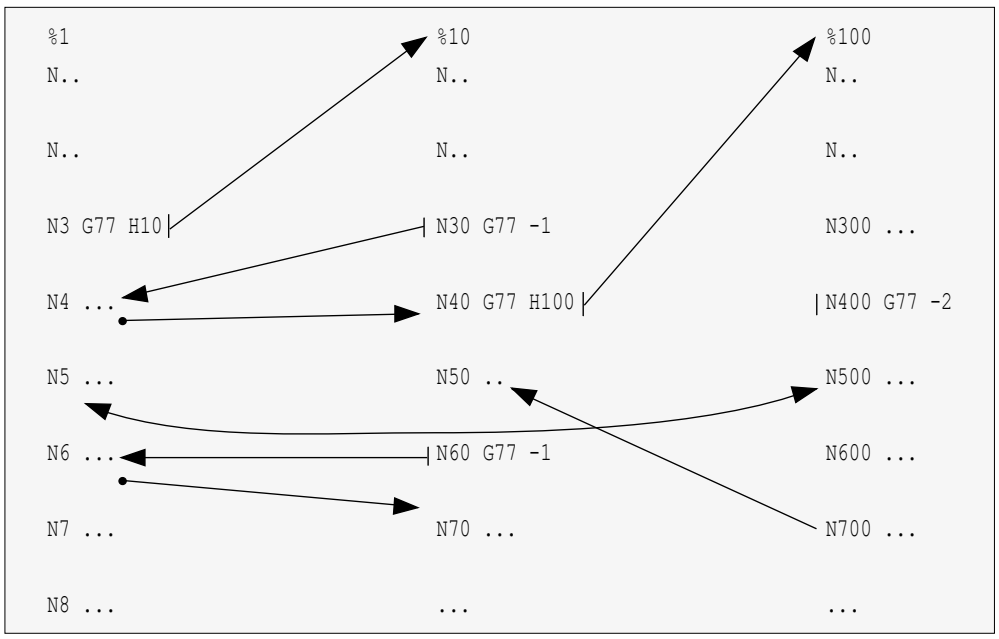
Function G77 -i is cancelled at the end of the block.

Notes

If the block called is in the «displayable» program, it is displayed on the program page (PROG) even if the subroutine that called it is not displayable.

It should be noted that:
- if the nesting level requested is not accessible, the system returns error message 26
- the blocks called by G77 -i must not include branches or other calls (G79, G77, etc.) or the system returns error message 26.

Example

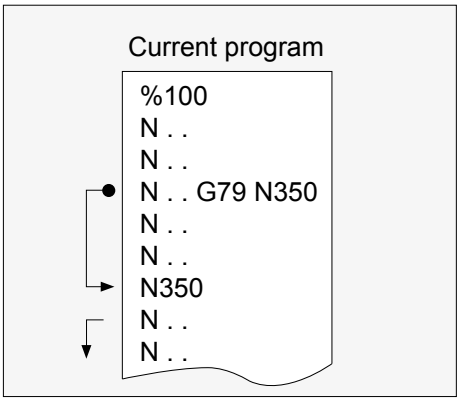


5.13.3 Branch to a sequence without return ► G79



G79	Conditional or unconditional branch to a sequence without return.

A conditional or unconditional branch is made to sequence number N.. associated with the function.



Syntax

N.. **G79** [L../E.. > = < Number] N..

G79	Conditional or unconditional branch to a sequence (an upstream or downstream branch is allowed).
L.. or E..	Variable L or parameter E tested in the condition (see chapter. 7.1 and 7.2).
> = <	Comparison symbols for the condition (up to two symbols can be used).
Number	Numerical expression of the condition.
N..	Mandatory argument defining the sequence number to which the jump is to be made.

Property of the Function

Function G79 is nonmodal.

Cancellation

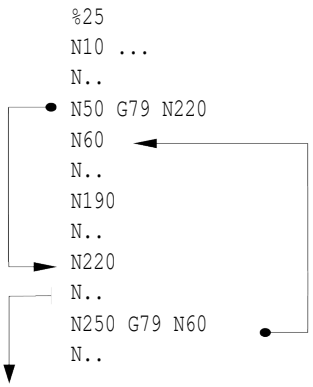
Function G79 is cancelled at the end of the block.

Notes

If the jump is conditional, the condition must be located between G79 and N.

Examples

Unconditional jump



```
%25
N10 ...
N..
N50 G79 N220
N60
N..
N190
N..
N220
N..
N250 G79 N60
N..
```

Jump to sequence N220

Jump to sequence N60

Conditional branch

Counting the number of machining operations performed and branching when the specified number is reached.

```
%75
N10 L5 = 20
N20 G52 Z..
N30 T01 D01 (TOOL)
N40 S800 M40 M03
N50 G00 X.. Y.. Z..
N60 G01 Z..
N70
N..
N..
N..
N140
N150 L5 = L5 -1
N160 G79 L5 = 0 N180
N170 G79 N50
N180 G52 Z.. M05
N190 M02
```

Variable L5 set to 20

Programming for one machining operation

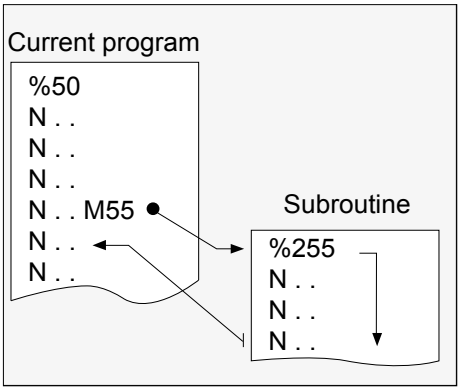
Count each operation down
Condition: If L5=0, branch to block N180
Branch to N50

5.13.4 Subroutine call by M function ► M..



M..	Subroutine branch by M function.

The M.. function branches to a subroutine whose number is assigned by the machine manufacturer (see Machine operation manual).



5.13.4.1 Subroutine call by M function

The machine builder can define up to 32 M functions that will each execute a particular subroutine.

The function is not different than a call by G77 EXCEPT that it is not possible to nest subroutines called by M function or by PLC.

Syntax

N.. **M..**

M..	Subroutine branch by M function.

Property of the Function

Function M.. programmed is modal however the subroutine is only executed once.

Notes

A subroutine called by an M function cannot itself call another subroutine by an M function. However, it can be nested with another type of call (by G function or PLC), but in no case may two calls of the same type be nested. Error 253 will be generated in case of nesting such subroutines.

A block including a subroutine call by M function may also include a positioning command.

Example: N.. M175 G00 X100

In this case, the block is processed in the following order:
- M175 sent to the PLC
- Movement X100 executed
- Call to the subroutine defined in machine parameter P35.

Return to current program

After execution of the subroutine, none of the data programmed earlier are restored.

The following data must be reprogrammed if necessary:
- modal preparatory G functions,
- modal technological S functions and miscellaneous M functions,
- correction D even if the tool was not changed,
- program variables L.

Context

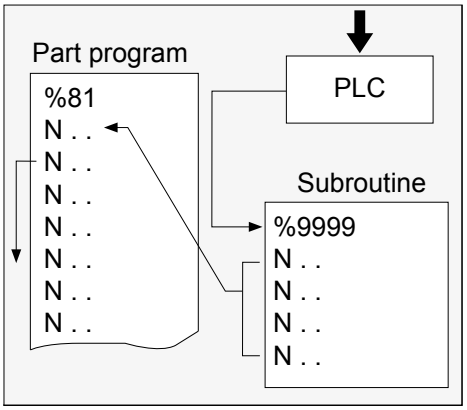
The calling program context can be saved at the beginning of the called program. It is then restored at the end of execution of the called program. The programmed status access symbols are used for save and restore.

5.13.5 Subroutine call by PLC



During execution of a part program, the PLC can generate a call to a specific subroutine. The subroutine number is always %9999. If it exists, this functionality has been implemented by the machine builder.

For further information, please refer to the machine operation manual.



Condition for the branch to subroutine %9999

Part program being executed in mode:

- automatic (AUTO),
- single step (SINGLE),
- rapid (RAPID).

A subroutine called by the PLC cannot itself call another subroutine by PLC. However, it can be nested with another type of call (by M function or G function), but in no case may two calls of the same type be nested. If a part program is already being executed, the branch to subroutine %9999 is only made at the end of an interruptible block or the end of execution of an uninterruptible block.

Definition

An uninterruptible block is a block containing a G function causing a branch to a subroutine (G31, G45, G81, etc.) or a block for which certain parameters must be known in order to execute the next block (sequencing of three blocks in PGP mode). If no part program is being executed when subroutine %9999 is called, the CYCLE stops at the end of execution of the subroutine (the part program is not executed). Functions G01 and G40 are set at the start of the subroutine.

During execution of subroutine %9999

The PLC ignores a new call or maintaining of the call to subroutine %9999 during execution of this subroutine.

End of subroutine %9999

At the end of execution of the subroutine, the system does not report to the PLC. It is the subroutine's responsibility to report on its execution for example by M function or external parameter E.

Return to current program

After execution of the subroutine, none of the data programmed earlier are restored.

The following data must be reprogrammed if necessary:

- modal preparatory G functions,
- modal technological S functions and miscellaneous M functions,
- correction D even if the tool was not changed,
- program variables L.

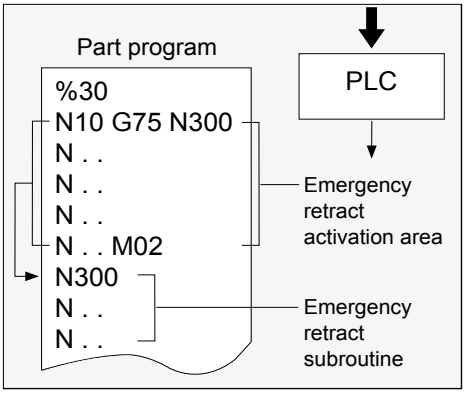
Context

The program context is not saved nor restored automatically. This can be done by programming using the program status access symbols. please refer to the Flexium+ Extended Programming manual M00055 for information on these symbols.

5.13.6 Emergency retract ► G75



G75	Declaration of an emergency retract subroutine.



The emergency retraction subroutine is activated by a request from the PLC. It can also be activated by a communication fault between NCK and PLC if this is defined by the machine parameters.

Activation stops the current program and causes a branch to address N.. of the last emergency retract subroutine declared.

Syntax

N.. **G75** N..

G75	Declaration of an emergency retract subroutine.
N..	Mandatory argument associated with the function and giving the address of the emergency retract subroutine first sequence number.

Properties of the Function

Function G75 is nonmodal. Argument N.. associated with the function is modal.

Cancellation

Declaration of a subroutine G75 N.. is cancelled by:
- cancellation function G75 N0,
- function G75 N.. with a different subroutine number,
- end of program function M02,
- a reset.

Condition for activation of the emergency retract subroutine

Part program being executed in one of the following modes:
- automatic (AUTO),
- single step (SINGLE),
- manual data input (MDI),
- rapid (RAPID).

In automatic mode, the emergency retraction program remains active until encountering a function M00 or M02.

An emergency retraction program can be activated:
- by reading of the block in which it is programmed,
- as long as the program or subroutine in which it is declared is not ended,
- as long as a new emergency retraction program has not been declared,
- as long as the emergency retraction function has not been cancelled by G75 N0.

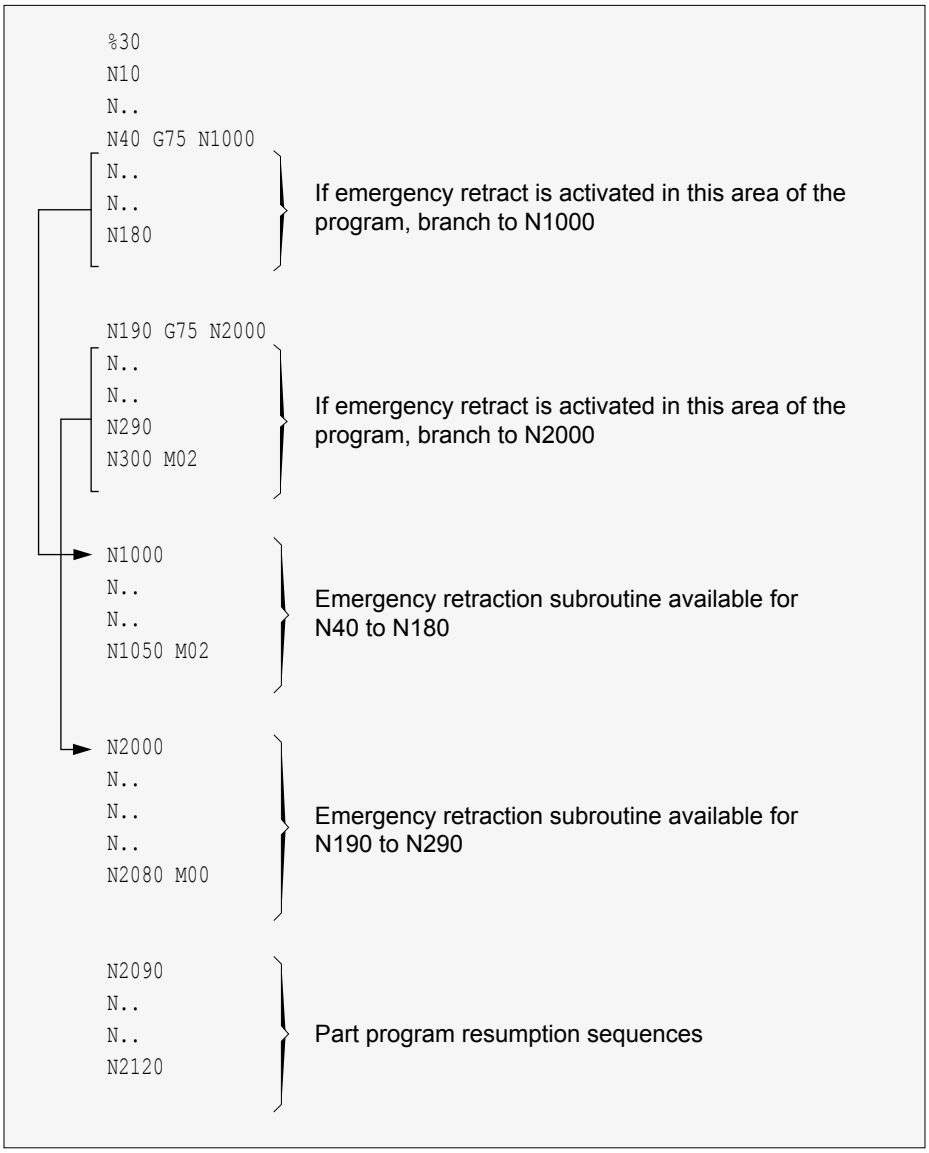
If emergency retraction is activated but no retraction program has been declared in the part program, it has the same effect as pressing Cycle stop key.

Activation of an emergency retraction is sent by the PLC bit:
Flexium_NCK.W_ALLCNC.EmergencyRetract.

It can then be generated automatically or on an operator request (refer to parameter E11022).

Example

Declaration of emergency retract subroutines from a main program



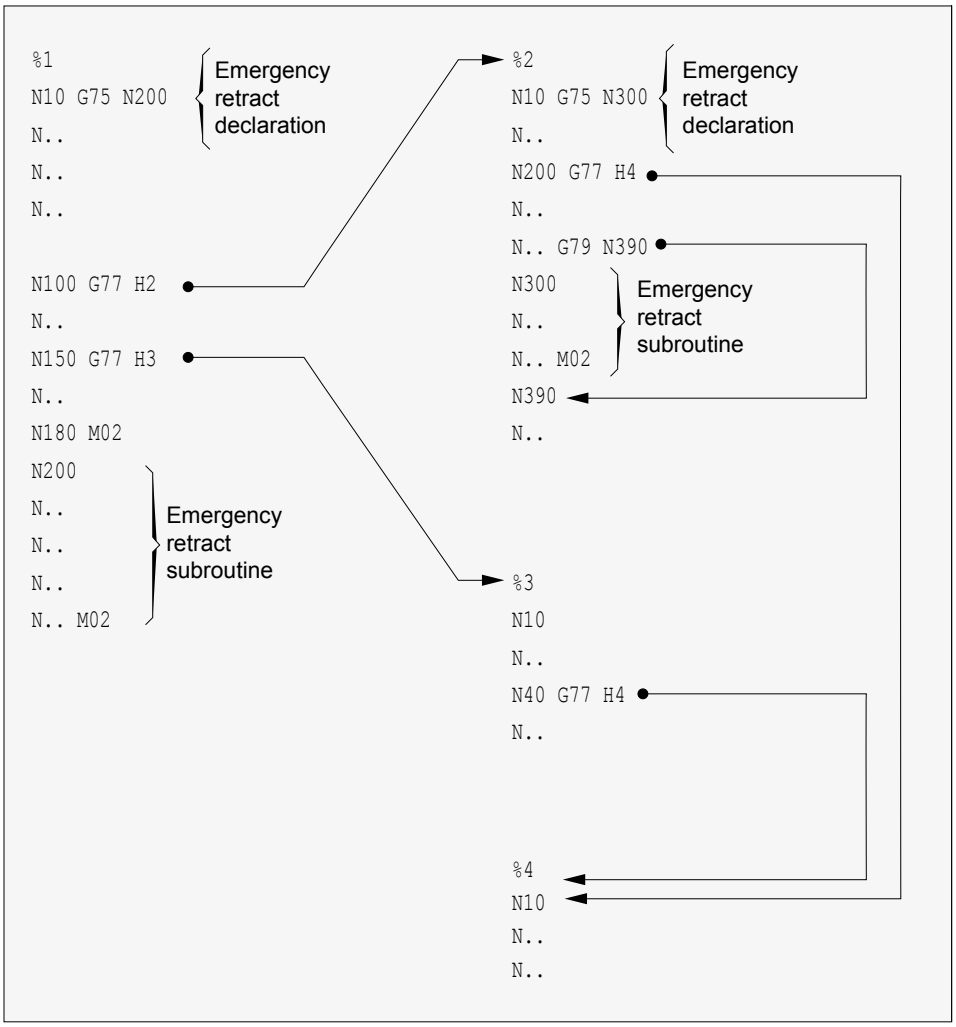
Declaration of emergency retract subroutines from a program including subroutine branches

When emergency retract is triggered, the branch will take place in the program or subroutine where it was declared. Exiting a subroutine clears the declaration and restores the previous (if any).

Examples

If an emergency retract subroutine is activated during execution of subroutine %4 called by N200 of %2, branch to sequence N300 of %2.

If an emergency retract subroutine is activated during execution of subroutine %4 called by N40 of %3, branch to sequence N200 of %1.



5.13.7 Block interrupt / Probing ► G10

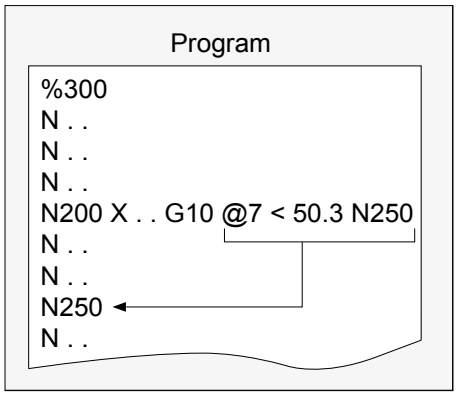


G10 Block interrupt.

G10 blocks are designed to be interrupted by a hardware input or by satisfying a position threshold for a particular axis.

After the interrupt, a branch may be arranged to a designated block. If no interrupt is detected, the next block is executed. If an interrupt occurs, all axis positions are stored.

Starting with revision 4.1.30 in addition to storing the positions with respect to the machine referential (in E7x001) the positions with respect to the last programmed frame will also be stored in parameters E7x009 (value per channel).



Note

The order of the data is important.

Syntax

N.. [G40] [G04 F..] [G00/G01/G02/G03] X.. Y.. Z.. **G10** [:n] [+X.. or F..] [@n < > Value] N.. [+ Number] [EF..]

G40	Tool radius offset cancel.
G04 F..	Interruptible dwell, but without branch to designated block.
G00/G01/G02/ G03	Interruptible interpolations.
X.. Y.. Z..	Interruptible axes.
G10	Block interrupt function.
:n	Numerical argument (number of interrupts from 1 to 99) significant only for hardware interrupts (on-the-fly dimension measurement). The Block interruption will occur after n interrupts (default 1).
+X.. or F..	Arguments defining a movement or dwell to be made after the interrupt(s). X..: A further movement on the trajectory to be made after the interrupt(s) before deciding whether to branch (possible on all the axes of the system, both measured and servo). F..: Dwell in secs during which the block may be interrupted.
@n < > Value	Argument defining a condition for comparison with a threshold. @n: Physical address of the axis concerned by the test (axis address n is between 0 and 31). < >: Mandatory comparison symbol (Less than or greater than). Value: Comparison threshold in the same unit as the other axes of the channel (mm or inches).
N.. + Number	Branch to block N.. after the interrupt, possibly followed by a further number to reach unnumbered blocks.
EF..	Maximum feed rate after an interrupt.

Property of the Function

Function G10 is nonmodal.

Cancellation

Function G10 is cancelled at the end of the block.

Notes

All the arguments for function G10 are optional.

It is not permitted to exit out of a loop directly (unlike G79)

Does not work with active tool radius correction (G41/G42).

Example for exiting a loop properly

```
FOR [I] = 1 TO [N] DO
[V] = [P(1,I)]+2
M201 G[G(V)] X[P(2,I)] Y[P(3,I)] I[P(4,I)] J[P(5,I)] F[GC(115)] G10 N121
CONT
N121 G79 N140
ENDF
N140 (After G10)
```

5.13.7.1 Analysis of the interrupt

When function G10 is programmed, an external interrupt may prematurely terminate the move. The end point programmed will then not be reached. This signal stops the movement and replaces the dimensions of the required position by the dimensions of the current position.

If a sequence number (N address within the block data, not block number of the considered sequence) was programmed in the same block as the G10 function, then the system will jump to this sequence number if and only if an interrupt has been triggered.

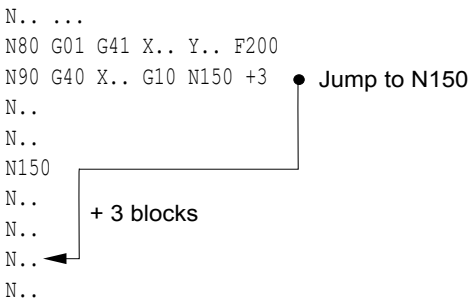
When the interrupt is generated, the dimensions of all axes in the channel are stored in external parameters E7x001 (axis position reference where x = physical address of the axis) and starting with revision 4.1.30 the positions with respect to the last programmed frame will also be stored in parameters E7x009.

New dimensional values can be detected by dynamic operators and stored in an indexed addressing table.

Examples

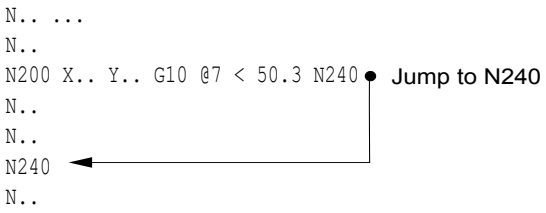
G10 block with a jump to an unnumbered block

Jump to the third block after N150 if a hardware interrupt is set.



Comparison with a threshold and jump to a block once the condition is satisfied

Condition: Jump to block N240 when the measurement on axis 7 is less than 50.3.



Expecting two interrupt + travelling an additional distance

```
N..
N200 G00 G40 X50
N210 G01 X100 Y120 G10 : 2 + X60 N250
N220
N230
N240
N250
N..
```

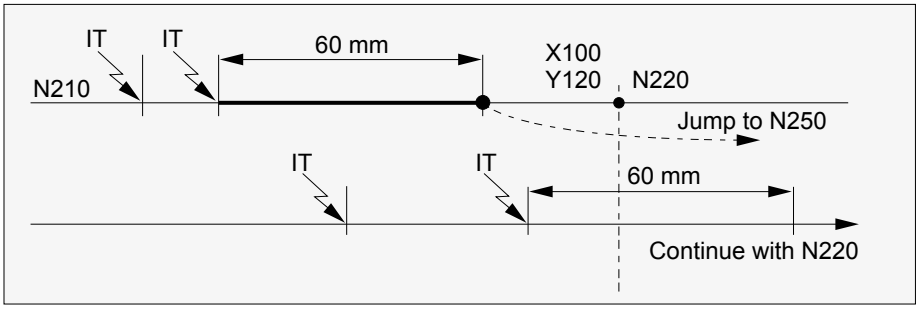
Jump to N220 or N250

Condition not satisfied

Condition satisfied

If 60mm beyond the second interrupt the end point is beyond the original programmed target:
- the jump condition is not satisfied and the program continues with the next block.
- no updates are made to the E7x001 positions.

Diagram:



5.13.7.2 Positions in programmed frame

This functionality available since revision 4.1.30 is activated when the parameter E11028 is set to 1 (it is cleared at power up and by reset).

The dimension are expressed in WCS (workpiece coordinate system) or in ICS (inclined coordinate system if any). RTCP or iPlane may be enabled. The optional part balancing is also taken into account.

According to the tool offsets compensation, the dimensions computed by this function are:

- The tool tip (if RTCP disable)
- The tool center point (if RTCP enable)

The tool radius offset compensation is ignored.

The dimensions provided are equivalent to the ones displayed in the page Prod of the HMI (Pos OP) with the 2 following differences:

- The dimensions are in internal unit
- The dimensions are always expressed in metric system (i.e. no inch) like the ones in E7x001

However the dimensions include possible geometrical transformations activated (G59, ED, …) as the ones displayed in the page Prod of the HMI (Pos OP).

The dimensions expressed in WCS (or ICS) are computed from the actual dimensions (in MCS), therefore several data are used to re-build these dimensions:

- DAT1, DAT2, DAT3
- Tool length (milling) or Tool Z-X lengths (turning)
- Programming on diameter or on radius (turning)
- Angle of an inclined axis (grinding)
- Configuration axes carrier/carried
- Rotary axis modulo (or not)

For a correct result it is supposed that above data above remain stable during probing.

The reference correction (E950xx) should be set to null.

In case of a G10 on comparison with respect to a threshold (general syntax: G10 @n <> Threshold), the value of the threshold remains expressed in machine frame.

5.13.7.3 Special use of sequence interrupt

Block sequencing without stopping movement

In M999 status the analysis of the following blocks is suspended until the interrupt is detected.

When the interrupt is detected, if the current interpolated position (to which is added the +X value programmed after G10) is beyond the programmed end point, it should be noted that:

- analysis of the following blocks is resumed immediately,
- in case of possible processing by parametric programming, the current block is considered ended, thereby authorizing read or write of external parameters E without effectively stopping current movements (this allows calculation of a positioning point according to the interrupt point and taking into account the geometric transformations made downstream of the interpolator),
- as soon as the next block is ready, the feed rate at the end of the current block is updated according to whether the path has changed or not.



In state M999, if the block was not interrupted, a subroutine call by the PLC is accepted at the end of execution of the block.

Feed rate limiting after an interrupt

Before the interrupt, the speed of movement in block G10 is the modal feed rate F programmed before G10. After the interrupt, the feed rate at the end of the current block may be different if argument EF and its value (less than F) are programmed in that block or an earlier block.

Example

```
N..
N.. G01 X200 F300
N..
N.. G00 X..
N100 G01 X0 F150 G10 +X2 EF100          Feed rate 100 mm/min after the interrupt
N..
```



It is possible to measure the axes positions on an interrupt signal without having to program G10 and therefore without block interruption or branches. This is accomplished through parameter E11019 and is explained more in details in the Flexium+ Extended programming manual M00055.

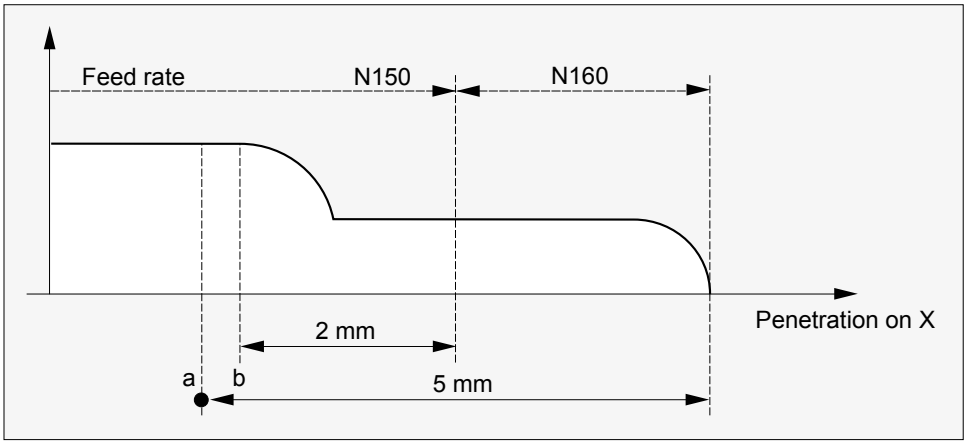
Example

The example below is defined for a system using inclined axis programming.

For the specific data used in the example, refer to the additional information given in this manual for:
- Inclined axes
- External parameters E...
- Symbolic variables [...].

```
%398
N..
N..
(Compute the offset between the machine dimension and program dimension in the program reference
system)
[K] = CE69001 [DECX] = UE70000 * [K] - [.IRX(1)]
N150 G01 X0 F200 G10 +X2 EF100
(Go to 5 mm from the interrupt point)
N160 [COTE] = [K] * UE70001 - [DECX] - 5 X[COTE] F100
N..
```

In the figure:
- point a: point measured on the interrupt
- point b: current position interpolated at the time of the interrupt.

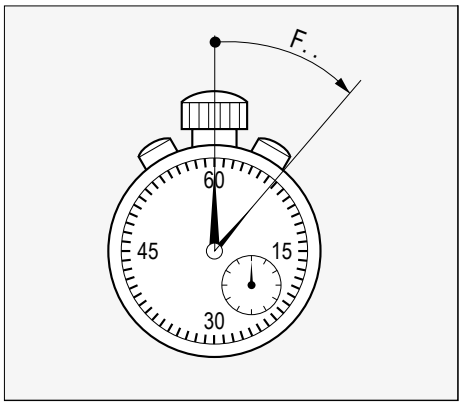


5.14 Standard functions

5.14.1 Dwell ► G04



G04	Programmable dwell.



Continuation of the program is interrupted for the time programmed with argument F.

Syntax

N.. **G04** F..

G04	Programmable dwell.
F..	Dwell in seconds (0.01 to 99.99 seconds, format F022). Mandatory argument F must be programmed immediately after the function.

Property of the Function

Function G04 is nonmodal.

The dwell amount as well as the current value are shown on the HMI and available in E parameters.

Function G04 is cancelled at the end of the block.



Function G04 may be cancelled before the end of the block when it is programmed with function G10.

Notes

Function G04 F.. does not cancel the feed rates programmed with F in the previous block(s).

Example:

```
N.. ...
N50 G01 Xa Ya F200          Feed rate 200 mm/min
N60 G04 F1.5                Dwell 1.5 seconds
N70 Xb Yb                   After 1.5 seconds machining resumes at 200 mm/min
N..
```

If G04 is programmed at the start of a block containing a move, the dwell will occur at the end of the block.

Example:

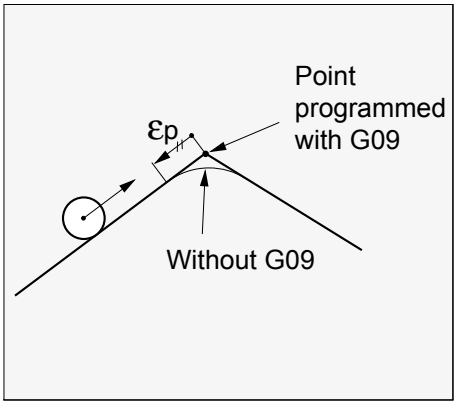
```
N.. ...
N.. G01 G04 F5 X100 Y100    Dwell after movement
N..
```

5.14.2 Accurate end of block Stop ► G09



G09	Accurate stop at end of block before continuation to next block.

When the function is programmed, the CNC system will wait until the axes are precisely at the programmed position before resuming execution.



Syntax

N.. **G09** [:R / :M / :RM / -] [G00/G01/G02/G03] X.. Y.. Z.. [F..]

G09	Accurate stop at the end of a block before continuation to the next block.
[:R / :M / :RM / -]	Defines Full stop, Modality and Cancellation (see table)
G00/G01/G02/G03	Linear or circular interpolation.
X.. Y.. Z..	Coordinates of the end point.
F..	Feed rate.

Property of the Function

Function G09 is nonmodal.

Cancellation

Function G09 is cancelled at the end of the block.

Variants

- It is possible to make the G09 function modal by: **G09:M**
- The modal function will be cancelled by: **G09-**
- Instead of accurate stop G09 could also be used to stop the move until the programmed position is reached without waiting for reaching the in-position window: **G09:R**
- Both functions can be used together G09: **RM or G09:MR** .

	Accurate positioning	**Modal**	**Cancellation**
G09	Yes	No	No
G09:R	No	No	No
G09:M	Yes	Yes	No
G09:RM (or G09:MR)	No	Yes	No
G09-	No	No	Yes
G09:	Analysis error: 2 Extension: 59		

Notes

The lag error εp is directly proportional to the feed rate.

The more acute the angle between two paths, the greater the smoothing effect at a given feed rate and therefore a given εp.

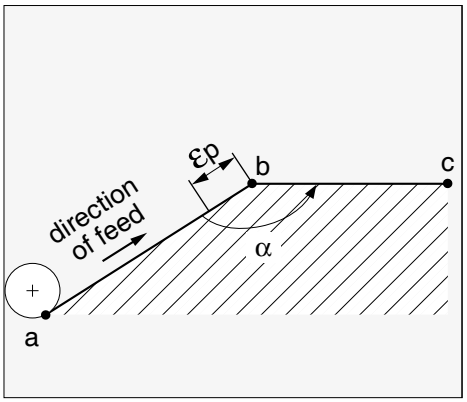
When G09 is programmed:
- the lag εp is decreased up to Zero at the end of the block except with :R,
- the feed rate is also zero at the end of the block.

Examples

G09 activated

The axes are decelerated along path ab at a distance εp from point b and go through point b.

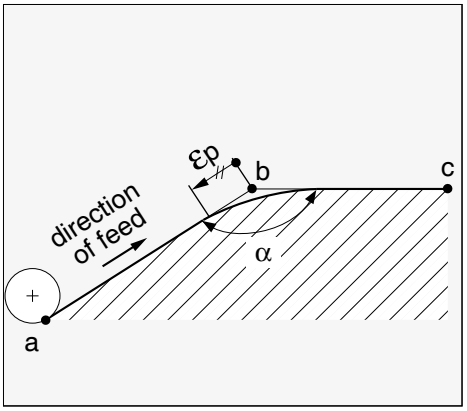
```
N.. ...
N100 G01 Xa Ya F500
N110 G09 Xb Yb
N120 G09 Xc Yc
N..
```



G09 inactive

The axes are not decelerated and do not pass exactly through point b.

The final path results from the feed rates along ab and bc and the angular value of the vectors.



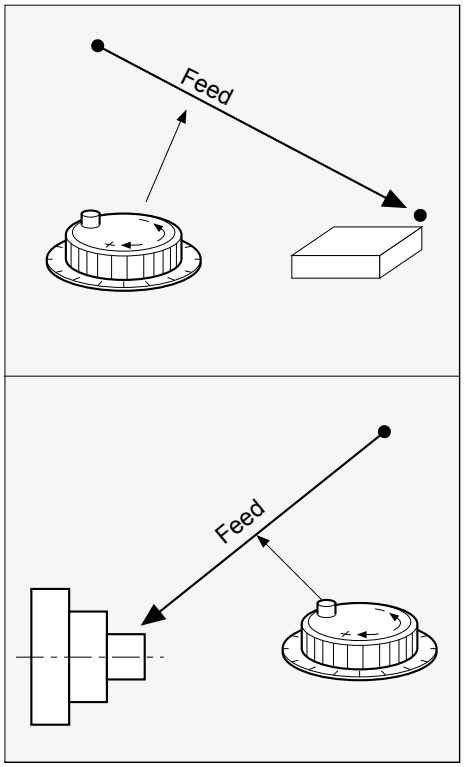
5.14.3 Feed increase ► G12



G12	Feed Increase by handwheel.

When the machine is equipped with handwheels, this function is used to increase the speed of movement on the linear or circular paths programmed in the block.

Overspeed uses the first handwheel.



Syntax

N.. [G01/G02/G03] **G12** X.. Y.. Z.. [F..] [$0 ...]

G01/G02/G03	Linear or circular interpolation.
M12	Enabling of rapid feed by handwheel.
X.. Y.. Z..	End point.
F..	Feed rate.
$0...	Message for the operator.

Property of the Function

Function G12 is nonmodal.

Cancellation

Function G12 is cancelled at the end of the block.

Notes

Continuation to the next block is carried out when the programmed position is reached.

The overspeed coefficient applied by function G12 is defined by the OEM.

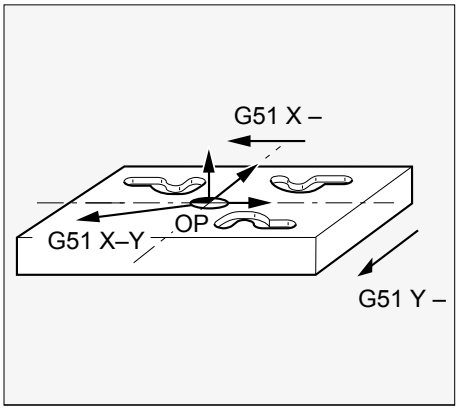
Example

```
N.. ...
N60 G01 G12 X.. Y.. F200 $0 ACTUATE THE HANDWHEEL
N..
```

5.14.4 Mirror function ► G51



G51	Mirror function.



This function allows symmetrical machining of program sections defining a quarter or half of the part.

The mirror is enabled or cancelled according to the axis and algebraic sign arguments programmed with the function.

Syntax

N.. **G51** X- Y- Z- A- B- C-

G51	Mirror function.
X- Y- Z- A- B- C-	The minus sign (-) enables the mirror function on the X, Y, Z axes or A- B- C-

Properties of the Function

Function G51 is nonmodal. The axis arguments (X, Y, Z, A, B, C) associated with the function are modal.

Cancellation

Mirroring on the programmed axes is cancelled by:
- function G51 followed by one or more arguments X+, Y+, Z+, A+, B+ or C+ cancelling the former state G51,
- the end of program function (M02),
- a reset.

Notes

When G51 is programmed:
- it must be programmed with at least one argument (axis and sign),
- it must be programmed alone with its arguments in the block,
- several axes can be enabled or inhibited in the same block,
- the axes to which the mirror function applies are displayable on the information page (INFO),
- if a mirror function is applied to a carried axis, it is also automatically applied to the carrier axis.

The mirror function affects:
- the sign of the programmed axis X, Y, Z, A, B or C which is reversed. The sign change is carried out with respect to the program origin defined by DAT1 and DAT2,
- the origin offsets programmed (G59),
- tool radius offset (G41, G42),
- the direction of rotation with circular interpolation (G02, G03).

The mirror function does not apply to:
- the part origin DAT1,
- the shift between the part origin and the program origin (DAT2),
- table off-centring (DAT3),
- the tool dimensions (L, R, @),
- programming with respect to the measurement origin (G52).

When the mirror function is enabled on the axis corresponding to the tool orientation axis, a new orientation must be programmed (G16 ...).

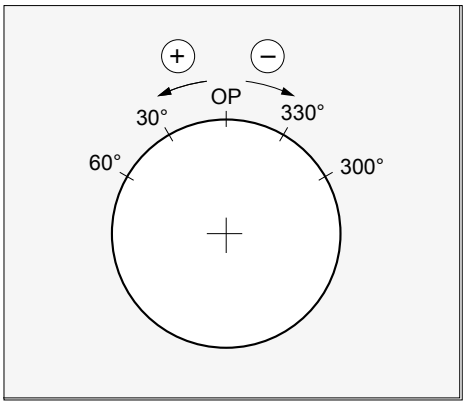
- The symbolic variable [.IBXM(i)] indicate mirroring on an axis.
  See manual M00055 for more information.

5.14.4.1 Influence of the mirror function on the A, B or C rotary axes

The function reverses the direction of rotation of the rotary axes and takes the complement to 360 degrees.

If B30 is programmed: the axis moves to 330 degrees in the negative direction.

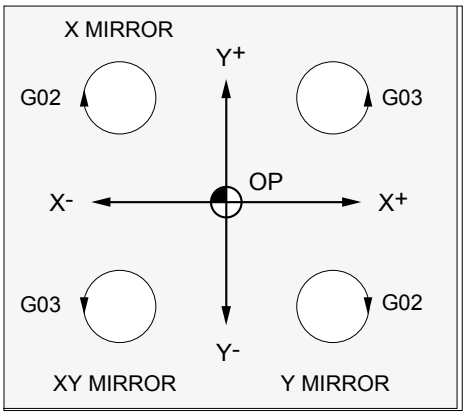
If B-300 is programmed: the axis moves to 60 degrees in the positive direction.



When the mirror function is applied to a rotary axis, it should be checked by a test before machining that the axis rotates in the required direction.

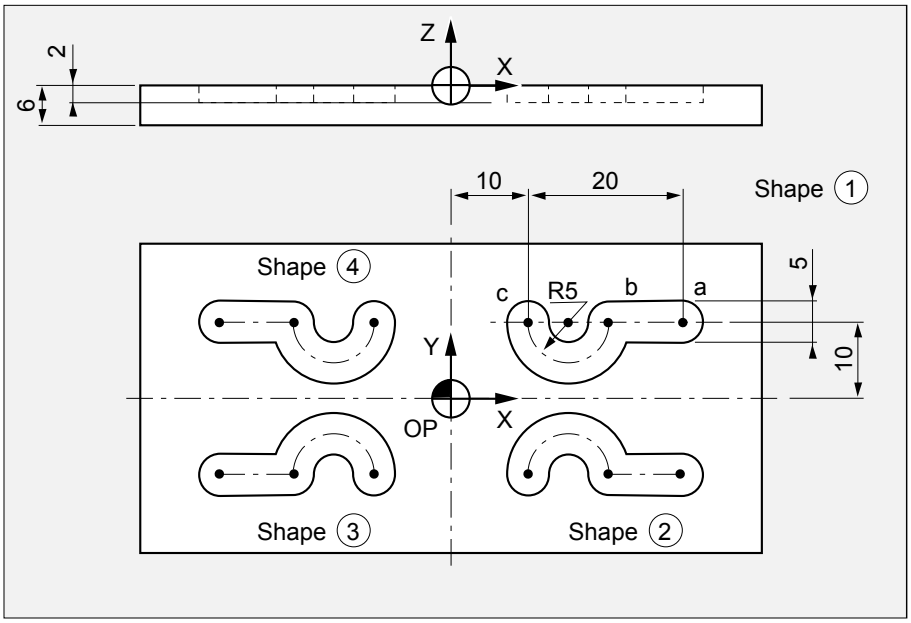
5.14.4.2 Influence of the mirror function on circular interpolations

The function reverses the clockwise (G02) and counter-clockwise (G03) rotations.



Example

Execution of a shape using mirror function G51 in the XY plane



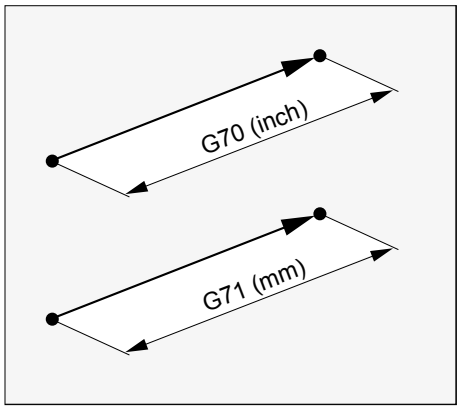

```
%30
N10 G00 G52 Z0
N20 T05 D05 M06 (CUTTER DIAMETER=5)
N30 S1500 M40 M03
N40 G00 X30 Y10 Z2          Point a, approach on XYZ
N50 G01 Z-2 F50 M08         Penetration on Z
N60 X20                     Point b
N70 G02 X10 Y10 R5          Point c
N80 G00 Z2                  Retraction on Z
N90 G51 Y-                  Mirror on Y
N100 G77 N40 N80            Execution of shape 2
N110 G51 X- Y-              Mirror on X and Y
N120 G77 N40 N80            Execution of shape 3
N130 G51 X- Y+              Mirror on X, cancellation of mirror on Y
N140 G77 N40 N80            Execution of shape 4
N150 G51 X+ Y+              Cancellation of mirror on X and Y
N160 G00 G52 Z0 M05 M09
N170 M02
```

5.14.5 Programming in Inches or Metric data ► G70, G71



G70	Inch data input.

This function is used to program data in inches.



G71	Metric data input.

This function is used to program data in metric units.

Syntax

N.. **G70/G71**

G70	Inch data input.
G71	Metric data input.

Properties of the Functions

Functions G70 and G71 are modal.

Function G71 is the default function.

Cancellation

Functions G70 and G71 cancel one another.

Notes

The default programming unit can be changed from inches to metric units and vice versa in machine parameter P7 (see Flexium⁺ Parameters Manual M00041).

It should be noted that the choice of the displayed unit (inches or metric units) is made by PLC variable *"Flexium_NCK.W_ALLCNC.InchUnit"*.

Formats specific to inch data input (G70)

Addresses	Formats
X, Y, Z, U, V, W, I, J, K	+044
P, Q, R, ER etc...	+044

Inch data input (G70) for program variable «L» and external parameters " E "

The programming must be suited to the operations carried out on the dimensions and external parameters " E " expressed in their own units. See chapter 7 and 8.

Examples

Programming of a dimension using program variable L

```
N.. ...
N.. G70
N.. L1 = 10
N.. G01 XL1                    L1 is equal to 10 inches
```

Programming of a dimension using an external parameter E

```
N.. ...
N.. G70
N.. E80000 = 100000
N.. G01 XE80000                E80000 is equal to 10 inches
                               (format 044)
```

Modification of a tool dimension by the program

External parameter E50001 represents the length of tool 1 (assuming tool management is not active).

The value stored into the "E" parameter is held in NC internal measurement unit (Refer to P4N2 setting on Flexium+ Parameters manual M00041).

The coefficient must be adjusted in relation to P4N2 setting "NC internal measurement unit".

In case of µm the coefficient is 25400, in case of 1/10µm the coefficient is 254000.

```
N.. ...
N.. G70
N.. L10 = E50001/25400         L10 = length of tool 1 converted to inches
N.. L2 = 3 + L10               L2 = length of tool 1 + 3 inches
N.. E50001 = L2 * 25400
```

Notes

In the example above it is assumed that the internal resolution of the system is 1µm.

The value 25400 needs to be adapted fort other resolutions (e.g. 254000 for 1/10µm resolution).

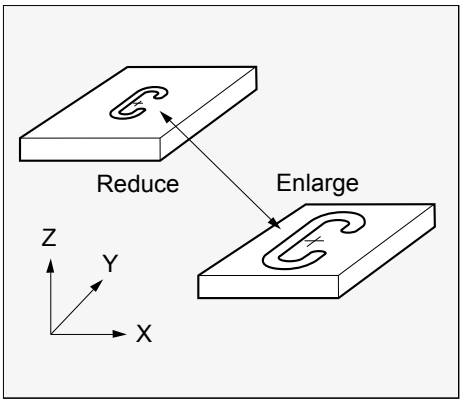
The system internal resolution is defined by the machine builder in parameter P42 N2.

5.14.6 Scaling factor ► G74, G73



G74	Scaling factor enable.

This function allows execution of scaled parts from a single programmed part or template. The scale ratio can be entered from the keyboard or programmed.



G73	Scaling factor cancel.

Syntax

N.. [G40] **G74/G73**

G74	Scaling factor enable. The ratio declared can be between 1 and 9999 to 1000 (0.001 to 9.999) and it must be an integer.
G73	Scaling factor cancel.

Properties of the Functions

Functions G73 and G74 are modal.

Function G73 is the default function.

Cancellation

Functions G73 and G74 cancel one another.

Function G74 is cancelled by the end of program function (M02).

Notes

Scaling is around the program origin (OP).

The ratio can be entered from the keyboard or programmed by external parameter E69000.

Functions G73 and G74 must be programmed:
- with the system in G40 mode (radius offset cancel),
- in a block that does not contain any circular interpolation,
- outside a sequence of incompletely defined PGP (Profile geometry Programming) blocks.

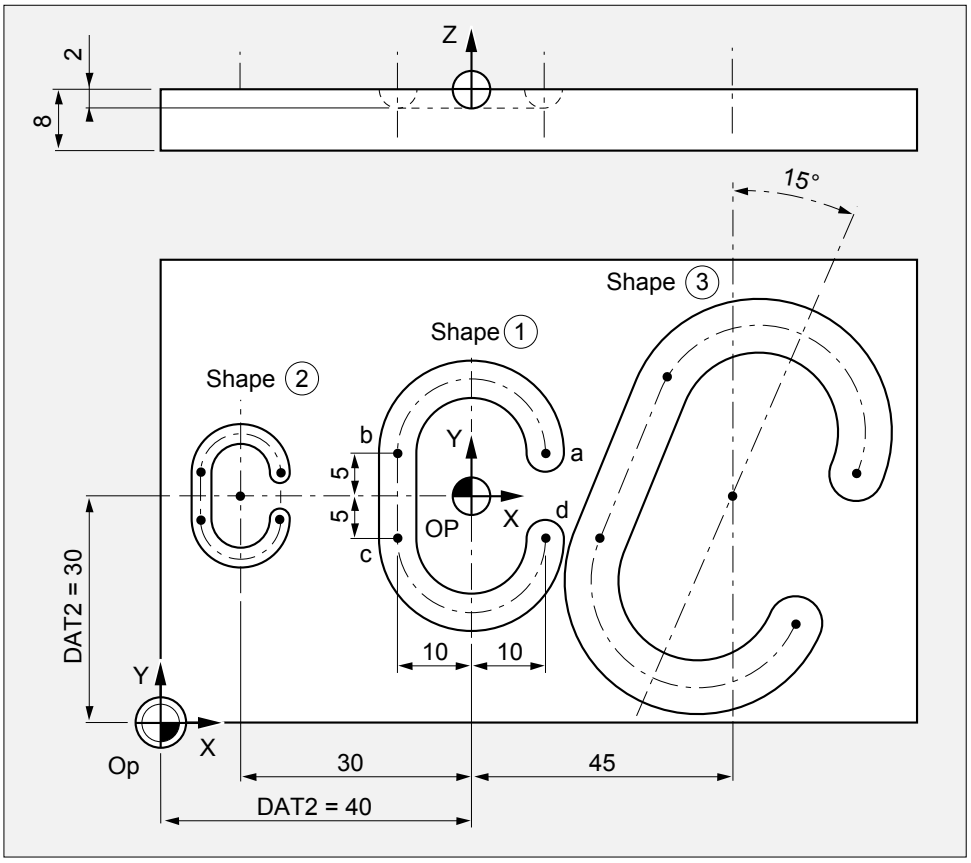
The ratio affects:
- the values programmed for the primary and secondary axes (X, Y, Z, U, V, W),
- the program origin offset (G59).

The ratio does not affect:
- the values programmed for the rotary axes (A, B, C),
- the part origin DAT1,
- the shift between the part origin and the program origin (DAT2),
- table off-centring (DAT3),
- tool dimensions (L, R, @),
- programming with reference to the measurement origin (G52),
- the clearance dimension for canned cycles G81 to G89.

Please refer to the Flexium+ HMI manual M00039 for operation of entering the scaling factor on the keyboard.

Example

Execution of shapes using scaling factors



- Shape 1: Scale 1 from the program zero.
- Shape 2: Scale 0.5 with origin offset (G59) on X-30.
- Shape 3: Scale 1.5 with origin offset (G59) on X45 and angular offset (ED) of 15 degrees.

As the origin offset is affected by the scaling factor, values assigned to G59 must be proportional to the programmed scaling factor.

Main program example

```
%300
N10 G00 G52 Z0
N20 T01 D01 M06 (BALL END CUTTER, DIAMETER=8)
N30 S2000 M03 M40
N40 G59 X0 Y0                          Zero origin offset
N50 G74 E69000 = 1000                  Scaling factor 1
N60 G77 H3                             Subroutine call for shape 1
N70 G59 X-30                           Origin offset
N80 G74 E69000 = 500                   Scaling factor 0.5
N90 G77 H3                             Subroutine call for shape 2
N100 G59 X45 ED-15                     Origin offset
N110 G74 E69000 = 1500                 Scaling factor 1.5
N120 G77 H3                            Subroutine call for shape 3
N130 G00 G52 Z0 M05 M09
N140 M02
```

Subroutine example

```
%3
$0 SHAPE C
N10 X10 Y5 Z3                          Point a
N20 G01 Z-2 F100 M08                   Approach on Z
N30 G03 X-10 Y5 R10 F300               Point b
N40 G01 Y-5                            Point c
N50 G03 X10 Y-5 R10                    Point d
N60 G73 G00 Z3                         Scaling factor cancel
```



Example

Programming the scaling factor by external parameter E69000

External parameter E69000 is a read/write parameter. The value programmed must be an integer.

```
N.. ...
N40 E69000 = 250                       Ratio 250/1000, i.e. reduction to 0.25
N50 G74 G00 X.. Z..
N..
N200 G73                               Scaling factor cancel
N..
```

The scaling factor can be tested by a conditional branch in the program.

```
N.. ...
N.. G79 E69000 = 300 N210              If the ratio is equal to 300, jump to block N210
N..
N..
N210
N..
```

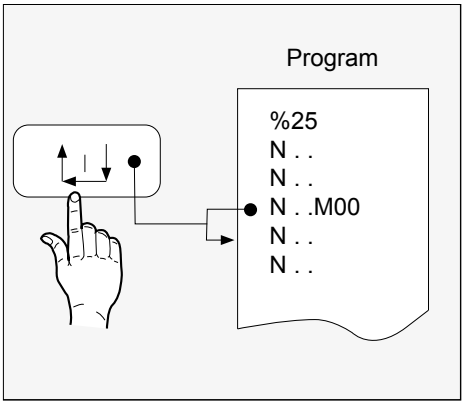
5.14.7 Program Stop ► M00



M00	Program stop.

This function stops execution of the current program.

After intervention or a check, the operator restarts the program.



Syntax

N.. [G40] **M00** [$0 ...]

G40	Radius offset cancel.
M00	Program stop.
$0 ...	Message for the operator.

Properties of the Function

Function M00 is a decoded nonmodal «post» function.

Cancellation

This function is cancelled by pressing the CYCLE key.

Notes

When function M00 is read in a block:
- continuation to the next block is interrupted and spindle rotation is stopped,
- the contents of the field in the status window are modified and the indicator CYCLE is replaced by M00.

After action or check by the operator, pressing the CYCLE key restarts the program.
The indicator M00 is replaced by CYCLE.
M00 must be programmed with the system in G40 mode (radius offset cancel).
Spindle rotation must be reprogrammed after a stop programmed by M00.

Transfer of function M00 to the PLC

Function M00 is transmitted to the PLC at the end of execution of the block in which it is programmed, but before waiting for channel synchronization if required (G78..) in a system with multiple channels (in this case, stopping of the spindle is taken into account when synchronization is completed).

Example

```
N.. ...
N.. G01 G41 X.. Y.. F200 M08
N..
N190 G00 G40 Z200 M09                 Tool retraction before intervention
N200 M00 $0 REMOVE CHIPS BEFORE       Stop and message
FINISHING
N210 G00 G41 X.. Y.. Z.. M03          Continuation of the program
N..
```

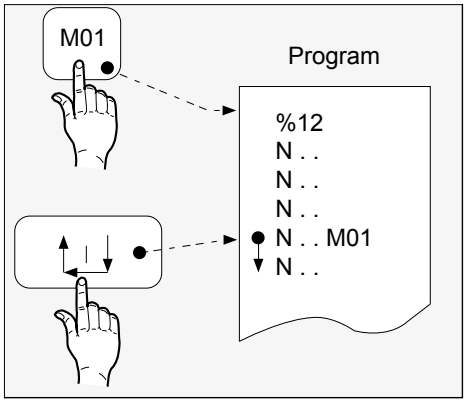
5.14.8 Optional Stop ► M01



M01	Optional stop.

When confirmed by the operator M01 causes a stop in program execution.

After intervention or check, the cycle is restarted by the operator.



Syntax

N.. [G40] **M01** [$0 ...]

G40	Radius offset cancel.
M01	Optional stop.
$0 ...	Message for the operator.

Properties of the Function

Function M01 is a decoded nonmodal "post" function.

Cancellation

The function is cancelled by pressing the CYCLE key.

Notes

When function M01 is read in a block (M01 confirmed):
- continuation to the next block is interrupted and spindle rotation is stopped,
- the contents of the field in the status window are modified and the indicator CYCLE is replaced by M00.

After action or check by the operator, pressing the CYCLE key restarts the program. The indicator M01 is replaced by CYCLE.
M01 must be programmed with the system in G40 mode (radius offset cancel).
Spindle rotation must be reprogrammed after a stop programmed by M01.

Transfer of function M01 to the PLC

Function M01 is transmitted to the PLC at the end of execution of the block in which it is programmed, but before waiting for channel synchronization if required (G78..) in a system with multiple channels (in this case, stopping of the spindle is taken into account when synchronization is completed).

Example

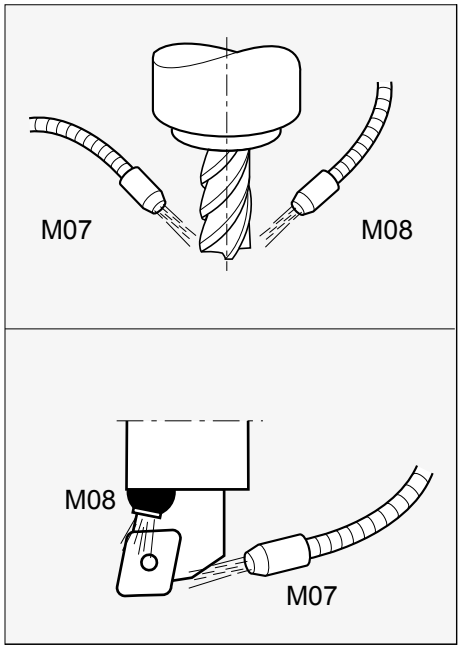
```
N.. ...
N.. G01 G42 X.. Y.. F200 M08
N..
N290 G00 G40 Z200 M09                Tool retraction before intervention
N300 M01 $0 CHECK DIMENSION 50       Stop if M01 is confirmed
EVERY 5 PARTS                        and message
N310 G00 G42 X.. Y.. M03             Continuation of the program
N..
```

5.14.9 Coolant ► M08, M07, M09



M08	Coolant 1 on.

M07	Coolant 2 on.



These functions turn on the coolant pumps.

M09	Coolant off.

This function stops the coolant pumps.

Syntax

N.. **M08/M07**

M08	Coolant 1 on.
M07	Coolant 2 on.

N.. **M09**

M09	Coolants 1 and 2 off.

Properties of the Functions

Functions M07 and M08 are decoded modal «pre» functions.

Function M09 is a decoded modal «post» function. It is the default function.

Cancellation

Functions M08 and M07 are cancelled by functions M09 or M02.

Example

```
N.. ...
N40 G00 X.. Y.. M08                Coolant 1 on
N50 G01 Z.. M07                    Coolant 2 on
N..
N230 G00 G52 Z-100 M05 M09         All Coolant off
N..
```

5.14.10 Axis clamping and unclamping ► M10, M11



M10	Clamp.

This function is used to clamp those axes with clamps that are not required to move.

M11	Unclamp.

This function cancels axis clamping. The machine works with all clamps released.

Syntax

N.. [G00/G01/G02/G03] **M10/M11** X.. Y.. Z.. A.. B.. C..

G00/G01/G02/G03	Linear or circular interpolation.
M10	Clamp.
M11	Un-clamp.
X.. Y.. Z.. A.. B.. C..	End point.

Properties of the Functions

Function M10 is a decoded modal "post" function.

Function M11 is a decoded modal "pre" function.

Cancellation

Functions M10 and M11 cancel one another.

Notes

The axes recognized as able to be clamped by function M10 are declared by the machine builder.

When function M10 is programmed, the system generates a timeout followed by a wait for an acknowledgement before executing the movements in the next block if some axes declared as clampable change status (will start or stop).

After a reset, function M10 is initialized on the channels which have axes that can be clamped.

Example

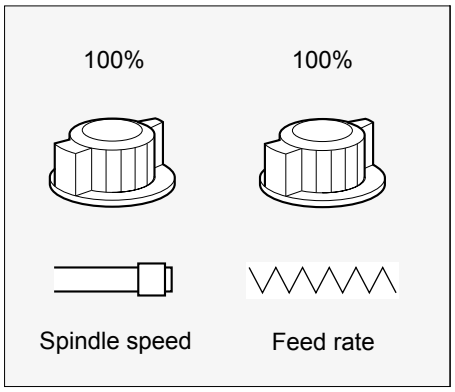
```
N.. ...
N40 M10                          Working with clamps enabled.
N50 G00 X.. Y.. Z.. B..          The X, Y, Z, B axes are not clamped.
N60 G01 X.. Y.. Z.. F200         The B axis which does not move is clamped.
N..
N210 M11                         Un-clamp all axes.
N..
```

5.14.11 Overrides inhibit ► M49, M48



M49	Spindle speed and feed rate potentiometer inhibit.

During execution of the program, the operator cannot override the spindle speed and feed rate. The potentiometers are forced to 100 percent.



M48	Spindle speed and feed rate potentiometer enable.

Syntax

N.. **M49/M48**

M49	Spindle speed and feed rate potentiometer inhibit.
M48	Spindle speed and feed rate potentiometer enable.

Properties of the Functions

Function M48 is a modal "post" function. It is the default function.

Function M49 is a modal "pre" function.

Cancellation

Functions M48 and M49 cancel one another.

Notes

Programming M49 causes:

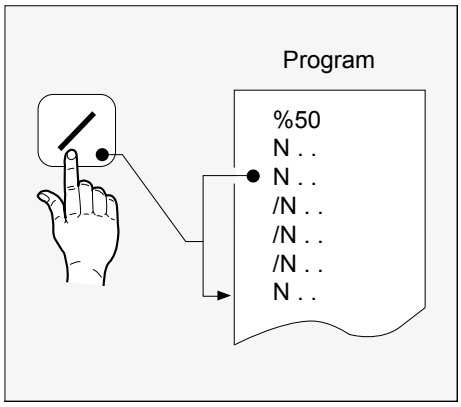
- Forcing of a feed rate of 100 percent (with M48, feed rate override possible from 0 to 120% of the value programmed with F).
- Forcing of a spindle speed of 100 percent (with M48, spindle speed override possible from 50 to 100% of the value programmed with S).

5.14.12 Block Skip ► / N



/	Block skip.

A block preceded by a slash " / " is ignored when block skip is enabled by the operator.



Syntax

/[i] N (Contents unimportant)

/	Block skip (/).
[i]	Optional number or suite of numbers defining the block skip level.
N	Block number.

Cancellation

By inhibiting block skip.

Notes

Block skip «/» is active when confirmed by the operator (the " / " indicator is displayed in the status window).

It is possible to define up to 32 levels of block skip. The following syntax exist:

/ X.. Y.. (standard functionnality)

/0 X.. Y.. (/0 is equivalent to /)

/i X.. Y.. (i =0..31)

/i,j,k N.. X.. (levels i,j,k considered)

/n-m N.. X.. Y.. (levels from n to m considered)

Combinations are also possible

/i,j,n-m,p,r X.. Y..

The different levels can be set via HMI (Mode Function bar / NC Function / Set Block skip level), via Parametric programming (E12000) or the PLC.

For additional information about setting the skip level, please refer to Flexium+ HMI operator's manual M00039.

5.15 Advanced Miscellaneous functions

5.15.1 Automatic homing



Homing can be done automatically on each of the machine axes by running subroutine %9990.

Conditions for running subroutine %9990

Subroutine %9990 is run by pressing the cycle start button:

- if the system is in homing mode,
- if no other program is being executed.

During execution of subroutine %9990

As soon as subroutine %9990 or one of its subroutines starts, the machine axes are authorized for programmed motion despite not being homed.

End of execution of subroutine %9990

At the end of execution of subroutine %9990 (by programming of M02), the program that was active when %9990 was called is re-instated.

5.15.2 Subroutine call on Reset %11000



On a reset, the NC channels can execute subroutine %11000.

Please note that there is one %11000 per channel except auxiliary channels.

Conditions for running subroutine %11000

Subroutine %11000 is run:
- if activated by the machine builder,
- if this subroutine is present in the memory.

Start of execution of subroutine %11000

Subroutine %11000 begins approximately 100 microseconds after the reset pulse goes high. It should be noted that this pulse remains high until the subroutine ends.

Contents of subroutine %11000

A subroutine %11000 must not include any executable functions such as:
- Movements on the axes
- Dwells (G04 F..)
- Miscellaneous functions (M..), etc.

Only the following operations are allowed in subroutine %11000:

- read and write of external parameters E and program variables L (see chapter 7). It should be noted that at the end of execution of the subroutines, variables L0 to L99 are reset but the others preserve their values,
- declaration of dynamic operators and enabling of interaxis calibration (see Flexium Extended Manual M00055),
- declaration and initialization of symbolic variables [...]. It should be noted that they must be saved by the SAVE function to be available after %11000 has ended.



Starting with version 4.1.20 it is possible under certain conditions to initialize L20 to L99 variables during %11000.
- This functionality is enabled when E11026 = 1 (channel specific).
- Only variables L20 to L99 can be written.
- The variables can only be written in the LAST line of %11000.

Revision 4.1.30 and above offer additional possibilities

As stated above %11000 cannot handle some function because it is executed during reset phase therefore it has been added the possibility to automatically execute more commands (excluding spindle and axis moves) after %11000 execution.

This is enabled by the global parameter E11029 and the commands are to be given in %11001. In independent mode E11029 must be set for each channel in which %11001 should be executed.

Specific errors linked to this feature :

NCERR_1 Programming error

EXT1_136 Axes movements in subroutine 11001

NCERR_1 Programming error

EXT1_121 Spindle commands in subroutine 11001

Warning 1009 : Subroutine 11001 not found channel %1 if the %11001 is not present in the memory with E11029=1

Plc Flags :

During the execution of the %11001 the PLC flags are set as below

Reset pulse low,

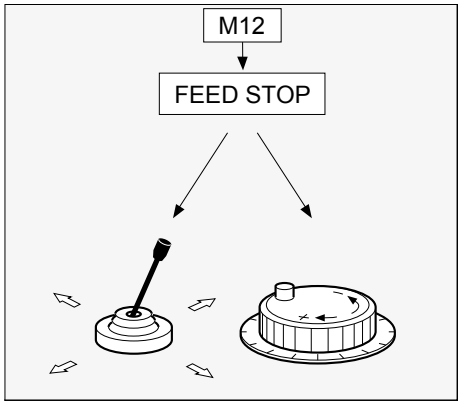
ProgramActive high

Program number %11001

5.15.3 Programmed feed stop for intervention ► M12



M12	Programmed feed stop.



Must be defined by the machine builder. Without this parametrisation M12 is just like another M function.

This function requires operator action after a cycle stop action.

The axis jogs or handwheel are made available.

Function M12 used a programmed feed stop.

When used as a programmed feed stop, M12 will also act as a meeting point between different NC channels

Syntax

N.. **M12** [$0...]

M12	Programmed feed stop.
$0...	Message for the operator.

Properties of the Function

Function M12 is a decoded nonmodal «post» function.

Cancellation

The function is cancelled by pressing the CYCLE key on the machine panel.

Notes

M12 should be activated by the machine builder to work as explained.
(see Machine Operation manual).

When function M12 enables the axis jogs or handwheel:
- the operator can only make movements in continuous jog mode (J.ILL),
- during axis movements, the system remains in the current execution mode: automatic (AUTO) or single step (SINGLE).

When the operator cancels the function after any axis movements, the program is resumed from the new position (axis recall not required).

Function M12 is ignored in the test (TEST) and sequence number search mode (SEARCH).

Example

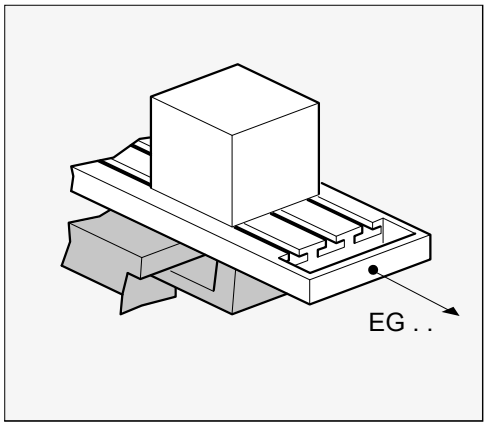
```
N.. ...
N300 G00 Z80 M12 $0 MOVEMENT ON X, THEN CYCLE START
N310 ..
N..
```

5.15.4 Acceleration reduction ► EG..



EG..	Programmed acceleration reduction.

This function followed by a value allows for modulating the axes acceleration from 1 to 100% of their maximum value as defined by the a machine parameters on the programmed movements. This allows the stresses due to movement of heavy loads to be limited.



Syntax

N.. **EG**..

EG..	Programmed acceleration reduction. A positive integer giving the percentage (between 1 and 100%) of the values set by the machine builder.

Property of the Function

Function EG.. is modal.

Function EG is forced to 100 percent at power on.

Cancellation

Function EG is cancelled by:

- programming a new value (EG..),
- the end of program function (M02),
- a reset.

Notes

During programmed interpolated movements, the reduced acceleration is always taken into account by the interpolators except in the case of Cycle stop or activation of the feed safety device.

Acceleration reduction is ignored in the following modes or functions:

- axis jog (JOG),
- homing (HOME),
- action after a program stop (M12).

Example

```
N.. ...
N40 EG50 G00 X..          Acceleration reduced to 50% on X
N..
```

5.15.5 Spindle acceleration reduction ► EGS



EGS...	Programmed spindle acceleration reduction.

This function followed by a value allows for modulating the spindle acceleration between 1 to 100% of the value defined by machine parameters for example to a variation of inertia.

It is valid for all spindles of a channel.

Syntax

N.. **EGS..**

EGS..	Programmed acceleration reduction. A positive integer giving the percentage (between 1 and 100%) of the values set by the machine builder.

Properties of the Functions

Function EGS.. is modal.

Function EGS is forced to 100 percent at power on.

Cancellation

Function EGS is cancelled by:
- programming a new value (EGS..),
- the end of program function (M02),
- a reset.

5.15.6 Temporary suspension of next block preparation ► G79+/-



G79 +/-	Temporary suspension of next block preparation in a sequence with movements.

Programming this function in a block with movements temporarily suspends preparation of the next block. A lead or lag (expressed in millimetres or seconds) at the beginning of the current block or at the end of execution of the current block is programmed on the restart of block processing.

Syntax (XY plane)

N.. [G00/G01/G02/G03] X.. Y.. Z.. **G79 +/-** X.. / F..

G00/G01/G02/G03	Linear or circular interpolation.
X.. Y.. Z..	End point.
G79	Temporary suspension of next block preparation in sequences with movement.
+/-	+ : Lag with respect to the start of the current block. - : Lead with respect to the end of execution of the current block.
X.. / F..	X.. : Lag or lead in millimetres. F.. : Lag or lead in seconds.

Properties of the function

Function G79 is nonmodal.

Cancellation

Function G79 is cancelled at the end of the block.

Restart of processing after programming of the function

As soon as analysis of the next block is restarted, it should be noted that:

- in case of possible processing by parametric programming, the current block is considered ended, thereby authorizing read or write of external parameters E without effectively stopping movements,
- as soon as the following block is ready, the feed rate at the end of the current block is updated,
- this allows write of external parameters E and discrete outputs or read and test of these parameters without stopping movements.

Examples

Timeout at the start of the current block

```
N..
N110 G00 X0 Y0
N120 G01 X100 G79+F4
N130 G79 E10000=1 N300
N140 Y20
N..
N300 G01 X200
N..
```

Movement along X with four second delay before analysis of block N130

Anticipation at the end of the current block

```
N..
N110 G00 X0 Y0
N120 G01 Y100 G79-X10
N130 ...
```

Movement along Y with anticipation of 10 mm before analysis of block N130

5.15.7 Calling a subroutine before executing a block

When the subroutine number is between %11002 and %11999, the subroutine is called **before execution** of the block containing the M function and block concatenation is forced (G999 status). It should noted that within the subroutine, execution of the functions is enabled by G998 and/or G997 as in subroutine calls by Gxxx function.

Example:

Execution of **spindle indexing during axis movement** in a block containing the programming: N.. X.. Y.. Z.. M19.



For further explanation on block concatenation and G997 / G998 please refer to Flexium+ Extended Programming manual M00055.

```
%11019 (SPINDLE INDEXING DURING MOVEMENT)
VAR     [br_pos] = [.BM62] *2+ [.BM65] *2+ [.BM64] - [.BM63] / 2&3+93524
        [M998] = [.BM999] - [.BM997] +998
        [tix(9)] [i]
ENDV
M997 (forced block sequencing)
FOR [i] = 1 TO 9 DO [tix(i)] = [.IBX(i)]
   BCLR [.IBX(i)] (stop movement)
ENDF
G997 (exit M19)
G999
FOR [i] = 1 TO 9 DO
   IF [tix(i)] = 1 THEN
        BSET [.IBX(i)] (enable movement)
   ENDI
ENDF
G997 (start movement)
(wait until spindle is in position)
WHILE E [br_pos] = 0 DO G4 F.1
ENDW
M[M998]
```

5.15.8 Temporary inhibition of MDI access ► M999, M998

The purpose of M999 is to authorize writing into variables L100 to L199, L900 to L999 and symbolic variables without holding block preparation.

In M998 status each attempt to write into these variables will result in stopping block preparation and therefore preventing cutter compensation.

Detailed information can be found in the Flexium+ Extended Programming manual M00055.



M999	Programmed inhibition of the manual data input (MDI) mode and subroutine calls by the PLC.

As long as this function is active the operator cannot call the MDI mode and the PLC cannot call subroutines.

M998	Reactivation of the manual data input (MDI) mode and subroutine calls by the PLC.

Syntax

N.. **M998/M999**

M999	Programmed cancellation of the manual data input (MDI) mode and subroutine calls by the PLC function.
M998	Reactivation of the manual data input (MDI) mode and subroutine calls by the automatic control function.

Properties of the Functions

Functions M998 and M999 are modal "pre" functions. Function M998 is the default function.

Cancellation

Functions M998 and M999 cancel one another (M998 and M999 can also be cancelled by M997 and M02).

M999 is cancelled by a reset.

Notes

Programming M999 allows:
- manual movements (JOG or MANUAL),
- use of enhanced feed by handwheel (G12),
- use of stop on switch (G10).

Programming M999 inhibits:
- switch to sequence number search (SEARCH) mode during execution of a sequence of blocks "masked" by programming M999 (ignored by the NC),
- action by the PLC or the operator during execution of a sequence of blocks.

Example

Use of variables L

In program %1, block N100 is not prepared and executed until block N90 is completed.

In program %2, block N100 is prepared before execution of N90 and there is no stop at the end of block N90.

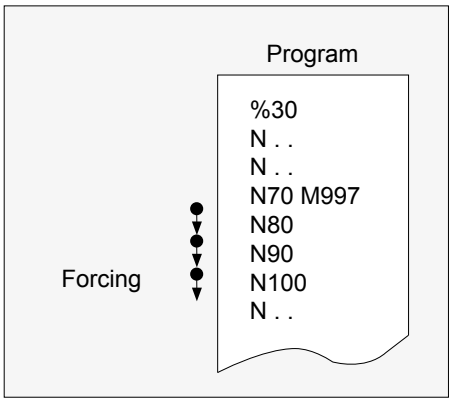
```
%1
N.. ...
N.. ...
N90 X.. Z..
N100 L110 = 1 + L110
N110 XL110
N120 L120 = L110 + L10
N130 ZL120
N..
N..
```

```
%2
N..
N80 M999
N90 X.. Z..
N100 L110 = 1 + L110
N110 XL110
N120 L120 = L110 + L10
N130 ZL120
N140 M998
N..
```

5.15.9 Force consecutive execution of blocks ► M997



M997	Force consecutive execution of blocks.



When this function is active continuous execution of program blocks is forced either in Single mode or MDI.

This is generally use to perform operation that should be made in sequence (e.g. Tool changer).

Syntax

N.. **M997**

M997	Forced block continuation.

Properties of the Function

Function M997 is a decoded modal «pre» function.

Cancellation

Function M997 is cancelled by functions M998, M999 and M02.

Notes

Use of function M997 in single step (SINGLE) mode

If the operator starts program execution in SINGLE mode, reading M997 during the program causes sequencing of the following blocks as though the system were in continuous mode.

Use of function M997 with a subroutine call by M function requested in manual data input (MDI) mode

Programming this function causes execution of the subroutine like in continuous mode.

5.15.10 Processing of blocks and programmed G and M functions ► G999, G998, G997



G999	Suspension of execution and forcing of block concatenation.

The blocks programmed after G999 are concatenated. The movements are no longer made on the axes and functions M, S and T are no longer processed.

These functions are used for creating custom cycles. They are not used in standard part programming and are explained more in details in the Flexium+ Extended Programming manual M00055.

G998	Enabling of execution of the blocks and part of the functions processed in state G999.

The blocks including movements and functions processed in state G999 are enabled and executed with the exception of certain functions that are only stored (decoded M "post" functions, dwells and synchronization functions for multiple channels).

G997	Enabling and execution of all the functions stored in state G999.

All the functions programmed (without exception) are enabled and executed, including those processed in state G999.

Syntax

N.. **G999** / **G998** / **G997**

G999	Suspends execution and forces block concatenation.
G998	Enables execution of the blocks and part of the functions processed in state G999.
G997	Enables and executes all the functions stored in state G999.

Properties of the Functions

Functions G999, G998 and G997 are modal.

Cancellation

Functions G999, G998 and G997 cancel one another.

Notes

Notes on Function G999

After execution has been suspended and the blocks have been concatenated by G999, analysis of the following blocks continues and the values encountered in these blocks of the part program are stored in special symbolic variables.

These variables can be used to perform computations or read programmed values, but the axes are not moved and the M, T and S functions are not processed.

A subroutine call by function Gxxx (see Flexium+ Extended Programming manual M00055) implicitly sets function G999. This function then has to be cancelled by including functions G998 and/or G997 in the subroutine.

During the return to the part program, state G999 is systematically reset as long as the subroutine call function (Gxxx) remains present and active (no G80).

Notes on function G998

Function G998 enables and executes:
- movements on the axes,
- auxiliary functions such as spindle speed (S), tool number (T), decoded «pre» M functions.

Function G998 stores the following functions but does not enable them:
- decoded «post» M functions (e.g. M05, M09, etc.),
- dwells (G04 F..),
- axis multichannel synchronization functions (G78 ...).

Example

Sequencing of functions G999, G998 and G997

```
%55
N.. ...
N.. G999 S2500 M03
N.. G00 X100 Y50 G04 F5
N.. G01 Z10
N.. ...
N.. M05
N.. G998


N.. ...
N.. ...
N.. X200 T02 M06
N.. ...
N.. G997
```

Blocks concatenated by G999

Enabling of S2500 M03 ("pre" M function) and movements on the axes to X100, Y50 and Z10 in G01 mode at F5

Movement to X200 and enabling of T02 M06

Enabling of G04 F4 (dwell) and M05 ("post" M function)

5.16 Dialog

5.16.1 Message transmission ► $0

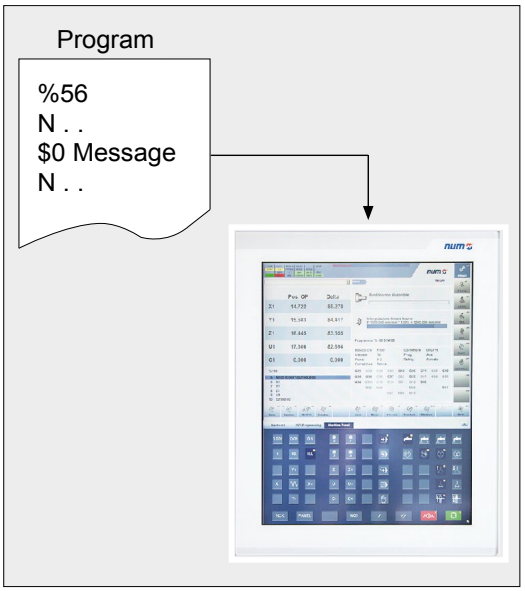
The $ character, optionally, followed by a zero sends a message from a part program to the display.

5.16.1.1 Message transmission to the display



$0	Message transmission to the display.

$0 sends the message to the system display (information message on the part program being executed).



Syntax

$0 MESSAGE or $+ Message complement

$0	Message sent to the display.
$+	The «+» sign extends the previous message.
MESSAGE	Message containing a maximum of 250 (alphanumeric) characters.

Cancellation

- $0 (without a message).
- End of program (M02).
- Reset.

Notes

The message sent by $0 can be read on the Messages area of HMI.

It should be noted that:

- The first 95 characters can be displayed it is however suggested to limit the string to 60 characters for a more harmonious visualization.
- Only one message can be sent at a time.
- Sending a new message overrides the previous one.
- The 0 after $ is optional, but it is recommended.

Multi language messages

Starting with version 4.1.10 it is possible to display messages in different languages:
The syntax is as follows:

$.	<line number> ;[Var1];...[Varn].
<Line number>	Refers to a line in an HMI file.
[Varx]	Refers to optional variables whose value will be included in the message.

As for $ messages extension of messages is possible.

Additional information is given in the Flexium+ Commissioning Manual M00044.

Extensions to the use of the $ character

The $ character can be used in parametric programming to:
- Display a message and wait for an answer from the operator.
- Display a message with a parametric value.

Examples

The message beginning in block N190 and including an extension is displayed in the following format:

«PROGRAM HALT. REMOVE CHIPS»

```
%30
$0 GROOVE ROUGHING FINISHING          Message
N10 ...
N..
N190 G00 Z300 $0 PROGRAM HALT.        Message
$+ REMOVE CHIPS                       Message extension
N200 M00
N.. $0                                Message cancellation
N..
```

Example

Use of $0 as flashing message

```
N..
N240 $0 * * * PREPARE PART INSPECTION * * *
N250 G04 F0.6
N260 $0 * * * <<<<<<<<<====>>>>>>>>> * * *
N270 G04 F0.3
N280 G77 N240 N270 S4
N290 M00 $0 * * * OPEN DOOR * * *
N300 $0
N..
```

5.16.2 Message display with request for an operator response ► L../E..,$



The «$» character can be used as variable in a parametric expression. It causes the program to wait for an operator entry.

The block containing the parametric expression can be preceded by a message indicating the type of data to be entered.

Syntax

L../E.. = Expression depending on **$**

L../E..	Program variable or external parameter.
$	Variable whose value must be entered by the operator.

Cancellation

- end of program (M02),
- reset.

Notes

Upon reading the «$» character associated with a variable (possibly accompanied by a message with a maximum of 39 characters), the system stops the program and waits for the operator to type an entry.

The characters typed are echoed on the upper right window of the HMI.

The entry can be numerical characters or one alphabetic character. Each type of entry corresponds to a specific use.

Numerical entry

The number of digits entered must not exceed eight plus sign and decimal point.

The limit values are as follows:
- Integer: ≤ 99999999 or ≤ -99999999
- Positive decimal: ≤ 0.0000001 or ≤ 9999999.9
- Negative decimal: ≤ -9999999.9 or ≤ -0.0000001.

The operator must confirm the entry by pressing the Enter key on the keyboard (otherwise the system does not proceed to the next block).



The entry must be validated by the Enter key. Before pressing this key editing the line is possible.

Alphabetic entry

Only the first character entered is analysed.

Any capital letter from A to Z is accepted. Each letter returns a value:
- A returns 1
- B returns 2

and so forth to Z = 26.

It should be noted that in a test program, each numerical value from 1 to 26 corresponds to alphabetic characters A to Z.

Example

Message Display

```
N.. ...
N50 E80000=50
$0 ENTER THE VALUE OF X =          Display of the message for the operator
N60 L0=E80000 + $                  Wait for entry of the value
N..
```

5.16.3 Display of messages with parametric value ► $



The «$» character can be used as the first term of a parametric expression whose value is displayed after being calculated by the system.

The block containing the parametric expression can be preceded by a message indicating the name of the variable displayed.

Syntax

$ = expression

$	Variable whose value is displayed.

Cancellation

- end of program (M02),
- reset.

Notes

An expression «$ =...» in a block after the branch function (G79) cannot be used as a branch condition.

Example

```
N..
L0=0                                  Initialization of variable L0
N90 L0=L0+1                           Increment of L0
$0 EXECUTION OF SUBROUTINE            Display of the first part of the message
$= L0                                 Display of the value following the previous message
G77 HL0                               Call to subroutine % (L0)
G79 L0<5 N90                          Condition: if L0 < 5, branch to block N90, else continue
N..
```

5.17 Storage of values

5.17.1 Transfer of the current values of L Variables and E Parameters into the part program ► G76



G76	Transfer of the current values of L variables, E parameters and symbolic variables into the specified program or program section.

This function is used to update the contents of a file called by addresses H.. and/or N.. N..

The file of L variables, E parameters and symbolic variables is updated with the new contents of the corresponding active data.

Syntax

N.. **G76** [H..] [N..][+number] [N..][+number]

G76	Transfer of the current values of the variables into the program specified.
H..	Specification of the program into which the values are transferred.
N.. N..	Specification of the program section into which the values are transferred.
+number	Specifies an offset to the line into which the values are transferred (after the block number specified).

Notes

The parameters to be transferred must be located at the beginning of blocks: variables and parameters located after another function in a block are ignored.

The specification of a variable or a parameter must be followed by the sign " = " and at least twenty two characters (spaces, algebraic sign, numbers, decimal point) designed to be replaced by a new value.

From revision 4.1.30 and up this space requirement doesn't exist any more. There is no need to reserve empty space in the file intended to receive the values

Failure to comply with this rule returns error message 97.

Obviously symbolic variables must exist before having the possibility to save them by G76.



The file modified with G76 is stored in NC volatile memory. It is saved only during a proper shut down of the system. Make sure the file has been properly saved before using the results.

Examples

Transfer of the values of L and E into the current program

In the example below, only parameters L101, E80001, L4 and L6 are modified. The other parameters are ignored.

```
N..
N40 G76 N100 N120
N..
N100 ..
L101=.......... E80001=..........
L4=.......... G04 E52002=..........   E52002 is not modified as it is located after G04
L6=..........
G01 X100 L3=..........                L3 is not modified
N120
N..
```

Transfer of the values of L and E into a subroutine

```
%125
N10 G77 H200 N50 N80                  Execution of blocks N50 to N80 of %200
N..
N..                                   Execution of the program after modification of the
                                      parameters by %200
N..
N600 G76 H200 N50 N80                 Update of the file
N610 M02

%200
N10 ...
N..
N50
L1=.......... E52002=..........
E80004=..........
N80
N..
```

Evolution since 4.1.30

```
%100
L0=1235.335 L1=885.63
G76 H1000 N100 N200
...
Before 4.1.30
%1000
N100 L0 = 123456789012345678901 2 L1 = 12345678901234567890 12 (Test)
N200

Result:
N100
L0 = 1235.335                          L1 = 885.63                          (Test)
N200

After 4.1.30
N100 L0 = 0 L1 = 0 (Test)
N200


Result:
N100
L0 = 1235.335 L1 = 885.63 (Test)
N200
```

5.17.2 Format of the numerical values

The format is introduced by the ***syntax FPF w.p***.

FPFw.p is **modal** when it is programmed ***outside G76+ block***.

When FPFw.p is programmed ***inside*** G76+ block, the w.p ***are local*** to this block. The outside values of w.p are ignored.

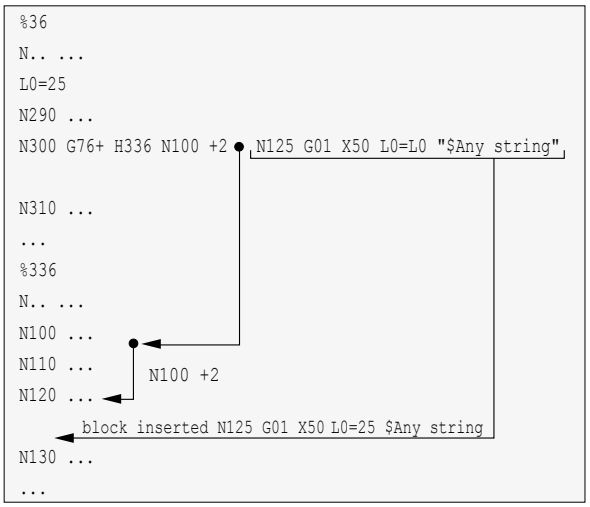
Several programming variants are possible:

N10 **FPFw.p**	-
N20 G76+ Hnn N.. E.. L... X.. A..	All floating values are output with the same w.p **modal** format.
N30 G76+ Hnn N... **FPFn.m** E.. L... X.. A..	All floating values are output with the same n.m **local** format.
N40 G76+ Hnn N... **FPFn1.m1** E.. L... **FPFn2.m2** X.. A.. **FPFn3.m3** I.. J...	E.. and L.. are output with the same n1.m1 **local** format. X... A... are are output with the same n2.m2 **local** format. I, J are are output with the same n3.m3 **local** format.
N50 G76+ Hnn N.. E.. L... **FPFn.m** X... A...	E.. and L.. are are output with the same **w.p modal** format defined in **N10 block**. X... A... are output with the same **n.m local** format.
N50 G76+ Hnn N.. E.. L... X... A...	All floating values are output with the same **w.p modal** format defined in **N10 block**, because no local defined format.

FPF 0.0 : standard behaviour.

Example

Block insertion in program %336 located in the RAM program area (area zero).



5.17.3 Saving symbolic variables in a dedicated memory zone ► G76+ S, G76- S



Syntax

G76+ S** [symbolic variable] ... [symbolic variable n]

G76+ symbolic variables	Save symbolic variables in a predefined memory location.
G76- S	Release a particular zone of the variables already saved inside.
Argument S	Argument S (values 1 to 32) is used to give the number of memory segment

This function is used to save symbolic variables in a predefined memory location.

It is reserved for use by dynamic operators and is further explained in the Flexium+ Extended Programming manual M00055.

The associated function G76- S.. is used to release a particular zone of the variables already saved inside.

5.17.4 Restoring symbolic variables from dedicated memory zone ► G76+ [Sym 1] ... [Sym n] S**



Syntax

G76+ [symbolic variable 1] ... [symbolic variable n] S**

G76+ symbolic variables S**	Restore symbolic variables from a predefined memory location.
Argument S	Argument S (values 1 to 32) is used to give the number of memory segment

This function is used to restore symbolic variables from a predefined memory location. It is the reciprocal function of G76+ S** [Sym].

It is reserved for use by dynamic operators and is further explained in the Flexium+ Extended Programming manual M00055.

5.17.5 ISO program or block creation/deletion ► G76+/-



G76+/- ISO program or block creation/deletion.

General

Programming of function G76+/- provides the following capabilities:
- Program creation,
- Program deletion,
- Block insertion in a program,
- Block deletion from a program.

This function is especially useful for creating part programs automatically (e.g. after teach-in by gauging of dimensions).

This function can only be used to create and delete programs and blocks located in part program area zero.

Before execution of a function G76+ (creation, insertion), the system makes sure no other program is active. If so, program execution is suspended as long as the other program remains active.

With G76, the program number can be indexed (e.g. H123.2). This is accepted providing no other program in any of the program areas has the same number.

If there is not enough memory space for program creation or block insertion, the system returns error message 266.

If the block including function G76 has a format error, the system returns error message 1 with an extension from 49 to 62 further explaining the error.

Properties of the Function

Function G76 is nonmodal and is cancelled at the end of the block.

Program created or edited via G76 are stored in dynamic RAM.
They can be saved either by transfer to the panel or by a proper shutdown of the CNC system. If the system is powered off abruptly and the file has not been transferred the content is lost.

5.17.6 Checking the existence of a file ► G76=H

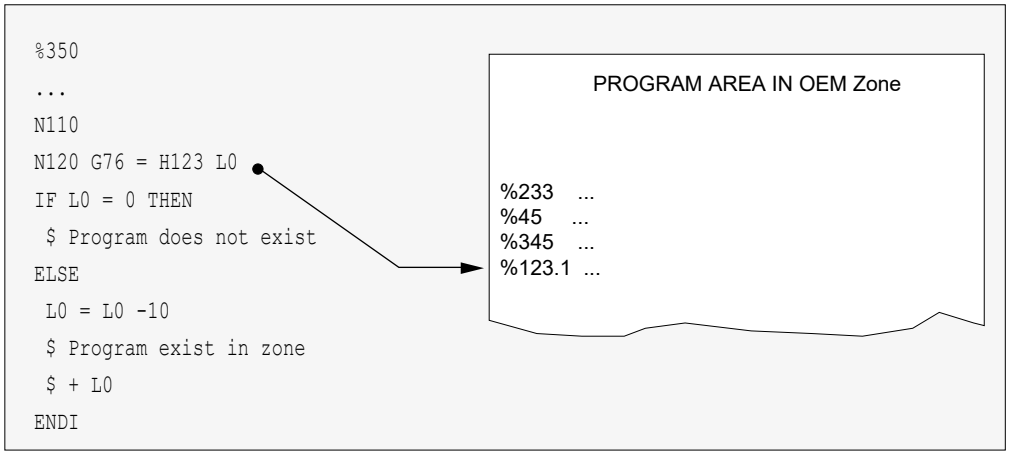
G76=H	Checking the existence of a file.

Before creating, deleting or inserting a block it might be necessary to check if a file already exists.

Syntax

N.. **G76** = H.. L..|E..

G76	Checking the existence of a file.
H	Represents the file searched.
L.. or E..	Result of the search. = 0 : The file does not exist. If L0 = 10 then file exists in zone User. If L0 = 11 then file exists in zone Customer. If L0 = 12 then file exists in zone OEM. If L0 = 13 then file exists in zone NUM.



Another syntax exists since 4.1.20 : SEARCH Ha:p Nb
This search the existence of program %a in memory zone p (optional).
If the program exist then the calling program will jump to bloc number Nb if the searched program does not exist then execution continues in sequence.

5.17.7 Creating a program ► G76+ H

The syntax below is used to create a program in area zero.



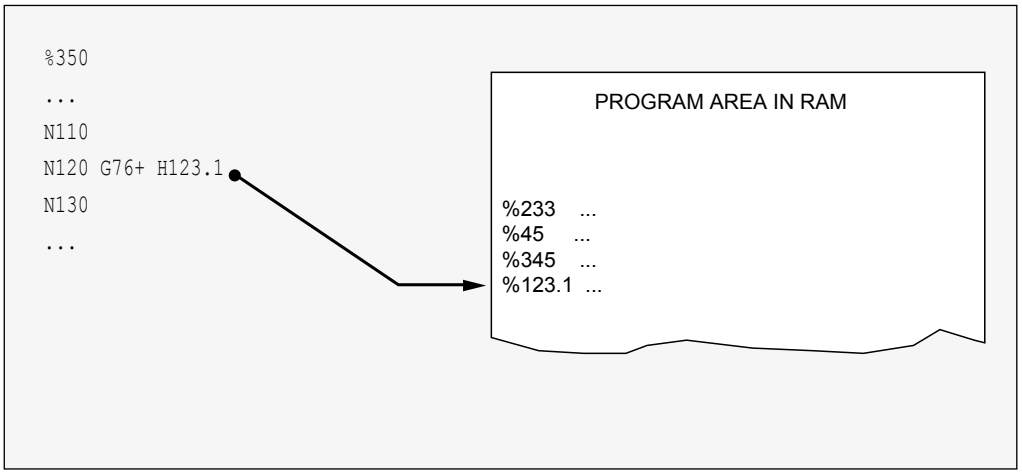
Syntax

N.. **G76+** H..

G76+	Program creation function.
H..	Number of the program to be created.

Example

Program creation in User memory zone.



5.17.8 Deleting a program ► G76- H

The syntax below is used to delete a program located in area zero.

Syntax

N.. **G76-** H..

G76-	Program deletion function.
H..	Number of the program to be deleted.

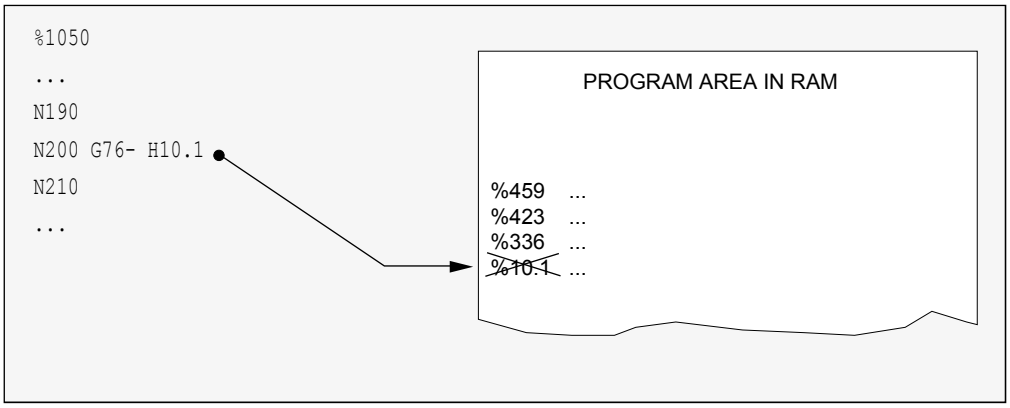
Notes

The program number must be the last word in the block.

Several cases can occur, depending on the area where the program is located:
- If a program in area zero has the number specified in the function, this program is deleted,
- If a program in an area other than area zero has the number specified in the function, deletion is refused and the system returns error message 266,
- If no program in any area has the number specified with the function, the command is accepted (but no program is deleted).

Example

Program deletion in RAM program area (area zero).



5.17.9 Inserting a block ► G76+

The syntax below is used to insert a block in an existing program.



Syntax

N.. **G76+** [H..] N..[+number] ISO block

G76+	Function defining block insertion.
H..	Specifies the program number where the block is to be inserted (optional: default H..: the ISO block is inserted in the program including function G76).
FPF w.p	Format of the numeric values inserted. w = width p = precision.
N.. +number	N..: Block number (mandatory). The insertion is made after this block unless "+number" is specified (note: the first block of program %.. is block N0). +number: (optional) defines the number of lines (after the block number specified) where the insertion is to be made.
ISO block	Block to be inserted consisting of ISO functions (see list under Notes).
A string in double quotes	Can be a comment or ISO functions as well.

Notes

The following ISO functions are accepted:
- N.. (block number),
- G.. (preparatory functions; several G functions are accepted in the block),
- X.. Y.. Z.. U.. V.. W.. A.. B.. C.. (axes and dimensions),
- I.. J.. K.. (circle center coordinates),
- P.. Q.. R.. (material vector for 3D radius correction),
- /.. (polynomial coefficients),
- L=.. to L99=.. (program variables).

The values associated with the above functions are immediate integer or fractional values, program variables (L..), symbolic variables [sym] or parameters E. When they are parameters E, the values are specified in the internal system unit (these data are edited in the program as immediate values, signed if negative, and in the internal unit defined for linear or rotary axes).

The block to be inserted must not have more than 120 characters or the system returns error message 1. If there is not enough memory space for block insertion, the system returns error message 266. If the block to be inserted does not exist, the system returns error message 25.

5.17.10 Deleting a block ► G76-

Syntax

N.. **G76-** [H..] N..[+number]

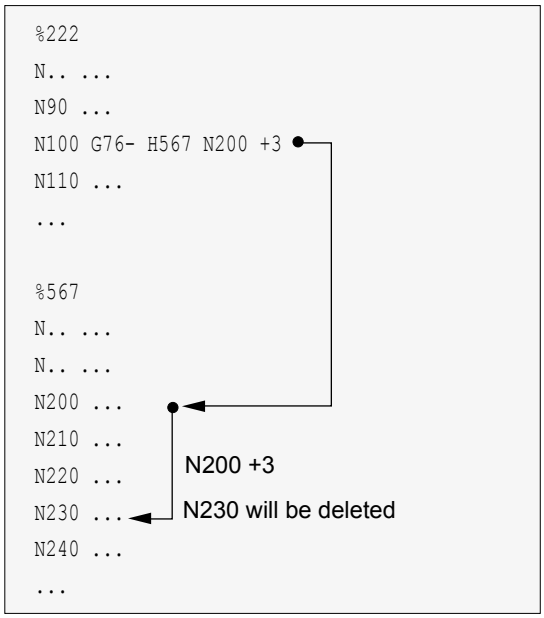
G76-	Block deletion function.
H..	Specifies the program number where the block is to be deleted (optional: default H..: the ISO block is deleted in the program including function G76).
N.. +number	N..: Block number (mandatory). It is this block that is deleted unless +number is programmed (note: the first block of program %.. is block N0). +number: (optional) specifies an offset to the line to be deleted (after the block number specified).

Notes

If the block to be deleted does not exist, the system returns error message 25.

Example

Deletion of a block in program %567 located in the RAM program area (area zero).

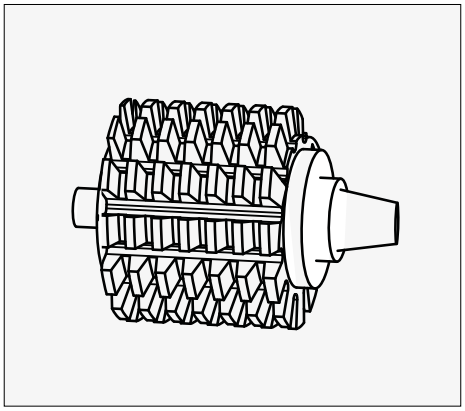


5.18 Electronic Gear box

5.18.1 MLEGB Spindle continuous rotation ► G770



G770	MLEGB Spindle Continuous rotation.



This function is intended to continuously run a rotary axis (like a spindle). It is mostly intended for use with MLEGB (Multi Level Electronic Gear Box) and is described in the Flexium+ Extended Programming Manual M00055.

Like a spindle a Cycle Stop does not stop the rotation

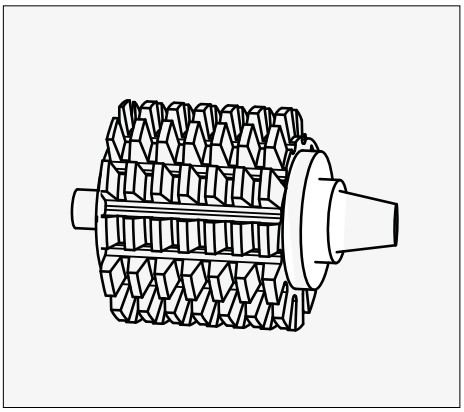
Syntax

G770 A/B/C [M3/4 S…| M5]

5.18.2 MLEGB Synchro Definition / Cancellation ► G771



G771	MLEGB Synchro Definition/ Cancellation.



This function is reserved for use with the MLEGB function. It is intended to define the ratio between the rotation of a master and a slave rotary axis.

They are several possible syntaxes. The complete description is in the Flexium+ Extended Programming Manual M00055.

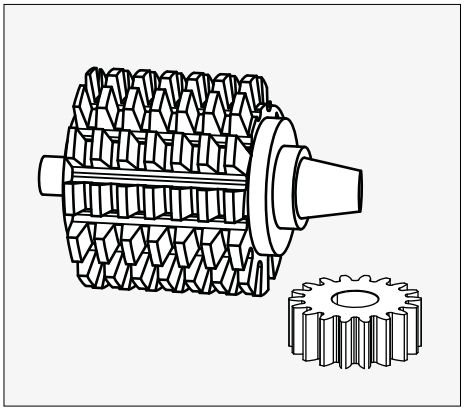
Syntax

G771+ A/B/C C/B/A P.. Q..

5.18.3 MLEGB Activation / Deactivation ► G775



G775	MLEGB Activation / Deactivation.



This function is reserved for use with the MLEGB function. It is intended to activate the synchronization of the slave axis to its master previously defined by G771.

The deactivation is performed by the command G775- C.

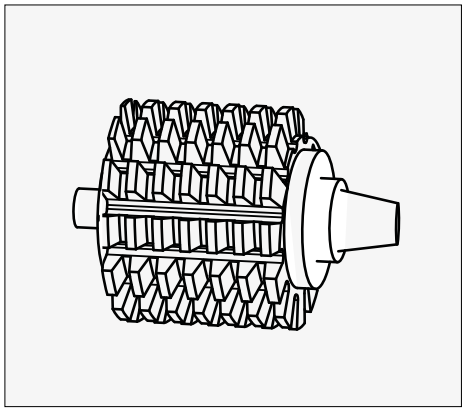
Like other MLEGB functions they are several possible syntaxes. The complete description is in the Flexium+ Extended Programming Manual M00055.

Syntax

G775+ A/B/C

5.18.4 MLEGB Waiting for synchronization completion before proceeding ► G776

G776	MLEGB Waiting for synchronization completion before proceeding.



This function is reserved for use with the MLEGB function. It is intended to wait before complete synchronization before resuming program execution.

The F value introduces a time-out.

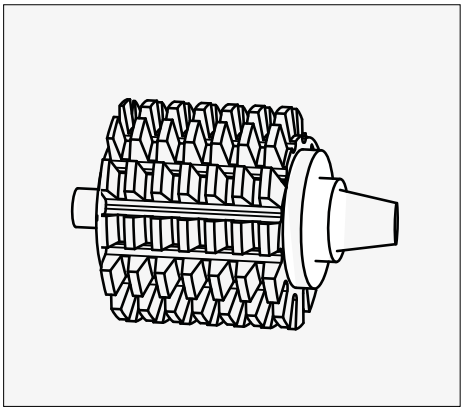
Like other MLEGB functions they are several possible syntaxes. The complete description is in the Flexium+ Extended Programming Manual M00055.

Syntax

G776 A/B/C F..

5.18.5 MLEGB Synchro Homing / Gear alignment ► G779

G779	MLEGB Synchro Homing/Gear alignment.



This function is reserved for use with the MLEGB function. It is intended to define the start position of all axes involved in the synchronization for a perfect gear coupling.

Like other MLEGB functions they are several possible syntaxes. The complete description is in the Flexium+ Extended Programming Manual M00055.

Syntax

G779+ C… B.. X… Z...

5.19 Special programming for Multiple Channels

5.19.1 Program declaration



A machining program for multiple channels is an association of the programs for each axis channel.

The program names have a common radical followed by an index specifying the channel, e.g.:

- %61.1 Program number for channel 1
- %61.2 Program number for channel 2

Program number format: %05.1 (indices .1 to .8).

The programs for each of the channels must be loaded in the NC memory. Otherwise, the machining program cannot be started (unless one or more channels are inhibited).

5.19.2 Programming notes

It is essential not to use the program start symbol «%» in comments in the program body.

Programmable Axes

The NC can control 32 axes in a maximum of 8 channels.

Each channel can include up to 9 axes declared by the machine manufacturer.

Example: Machine with 9 axes in 3 channels

- Channel 1: axes X Y Z A
- Channel 2: axes U V W B
- Channel 3: axis C

Axes in different channels can have the same name. In the above example, axes U V and W of channel 2 could have been named X, Y and Z.

For part and program set-up it is possible to selectively inhibit a particular channel. Please refer to the manufacturer's manual for details.

Tool Corrections

The tool correction tables are common to all the programs.

Program Variables

L variables can be used in each program.

L0 to L99, L100 to L199 and L900 to L959: these variables can be used for each axis channel (no interaction between programs).

External Parameters

External parameters E are common to all the programs (except for parameters E50000, E51000, E6x000 and E7x000 which relate to their individual axis channel).

Subroutine branches for Multiple Channels

Subroutines which are not part of the machining program must always include the index of the calling channel. For instance: %xxx.i (i = channel number).

Branch to Automatic Homing Subroutine

Homing can be carried out automatically on each of the axes of the channel by running subroutine %9990.i (i = channel number).

Requirements for running subroutine %9990.i

Action on the cycle start button in homing mode runs subroutine %9990.i for the channel if no other program is being executed.

The automatic homing program is run simultaneously on all the existing CNC channels.

Subroutine %9990.i can also be called as a subroutine from another program while preserving the property of being able to move the machine axes before being homed.

This possibility can be used for homing an Auxiliary channel. If such a program can be used in homing mode or as a subroutine, it can also end as follows:

```
IF [.IRH(1)] = 9990.i THEN M02          If it is not from a subroutine, then set M02
ENDI
```

For information on the above programming, refer to the Flexium+ Extended Programming manual M00055.

Subroutine Call by a Reset

On a reset, the NC channels can execute subroutines with numbers %11000.i (i = channel number).

Aside from the fact that the subroutine numbers include the index (i) of each NC channel, the call is the same as the subroutine call by reset for a single channel.

Subroutine Call by the PLC

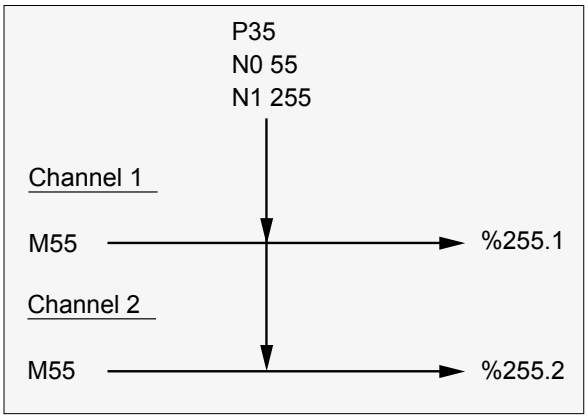
The automatic control function can call and execute a subroutine with number %9999.i (i = channel number).

The call is the same as a subroutine call with a single channel except for the CNC channel number (i).

Subroutine Call by M Function

Example

With two channels (1 and 2), call of two subroutines %255.1 and %255.2 with different contents by function M55.



5.19.3 Channel synchronization with markers ► G78



G78	Channel synchronization with markers.

This function identifies and manages the synchronization steps in the execution of each program. It is obviously meaningful in multi channels systems in common mode.

Syntax

N.. **G78** Q.. Pj.i

G78	Channel synchronization.
Q..	Declaration of a marker in the current channel.
Pj.i	Wait for a marker from another channel. (optional). Argument P is defined by two digits separated a decimal point: - j is the marker number tested, - i is the index of the channel whose marker is tested.

The P argument can be repeated if necessary for other channels.

Example: `G78 Q5 P3.2 P5.3 P2.4`

Defines marker 5 for the current channel, waits for marker 3 passed on channel 2, marker 5 passed on channel 3 and marker 2 on channel 4.

Property of the Function

Function G78 is nonmodal.

Cancellation

Function G78 is cancelled at the end of the block.

Argument Q associated with the function has a default value of zero (Q0).

During the program, the markers can be reset by programming G78 Q0.

Notes

Function G78 can be followed by several arguments, but at least one is mandatory.

The declaration of a marker and the program continuation conditions can be programmed in the same block, e.g.:

```
N.. G78 Q3 P5.2 P6.3
```

The channel selected and the associated step transition conditions can be enabled selectively by the automatic control function.

The markers included in a program must be numbered in increasing order, but not necessarily sequentially, e.g.:

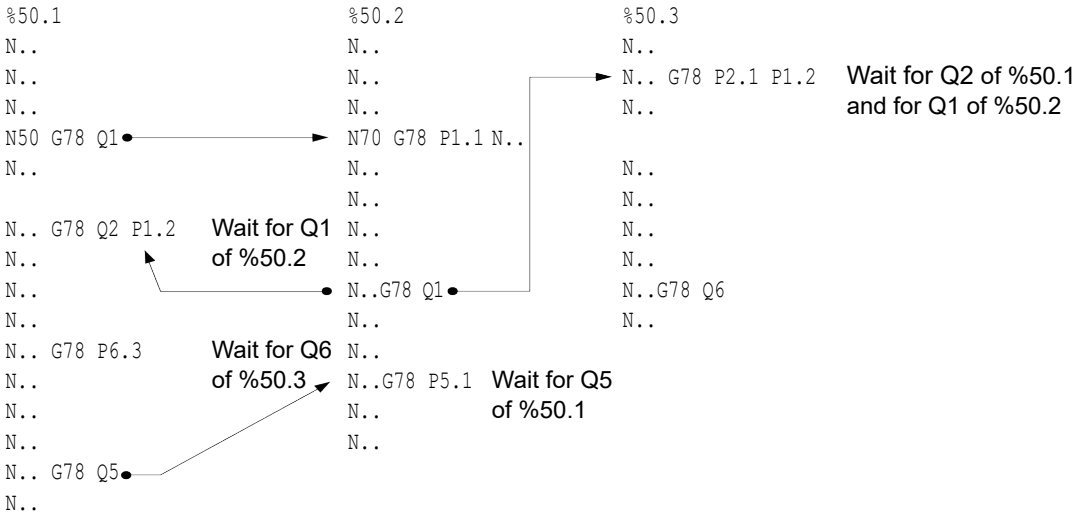
```
N.. ...
N90 G78 Q1 ...
N..
N200 G78 Q6 ...
N..
```



Channel markers are set as their programs reach them.

Example

Program synchronization with markers



Sequence N70 of program %50.2 is crossed if program %50.1 reaches or has passed sequence N50.

Program %50.3 does not start until program %50.1 reaches marker Q2 and program %50.2 reaches or passes marker Q1.

If channel 3 is inhibited by the PLC program, the steps related to program %50.3 are ignored in programs %50.1 and %50.2 (wait G78 P6.3 of %50.1 is ignored).

5.19.3.1 Programming a meeting point

A meeting point can be programmed by function G78 or by programming the M0, M1 or M12 functions.

5.19.3.2 Programming by function G78

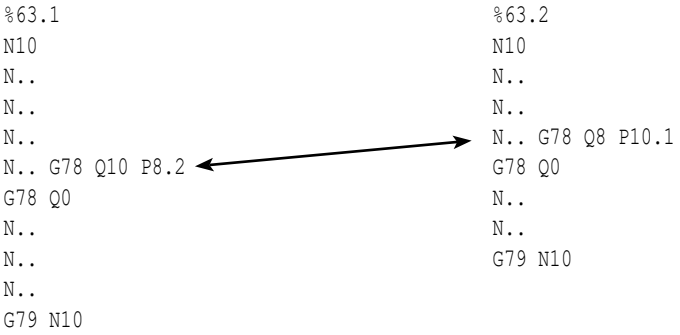
When a meeting point has been programmed, the programs do not continue until the other programs have reached their respective markers.

When the meeting point has been reached by all the channels, the markers can be reset by programming G78 Q0.

Before the markers are reset by G78 Q0, the previous block must necessarily be a meeting point, i.e. a complete synchronization on all the programs.

Example:

Programming a meeting point on two channels (with complete synchronization).

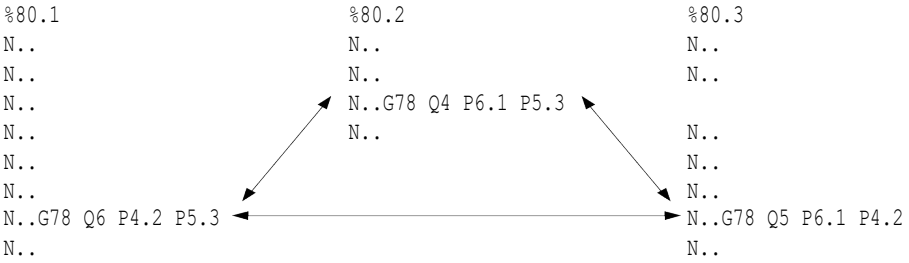


The end of program command (M02) cancels the conditions of that channel and is equivalent to inhibiting the program.

If all the programs are on wait, the system returns error message 33 (synchronization error). If this error is detected in test mode (TEST) or sequence number search mode (SEARCH), it is necessary to carry out a reset and to correct the program.

Example

Programming a meeting point with G78 on 3 channels



The three programs continue together when markers Q6, Q4 and Q5 are reached.

5.19.3.3 Programming with M functions

The following functions are used as meeting points:
- programmed feed stop with intervention (M12),
- program stop (M00),
- optional stop (M01).

When M00 or M01 (enabled) are programmed in only one of the programs, the movements of the axes in the channel are stopped and the system waits for the end of the other programs (M02) before executing M00. When the cycle restarts, only this program continues.

It should also be noted that functions M00 and M01 (even not enabled) reset the markers.

When M01 is programmed in all the programs but was not enabled, each of the M01 sequences is a meeting point, but the cycle continues automatically when all the programs are on their respective M01 commands.

5.20 Special programming of Auxiliary channels

5.20.1 Program declaration and storage

The programs executed by the auxiliary channels are identified by the radical 9998 followed by an index defining the channel, e.g.:



- %9998.2 Program executed by auxiliary channel 2.
- %9998.3 Program executed by auxiliary channel 3.

Program number format: %9998.n : program executed by auxiliary channel (n=2 to 8) assuming the channel n (and above) are declared as auxiliary channels.

The programs and subroutines must be stored in a CNC memory zone different of the User Zone (Customer, OEM or NUM zone).

When the main program is stored after creation or replaced after editing, a reset on this channel.

Structure of an auxiliary channel program

When different functions should be processed by an auxiliary channel, the PLC must specify the function called, which can be done by external parameter E40000, for instance.
At the start of the program:

- an M function is used as report
- external parameter E40000 contains the sequence number of each function requested (Na, Nb, etc.).

Example

Program %9998.2 (channel 2 is declared as auxiliary) below contains several functions.
The second program block defines a jump to the sequence number contained in parameter E40000.

```
%9998.2
N1 M..                          Wait for start by AckMFunction
G79 NE40000                     Jump to the sequence defined

Na                              Process the first function
..
..
..
G79 N1                          Jump to the sequence

Nb                              Process the second function
..
..
..
G79 N1                          Jump to the sequence
..
```



By default auxiliary channels can not read or access the User program zone (zone 0). From version 4.2.0 a machine parameter P7N4.6 permits this operation.

5.20.2 Programming of the Auxiliary channels

The auxiliary channels are programmed in the same way as the NC channels but there are restrictions concerning the use of certain functions.

Restrictions

The following functions are unavailable:
- optional stop (M01),
- programmed feed stop (M12).

5.20.2.1 Emergency retraction on an Auxiliary Channel

Emergency retraction (function G75...) on an auxiliary channel allows a particular sequence to be executed in automatic mode at the request of the PLC when the same channel is in a cycle.
It should be noted that:
- No particular condition is required as to the current mode on the other channels
- The emergency retraction request on an auxiliary channel only affects that channel.

Activation of emergency retraction

Function G75 N.. defines a branch to a sequence N.. when emergency retraction is activated.
Activation is cancelled by declaring a new address G75 N.. or G75 N00.
The emergency retraction is activated from the PLC either automatically or on a manual action.
For further information please refer to the Machine Operation Manual.

It should be noted that:
- If the channel is executing a cycle (automatic mode "AUTO" or single step mode "SINGLE"), activation halts the program being executed. The halt is followed by a branch to sequence N.. programmed after G75 followed by execution of the retraction program until one of functions M00 or M02 is encountered.
- Signal *Flexium_NCK.RCNC.Channel[0].EmergencyRetract* remains set throughout the duration of the emergency retraction program. This signal is reset by encountering function M00 or M02 which ends the emergency retraction program.



If an emergency retraction is activated but no G75 N.. was programmed, this activation stops the axes of the channel and resets the channel (RAZ).

5.20.2.2 Exchanging axes between channels

The channels can exchange axes via the part program. The axes are exchanged using external parameters E7x005 (x = axis number).

After a reset, the channel (PLC or NC) releases the axes that were not assigned to it by machine parameter and assigns them to the priority channel providing the priority channel is:
- in the end of program state (M02), or
- with no program being executed for an auxiliary channel.

Similarly, during a reset concerning it, a channel can recover the axes released by other channels that are assigned to it by machine parameter:
- after a reset,
- by programming external parameter E7x005.

It should be noted that if a channel is not in state M02 when its axes are released, it cannot recover them until the next reset concerning it, i.e.:
- an NC reset for an NC channel,
- a reset on the channel for an auxiliary channel.



When the cycle starts, no checks are made on assignment of the axes to the channel. All the axes assigned to the channel after the last reset were assigned by the machine builder, but certain axes assigned to it after the first initialization may not be reassigned if they have already been assigned to another channel during a cycle (it is therefore up to the programmer where required to ensure the configuration of the axes in the channel in order to authorize start of the cycle).

5.21 Special features of mixed Machines (MX)



Machines combining standard milling functions with turning functions are called mixed machines.

Such mixed machines involve:
- Special features related to the machine axes,
- Special programming of certain ISO functions.

Below, the descriptions are limited to the following types of machines:
- borers,
- mixed machines.

5.21.1 Features related to the Machine Axes

Programmable axes on MX:
- Primary axes X, Y and Z,
- Secondary axes U, V and W,
- Rotary axes A, B and C.

Axis Declaration

Boring type machines

For such machines, radial axis U must be declared independent.

Borers or mixed machines

On such machines, the Z and W axes can be declared as carried or independent.



When Z and W are declared carried, the tool length correction is applied to only one of these two axes.

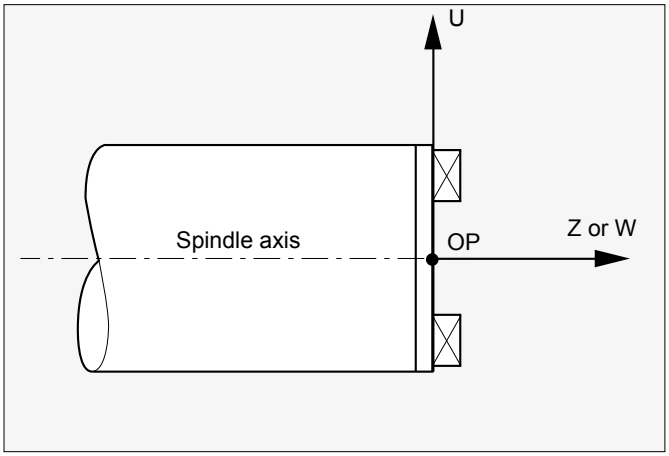
X and U Axes with Reference to Diameter or Radius

The X and U axes can be declared programmed with reference to the diameter or radius.

Programming the X and U axes with reference to diameter is used only when programming function G20 defining the ZX interpolation plane for turning.

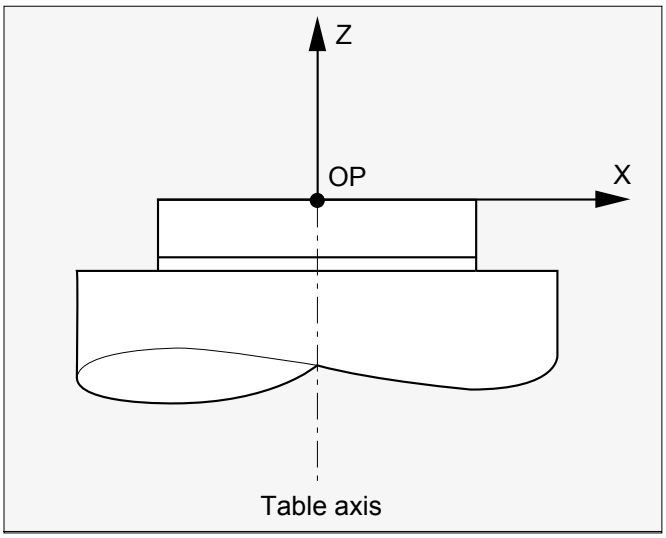
5.21.1.1 Boring machine

The program origin (OP) for the U axis is defined in the spindle axis. Homing on the U axis is carried out in the same way as on the other axes (with DAT1 and DAT2).



5.21.1.2 Mixed type machine

The program origin (OP) for the X axis is defined in the table axis.



5.21.2 Assignment of tool dimensions to the Axes

Types of Tool Dimensions Used on Mixed Machines

- Type 0: Milling tool (dimensions D.. L.. R.. @..)
- Type 1: Turning tool (dimensions D.. X.. Z.. C..)
- Type 2: Boring tool (dimensions D.. U.. Z.. C..)

Tool Type Selection

The tool type (0, 1 or 2) is determined when entering (or loading) the tool dimensions from the keyboard, a peripheral or by programming (for additional information, refer to the Flexium+ HMI Operator manual M00039).

The selection of the correction type requires declaring two mandatory addresses for each tool dimension type:
- L.. and R.. for type 0 (milling)
- X.. and Z.. for type 1 (turning)
- U.. and Z.. for type 2 (boring).

Notes on tool dimension assignment

Milling

The tool length L is assigned to the tool axis specified by tool orientation function G16. The tool axis must be a primary axis or a carried secondary axis. In no case can it be an independent secondary axis.

The tool radius R for radius offset (G41 or G42) is assigned to the two axes of the interpolation plane (G17, G18 or G19), be they primary, secondary, carried or independent.

Turning

The X and Z dimensions are assigned to axes X and Z or U and W if they are declared as carried (see machine parameter P64).

The tool radius R for radius offset (G41 or G42) is assigned to the two axes of the interpolation plane (G20), be they primary, secondary, carried or independent.

Boring

The U dimension is assigned only to the U axis.

The Z dimension is assigned to the Z axis or to the W axis if it is declared as carried.

The tool radius R for radius offset (G41 or G42) is assigned to the two axes of the interpolation plane (G20), be they primary, secondary, carried or independent.

Requirements for taking into account the tool dimensions

Type 0 (milling)

The tool L, R and @ dimensions are taken into account when the type 0 correction number Dxxx and the axes concerned are programmed. The interpolation plane is G17, G18 or G19.

Type 1 (turning)

The tool X, Z and R dimensions are taken into account when the type 1 correction number Dxxx and the axes concerned are programmed. The tool nose orientation C is taken into account on the X and Z axes. The interpolation plane is G20.

Type 2 (boring)

The tool U, Z and R dimensions are taken into account when the type 2 correction number Dxxx and the axes concerned are programmed. The tool nose orientation C is taken into account on the U and Z axes. The interpolation plane is G20.



For type 1 tools (turning) or type 2 tools (boring), the tool length correction is not taken into account on the Y and V axes.

5.21.3 Preselection of the program Origin (with function G92)

Function G92 includes an extension dedicated to boring and mixed type machines.

The axes and values programmed after the function define the current position of the tool with respect to the new program origin. DAT1 is computed according to the current tool length L; taking into account of L is therefore defined by the tool type: 0, 1 or 2.

Type 0 (milling)

L is taken into account according to the tool axis orientation function (G16 P, G16 Q or G16 R) on:

- the main axis if it is independent
- one of the axes if they are carrier/carried axis pairs.

Type 1 (turning)

L is taken into account on the X axis (or U if X and U are carried) and the Z axis (or W if Z and W are carried).

Type 2 (boring)

L is taken into account on axes U and Z (or W if Z and W are carried).

Error Messages

If the tool type is incompatible with the interpolation plane, the system returns error message 77:

- Tool type 0 (milling) with turning plane G20, G21 or G22,
- Tool type 1 (turning) or type 2 (boring) with milling plane G17, G18 or G19.X

5.21.4 Notes on programming the ISO Functions

Certain ISO functions used for milling and turning have special features, generally related to the machine axes declared. The other functions are used in the standard way.

Milling Function

G17, G18, G19: Interpolation plane selection

Possibility of various machining and contouring operations in the three main planes involving all the primary axes (X, Y, Z), secondary axes (U, V, W) and rotary axes (A, B, C).

Milling cycles

All the standard milling cycles can be used in the three interpolation planes.

Turning Functions

G20, G21, G22: Interpolation plane selection

Possibility of transforming the ZX coordinate system (G20) by programming functions G21 and G22 with the rotary C axis (do not confuse the spindle with the rotary C axis). Programming of G20, G21 or G22 sets function G18 (ZX plane for milling).

Turning cycles

The following cycles can only be used in the ZX plane (G20):

- G63: Roughing cycle with groove.
- G64: Turn/face roughing cycle.
- G65: Groove roughing cycle.
- G66: Plunging cycle.
- G33: Constant lead thread cutting. Possible on X or U or Z or W axes.
- G38: Sequenced thread cutting. Possible on X or U or Z or W axes.

G94 or G95: Feed rate

Possible feed rate in mm/min (G94) or mm/rev (G95) on all the linear axes (X, Y, Z, U, V, W) and selection of the measured spindle in G95 (see M66 to M69).

G96: Constant surface speed in m/min

Possibility of defining the speed of one of the spindles (see M66 to M69) according to the position of the X or U axis. This function is only active in planes G18 (milling) or G20 (turning).

The choice of the X or U axis is related to the type of correction selected (1 or 2).

Page deliberately empty

Summary

6 Profile Geometry Programming 463

6.1 Profile Geometry 463
6.1.1 Definition of the addresses characterizing PGP 464
6.1.1.1 Addresses with values ► X../Y.. , Z.., EA, X.. Y.., I.. , K.., J.., R.., EB+, EB- 464
6.1.1.2 Addresses without values ► ET, ES, E+/E- 467
6.1.2 Programming of Geometric Elements 469
6.1.2.1 Programming of fully defined Geometric Elements 469
6.1.2.2 Programming of Geometric Elements not completely defined 471
6.1.3 Programming chamfers and fillets between two elements 483

Page deliberately empty

6 Profile Geometry Programming

6.1 Profile Geometry



General

The system gives the user the possibility of programming all or part of a profile consisting of geometric elements.

The system computes the undefined connecting and intersection points between elements located in the same plane.

The points can be computed between the following elements:
- line/line,
- line/circle,
- circle/circle.

Profile Geometry Programming (PGP):
- can coexist with ISO programming,
- can only be used with absolute dimensions (G90),
- applies to the plane selected XY, ZX, YZ for Milling Machines by function G17, G18 or G19 (the change of plane must be programmed on a fully defined point), for Turning Machines it applies for interpolation planes ZX, XY, YZ (by function G20, G21 or G22)
- allows programming of the tool axis (Z in the XY plane) in a block.

Definition of the Geometric Elements

Profile Geometry Programming (PGP) is carried out by writing a sequence of blocks.

Each block includes a geometric element which can be:
- a line segment, or
- an arc.

A geometric element can be completely or incompletely defined in a block.

Completely defined elements are :
- the end point of a line,
- the end point of an arc and the coordinates of the center or the radius.

If an element is incompletely defined, then additional information must be contained in the next block or the next two blocks (excluding fillets and/or chamfers).

The necessary and sufficient blocks allowing the system to compute all the coordinates of a geometric element is called a «geometric entity» (see definition).

The origin of a geometric entity is the start point of the first element.

The start point is:
- programmed in the previous block, or
- already computed by the system (the first block of an entity can also be the last block of the previous entity).

Definition of an entity

A PGP geometric entity is a self-sufficient part of a profile.



Most following examples are given in the XY plane for Milling Machine system. They apply for all couple of interpolable axes for Turning or Milling Machine systems as well.

6.1.1 Definition of the addresses characterizing PGP

The following PGP addresses are processed in the XY plane (G17).
For the other planes (ZX and YZ), use the axes of the plane selected.

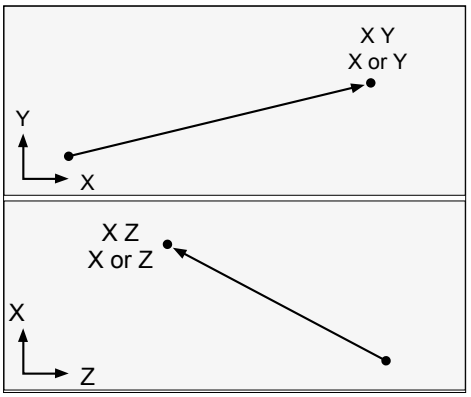
6.1.1.1 Addresses with values ► X../Y.. , Z.., EA, X.. Y.., I.. , K.., J.., R.., EB+, EB-



X../Y..,Z.. or X.. Y.., Z..

X../Y..,Z.., or X.. Y.., Z.., :

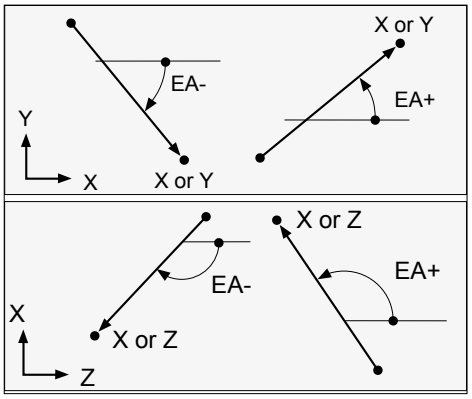
Coordinates of the end point of a line



EA..

EA.. :

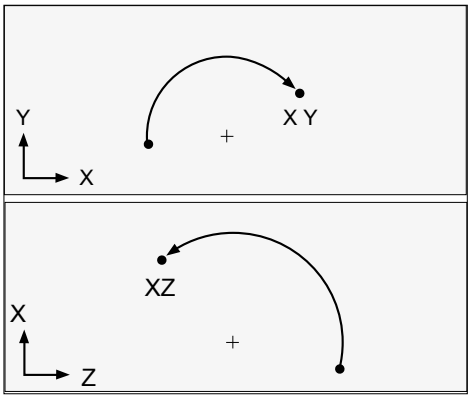
Angle element of a line



X.. Y.. Z..

X.. Y.. Z..:

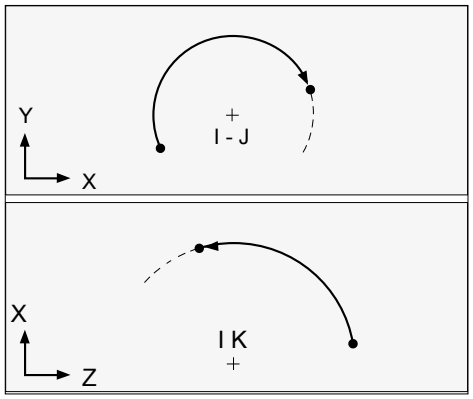
Coordinates of the end point of a circle



I.. J.. K..

I.. J.. K.. :

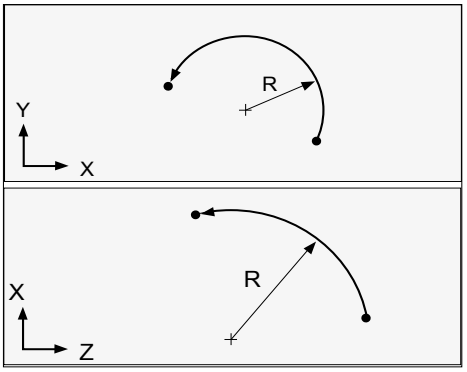
Coordinates of the center of a circle



R..

R.. :

Radius of a circle



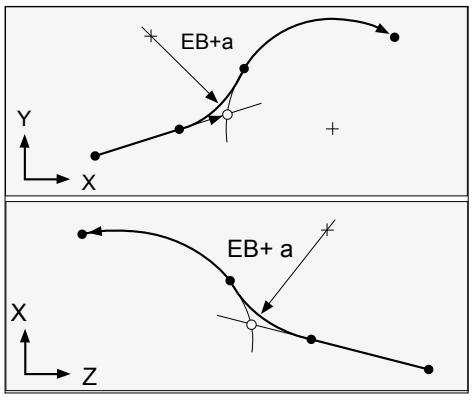
EB+..

EB+.. :

Fillet between two intersecting elements

(e.g. line/circle).

The block containing EB+.. and the next block are connected by a fillet (a=value programmed with EB+).

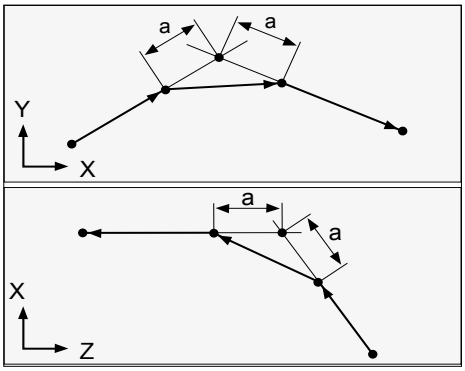


EB-..

EB-.. :

Chamfer between two intersecting lines (only)

The block containing EB-.. and the next block are connected by a chamfer (a=value programmed with EB-).



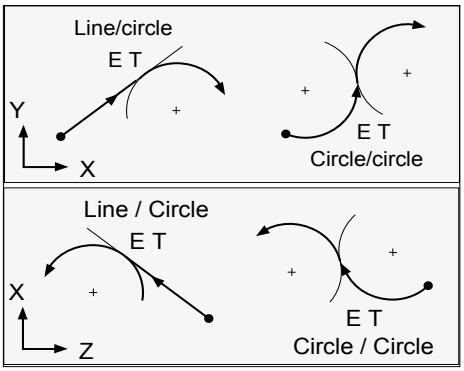
6.1.1.2 Addresses without values ► ET, ES, E+/E-



ET

ET : Tangent element

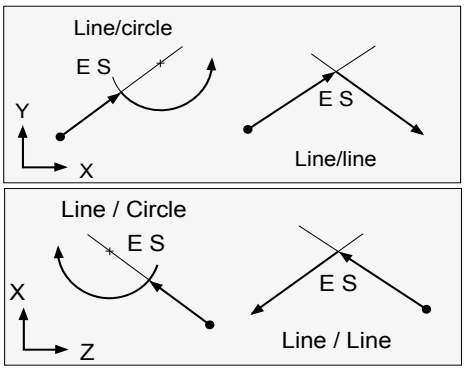
The block containing ET and the next block are tangent. ET is optional unless it is the only function characterizing the element, in which case it is mandatory (see chapter 6.1.2.2, Figures 10 and 14).



ES

ES : Secant element

The block containing ES and the next block intersect. If the intersection point of two secant elements is not programmed, ES must be specified in the first block.



E+/E-

E+/E- : Discriminant

The discriminant is used to dispel the ambiguity when the programming of one or more blocks leaves a choice between two possible solutions.

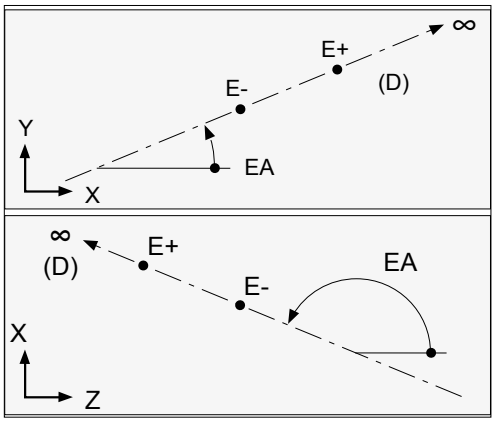
When the discriminant determines an element of an entity:
- it must be programmed in the first block of this entity,
- the + sign or - sign specifies the position of a characteristic point of one of the two solutions with respect to an imaginary line pointing in direction D.

The characteristic points can be:
- the intersection point between two secant elements,
- the point of tangency between two elements,
- the position of the center of a circle.

The pointing line (D) is:
- the line defined by angle EA.. (if one of the elements of the entity is thus defined),
- the line connecting a known point of the first element to a known point of the last element of the entity (pointing from the first to the last). This known point is preferably the center of a circle programmed by I and J or, lacking such, another programmed point.

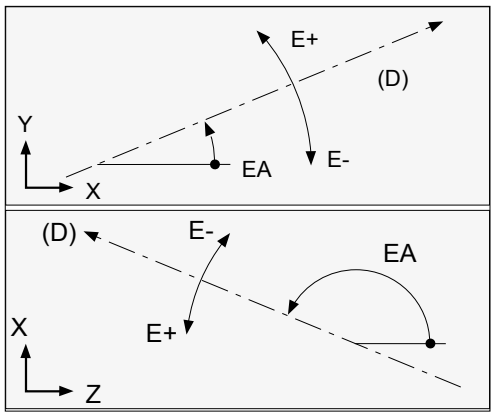
Characteristic points located on the pointing line (D)

- E+ defines the point closest to + ∞ (plus infinity) on line D.
- E- defines the point closest to - ∞ (minus infinity) on line D.



Characteristic points on either side of the pointing line (D)

- E+ defines a point to left of line D.
- E- defines a point to the right of D.



The discriminant can be combined with address ES (secant element) and address ET (tangent element). Example:

- E± secant is programmed as ES+ or ES- .
- E± tangent is programmed as ET+ or ET- .

Programming of the discriminant with secant elements

For line/circle and circle/circle intersecting elements, the system will compute two possible solutions. Programming of the discriminant with ES (ES+ or ES-) is therefore mandatory (see Sec. 6.1.2.2, figures 3a and 3b for instance).

Programming the discriminant with tangent elements

The system allows only two possible solutions (only continuous tangencies without backtracking are allowed by the system).

When there are two possible solutions, these solutions result in:
- the creation of an arc less than 180 degrees, and
- the creation of an arc greater than 180 degrees.

In the two cases, programming of the discriminant with ET is optional. The default solution chosen by the system is that with the smallest arc (see chapter 6.1.2.2, Figures 8a and 8b, for instance).

Only exception:

Circle whose center is inside the next circle and characterized only by: the coordinates of this center and the fact as it is tangent to the next circle (see Sec. 6.1.2.2, Figures 5b, 12b, 23b).

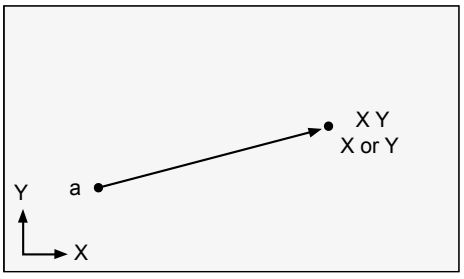
6.1.2 Programming of Geometric Elements

6.1.2.1 Programming of fully defined Geometric Elements

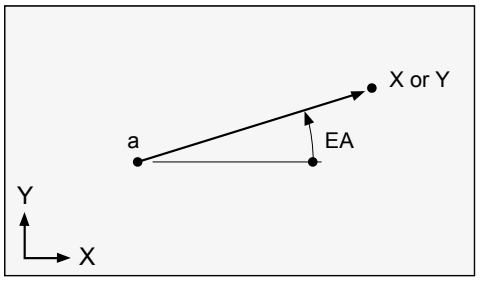
Fully defined linear geometric element (point «a» defined)

Fully defined linear geometric element (point «a» defined)



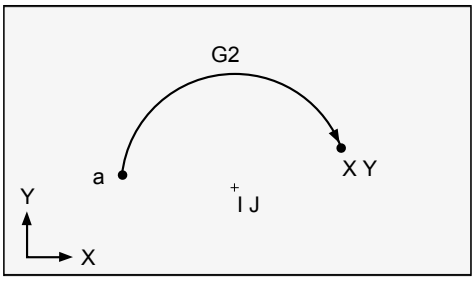



```
     N.. G01 X..
or   N.. G01 Y..
or   N.. G01 X.. Y..
```

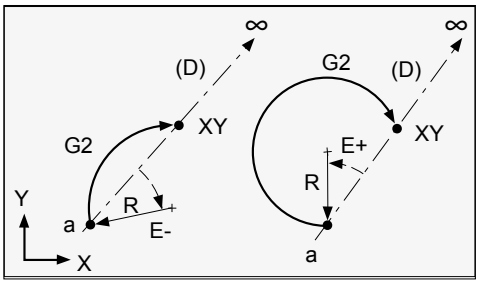



```
     N.. G01 EA.. X..
or   N.. G01 EA.. Y..
```

Fully defined circular geometric element (point «a» defined)

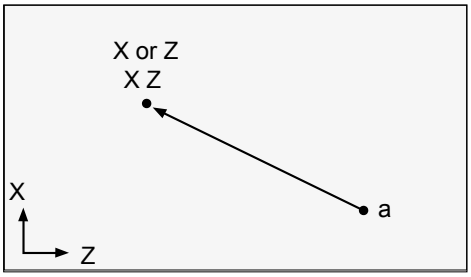



```
N.. G02/G03 X.. Y.. I.. J..
```




```
N.. G02/G03 X.. Y.. R.. E+/E-
```

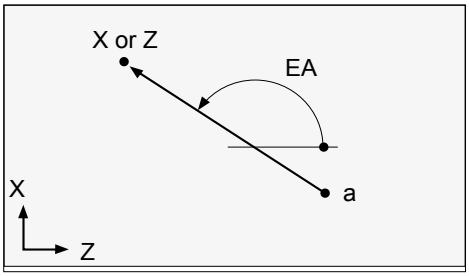
Fully defined linear geometric element (point «a» defined)



N.. G01 X..

or N.. G01 Z..

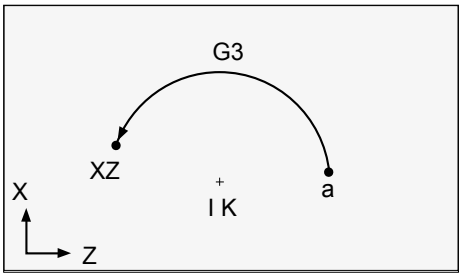
or N.. G01 X.. Z..



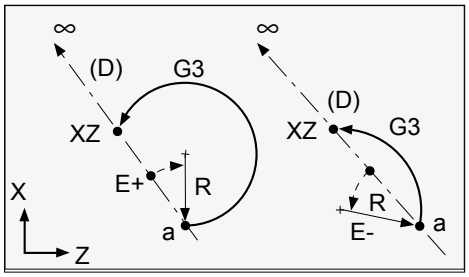
N.. G01 EA.. X..

or N.. G01 EA.. Z..

Fully defined circular geometric element (point «a» defined)



N.. G02/G03 X.. Z.. I.. K..



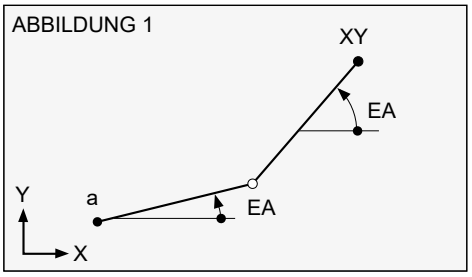
N.. G02/G03 X.. Z.. R.. E+/E-

6.1.2.2 Programming of Geometric Elements not completely defined

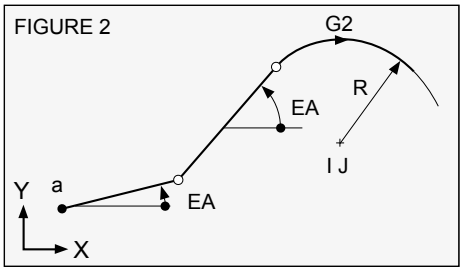
Geometric elements defined using the following block(s)



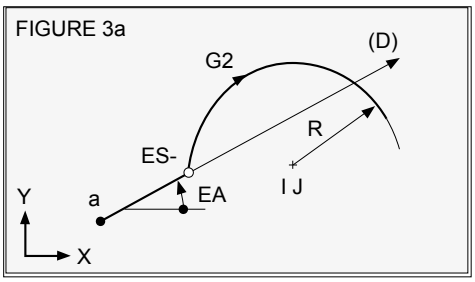
The first element is a line (start point "a" is fully defined)



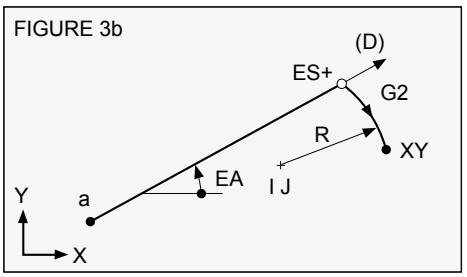

```
N.. G01 EA.. ES
N.. EA X.. Y..
```



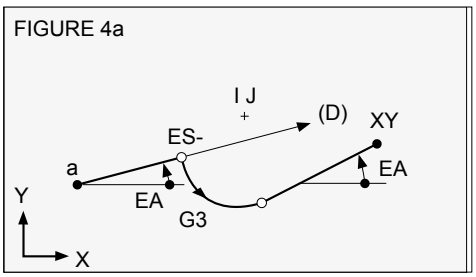

```
N.. G01 EA.. ES
N.. EA..
N.. G02/G03 I.. J.. R../X.. Y..
```



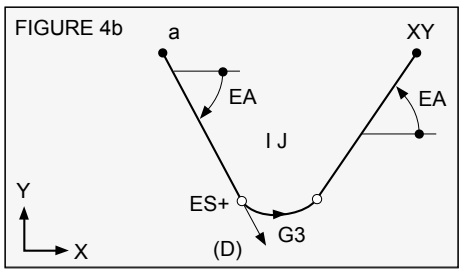

```
N.. G01 EA.. ES-
N.. G02/G03 I.. J.. R../X.. Y..
```



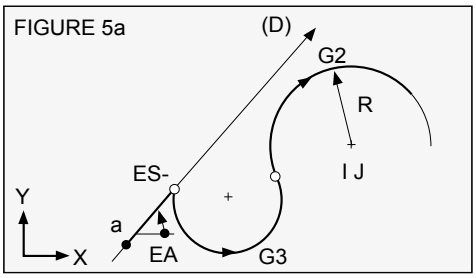

```
N.. G01 EA.. ES+
N.. G02/G03 I.. J.. R../X.. Y..
```




```
N.. G01 EA.. ES-
N.. G02/G03 I..J..
N.. G01 EA.. X.. Y..
```

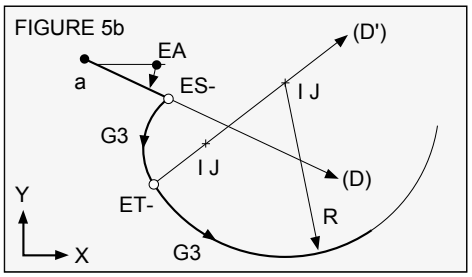



```
N.. G01 EA.. ES+
N.. G02/G03 I..J..
N.. G01 EA.. X.. Y..
```



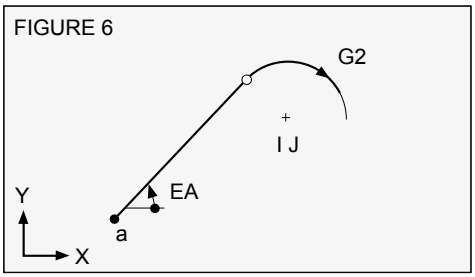

```
N.. G01 EA.. ES-

N.. G02/G03 I.. J..
N.. G02/G03 I.. J.. R../X.. Y..
```

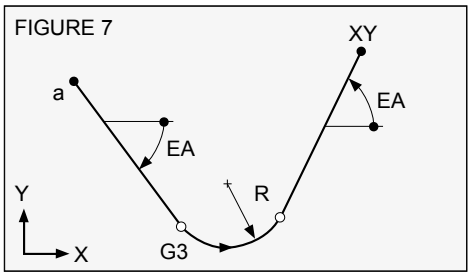



```
N.. G01 EA.. ES-

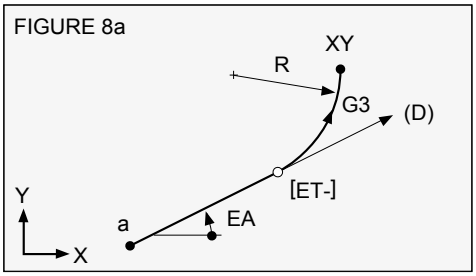
N.. G02/G03 I.. J.. ET-
N.. G02/G03 I.. J.. R../X.. Y..
```



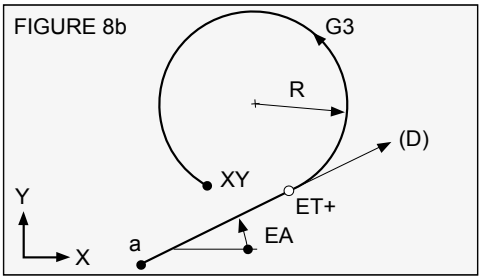

```
N.. G01 EA..
N.. G02/G03 I.. J..
```



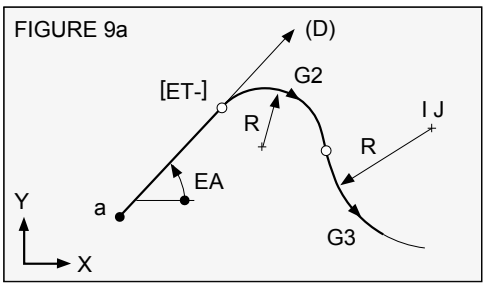

```
N.. G01 EA..
N.. G02/G03 R..
N.. G01 EA.. X.. Y..
```



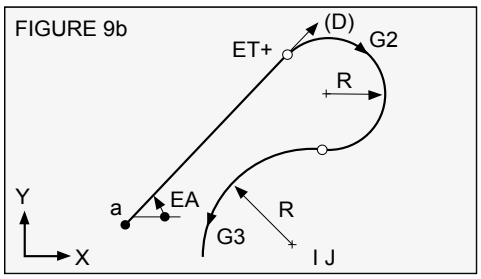

```
N.. G01 EA..
N.. G02/G03 R.. X.. Y..
```




```
N.. G01 EA.. ET+
N.. G02/G03 R.. X.. Y..
```

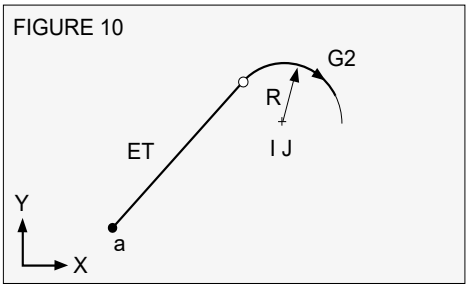



```
N.. G01 EA..
N.. G02/G03 R..
N.. G02/G03 I.. J.. R../X.. Y..
```



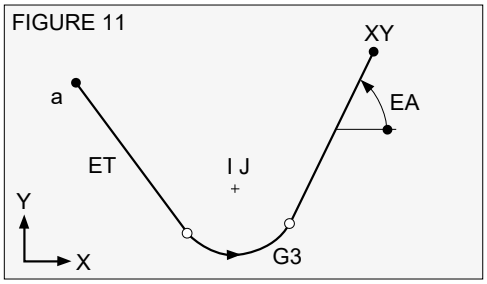

```
N.. G01 EA.. ET+
N.. G02/G03 R..
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 10



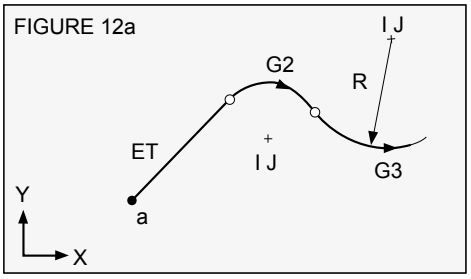

```
N.. G01 ET
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 11



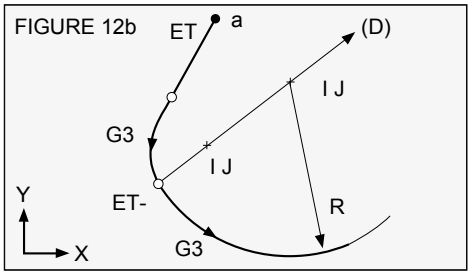

```
N.. G01 ET
N.. G02/G03 I.. J..
N.. G01 EA.. X.. Y..
```

FIGURE 12a




```
N.. G01 ET
N.. G02/G03 I.. J..
N.. G02/G03 I.. J.. R../X.. Y..
```

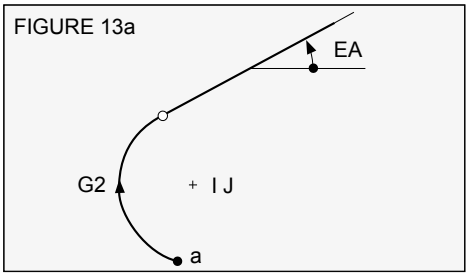
FIGURE 12b




```
N.. G01 ET
N.. G02/G03 I.. J.. ET-
N.. G02/G03 I.. J.. R../X.. Y..
```

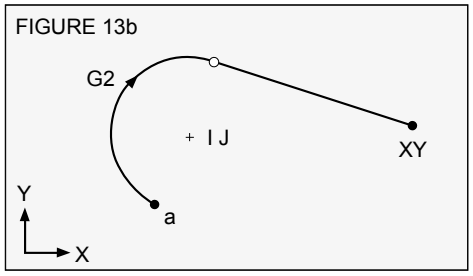
The first element is a circle (start point «a» is fully defined).

FIGURE 13a



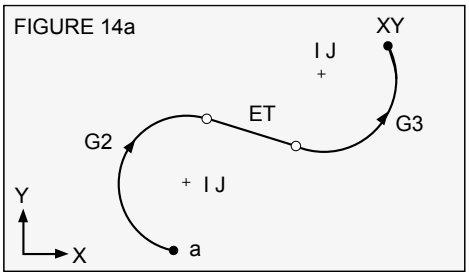

```
N.. G02/G03 I.. J..
N.. G01 EA..
```

FIGURE 13b



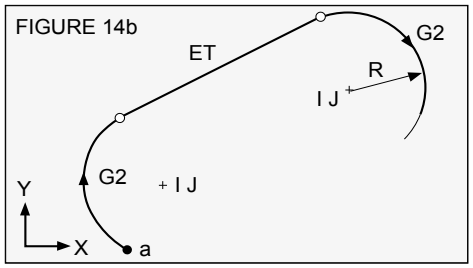

```
N.. G02/G03 I.. J..
N.. G01 X.. Y..
```

FIGURE 14a



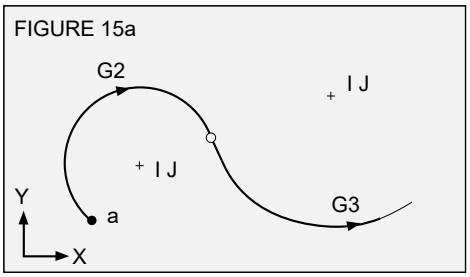

```
N.. G02/G03 I.. J..
N.. G01 ET
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 14b



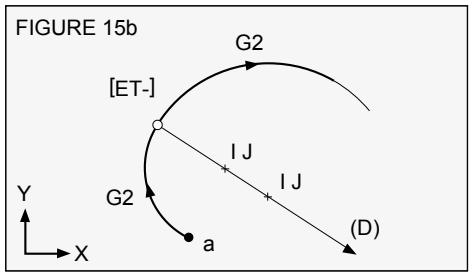

```
N.. G02/G03 I.. J..
N.. G01 ET
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 15a



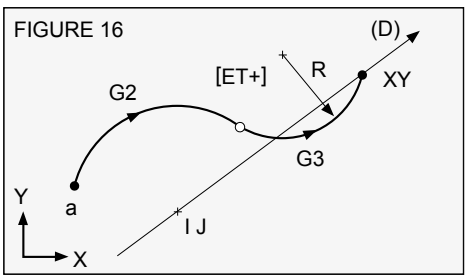

```
N.. G02/G03 I.. J..
N.. G02/G03 I.. J..
```

FIGURE 15b



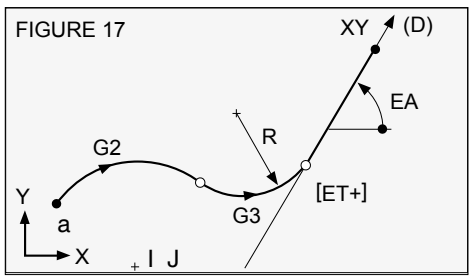

```
N.. G02/G03 I.. J..
N.. G02/G03 I.. J..
```

FIGURE 16



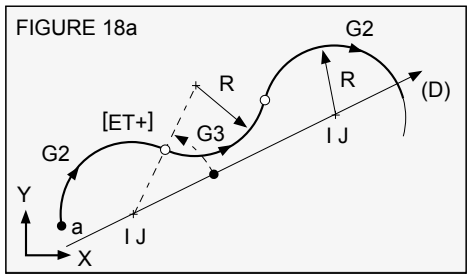

```
N.. G02/G03 I.. J..
N.. G02/G03 R.. X.. Y..
```

FIGURE 17



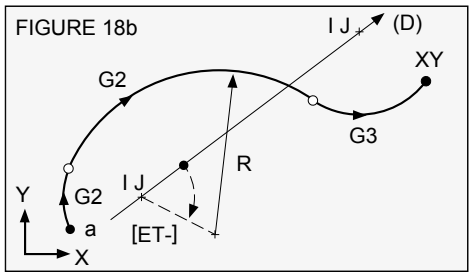

```
N.. G02/G03 I.. J..
N.. G02/G03 R..
N.. G01 EA.. X.. Y..
```

FIGURE 18a



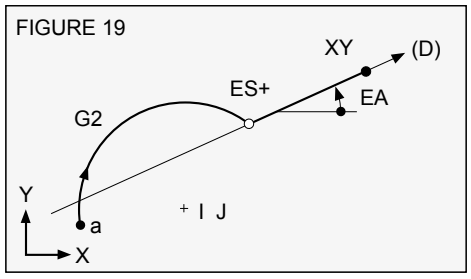

```
N.. G02/G03 I.. J..
N.. G02/G03 R..
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 18b



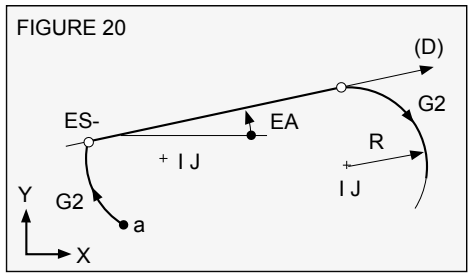

```
N.. G02/G03 I.. J..
N.. G02/G03 R..
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 19



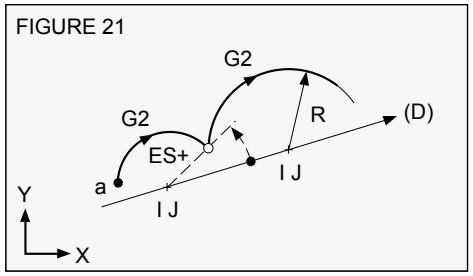

```
N.. G02/G03 I.. J.. ES+
N.. G01 EA.. X.. Y..
```

FIGURE 20



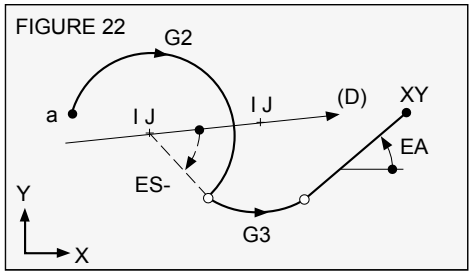

```
N.. G02/G03 I.. J.. ES-
N.. G01 EA..
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 21



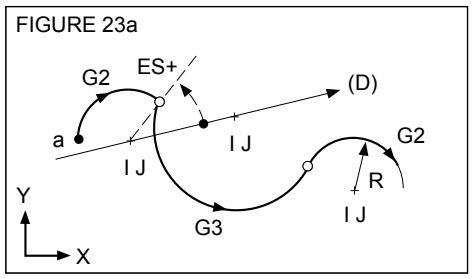

```
N.. G02/G03 I.. J.. ES+
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 22



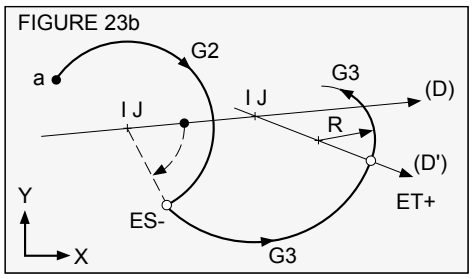

```
N.. G02/G03 I.. J.. ES-
N.. G02/G03 I.. J..
N.. G01 EA X.. Y..
```

FIGURE 23a



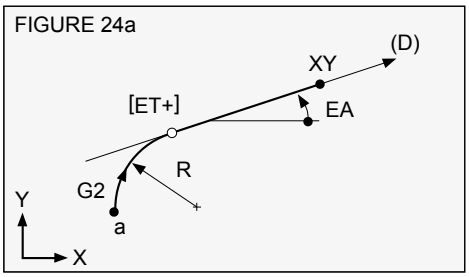

```
N.. G02/G03 I.. J.. ES+
N.. G02/G03 I.. J..
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 23b



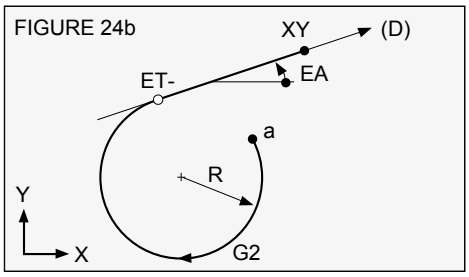

```
N.. G02/G03 I.. J.. ES-
N.. G02/G03 I.. J.. ET+
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 24a



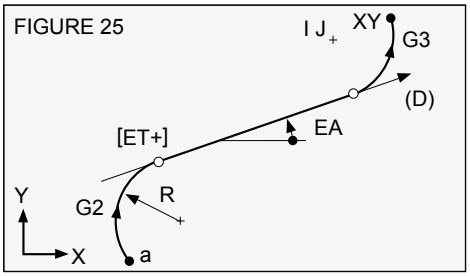

```
N.. G02/G03 R..
N.. G01 EA.. X.. Y..
```

FIGURE 24b



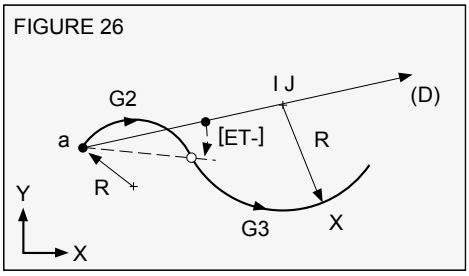

```
N.. G02/G03 R.. ET-
N.. G01 EA.. X.. Y..
```

FIGURE 25



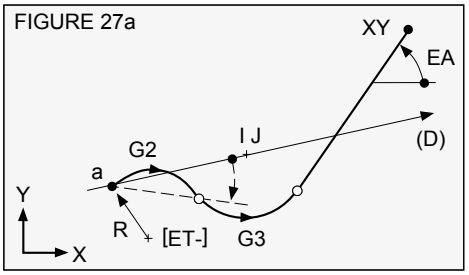

```
N.. G02/G03 R..
N.. G01 EA..
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 26



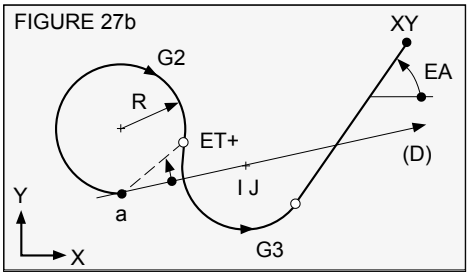

```
N.. G02/G03 R..
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 27a



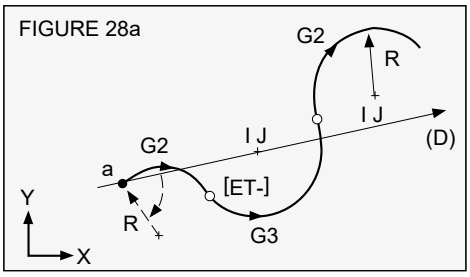

```
N.. G02/G03 R..
N.. G02/G03 I.. J..
N.. G01 EA.. X.. Y..
```

FIGURE 27b



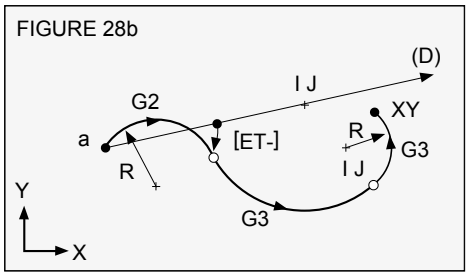

```
N.. G02/G03 R.. ET+
N.. G02/G03 I.. J..
N.. G01 EA.. X.. Y..
```

FIGURE 28a




```
N.. G02/G03 R..
N.. G02/G03 I.. J..
N.. G02/G03 I.. J.. R../X.. Y..
```

FIGURE 28b

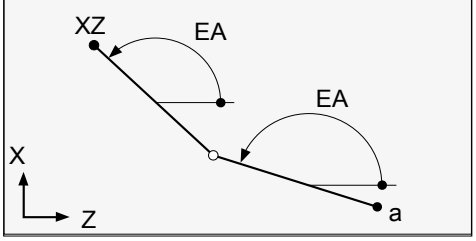



```
N.. G02/G03 R..
N.. G02/G03 I.. J..
N.. G02/G03 I.. J.. R../X.. Y..
```

Geometric elements defined using the following block(s)

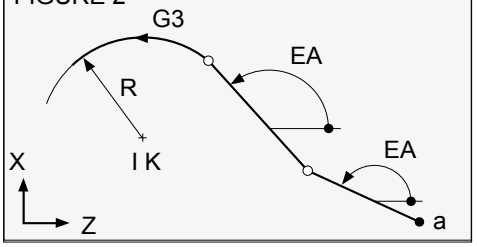
The first element is a line (start point «a» is fully defined).

FIGURE 1



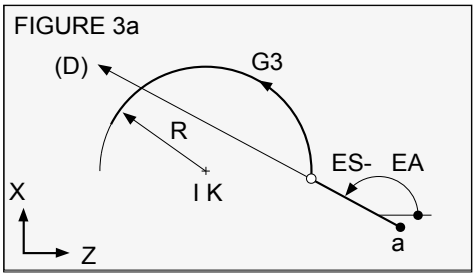

```
N.. G01 EA.. ES
N.. EA X.. Z..
```

FIGURE 2



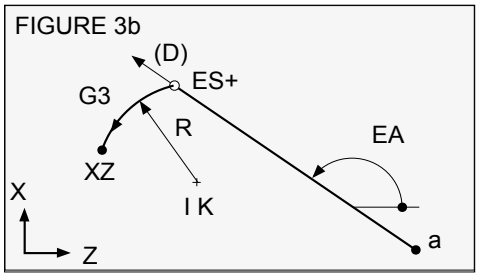

```
N.. G01 EA.. ES
N.. EA..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 3a



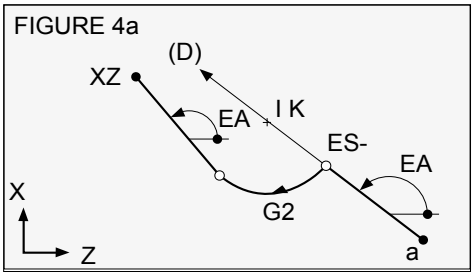

```
N.. G01 EA.. ES-
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 3b



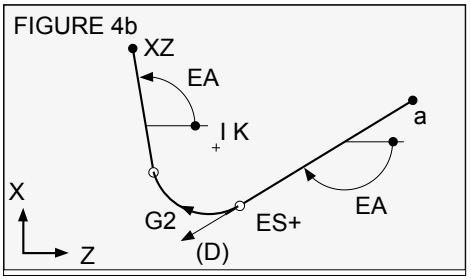

```
N.. G01 EA.. ES+
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 4a



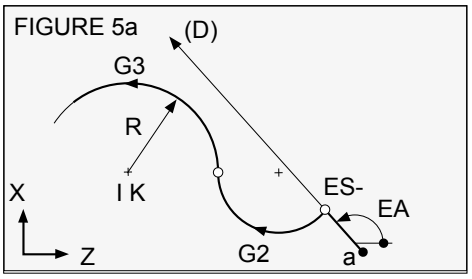

```
N.. G01 EA.. ES-
N.. G02/G03 I..K..
N.. G01 EA.. X.. Z..
```

FIGURE 4b



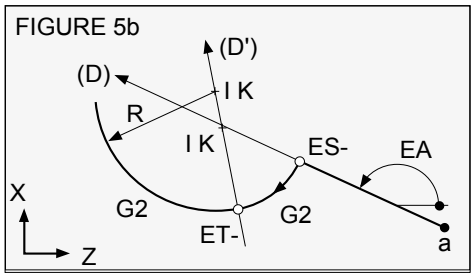

```
N.. G01 EA.. ES+
N.. G02/ G03 I..K..
N.. G01 EA.. X.. Z..
```

FIGURE 5a



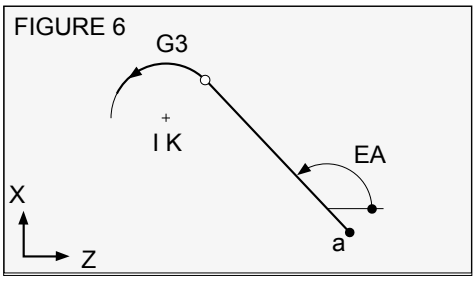

```
N.. G01 EA.. ES-
N.. G02/G03 I.. K..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 5b



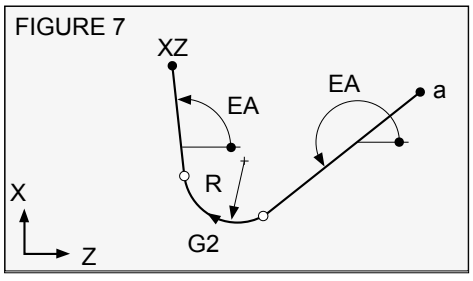

```
N.. G01 EA.. ES-
N.. G02/G03 I.. K.. ET-
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 6



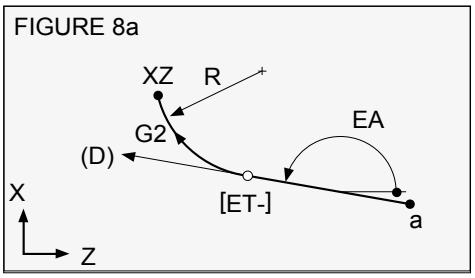

```
N.. G01 EA..
N.. G02/G03 I.. K..
```

FIGURE 7



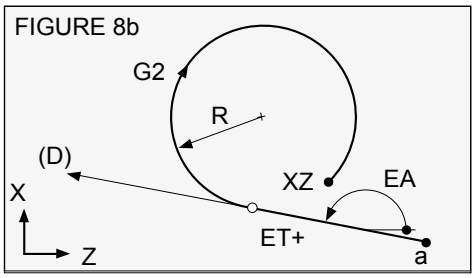

```
N.. G01 EA..
N.. G02/G03 R..
N.. G01 EA.. X.. Z..
```

FIGURE 8a



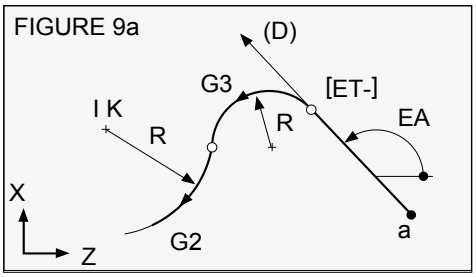

```
N.. G01 EA..
N.. G02/G03 R.. X.. Z..
```

FIGURE 8b



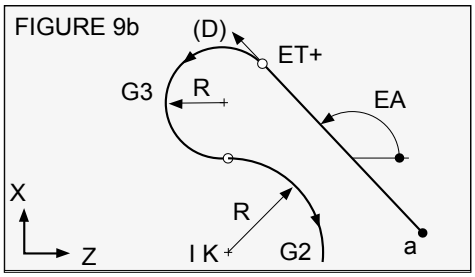

```
N.. G01 EA.. ET+
N.. G02/G03 R.. X.. Z..
```

FIGURE 9a



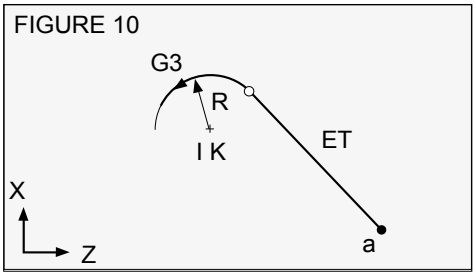

```
N.. G01 EA..
N.. G02/G03 R..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 9b



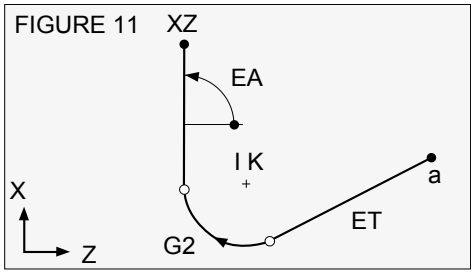

```
N.. G01 EA.. ET+
N.. G02/G03 R..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 10



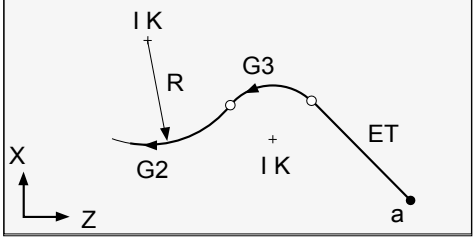

```
N.. G01 ET
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 11



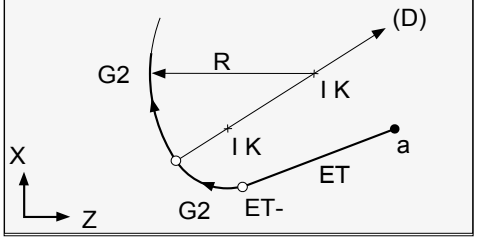

```
N.. G01 ET
N.. G02/G03 I.. K..
N.. G01 EA.. X.. Z..
```

FIGURE 12a




```
N.. G01 ET
N.. G02/G03 I.. K..
N.. G02/G03 I.. K.. R../X.. Z..
```

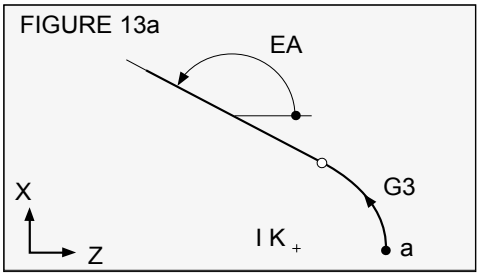
FIGURE 12b




```
N.. G01 ET
N.. G02/G03 I.. K.. ET-
N.. G02/G03 I.. K.. R../X.. Z..
```

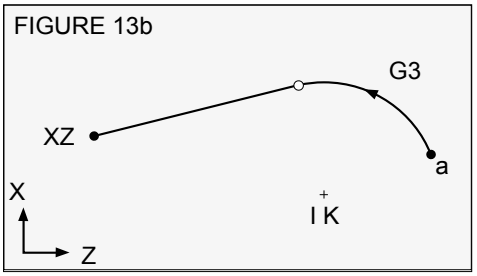
The first element is a circle (start point "a" is fully defined).

FIGURE 13a



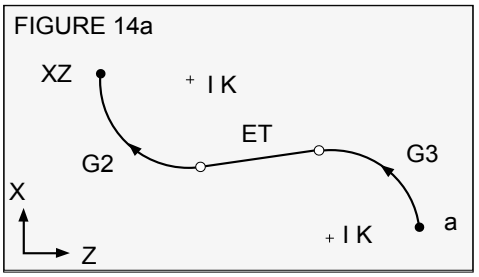

```
N.. G02/G03 I.. K..
N.. G01 EA..
```

FIGURE 13b



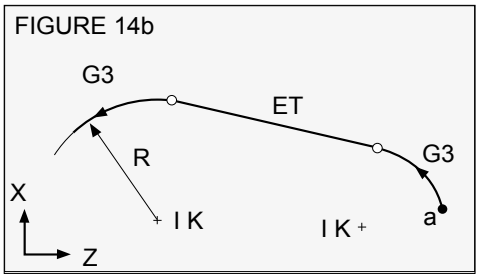

```
N.. G02/G03 I.. K..
N.. G01 X.. Z..
```

FIGURE 14a



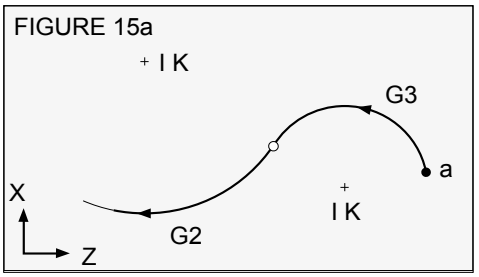

```
N.. G02/G03 I.. K..
N.. G01 ET
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 14b



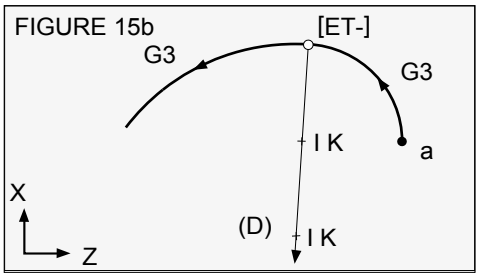

```
N.. G02/G03 I.. K..
N.. G01 ET
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 15a



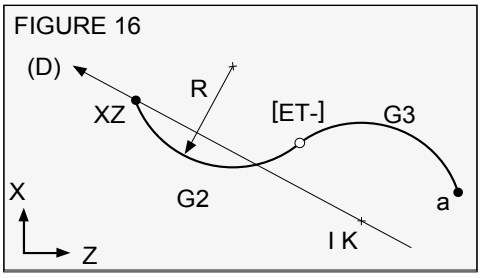

```
N.. G02/G03 I.. K..
N.. G02/G03 I.. K..
```

FIGURE 15b



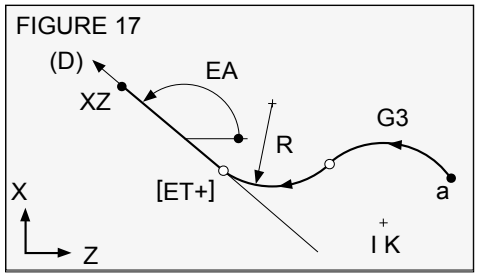

```
N.. G02/G03 I.. K..
N.. G02/G03 I.. K..
```

FIGURE 16



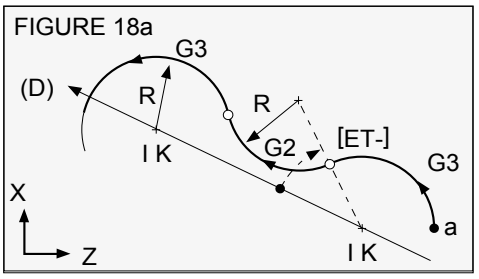

```
N.. G02/G03 I.. K..
N.. G02/G03 R.. X.. Z..
```

FIGURE 17



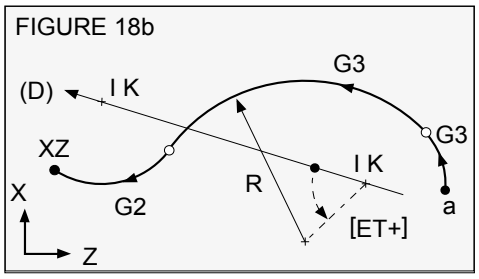

```
N.. G02/G03 I.. K..
N.. G02/G03 R..
N.. G01 EA.. X.. Z..
```

FIGURE 18a



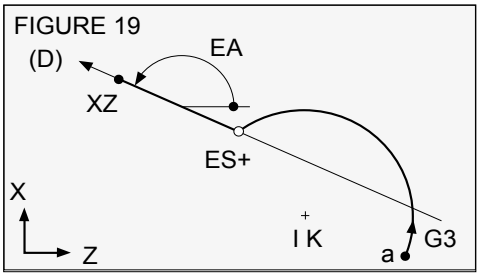

```
N.. G02/G03 I.. K..
N.. G02/G03 R..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 18b



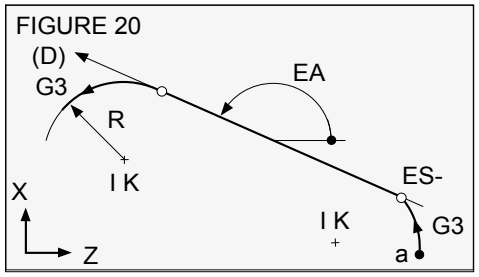

```
N.. G02/G03 I.. K..
N.. G02/G03 R..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 19



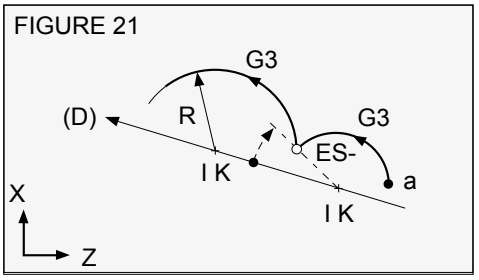

```
N.. G02/G03 I.. K.. ES+
N.. G01 EA.. X.. Z..
```

FIGURE 20



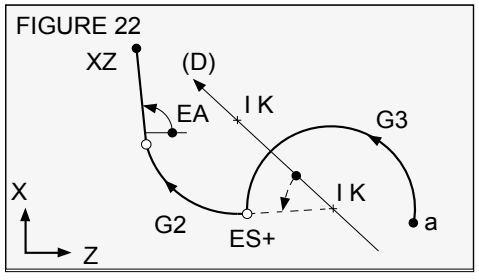

```
N.. G02/G03 I.. K.. ES-
N.. G01 EA..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 21



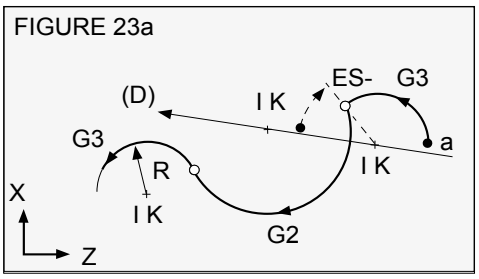

```
N.. G02/G03 I.. K.. ES-
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 22



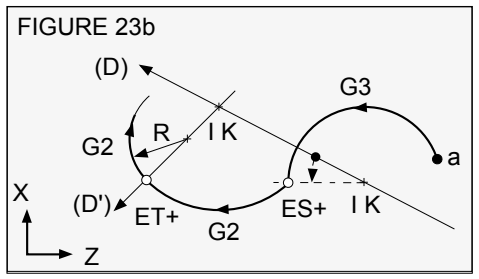

```
N.. G02/G03 I.. K.. ES+
N.. G02/G03 I.. K..
N.. G01 EA X.. Z..
```

FIGURE 23a



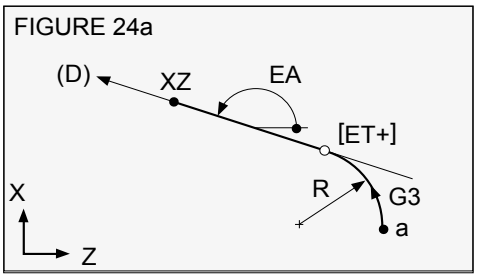

```
N.. G02/G03 I.. K.. ES-
N.. G02/G03 I.. K..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 23b



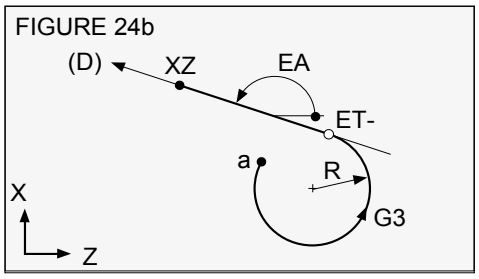

```
N.. G02/G03 I.. K.. ES+
N.. G02/G03 I.. K.. ET+
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 24a



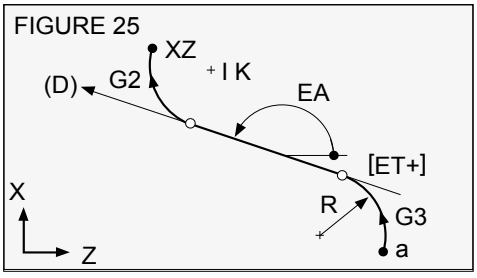

```
N.. G02/G03 R..
N.. G01 EA.. X.. Z..
```

FIGURE 24b



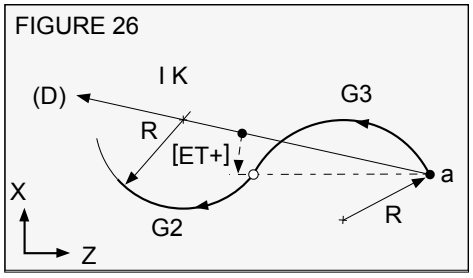

```
N.. G02/G03 R.. ET-
N.. G01 EA.. X.. Z..
```

FIGURE 25



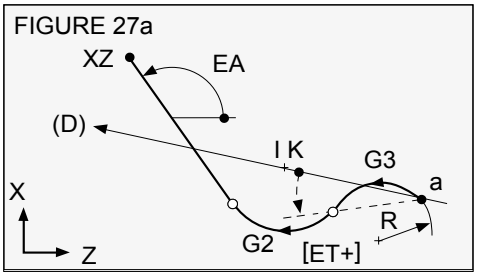

```
N.. G02/G03 R..
N.. G01 EA..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 26



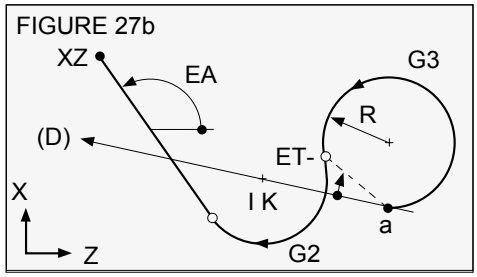

```
N.. G02/G03 R..
N.. G02/G03 I.. K.. R../X.. Z..
```

FIGURE 27a



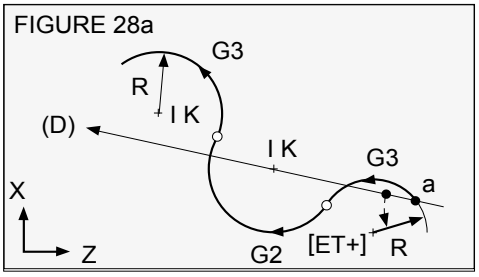

```
N.. G02/G03 R..
N.. G02/G03 I.. K..
N.. G01 EA.. X.. Z..
```

FIGURE 27b



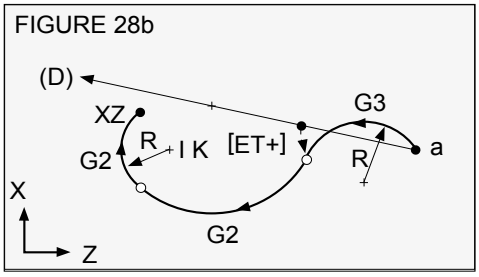

```
N.. G02/G03 R.. ET-
N.. G02/G03 I.. K..
N.. G01 EA.. X.. Z..
```

FIGURE 28a




```
N.. G02/G03 R..
N.. G02/G03 I.. K..
N.. G02/G03 I.. K.. R../X.. Z..
```

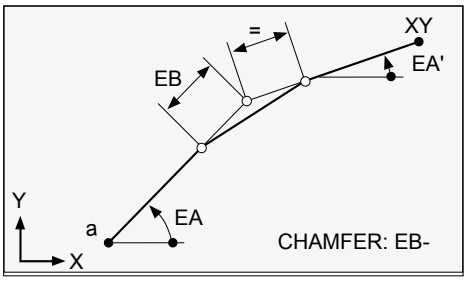
FIGURE 28b



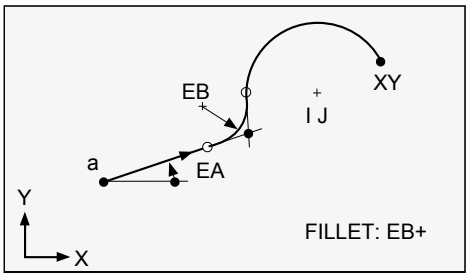

```
N.. G02/G03 R..
N.. G02/G03 I.. K..
N.. G02/G03 I.. K.. R../X.. Z..
```

6.1.3 Programming chamfers and fillets between two elements



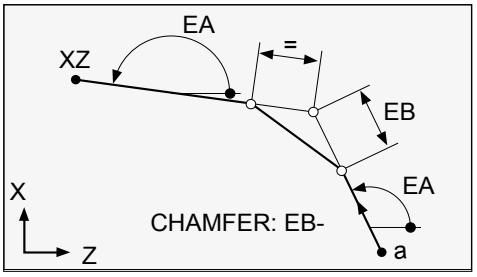



```
N.. G01 EA.. ES EB-..
N.. G01 EA.. X.. Y..
```

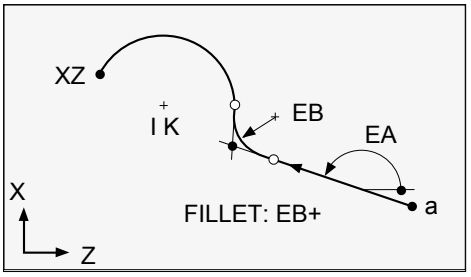



```
N.. G01 EA.. ES- EB+..
N.. G02/G03 I.. J.. X.. Y..
```





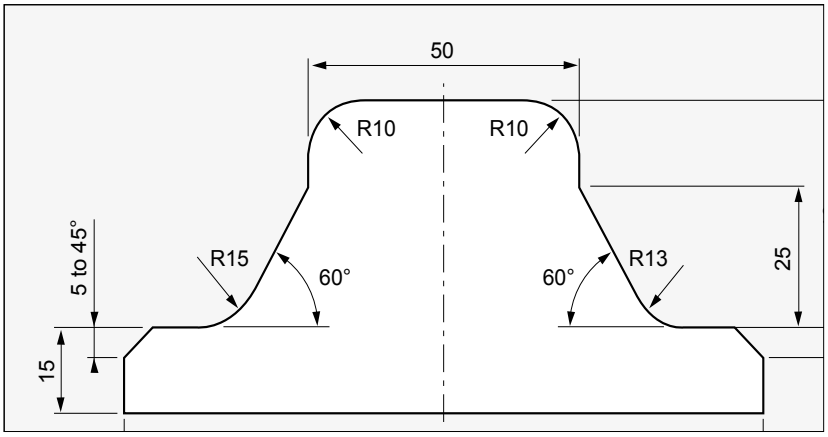

```
N.. G01 EA.. ES EB-..
N.. G01 EA.. X.. Z..
```



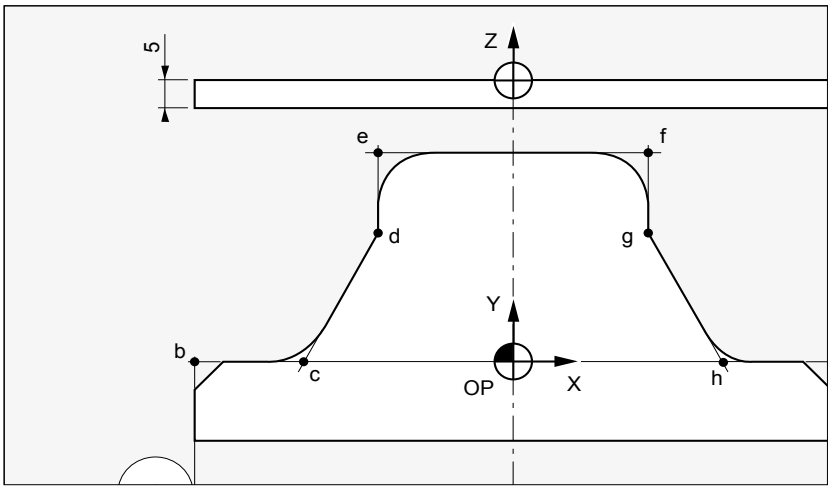

```
N.. G01 EA.. ES- EB+..
N.. G02/G03 I.. K.. X.. Z..
```

Examples of Programming in PGP

Milled part #1 profile definition



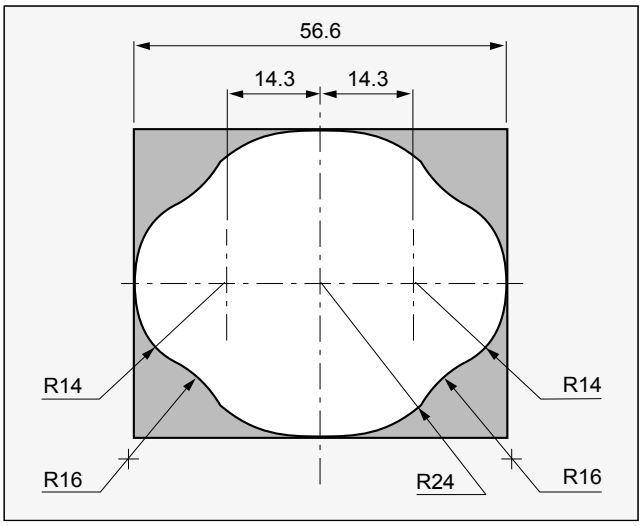
Machining paths



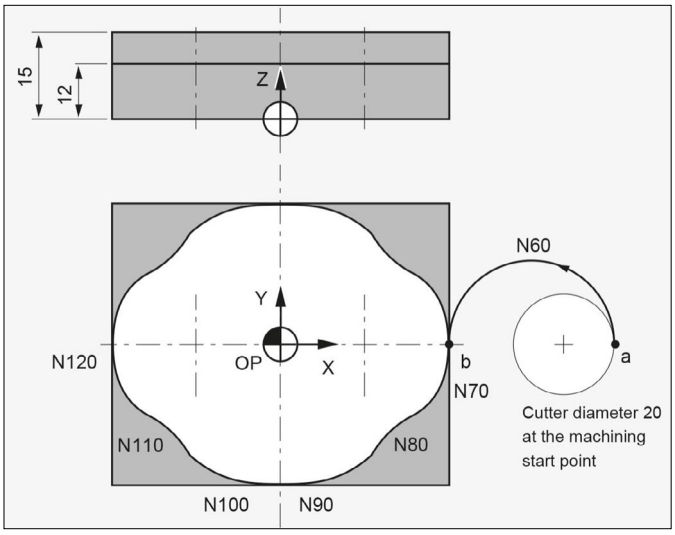

```
%61
N10 G00 G52 Z0
N20 T03 D03 M06 (CUTTER DIAMETER=10)
N30 S600 M40 M03
N35 G92 R10
N40 G00 G41 X-60 Y-25          Point a, approach
N50 Z-6                        Position on Z
N60 G01 Y0 EB-5 F150 M08       Point b
N70 EA0 ES EB15                Point c
N80 EA60 X-25 Y25              Point d
N90 Y40 EB10                   Point e
N100 X25 EB10                  Point f
N110 Y25                       Point g
N120 EA-60 Y0 EB13             Point h
N130 X60 EB-5                  Point i
N140 Y-20                      Point j
N150 G00 G40 G52 Z0 M05 M09    Retraction
N160 M02
```

Half part profile in XY plane

Use of the mirror function to machine the other half-profile in the XY plane (G17).



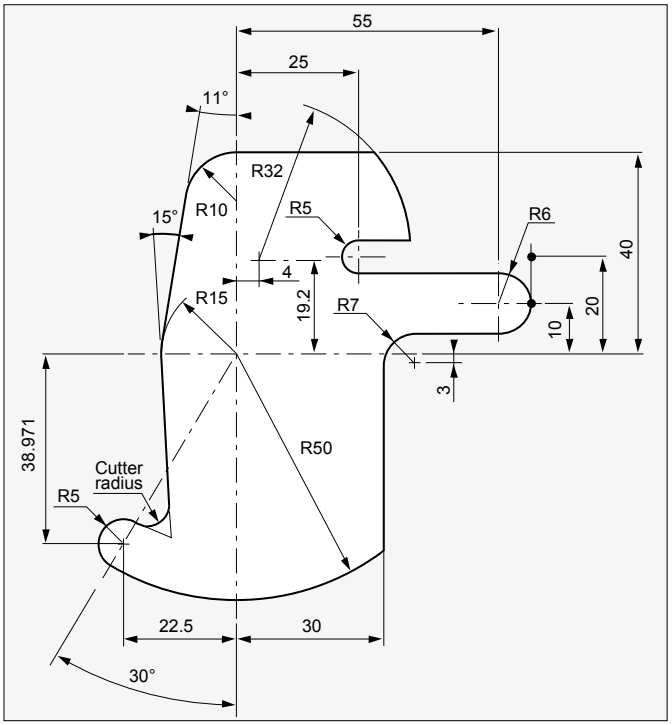
Machining paths



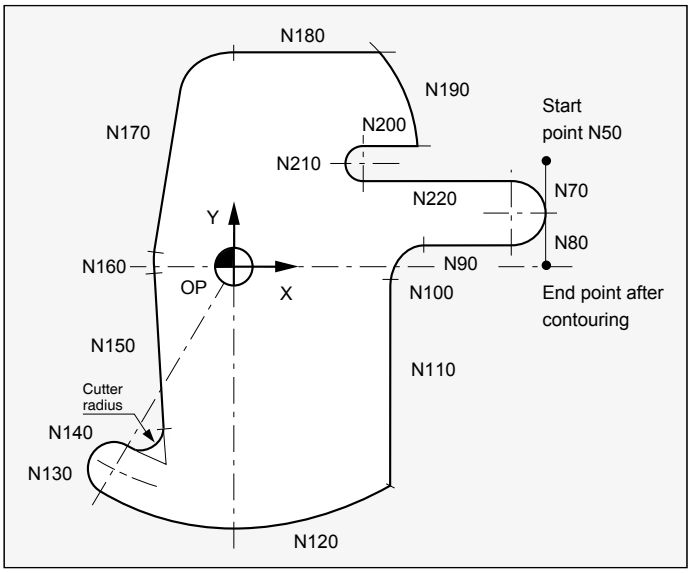

```
%3
N10 G00 G52 Z0
N20 T01 D01 M06 (CUTTER DIAMETER=20)
N30 S280 M40 M03
N40 G00 G41 X60 Y0                    Point a, approach
N50 Z12 M08                           Position on Z
N55 G92 R10                           Limit value of tangential feed rate
N60 G03 X28.3 Y0 I44.15 J0 F200       Point b
N70 G02 I14.3 J0 R14 ET               Machining of the first half-profile
N80 G03 R16 ET
N90 G02 X0 Y-24 I0 J0
N100 G02 I0 J0 R24 ET
N110 G03 R16 ET
N120 G02 X-28.3 Y0 I-14.3 J0
N130 G03 X-60 Y0 I-44.15 J0
N140 G51 X- Y-                        Mirror
N150 G77 N60 N120                     Machining of the second half-profile
```

```
N160 G51 X+ Y+
N170 G00 G40 G52 Z0 M05 M09
N180 M02
```

Milled Part#2 Profile definition



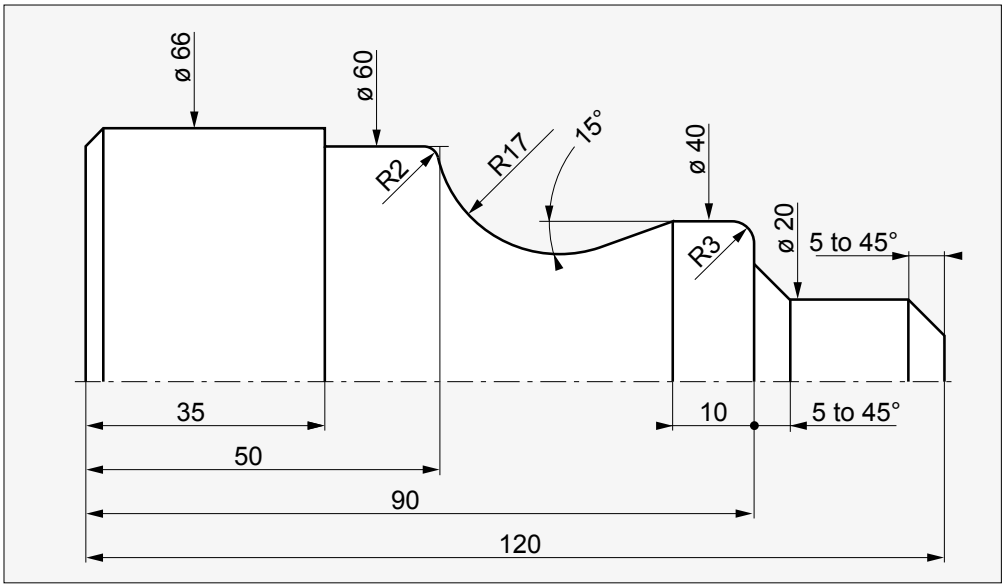
Machining paths



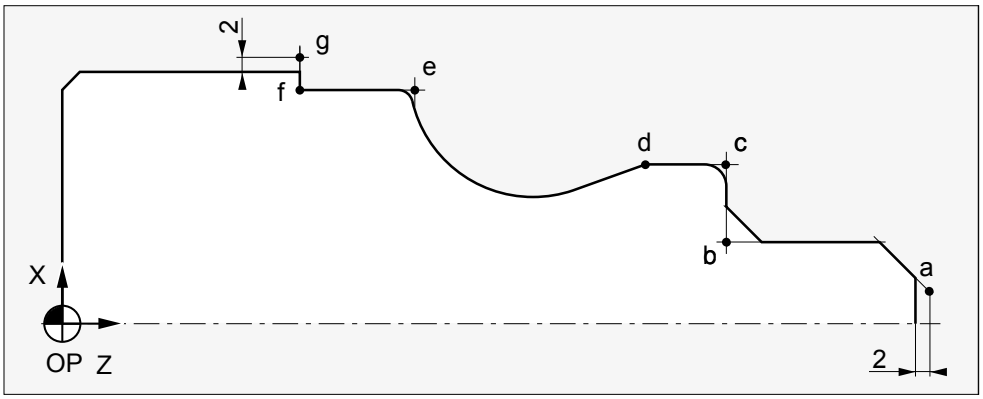

```
%18
N10 G00 G52 Z0
N20 T02 D02 M06 (CUTTER DIAMETER=6)
N30 S1200 M40 M03
```

```
N40 G92 R4                    Limiting of the tangential feed rate
N50 G00 G41 X61 Y20 Z3        Start point
N60 G01 Z-5 F50 M08           Penetration on Z
N70 Y10 F120
N80 G02 I55 J10
N90 G01 ET
N100 G03 I37 J-3 R7
N110 G01 EA-90 ES+
N120 G02 I0 J0 R50
N130 I-22.5 J-38.971
N140 G03 I0 J0 ES-
N150 G01 EA94
N160 G02 I0 J0 R15
N170 G01 EA79 Y40 EB10
N180 EA0 ES+
N190 G02 I4 J19.2 R32 ES-
N200 G01 EA180
N210 G03 I25 J21 R5
N220 G01 ET
N230 G02 I55 J10 X61 Y10
N240 G01 Y0 F500
N250 G00 G40 G52 Z0 M05 M09
N260 M02
```

Turned part #1 profile definition



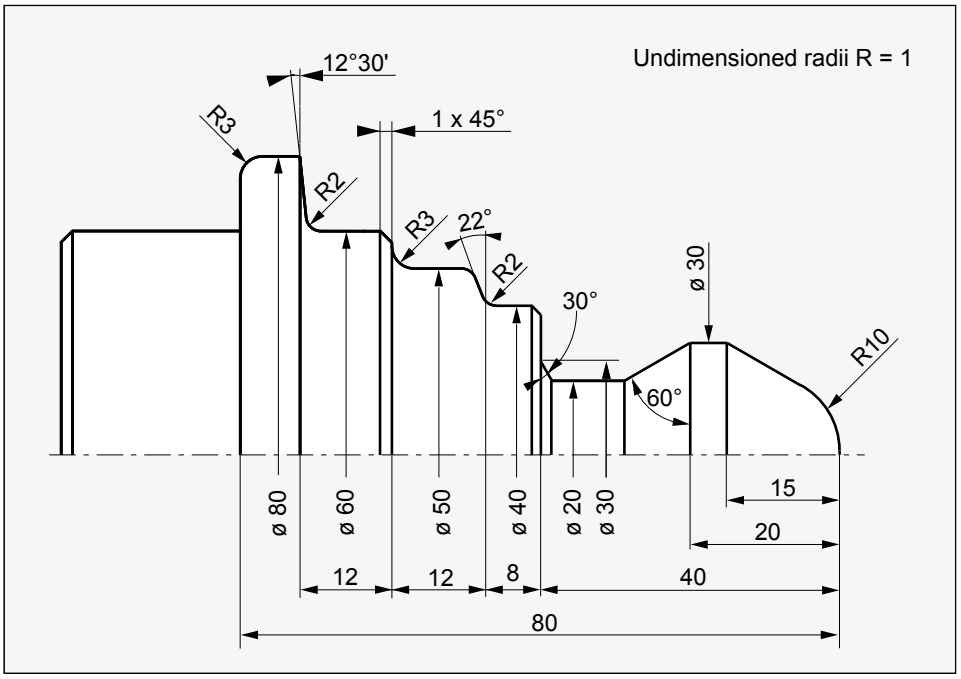
Machining paths



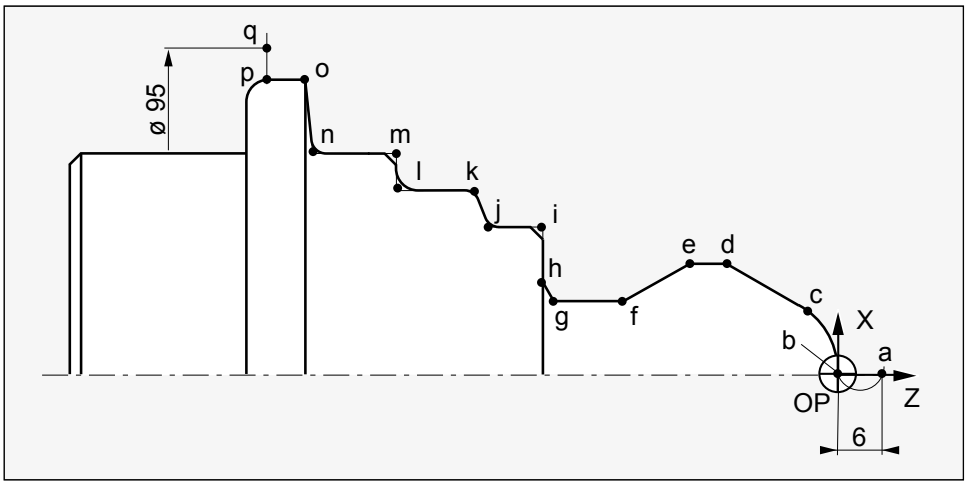

```
%88
N10 ...
N.. G92 S3000
N..
N90 G00 G52 X0 Z0
N100 T03 D03 M06 (COPYING TOOL R0.4)
N110 G97 S900 M04
N120 G00 G42 X6 Z122                    Point a, approach
N130 G96 S220
N140 G95 F0.1
N150 G01 EA135 ES
N160 EA180 X20 Z90 EB-5                 Point b
N170 X40 EB3                            Point c
N180 EA180 Z80 ES                       Point d
N190 EA195
N200 G02 X60 Z50 R17 EB2                Point e
N210 G01 Z35                            Point f
```

```
N220 X70                                  Point g
N230 G00 G40 G52 X0 Z0 G97 S900
…
```

In PGP, part profile definition



Machining paths




```
%109
N10 …
N.. G92 S3000
N..
N110 G00 G52 X0 Z0
N120 T05 D05 M06 (COPYING TOOL R0.4)
N130 G97 S900 M04
N140 G00 G42 X0 Z6                        Point a, approach
N150 G92 S3000
```

```
N160 G96 S150
N170 G95 F0.08
N180 G02 X0 Z0 I0 K3          Point b
N190 G03 I0 K-10              Point c
N200 G01 X30 Z-15             Point d
N210 Z-20                     Point e
N220 EA-150 X20               Point f
N230 EA180 ES                 Point g
N240 EA120 X30 Z-40           Point h
N250 X40 EB1                  Point i
N260 EA180 Z-48 EB2           Point j
N270 EA 112 X50 EB1           Point k
N280 Z-60 EB3                 Point l
N290 X60 EB-1                 Point m
N300 EA180 ES EB2             Point n
N310 EA102.5 X80 Z-72 EB1     Point o
N320 Z-78                     Point p
N330 G00 X95                  Point q
N340 G40 G52 X0 Z0 G97 S900
N..
```

Page deliberately empty

Page deliberately empty

Summary

7 Parametric Programming 495

7.1 Operations and functions 495
7.1.1 Arithmetic Operations 495
7.1.2 Arithmetic Functions 496
7.1.2.1 Functions available since first versions of Flexium+ 496
7.1.2.2 Functions available since Flexium+ revision 4.1.10 497
7.1.3 Logic Operations 500
7.1.4 Assignment 500
7.1.5 Comparison Symbols Used with L Variables 501
7.1.6 Conversion in Internal Unit 501
7.1.7 Curly Brackets 501
7.1.8 Priorities 502
7.2 Programming syntax for L variables 503
7.2.1 Assignment of a Variable to an NC function 504
7.2.2 Test of a variable for a conditional branch 506
7.2.3 Notes on programming variables L100 to L199 and L900 to L959 507
7.2.4 Equivalence of variables L900 to L925 507
7.2.5 Symbolic addressing of variables L900 to L925 and L926 to L951 507
7.3 Programming syntax for E variables 516
7.4 Address equivalences 517
7.5 Errors and Warnings 518

Page deliberately empty

7 Parametric Programming



Parametric programming uses functions that can be assigned to all the NC addresses in place of numerical values and that can be used as particular functions.

Functions used in parametric programming:
- Program L variables,
- External E parameters.

7.1 Operations and functions

7.1.1 Arithmetic Operations

Arithmetic operation	*Symbol*	*Arithmetic operation*	*Symbol*
Addition	+	Multiplication	*
Subtraction	-	Division	/



Division by zero is impossible and results in error 94.

7.1.2 Arithmetic Functions

7.1.2.1 Functions available since first versions of Flexium+

Arithmetic function	*Symbol*	*Arithmetic function*	*Symbol*
Sine	S	Arc tangent	A
Cosine	C	Square root	R
Truncation	T		

Sine (S) and cosine (C)	The term following these functions is in degrees. If the argument is an E parameter it will be automatically scaled by 1/1000000.
Truncation	Extraction of the integer value of the number following the symbol.
Arc tangent	The result of the operation is in degree.



Extraction of the square root of a negative number is impossible and results in error 94.

7.1.2.2 Functions available since Flexium+ revision 4.1.10

This version has introduced several new arithmetic and trigonometric functions.

They can be recognized by the prefix '**_**'.

At the difference of previous functions the arguments are real numbers therefore there is no scaling as it could be the case with previous function which are maintained for compatibility.

Function	**Syntax**
Arc sine	**_ASIN**
Arc cosine	**_ACOS**
Arc tangent	**_ATAN**
Arc tangent (on -180°, 180°)	**_ATAN2**
Arc tangent (on 0, 360°)	**_ATAN3**
Exponential	**_EXP**
Natural logarithm	**_LN**
Base 10 logarithm	**_LOG**
Absolute value	**_ABS**
Tangent	**_TAN**
Hyperbolic sine	**_HSIN**
Hyperbolic cosine	**_HCOS**
Hyperbolic Tangent	**_HTAN**
Maximum	**_MAX** **a = _MAX _b/d/..... or _MAX @[V(n)/I+]**
Minimum	**_MIN** **a = _MIN _b/d/..... or _MIN @[V(n)/I+]**
Conversion to radians	**_RAD**



- The syntax a= _MAX @[V(n)]/I searches for the MAX element of vector V among the first I elements. The syntax for _MIN is equivalent. argument I is optional in this case the search is made on the full set of elements.
- The function _ATAN2 _MAX, _MIN are available from 4.1.20 on. The function _ATAN3 is available from 4.1.30 and above.
- Please note to always insert a space between the keyword and the arguments. Failure to do so may result in an interpretation error : *Ex Arc Tangent* of two should be written *_ATAN 2 and not _ATAN2*. As this was not enforced, existing program could present an incompatibility.

As well as:

Function	**Syntax**
Conversion to degrees	**_DEG**
Smallest integer value greater than or equal to x	**_CEIL**
Largest integer value less than or equal to x	**_FLOOR**
Nearest Integer	**_ROUND**
Remainder of division	**| L1 = 27 | 5** **result : L1 = 2** An error message 94.41 will be issued when the second operand of the Remainder function is null
Exponentiation	**^** **L1 = 2 ^ 5** **-> L1 = 32**
Constant π	**_PI** **L1 = _PI** **-> L1 = 3.141593**

Previously existing functions will have both syntaxes: Their original and the new one:

Function	**Syntax**
Sine	**S or _SIN**
Cosine	**C or _COS**
Arc tangent	**A or _ATAN**
Square root	**R or _SQR**
Truncation	**T or _TRUNC**



It is not possible to write two consecutive functions; each function must be followed by a value.

- L7= _SIN_DEG_PI is forbidden;
- L7= _SIN{_DEG_PI} is authorized.

As stated in the previous chapter, the argument of the previous Sine (S) and Cosine (C) functions are scaled by 1000000 when the input is an E parameter. This is kept for compatibility reasons. But the new functions do not experience this scaling as they natively handle real numbers.

Illustration

```
L1 = 1.732
E80001 = AL1                          \\ E80001 = 60°
E80002 = SE80001                      \\ E80002 = 0.00000105 <> sin(60) (wrong)
E80003= _SINE80001                    \\ E80003 = 0.866 = 60° (correct)

E80004 = E80001 * 1000000             \\ in millionths of degrees
E80005 = SE80004                      \\ E80004 = 0.866 = sin(60) (correct)
```

E80006 = S{E80001 * 1000000} \\ will return a wrong result, **as the term within curly brackets is considered as a value**, there is no scaling.

7.1.3 Logic Operations

Logic operation	Symbol	Logic operation	Symbol
AND	&	OR	!

AND and OR	Before performing the operation, the operands will be automatically truncated. The operation is performed on the binary pattern.

7.1.4 Assignment

Assign value in Binary	_BIN	Assign value in Hexadecimal	_HEX

The value is accepted in uppercase or lower.

Notes

It is possible to have several spaces between the keyword and the data but NO spaces within the data string.

`ex L0 =_HEX f A10 =>` stores the decimal value 15 into L0 and requests A to move to position 10

7.1.5 Comparison Symbols Used with L Variables

Comparison symbol		Comparison symbol	
Equal to	=	Greater than or equal to	> =
Greater than	>	Less than or equal to	< =
Less than	<	Different from	< >

7.1.6 Conversion in Internal Unit

In parametric expressions, functions U and M are used to convert values expressed in the internal system unit (see Chapters 2 and 3) to the programming unit:
- Function U is specific to linear axes
- Function M is specific to rotary axes.

Use of function U

Case of a system with micrometers (µm) as internal unit for linear axes.

Reminder: Programming in inches by G70 or metric system by G71.

- U254000 returns the value 254 for G71 (254 mm)
- U254000 returns the value 10 for G70 (10 inches).

Use of function M

Case of a system with 0.0001 degree as internal unit for rotary axes.

External parameter E74000 defines the reference position of axis 4 (B axis).

Conversion from 0.0001 degree (internal unit) to the programming unit in degrees. For any position on the B axis, the programming is:

- L0=ME74000 or L0=[.IRX(8)]
- Variable L0 will hold the position on the B axis in degrees.

7.1.7 Curly Brackets

Since version 4.1.10.00 the curly brackets {} are available to the user.

These can be used for arithmetic or also for boolean calculations.

7.1.8 Priorities

There is no priority between the different operators and the expressions are evaluated in the order they ware written.

However it is possible to regroup operations with a separator; as the parentheses are used for comments this operator is the curly bracket.

Therefore the following operations are valid:

```
L1 = 10
L2 = 3
L3 = 4
L0 = L1 + L2 * L3                       (L0 = 52)
L0 = L1 + {L2 * L3}                     (L0 = 22)
X{L0 * 2}                               (X44)
X{L1 + {L2 * L3}}                       (X22)
L5 = CL0 + {L1 * {L2 + 4}}              (L5 = 70.927)

L1 = 3
L5 = C{L0 + {L1 * {L2 + 4}}}            (L5 = 0.731)
L5 = 100 * {C{L0 + L1 * {L2 + 4}}}      (L5 = -99.619)
X{100 * {C{L0 + L1 * {L2 + 4}}}}        (X = -99.619)

F{{L1*10} + {L0 * 15}}                  (F360)
M{L1 * 2 + 2}                           (M8)

L1 = 10
L2 = 80000
L11 = 25
L12 = 2
FOR L0= 1 TO {L11 - 5}                  \\ Loop from 1 to 20
E{L2  + L0} = L{L1 + 2}                 \\ E800xx = L12
L12 = L12 + L12                         \\ L12 = 2* L12
ENDF

M2          E80001 = 2
            E80002 = 4
            ...
            E8000n = 2^n
            ...
            E80020 = 1048576 (= 2^20)
```



Curly brackets can also be used in in logical and relational expressions which are present in structured instructions (IF, REPEAT, WHILE, FOR, ...).

7.2 Programming syntax for L variables

Programming syntax for L variables is given as Conway diagrams and followed by programming examples.

Definition

Variables are elements that can be substituted for numerical values to provide a programming aid. Program variables are defined by letter address L followed by a number from one to three digits.

L variables represent decimal numbers are should be used in order to keep the maximum precision on floating point operation (divide, square root, trigonometric etc.)

The highest possible value for a parameter (E or L) is about 1.7*10^308.

List of L Variables

- Variables L0 to L99,
- Variables L100 to L199,
- Variables L900 to L959.

Variables L0 to L99, L100 to L199, L900 to L959 have the same format and use but cause differences in programming (see chapter 7.2).

Variable Assignments

L Variables can be assigned to all the programmable NC addresses.

The assignment of an L variable to an NC address automatically results in the correct use of the unit for the programmed address.

Initialization

The variables are initialised to zero:
- at power on,
- at the end of the program (M02),
- after a reset.

Application

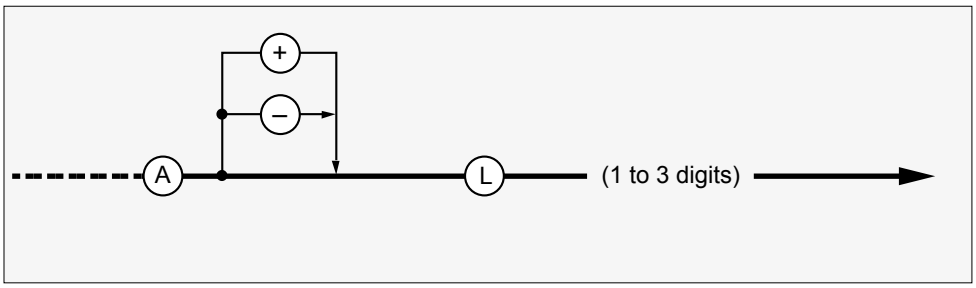
The values assigned to L variables can be:
- integer or decimal numbers (maximum 8 digits plus sign),
- fixed values or values resulting from operations.

Use

L Variables can be used:
- to perform operations,
- for increments and decrements,
- for conditional branches with function G79 after comparison with an expression, use caution with equivalence tests.
- jointly with programming of external E parameters to make transfers.

7.2.1 Assignment of a Variable to an NC function

Syntax



A	NC function.
+	**/ -**Sign.
L	Variable used as numerical value.

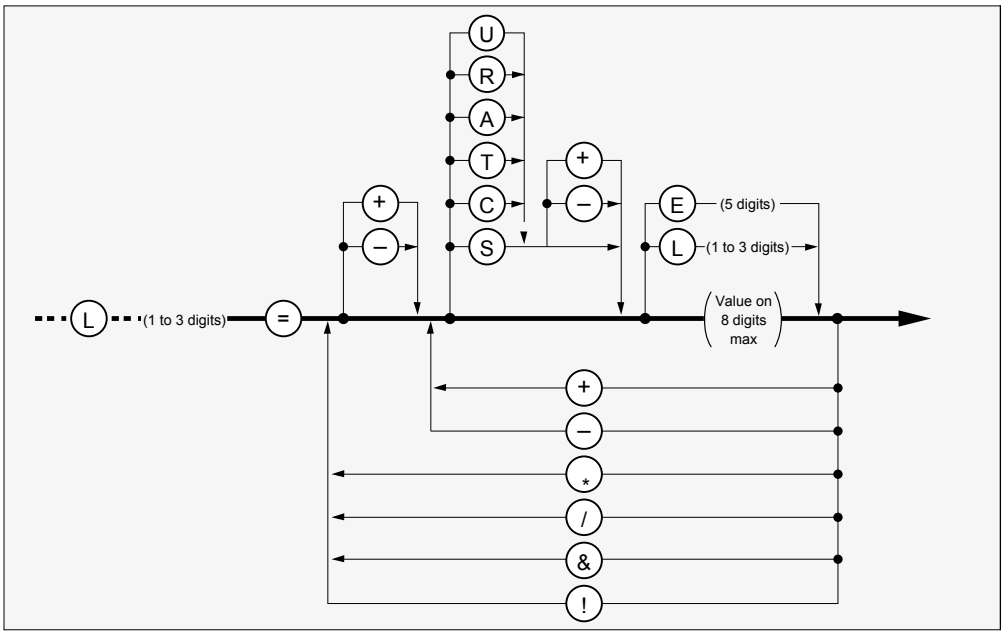
Example

Use of a variable with NC addresses that have different units.

Assignment of variable L5 to addresses X and F.

```
N..
L5 = 18                  Declaration of the value of variable L5
N..
N.. G00 XL5              L5 equals 18 mm with the X address
N..
N80 G94 G01 Z70 FL5      L5 equals a feed rate of 18 mm/min with the F address
N..
```

Declaration of a variable in the Program
Syntax



The operations given in this diagram are detailed in the previous pages.

Notes

When the result of an operation giving a fractional number is assigned to an L variable, the system only keeps the first eight digits (integer and decimal parts) and ignores the rest. If the integer part of the result exceeds eight digits, the system reports an error.

Note on > = < Tests on L parameters

Be careful when making test on L variable. As an L variable is coded in Floating point the actual value can be slightly different from the displayed value. This is due to the principle of floating point calculation and cannot be considered as a default. The difference can be on the 10th or further digit after the decimal point. Therefore when making test on L values it is always better to check boundaries rather than exact value.

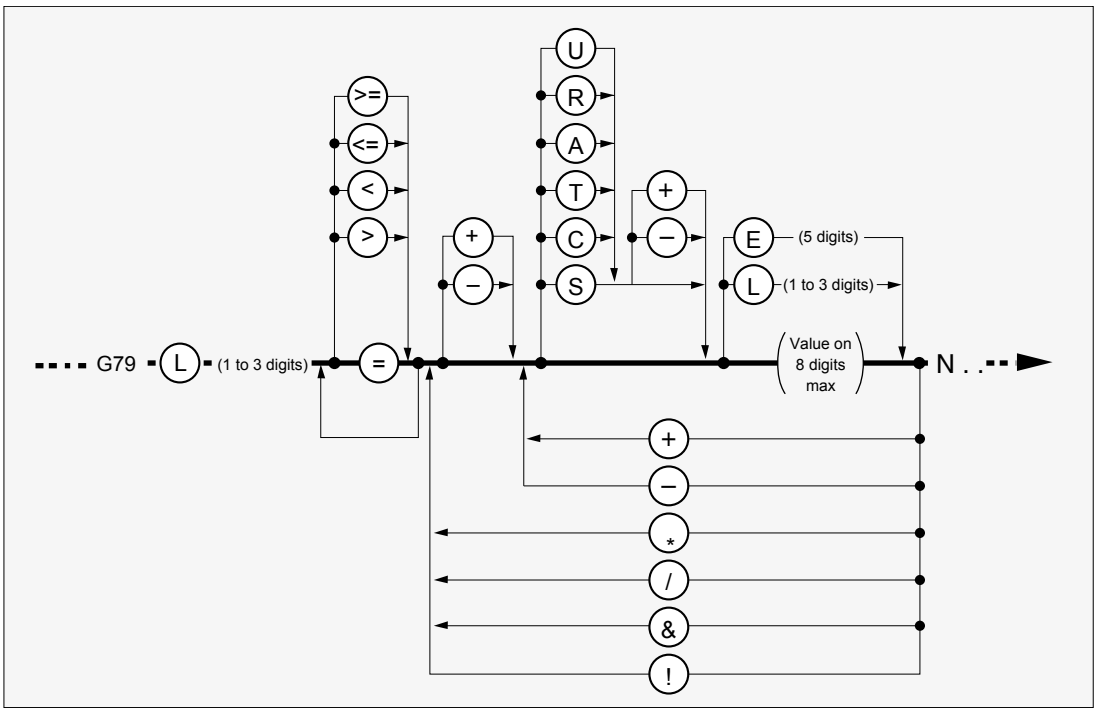
Example

Use of the variables with arithmetic operations.

`N..`	
`L1 = 5`	Declaration of the value of L1
`L2 = L1 + 5.3 * 3 * S30`	After the operation, L2 takes on the value 15.45 (sine 30° = 0.5)
`L3 = 100 / 3`	After the operation, L3 takes on the value 33.333333 (limitation to eight digits)
`N..`	
`N90 G00 XL2 Z30`	The value of L2 (15.45) is assigned to the X axis
`N100 XL3`	The value of L3 (truncated to 33.333 if the resolution is .001) is assigned to the X axis
`N..`	

7.2.2 Test of a variable for a conditional branch

Syntax



Example

Use of a variable with a conditional test on the variable contents.

```
N..
N50 L1 = 0                    Initialization of the variable with zero
N60 L1 = L1 + 1               Increment of the variable
N..
N..
N200 G79 L1 < 10 N60          Condition: if L1 < 10, jump to N60; else continue
N210
N..
```

7.2.3 Notes on programming variables L100 to L199 and L900 to L959

Variables L100 to L199

Variables L100 to L199 have the same format and use as variables L0 to L99 but cause a different in programming.

Writing into a variable L100 to L199 suspends preparation of the block to which it belongs until execution of the previous block is completed. This is to prevent unexpected interaction if an external event (PLC generated subroutine or operator input in MDI for example) occurs that should generate a second block preparation sequence.

A block including a variable L100 to L199 cannot therefore be preceded by a block whose execution requires knowledge of the following block(s), e.g.:
- Profile Geometry Programming (PGP), or
- tool radius offset (G41, G42).

Programming of variables L100 to L199 with function M999

Writing of variables and transfer of current values into the part program are normally suspended until execution of the previous blocks is completed.

Programming M999 prohibits the operator or PLC from intervening in execution of a sequence of blocks, therefore allowing writing into these variables without stopping block preparation.

Variables L900 to L959

It is strongly not recommended to use variables L900 to L959 in a program with canned cycles (G81, G82, etc.), since they risk being overwritten when a cycle is called.

7.2.4 Equivalence of variables L900 to L925

Variables L900 to L925 are equivalent to addresses A to Z respectively.

Example:

A = 250 is equivalent to L900 = 250

B = 1234 is equivalent to L901 = 1234 (and so forth to L925 for Z).

7.2.5 Symbolic addressing of variables L900 to L925 and L926 to L951

Variables L900 to L925 and L926 to L951 can be addressed by alphabetic symbols preceded by «'« (apostrophe). The variables can be used on the left or right side of an equation.

Variables L900 to L925

Variables L900 to L925 can be addressed by symbols 'A to 'Z respectively.

Example:

'C = 'A + 'B is equivalent to L902 = L900 + L901

C = 'A+'B is equivalent to L902 = L900 + L901

C = A+B not permitted.

Variables L926 to L951

Variables L926 to L951 can be addressed by symbols 'EA to 'EZ respectively
('EA = L926, 'EB = L927, etc. down to 'EZ = L951).

Example:

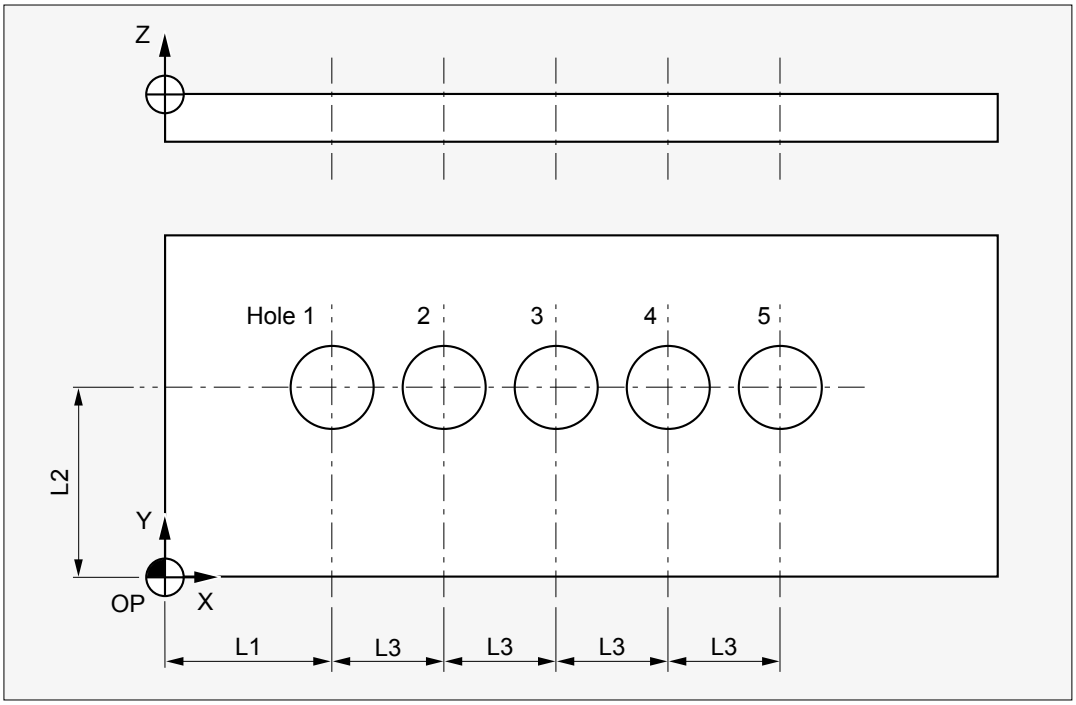
'A = 'B - 'EA/'EZ is equivalent to L900 = L901-L926/L951

Examples

Drilling holes with a conditional test

Execution of five holes in the XY plane (G17) with programming of a single drilling cycle.

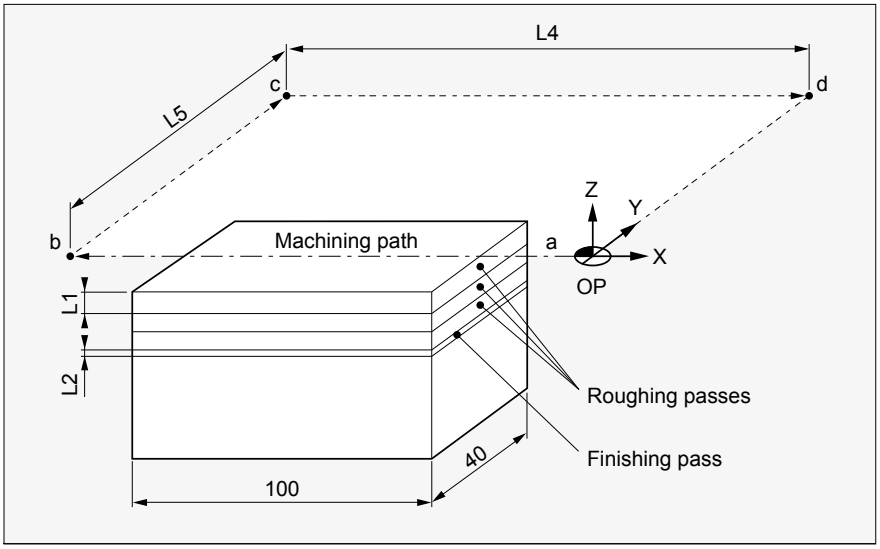





```
%40
L1=20 (START ON X)
L2=25 (START ON Y)
L3=15 (INCREMENT ON X BETWEEN HOLES)
N10 G00 G52 Z-50
N20 T01 D01 M06 (DRILL)
N30 S600 M40 M03
N40 XL1 YL2                    Positioning in the axis of hole 1
N50 L4=L4+1
N60 G81 Z-10 ER2 F100          Drilling in cycle
N70 G00 G80 G91 XL3            Movement in increment
N80 G90 G79 L4 < 5 N50         Condition: if L4 < 5, branch to block N50, else continue
N90 G00 G80 Z200 M05
N100 M02
```

Calculating the number of passes required
Facing in the XY plane (G17) by consecutive roughing and finishing passes in Z.
The diameter of the cutter is greater than the width of the part.

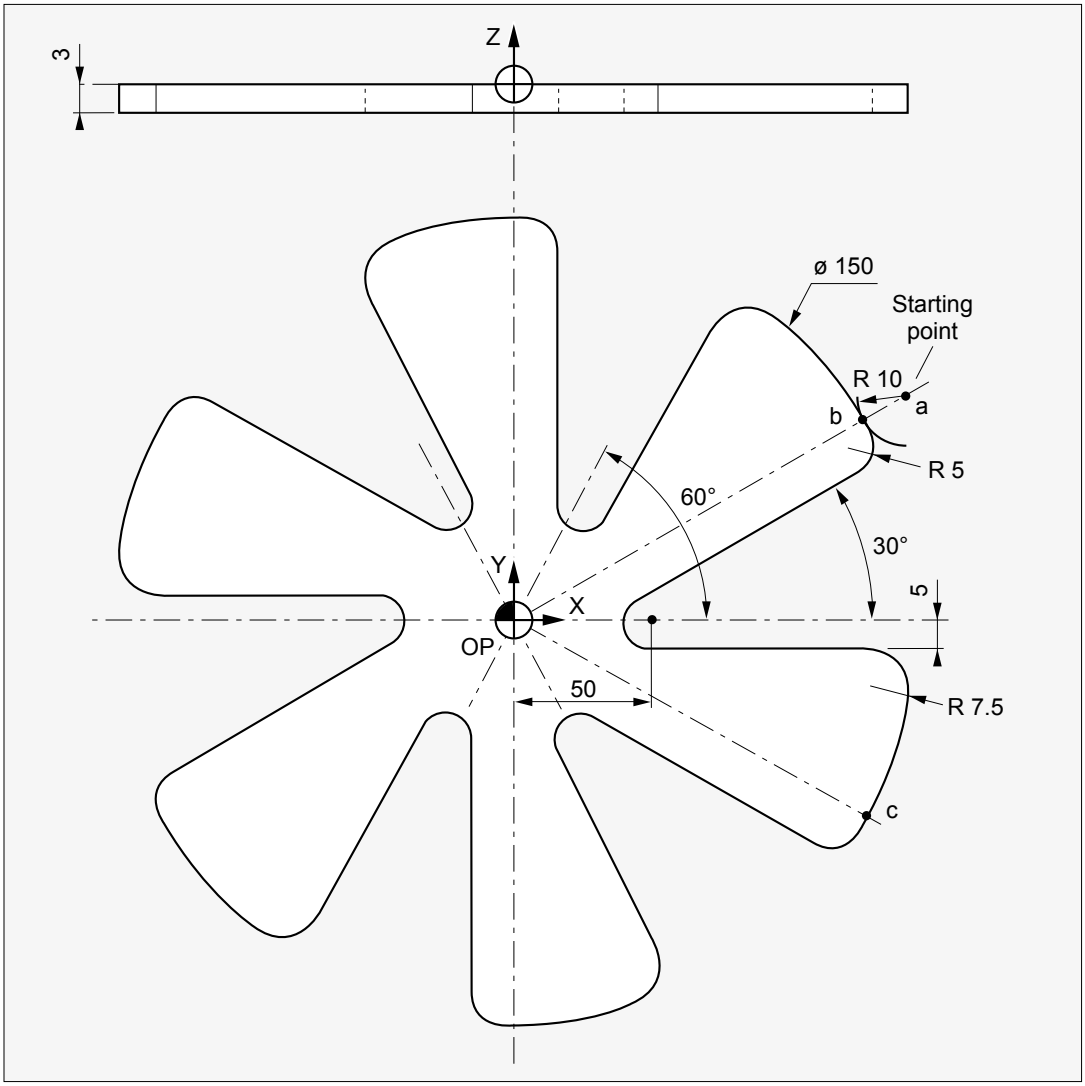
Program origin:
- X OP = Cutter radius plus clearance,
- Y OP = Cutter axis in the part axis,
- Z OP = Unmachined top surface of the part.




```
%25
$0 FACING Program
L1= 3 (Z VALUE OF ROUGHING PASS)
L2= 0.2 (Z VALUE OF FINISHING PASS)
L3= 3 (NUMBER OF ROUGHING PASSES)
L4= 200 (PASS LENGTH ON X)
L5= 100 (RETRACTION ON Y FOR RETURN)
L6= 150 (ROUGHING ROTATION SPEED)
L7= 120 (ROUGHING FEED RATE)
L8= 150 (FINISHING ROTATION SPEED)
L9= 80 (FINISHING FEED RATE)
N10 G00 G52 Z-50
N20 T01 D01 M06 (FACING CUTTER DIAMETER=50)
N30 SL6 M40 M03
N40 X0 Y0 Z30                          Point a
N50 Z0                                 Approach on Z0
N60 G91 Z-L1                           Pass setting on Z
N70 G01 X-L4 FL7 M08                   Point b
N80 G00 YL5                            Point c
N90 XL4                                Point d
N100 Y-L5                              Point e
N110 L3 = L3 - 1                       Decrement
N120 G79 L3 < 0 N140                   Condition: if L3 < 0, branch to block N40, else continue
N130 G79 N60
$0 FINISHING
N140 G00 Z-L2                          Finishing pass setting on Z
N150 G01 X-L4 SL8 FL9                  Finishing
N160 G00 G90 Z0 M05 M09
```

Calculating positions

Finishing of the profile of a gear wheel with use of PGP in the XY plane (G17)



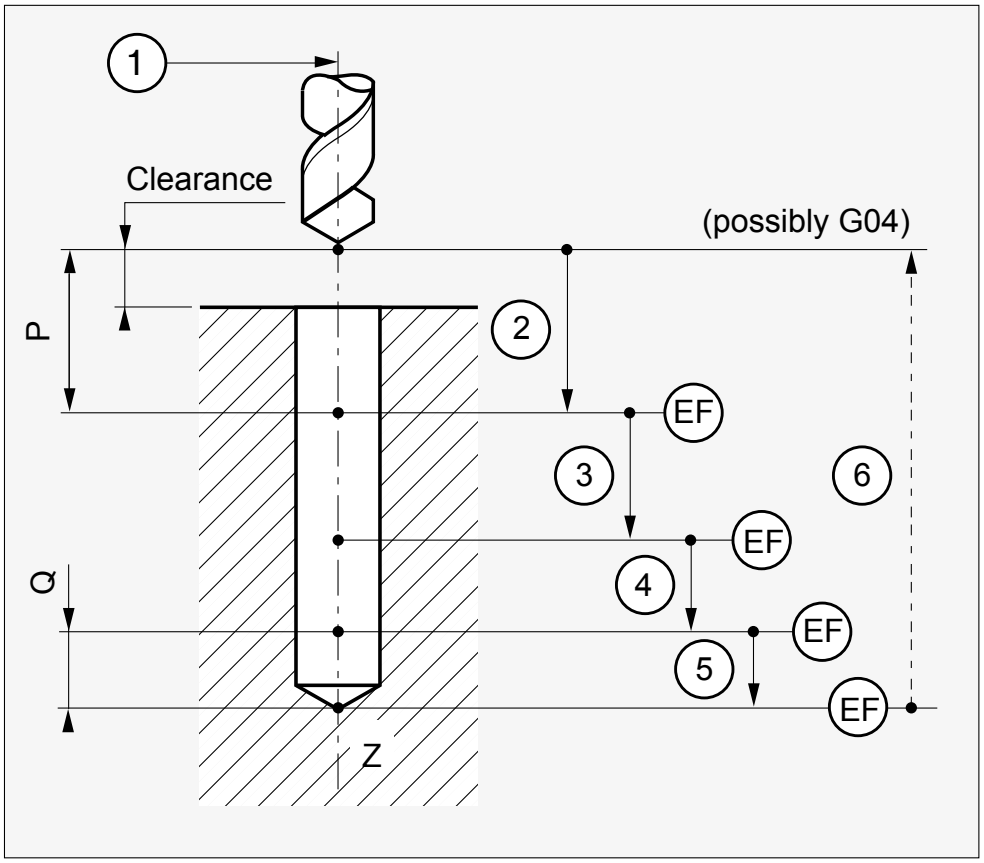

```
%21
N10 G00 G52 Z0
N20 T12 D12 M06 (CUTTER DIAMETER=8)
N30 S1000 M40 M03
L0 = 75 + 10 * C30                  Calculation of start point a on X
L1 = 75 + 10 * S30                  Calculation of start point a on Y
N35 G92 R1                          Limiting of tangential feed rate
N40 XL0 YL1 Z3 M08                  Point a, approach
N50 G01 Z-4 F50                     Approach on Z
N60 G41 EA180 ES+ F200
L3 = 75 * C30                       Calculation of point b on X
L4 = 75 * S30                       Calculation of point b on Y
N70 G03 XL3 YL4 IL0 JL1
N80 G02 I0 J0 ES- EB5
N90 G01 EA 210
```

```
N100 G03 I50 J0 R5
N110 G01 EA0 ES+ EB7.5
N120 G02 XL3 Y-L4 I0 J0
N130 EDL10
N140 G79 L10 > -360 N80
N150 G03 IL0 JL1 R10 ES-
N160 G00 G40 EA90 XL0 YL1
N170 G40 G52 Z0 M05 M09
N180 M02
```

Point c
Angular offset
Test on 360°

Drilling holes on a circle

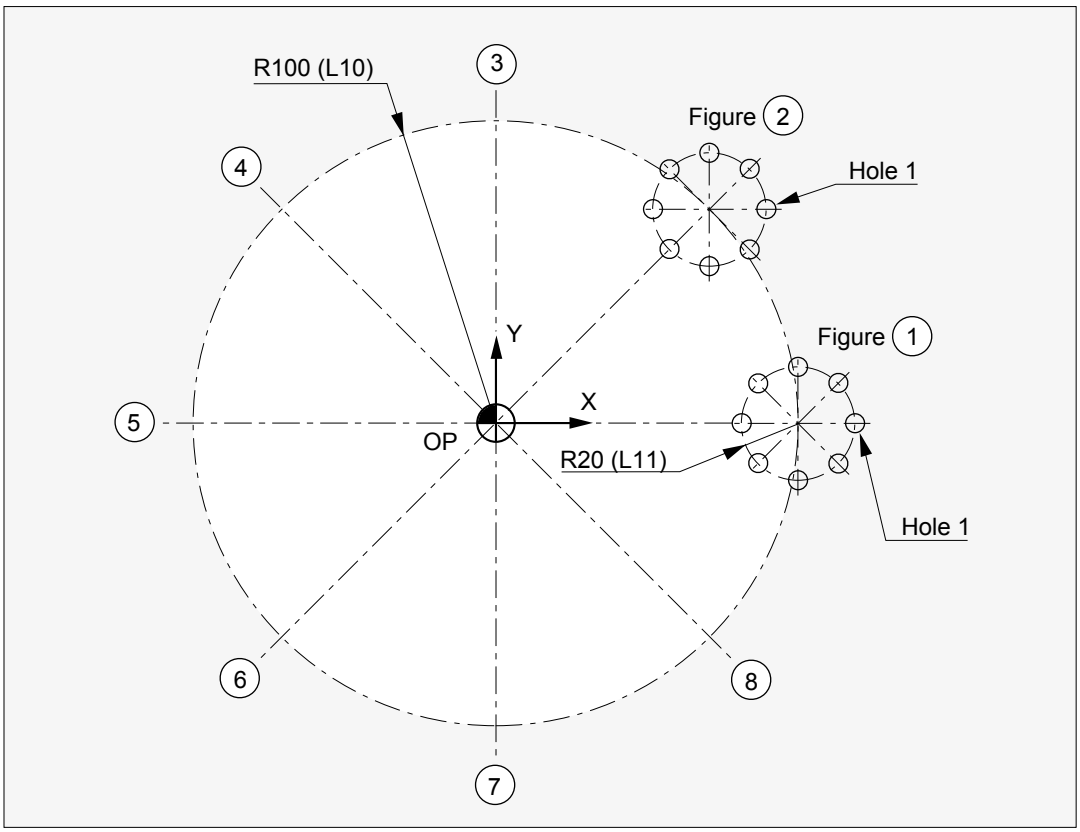
Movement by circular interpolation between holes.




```
%90
N10 G00 G52 Z0
N20 T01 D01 M06 (DRILL)
N30 S500 M03 M40
N40 L1=100 (RADIUS)
N50 L2=5 (APPROACH Z)
N55 L3=-5 (DRILLING Z)
N60 L4=30 (ED)
N65 L5=12 (NUMBER OF HOLES)
N70 X0 YL1                          Point a, approach on XY
N80 ZL2
N90 G79 N150                        Jump to block N150
N100 G81 ZL3 F100                   Cycle
N110 G80 G91 EDL4                   Angular offset by 30 degrees
N120 L10=L10+1                      Increment
N130 G79 L10=L5 N160                Condition: if L10 = 5, jump to block N160
N140 G90 G03 X0 YL1 I0 J0 F5000     Circular movement
N150 G77 N100 N140 SL5              Repeat the cycle
N160 G00 G52 Z-50
N170 G52 X-100 Y-50 M05
N180 M02
```

Calculating positions, origin offsets and angular offsets

Drilling cycle with 8 circular patterns each including eight 6 mm diameter holes.



Main Program

```
%92
N10 G00 G52 Z0
N20 G52 X0 Y0
N30 T11 D11 M06 (DRILL DIA. 6)
N40 S4500 M03 M40
N50 L10=100 (RADIUS 1) L11=20 (RADIUS
2)
N60 L0=0 L1=L10*CL3 L2=L10*SL3 G59
XL1 YL2
N70 G77 H93                          Subroutine branch
N80 L3=L3+45 G79 L3<360 N60          Test on 360 degrees
N90 G00 G52 Z0 M05
N100 M02
```

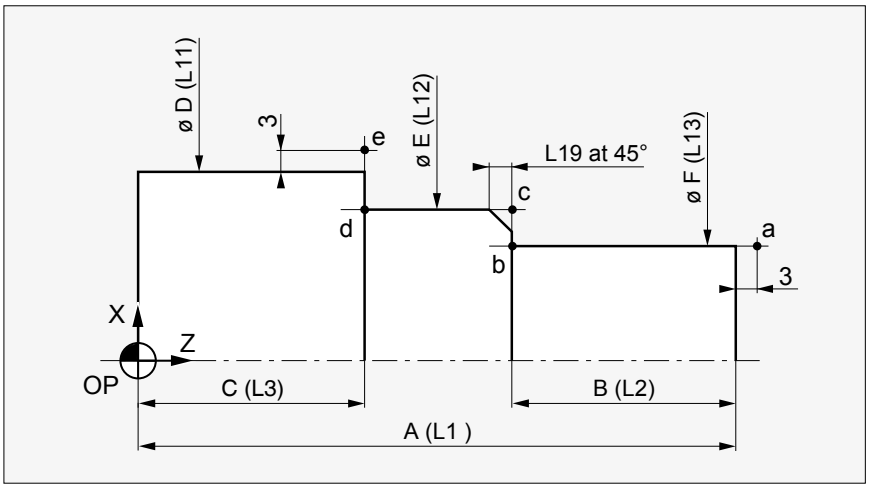
Subroutine

```
%93
N10 EDL0 G81 XL11 Y0 ER2 Z-10 F500   Cycle
N20 L0=L0+45 G79 L0<360 N10          Test on 360 degrees
N30 G80 G00 Z5
```

Examples

Programming L Variables

Use of variables for machining parts of a family
Machining of part 2 according the table.

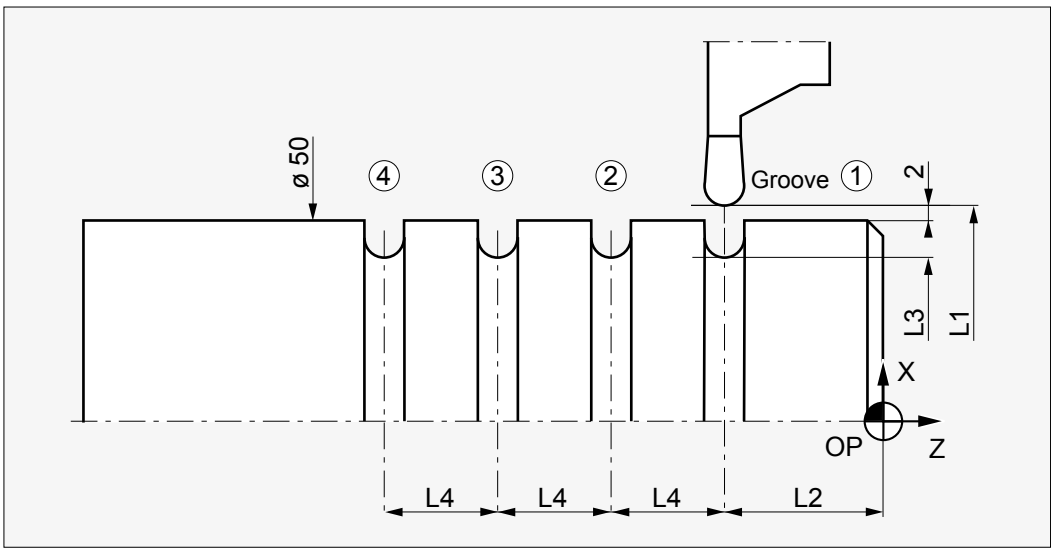


Variable assignment

Variables	L1	L2	L3	L11	L12	L13	L19
Dimension	A	B	C	D	E	F	Chamfer
Part 1	80	30	30	50	40	30	2
Part 2	85	28	35	48	39	28	1.5

```
%18
L1=85 (LENGTH A)
L2=28 (LENGTH B)
L3=35 (LENGTH C)
L11=48 (DIAMETER D)
L12=39 (DIAMETER E)
L13=28 (DIAMETER F)
L19=1.5 (CHAMFER)
N10 G00 G52 X0 Z0
N20 T01 D01 M06 (EXTERNAL TOOL)
N30 G97 S900 M40 M03
N40 G95 F0.2
L4=L1+3                          Computation of point Za
N50 G42 XL13 ZL4                 Point a
N60 G92 S2000
N70 G96 S160
L5=L1-L2                         Computation of point Zb
N80 G01 ZL5                      Point b
N90 XL12 EB-L19                  Point c
N100 ZL3                         Point d
L6=3*2+L11                       Computation of point Xe
N110 XL6                         Point e
N120 G40 G77 N10 N10 G97 S900 M05
N130 M02
```

Use of variables with conditional test on the number of grooves
Execution of four grooves by programming a single G83 cycle.




```
%42
L1=54 (START X)
L2=20 (START Z)
L3=40 (GROOVE BOTTOM DIAMETER)
L4=15 (STEP BETWEEN GROOVES)
L10=50 ((CONSTANT SURFACE SPEED)
L11=0.1 (FEED RATE))
N10 G00 G52 X0 Z0
N20 G16 P+
N30 T01 D01 M06 (GROOVING TOOL)
N40 G97 S900 M40 M03
N50 XL1 Z-L2                    Groove 1 axis position
N60 L5=L5+1                     Increment a counter
N70 G92 S2000
N80 G96 SL10
N90 G83 XL3 P4 Q1 G95 FL11      Groove in cycle
N100 G00 G80 G79 L5=4 N130      Condition: if L5=4 go
                                to block N130
N110 G91 Z-L4                   Spacing between two grooves
N120 G90 G79 N60                Branch to execute the next groove
N130 G77 N10 N10 G97 S900 M05
N140 M02
```

7.3 Programming syntax for E variables

E variables are very similar to L variables, they can also be expressed using the decimal point.

There is however a very important point to consider.

E variables generally depend about external conditions, therefore they are correctly evaluated only when they are needed.

Due to the look ahead false interpretation could occur if those external conditions change between block preparation and block execution. For this reason the block preparation is halted at each operation on an E parameter (except writing into an E8xxxx in M999 status).

The result is that it is not possible to deal with such parameters in case of cutter compensation because cutter compensation requires knowledge of the following blocks.

7.4 Address equivalences



This table is used to specify the equivalence between two axes. For instance, it allows a program to be written for another channel, such as UW.

The symbol @ followed by an address (A to Z) specifies an equivalent axis.

The declaration of an equivalent address is programmed by:

@ [A " " Z] = [A " " Z]

The assignment of a value to an equivalent function is programmed by:

@ [A " " Z] followed by the value

Example :

```
@ X = U
```

@ X/300 is equal to U 300

The address equivalence table is reset at power on, by a reset or by M02 (@A = A, @B = B, ..., @Z = Z).

The declaration of a new equivalent address (@ [A " " Z] = [A " " Z]) suspends preparation of the block in which it is programmed until execution of the previous block is completed.

Example :

```
G0 X0 Z0 U0 W0

@X  = X
@Z  = Z
G0 @X250 @Z100

$check 1    Physical axes X and Z moved
M0

@X  = U
@Z  = W

G0 @X250 @Z100

$check 2  Physical axes U and W moved
M0

M2
```

7.5 Errors and Warnings

During parametric programming it is possible to generate errors or warnings if the result of an operation doesn't match with the expected value for example.

Contrary to a warning, an error will stop part program execution and can only be acknowledged by a reset.

A text can be associated with a warning or an error. This requires an action in the internal system files of the system and is generally reserved to the OEM.

In case the user wants to associate a text with a warning or an error, the easiest way is generally to first introduce this text with a $ message.

Error syntax

E.xxx /yy

xxx error number

yy error extension number (additional precision)

Warning syntax

EW. xxx

xxx <2000 the warning are self cleared after a certain time

xxx >= 2000 the warning is permanent and can be cleared by EW.- xxx

The list of errors and warnings reserved by NUM are indicated at the end of this manual.

However the machine builder can complete this list.

Make sure that the errors or warning numbers you are using are consistent and do not interfere with the existing.

Page deliberately empty

Page deliberately empty

Summary

8 External E Parameters 523
8.1 E Parameters description 523
8.1.1 External E Parameter programming syntax.....525
8.1.2 Assignment of an External Parameter to an NC Function.....525
8.1.3 Declaration of an External Parameter in the Program.....526
8.1.4 Test on an External Parameter for a conditional branch 527
8.2 How to read an E parameters table description 528
8.2.1 Parameter format and values 529
8.3 E Parameters - List of E parameters sorted by function..... 530
8.4 List of detailed E parameters sorted numerically 542

Page deliberately empty

8 External E Parameters

8.1 E Parameters description



Definition

External E parameters are used by the part program to access information contained in the PLC or CNC memory.

External parameters are defined by address letter *E* followed by five digits.



Contrary to previous NUM control systems family, E parameters are floating point (real number) values to bring additional precision.
When related to moves, positions, tools, etc. they represent internal units as before. However caution is required when they are used in calculations (previously a rounding or truncation was by construction systematic). A verification is necessary in case of existing programs.

Examples: (system in metric - precision 0.001mm)

Sequence	Result on Flexium	Result on Flexium+
L0 = 123 XL0	 X123.000	 X123.000
E80000 =1234 YE80000	 Y1.234	 Y1.234
E80001 = 255 E80002 = E80001/2 E80003 = E80002 / 2	255 128 256	255 **127.5** **255**
L0=0.978 L1=AL0 E80000 = AL0	0.978 44.362763 44	0.978 44.362763 **44.362763 (ArcTangent)**
E80000 = 255 E80001= E80000 /2 E80002= E80001 * 2 E80003= E80002 & 1	255 128 256 0	255 **127.5** **255** **1 (Logical AND)**

Application

Depending on the external E parameters, the part program accesses the CNC or PLC memory for *Read only or Read/Write* operations.

Assignment

The assignment of an external parameter to an NC address causes the same units to be used for the E parameter and the programmed address.

Initialization

The parameters E100xx, E200xx, E30xxx, E33xxx, E34xxx, E36xxx, E37xxx, E40xxx, E43xxx, E44xxx, E46xxx, E47xxx, E41xxx, E41001, E41002, E83000 and E84000 are initialized on power-up. The other external parameters are never reset by the system.

Use of the External E Parameters

The external parameters can be used:
- for operations,
- for increments and decrements,
- for conditional branches with function G79 after comparison with an expression,
- jointly with L variables to make transfers.

Precautions for Use

The use of the external parameters obeys certain rules:
- parameters E20000, E40000, E41000, E41001, E41002, E70000 cannot be *«written»* by the program as they are read-only parameters;
- an operation on an external parameter stops movement at the end of the previous block;
- a block including an external parameter cannot be preceded by a block whose execution requires knowledge of the following block(s).

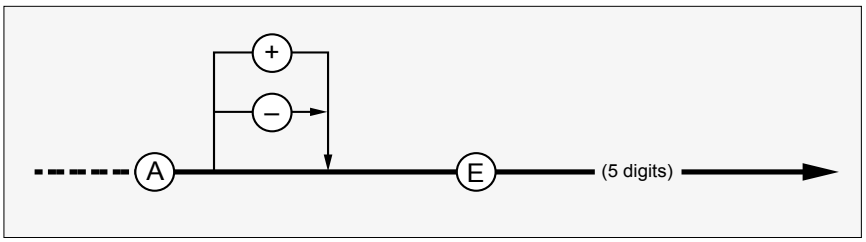
  Example
  - Profile Geometry Programming (PGP),
  - Tool radius offset (G41, G42).

8.1.1 External E Parameter programming syntax

The programming syntax for external E parameters is given as Conway diagrams followed by programming examples.

8.1.2 Assignment of an External Parameter to an NC Function

Syntax



A	NC function
+ / -	Sign
E	External parameter used as numerical value

Notes

External E parameters have integer values. When they are assigned to a function with a decimal value, the decimal point is implicit and depends on the format of the function.

When the value of the parameter is not compatible with the format of the function (too many digits, etc.), the system reports an error.

Example

Use of an external parameter with NC addresses in different units.

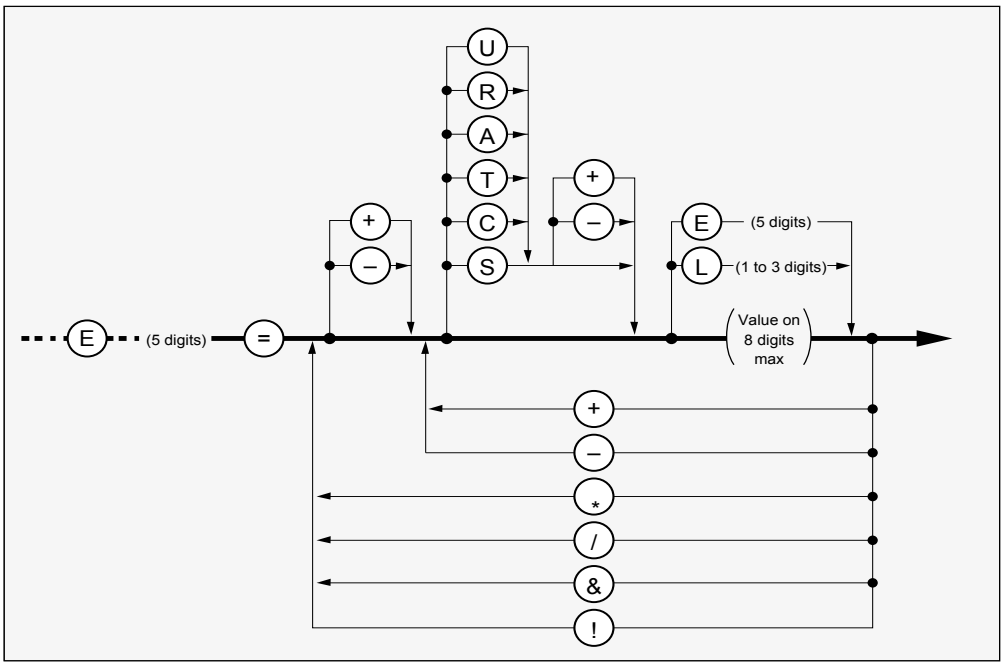
Assignment of external parameter E80000 to addresses X and B (C for Turning).

```
N..
E80000 = 18000          Declaration of the value of the parameter
G00 XE80000             The value of E80000 is equivalent to 18 millimetres
                        (internal unit = µm)
G00 BE80000             The value of E80000 is equivalent to 1.8 degrees
                        (format B034, i.e.  ten-thousandths of a degree)
```

It should be noted that if the feed rate F is programmed as FE80000 (example), the feed rate F takes on the integer value declared in parameter E (in this case, the decimal digits are excluded with F).

8.1.3 Declaration of an External Parameter in the Program

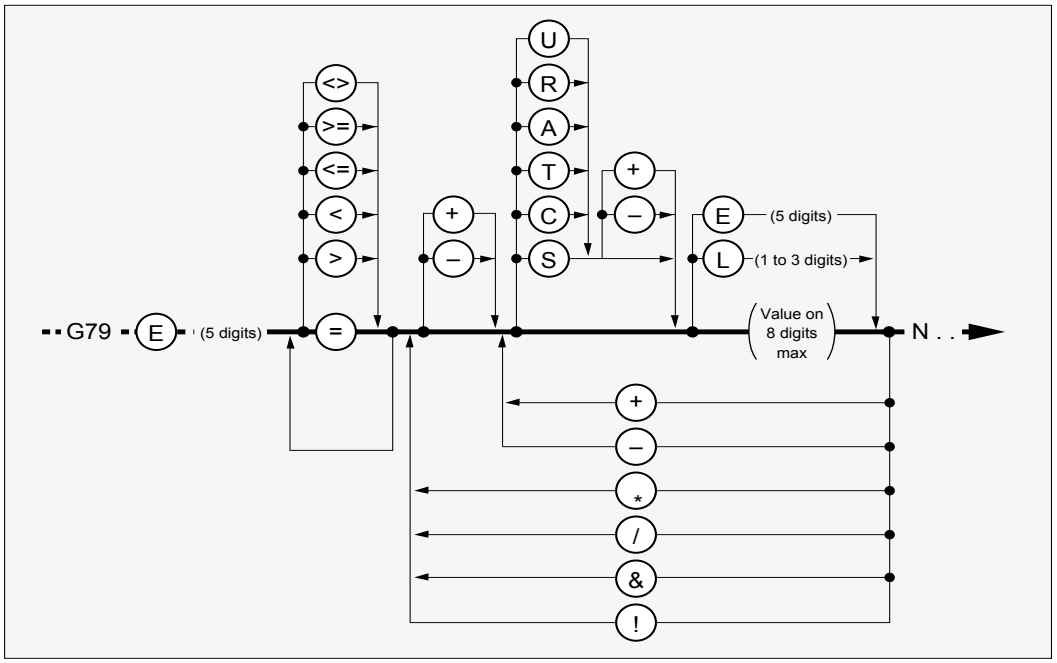
Syntax



The operations given in this diagram are detailed in the previous pages.

8.1.4 Test on an External Parameter for a conditional branch

Syntax



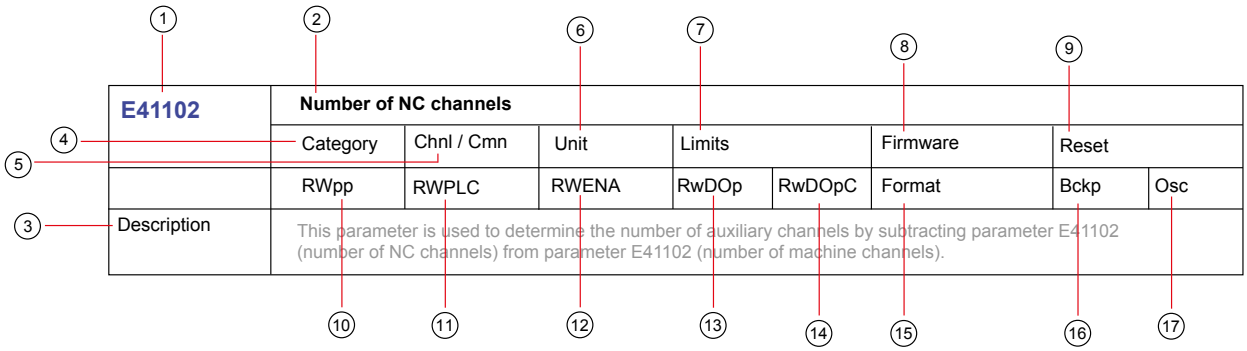
Example

Use of an external parameter with a conditional test.

```
N.. ...
N50 E56003 = E56003+1
G79 E56003 = > 10 N50
N..
```

Increment of the parameter
Condition: if E56003 ≥10, branch to N50, else continue

8.2 How to read an E parameters table description



1	E parameter number
2	E parameter description
3	E parameter label
4	Category. See list of category on chapter 8.4.
5	Global or channel specific parameter
6	Unit of parameter
7	Limit of parameter
8	Parameter available from software version (a '-' means available before version 4)
9	Parameter affected by reset.
10	Access by part program (Read, Write or Read/Write).
11	Access by PLC.
12	Access by ENA function.
13	Access by classic dynamic operators.
14	Access by classic dynamic operators in C.
15	Parameter format for access by dynamic operators in C.
16	Parameter saved in Power Off or. Default value (*).
17	Parameter can be used by NCK MultiBuffer / Oscilloscope (internal use). - means currently not available.

(*) - Value: the startup value is defined externally (e.g. Analog input).
- Last value : the parameter value before shutdown.
- PLC value : the power up value is defined by the PLC.

Read Access by Dynamic Operators in C is always possible.
Write access follows the same rules as access by Part Program.

8.2.1 Parameter format and values

UINT 8	8 bits unsigned integer	0.. 255
SINT 8	8 bits singed integer	-128.. 127
UINT16	16 bits unsigned integer	0..65535
SINT16	16 bits signed integer	-32768..32767
UINT32	32 bits unsigned integer	0..4294967295
SINT32	32 bits signed integer	-2147483648..2147483647
DOUBLE	64 bits floating point	

8.3 E Parameters - List of E parameters sorted by function

The table below lists all parameters sorted by function (alphabetical).
The following table lists the parameters in numerical order.

The different functions are indicated in the table as follow:

ALL	General purpose parameters
Axes	Axes related values
Cutting	Cutting (Jet) parameters
Debug	Reserved parameters
Drives	Drives parameters
EGB	Electrical Gear Box parameters
Gantry	Gantry parameters
HSC	High speed cutting related parameters
Info	Parameters returning machining information
I/O	I/Os related parameters
Prg	Part program related parameters
RTCP	RTCP parameters
Servo	Servo adjustments parameters
Spindles	Spindles related parameters
Synchro	Spindle synchronization parameters
SSB	SSB parameters (Smart Servo Balance)
Tools	Tools related parameters

Click on **Exxxx** to give the E parameter detailed information. See next pages.

Axes related parameters		
E parameter	**Function**	**Description**
E11012	Axes	Zero following error operation
E6x000	Axes	DAT1 on axis x (x = axis address 0 to 8)
E6x001	Axes	DAT2 on axis x (x = axis address 0 to 8)
E6x002	Axes	Minimum dynamic machine travel on the axis "x"
E6x003	Axes	Maximum dynamic machine travel on the axis "x"
E6x004	Axes	Eccentricity offsets on axis "x" (DAT3)
E6x005	Axes	Offset programmed by G59 on axis "x"
E6x008	Axes	Monitoring a minimal travel
E6x009	Axes	Monitoring a maximal travel
E6x012	Axes	Angular offsets on RTCP rotary axes
E6xnn0	Axes	DAT1 offsets table number nn applied on axis x
E6xnn1	Axes	DAT2 offset table number nn applied on axis x
E7x001	Axes	Sampling the axes positions for on-the-fly probing
E7x002	Axes	Minimum static machine travel
E7x003	Axes	Maximum static machine travel
E7x004	Axes	Direction of movement of the axes being interpolated
E7x005	Axes	Axes address re-assignment
E7x006	Axes	Axis coupling
E7x007	Axes	Axes programmed by diameter
E7x008	Axes	Axes with movement outside monitored dynamic limits
E7x009	Axes	Position recorded on the fly in program referential (if E11028)
E7x100	Axes	Interpolator position references
E7x101	Axes	Interpolation speed limiting
E79002	Axes	Value of the feed rate potentiometer assigned to the channel
E79003	Axes	Distance remaining to be travelled in the block being executed
E79004	Axes	Current feed rate on the path programmed in the block
E79005	Axes	Minimal value for feed override
E79006	Axes	Maximum value for feed override
E79009	Axes	Distance to go within the block
E79010	Axes	Remaining distance within the block
E79011	Axes	Expansion factor
E79012	Axes	Current feed rate on the path programmed in the block NOT filtered
E79013	Axes	Feed rate override writeable with the Dyn Operator
E79014	Axes	Maximum feed in feed mode (reserved)
E79020 to E79022	Axes	Tool vector orientation in machine referential - valid in RTCP
E79023 to E79025	Axes	Tool vector orientation in turntable referential - valid in RTCP

E79026 to E79028	Axes	Tool vector orientation in program referential - valid in RTCP
E90000 to E90031	Axes	Physical axis measure value
E90032 to E90063	Axes	Lag error
E91000 to E91031	Axes	Servo or non-servo status of the axes
E91100 to E91131	Axes	Homing status on an axis
E91200 to E91231	Axes	N/M AUTO axes
E91300 to E91331	Axes	Clampable axis enable state
E91400 to E91431	Axes	Homing forced by user on an axis
E92000 to E92031	Axes	Axis origin switch enable state (analog axes only)
E92100 to E92131	Axes	Axis error mask (Delayed Power down)
E92200 to E92231	Axes	Scaling the lag display (Do not use anymore)
E92300 to E92331	Axes	List of filtered axes (Tau filter)
E92400 to E92403	Axes	Handwheels errors mask
E93000 to E93031	Axes	Axis @xx home switch status
E93100 to E93131	Axes	Measured axis
E93200 to E93231	Axes	Axis declared as rotary modulo 360 degrees
E93232 to E93263	Axes	Modulo value of axes
E93300 to E93331	Axes	Axis homing direction
E93400 to E93431	Axes	Homing without switch (analog axes)
E93500 to E93531	Axes	Axis in-position window
E93700 to E93731	Axes	Flag Axis in error
E93800 to E93803	Axes	Flag Handwheel in error
E94000 to E94031	Axes	Inter-axis calibration definition
E94100 to E94131	Axes	Association of a slave axis with a master axis
E94200 to E94231	Axes	Association Drive/ axis or Spindle for commutation
E94300 to E94303	Axes	Physical handwheel coupler
E94400 to E94431	Axes	Torque and Reference control (0=NCK; 1=FXTools/FXServer)
E95000 to E95031	Axes	Axis reference correction
E95100 to E95131	Axes	Position of the origin switch with respect to the machine origin
E95200 to E95231	Axes	Axis (or spindle) measurement correction
E95300 to E95331	Axes	Differential offset value
E95400 to E95431	Axes	Measure correction for “volumetric error compensation” (Vecomp)
E95500 to E95531	Axes	Measure correction from PLC (via ENA)
E95600 to E95631	Axes	Measure correction from interaxes calibration
E95700 to E95731	Axes	Measure correction from axes calibration plus backlash
E97000 to E97031	Axes	Maximum axis speed
E97100 to E97131	Axes	Axis acceleration at programmed feed rate
E97200 to E97231	Axes	Axis acceleration in Rapid

E97400 to E97431	Axes	Dynamic axis acceleration at work rate
E97500 to E97531	Axes	Dynamic axis acceleration in rapid
E97600 to E97631	Axes	Minimum machine travel on X axis
E97700 to E97731	Axes	Maximum machine travel on X axis
E98000 to E98031	Axes	Servo coefficient value
E98400 to E98431	Axes	Delay to apply the drive reference (Do not use anymore)
E98500 to E98531	Axes	Partial feed forward per axis (without HSC)

Cutting parameters		
E parameter	**Function**	**Description**
E69005	Cutting	Tilted nozzle compensation : Tilt angle value
E69006	Cutting	Tilted nozzle compensation : Height of the part
E69007	Cutting	Tilted nozzle compensation : Radius smoothing for jet cutting
E69008	Cutting	Tilted nozzle compensation : Type of head for jet cutting
Drives parameters		
E parameter	**Function**	**Description**
E35100 to E35131	Drive	First 32 bits test Drive point value
E35132 to E35163	Drive	Second 32 bits Drive test point value
E35200 to E35231	Drive	Code number of the first test point
E35232 to E35263	Drive	Code number of the second test point
E35264 to E35295	Drive	Enable the reading test point
E35300 to E35363	Drive	Select a set of Drive parameters

Gantry parameters		
E parameter	**Function**	**Description**
E96000 to E96031	Gantry	Duplicated axis in automatic modes
E96100 to E96131	Gantry	Duplicated axis in manual mode (JOG)
E96200 to E96231	Gantry	Synchronized axis
E96300 to E96331	Gantry	Symmetrically driven axis
E96400 to E96431	Gantry	Axes synchronization on the fly with null offset

High speed cutting parameters		
E parameter	**Function**	**Description**
E11013	HSC	Type of acceleration
E32003	HSC	Minimal angle for speed analysis
E32004	HSC	Chordal error
E32005	HSC	Number of total speed anticipation filter terms
E32006	HSC	Time to reach maximum acceleration
E32008	HSC	Time to reach a maximal jerk (derivative of the acceleration)

E32009	HSC	Feed rate limitation in case of curve radius variation
E32010	HSC	Coefficient (in 1/256) of the maximum radial acceleration
E32011	HSC	Number of terms of the speed averaging filter
E97300 to E97331	HSC	Maximum speed authorized at angle crossing
E98100 to E98131	HSC	Axis acceleration anticipation time constant
E98200 to E98231	HSC	Amplitude of the anti-stick pulse on reversal
E98300 to E98331	HSC	Time constant to absorb the anti-stick pulse
E98532 to E98563	HSC	Partial feed forward per axis (with HSC)
I/Os parameters		
E parameter	**Function**	**Description**
E34000 and E34001	I/O	SINT8 parameters to write the NCK embedded digital outputs
E34010 and E34011	I/O	SINT8 parameters to read the NCK embedded digital outputs value
E34100 and E34101	I/O	UINT16 bits parameters to write the NCK embedded analog outputs
E34110 and E34111	I/O	UINT16 bits parameters to read the NCK embedded analog outputs value
E34200 and E34201	I/O	Digital Output Locked by PLC
E34300 and E34301	I/O	Analog Output Locked by PLC
E36000 and E36002	I/O	UINT16 parameters to write the PLC memory
E36003 and E36999	I/O	UINT16 parameters to write the PLC memory
E43000 to E43009	I/O	Indirect addressing of the PLC Inputs
E43010 to E43999	I/O	SINT8 Addressing PLC Inputs
E44000 and E44001	I/O	SINT8 parameters to read the NCK embedded digital inputs
E44100 to E44103	I/O	UINT16 parameters to read the NCK embedded analogues inputs

SSB parameters (Smart Servo Balance)		
E parameter	**Function**	**Description**
E98600 to E98631	SSB	Anticipation time constant
E98632 to E98663	SSB	Delay filter time constant (last 32 values)
E98664 to E98695	SSB	Acceleration anticipation time constant for corrector SSB filter

Part program parameters		
E parameter	**Function**	**Description**
E11000	Prg	Angular offset (function ED..) enabled
E11001	Prg	Program origin offset (function G59) enabled
E11003	Prg	Processing of mirror function (function G51) enabled
E11005	Prg	Programming by diameter enabled
E11006	Prg	Tool center programming (Turning Machine)
E11007	Prg	Spindle speed potentiometer enabled
E11008	Prg	Drawing of a Complete Circle

E11009	Prg	Enables or inhibits the full revolution on rotary axes
E11010	Prg	Shortest rotation direction selection
E11011	Prg	Indexed names of axes enabled
E11014	Prg	Deceleration over several blocks
E11015	Prg	Angle crossing management enabled
E11016	Prg	Selects front or rear turret (Turning Machine)
E11017	Prg	Inclined plane function enabled (G24+)
E11018	Prg	RTCP function enabled (G26+)
E11019	Prg	Storing axes position on hardware interrupt
E11020	Prg	Allows writing the reference position for a "Virtual" axis
E11021	Prg	Activation of the speed averaging filter outside HSC
E11022	Prg	Emergency retract cause
E11023	Prg	EGB leader axis in emergency
E11024	Prg	Axes and Spindle behaviour during Reset
E11025	Prg	List of axes displayed
E11026	Prg	Optional initialization of L20-L99 in %11000
E11027	Prg	Enable offsets on RTCP rotary axes
E11028	Prg	Record measured position in program frame as well
E11029	Prg	Execution of %11001
E11030	Prg	Inhibition of the default numerical 0
E11032	Prg	Computation of the angle crossing speed in G21
E11033	Prg	Enable segmentation in G21
E11034	Prg	Inhibits feed dependency on radius in G20
E12000	Prg	Enable block skip levels
E20100 to E20101	Prg	Status of the NCK probing inputs
E21000 to E21255	Prg	Check of software option bits
E31002	Prg	Part program zone protection
E31003	Prg	Protection of structures filled with symbolic variables
E32000	Prg	Minimum interpolation block execution time
E32001	Prg	Overspeed coefficient on the path in G12
E32002	Prg	Servo-control error tolerated on a circle
E32014	Prg	Radius on the trajectory in case of G2,G3,G23,G01, spline and polynomial interpolation
E32020	Prg	Segmentation value in G21
E32055	Prg	Difference of radius admitted by C.OFF
E32061	Prg	Tool radius already included in the trajectory
E41008	Prg	Active DAT1 set (G5n)
E41009	Prg	Active DAT2 set (G5nn)
E49001 to E49128	Prg	Dynamic Operation number

E49200 to E49328	Prg	Reserved : Dynamic Operation elapsed time
E6x006	Prg	RESERVED : WPC Linear and rotary offsets
E6x007	Prg	RESERVED : WPC Angular offsets applied on A / B / Linear and rotary offsets
E6x008	Prg	Monitoring a minimal dynamic travel
E6xnn0	Prg	DAT1 offsets table number nn applied on axis x
E6xnn1	Prg	DAT2 offset table number nn applied on axis x
E69000	Prg	Scaling factor
E69001	Prg	Inclined axis angle (grinders)
E69002	Prg	Inclined wheel angle (grinders)
E69003	Prg	Axis re-assignment
E69004	Prg	Tool disengagement in G41/G42 for an angle bigger that 120°
E79014	Prg	Segmentation value
E7x000	Prg	Position reference of an axis in the channel
E91500 to E91531	Prg	Early block change – Status of BlockChanged [0..31]
E91600 to E91631	Prg	Early block change – BlockChangeCtrl [0..31]
E99000 to E99031	Prg	Logical drive address to remote drive address coupling
Reserved parameters (NUM Specific)		
E parameter	**Function**	**Description**
E13000	Debug	RESERVED : Axes with reference monitoring
E13001	Debug	RESERVED : Axes monitoring enable
E13002	Debug	RESERVED : Spindles with reference monitoring
E13003	Debug	RESERVED : Spindles monitoring enable
E13004	Debug	RESERVED : Monitoring activation
E13005	Debug	RESERVED : Cyclic task monitoring
E13006	Debug	RESERVED : Maximum of cyclic task monitoring
E13007	Debug	RESERVED : Enable Dynamic operators time monitoring
E32050	Debug	RESERVED :Reason for not achieving look ahead
E32051	Debug	RESERVED :Reason for not reaching end of block velocity
E32052	Debug	RESERVED :Extended information about E32051
E32053	Debug	RESERVED :Reason for not reaching maximum velocity
E32054	Debug	RESERVED :Number of preprocessed blocks
E41003	Debug	RESERVED: Do not use anymore
E41007	Debug	RESERVED : Do not use anymore
E49200 to E49328	Debug	RESERVED : Dynamic Operation elapsed time
E93600 to E93631	Debug	Type of encoder (analog axes)
RTCP parameters		

E parameter	Function	Description
E32007	RTCP	Acceleration coefficient in RTCP
E32012	RTCP	Error tolerated on circles programming
E32015	RTCP	Maximum length for RTCP segments
E32056	RTCP	Length of segment (linear) to interpolate in RTCP, Polynomial
E32057	RTCP	Length of segment (angular) to interpolate in RTCP, Polynomial
E32058	RTCP	Curvature error tolerated in RTCP
E32059	RTCP	Maximum angle at pivot point for allowing small segments
E32060	RTCP	Max number of BFFs in an orientation singularity

Servo adjustments parameters		
E parameter	**Function**	**Description**
E32008	Servo	Time to reach a maximal jerk (derivative of the acceleration)

Spindles parameters		
E parameter	**Function**	**Description**
E79000	Spindles	Reference position of the current measured spindle
E79001	Spindles	Speed demand for the current main spindle
E79007	Spindles	Minimal value for the main spindle override
E79008	Spindles	Maximum value for the main spindle override
E90400 to E90431	Spindles	Spindle angular position
E90432 to E90463	Spindles	Spindle modulo
E90464 to E90495	Spindles	C axis / Spindle measurement conversion used for Spindle/ axis switching
E90500 to E90531	Spindles	Spindle speed setting
E90532 to E90563	Spindles	Spindle position set value
E90564 to E90595	Spindles	Spindle on speed set value
E90600 to E90631	Spindles	Maximum spindle Indexing speed
E90632 to E90663	Spindles	Spindle in position window
E90664 to E90695	Spindles	Spindle indexing gain
E90700 to E90731	Spindles	Spindle indexing acceleration
E90732 to E90763	Spindles	Spindle stopped threshold
E90764 to E90795	Spindles	Tolerance interval coefficient
E90800 to E90831	Spindles	Enable synchronization on master spindle
E90832 to E90863	Spindles	Spindle speed for maximum torque
E90864 to E90895	Spindles	Spindle speed for maximum power
E90900 to E90931	Spindles	Position gap for master slave spindle synchronization
E90932 to E90963	Spindles	Spindle Integration constant
E90964 to E90995	Spindles	Spindle jog speed
E91032 to E91063	Spindles	Servo or non-servo status of the spindles

E91132 to E91163	Spindles	Homing status on a spindle
E91164 to E91195	Spindles	Channel in which each spindle is located
E91432 to E91463	Spindles	Homing forced by user on a spindle
E91600	Spindles	Status of Early Block Change control
E93532 to E93563	Spindles	Spindle in-position window
E94132 to E94163	Spindles	Master spindle for spindle synchronization

Spindle synchronization parameters		
E parameter	**Function**	**Description**
E90900 to E90931	Synchro	Position gap for master slave spindle synchronization
E90932 to E90963	Synchro	Spindle Integration constant
E94132 to E94163	Synchro	Master spindle for spindle synchronization

Tools parameters		
E parameter	**Function**	**Description**
E50000	Tools	Current tool correction number
E50001 to E50255	Tools	Tool length or X dimension
E51000	Tools	Current tool axis orientation
E51001 to E51255	Tools	Tool end radius or Z dimension
E52001 to E52255	Tools	Tool radius
E53001 to E53255	Tools	Dynamic length or X incremental correction
E54001 to E54255	Tools	Dynamic radius or Z incremental correction
E55001 to E55255	Tools	Tool tip orientation
E56001 to E56255	Tools	Tool table H parameter (available)
E57001 to E57255	Tools	Tool type definition
E58000	Tools	Tool loaded in main spindle
E58001 to E58255	Tools	Tool geometry code
E59000	Tools	Overstock in roughing
E2000001 to E200xxyy	Tools	Tool length or X dimension for Turning Machines
E2010001 to E201xxyy	Tools	Cutter tip radius @ of correction xxx or Z dimension for Turning Machines
E2020001 to E202xxyy	Tools	Tool radius R of correction xxx
E2030001 to E203xxyy	Tools	Dynamic length correction DL of correction xxx
E2040001 to E204xxyy	Tools	Dynamic radius correction DR of correction xxx
E2050001 to E205xxyy	Tools	Tool tip orientation
E2060001 to E206xxyy	Tools	Available parameters (H of the table of dynamic corrections)

E2070001 to E207xxyy	Tools	Parameter defining the tool type of correction xxx
E2080001 to E208xxyy	Tools	Corrector Life Value
E2090001 to E209xxyy	Tools	Corrector Life Warning
E2100001 to E210xxyy	Tools	Corrector Life Error
E2110001 to E211xxyy	Tools	Corrector Life Type
E2120001 to E212xxyy	Tools	Corrector Max Speed
E2130001 to E213xxyy	Tools	Corrector Max Feed

Parameters returning machining information		
E parameter	**Function**	**Description**
E13008	Info	Part program execution time
E13009	Info	Part program machining time
E13010	Info	General NC operating time
E13011	Info	Last part program execution time
E13012	Info	Last part program machining time
E13013	Info	Total part program execution time
E13014	Info	Total part program machining time
E13015	Info	G04 Fxx programmed Time
E13016	Info	G04 Fxx Remaining Time
E32016	Info	Current line number in execution
E32017	Info	Current block number in execution
E32018	Info	NCK Software version
E32019	Info	FPGA Firmware version

General purpose parameters		
E parameter	**Function**	**Description**
E10000 to E10031	ALL	Boolean Read / Write parameters for exchange with the PLC application
E20000 to E20031	ALL	Boolean Read parameters for exchange with the PLC application
E30000 to E30255	ALL	DOUBLE Read / Write parameters for exchange with the PLC application
E33000 to E33009	ALL	SINT8 Addressing of the PLC Outputs
E33010 to E33999	ALL	SINT8 Addressing of the PLC Outputs
E37000 to E37127	ALL	UINT8 Read parameters to access the PLC memory
E40000 to E40255	ALL	DOUBLE Read parameters to access the PLC memory
E41000	ALL	Current CNC mode

E41001	ALL	Current channel number
E41002	ALL	Number of channels
E41004	ALL	System Serial number
E41005	ALL	Sampling period
E41006	ALL	Channel position servo-loop time constant
E41010	ALL	Interrupt channel triggered
E41102	ALL	Number of NC channels
E41103	ALL	Returns the NVRAM available in the NCK (in byte)
E41104	ALL	Status of Tool management (P7 N3 3)
E41105	ALL	Change CNC Operating mode
E46000 to 46002	ALL	UINT16 Read parameters to access the PLC memory
E46003 to 46999	ALL	UINT16 Read parameters to access the PLC memory
E47000 to E47127	ALL	UINT8 Read Read parameters to access the PLC memory
E80000 to E80199	ALL	Local DOUBLE R/W parameters
E81000 to E81999	ALL	Master axis position reference
E82000 to E82999	ALL	Slave axis corrections with respect to the master axes
E83000 to E83999	ALL	General purpose DOUBLE R/W parameters
E84000 to E84999	ALL	General purpose DOUBLE R/W parameters
E88000 to E88999	ALL	General purpose NUM specific parameters
E3000000 to E3000031	ALL	Delay or anticipation for inter-sample M functions
E3000032 to E3000063	ALL	Delay or anticipation for Timed M

MLEGB parameters (Multi Level Electronic Gear Box - EGB in short)		
E parameter	**Function**	**Description**
E7x200	EGB	EGB Synchro homing on first leader axis
E7x201	EGB	EGB Synchro homing on second leader axis
E7x202	EGB	EGB Synchro homing on third leader axis
E7x203	EGB	EGB Synchro homing on fourth leader axis
E7x204	EGB	EGB Synchro homing on fifth leader axis
E7x205	EGB	EGB Synchro homing on follower axis
E7x206	EGB	Current EGB tolerance
E7x207	EGB	Current EGB Synchro error
E7x208	EGB	Maximum value reached in Ex7x207
E7x209	EGB	Minimum value reached in Ex7x207
E7x210	EGB	Consider leader axis as rotary (1) or linear (0)
E7x211	EGB	MLEGB works in OP (0) or OM (1) referential
E7x212	EGB	Slave axis respects position, velocity and acceleration (1)
E7x213	EGB	Lower position limit of the axis

E7x214	EGB	Upper position limit of the axis
E7x215	EGB	Values of position limits in OP (0) or OM (1) referential

8.4 List of detailed E parameters sorted numerically



E10000 to E10031	Boolean Read / Write parameters for exchange with the PLC application							
	ALL	Cmn	None	0/not 0		-	NORST	
	RWpp	Rplc	RENA	RW (type 3)	RWDopC	UINT32	0	Osc (diff. value)
Description	Parameters used to report a Boolean information to the PLC program on *"Flexium_NCK.RCNC.E10000[x]"* variable. They can be read or written by part program and dynamic operators. There is no wait time for an acknowledgement from the PLC program. The function of these parameters is defined by the machine builder.							

E11000	Angular offset (function ED..) enabled							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	Parameter used to enable or disable the angular offset function (ED). .							

E11001	Programmed origin shift (G59) enabled							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	Parameter used to enable or disable the programmed origin shift (G59).							

E11003	Processing of mirror function (function G51) enabled							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	Parameter used to enable or disable the taking into account of programmed mirror function. For additional information, please refer to the G51 function in this manual.							

E11005	Programming by diameter enabled							
	Prg	Chnl	None	0/not 0		-	RSTYES	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	See P4	Osc (diff. value)
Description	This parameter is reset to force programming by radius on the X (or U) axis. Programming by diameter is possible only with G20 (programming in X, Y, Z polar coordinates), provided it is enabled by machine parameter P4. For additional information, please refer to the G20 function in this manual.							

E11006	Tool center programming (Turning Machine)							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	Setting this parameter is equivalent to a tool orientation C0.							

E11007	Spindle speed potentiometer enabled)							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	When this parameter is set to «0» for a channel, the spindle speed used for this channel is set to 100 percent.							

E11008	Drawing of a Complete Circle							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	With circular interpolation, this parameter is set to «0» to prevent a complete circle when the start point and end point are the same. This parameter is set to "1" after a reset, i.e. permitting full circles.							

E11009	Enabling full revolution on rotary axes							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	This parameter when set to 0 prevents the execution of a complete turn when the same position is programmed for a rotary axis. It is set to 0 after a reset therefore permitting full revolution in one block. Remark. In G0 if the same position is programmed, no movement will occur whatever the value in E11009. For additional information, please refer to the G0 and G01 functions in this manual.							

E11010	Shortest rotation direction selection							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	This parameter when set to 0 select the shortest path for a rotary axis (except if a sign is explicitly programmed). It is set to "1" after a reset therefore the direction of rotation is selected by the sign preceding the positioning value.							

E11011	Indexed names of axes enabled							
	Prg	Chnl	None	0/1		4.1.20	RST YES	
	RWpp	NOPlc	RENA	RDOp (type 2)	RWDopC	UINT16	P7 N4.2	Osc
Description	Possibility to program axes using an index. Examples X1, X2, X3, Y , Z1, Z2 Axes names among X,Y,Z, U,V, W, A, B, C, indexes 1..9.							

E11012	Zero following error operation							
	Axes	Chnl	1/256	0…512		-	NORST	
	RWpp	NOPlc	RENA	NODOP	RWDopC	DOUBLE	1	Osc (diff. value)
Description	For high precision contouring, this parameter enables cancellation of the following error. The parameter is set to enable cancellation and reset to inhibit it. A system reset does not affect the value of parameter E11012. For additional information, please refer to P55 and P19 machine parameter in Flexium+ Parameters manual M00041.							

E11013	Type of acceleration							
	HSC	Chnl	None	-1, 0, 1		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	SINT8	P53N0	Osc (diff. value)
Description	This parameter is set to define the type of acceleration. At power on, this parameter is set as declared in machine parameter P53 N0, N3, ..., N21. Among the E110xx parameter, this is the only one with three possible values: • 0 : constant acceleration • 1 : Triangular acceleration • -1 : Trapezoidal or parabolic acceleration. Please refer to machine parameter P53 on Flexium+ Parameters manual M00041.							

E11014	Deceleration over several blocks							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	The deceleration function is enabled when this parameter is set to «1» and disabled when it is set to «0». A reset sets this parameter to «1».							

E11015	Angle crossing management enabled							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT16	1	Osc (diff. value)
Description	When this parameter is set to «0», the angles are not analysed when calculating the end of block speed.							

E11016	Selects front or rear turret (Turning Machine)							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RDopC	SINT8	P5N1	Osc (diff. value)
Description	Parameter used on lathe to check if the turret is a front turret (1) or a rear turret (0).							

E11017	Inclined plane function enabled (G24+)							
	Prg	Chnl	None	0/1		-	NORST	
	Rpp	NOPlc	RENA	R (type 2)	RDopC	UINT16	0	Osc (diff. value)
Description	This parameter set to indicate that the inclined plane function (G24+) is enabled. For additional information, please refer to the G24+ function in this manual.							

E11018	RTCP function enabled (G26+)							
	Prg	Chnl	None	0/1		-	NORST	
	R	NOPlc	RENA	R (type 2)	RDopC	UINT16	0	Osc (diff. value)
Description	This parameter set to indicate that the RTCP function (G26+) is enabled. For additional information, please refer to the G26+ function in this manual.							

E11019	Storing axes position on hardware interrupt							
	Prg	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT8	0	Osc (diff. value)
Description	When E11019 is set the system will record the axes position on each interrupt like in G10 except that the interruption will not terminate the current block.							

E11020	Allows writing the reference position for a "Virtual" axis							
	Prg	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT8	0	Osc (diff. value)
Description	This parameter is used for allowing to writing the position on a *'Virtual* axis.							

E11021	Activation of the speed averaging filter outside HSC							
	HSC	Chnl	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type 2)	RWDopC	UINT8	0	Osc (diff. value)
Description	This parameter is used to activate a speed averaging system on systems. It excludes using HSC simultaneously. If this speed averaging filter is used, then HSC is not available.							

E11022	Emergency retract cause							
	Prg	Cmn	None	0/1/2/3		-	RSTYES	
	Rpp	NOPlc	RENA	R (type 2)	RWDopC	SINT8	0	Osc (diff. value)
Description	This parameter is set when an emergency retract is automatically triggered. • Set to 1 by loss of communication between NC and PLC (see P99 N3 on Flexium+ Parameters manual M00041). • Set to 2 by Emergency activated by the SAMX board.							

E11023	EGB leader axis in emergency							
	EGB	Cmn	None	0...31-1		-	RSTYES	
	Rpp	NOPlc	RENA	NODop	RWDopC	SINT32	1	Osc
Description	These parameters return the physical address of an axis in synchronization fault. This parameter is related to the option Electronic Gearbox and is only significant when this option is active.							

E11024	Axes assignment after reset							
	Axes	Cmn	None	0..3		4.1.10	NORST	
	RWpp	NOPLC	RENA	NODOp	RWDOpC	Byte	No	Osc
Description	Axes and spindles behaviour at RAZ: 0 = axes and spindles return to the project configured channel. If bit 1 is set (E11024=1) the axes keep staying in the channel when RAZ is given. If bit 2 is set (E11024=2) the spindles keep staying in the channel when RAZ is given.							

E11025	List of axes displayed							
	Prg	Cmn	None	Any value		4.1.20	NO	
	RWpp	NOPLC	NOENA	NODOp	RDOpC	UINT 32	P0	Osc
Description	Possibility to hide or show the display of particular axes.							

E11026	Optional initialization of L20 – L99 by %11000							
	Prg	Chnl	None	0 / 1		4.1.20	NO	
	RWpp	NOPLC	RENA	NDOp	RDOpC	UINT 16	0	Osc
Description	When set this parameter permits to initialize L20 to L99 in the LAST LINE of %11000.							

E11027	Enable offsets on RTCP rotary axes							
	Prg	Chnl	None	0 / 1		4.1.20	YES	
	RWpp	NOPLC	RENA	RDOp (2)	RDOpC	UINT 8	0	Osc
Description	The E11027 parameter is created in order to ignore or to allow explicitly the use of E6x012 (in the new macro %10151) and the new fourth numerical value of TA, TB, TC, PA, PB, PC in G26 syntax. If E11027=0, the 3 angular offsets E6x012 are ignored (in the new macro %10151) and the new numerical values of G26 **are optional** (the values are ignored and considered as null even if programmed).							

E11028	On the fly recording of position in part referential							
	Prg	Chnl	None	0/1		4.1.30	RST	
	RWPP	NOPLC	NOENA	NOOp	RWDOpC	Byte	0	Osc
Description								

E11029	Execution of %11001							
	Prg	Cmn	None	0/1		4.1.30	RST	
	RWPP	NOPLC	NOENA	NOOp	RWDOpC	Byte	0	Osc
Description	This parameter allows execution of %11001 (execution at the end of reset). As it is cleared by a reset E11029 = 1 should be written in %11000 if such a program is to be executed							

E11030	Inhibition of the default 0 after a program address							
	Prg	Cmn	None	0/1 4.2.0			NORST	
	RWPP	NOPLC	NOENA	NO DOp	RWDOpC	Byte	1	Osc
Description	This parameter permits or inhibit (when 1) the possibility of no numerical value after an address (ex : GX accepted or not as G0 X0)							

E11032	Computation of the angle crossing speed in G21							
	Prg	Chnl	None	0/1		4.1.30	NORST	
	RWPP	NOPLC	NOENA	RWDOp	RWDOpC	UINT32	1	Osc
Description	Please refer to programming in cartesian coordinates in this manual.							

E11033	Enable segmentation in G21							
	Prg	Chnl	None	0/1		4.1.30	NORST	
	RWPP	NOPLC	NOENA	RWDOp	RWDOpC	UINT32	1	Osc
Description	Please refer to programming in cartesian coordinates in this manual.							

E11034	Inhibits feed dependency on radius in G20							
	Prg	Chnl	None	0/1		4.1.30	NORST	
	RWPP	NOPLC	NOENA	RWDOp	RWDOpC	UINT32	1	Osc
Description	Please refer to programming in polar coordinates in this manual.							

E11035 to E11063	Reserved							
	--	--	--	0/1		4.1.30	--	
	--	--	--	--	RWDOpC	UINT32	0	Osc
Description	.....							

E12000	Enable block skip levels							
	Prg	YES	None	2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODOP	RWDopC	UINT32	1	Osc
Description	This parameter is used to activate up to 32 block skip levels The value can be changed in Reset status and at any moment if P7 N3.1 = 1. For additional information please refer to Flexium+ HMI operator's manual M00039.							

E13000	RESERVED: Axes with reference monitoring							
	Debug	Cmn	None	2^32-1		-	NORST	
	RWpp	NOPlc	RENA	NODOP	RWDopC	UINT32	0	Osc
Description	This parameter is reserved for NUM Service only!.							

E13001	RESERVED: Axes monitoring enable							
	Debug	Cmn	None	0/not 0/2		-	RSTYES	
	RWpp	NOPlc	RENA	NODOP	RWDopC	SINT8	0	Osc (diff. value)
Description	This parameter is reserved for NUM Service only!.							

E13002	RESERVED: Spindles with reference monitoring							
	Debug	Cmn	None	+/-2^31-1		-	NORST	
	RWpp	NOPlc	RENA	NODOP	RWDopC	SINT32	0	Osc
Description	This parameter is reserved for NUM Service only!							

E13003	RESERVED: Spindles monitoring enable							
	Debug	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	NODOP	RWDopC	SINT8	0	Osc
Description	This parameter is reserved for NUM Service only!.							

E13004	RESERVED: Monitoring activation							
	Debug	Cmn	None	0/1/2/3 (other values: same as 0)		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDopC	SINT32	0	Osc
Description	This parameter is reserved for NUM Service only!							

E13005	RESERVED: Cyclic task monitoring							
	Debug	Cmn	uSec	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDopC	SINT32	0	Osc
Description	This parameter is reserved for NUM Service only!.							

E13006	RESERVED: Maximum of cyclic task monitoring							
	Debug	Cmn	uSec	Any value		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDopC	SINT32	0	Osc
Description	This parameter is reserved for NUM Service only!.							

E13007	RESERVED: Enable Dynamic operators time monitoring							
	Debug	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDopC	SINT8	0	Osc (diff. value)
Description	This parameter is reserved for NUM Service only!.							

E13008	Part program execution time							
	Info	Chnl	ms	2^32-1		-	NO	
	Rpp	NO	RENA	NOPlc	RWDopC	UINT32	NO	Osc
Description	This parameter returns in milliseconds the time the part program has been running The count starts at the first start command and stops on M2 or Reset.							

E13009	Part program machining time							
	Info	Chnl	ms	0…2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDopC	UINT32	E13012	Osc
Description	This parameters returns the machining time in milliseconds Machining time is counted when the spindle is running and axis moving at interpolation feed rate (not G00).							

E13010	General NC operating time							
	Info	Chnl	ms	2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDopC	UINT32	Last value	Osc
Description	This parameter returns the total time under which has been powered on in seconds.							

E13011	Last part program execution time							
	Info	Chnl	ms	2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDopC	UINT32	Last value	Osc
Description	This parameter returns the value of the last part program execution. It presents the same characteristics as E13008.							

E13012	Last part program machining time							
	Info	Chnl	ms	2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDopC	UINT32	Last value	Osc
Description	This parameter returns the value of the last part machining time. It presents the same characteristics as E13009.							

E13013	Total part program execution time							
	Info	Chnl	sec	2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDopC	UINT32	Last value	Osc
Description	This parameter returns the value of the last part machining time. It presents the same characteristics as E13009.							

E13014	Total part program machining time							
	Info	Chnl	sec	2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDopC	UINT32	Last value	Osc
Description	This parameter returns the TOTAL time of machining. It presents the same characteristics as E13009.							

E13015	G04 Fxx programmed Time							
	Info	Chnl	1/100 sec	2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDopC	UINT32	0	Osc
Description	This parameter returns the programmed value of a dwell (G04 F...) in hundreds of seconds.							

E13016	G04 Fxx Remaining Time							
	Prg	Chnl	1/100 sec	2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDopC	UINT32	0	Osc
Description	This parameter returns the remaining value of a dwell (G04 F...) in 1/100 of seconds.							

E20000 to E20031	Boolean Read parameters for exchange with the PLC application							
	ALL	NO	None	0/1			NORST	
	Rpp	W	RENA	R (type 3)	RWDopC	UINT32	0	Osc (diff. value)
Description	Parameters used to retrieve a Boolean information from the PLC program on *"Flexium_NCK.WCNC.E20000[x]"* variable. They can be read by part program and dynamic operators. The function of these parameters is defined by the machine builder. Please also refer to Flexium+ Extended Programming manual M00055 "Dynamic operators"..							

E20100 to E20101	Status of the NCK probing inputs							
	Prg	NO	None	0/1		-	NORST	
	Rpp	NO	RENA	NO	RWDopC	SINT32	Value	Osc
Description	These two parameters give the status of each of the probing signals located on the X9 NCK connector. E20100 corresponds to PB0 on pin 13 and E20101 to PB1 on pin 15. The parameters are set to 1 if the corresponding input signal is at 24Vdc, they are equal to 0 otherwise.							

E21000 to E21255	Check of software option bits							
	Prg	Cmn	None	0/1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDopC	UINT16	Value	Osc (diff. value)
Description	Parameters addressing binary values. These parameters are used to test the presence of s/w options 0 to 255. The parameter is set when the option is present.							

E30000 to E30255	DOUBLE R/W parameters for exchange with the PLC application							
	ALL	Cmn	None	0..2^32-1			NORST	
	RWpp	Rplc	RENA	NODop	RWDopC	DOUBLE	0	-
Description	Parameters used to report numerical values to the PLC program on *"Flexium_NCK. RCNC.E30000[x]"* variable. They can be read or written by part program. There is no wait time for an acknowledgement of the PLC program. The function of these parameters is defined by the machine builder.							

E31002	Part program zone protection							
	Prg	Cmn	None	0/1		-	RSTYES	
	RWpp	NOPlc	RENA	NODop	RWDopC	SINT8	0	Osc
Description	When set this parameter prevents any change in the part program area. This feature is useful when binary files are accessed by dynamic operators through their physical addresses. A relocation of these files is therefore forbidden leaving the addresses valid.							

E31003	Access to memory segments							
	Prg	Cmn	None	0/1		-	RSTYES	
	RWpp	NOPlc	RENA	NODop	RWDopC	SINT8	0	Osc
Description	In order to save Symbolic variables via Dynamic operators 32 segments of memory are defined by P59 (see G76+ S… in the Flexium+ Extended Programming manual M00055). This parameter is intended to prevent write access into the segment area when set to 1.							

E32000	Minimum interpolation block execution time							
	Prg	Cmn	None	0/1		-	RSTYES	
	RWpp	NOPlc	RENA	NODop	RWDopC	SINT8	P51 N0	Osc (diff. value)
Description	Parameter defining the minimum execution time for a block containing a linear or circular interpolation, expressed in milliseconds (ms). This parameter can be modified by programming. It only affects the channel in which it is programmed (it is initialized with the value of P51 by a reset).							

E32001	Overspeed coefficient on the path in G12							
	Prg	Cmn	1/1024	0.. 65535		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDopC	UINT16	P13*N8/N9 * 1024	Osc
Description	Parameter defining the handwheel overspeed coefficient expressed in 1/1024 (with G12). The variations of the increments on the first handwheel multiplied by this coefficient generate an overspeed applied to the programmed path. The new value written is saved until the NC is reset. For additional information, please refer to the G12 function in this manual.							

E32002	Servo-control error tolerated on a circle							
	Prg	Chnl	P4N2	any value < 32767		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDopC	DOUBLE	P52N0	Osc
Description	Parameter defining the servo-control error tolerated on a circle, expressed in NC internal measurement unit (P4N2). It is the image of machine parameter P52.							

E32003	Minimal angle for speed analysis							
	HSC	Chnl	Degrees	0…360		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDopC	DOUBLE	0	Osc (diff. value)
Description	Parameter defining the angle above which angle crossing speed is always analysed. This parameter is set to «0» at power on meaning the analyse is always carried out. Its value in degrees must be declared by programming to define a threshold for this analyse. The last programmed value is saved after a reset.							

E32004	Chordal error							
	HSC	Chnl	P4N2	any value > 0		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	10	Osc
Description	Parameter defining the chordal error in Internal linear units. At power on, this value is initialized with 10 and can be modified by programming. The last value programmed is saved after a reset.							

E32005	Number of total speed anticipation filter terms							
	HSC	Chnl	None	0...13		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	UINT16	P55N8…N15	Osc
Description	Parameter defining the number of terms of the total speed anticipation filter. This parameter is assigned to all the axes of the channel in which it is declared. At power up the parameter is initialized with the values of P55 N8 to N15 (according to the channel). Writing of this parameter is enabled if: • The *«high precision contouring»* option is present; otherwise the system returns error message 4, • Anticipation is disabled (E11012 = 0), or the system returns error message 95, • It is below 14, or the system returns error message 94. The last value programmed is saved after a reset.							

E32006	Time to reach maximum acceleration							
	HSC	YES	ms	+/-65535*P50*10-3 (used its absolute value only)		-	NORST-	
	RWpp	NO	RENA	NO	RWDOpC	UINT16	P53N1	Osc (diff. value)
Description	This parameter defines the time to get the maximal acceleration. It is used when acceleration is not constant (triangular or trapezoidal). The default value is defined by machine parameter P53 N1, N4, ..., N22. • When E11013 = 1, this parameter define the minimal time to reach the maximal velocity. • When E11013 = -1, this parameter define the minimal time to reach the maximal acceleration.. With trapezoidal acceleration, the setting of E32006 is limited to 200 sampling periods							

E32007	Acceleration coefficient in RTCP							
	Prg	Chnl	1/128	0...128 (0 = ineffective)		-	RSTYES	
	RWpp	NO	RENA	-	RWDOpC	UINT16	0	Osc
Description	This parameter is used to adjust the acceleration in RTCP when rotary axes are involved. Its purpose is to forbid over acceleration due to linear axes moves generated by the rotary axes displacements. The value defined in 1/128 is cumulated with the EG function.							

E32008	Time to reach maximal jerk (derivative of the acceleration)							
	HSC	Chnl	ms	+/-65535 (used its absolute value only)		-	NORST	
	RWpp	NOPIc	RENA	NODop	-	UINT16	P53N2	Osc (diff. value)
Description	When non zero, it allows for trapezoidal jerk (parabolic acceleration). • This parameter is significant only when E11013 = -1. • With E32008 = 0 The jerk is constant and acceleration trapezoidal. • With E32008 > 0 The jerk is trapezoidal and acceleration parabolic. the value of E32008 is limited to 200 samplings and should not exceed half of E32006 For additional information, please refer to machine parameter P53..							

E32009	Feed rate limitation in case of curve radius variation							
	HSC	Cmn	None	+/-65535 (used its absolute value only)		-	NORST	
	RWpp	NOPIc	RENA	NODop	RWDOpC	UINT16	0	Osc
Description	This parameter is used in HSC to define the error tolerated in case of curvature variation. The value is defined as 1500 * square root of the tolerated error.							

E32010	Coefficient (in 1/256) of the maximum radial acceleration							
	HSC	Cmn	1/256	0...65535		-	NORST	
	RWpp	NOPIc	RENA	NODop	RWDOpC	UINT16	0	Osc
Description	Range from 0 to 65535 with zero equivalent to 256.							

E32011	Number of terms of the speed averaging filter							
	HSC	Chnl	None	0.66 (if greater than 66, 66 is written; if < 0, error)		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	UINT16	NO	Osc
Description	To be used outside HSC. order and number of terms of the speed averaging filter.							

E32012	Error tolerated on circles programming							
	RTCP	Cmn	P4N2	20...255		-	RSTYES	
	RWpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	P52 N2	Osc
Description	This parameter defines the biggest difference allowed between the start radius and the end radius in circular interpolation (a higher value will trigger error 107). This parameter represents the machine parameter P52N2 and is reinitialized with its value at power up and after a reset (base 20 increments).							

E32014	Radius on the trajectory in case of G2,G3,G23,G01, spline and polynomial interpolation							
	Prg	Chnl	P4N2	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	R (type 2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter return the current trajectory radius whatever the type of interpolation (including small segments of G1, spline or polynomial interpolation).							

E32015	Maximum length for RTCP segments							
	RTCP	Chnl	Degrees	1...20		4.1.00.0	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	0	Osc (diff. value)
Description	This parameter permits to adjust the RTCP function by defining the maximum value for segmentation.							

E32016	Current line number in execution							
	Info	Chnl	None	0.. 2147483648		4.1.10.0	-	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT	No	Osc
Description	Line number currently in execution.							

E32017	Current block number in execution							
	Info	Chnl	None	0.. 2147483648		4.1.10	0	
	Rpp	NOPLC	RENA	NODOp	RDOpC	UINT	No	Osc
Description	Bloc number currently in execution.							

E32018	NCK Software version							
	Info	Cmn	None	0.. xx (UINT)		4.1.10	NORST	
	Rpp	NOPLC	RENA	NODOp	RDOpC	UINT	Bckp	Osc
Description	The value is returned in HEX.							

Version	Hex	Decimal
04.01.10.00	04010A00	67176960
04.01.20.00	04011400	67179520
04.01.30.00	04011E00	67182080

E32019	FPGA Firmware version							
	Info	Cmn	None	0.. x (UINT)		4.1.10	NORST	
	Rpp	NOPLC	RENA	NODOp	RDOpC	UINT	Bckp	Osc
Description	The value is returned in HEX.							

E32020	Maximum length in XY for G21 segmentation							
	Prg	Chnl	None	Any value		4.2.0	NO	
	RWPP	NOPLC	RENA	NODOp	RDOPC	UINT32	NOBckp	Osc
Description	Segmentation is inhibited if this parameter is set to 0							

E32050	Debug parameter : reserved							
	Debug	Chnl	None	0..255		4.1.20	NO	
	RWpp	NOPLC	NOENA	NDOp	RDOpC	UINT 8	NoBckp	Osc
Description	This reserved parameter indicates that the look ahead of the current block has not been executed and what is the reason. This parameter should be considered on NUM notice only.							

E32051	Debug parameter : reserved							
	Debug	Chnl	None	0..255		4.1.20	NO	
	RWpp	NOPLC	NOENA	NDOp	RDOpC	UINT 8	NoBckp	Osc
Description	This reserved parameter indicates that the look ahead of the current block has not been executed and what is the reason. This parameter should be considered on NUM notice only.							

E32052	Debug parameter : reserved							
	Debug	Chnl	None	0..255		4.1.20	NO	
	RWpp	NOPLC	NOENA	NDOp	RDOpC	UINT 8	NoBckp	Osc
Description	This reserved parameter extends information of E32051.							

E32053	Debug parameter : reserved							
	Debug	Chnl	None	0..255		4.1.20	NO	
	RWpp	NOPLC	NOENA	NDOp	RDOpC	UINT 8	NoBckp	Osc
Description	This reserved parameter indicates why the maximum velocity has been limited This parameter should be considered on NUM notice only.							

E32054	Debug parameter : reserved							
	Debug	Chnl	None	0..255		4.1.20	NO	
	RWpp	NOPLC	NOENA	NDOp	RDOpC	UINT 8	NoBckp	Osc
Description	This reserved parameter indicates why the maximum velocity has been limited This parameter should be considered on NUM notice only.							

E32055	Difference of radius admitted with C.Off							
	Prg	Chnl	None	Any value		4.2.0	NO	
	RWPP	NOPLC	NOENA	NODOp	RDOPC	UINT32	NOBckp	Osc
Description	This parameter defines the maximum difference between the start radius and the end radius with the function C.OFF and variable offsets							

E32056	Length of segment to interpolate in RTCP, Polynomial							
	RTCP	Chnl	Mm	0.1 to 20 mm		4.1.30	RST	
	RWPP	NOPLC	NOENA	NODOp	RWDOpC	Double	0	Osc
Description	When 0 this parameter permits segmentation by time, when not 0 it defines the linear distance of segmentation.							

E32057	Angular step to segment in RTCP							
	RTCP	Chnl	Degrees	0.1 to 20°		4.1.30	NORST	
	RWPP	NOPLC	NOENA	NODOp	RWDOpC	Double	0	Osc
Description	When 0 this parameter permits segmentation by time, when not 0 it defines the angular distance of segmentation.							

E32058	Curvature error tolerated in RTCP							
	RTCP	Chnl	P4N2	0 to 65535		4.1.30	NORST	
	RWPP	NOPLC	NOENA	NODOp	RWDOpC	Douple	65535	Osc
Description	Maximal error on the pivot point in RTCP – Similar to E32002 which is on the trajectory (tool tip).							

E32059	Maximum angle at pivot point for allowing small segments							
	RTCP	Chnl	Degrees	0 to 360°		4.1.30	NORST	
	RWPP	NOPLC	NOENA	NODOp	RWDOpC	Double	0	Osc
Description	Equivalent to E32003 which is on the trajectory (tool tip).							

E32060	Max number of BFFs in an orientation singularity							
	RTCP	Chnl	None	0 to 10		4.1.30	RST	
	RWPP	NOPLC	NOENA	NODOp	RWDOpC	UINT8	0	Osc
Description	In G30 or G34, number of blocs to anticipate and balance the delta position of the rotary axes to exit a singularity in RTCP.							

E32061	Tool radius offset already included in the trajectory							
	Prg	Chnl	None	Any value		4.2.0	NO	
	RWPP	NOPLC	NOENA	NODOp	RDOPC	UINT32	NOBckp	Osc
Description	This parameter is useful for G92 L in order to calculate the tangential feed rate. It defiines the tool offset already included in the trajectory whatever the mean to define this tool offset (C.OFF function or other). The main use is to prevent collision by allowing the standard tool correction to use a limited value (generally used on grinders).							

E33000 to E33009	Indirect addressing of the PLC Outputs							
	ALL	Cmn	None	0..255		-	NORST	
	RWpp	Rplc	RENA	W (type 3)	RWDOpC	SINT8	0	-
Description	Parameters used to report to the PLC Output image via dedicated "*Flexium_NCK.RCNC.E33000[x]*" variables (999 unsigned integer [8 bits]). These parameters are sent without waiting for a report by the PLC. They can be written by part program. If a value out from the range 0 ... +255 is entered to the E parameter, the NC doesn't report any error and manages its less significant byte							

E33010 to E33999	Indirect addressing of the PLC Outputs							
	ALL	NO	None	0…255			NO	
	RWpp	Rplc	RENA	NO	RWDOpC	SINT8	NO	-
Description	Parameters used to report to the PLC Output image via dedicated "*Flexium_NCK.RCNC.E33000[x]*" variables (999 unsigned integer [8 bits]). These parameters are sent without waiting for a report by the PLC. They can be written by part program. If a value out from the range 0 ... +255 is entered to the E parameter, the NC doesn't report any error and manages its less significant byte They are equivalent to the parameters E33000 to E33009 except that they cannot be written via dynamic operator.							

E34000 to E34001	SINT8 R/W parameters access the NCK embedded digital outputs							
	I/O	Cmn	None	0…255		-	NORST	
	RWpp	NOPlc	RENA	W (type 3)	RWDOpC	SINT8	0	Osc
Description	Parameters used to access the NCK embedded digital outputs: • E34000 = *out7..out0* connector X10. • E34001 = *out15..out8* connector X10. They can be read or written by part program and only written via dynamic operators. Reminder. The NCK embedded digital outputs can only be managed via Parameters E3400x if the PLC permits by the variable *Flexium_NCK.WCNC.IO.LockDigitalOutput[x]*:= FALSE.							

E34010 to E34011	SINT8 Read parameters access the NCK embedded digital outputs value							
	I/O	Cmn	None	0…255		-	NORST	
	Rpp	NOPlc	RENA	R (type 3)	RWDOpC	SINT8	0	Osc
Description	Parameters used to read the NCK embedded digital outputs value: • E34000 = *out7..out0* connector X10. • E34001 = *out15..out8* connector X10. They can be read by part program and by via dynamic operators. This is particularly useful when the digital outputs are written by the PLC.							

E34100 to E34101	UINT16 R/W parameters access the NCK embedded analogues outputs							
	I/O	Cmn	None	0...65535		-	NORST	
	RWpp	NOPlc	RENA	W (type 3)	RWDOpC	UINT16	0	Osc
Description	Parameters used to access the NCK embedded analogue outputs: • E34100 = Analogue out0 connector X9. • E34101 = Analogue out1 connector X9. They can be read or written by part program and only written via dynamic operators. The relationship between value and output voltage is: (Absolute value on bits 0..14, sign on bit 15) • 0 -> 0V / 32767 -> +10V / 32768 -> 0V / 65535 -> -10V Reminder: The NCK analogue outputs can only be managed via E3410x parameters if the PLC permits via the variable *Flexium_ NCK.WCNC.IO.LockAnalogOutput[x]*:= FALSE.							

E34110 to E34111	UINT16 Read parameters access the NCK embedded analog outputs value							
	I/O	Cmn	None	0...65535	-	-	NORST	
	Rpp	NOPlc	RENA	R (type 3)	RWDOpC	UINT16	0	Osc
Description	Parameters used to read the NCK embedded analog outputs value: • E34100 = Analogue out0 connector X9. • E34101 = Analogue out1 connector X9. They can be read by part program and by via dynamic operators. This is particularly useful when the analog outputs are written by the PLC.							

E34200 to E34201	Digital Output Locked by PLC							
	I/O	Cmn	None	0/1		-	NORST	
	Rpp	RWplc	RENA	NODop	RWDOpC	UINT8	0	Osc (diff. value)
Description	These two parameters reflect the status of *"Flexium_NCK.WCNC.IO.LockDigitalOutput[x]"* PLC variables. If they are set then the Digital Output cannot be managed by the NCK but only by the PLC.							

E34300 to E34301	Analog Output Locked by PLC							
	I/O	Cmn	None	0/1		-	NORST	
	Rpp	RWplc	RENA	NODop	RWDOpC	UINT8	0	Osc (diff. value)
Description	These two parameters reflect the status of *"Flexium_NCK.WCNC.IO.LockAnalogOutput[x]"* PLC variables. If they are set then the Digital Output cannot be managed by the NCK but only by the PLC.							

E35100 to E35131	First SINT32 test Drive point value							
	Drive	NO	None	216 -1	-	NO	NORST	
	Rpp	NO	RENA	R (type 2)	RWDOpC	SINT32	Value	Osc
Description	These parameters define the first 32 bits test point value for drive 00 (E35100) to 31 (E35131) selected by V130 in this drive. For additional information, please refer to NUMDrive X Parameters manual M00031 "V130 Associate MP with TP1".							

E35132 to E35163	Second SINT32 Drive test point value							
	Drive	NO	None	216 -1	-	NO	NORST	
	Rpp	NO	RENA	R (type 2)	RWDOpC	SINT32	Value	Osc
Description	These parameters define the second 32 bits test point value for drive 00 (E35132) to 31 (E35164) selected by V131. For additional information, please refer to NUMDrive X Parameters manual M00031 "V131 Associate MP with TP2".							

E35200 to E35231	Code number of the first test point							
	Drive	NO	None	+/-2^31-1		-	NORST	
	RWpp	NO	RENA	NO	RWDOpC	SINT32	V130	Osc (diff. value)
Description	These parameters define the code number of the first test point to read in drive 00 to 31 (equivalent to drive parameter V130.) For additional information, please refer to NUMDrive X Parameters manual M00031 "V130 Associate MP with TP1".							

E35232 to E35263	Code number of the second test point							
	Drive	NO	None	+/-2^31-1		-	NORST	
	RWpp	NO	RENA	NO	RWDOpC	SINT32	V131	Osc (diff. value)
Description	These parameters define code number of the second tests point to read in drive 00 (E35232) to 31 (equivalent to drive parameter V131). For additional information, please refer to NUMDrive X Parameters manual M00031.							

E35264 to E35295	Enable the reading test point							
	Drive	Cmn	None	0/1		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	SINT32	0	Osc (diff. value)
Description	These parameters when set to 1 enable the reading the test point in drive 00 (E35264) to 31 (E35295). For additional information, please refer to NUMDrive X Parameters manual M00031.							

E35300 to E35363	Select a set of Drive parameters							
	Axes	Cmn	None	0/1/10/11		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	SINT32	0	-
Description	These parameters are used to select two sets set of drive parameters (Position Loop and Speed Loop) in the form E353xx = yz • xx : Drive number 0 to 63 (for spindles = drive address +32). • y : Position loop parameter set (0/1). • z : Speed loop parameter set (0/1). For additional information, please refer to NUMDrive X Parameters manual M00031.							

E36000 to E36002	UINT16 R/W parameters to access the PLC memory							
	I/O	Cmn	None	0…65535		-	RSTYES	
	RWpp	Rplc	RENA	W (type 3)	RWDOpC	UINT16	0	-
Description	Parameters used to send numerical values to the PLC program on *"Flexium_NCK.RCNC".E36000[x]"* variable. These parameters are sent without waiting for a report by the PLC. They can be read and written by part program. The function of these parameters is defined by the machine builder. If a value out from the range 0 ... +65535 is entered to the E parameter, the NC doesn't report any error and manages its less significant word. The first 3 parameters (up to E36002) can be written via dynamic operator. For additional information, please refer to Flexium Extended Programming manual M00020 "Dynamic operators".							

E36003 to E36999	UINT16 R/W parameters to access the PLC memory							
	I/O	Cmn	None	0…65535		-	RSTYES	
	RWpp	Rplc	RENA	NODop	RWDOpC	UINT16	0	-
Description	Parameters used to send numerical values to the PLC program on *"Flexium_NCK.RCNC".E36000[x]"* variable. These parameters are sent without waiting for a report by the PLC. They can be read and written by part program. The function of these parameters is defined by the machine builder. If a value out from the range 0 ... +65535 is entered to the E parameter, the NC doesn't report any error and manages its less significant word. They are equivalent to parameters E36000 to E36002 except that they cannot be written via dynamic operator. For additional information, please refer to Flexium Extended Programming manual M00020 "Dynamic operators". They are equivalent to parameters E36000 to E36002 except that they cannot be written by Dynamic Operators. They are accessible by Dynamic operators in C but will not be written faster than Part program.							

E37000 to E37127	SINT8 Read parameters to access the PLC memory							
	ALL	Cmn	None	0..255		-	NORST	
	Rpp	Rplc	RENA	W (type 3)	RWDOpC	SINT8	0	Osc
Description	Parameters used to report numerical values to the PLC program on *"Flexium_NCK.RCNC.E37000[x]"* variable. The parameters are sent without waiting for a report by the PLC. They can be written via dynamic operator only. The function of these parameters is defined by the machine builder. For additional information, please refer to Flexium+ Extended Programming manual M00055 "Dynamic operators".							

E40000 to E40255	DOUBLE Read parameters to access the PLC memory							
	ALL	Cmn	None	2^31 -1			NORST	
	Rpp	Wplc	RENA	NODop	RWDOpC	DOUBLE	PLC value	-
Description	Parameters used to retrieve numerical values from the PLC program on *"Flexium_NCK.RCNC.E40000[x]"* variable. They can be read by part program. The function of these parameters is defined by the machine builder.							

E41000	Current CNC mode							
	ALL	Chnl	None	0..15		-	NORST	
	Rpp	NO	RENA	NO	RWDOpC	SINT8	Mode	Osc (diff. value)
Description	This parameter holds a number corresponding to the current mode: • 0: continuous mode • 1: single mode • 2: MDI mode • 3: rapid mode • 4: sequence number search mode • 6: test mode • 7: manual mode • 8: home mode.							

E41001	Current channel number							
	ALL	Chnl	None	0..7		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT16	Chnl	Osc
Description	This parameter read in a program returns the channel number to which the program is assigned. (channel 1: E41001=0, channel 2: E41001=1, etc.).							

E41002	Number of channels							
	ALL	Cmn	None	0..8		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT16	P97	Osc
Description	This parameter returns the number of channels available in the system.							

E41004	System Serial number							
	ALL	Cmn	None	+/-2^31-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	SINT32	Serial	Osc (diff. value)
Description	This parameter returns the serial number of the CNC.							

E41005	Sampling period							
	ALL	Cmn	µs	1000…10000		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	SINT32	P50	Osc (diff. value)
Description	This parameter returns the value of the sampling time.							

E41006	Channel position servo-loop time constant							
	ALL	Chnl	ms	0…2^32-1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	SINT32	P56 Nx	-
Description	This parameter defines the time constant in milliseconds (ms). It is the image of machine parameter P56. Please refer to machine parameter P56 on Flexium+ Parameters manual M00041.							

E41008	Active DAT1 set (G5n)							
	Prg	Chnl	None	1…5		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT16	1	Osc
Description	This parameter returns the number of the current DAT1 set (1 : G54 .. 5 : G58).							

E41009	Active DAT2 set (G5n)							
	Prg	Chnl	None	1…99		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT16	1	Osc
Description	This parameter returns the number of the current DAT2 set (1 : G501 .. 99 : G599).							

E41010	Interrupt channel triggered							
	ALL	Chnl	None	0,1,2,3/10,20,30,40		-	RSTYES	
	Rpp	NOPlc	RENA	R (type 2)	RWDOpC	UINT16	0	Osc

Description	This flag indicates on which interrupt channel an interrupt has occurred.
	= 0 NCK reset.
	If only Rising or Falling edge are enabled the values can be:
	= 1; // PB0
	= 2; // PB1
	= 3; // occurring at the same time PB0+PB1.
	If both rising and falling edges are enable then the values can be:
	= 10; // PB0 Rising TRIG
	= 20; // PB0 Falling TRIG
	= 30; // PB1 Rising TRIG
	= 40; // PB1 Falling TRIG.							

E41102	Number of NC channels							
	ALL	Cmn	None	1…8		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT16	P97 N0	Osc
Description	This parameter is used to determine the number of auxiliary channels by subtracting parameter E41102 (number of NC channels) from parameter E41002 (number of channels).							

E41103	Returns the NVRAM available in the NCK (in byte)							
	ALL	Cmn	byte	1..131072		4.1.00.0	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT32	Value	Osc
Description	This parameter returns the size of NVRAM available in the NCK. The main use is for diagnostic.							

E41104	Status of Tool management (P7 N3 .3)							
	ALL	Cmn	None	0..1		4.1.0.0	NORST	
	Rpp	NOPLC	RENA	NODop	RWDOpC	SINT8	P7 N3.30	-
Description	This parameter returns the status of P7 N3.3. This gives information whether the Tool Management option is active or not.							

E41105	Change CNC Operating mode							
	ALL	Chnl	None	0..15		4.1.10.0	NORST	
	RWpp	NoPLC	RWENA	NODop	RWDOpC	SINT8	0	Osc
Description	This parameter permits to change the CNC operating mode: • 0: continuous mode • 1: single mode • 2: MDI mode • 3: rapid mode • 4: sequence number search mode • 6: test mode • 7: manual mode • 8: home mode.							

E43000 to E43009	Indirect addressing of the PLC Inputs							
	I/O	NO	None	0…255		-	NORST	
	Rpp	Wplc	RENA	R (type 3)	RWDOpC	SINT8	Value	-
Description	Parameters used to retrieve image of the PLC inputs via dedicated *"Flexium_NCK.RCNC.E43000[x]"* variables. They can be read by part program. These parameters can be read via dynamic operator. The function is defined by the machine builder.							

E43010 to E43999	Indirect addressing of the PLC Inputs							
	I/O	NO	None	0…255		-	NORST	
	Rpp	Wplc	RENA	NO	RWDOpC	SINT8	Value	-
Description	Parameters used to retrieve image of the PLC inputs via dedicated *"Flexium_NCK.RCNC.E43000[x]"* variables. They can be read by part program. The function of these parameters is defined by the machine builder. They are equivalent to parameters E43000 to E43009 except that they cannot be read via dynamic operator. They can be read by dynamic operators in C but at the same pace than by part program.							

E44000 and E44001	SINT8 Read parameters to access the NCK embedded digital inputs							
	I/O	Cmn	None	0…255		-	NORST	
	Rpp	Rplc	RENA	R (type 3)	RWDOpC	SINT8	Value	Osc
Description	Parameters used to access the NCK embedded digital inputs: • E44000 = *in7..in0* connector X10. • E44001 = *in15..in8* connector X10. They can be read by part program and via dynamic operators. For additional information please refer to Flexium+ Extended Programming manual M00055 "Dynamic operators".							

E44100 to E44103	UINT16 Read parameters access the NCK embedded analogues inputs							
	I/O	Cmn	None	0…4095		-	NORST	
	Rpp	Rplc	RENA	R (type 3)	RWDOpC	UINT16	Value	Osc
Description	Parameters used to access the NCK embedded analogue inputs: • E44100 = Analogue *"in0"* connector X9. • E44101 = Analogue *"in1"* connector X9. • E44102 = Analogue *"in2"* connector X9. • E44103 = Analogue *"in3"* connector X9. They can be read by part program and via dynamic operators. The relationship between value and output voltage is: 0V -> 0 +10V -> 4095. Please also refer to Flexium+ Extended Programming manual M00055 "Dynamic operators".							

E46000 to E46002	UINT16 Read parameters to access the PLC memory							
	ALL	Cmn	None	0…65535		-	NORST	
	Rpp	Wplc	RENA	R (type 3)	RWDOpC	UINT16	Value	-
Description	Parameters used to retrieve numerical values from the PLC program on "Flexium_NCK.RCNC.E46000[x]" variable. They can be read by part program. The function of these parameters is defined by the machine builder. The parameters up to E46002 can be read via dynamic operator. Please also refer to Flexium+ Extended Programming manual M00055 "Dynamic operators".							

E46003 to E46999	UINT16 Read parameters to access the PLC memory							
	ALL	Cmn	None	0…65535		-	NORST	
	Rpp	Wplc	RENA	NODop	RWDOpC	UINT16	Value	-
Description	Parameters used to retrieve numerical values from the PLC program on "*Flexium_NCK.RCNC.E46000[x]*" variable. They can be read by part program. The function of these parameters is defined by the machine builder. They are equivalent to parameters E46000 to E46002 except that they cannot be read via dynamic operator. Please also refer to Flexium+ Extended Programming manual M00055 "Dynamic operators".							

E47000 to E47127	UINT8 Read input parameters to access the PLC memory							
	ALL	Cmn	None	0…255		-	NORST	
	Rpp	Wplc	RENA	R (type 3	RWDOpC	SINT8	Value	Osc
Description	Parameters used to retrieve numerical values from the PLC program on the *"Flexium_NCK.RCNC.E47000[x]"* variable. They can be read by part program and via dynamic operator. The function of these parameters is defined by the machine builder. For additional information, please refer to Flexium+ Extended Programming manual M00055 "Dynamic operators".							

E49001 to E49128	Dynamic Operation number							
	Prg	Cmn	None	0...25		-	RSTYES	
	Rpp	NOPlc	RENA	NODop	RWDOpC	SINT16	0	Osc (diff. value)
Description	These parameters giving the dynamic operator numbers declared in operations 1 to 128. For additional information please also refer to Flexium+ Extended Programming manual M00055 "Dynamic operators".							

E49200 to E49328	Reserved : Dynamic Operation elapsed time							
	Prg	Cmn	µs	0...32767/-1 (overflow)			RSTYES	
	Rpp	NO	RENA	NODop	RWDOpC	SINT16	0	Osc
Description	This parameter is used to measure the time required for an operation. It is validated by E13007. This parameter is reserved for diagnostic by NUM and should not be used in operation.							

E50000	Current tool correction number							
	Tools	Chnl	None	0…255		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	SINT16	Value	Osc (diff. value)
Description	This parameter is related to the D function. It reflects the current tool offset number selected. The parameters E50000 to E57255 are valid when the tool management is not active.							

E50001 to E50255	Tool length or X dimension of tool offsets							
	Tools	Cmn	P4N2	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	xxx means from 001 to 255. This parameter corresponds to the values of the tool length (Milling tool) or X dimension (Turning tool) expressed in internal dimension units. This range of parameters is valid only when the tool management option is not active.							

E51000	Current tool axis orientation							
	Tools	Chnl	None	0…102		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	SINT16	Value	Osc (diff. value)
Description	Parameter giving the physical address of the axis parallel to the tool direction (G16...). If the tool axis orientation is negative, the value of 100 is added to the axis address, i.e.: • 0 for X+ 100 for X- • 1 for Y+ 101 for Y- • 2 for Z+ 102 for Z- Example: E51000 = 2 (positive orientation on the Z axis), E51000 = 102 (negative orientation on the Z axis). For additional information, please refer to the G16 on chapter 5. This range of parameters is valid only when the tool management option is not active.							

E51001 to E51255	Tool end radius or Z dimension of tool offsets							
	Tools	Cmn	P4N2	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	xxx means from 001 to 255. This parameter corresponds to the values of the tool length (Milling tool) or Z dimension (Turning tool) expressed in internal dimension units. This range of parameters is valid only when the tool management option is not active.							

E52001 to E52255	Tool radius							
	Tools	Cmn	P4N2	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	xxx means from 001 to 255. This parameter corresponds to the values of the tool nose radius (Turning tool) or shaft radius (milling) expressed in internal dimension units. This range of parameters is valid only when the tool management option is not active.							

E53001 to E53255	Dynamic length (or X) incremental correction							
	Tools	Cmn	P4N2	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	xxx means from 001 to 255. This parameter corresponds to the values of the dynamic tool correction in length (Milling tool) or diameter (Turning tool) expressed in internal dimension units. This range of parameters is valid only when the tool management option is not active.							

E54001 to E54255	Dynamic radius (or Z) incremental correction							
	Tools	Cmn	P4N2	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	xxx means from 001 to 255. This parameter corresponds to the values of the dynamic tool correction in radius (Milling tool) or in Z (Turning tool) expressed in internal dimension units. This range of parameters is valid only when the tool management option is not active.							

E55001 to E55255	Tool tip orientation							
	Tools	Cmn	None	0...8		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	SINT8	Value	Osc
Description	xxx means from 001 to 255. This parameter corresponds to the value of the tool tip orientation (Turning or Boring tool). This range of parameters is valid only when the tool management option is not active.							

E56001 to E56255	Tool table H parameter (available)							
	Tools	Cmn	None	+/-2^31-1		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	SINT32	Value	Osc
Description	This parameter is used for management of tool wear or any other information (maximum 8 digits) as it does not have a predefined use in the tool management by the NCK. The values can be manipulated by the part program. A common use is for tool life management The values of H can be entered manually (see Flexium+ HMI Operator Manual M00039). This range of parameters is valid only when the tool management option is not active.							

E57001 to E57255	Tool type definition							
	Tools	NO	None	0...2		-	NORST	
	RWpp	NO	RWENA	R (type2)	RWDOpC	SINT8	Value	Osc
Description	xxx means from 001 to 255. This parameter corresponds to the tool type: • 0: Milling tool, • 1: Turning tool, • 2: Boring tool. This range of parameters is valid only when the tool management option is not active.							

E58000	Tool loaded in main spindle							
	Tools	Chnl	None	0…2^32-1		4.1.0.0	NORST	
	Rpp	RWplc	RENA	R (type2)	RWDOpC	UINT32	Value	Osc
Description	This parameter returns the tool currently present in the spindle. It is filled automatically when the Tool Management option is present. Without this option it can be filled by the PLC program method : *LoadToolNumber*.							

E58001 to E58255	Tool geometry code / Tool in spindle							
	Tools	Cmn	None	0...2^32-1		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	-	Value	Osc
Description	Unless it is used in the machine application, this is a free parameter which can be handled in a similar way as the E56xxx parameters (except that it can not be entered via HMI). This list of parameter is not used by the CNC software. It is available for the OEM. The general usage is for storing a tool geometry characteristic. This range of parameters is not available in case of Tool Management. When tool manager is active this parameter indicates which tool number is available in each spindle.							

E59000	Overstock in roughing							
	Tools	Chnl	P4N2	0….5		-	RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter permits to create an overstock during roughing operation by permitting to increase the tool radius offset value applied in compensation (G41 - G42). It should be reset to 0 to achieve the real path.							

E6x000	DAT1 on axis x (x = axis address 0 to 8)							
	Axes	Chnl	P4N2	any value (double range)			NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	These parameters correspond to the current DAT1 values in internal units (µm in general).							

E6x001	DAT2 on axis x (x = axis address 0 to 8)							
	Axes	Chnl	P4N2	any value (double range)			NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	These parameters corresponds to the DAT2 values in internal units (µm in general).							

E6x002	Minimum dynamic machine travel on the axis "x"							
	Axes	Chnl	P4N2	any value (double range)			RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Minimum	Osc
Description	These parameters correspond to the maximum dynamic machine axis travel in internal unit (µm in general) . The machine travels are constrained by software limits entered when setting up the machine (values contained in machine parameter P17). It may prove necessary to modify these limits according to the part to be machined or the environment (collision avoidance). Parameters E60002 and E60003 are used to modify the limits inside those defined by parameter P17. At power on, after a reset or an M02, these parameters are initialized with the value 99999999 (+ or - according to whether it is the maximum or minimum). During machining or manual move, the system makes sure the tool tip remains within the limits defined.							

E6x003	Maximum dynamic machine travel on the axis "x"							
	Axes	Chnl	P4N2	any value (double range)			RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Maximum	Osc
Description	These parameters correspond to the maximum dynamic machine axis travel in internal unit (µm in general) . Please refer to E6x002 parameter for the description.							

E6x004	Eccentricity offsets on axis "x" (DAT3)							
	Axes	Chnl	P4N2	any value (double range)			NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	These parameters are used to define the eccentricity values (equivalent to DAT3). The values are expressed in internal units (x=axis address from 0 to 5).							

E6x005	Offset programmed by G59 on axis "x"							
	Axes	Chnl	P4N2	any value (double range)			RSTYES	
	RWpp	NO	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc (diff. value)
Description	These parameters are used to define the G59 offset along an axis. The values are expressed in internal units. For additional information please refer to the G59 on chapter 5.							

E6x006	RESERVED : WPC Linear and rotary offsets							
	Prg	Chnl	P4N2	any value (double range)		-	RSTYES	
	RWpp	NO	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	This channel specific parameter defines the position offsets of the stock origin with respect to program origin for the linear axes (x = 0..5) and the Euler angles of stock orientation shift for the rotary axes.							

E6x007	RESERVED : WPC Angular offsets applied on A / B / Linear and rotary offsets							
	Prg	Chnl	P4N2	any value (double range)		-	RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	This channel specific parameter contains the information related to the compensation of workpiece orientation applied to the rotary axes (A, B, C) involved in the orientation of the tool (head rotation type), the stock (table rotation type), or both in mixed type configuration. It is meaningless for linear axes therefore the range of values for x is 6 to 8 only.							

E6x008	Monitoring a minimal dynamic travel							
	Axes	Chnl	P4N2	Any value (double range)		4.1.20	YESRST	
	RWpp	NOPLC	RWENA	R (type 2)	RDOpC	DOUBLE	NoBckp	Osc
Description	This parameter defines a minimum travel but execution is not stopped if it is trespassed; The information about trespassing is returned in E7x008.							

E6x009	Monitoring a maximal dynamic travel							
	Axes	Chnl	P4N2	Any value (double range)		4.1.20	YESRST	
	RWpp	NOPLC	RWENA	R (type 2)	RDOpC	DOUBLE	NoBckp	Osc
Description	This parameter defines a maximum travel but execution is not stopped if it is trespassed; The information about trespassing is returned in E7x008.							

E6x012	Angular offset on RTCP rotary axes							
	Axes	Chnl	P4N2	Any value (double range)		4.1.20	NORST	
	RWpp	NOPLC	RWENA	R (type 2)	RDOpC	DOUBLE	NoBckp	Osc
Description	This parameter permits to define the angular offsets on the rotary axes.							

E69000	Scaling factor							
	Prg	Chnl	1/1000	1...10000		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	UINT16	Value	Osc
Description	These parameters are used to define the scaling factor. The values are expressed in thousands from 1/1000 up to 10000/1000.							

E69001	Inclined axis angle (grinders)							
	Prg	Chnl	1/1000000	+/-80000000		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	This parameter is used to define the angle between X and Z axes on a grinder.							

E69002	Inclined wheel angle (grinders)							
	Prg	Chnl	1/1000000	+/-80000000		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	This parameter is used to define the wheel orientation angle on a grinder.							

E69003	**Axis re-assignment**							
	Axes	Chnl	Coded	0...828180		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	UINT32	0	Osc (diff. value)

Description

Axis re-assignment modifies the part programming three-axis reference system Xp Yp Zp with respect to the physical reference system of the machine, Xm, Ym, Zm to modify the programming reference system according to the workpiece position on the machine.

Axis re-assignment applies only to three-axis milling channels and can be forward (positive) or reverse (negative). It should be noted that axis assignment:

- Repositions the programmed dimensions, the interpolation plane (possibly reversing the direction of rotation of circles), the PQR material vectors, the coefficients of the polynomial functions, the 2D tool correction (possibly reversing G41/G42) and the tool direction (G16...)
- Is applied after programmed geometric transformations such as angular offset (ED..), programmed offset (G59..), mirroring (G51..) and scaling factor (G74).

Axis assignment does not apply to dimensions programmed with respect to the measurement origin (G52..). Datum shifts DAT1, DAT2 and DAT3 are not affected by axis assignment. When programming G16.., the programmed tool direction is defined with respect to the axis assignment and, after a system reset, assumes the assignment of the program Z.

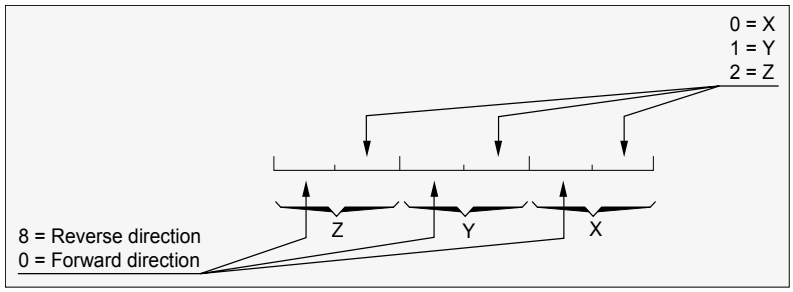
Axis assignment by parameter E69003

Parameter E69003 is used to read the current assignments or declare new ones. This parameter includes three words:

- The units and tens digits contain the X axis assignment,
- The hundreds and thousands digits contain the Y axis assignment,
- The ten thousands and hundred thousands digits contain the Z axis assignment.

In each word, the value 0 addresses the X machine axis, the value 1 addresses the Y machine axis and the value 2 addresses the Z machine axis to which is added the value 80 if the assignment is reversed.

Example:
Axis assignment XR+ YQ- ZP- is declared by E69003 = 808102.



Remarks

The tool direction (G16..) in the program reference system must be declared again after changing the axis assignments by programming parameter E69003.

A format error in programming of E69003 returns error message 94.

Programming of turning planes G20, G21 and G22 with reassigned axes returns error message 2.

Programming of E69003 with a turning plane not disabled returns error message 95.

For additional information, please refer to the G16 in this manual.

E69004	Tool disengagement in G41/G42 for an angle bigger that 120°							
	Prg	Chnl	P4N2	any value (double range)		-	RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter is used to prevent contact between the tool and the part at the angle tip. For additional information, please refer to the G41/G42 on chapter 5.							

E69005	Tilted nozzle compensation : Tilt angle value							
	Cutting	Chnl	P4N2	0 to 60° in internal units		-	RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter is used with the tilted nozzle compensation option. It defines the orientation angle of the cutting head.							

E69006	Tilted nozzle compensation : Height of the part							
	Cutting	Chnl	P4N2	0 to 10 mm in internal units		-	RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter is used with the tilted nozzle compensation option. It defines the height of the part to being cut.							

E69007	Tilted nozzle compensation : Radius smoothing for jet cutting							
	Cutting	Chnl	P4N2	0 to 50 mm in internal units		-	RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter is used with the tilted nozzle compensation option. For additional information please refer to Flexium⁺ Extended Programming manual M00055.							

E69008	Tilted nozzle compensation : Type of head for jet cutting							
	Cutting	Chnl	None	0...65535		-	RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	UINT16	0	Osc
Description	This parameter is used with the tilted nozzle compensation option. The value is coded and defined the structure of the cutting head. For additional information please refer to Flexium⁺ Extended Programming manual M00055.							

E6xnn0	DAT1 offsets table number nn applied on axis x							
	Prg	Chnl	P4N2	any value (double range)		-	RSTYES	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	These parameters represent the list of all DAT1 values for axis x internal units (µm in general).							

E6xnn1	DAT2 offset table number nn applied on axis x							
	Prg	Chnl	P4N2	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	These parameters represent the list of all DAT2 values for axis x internal units (µm in general).							

E79000	Reference position of the current measured spindle							
	Spindle	Chnl	1/1000000 Deg.	0...3599999		-	NORST	
	Rpp	NOPlc	R (in internal unit. 360 degrees = E90432)	R (type2)	RWDOpC	SINT32	0	Osc (diff. value)
Description	Position in millionths of a degree (must increases in M4). When the position is accessed by dynamic operators, it is expressed as 1/4096 revolution modulo 2^16. This is a Read-only parameter (except via the dynamic operators).							

E79001	Speed demand for the current main spindle							
	Spindle	Chnl	1/16384	+/-32767		-	RSTYES	
	Rpp	NOPlc	RENA	RW (type1)	RWDOpC	SINT16	0	Osc
Description	The value is: (speed demand/max speed of the range) x 16384 In parametric programming, this parameter is a read-only parameter and returns the programmed value (S) as a function of the speed range. This value is encoded on 15 bits. When this parameter is programmed by dynamic operators, the sign bit is used by the system only in state M05, Otherwise, it is forced to insure the direction of rotation programmed by M03 or M04. The spindle potentiometer is not active in this case.							

E79002	Value of the feed rate potentiometer assigned to the channel							
	Axes	Chnl	1/128	0,,,32767		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	SINT32	Value	Osc
Description	These parameters reflects the feed rate override coefficient. (E79002 = 128 corresponds to the 100 percent setting of the potentiometer). For use with dynamic operators, refer to E79013 whose resolution is higher.							

E79003	Distance remaining to be travelled in the block being executed							
	Axes	Chnl	P4N2	any value (double range)		-	RSTYES	
	Rpp	NOPlc	RENA	RW (type1)	RWDOpC	DOUBLE	0	Osc
Description	The distance on the path programmed in the block being executed is expressed in the internal unit of the system. This parameter can only be modified by dynamic operators (read-write). Please also refer to Flexium+ Extended Programming manual M00055 "Dynamic operators". The CNC can in some cases break down the block in smaller ones (e.g. RTCP). In such a case E79003 refers to the distance remaining in the smaller blocks.							

E79004	Current feed rate on the path programmed in the block							
	Axes	Chnl	UINTs/ scan	0...32767		-	RSTYES	
	Rpp	NOPIc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	The value is expressed in internal system unit by sampling. This read-only parameter can only be modified by dynamic operators. For additional information, please refer to the chapter "Dynamic operators" in the Flexium+ Extended Programming manual M00055.							

E79005	Minimal value for feed override							
	Axes	Chnl	None	0...255		-	RSTYES	
	RWpp	NOPIc	RENA	R (type2)	RWDOpC	UINT16	P67	Osc (diff. value)
Description	This parameter is used to define the minimal value of the feed rate override.							

E79006	Maximum value for feed override							
	Axes	Chnl	None	0...255		-	RSTYES	
	RWpp	NOPIc	RENA	R (type2)	RWDOpC	UINT16	P67	Osc (diff. value)
Description	This parameter is used to define the maximum value of the feed rate override.							

E79007	Minimal value for the main spindle override							
	Spindle	Chnl	None	0...255		-	RSTYES	
	RWpp	NOPIc	RENA	R (type2)	RWDOpC	UINT16	P67	Osc (diff. value)
Description	This parameter is used to define the minimal value of the main spindle speed override.							

E79008	Maximum value for the main spindle override							
	Spindle	Chnl	None	0...255		-	RSTYES	
	RWpp	NOPIc	RENA	R (type2)	RWDOpC	UINT16	P67	Osc (diff. value)
Description	This parameter is used to define the maximum value of the main spindle speed override.							

E79009	Distance to go within the block							
	Axes	Chnl	P4N2	any value (double range)		-	RSTYES	
	Rpp	NOPIc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter is used to find the distance to be travelled within the block. It is generally used in conjunction with E79010 to determine the percentage of execution of a block.							

E79010	Remaining distance within the block							
	Axes	Chnl	P4N2	any value (double range)		-	RSTYES	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter gives the distance remaining in the current programmed block. It is used in conjunction with E79009 to compute the percentage of execution of a block. This parameter is very similar to E79003 except that E79003 indicates the distance per broken down blocks when they are some (e.g. RTCP). Additionally this parameter is Read only even by dynamic operators.							

E79011	Expansion factor							
	Axes	YES	None	any value (double range)		-	RSTYES	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	The CNC may in some case *"zoom"* the interpolation data to increase precision. This parameter gives the expansion factor.							

E79012	Current feed rate on the path programmed in the block NOT filtered							
	Axes	Chnl	Internal unit/NC sample	0...32767		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	NO	Osc
Description	This parameter is equivalent to E79004 but the value is not filtered.							

E79013	Feed rate override writeable with the Dyn Operator							
	Axes	Chnl	1/16384	0...32767		4.1.00.0	RSTYES	
	Rpp	NOPlc	RENA	RW (type1)	RWDOpC	UINT32	0	Osc
Description	This parameter defines the feed override value written by dynamic operator. Its value will be taken into account when the value it is different of 0. This parameter is for use with dynamic operators due to its higher resolution. It is activated by P7 N3.7.							

E79014	Maximum feed rate in G92 L or G92 Q							
	Prg	Chnl	None	Any value		4.2.0	NO	
	RWPP	NOPLC	NOENA	NODOp	RDOPC	UINT32	NOBckp	Osc
Description	In G92 L or G92 Q the cartesian velocity could become very high when executing small circumferences with a big tool radius. This parameter permits to limit such velocity. A value of 0 means there is no limitation.							

E79020 to E79022	Tool vector orientation in machine referential - valid in RTCP							
	Axes	Chnl	Vector norme	-1 <= .. and <= +1		4.1.20	NORST	
	Rpp	NOPLC	NOENA	R (type 1)	RDOpC	DOUBLE	NOBckp	Osc
Description	This parameter returns the tool vector coordinates expressed in the machine referential.							

E79023 to E79025	Tool vector orientation in turntable referential - valid in RTCP							
	Axes	Chnl	Vector norm	-1 <= .. and <= +1		4.1.20	NORST	
	Rpp	NOPLC	NOENA	R (type 1)	RDOpC	DOUBLE	NOBckp	Osc
Description	This parameter returns the tool vector coordinates expressed in the turntable referential.							

E79026 to E79028	Tool vector orientation in program referential - valid in RTCP							
	Axes	Chnl	Vector norm	-1 <= .. and <= +1		4.1.20	NORST	
	Rpp	NOPLC	NOENA	R (type 1)	RDOpC	DOUBLE	NOBckp	Osc
Description	This parameter returns the tool vector coordinates expressed in the program referential.							

E7x000	Position reference of an axis in the channel							
	Prg	Chnl	P4N2	any value (double range)		-	RSTYES	
	Rpp	NOPlc	RENA	RW (type1)	RWDOpC	DOUBLE	Value	Osc (diff. value)
Description	This parameters defines the axes position reference (with respect to the machine origin OM) as generated by the interpolator. The values are expressed in internal units. These are Read-only parameters (except by dynamic operators) Please also refer to Flexium+ Extended Programming manual M00055 "Dynamic operators".							

E7x002	Minimum static machine travel							
	Axes	Chnl	P4N2	any value (double range)		-	NORST	
	Rpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	P17	Osc
Description	This parameter is the image of machine parameter P17 (see also E6x002 and E6x003). For additional information, please refer to the P17 machine parameter in the Flexium+ Parameters manual M00041.							

E7x003	Maximum static machine travel							
	Axes	Chnl	P4N2	any value (double range)		-	NORST	
	Rpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	P17	Osc
Description	This parameter is the image of machine parameter P17 (see also E6x002 and E6x003). For additional information, please refer to the P17 machine parameter in Flexium+ Parameters manual M00041.							

E7x004	Direction of movement of the axes being interpolated							
	Axes	Chnl	P4N2	any value (double range)			NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter is used only in conjunction with dynamic operators. With linear interpolation, this parameter contains the incremental move for the current block on the relevant axis. With circular interpolation, this parameter contains the component on axis x of the vector with length R tangent to the path (applies only to the linear axis of the interpolation plane).							

E7x005	Axes address re-assignment							
	Axes	Chnl	None	1...32 / 1000…1031 (remote axes)		-	RSTYES	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	SINT16	P9	Osc (diff. value)

Description

These parameters (channel dependent) return the physical address of a program axis (a program axis is assigned to a physical axis by machine parameter P9).

When no physical axis is assigned to the program axis, the parameter takes on the value -1.

Examples:

- If word N8 of parameter P9 has the value 26 (program axis 6=A of channel 3 assigned to physical address 8) then in the third channel, E76005 is equal to 8 (physical axis 8 assigned to program axis 6).

Writing to these parameters can assign a different physical axis to a program axis.

A physical axis must be freed before being able to be assigned to a new program axis (by assigning the value -1 to the parameter of the program axis with which the physical axis was originally associated).

Example:

- In a channel, E75005=2 (physical axis 2 assigned to program axis 5=W).
- In the same channel, E74005=8 (physical axis 8 assigned to program axis 4=V).
- To assign physical axis 8 to program axis 5 (W):
  - release physical axis 8 from program axis 4 (E74005=-1),
  - assign physical axis 8 to program axis 5 (E75005=8).



A new physical axis assigned to a program axis must be of the same nature as the previous axis (linear or rotary axis, servo or not, modulo or with limited excursion, etc.).

A reset or end of program (M02) restores the default conditions for the axes setting.

Please refer to machine parameter P9 on Flexium+ Parameters manual M00041.

E7x006	Axis coupling							
	Axes	Chnl	None	0/not 0			RSTYES	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT16	P64	Osc (diff. value)

Description

These parameters are used to create or cancel carrier/carried axis pairs.

They apply to primary or secondary linear axes (program axis number between 0 and 5).

The value «0» assigned to the parameter cancels the primary/secondary axis pairing to which the axis specified belongs and a «1» specifies that the axis is carried.

When the parameter related to an axis is modified, the parameter related to the other axis in the primary/secondary axis pair is automatically modified.

A reset reinstates the axis pairs as specified by machine parameter.

E7x007	Axes programmed by diameter							
	Axes	Chnl	None	0/not 0		-	RSTYES	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	UINT8	P4	Osc (diff. value)

Description

Only parameters E70007 (X axis) and E73007 (U axis) are programmable by diameter.

These parameters are reset for programming by radius and set for programming by diameter.

E7x008	Axes with movement outside monitored dynamic limits							
	Axes	Chnl	P4N2	Any value (double range)		4.1.20	NORST	
	RWpp	NOPLC	NOENA	RW (type 1)	RDOpC	UINT 8	NOBckp	Osc
Description	This parameter indicates when an axis has passed the position window defined by E6x008 / E6x009. This permits to monitor the position without triggering an error. It is available for writing only to be reset.							

E7x009	Position recorded on the fly expressed in program referential (MCS or ICS)								
	Axes	Chnl	P4 N2	Any value (double range)		4.1.40		NORST	
	RPP	NOPLC	NOENA	RW (type 1)	RDOpC	RWDOPC	Double	Value	Osc
Description	Equivalent to E7x001 with the difference that the value is expressed in the program referential instead of the machine referential.								

E7x100	Interpolator position references							
	Axes	Chnl	P4N2	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	RW (type1)	RWDOpC	DOUBLE	Value	Osc
Description	This parameter return the position references from the interpolators. The interpolated references E7x100 (with or without transformation) are copied into the references used in servo- controls (E7x000) only for the axes declared as measured and Interpolated. Interaxes calibration uses these references after transformation for computing corrections.							

E7x101	Interpolation speed limiting							
	Axes	Chnl	1/100	0...100		-	RSTYES	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	100	Osc (diff. value)

Description	Write of this parameter limits the interpolation speed on logical axis x in the channel where it is programmed according to the relation:
	$V\ ipo\ (x) = P30 * \frac{E7x101}{100}$
	However, the maximum speed on the axis is not modified.
	The limit value accepted is between 0 and 100 (percent) (rounded off to the nearest integer).
	These parameters are forced to 100 percent at power on. They are all reset to 100 percent after a reset.
	Possible uses of this parameter:
	• Typical use: reserves part of the speed for geometric transformations carried out downstream of the interpolation, for instance with RTCP with or without axis driven in NM_AUTO mode (see Flexium Extended programming manual M00020EN).
	• Special use with E7x101 = 0: axis x is assigned a zero speed:
	  • if it is programmed alone in a block, it is ignored
	  • if it is programmed with other axes, only the movements on the other axes are carried out.
	It should be noted that:
	• Programming a non-CNC channel or programming a non-existent axis returns error message 91.
	• If the programmed value is negative or above 100, the system returns error message 92.							

E7x200	EGB Synchro homing on first leader axis							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the homing offset of the first leader axis. For information about this function please refer to the Commissioning manual M00044.							

E7x201	EGB Synchro homing on second leader axis							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the homing offset of the second leader axis. For information about this function please refer to the Commissioning manual M00044.							

E7x202	EGB Synchro homing on third leader axis							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the homing offset of the third leader axis. For information about this function please refer to the Commissioning manual M00044.							

E7x203	EGB Synchro homing on fourth leader axis							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the homing offset of the fourth leader axis. For information about this function please refer to the Commissioning manual M00044.							

E7x204	EGB Synchro homing on fifth leader axis							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the homing offset of the fifth leader axis. For information about this function please refer to the Commissioning manual M00044.							

E7x205	EGB Synchro homing on follower axis							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the homing offset of the follower axis. For information about this function please refer to the Commissioning manual M00044.							

E7x206	Current EGB tolerance							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	P116	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the synchronization tolerance. It is equivalent to the parameter P116 words N0 to N31. For information about this function please refer to the Commissioning manual M00044.							

E7x207	Current EGB Synchro error							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the current synchronization error. For information about this function please refer to the Commissioning manual M00044.							

E7x208	Maximum value reached in Ex7x207							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the maximum value of the synchronization error. For information about this function please refer to the Commissioning manual M00044.							

E7x209	Minimum value reached in Ex7x207							
	EGB	Chnl	P4N2	any value (double range)		4.0.30.0	NORST	
	RWpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This parameter, only significant with the Electronic Gearbox features, contains the minimum value of the synchronization error. For information about this function please refer to the Commissioning manual M00044.							

E7x210	Consider leader axis as linear or rotary							
	EGB	Chnl	None	0 - 8		4.2.0	NO	
	RWPP	NOPLC	NOENA	Rdop (type 2)	RDOPC	Byte	NOBckp	Osc
Description	In G92 L or G92 Q the cartesian velocity could become very high when executing small circumferences with a big tool radius. This parameter permits to limit such velocity. A value of 0 means there is no limitation.							

E7x211	MLEGB works in OP (0) or OM (1) referential							
	EGB	Chnl	None	0 - 8		4.2.0	NO	
	RWPP	NOPLC	NOENA	Rdop (type 2)	RDOPC	Byte	NOBckp	Osc
Description	Defines the referential of the EGB data							

E7x212	Slave axis respects position, velocity and acceleration (1)							
	EGB	Chnl	None	0 - 8		4.2.0	NO	
	RWPP	NOPLC	NOENA	Rdop (type 2)	RDOPC	Byte	NOBckp	Osc
Description	The follower position is tightly linked to the leader position. This means that no control on velocity or acceleration of follower axis occurs, it simply follows the leader wheatear or not it had to overcome the dynamic limitations. If the follower is requested to respect his own velocity and acceleration or the position limit, the parameter E7x212 have to be set. With this bit set the axis will respect the position limits, the velocity limits and acceleration limits. This inevitably means that the leader-follower link might be not respected as the leader is not influenced by this parameter but the follower is limited in his dynamic response.							

E7x213	Lower position limit of the axis							
	EGB	Chnl	None	Any value		4.2.0	NO	
	RWPP	NOPLC	RWENA	Rdop (type 2)	RDOPC	Byte	NOBckp	Osc
Description	It is possible to limit the follower axis target position using specific E parameters E7x213 and E7x214 respectively for the negative value and the positive value. The value is interpreted in machine origin or work piece origin according to E7x215 If E7x215 is set the values are interpreted as machine origin, otherwise they are in work piece origin.							

E7x214	Upper position limit of the axis							
	EGB	Chnl	None	Any value		4.2.0	NO	
	RWPP	NOPLC	RWENA	Rdop (type 2)	RDOPC	Byte	NOBckp	Osc
Description	It is possible to limit the follower axis target position using specific E parameters E7x213 and E7x214 respectively for the negative value and the positive value. The value is interpreted in machine origin or work piece origin according to E7x215 If E7x215 is set the values are interpreted as machine origin, otherwise they are in work piece origin.							

E7x215	Values of position limit given in OP (0) or OM (1) referential							
	EGB	Chnl	None	0 - 8		4.2.0	NO	
	RWPP	NOPLC	NOENA	Rdop (type 2)	RDOPC	Byte	NOBckp	Osc
Description	The values of the follower axes are interpreted in machine origin or work piece origin according to E7x215. If E7x215 is set the values are interpreted as machine origin, otherwise they are in work piece origin.							

E80000 to E80199	Local DOUBLE R/W parameters							
	ALL	Cmn	None	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	RW (type1)	RWDOpC	DOUBLE	Value	Osc
Description	Generic purpose parameters. Backed-up.							

E81000 to E81999	Master axis position reference							
	ALL	Cmn	None / P4N2	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	RW (type1)	RWDOpC	DOUBLE	Value	Osc
Description	Interaxes calibration tables - master axes data. These parameters can also be used as generic purpose parameters. Backed-up.							

E82000 to E82999	Slave axis corrections with respect to the master axes							
	ALL	Cmn	None / P4N2	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	RW (type1)	RWDOpC	DOUBLE	Value	Osc
Description	Interaxes calibration tables - slave axes data. These parameters can also be used as general purpose parameters. Backed-up.							

E83000 to E83999	General purpose DOUBLE R/W							
	ALL	Cmn	None	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	RW (type1)	RWDOpC	DOUBLE	0	Osc
Description	Generic purpose parameters. These parameters are not backed-up. They are used in some specific NUM macro and must not be used in any customer application.							

E84000 to E84999	General purpose DOUBLE R/W parameters							
	ALL	Cmn	None	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	RW (type1)	RWDOpC	DOUBLE	0	Osc
Description	Generic purpose parameters. These parameters are not backed-up.							

E88000 to E88999	General purpose NUM specific parameters							
	ALL	Cmn	None	any value (double range)		-	NORST	
	RWpp	NOPlc	RWENA	RW (type1)	RWDOpC	DOUBLE	0	Osc
Description	Generic purpose parameters. These parameters are not backed-up. They are used in some specific NUM macro and must not be used in any customer application.							

E90000 to E90031	Physical axis measure value							
	Axes	Cmn	P4N2	any value (double range)		-	NORST	
	RWpp	NOPlc	RENA	RW (type1)	RWDOpC	DOUBLE	Value	Osc (diff. value)
Description	Returns the physical axis measurement. The value can be written only for non-servo axes.							

E90032 to E90063	Lag error							
	Axes	Cmn	P4N2	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	Value	Osc
Description	This range of parameters returns the lag value of the axes.							

E90400 to E90431	Spindle angular position							
	Spindle	Cmn	1/1000000 Deg	0…360*106		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	Value	Osc (diff. value)
Description	These parameters return the spindle angular position for each spindle of the system. They relate to the physical spindle number.							

E90432 to E90463	Spindle modulo							
	Spindle	Cmn	1/1000000 Deg	0…360*10^6		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	Value	Osc
Description	These parameters return the spindle angular modulo for each spindle of the system. They relate to the physical spindle number. The spindle modulo is the power of two immediately higher than the spindle resolution defined by P40.							

E90464 to E90495	C axis / Spindle measurement conversion used for Spindle/ axis switching							
	Spindle	Cmn	None	0…32767 (-1 error)		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	SINT16	Value	Osc
Description	This is a read only parameter used to display the coefficient between the axis measure increment and the spindle measure increment in case of Spindle / Axis switching. It is intended for verification. For additional information please refer to Spindle/Axis switching in the Flexium+ Commissioning manual M00044.							

E90500 to E90531	Spindle speed setting							
	Spindle	Cmn	None	0…32767		-	RSTYES	
	Rpp	NOPlc	RENA	R (type1)	RWDOpC	UINT16	0	Osc
Description	This parameter returns the speed setting relative to a spindle physical address. The value 32767 corresponds to the full speed of the motor in the selected range. The most significant bit (8000h) defines the direction of rotation.							

E90532 to E90563	Spindle position set value							
	Spindle	Cmn	1/1000000 Deg.	+/-231-1		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	SINT32	0	Osc
Description	Parameter related to the spindle physical address. It is used to define an offset of position in spindle synchronization.							

E90564 to E90595	Spindle on speed set value							
	Spindle	Cmn	1/1000000 Deg.	+/-231-1		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	SINT32	0	Osc
Description	This parameter is used with Spindle synchronization. It defines the on speed set value.							

E90600 to E90631	Maximum spindle Indexing speed							
	Spindle	Cmn	Rpm	0…65535		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT16	P43	Osc
Description	Parameter related to the spindle physical address. The new values set for these parameters are saved until the next NC reset. Value expressed Speed in rev/min. Please refer to the P43 machine parameter on Flexium+ Parameters manual M00041.							

E90632 to E90663	Spindle in position window							
	Spindle	Cmn	P4N2	0…65535		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT16	P44	Osc
Description	Parameter related to the spindle physical address. The new values set for these parameters are saved until the next NC reset. Value expressed in internal units. Please refer to the P40 and P44 machine parameters on the Flexium+ Parameter manual M00041.							

E90664 to E90695	Spindle indexing gain							
	Spindle	Cmn	1/min	0…65535		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT16	P95	Osc
Description	Parameter related to the spindle physical address. The new values set for these parameters are saved until the next NC reset. Value expressed in rev/min/rev. Please refer to the P45 machine parameter on the Flexium+ Parameter manual M00041.							

E90700 to E90731	Spindle indexing acceleration							
	Spindle	Cmn	deg/s² (P32N64..95)	0...99999999.0			NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	P32 N64	Osc
Description	Parameter related to the spindle physical address. The new values set for these parameters are saved until the next NC reset. Value expressed in Deg/s2. Please refer to the P32 machine parameter on the Flexium+ Parameter manual M00041.							

E90732 to E90763	Spindle stopped threshold							
	Spindle	Cmn	Rpm	0…65535		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT16	10	Osc
Description	Parameter related to the spindle physical address. They define the speed threshold below which the spindle is considered to be stopped. This parameter is expressed in rpm. It is initialized with 10 at power on and can be modified by programming. This parameter is also used as tolerance threshold and detection of the speed reached.							

E90764 to E90795	Tolerance interval coefficient							
	Spindle	Cmn	1/256	0…65535		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT16	13	Osc
Description	Parameter related to the spindle physical address. Parameter defining the tolerance interval coefficient as a function of the speed. This parameter expressed as 1/256 is initialized with 13 (i.e. 5 percent) at power on and can be modified by programming.							

E90800 to E90831	Enable synchronization on master spindle							
	Spindle	Cmn	None	0…65535		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT16	0	Osc
Description	These parameters are used in spindle synchronization. For additional information about the use of these parameters will be found in the commissioning manual under "Spindle synchronization". For the master spindle, this parameter when >0 validates a time constant modification according to the speed. For the slave spindles this parameter defines (in 1/1000) the target of time constant reduction at high speeds. e.g. 1200 T1 = T0 /1.2 (T1 time constant at high speeds, T0 at low speeds).							

E90832 to E90863	Spindle speed for maximum torque							
	Spindle	Cmn	Rpm	0…65535		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT16	0	Osc
Description	Information about the use of these parameters will be found in the Flexium+ Commissioning manual M00044 under *"Spindle synchronization"*.							

E90864 to E90895	Spindle speed for maximum power							
	Spindle	Cmn	Rpm	0…65535		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT16	0	Osc
Description	Information about the use of these parameters will be found in the Flexium Commissioning manual M00044 under *"Spindle synchronization"*.							

E90900 to E90931	Position gap for master slave spindle synchronization							
	Spindle	Cmn	IU	+/-2^31-1		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	SINT32	0	Osc
Description	Parameter related to the spindle physical address.							

E90932 to E90963	Spindle Integration constant							
	Spindle	Cmn	IU	+/-32767		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	SINT16	0	Osc (diff. value)
Description	These parameters show the synchronization error for the spindles.							

E90964 to E90995	Spindle Jog speed							
	Spindle	Cmn	RPM	1..60		4.1.20	NORST	
	RWpp	NOPLC	NOENA	R (type 1)	RDOpC	USINT	50	Osc
Description	This parameter permits to define the spindle speed in Jog.							

E91000 to E91031	Servo or non-servo status of the axes							
	Axes	Cmn	None	0/not 0		-	RSTYES	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	P120	Osc (diff. value)
Description	These parameters reflect the servo or non-servo status of an axis (a physical axis is declared servo or non-servo by machine parameter P120 N2) and are used to modify this state. A «1» in this parameter indicates that the corresponding axis is servo controlled and a «0» that it is non-servo. A reset or end of program (M02) reinstates the axes as specified by machine parameter P120.							

E91032 to E91063	Servo or non-servo status of the spindles							
	Spindle	Cmn	None	0/not0/-1		-	RSTYES	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	0	Osc (diff. value)
Description	These parameters reflect the servo or non-servo status of a spindle (a physical spindle is declared servo or non-servo by machine parameter P120 N2) and are used to modify this state. A «1» in this parameter indicates that the corresponding spindle is in position closed loop and a «0» that it is not. A reset or end of program (M02) reinstates the axes as specified by machine parameter P120.							

E91100 to E91131	Homing status on an axis							
	Axes	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	RW (type1)	RWDOpC	UINT32	1	Osc (diff. value)
Description	These parameters indicate whether the axis homing procedure has been done. A «1» indicates that a HOME has not been completed on the axis and a «0» that it has been completed. This parameter addresses the axes.							

E91132 to E91163	Homing status on a spindle							
	Spindle	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	1	Osc (diff. value)
Description	These parameters indicate whether the axis homing procedure has been done. A «1» indicates that a HOME has not been completed on the spindle and a «0» that it has been completed. This parameter addresses the spindles. Since homing is automatic on the spindles at the first turn, this makes it possible to reset the spindle position reference.							

E91164 to E91195	Channel number where the spindle is located							
	Spindle	Cmn	None	-1..8		4.1.20	NORST	
	Rpp	NOPLC	NOENA	R (type 1)	RDOpC	USINT	NoBckp	Osc
Description	This parameter returns for each spindle the channel where it is located. -1 if the spindle is not associated with any channel.							

E91200 to E91231	N/M AUTO axes							
	Axes	Cmn	None	0/not 0		-	RSTYES	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	0	Osc (diff. value)
Description	These parameters define the axes concerned by the N/M AUTO function. Use of the list of E912xx parameters concerns only the servo-controlled axes (error message 99 is generated by an attempt to assign an axis that is not servo-controlled). Parameters E912xx cannot be used on the PLC axes. The number of parameters that can be used is limited to five (error message 99 is generated if this number is exceeded). When the N/M AUTO function is enabled, the axes concerned by this function are no longer controlled by the interpolators. However, the interpolator computation is continued in parallel to allow the path to be resumed after inhibiting of function N/M AUTO. Only a parameter E912xx belonging to the channel can be written (error message 99 is generated by an attempt to write E912xx for a channel different from the one of the part program in which it is programmed). After a reset, parameters E912xx are reset to zero. When a parameter E912xx is reset, the part program in which it is programmed is stopped if the corresponding N/M AUTO axis is being moved by an axis jog or handwheel. Execution of the program is not resumed until the axis is stopped. For additional information on the N/M AUTO function, refer to the Flexium+ Extended Programming manual M00055.							

E91300 to E91331	Clampable axis enable state							
	Axes	Cmn	None	0/not 0			RSTYES	
	RWpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	P8	Osc (diff. value)
Description	This parameter enables or disables axis clamping by programming. Example: • E91308 = 1: axis 08 can be clamped • E91308 = 0: axis 08 cannot be clamped. A reset applies the settings of machine parameter P8 to the axes. The list of axes that can be clamped is sent to the PLC (by PLC variable *RCNC.Axis[i].Clamp*).							

E91400 to E91431	Homing forced by user on an axis							
	Axes	Cmn	None	0/not 0		4.1.0.0	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	0	Osc (diff. value)
Description	This range of parameters indicates (when SET) that the axis home done status has been 'forced' by a voluntary action. In such case the axes can be out of position therefore this parameter should be considered to prevent any dangerous move (Tool change for example).							

E91432 to E91463	Homing forced by user on a spindle							
	Spindle	Cmn	None	0/not 0		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	0	Osc (diff. value)
Description	This range of parameter indicates (when SET) that the spindle home done status has been 'forced' by a voluntary action. It is equivalent to parameter E91400 to E91431 but only concerns the spindle In such case the axes can be out of position therefore this parameter should be considered to prevent any dangerous move (Tool change for example).							

E91500 to E91531	Early block change – Status of BlockChanged[0..31]							
	Prg	Cmn	None	0 / Not 0		4.1.20	RSTYES	
	Rpp	RPLC	NOENA	R (type 2)	RDOpC	UINT32	NoBckp	Osc
Description	This parameter indicates an early Block Change. Refer to the EBC function in Flexium+ Extended Programming Manual M00055.							

E91600 to E91631	Early block change – BlockChangeCtrl[0..31]							
	Prg	Cmn	None	0 / Not 0		4.1.20	RSTYES	
	RWPP	RWPLC	NOENA	R (type 2)	RDOpC	UINT32	NoBckp	Osc (diff. value)
Description	This parameter return the status of the Early Block Change function in the PLC. Refer to the EBC function in Flexium+ Extended Programming Manual M00055.							

E92000 to E92031	Axis origin switch enable state (analog axes only)							
	Axes	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	RW (type1)	RWDOpC	UINT32	0	Osc (diff. value)
Description	When this parameter is set the system requests the initialization of the measurement: Absolute encoder: the answer is immediate and the parameter is reset. Non absolute encoder: the parameter is reset when the "zero" encoder is encountered during the homing procedure and the measurement initialized.							

E92100 to E92131	Axis error mask (Delayed Power down)							
	Axes	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	RW (type1)	RWDOpC	UINT32	0	Osc
Description	Delayed power down masks the error display and power fail if the corresponding bit is set. These parameters are intended to be used with dynamic operators in order to mask a machine error on an axis long enough to take any appropriate action (tool disengagement for example).							

E92300 to E92331	List of filtered axes (Tau filter)							
	Axes	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	RW (type1)	RWDOpC	UINT32	0	Osc
Description	This parameter is used to validate or inhibit the phase lag and lead filter. This filter can be used to adjust the lag error for axes with different gain values. For additional information about the TAU filter, please refer to the Flexium+ Extended Programming manual M00055.							

E92400 to E92403	Handwheels errors mask							
	Axes	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	RW (type1)	RWDOpC	UINT32	0	Osc
Description	Delayed power down masks the error display and power fail if the corresponding bit is set. These parameters are intended to be used with dynamic operators in order to mask a machine error on a spindle long enough to take any appropriate action (tool disengagement for example).							

E93000 to E93031	Axis @xx home switch status							
	Axes	Cmn	None	0/1		-	NORST	
	Rpp	NOPlc	RENA	NO	RWDOpC	UINT16	0	Osc
Description	Status of the home switch for the axes. A "1" indicates that the corresponding switch is open.							

E93100 to E93131	Measured axis							
	Axes	Cmn	None	0/1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT32	P120 N4	Osc (diff. value)
Description	A «1» indicates that the axis is measured. This parameter is the inverted image of machine parameter P120 N4.							

E93200 to E93231	Axis declared as rotary modulo 360 degrees							
	Axes	Cmn	None	0/1 (error if different from 0 or 1)		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	UINT32	P1 N0	Osc (diff. value)
Description	A «1» indicates that the axis is declared rotary modulo 360 degrees. This parameter is the image of machine parameter P1 N1.							

E93232 to E93263	Modulo value of axes							
	Axes	Cmn	Deg (P68)	1..10000		4.1.20	NORST	
	RWPP	NOPLC	RENA	NODOp	RDOpC	DOUBLE	P68	Osc
Description	This parameter returns or defines the modulo value of the rotary axes.							

E93300 to E93331	Axis homing direction							
	Axes	Cmn	None	0/1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT32	P15 N0	Osc (diff. value)
Description	A «1» indicates that the axis is moving in the negative direction in homing mode. This parameter is the image of machine parameter P15 Word N0. Please refer to the P15 machine parameter on the Flexium⁺ Parameter manual M00041.							

E93400 to E93431	Homing without switch (analog axes)							
	Axes	Cmn	None	0/1		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT32	P15 N1	Osc (diff. value)
Description	A «1» indicates that the origin switch is not wired on this axis. This parameter is the image of machine parameter P15 N1. Please refer to the P15 machine parameter on the Flexium⁺ Parameter manual M00041.							

E93500 to E93531	Axis in-position window							
	Axes	Cmn	None	0/1			RSTYES	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT32	0	Osc (diff. value)
Description	This parameter is used to define if an axis is stopped and within the in position window or not. Its state is the image of the bits of *RCNC.Axis[i].Moving* transmitted to the PLC.							

E93532 to E93563	Spindle in-position window							
	Spindle	Cmn	None	0/1		-	RSTYES	
	Rpp	NOPlc	RENA	NODop	RWDOpC	UINT32	0	Osc (diff. value)
Description	This parameter is used to define when a spindle is stopped and within the in-position window. The image in the PLC is *RCNC.Spindle[s].PositionReached*.							

E93600 to E93631	Type of encoder (analog axes)							
	Debug	Cmn	-	-		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	SINT32	Value	Osc (diff. value)
Description	There is currently only one type of encoder possible (TTL). This parameter is reserved in case of future evolution.							

E93700 to E93731	Flag Axis in error							
	Axes	Cmn	None	0/1		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	0	Osc
Description	These parameters reflects which axes (if any) have detected a machine error (excess lag).							

E93800 to E93803	Flag Handwheel in error							
	Spindle	Cmn	None	0/1		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	UINT32	0	Osc
Description	These parameters define which handwheel (if any) have detected a machine error (excess lag).							

E94000 to E94031	Inter-axis calibration definition							
	Axes	Cmn	None	-1/0…31		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	SINT16	-1	Osc
Description	These parameters are used to define the relationship between slave and master. For additional information about inter axes calibration please refer to the Flexium+ Commissioning Manual M00044.							

E94100 to E94131	Association of a slave axis with a master axis							
	Axes	Cmn	None	-1/0…31		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	SINT16	P27	Osc
Description	These parameters permit to define relations between slaves and master axes. The value of a particular parameter will contain the physical number of its master axis if any (-1 if no coupling) The axes must be associated before slaving an axis to a master axis for duplication or synchronization. Please refer to the chapter *"Duplicated and synchronized axes"* in the Flexium+ Commissioning Manual M00044.							

E94132 to E94163	Master spindle for spindle synchronization							
	Spindle	Cmn	None	-1/0…31		-	NO	
	RWpp	NOPlc	RENA	NO	RWDOpC	SINT16	NO	Osc
Description	Information about the use of these parameters will be found in the Flexium+ Commissioning Manual M00044 under *"Spindle synchronization"*.							

E94200 to E94231	Association Drive / Axis for Spindle for commutation							
	Axes	Cmn	None	-1/0…31		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	SINT16	-1	Osc
Description	This parameter is used to switch either the Spindle or axis command to the same drive in case of spindle/axis commutation. This parameter defines if the spindle xx is controlled as a spindle (value 32..63) or the associated axis (value 0..31). Value -1 for disconnection. This parameter is mostly used for Spindle / Axis switching. For additional information about this feature please refer to the Flexium+ Commissioning Manual M00044.							

E94300 to E94303	Physical handwheel coupler							
	Axes	NO	None	0..3 (error if out of range)		-	NORST	
	RWpp	NO	RENA	NO	RWDOpC	SINT16	P14/P70	Osc
Description	This parameter is used to switch an axis analog output to an analog coupler.							

E94400 to E94431	Torque and Reference control (0 = NCK ; 1 = FXTools / FXServer)							
	Axes	Cmn	None	0/not 0		4.1.0.0	NORST	
	RWpp	RWplc	RENA	NODop	RWDOpC	SINT8	0	Osc (diff. value)
Description	This range of parameter permits to define if the torque and reference control are managed by the NCK or by Flexium Tools							

E95000 to E95031	Axis reference correction							
	Axes	Cmn	P4N2/N4	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	RW (type1)	RWDOpC	DOUBLE	0	Osc
Description	These parameters can only be written by interaxes calibration or dynamic operators. Please refer to chapter *"Axes calibration"* in the Flexium+ Commissioning Manual M00044, or *"Dynamic Operators"* in the Flexium+ Extended Programming manual M00055.							

E95100 to E95131	Position of the origin switch with respect to the machine origin							
	Axes	Cmn	P4N2/N4	any value (double range)		-	NORST	
	Rpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	P16	Osc
Description	These parameters are the image of the P16 machine parameter. It can be written by ENA and taken into account immediately by the NCK. Please refer to the P16 machine parameter on the Flexium+ Parameter manual M00041.							

E95200 to E95231	Axis (or spindle) measurement correction							
	Axes	Cmn	P4N2/N4	any value (double range)		-	NORST	
	Rpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	These parameters return the current value of the measure correction of a particular axis or spindle. This correction is the sum of corrections due to backlash and axis calibration.							

E95300 to E95331	Differential offset value							
	Axes	Cmn	P4N2/N4	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	RW (type1)	RWDOpC	DOUBLE	0	Osc
Description	These are a Read only parameters returning the amount of differential offset generated by Dynamic operators code 24. For additional information please refer to Dynamic Operator 24 in the Flexium+ Extended Programming manual M00055.							

E95400 to E95431	Measure correction for “volumetric error compensation” (Vecomp)							
	Axes	Cmn	P4N2/N4	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	This range of parameters is only significant with the Volumetric Error Compensation option. It indicates the current measure correction.							

E95500 to E95531	Measure correction from PLC (ENA)							
	Axes	Cmn	P4N2/N4	any value (double range)		-	NORST	
	Rpp	NOPlc	RWENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	These parameters return the measure correction given by the PLC (ENA).							

E95600 to E95631	Measure correction from interaxes calibration							
	Axes	Cmn	P4N2/N4	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	These parameters return the measure correction given by the interaxes calibration table.							

E95700 to E95731	Measure correction from axes calibration plus backlash error compensation							
	Axes	Cmn	P4N2/N4	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	R (type2)	RWDOpC	DOUBLE	0	Osc
Description	These parameters return the measure correction given by the axes calibration table.							

E96000 to E96031	Duplicated axis in automatic modes							
	Gantry	Cmn	None	0/not 0		-	NORST	
	RWpp	NOPlc	RENA	RDop	RWDOpC	UINT32	0	Osc (diff. value)
Description	These parameters are used in case of axes duplication and synchronization. When set to a '1' the axis will be duplicated with its master in the automatic modes. If it is a «0», it is not.							

E96100 to E96131	Duplicated axis in manual mode (JOG)							
	Gantry	Cmn	None	0/1		-	NORST	
	RWpp	NOPlc	RENA	RDop	RWDOpC	UINT32	0	Osc (diff. value)
Description	These parameters are used in case of axes duplication and synchronization. When set to a '1' , the axis will be duplicated with its master in manual mode. (JOG / HOME / INTERVENTION) If it is a «0», it is not.							

E96200 to E96231	Synchronized axis							
	Gantry	Cmn	None	negative values or 0 / positive values		-	NORST	
	RWpp	NOPlc	RENA	RDop	RWDOpC	UINT32	P28	Osc (diff. value)
Description	These parameters are used in case of axes duplication and synchronization. When set to a '1', the axis will be synchronized with its master. If it is a «0», it is not.							

E96300 to E96331	Symmetrically driven axis							
	Gantry	Cmn	None	0/1		-	NORST	
	RWpp	NOPlc	RENA	RDop	RWDOpC	UINT32	0	Osc (diff. value)
Description	These parameters are used in case of axes duplication. When set to a '1', the axis is driven symmetrically. If it is a «0», it is driven in the same direction as the master.							

E96400 to E96431	Axes synchronization on the fly with null offset							
	Gantry	Cmn	None	0/1		4.1.20	NORST	
	RWPP	NOPLC	RENA	RDOp(2)	RDOpC	UINT32	P28	Osc
Description	This parameter permits to synchronize axes on the fly (refer to Flexium+ Commissioning manual M00044).							

E97000 to E97031	Maximum axis speed							
	Axes	Cmn	mm/min or deg/min	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	NODop	RDopC	DOUBLE	P30	Osc (diff. value)
Description	These parameters return the maximum axis speed.							

E97100 to E97131	Axis acceleration at programmed feed rate							
	Axes	Cmn	mm/s² or deg/s²	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	P32	Osc
Description	This parameter defines the acceleration used in interpolation moves (G1, G2, G3, Not rapid Manual) expressed in mm/s² or deg/s².							

E97200 to E97231	Axis acceleration in Rapid							
	Axes	Cmn	mm/s² or deg/s²	any value (double range)		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	P32	Osc
Description	This parameter defines the acceleration used in rapid move (G00, Manual rapid) expressed in mm/s² or deg/s².							

E97300 to E97331	Maximum speed authorized at angle crossing							
	HSC	Cmn	mm/min	1…MAX_SPEED		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	P33	Osc
Description	This parameter is the image of machine parameter P33. Its value is specified in mm/minute. For additional information, please refer to the Flexium+ Parameters manual M00041.							

E97400 to E97431	Dynamic value of axis acceleration at work rate							
	Axes	Cmn	mm/s² or deg/s²	Any value (double range)	4.1.40		NORST	
	RWPP	NOPLC	NOENA	NODOP	RWDOPC	Double	P30	Osc
Description	Equivalent to E97100 to E97131 but modifiable on real time.							

E97500 to E97531	Dynamic value of axis acceleration in rapid							
	Axes	Cmn	mm/s² or deg/s²	Any value (double range)	4.1.40		NORST	
	RWPP	NOPLC	NOENA	NODOP	RWDOPC	Double	P30	Osc
Description	Equivalent to E97200 to E97231 but modifiable on real time.							

E97600 to E97631	Minimum machine travel per axis							
	Axes	Comn	P4N2	Any value (double range)	4.1.40		NO	
	RWPP	NOPLC	NOENA	NODOP	RWDOPC	Double	P17	Osc
Description	Equivalent to E6x002 but access is by physical address for example in case of swapping axes between channels							

E97700 to E97731	Maximum machine travel per axis							
	Axes	Comn	P4N2	Any value (double range)	4.1.40		NO	
	RWPP	NOPLC	NOENA	NODOP	RWDOPC	Double	P17	Osc
Description	Equivalent to E6x003 but access is by physical address for example in case of swapping axes between channels							

E98000 to E98031	Servo coefficient value							
	Axes	Cmn	0.001*DAC inc/ internal inc	0…99999999		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	P73	-
Description	This parameter is used to modify the servo loop coefficient on the analog axes connected to the NCK.							

E98100 to E98131	Axis acceleration anticipation time constant							
	HSC	Cmn	µs	0…65535		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	P19 N0..31	-
Description	When machining at very high speed, this parameter is used to program the axis acceleration anticipation. After an initialization, the value of machine parameter P19 is reused. Please refer to the P19 machine parameter on the Flexium⁺ Parameter manual M00041.							

E98200 to E98231	Amplitude of the anti-stick pulse on reversal							
	HSC	Cmn	internal increment (P4N2/N4)	0…65000000		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	P19 N32...63	-
Description	This parameter is the image of machine parameter P19 (words N32 to N63). It is expressed in internal units. Please refer to the P19 machine parameter on the Flexium+ Parameter manual M00041.							

E98300 to E98331	Time constant to absorb the anti-stick pulse							
	HSC	Cmn	µs	0…655350		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	P19 N64..95	-
Description	This parameter is the image of machine parameter P19 (words N64 to N95). Its value is expressed in micro seconds. Please refer to the P19 machine parameter on the Flexium+ Parameter manual M00041.							

E98400 to E98431	Delay to apply the drive reference (Do not use anymore)							
	Axes	Cmn	µs	-		-	NORST	
	Rpp	NOPlc	RENA	NODop	RWDOpC	DOUBLE	-	-
Description	This parameter is not used for machining. It is intended for calculation purposes. It is used by NUM for some internal macros.							

E98500 to E98531	Partial feed forward per axis (without HSC)							
	Axes	NO	1/256	0…512		-	NORST	
	RWpp	NO	RENA	NO	RWDOpC	DOUBLE	P55 N0..7	-
Description	This parameter can be used to adjust the lag values with axes having very different characteristics. The values are expressed in 1/256. This first set of 32 values defines the partial feed forward without HSC validated.							

E98532 to E98563	Partial feed forward per axis (with HSC)							
	HSC	NO	1/256	0…512		-	NORST	
	RWpp	NO	RENA	NO	RWDOpC	DOUBLE	P55	-
Description	This parameter can be used to adjust the lag values with axes having very different characteristics. The values are expressed in 1/256. This second set of 32 values defines the partial feed forward with HSC validated.							

E98600 to E98631	Anticipation time constant							
	SSB	Cmn	1/100 ms	0..65535	-	NO	NORST	
	RWpp	NOPlc	RENA	NO	RWDOpC	UINT16	0	-
Description	This parameter is used to adjust speed anticipation in SSB operation. For additional information about the SSB (Smart Servo Balance) operation, please refer to the Flexium+ Commissioning manual M00044.							

E98632 to E98663	Delay filter time constant (last 32 values)							
	SSB	Cmn	1/100 ms	0..65535		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	UINT16	0	-
Description	This parameter is used to adjust the delay filter time constant in SSB operation For additional information about the SSB (Smart Servo Balance) operation, please refer to the Flexium+ Commissioning manual M00044.							

E98664 to E98695	Acceleration anticipation time constant for corrector SSB filter							
	SSB	Cmn	1/100 ms	0..65535		-	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	UINT16	0	-
Description	This range of parameters defines the acceleration anticipation with the SSB function (Smart Servo Balance) For additional information about this function, please refer to the Flexium+ Commissioning manual M00044.							

E99000 to E99031	Logical drive address to remote drive address coupling							
	Prg	Cmn	None	0…31		4.1.0.0	NORST	
	RWpp	NOPlc	RENA	NODop	RWDOpC	SINT16	0	-
Description	This parameter is related to the shared axes function. Further information will be available in the Flexium+ Commissioning manual M00044.							

E2000001 to E200xxyy	Tool length L of correction xxx or X dimension for Turning Machines							
	Tools	Cmn	P4N2	+/-999999999		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	DOUBLE	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the length L or X dimension (for turning machines) of the tool in spindle xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E200abcd are available only when the Tool management function is active.							

E2010001 to E201xxyy	Cutter tip radius @ of correction xxx or Z dimension for Turning Machines							
	Tools	Cmn	P4N2	+/-999999999		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	DOUBLE	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the cutter tip radius @ or Z dimension (for turning machines) of the tool in spindle xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E201abcd are available only when the Tool management function is active.							

E2020001 to E202xxyy	Tool radius R of correction xxx							
	Tools	Cmn	P4N2	+/-999999999		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	DOUBLE	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the tool radius R of the tool in spindle xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E202abcd are available only when the Tool management function is active.							

E2030001 to E203xxyy	Dynamic length correction DL of correction xxx							
	Tools	Cmn	P4N2	+/-999999999		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	DOUBLE	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the dynamic length correction DL (or X dimension turning machines) of the tool in spindle xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E203abcd are available only when the Tool management function is active.							

E2040001 to E204xxyy	Dynamic radius correction DR of correction xxx							
	Tools	Cmn	P4N2	+/-999999999		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	DOUBLE	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the dynamic radius correction DR of the tool in spindle. xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E204abcd are available only when the Tool management function is active.							

E2050001 to E205xxyy	Tool tip orientation							
	Tools	Cmn	None	0...8		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	SINT8	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the tool tip orientation of the tool in spindle. xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E205abcd are available only when the Tool management function is active.							

E2060001 to E206xxyy	Available parameters (H of the table of dynamic corrections)							
	Tools	Cmn	None	+/-2^31-1		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	SINT32	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the available H parameter of the tool in spindle. xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E206abcd are available only when the Tool management function is active.							

E2070001 to E207xxyy	Parameter defining the tool type of correction xxx:							
	Tools	Cmn	None	0...2		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	SINT8	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the tool type definition of the tool in spindle. xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E207abcd are available only when the Tool management function is active.							

E2080001 to E208xxyy	Corrector Life Value							
	Tools	Cmn	ms	0…2^32-1		4.1.0.0	NORST	
	Rpp	NOPlc	-	NODop	RWDOpC	SINT32	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the tool life value (Accept only 0 to reset the value) of the tool in spindle xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E208abcd are available only when the Tool management function is active.							

E2090001 to E209xxyy	Corrector Life Warning							
	Tools	Cmn	ms	+/-999999999		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	DOUBLE	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the Life Warning of the tool in spindle xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E209abcd are available only when the Tool management function is active.							

E2100001 to E210xxyy	Corrector Life Error							
	Tools	Cmn	ms	+/-999999999		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	DOUBLE	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the Life Error of the tool in spindle xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E210abcd are available only when the Tool management function is active.							

E2110001 to E211xxyy	Corrector Life Type							
	Tools	Cmn	None	0...3		4.1.0.0	NORST	
	RWpp	NOPlc	RWENA	NODop	RWDOpC	SINT8	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the Life Type of the tool in spindle xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E211abcd are available only when the Tool management function is active.							

E2120001 to E212xxyy	Corrector Max Speed							
	Tools	Cmn	Rpm	0…232-1		4.1.00.0	NORST	
	RWpp	NOPlc	RWENA	R	RWDOpC	SINT32	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the Max speed of the tool in spindle xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E212abcd are available only when the Tool management function is active.							

E2130001 to E213xxyy	Corrector Max Feed							
	Tools	Cmn	mm/min	0…232-1		4.1.00.0	NORST	
	RWpp	NOPlc	RWENA	R	RWDOpC	SINT32	Value	Osc
Description	This range of parameters is only available with the Tool Management option. It defines the Max feed associated with the tool in spindle. xx addresses the spindle logical number where the tool is inserted (00 for main spindle) yy addresses the tool offset number. The parameters in the range E213abcd are available only when the Tool management function is active.							

E3000000 to E3000031	Delay or anticipation for Inter sample M functions							
	ALL	Chnl	µSec	-(P50) ..32000		4.1.10	NORST	
	RWpp	NOPLC	RENA	NODOp	RDOpC	SINT	Bckp	Osc
Description	For details about Inter sample M functions please refer to Flexium⁺ Extended Programming manual M00055.							

E3000032 to E3000063	Delay or anticipation for Timed M							
	ALL	Chnl	mSec	+/-32000		4.1.10	NORST	
	RWpp	NOPLC	RENA	NODOp	RDOpC	SINT	Bckp	Osc
Description	For timed M please refer to Flexium⁺ Extended Programming manual M00055.							

Page deliberately empty

Summary

9 List of Errors and Warnings.....607
9.1 Errors and error extensions summary table 607
9.2 CNC Warning (HMI messages) 679
9.3 Drives warning 682

Page deliberately empty

9 List of Errors and Warnings

9.1 Errors and error extensions summary table

Click on **xxx** to give the error detailed information.



Errors are now presented in the form Error + Extension. The extension is intended to give additional information about the cause of the error.

Error n.	Description
Error 1	Programing error
Error 2	G Function syntax error
Error 3	The attribute of a G code is wrongly positioned
Error 4	Option not enabled or conflict between option and parameter
Error 5	Geometric Programing option not enabled
Error 6	Error in Spline function programing
Error 7	Movements parallel to the inclined axes
Error 8	Tool corrector number too high
Error 9	Too many consecutive non executable blocks
Error 10	iNCK protocol error
Error 11	Shared drives programing error
Error 12	Time out coupling/discarding remote drive
Error 13	Error on EGB axes
Error 14	Inclined plane option missing
Error 15	Rigid tapping and un-tapping
Error 16	Error in activating RTCP
Error 17	Missing closing bracket
Error 18	Servo error: P50 too small
Error 19	Error getting address
Error 20	M02 missing or no current program active
Error 21	Error in single axis positioning
Error 22	Prohibited functions in pole definition blocks
Error 24	Error in the definition of an inclined plane
Error 25	Unknown sub-program or block number
Error 26	Sub-program nesting too deep
Error 27	Tool Radius Correction
Error 28	Syntax error in CSS or working radius definition
Error 29	Error in spindle speed programing
Error 32	Axis referencing error
Error 33	All channels waiting for synchronization
Error 34	Minimum radius for interpolation reached - G21
Error 35	Sequence number not found in SEARCH
Error 36	Two reduced speed programed
Error 37	Maximum feed exceeded in thread cutting
Error 39	Axis synchronization fault
Error 40÷71	Excessive lag value on axis 0 ÷ 31
Error 72	Incremental programing after an incomplete block

Error 73	G38 : Missing argument
Error 75	G20 to G21 or G22 transition
Error 76	Canned cycles forbidden in G21 / G22
Error 77	Tool type not compatible with the operation
Error 78	Syntax error in channel synchronization
Error 79	Invalid TANDEM Configuration
Error 80	Drive not in HALT
Error 81	Wrong spindle configuration
Error 82	First or last bloc of profile definition incomplete
Error 83	M Timed and Inter-sample
Error 91	Unknown parameter or variable
Error 92	Error in assignment to a parameter or variable
Error 93	Dynamic operators (Error in the definition of a parameter)
Error 94	Forbidden operation
Error 95	Trying to write to a read only parameter
Error 96	No E parameter access because the previous block is incomplete
Error 97	Assignment of a parameter in G76 not possible
Error 98	Writing by a channel of a dynamic operator already used
Error 99	Error associated with the N/M AUTO function
Error 101	PGP: Insufficient data in the programing of a circle
Error 102	PGP: The missing coordinate cannot be calculated
Error 103	PGP : Chamfer or Fillet
Error 107	PGP: Circle programed
Error 110	PGP: Syntax error in the first of 2 blocks
Error 111	PGP: Syntax error in the second block
Error 112	PGP: Line-line intersection
Error 113	PGP : Impossible to calculate the point of intersection
Error 114	PGP: Intersection or tangent point not defined by ET+, ET-, ES+ or ES-
Error 121	PGP: Syntax error in last of 3 blocks
Error 122	PGP : The first two blocks are non-secant lines
Error 123	PGP : Data in the 3 blocks doesn't allow the tangent points to be calculated
Error 124	PGP: Tangent point second-third block not defined by ET+ or ET-
Error 130	No axis movement in 1 of the 2 blocks connected by a fillet or chamfer
Error 131	M0, M1 or M2 is not allowed in a block with a fillet or chamfer
Error 133	Curve programing error in Spline and Angular shift
Error 135	A chamfer can only be between 2 lines
Error 136	More than 2 non movement blocks between 2 incompletely defined elements
Error 137	Change of interpolation plane even though the block is not valid
Error 138	Change of interpolation plane must be in G40
Error 139	Two parallel [carried] axes programed in the same block
Error 140	Programing error in radius correction
Error 141	Programing error in G41/G42 with collision avoidance
Error 142	Error in jet cutting activation
Error 143	Error in tool vector programing or re-orientation
Error 144	Quantified axis displacement is different from the increment
Error 145	G29: Absolute value of [P*P + Q*Q + R*R - 1000mm] higher than 1 mm
Error 146	3D Tool offsets, with G29 or G43
Error 147	In jet cutting no path possible
Error 148	Number of interpolated axes higher than the maximum allowed
Error 149	Tool radius too large in relation to the programed trajectory
Error 150	X axis out of limits

Error 151	Y axis out of limits
Error 152	Z axis out of limits
Error 153	U axis out of limits
Error 154	V axis out of limits
Error 155	W axis out of limits
Error 156	A axis out of limits
Error 157	B axis out of limits
Error 158	C axis out of limits
Error 159	Move demanded on a not-homed axis
Error 160	Slave axis homing error
Error 161	Synchronized axis referencing error
Error 162	Emergency stop: G75 not declared
Error 163	Torque and Reference management error
Error 164	Axis out of limits
Error 190	Too many nested jumps or loops [15 max]
Error 191	Syntax error in structured programing
Error 192	Unknown [or forbidden] key word in this program context
Error 193	Structuration error
Error 195	Stack overflow
Error 196	Error in the declaration of the table index
Error 197	Use of an undeclared symbol
Error 198	Syntax error in declaring the symbol for a variable
Error 199	Syntax error in declaring a variable
Error 200	Poor signal or complementarity error on handwheel 0
Error 201	Poor signal or complementarity error on handwheel 1
Error 202	Poor signal or complementarity error on handwheel 2
Error 203	Poor signal or complementarity error on handwheel 3
Error 209	Reboot NC
Error 210	Feedback fault on axis 0
Error 211	Feedback fault on axis 1
Error 212	Feedback fault on axis 2
Error 213	Feedback fault on axis 3
Error 214	Feedback fault on axis 4
Error 215	Feedback fault on axis 5
Error 216	Feedback fault on axis 6
Error 217	Feedback fault on axis 7
Error 218	Feedback fault on axis 8
Error 219	Feedback fault on axis 9
Error 220	Feedback fault on axis 10
Error 221	Feedback fault on axis 11
Error 222	Feedback fault on axis 12
Error 223	Feedback fault on axis 13
Error 224	Feedback fault on axis 14
Error 225	Feedback fault on axis 15
Error 226	Feedback fault on axis 16
Error 227	Feedback fault on axis 17
Error 228	Feedback fault on axis 18
Error 229	Feedback fault on axis 19
Error 230	Feedback fault on axis 20
Error 231	Feedback fault on axis 21
Error 232	Feedback fault on axis 22

Error 233	Feedback fault on axis 23
Error 234	Feedback fault on axis 24
Error 235	Feedback fault on axis 25
Error 236	Feedback fault on axis 26
Error 237	Feedback fault on axis 27
Error 238	Feedback fault on axis 28
Error 239	Feedback fault on axis 29
Error 240	Feedback fault on axis 30
Error 241	Feedback fault on axis 31
Error 245	Digital servo loop fault
Error 246	Emergency request
Error 247	Drive parameter fault DISC NT+
Error 248	Axis drive or digital spindle fault
Error 249	Communication error between CNC and PLC
Error 250	Invalid PERSOChip
Error 251	Inconsistent PERSOChip Data
Error 252	Invalid license File
Error 253	Nesting of subroutines called by M function not allowed
Error 254	Floating point unit exception error
Error 266	Insufficient memory
Error 267	Please get in contact with NUM AG Switzerland. The system has a major problem
Error 300	Axis/Spindle 0 DISC-NT+ configuration fault
Error 301	Axis/Spindle 1 DISC-NT+ configuration fault
Error 302	Axis/Spindle 2 DISC-NT+ configuration fault
Error 303	Axis/Spindle 3 DISC-NT+ configuration fault
Error 304	Axis/Spindle 4 DISC-NT+ configuration fault
Error 305	Axis/Spindle 5 DISC-NT+ configuration fault
Error 306	Axis/Spindle 6 DISC-NT+ configuration fault
Error 307	Axis/Spindle 7 DISC-NT+ configuration fault
Error 308	Axis/Spindle 8 DISC-NT+ configuration fault
Error 309	Axis/Spindle 9 DISC-NT+ configuration fault
Error 310	Axis/Spindle 10 DISC-NT+ configuration fault
Error 311	Axis/Spindle 11 DISC-NT+ configuration fault
Error 312	Axis/Spindle 12 DISC-NT+ configuration fault
Error 313	Axis/Spindle 13 DISC-NT+ configuration fault
Error 314	Axis/Spindle 14 DISC-NT+ configuration fault
Error 315	Axis/Spindle 15 DISC-NT+ configuration fault
Error 316	Axis/Spindle 16 DISC-NT+ configuration fault
Error 317	Axis/Spindle 17 DISC-NT+ configuration fault
Error 318	Axis/Spindle 18 DISC-NT+ configuration fault
Error 319	Axis/Spindle 19 DISC-NT+ configuration fault
Error 320	Axis/Spindle 20 DISC-NT+ configuration fault
Error 321	Axis/Spindle 21 DISC-NT+ configuration fault
Error 322	Axis/Spindle 22 DISC-NT+ configuration fault
Error 323	Axis/Spindle 23 DISC-NT+ configuration fault
Error 324	Axis/Spindle 24 DISC-NT+ configuration fault
Error 325	Axis/Spindle 25 DISC-NT+ configuration fault
Error 326	Axis/Spindle 26 DISC-NT+ configuration fault
Error 327	Axis/Spindle 27 DISC-NT+ configuration fault
Error 328	Axis/Spindle 28 DISC-NT+ configuration fault
Error 329	Axis/Spindle 29 DISC-NT+ configuration fault

Error 330	Axis/Spindle 30 DISC-NT+ configuration fault
Error 331	Axis/Spindle 31 DISC-NT+ configuration fault
Error 400	Loading dynamic operators in C: The size of user code is too big
Error 401	Loading dynamic operators in C: Format error
Error 402	Loading dynamic operators in C: Checksum error
Error 403	The system has insufficient memory for dynamic operators in C
Error 404	Loading dynamic operators in C: Open error
Error 405	Loading dynamic operators in C: Read error
Error 406	Loading dynamic operators in C: Close error
Error 407	Loading dynamic operators in C: The directory is empty
Error 410	Dynamic operators in C: Number of parameters passed does not match
Error 411	Dynamic operators in C: User error in INIT
Error 413	Unrecognised dynamic operators in C
Error 414	Dynamic operators in C without MAIN
Error 420	Dynamic operators in C: User error in the QUIT function
Error 421	Dynamic operators in C: User Error in the QUIT function
Error 423	Dynamic operators in C: Range of function in C not from [0..100]
Error 424	Internal system error
Error 500	Tilted Planed configuration: Invalid option FXSO000914
Error 501	Invalid T argument given the 5 axes kinematic configuration
Error 502	T arg.: positive index of setting table of RTCP function : T1 up to T8
Error 503	T arg.: index limited to 8 configuration
Error 504	Non authorized secondary axis
Error 510	RTCP configuration: Invalid option FXSO000154
Error 511	Syntax error in the current block G151Sn or G150Sn...
Error 512	Angular measurement by encoder is required by RTCP control
Error 513	Error in the call conditions of RTCP by G77statement...
Error 514	Programing error of RTCP or iPLANE function...
Error 515	Programing error of tool retraction
Error 517	Programing Error in the cycle248 of 5-axis kinematics calibrating
Error 518	Error occurred while execution of kinematic measurement cycle
Error 519	Sphere solving error
Error 530	Failure of part features measuring cycle or failure of the probe calibration cycle
Error 600	Curve number must be different of zero
Error 601	N... N... must be programed
Error 602	No axes programed in the first block
Error 603	Slope of the curve undefined
Error 604	Less than three blocks in the profile
Error 605	Unknown curve number
Error 700	Configuration error detected by VEComp G246 cycle
Error 701	Programing error detected by VEComp G246 cycle
Error 702	VEComp G246 cycle Error detected in the initialisation routine
Error 703	VEComp: Error detected in the RTS Opdyn task
Error 704	VEComp ROTARY G247 cycle Programing Error
Error 705	VEComp ROTARY G247 cycle Error at initialising process
Error 706	VEComp ROTARY G247 cycle Error in the RTS task
Error 750	G725 Programing Error
Error 751	G725 Width of the trochoidal path too small for the tool programed
Error 752	G725 Width of the trochoidal path larger than the diameter of at least one arc in the profile
Error 753	G725 Illegal value for EG2/EG3 argument for trochoidal machining
Error 754	G725 Illegal value for EE Finishing argument for trochoidal machining

Error 755	G725 Invalid option 106.
Error 760	G730 Programing Error:
Error 761	G730 Axial path less than engraving depth
Error 762	G730 Height of the letters less or equal to 0
Error 763	G730 Space between letters less or equal to 0
Error 764	G730 invalid option 107
Error 830	G730 Programing Error:
Error 831	Spindle not turning
Error 832	End point, P and K must be programed
Error 833	Retract value inadequate
Error 834	EB value: -90 < EB < +90
Error 835	No negative values for P, Q, R and K
Error 836	The interpolation plane must be G18 or G20
Error 837	Illegal Value for F or S
Error 838	Re-Thread: incoherent data
Error 839	Search In Block error
Error 840	G38+ NOT allowed with argument ER
Error 841	ER value: 0 < ER < +90
Error 842	Thread feed X axis exceeds speed limit
Error 843	End of thread feed Z axis exceeds speed limit
Error 844	Thread feed Z axis exceeds speed limit
Error 845	End of thread feed X axis exceeds speed limit
Error 846	Start of thread feed Z axis exceeds speed limit
Error 847	Start of thread feed X axis exceeds speed limit
Error 848	End of Thread movement argument R too long
Error 849	End of Thread movement argument ER too big
Error 850	Reduced retract value inadequate
Error 851	EK < 1 in G84
Error 852	ER Mandatory in MDI mode
Error 853	Cycle programed without tool axis pre-positioning
Error 862	End point, P or R must be programed
Error 863	End point incompatible with EA
Error 864	Milling tool prohibited in G66
Error 871	Finish profile limits undefined
Error 872	Dimension(s) missing in blank definition
Error 873	P or R not programed
Error 874	Blank incompatible with finish profile
Error 875	No intersection of EA with profile
Error 876	Wrong rake angle EB
Error 880	Cycle axis not known
Error 881	Parameter value not compatible
Error 882	Bottom of hole dimension missing
Error 883	Pitch (I J K) or retract (P) missing
Error 884	More than 9 thread starts
Error 885	Pocket not compatible with selected plane
Error 886	Tool incompatible with programed radius
Error 887	Cut > tool diameter
Error 888	Dwells prohibited in this cycle
Error 889	Syntax error
Error 890	Incompatible tool orientation
Error 891	Return plane = bottom of hole

Error 892	Axial feed missing
Error 893	Lateral feed missing
Error 894	ER is prohibited in G20
Error 895	G21,G22 is prohibited in cycle
Error 896	Dimension incompatible with tool radius
Error 897	Length of oblong pocket < diameter
Error 898	Tool corrector missing
Error 899	Spindle not assigned to this channel or spindle/channel incompatibility

Error 1	Programing error
Extension	**Meaning of the error**
1	Spline : Block number too high
2	Spline : Non alphabetic character
4	Spline : Only parameters admitted L0-L99
5	Spline : Missing '=' character after L parameter
6	Helix length too long
7	Helix length too long
8	Number of dynamic operator higher than 128 or keywords MOVE/R.OFF not expected
9	Missing : after an M62
10	Executable block in a part program called on RAZ procedure (%110000)
11	UNUSED
12	Unknown character < 'A' or > 'Z'
13	Unknown character
14	Wrong error extension programed
15	Block number to edit in G76 too large
16	Missing '=' char or keywords BCLR/BUILD/P.BUILD/PULL not expected
17	H parameter used outside a G function or keywords WHILE/ THEN/SEARCH/PUSH not expected
18	Operator message $./$.+ : incorrect syntax
19	G91 : Number of rotary axis revolutions exceeds 16383
20	Wrong or missing argument U/V/W/X/Y/Z
21	Wrong or missing argument U/W/X/Y/Z
22	ED: Angular offset option not activated
23	Programing of feed F with a negative or null value or keywords IF/FOR/ENDF not expected
24	S: Programing spindle speed over > 65535
25	Unknown character < 'A' or > 'Z'
26	Unknown character < 'A' or > 'Z'
27	EF: Programing value is negative
28	EH: Unknown function
29	EK: Unknown function
30	EN: Unknown function
31	EO: Unknown function
32	EU: Unknown function
33	EV: Unknown function
34	Missing '.' in programable error number E.nnn
35	EA.. 'EZ: Wrong symbolic addressing of L926-L950 variables
36	E.xxx: programable error number is null
37	Block number too high
38	Additional block jump value too high
39	G39+: S programed twice
40	G39+: K programed twice

41	G39+: Programed axis does not exist
42	G39+: Z programed twice
43	G39+: M3/M4 programed twice
44	G39+: Spindle already in rotation
45	G39+: Spindle speed range programed twice
46	Mirror function is active on the programed axis
47	G04 : Programing with a negative value
48	G61/G62: Unknown character after '@'
49	Block number to edit in G76 too large
50	G76+/- : No more arguments allowed
51	G76+ : insertion block character not allowed
52	G76+ : block exceeds maximum length
53	G76+ : End of string not found
54	G76+ N.. .... (block to insert) error on a list of polynomial coefficients
55	G76+: block to be inserted longer than 120 characters
56	G76+ H<bin> = ... Character '+' is missing after G76
57	G76+ H<bin> =<space allocated> ...Reservation size is null
58	G76+ H<bin> =<space allocated> … Reserved size must be the last data on the line
59	G76+ H<bin> = H.. Arguments are not alphabetic characters
60	G76+ H<bin> = H.. Arguments are not valid axes name
61	G76+ H<bin> = H.. Wrong or too much arguments
62	Tool offset programed in a part program called on Reset procedure (%11000)
63	G10 : Wrong physical address for an axis
64	G10 : Axis number too high
65	G92R : curve radius has a negative value
66	G92S: spindle speed too high
67	G96 : Spindle speed too high
68	G97 : Spindle speed too high
69	Maximum sequence number exceeded: 2 147 483 647
70	Slash character "/" for block skipping is not the first in the block or keyword CUT not expected
71	G77: repetitions number exceeds maximum value
72	Wrong operand in an expression
73	Unknown memory area in the syntax VAR H.. :s (0<s<4)
74	Unhandled character
75	G function not admitted in a part program called on Reset procedure (%11000)
76	Programing of a spindle speed S > 655354
77	M62 : Spindle address higher than 32
78	M62: Can't activate spindle. Is assigned to another channel
79	M66: Can't activate measure spindle.
80	M62: Can't activate spindle. Is assigned to another channel
81	M61: Wrong spindle

82	M62: Wrong spindle
83	M62: Wrong spindle
84	M66: Wrong spindle
85	M03: Wrong spindle
86	M04: Wrong spindle
87	M05: Wrong spindle
88	M19: Wrong spindle
89	M40: Wrong spindle
90	M41: Wrong spindle
91	M42: Wrong spindle
92	M43: Wrong spindle
93	M44: Wrong spindle
94	M45: Wrong spindle
95	M62: Can't assign spindle rank
96	M66: Can't activate measure spindle.
97	M66: Can't activate measure spindle.
98	M41-M45: Wrong spindle
99	M41-M45: Spindle is synchronized
100	M3 M4: Wrong spindle
101	M5: Wrong spindle
102	M19: Wrong spindle
103	M61: Wrong spindle / Spindle rotating
104	M61: Wrong spindle / Spindle rotating
105	"/" : Wrong argument of skip block code
106	"/": Argument value out of range
107	G92S: illegal spindle
108	Missing closing bracket
109	Unknown function
110	Function not handled in this context
111	Missing input string
112	SCPY: missing second input string
113	SCAT: missing second input string
114	SNDEL: missing number of characters to delete
115	SNDEL: missing index of first character to delete
116	SNINS: missing second input string
117	Binary number not valid
118	Hexadecimal number not valid
119	G164 : Number of rotary axis revolutions exceeds 16383
120	Missing character " for end of string
121	Spindle command in subroutine %11001
122	Wrong index

123	Invalid address. Must be (X,Y,Z,U,V,W,A,B,C)
124	SSUB: missing second input string
125	SSUB: missing number of characters
126	SSUB: missing index of first character
127	Syntax error: Missing '}'
128	G76: Unknown character
129	MAX/MIN syntax is not correct
130	Error declaring argument of FPF
131	ATAN2 / ATAN3 syntax is not correct
132	Extended function A: index error
133	Extended function: missing =
134	Encoded M functions: too many standard M functions programed in the same block
135	M62: Can't release spindle
136	Axes movements in subroutine 11001
137	Symbolic Variable table too big
138	G92L : length has a negative value

Error 2	G Function syntax error
Extension	**Meaning of the error**
1	G38 : Option non activated
2	G38+ : 'C' character after 'E' is missing
3	G38+ : J or EC argument is missing
4	G39+: One of the four mandatory arguments (M3/M4,S,K or Z) is missing
5	G39+: S is too large (>65535)
6	G39+: K is negative or zero
7	"G39+: An M function other than M3/M4 or M40-M45- ""M02"" is programed"
8	G39+: Unknown argument is programed
9	G39+: K is negative or zero
10	G39: Missing argument 'K'
11	G39: Work axis not programed
12	G39: Too many linear axes programed
13	G04 : 'F' argument is missing
14	G51: Missing arguments
15	G51: Missing arguments
16	G51: Missing arguments
17	G51: '+' is missing
18	G16 : Argument different of P, Q, R
19	G16 : The tool direction (+/-) is missing
20	Circular interpolation option is not activated
21	G24 : The sign + or - is missing
22	G26: '+' is missing

23	G59: Missing argument
24	G59: Missing argument
25	G59: Wrong argument
26	G59: Argument forbidden (P/Q/R)
27	G43: Required option 33
28	G29: Required options 30 or 33
29	UNUSED
30	G76: Required option 47
31	G76: Missing argument (H/N)
32	G76: Missing second 'N' argument
33	G76+ H<bin>: missing arguments
34	G77: Wrong argument
35	G79: Missing argument 'N'
36	G10 or G79: the value of argument X or F is negative
37	G10 or G79: argument different from X or F
39	G10 : Too many interrupts
40	G75: Required option 70
41	G75: Missing argument 'N'
42	G17/G18/G19: Required option 61
43	G20/G21/G22: Required option 62
44	G21: Required option 50 (Cartesian/polar)
45	G20/G21/G22: Turn functions prohibited
46	G74 : Required option 44
47	G92 F: Non zero value affected to axis is forbidden.
48	G92 F: Set of non-interpolated axes in the channel.
49	G92 F: Set of linear and rotary axes forbidden.
50	Unknown G function [G256 ..G900[
51	F' argument is missing after switch of G93/G94/G95
52	Unknown G function (G9xx)
53	Unknown G function (G99x)
54	BUILD: H/N argument is missing.
55	BUILD: Only one N argument is programed
56	Unknown G function (G5xx)
57	G76 S missing + or - after G76
58	G76+/G76- S no programing of first symbolic variable
59	G09 Missing argument
60	G770 Invalid axis
61	G770 Expected S function
62	G771 Invalid axis
63	G771 Invalid axis
64	G771 Expected P/Q function

65	G771 Expected P/Q function
66	G771 Too many leaders
67	G771 Synchronisation active.
68	G771 Synchronisation not defined
69	G771 Syntax error
70	G771 Syntax error
71	G770 No more arguments allowed
72	G771 Only cubic splines are allowed
73	G775 Axis is not leader or follower
74	G770 Syntax error
75	G776 Invalid axis
76	G779 Invalid axis
77	G779 Invalid axis
78	G779 Axis programed twice
79	G779 Leader not found
80	G779 illegal Follower
81	G779 Syntax error
82	G770 Illegal M code
83	G771 Follower active. On fly changes are not allowed
84	G771 table not found
85	G771 Wrong table format
86	G771 Too many points in table
87	G775 At least one leader required
88	G771 Synchronization mode not allowed
89	G775 illegal Follower
90	G775 - Axis without Origin
91	G775 Syntax error
92	G775 No more arguments expected
93	G776 No more arguments expected
94	G771 Periodic spline must start at 0
95	G771 Leader position in table must be ascending
96	G771 Leader position in table must be ascending
97	G770 Error. NM-AUTO not enabled
98	G771 Syntax error
99	G771 ED axis not defined
100	G771 ED axis not defined
101	G771 Syntax error
102	G771 Syntax error
103	G771 Syntax error
104	G771 Syntax error
105	G771 Syntax error

106	G771 Syntax error
107	G771 Syntax error
108	G771 Q=0 div by 0 hazard
109	G771 Recursive EGB link
110	G76 File in Zone 1, 2, 3
111	Function require option G81
112	Unknown G function
113	G771 Follower active. On fly changes are not allowed
114	EGB option not activated
115	G76+ [..] [..] S… missing after variables
116	G76- [..] [..] forbidden
117	G771 P and Q must be both integers
118	G39: Syntax error
119	G76= Syntax error
120	G76= Syntax error
121	G164 : Option not activated
122	G164: First N argument is missing
123	G164: Second N argument is missing
124	G164: Too many arguments N
125	G164: Unknown E.. argument
126	G164: Unknown argument
127	License of option FXSO000462 is required
128	EBC Q value out of range
129	EBC I/O/P/R value out of range
130	Invalid axis
131	Unexpected function in G748
132	Unexpected function in G748
133	Invalid feed
134	License of option FXSO000466 is required
135	License of option FXSO000467 is required
136	License of option FXSO000465 is required
137	G771 table already in use

Error 3	The attribute of a G code is wrongly positioned
Extension	**Meaning of the error**
1	Error on argument 'N'
2	Error on argument 'X'
3	Error on argument 'P'
4	Error on argument 'I'
5	Error on argument 'F'
6	Error on argument 'S'

7	Error on argument 'H'
8	Error on argument 'EP'

Error 4	Option not enabled or conflict between option and parameter
Extension	**Meaning of the error**
1	Option "Early Block Change" not activated
2	UNUSED
3	'Dynamic Operators' option required to access an E81xxx/E82xxx parameter
4	Trying to write parameter E11019 without probing option
5	Trying to write parameter E11012 without HSC option
6	Trying to write parameter E32005 without HSC option
7	Trying to write parameter E96xxx without Duplicated or synchronized axes option
8	Dynamic operator not allowed in MDI mode
9	Dynamic operator option is missing
10	Option 'Dynamic operators' missing
11	Option 'Dynamic operators in C' missing
12	G61 : 'Bspline' option missing
13	G39 : 'Rigid tapping' option missing
14	'Structured Programing' option missing
15	'Table building' option missing
16	G46 : Pocket option missing
17	Trying to write parameter E11028 without license of option FXSO000520

Error 5	Geometric Programing option not enabled
Extension	**Meaning of the error**
1	Geometric programing option not activated (EB+/EB-)
2	Geometric programing option not activated (EB+/EB-)

Error 6	Error in Spline function programing
Extension	**Meaning of the error**
1	B-Spline polynomial degree too high
2	Polynomial interpolation option missing
3	Spline coefficient table full
4	Spline coefficient table full
5	G62: Spline wrong pole definition
6	Polynomial interpolation: null segment definition
7	G164: Polynomial interpolation option missing
8	G164: Spline coefficient table full
9	G164: Spline coefficient table full

Error 7	Movements parallel to the inclined axes
Extension	**Meaning of the error**
1	G05 Programing is not in plane G20

2	G05 Interpolation is not in G00 or G01
3	X is not programed after G05
4	G07 Programing is not in plane G20
5	G07 Interpolation is not in G00 or G01
6	X and Z are not programed after G07

Error 8	Tool corrector number too high
Extension	**Meaning of the error**
1	Max 32 tool correction admitted without 255 offsets option
2	Max 255 tool correction admitted
3	Tool Manager D correction too high
4	Tool Manager - Loaded Tool = 0

Error 9	Too many consecutive non executable blocks
Extension	**Meaning of the error**
1	Consecutive non executable blocks >= 20000

Error 10	iNCK protocol error
Extension	**Meaning of the error**
1	iNCK protocol error: bActive fail on slave
2	iNCK protocol error: uSafeCnt fail on master
3	iNCK protocol error: uSafeCnt fail on master
4	iNCK protocol error: bActive fail on slave
5	iNCK protocol error: uSafeCnt fail on slave
6	iNCK protocol error: uSafeCnt fail on slave

Error 11	Shared drives programing error
Extension	**Meaning of the error**
1	Drive not declared as remote
2	Drive already remote
3	Invalid address
4	Drive already remote
5	Drive not remote on this channel
6	Drive not remote on this channel
8	Invalid address
9	Drive not declared as remote
10	Invalid address
11	Drive not declared as remote
12	Axis not declared as remote
13	Axis not currently remote

Error 12	Time out coupling/discarding remote drive
Extension	**Meaning of the error**
1	Time out coupling shared drive

2	Time out waiting axis in standstill
3	Time out discarding drive
4	Time out waiting spindle in standstill

Error 13	Error on EGB axes
Extension	**Meaning of the error**
1	Time out synchronizing EGB axis
2	Synchro Lost
3	Zero Reference Lost
4	Table Error
5	Table Error
6	Table Error
7	EGB follower out of software limits
8	EGB or rotating axis. Can't release in channel

Error 14	Inclined plane option missing
Extension	**Meaning of the error**
1	Inclined plane option not activated

Error 15	Rigid tapping and retract
Extension	**Meaning of the error**
1	Tapping distance too short
2	Distance at constant deceleration too short
3	Spindle homing not done
4	Distance at constant acceleration too short

Error 16	Error in activating RTCP
Extension	**Meaning of the error**
1	License of option FXSO000154 is required
2	G26: G26 Syntax error +/- missing
3	G26 is not allowed if G24 is active
4	G26: Tool vector components I, J, K incomplete
5	G26: I or J or K programed 2 times
6	G26: Tool vector components I, J, K incomplete
7	G26: Number of pivot connections arguments + inclined plane > 16
8	G26: No Additional statements twist after TD
9	G26: One inclined plane permitted after PD
10	G26: No existing physical axis
11	G26: PA B or C prohibits behind 'PD'
12	G26: Missing character '/'
13	G26: Missing character '/'
14	G26: Syntax error in the declaration of an inclined plane PI / PJ / PK
15	G26: Missing character '/'

16	G26: Tool vector I, J, K missing with a head twist
17	G26: Non-compliance between the pivot axes and the physical axes
18	G26: cos / sin unscheduled statement after PD
19	G26: Twist axis number inconsistent
20	G26: If the number of twist = 0, IJK must not be programed
21	G26: The first post of turntable must be a node
22	G26: Tool change programed with G26 active
23	Main axis not active with RTCP
24	G26: Main axis not active
25	G26: missing angular shift on rotary axis

Error 17	Missing closing bracket
Extension	**Meaning of the error**
1	End of block in a comment
2	G76: End of block in a comment

Error 18	Servo error: P50 too small
Extension	**Meaning of the error**
1	Timeout of task Axis

Error 19	Error getting address
Extension	**Meaning of the error**
1	After @ missing argument […], H or S
2	Program number inconsistent
3	Structure number inconsistent

Error 20	M02 missing or no current program active
Extension	**Meaning of the error**
1	Cycle start after reset not authorized in drip feed mode
2	B-Spline : Empty block
3	B-Spline : End of program encountered
4	Empty block
5	M02 missing or no current program active
6	Missing block
7	Current Part Program corrupted
8	B-Spline: Current Part Program corrupted
9	G76: Current Part Program corrupted
10	G76+: Current Part Program corrupted
11	VAR: Current Part Program corrupted
12	VAR: Current Part Program corrupted
13	VAR N..N..: Current Part Program corrupted

Error 21	Error in single axis positioning
Extension	**Meaning of the error**
1	Wrong positioning type
2	Wrong acceleration
3	Wrong velocity
4	Wrong window tolerance
5	Wrong positioning type
6	Wrong acceleration
7	Wrong velocity
8	Wrong window tolerance

Error 22	Prohibited functions in pole definition blocks
Extension	**Meaning of the error**
1	Argument N without D
2	Polynomial degree ND.. below 1
3	Polynomial degree ND.. above 5
4	Argument not admitted (char not in 'A' .. 'Z')
5	Argument not admitted (char = {D,E,F,G,H,L,M,N,O,S,T})
6	Argument not admitted (char = {I,J,K,P,Q,R})
7	Argument 'H' is not programed
8	No axis programed
9	More than six axes programed
10	Argument not admitted (char not in 'A' .. 'Z')
11	Only function admitted : G60
12	Argument not admitted (only 'G' and 'L' admitted)
13	G164: No axis programed in the first block

Error 23	Configuration error
Extension	**Meaning of the error**
1	Parameter V310 not coherent with P77
2	Parameter V324 bit 2 not coherent with P77

Error 24	Error in the definition of an inclined plane
Extension	**Meaning of the error**
1	Homing request when RTCP active
2	Homing request for an axis of inclined plane
3	Disabling G24 with RTCP active
4	Activation of the function when it is already enabled
5	Reference system origin missing
6	Axis origin programed twice
7	Parallel axis yet programed
8	Not all axes programed
9	Wrong declaration of the conversion matrix coefficients

10	Conversion matrix: missing coefficients
11	Value of conversion matrix coefficients out of range
12	Rotary axis programed twice
13	For rotary axes only T or F admitted
14	Not all rotary axes programed
15	Determinant of conversion matrix is null
16	Value of inverse conversion matrix coefficients out of range
18	The axis of the pivot point is not servo-controlled
19	The axis of the pivot point does not ex
20	G52 prohibited when inclined plane is active

Error 25	Unknown sub-program or block number
Extension	**Meaning of the error**
1	Program not found
2	G76+ : Cannot overwrite existing program
3	G76 : The program does not exist
4	G76+ : Source program not found
5	Unknown sub-program
6	Unknown block number
7	Block number not found
8	Block number not found
9	Program not found
10	Block number not found
11	Program not found

Error 26	Sub-program nesting too deep
Extension	**Meaning of the error**
1	Too many subroutine nesting levels (max 16)
2	The blocks called by G77 -i must not include branches or other calls
3	G77 -i : Nesting level requested is not accessible
4	Too many subroutine nesting levels (max 16)
5	Wrong context of subroutine
6	The subroutine invoked does not exist
7	The blocks called by G77 -i must not include branches or other calls

Error 27	Tool Radius Correction
Extension	**Meaning of the error**
1	G52 must be programed in G40 mode

Error 28	Syntax error in CSS or working radius definition
Extension	**Meaning of the error**
1	G96 must be followed by S
2	G96 : X or U not programed

3	G97 must be followed by S
4	G92 not programed
5	G92 speed programed over the maximum of the range
6	G92 speed programed over the programable limit

Error 29	Error in spindle speed programing
Extension	**Meaning of the error**
1	No range programed for CSS
2	Maximum speed of the range max = 0
3	Spindle speed higher than the maximum of the range
4	No range search option
5	S does not belong to any range
6	M19: Gear not inserted
7	Attempting spindle start without S
8	Not existing spindle
9	Not existing spindle
10	Not existing spindle
11	Not existing spindle
12	Programed G95 without active spindle

Error 32	Axis referencing error
Extension	**Meaning of the error**
1	Axis already on the home switch and P15N1 = 0

Error 33	All channels waiting for synchronization
Extension	**Meaning of the error**
1	Deadlock between the NC channels
2	Deadlock between the NC channels in SEARCH mode

Error 34	Minimum radius for interpolation reached - G21
Extension	**Meaning of the error**
1	Radius null
2	Tangential speed < F_MIN

Error 35	Sequence number not found in SEARCH
Extension	**Meaning of the error**
1	Block Number Search error : Block number not found
2	Block before the one to re-engage not complete
3	Fast Search mode: block not found
4	Re-engagement in transient state G40-G41/G42

Error 36	Two reduced speed programed
Extension	**Meaning of the error**
1	For the same axis in JOG mode
2	For the same axis in programing mode

3	For the same handwheel
4	For the same spindle
5	For the same axis in programing mode

Error 37	Maximum feed exceeded in thread cutting
Extension	**Meaning of the error**
1	Over-speed in thread cutting
2	Over-speed in thread cutting
3	Over-speed in thread cutting

Error 39	Axis synchronization fault
Extension	**Meaning of the error**
1	Synchronization error higher than the maximum (P25 N32)
2	Synchronization error higher than the axis maximum (P25 N0..31)
3	Synchronization error higher than the overall maximum (P25 N32)
4	Synchronization error higher than the axis maximum (P25 N0..31)
5	Synchronization error higher than the overall maximum (P25 N32)
6	Dangerous synchronization error at the system start up

Errors 40 ÷ 71	Excessive lag value on axis xx
Errors xx	**Description**
40	Excessive lag value on axis 0
41	Excessive lag value on axis 1
42	Excessive lag value on axis 2
43	Excessive lag value on axis 3
44	Excessive lag value on axis 4
45	Excessive lag value on axis 5
46	Excessive lag value on axis 6
47	Excessive lag value on axis 7
48	Excessive lag value on axis 8
49	Excessive lag value on axis 9
50	Excessive lag value on axis 10
51	Excessive lag value on axis 11
52	Excessive lag value on axis 12
53	Excessive lag value on axis 13
54	Excessive lag value on axis 14
55	Excessive lag value on axis 15
56	Excessive lag value on axis 16
57	Excessive lag value on axis 17
58	Excessive lag value on axis 18
59	Excessive lag value on axis 19
60	Excessive lag value on axis 20
61	Excessive lag value on axis 21

62	Excessive lag value on axis 22
63	Excessive lag value on axis 23
64	Excessive lag value on axis 24
65	Excessive lag value on axis 25
66	Excessive lag value on axis 26
67	Excessive lag value on axis 27
68	Excessive lag value on axis 28
69	Excessive lag value on axis 29
70	Excessive lag value on axis 30
71	Excessive lag value on axis 31 **(See 71 extension for all errors here above)**

Error 71	Excessive lag value on axis 31
Extension	**Meaning of the error**
1	Lag error on an axis in failure
2	NC error machine (dynamic lag control P57)
3	NC error machine (overcoming maximum following error P23)

Error 72	Incremental programing after an incomplete block
Extension	**Meaning of the error**
1	G91: after an incomplete block
2	G91: First absolute block required
3	G91: First absolute block required
4	G91: First absolute block required
5	G91: First absolute block required
6	G91: First absolute block required
7	G91: First absolute block required
8	G91: First absolute block required
9	G91: First absolute block required
10	G91: First absolute block required
11	G91: First absolute block required
12	G91: First absolute block required

Error 73	G38 : Missing argument
Extension	**Meaning of the error**
1	G38 : Missing K argument

Error 74	Errors in G46 Pocket
Extension	**Meaning of the error**
1	G46: Format error
2	G46: Block not found in search mode
3	G46: Working memory busy
4	G46: Program number to large
5	G46: NU number not included in those authorized

6	G46: TEST mode required after modification or first use
7	G46: No dimension programed in the contouring plane
8	G46: First positioning block missing
9	G46: Insufficient memory size
10	G46: Character not authorized
11	G46: Address missing or unknown address
12	G46: An incomplete contour block
13	G46: Tool direction not perpendicular to the contouring plan
14	G46: Actual tool not compatible with technical data of the pocket
15	G46: Changing the contouring plane between definition and machining
16	G46: 2 nested definitions
17	G46: NU0 programed with G59
18	G46: Pocket depth is null
19	G46: start point or end point coordinates are incomplete
20	G46: Spindle rotation is compatible with definition
21	G46: G function not authorized in G46
22	G46: First contour block incomplete
23	G46: Pocket definition missing
24	G46: Internal error of the algorithm
25	G46: start point or end point coordinates are incomplete
26	G46: Insufficient memory size
27	G46: Insufficient memory size
28	G46: Insufficient memory size
29	G46: Insufficient memory size
30	G46: Insufficient memory size
31	G46: Insufficient memory size
32	G46: Insufficient memory size
33	G46: Insufficient memory size
34	G46: Insufficient memory size
35	G46: Insufficient memory size
36	G46: Insufficient memory size
37	G46: Insufficient memory size
38	G46: Insufficient memory size
39	G46: Insufficient memory size
40	G46: Insufficient memory size
41	G46: Insufficient memory size
42	G46: Insufficient memory size
43	G46: Working memory busy
44	G46: NU number not included in those authorized
45	G46: Character not authorized
46	G46: Character not authorized
47	G46: Character not authorized

48	G46: Character not authorized
49	G46: Character not authorized
50	G46: Character not authorized
51	G46: Tool direction not perpendicular to the contouring plan
52	G46: Actual tool not compatible with technical data of the pocket
53	G46: Changing the contouring plane between definition and machining
54	G46: Changing the contouring plane between definition and machining
55	G46: Changing the contouring plane between definition and machining
56	G46: Changing the contouring plane between definition and machining
57	G46: Changing the contouring plane between definition and machining
58	G46: Changing the contouring plane between definition and machining
59	G46: Pocket depth is null
60	G46: Pocket depth is null

Error 75	G20 to G21 or G22 transition
Extension	**Meaning of the error**
1	Last block incomplete
2	The starting radius is either negative or zero

Error 77	Tool type not compatible with the operation
Extension	**Meaning of the error**
1	Turning or boring tool outside G20/G21 or G22 status
2	Milling tool with turning plane G20, G21 or G22

Error 78	Syntax error in channel synchronization
Extension	**Meaning of the error**
1	G78 without parameters 'P' and 'Q'
2	G78 Q.. New value lower than previous
3	G78 P .. Character '.' missing
4	G78 P.. Index of channel must be positive
5	No M00, M01 or M02 with G78 P..

Error 79	Invalid TANDEM Configuration
Extension	**Meaning of the error**
1	Parameter P85 not compatible with V160

Error 80	Drive not in HALT
Extension	**Meaning of the error**
1	Drive not in HALT when trying to write E942xx

Error 81	Wrong spindle configuration
Extension	**Meaning of the error**
1	Invalid spindle configuration (sensor-less spindle : Not used at the moment)
2	Invalid spindle gear configuration (Speed limit > Speed Max)
3	2 identical logical address for spindle

Error 82	First or last bloc of profile definition incomplete
Extension	**Meaning of the error**
1	First bloc incomplete
2	Last bloc incomplete

Error 83	M Timed and Inter-sample
Extension	**Meaning of the error**
1	M timed: already waiting execution
2	M timed: already waiting execution
3	M intersample: time out of range
4	M timed: time out of range
5	M timed-intersample: Wrong E parameter
6	M timed-intersample: Wrong E parameter
7	M intersample: Final M function already programed in this block
8	M intersample: Initial M function already programed in this block
9	M timed: Final M function already programed in this block
10	M timed: Initial M function already programed in this block

Error 84	Homing not possible
Extension	**Meaning of the error**
1	Homing forced with inclined plane active
2	Homing forced on a slave axis
3	Homing forced on a master coupled axis
4	Homing forced on a missing axis
5	Character not admitted
6	Homing forced with RTCP active

Error 91	Unknown parameter or variable
Extension	**Meaning of the error**
1	Trying to read E79007 for a spindle with number negative or higher than 32
2	Trying to read E79007 for an unknown spindle
3	Trying to write E79007/8 for a spindle with unknown reference
4	Trying to write E79007/8 for a spindle with number negative or higher than 32
5	Trying to read Lxx parameter with xx higher than 20 and xx less than 100
6	Trying to read an unknown L parameter
7	Trying to read a E6x007parameter with x less than 6 (only axes A,B,C allowed)
8	Trying to write an unknown E7xxxx parameter
9	Trying to read a wrong E320xx parameter
10	Trying to read an unknown E parameter (parameter less than 10000) or keywords SAVE/ELSE/DELETE not expected
11	Trying to read an unknown E parameter
12	Trying to read an unknown E parameter
13	Trying to read E79008 for a spindle with number negative or higher than 32

14	Trying to read E200xx with xx higher than 31
15	Trying to read E20xxx with xxx higher than 101
16	Trying to read an unknown E110xx parameter
17	Trying to write E79008 for an unknown spindle
18	Trying to read a wrong E41xxx parameter
19	Trying to read an E342xx parameter with xx higher or equal than 2
20	Trying to read an E35xxx parameter with xx higher or equal than 96
21	Trying to write an E351xx parameter with xx higher or equal than 96
22	Trying to read an E93Yxx parameter with xx higher or equal than 32
23	Trying to read an E935xx parameter with xx higher or equal than 64
24	Trying to read an E938xx parameter with xx higher or equal than 4
25	Trying to read an E5Yxxx parameter for a not existing tool (xxx = 0 or xxx higher than 255)
26	Trying to read an unknown E120xx parameter
27	Trying to read an E6xnnY parameter with Y higher than 1
28	Trying to read/write a wrong E69xxx parameter
29	Trying to write an unknown E110xx parameter
30	Trying to read a wrong E7xxxx parameter
31	Error reading E79001parameter
32	Trying to read an E351xx parameter with xx higher or equal than 64
33	Trying to read an E81xxx/E82xxx parameter with xxx higher than table dimension
34	Trying to write an unknown E130xx parameter
35	Trying to read an E92Yxx parameter with xx higher or equal than 32
36	Trying to read an E924xx parameter with xx higher or equal than 4
37	Trying to read an E930xx parameter with axis/spindle not declared
38	Trying to read an unknown E130xx parameter
39	Trying to write an unknown E120xx parameter
40	Trying to read an unknown E12xxx parameter
41	Trying to write E12000 a read only parameter
42	Trying to read an E91432/64 parameter for a not declared spindle
43	Trying to read an E91132/64 parameter for a not declared spindle
44	Trying to write an E91132/64 parameter for not declared spindle
45	Trying to read/write an E96Yxx parameter with xx higher or equal than 32
46	Trying to read/write an E97Yxx parameter with xx higher or equal than 32
47	Trying to read wrong an E97xxx parameter
48	Trying to read 980xx with xx higher than 32
49	Trying to read 981xx with xx higher than 32
50	Trying to read 982xx with xx higher than 32
51	Trying to read 983xx with xx higher than 32
52	Trying to read 984xx with xx higher than 32
53	Trying to read 985xx with xx higher than 64
54	Trying to read 986xx with xx higher than 96

55	Writing E97000, E97100 et E97200 not allowed
56	Trying to read E79000 for an unknown spindle
57	Trying to read E90Yxx for a not declared spindle
58	Trying to read an E90xxx parameter
59	Trying to read/write an E907xx for a not declared spindle
60	Trying to read a wrong E90xxx parameter
61	Error writing a E90xxx parameter
62	Trying to read/write an E909xx for a not declared spindle
63	Trying to read/write E905xx for a not declared spindle
64	Trying to write E910xx for a not declared spindle
65	E7x101 : Programing a non-CNC channel
66	E7x101 : Programing a non-existent axis
67	Trying to write E912xx without the 'N/M auto' option
68	Trying to read a wrong E5xxxx parameter
69	Unknown E parameter
71	Trying to write an E90Yxx for a not declared spindle
72	Trying to read a wrong E6x00y parameter (wrong origin)
73	Trying to read/write an E6xNN0 parameter with NN higher than 5
74	Trying to read/write an E6xNN1 parameter with a wrong origin number
75	Trying to read an unknown E790xx parameter
76	Trying to write an E6xNN0/1 with G53 active
77	Trying to read an E351xx parameter with xx higher or equal than 64
78	Trying to read an E351xx parameter with xx higher or equal than 64
79	E6x012 limited to A, B, C program axes

Error 92 Error in assignment to a parameter or variable

Extension	Meaning of the error
1	Trying to write E79005/6 with minimal feed override negative
2	Trying to write E79005/6 with maximal feed override less than 100
3	Trying to write E79005/6 with maximal feed override less than minimal or minimal higher than 100
4	Trying to write E79005/6 with maximal feed override higher than 255
5	Trying to write E79007/8 with minimal feed override negative
6	Trying to write E79007/8 with maximal feed override less than 100
7	Trying to write E79007/8 with maximal feed override less than minimal or minimal higher than 100
8	Trying to write E79007/8 with maximal feed override higher than 255
9	Trying to write a wrong value in E352xx
10	Trying to write E941xx : Request to set as master a non-declared spindle
11	Trying to write E940xx for an axis transformed by inclined plane or RTCP
12	Trying to write a wrong value in E353xx
13	Trying to write E910xx of an axis transformed by inclined plane or RTCP
14	Parameter E70005 : Trying to write a value less than-1 or higher than32
15	Trying to write E941xx : New value is out of range

16	Trying to write E41105 : New value is out of range (0 .. 8)
17	Trying to write E920xx of slave axis
18	Trying to write in E932xx a value different of 0 or 1
19	Trying to write E932xx for an axis not belonging to the channel
20	Trying to write E932xx for a non-rotating or not-measured axis
21	Trying to write E932xx of a servo axis
22	Trying to write E932xx of an axis transformed by inclined plane or RTCP
23	Trying to write E940xx with a value less than-1 or higher or equal than 32
24	Trying to write E940xx : Trying to write a new master while old master still active
25	Trying to write E940xx : Trying to write a new master for an axis not in the current channel
26	Trying to write E940xx : Trying to write a new master in another channel
27	Trying to write E941xx : New master axis is the slave axis
28	Trying to write E941xx : Changing master on an already slaved axis
29	Trying to write E941xx : Request to set as slave a non-declared spindle
30	Trying to write E941xx : Slave axis is synchronized or duplicated
31	E942xx address drive or coupler outside of the range allowed
32	E942xx address application outside of the range allowed
33	E942xx in option 127 : address axis application NOT existing
34	E942xx in option 127 : address spindle application NOT existing
35	E942xx switching to analogue coupler : address axis application analogue NOT existing
36	E942xx switching to analogue coupler : address spindle application analogue NOT existing
37	E943xx address handwheel coupler with DAC NOT existing
38	E943xx switching application to handwheel coupler : address axis application analogue NOT existing
39	Trying to write parameter E96xxx for an axis without a master defined
40	Trying to write parameter E96xxx for an axis synchronized by P28
41	Trying to write parameter E962xx of an already duplicated axis
42	Trying to write parameter E96xxx of an already synchronized axis
43	Negative parameter
44	Parameter value higher than the maximum value of the function
45	Programed value of E7x101 is negative or above 100
46	Trying to write E941xx for spindle : The new value is out of range
47	Trying to write E941xx : The new master spindle is the slave spindle
48	E942xx switching OFF an application from a drive: address drive NOT existing
49	E942xx switching ON an application to a drive: address drive NOT existing
50	E942xx switching ON an application to a drive: address drive and address application inconsistent
51	E942xx switching ON an application to a drive: address axis application NOT existing
52	E942xx switching ON an application to a drive: address spindle application NOT existing
53	E942xx switching applications to drive : address drive NOT existing
54	E942xx switching applications to drive : address drive and address application inconsistent
55	E943xx address handwheel coupler outside of the range allowed
56	E943xx address application outside of the range allowed

57	E943xx switching application to analogue coupler: address spindle application analogue NOT existing
58	Trying to write out of range value of L../E... with G76 function
59	Trying to write out of range value of L../E... with G76 function
60	Forbidden operation: Torque/Reference request active
61	Illegal value in E11024
62	Trying to write E11025 : Value is out of limits
63	Trying to write E352xx : Value is out of limits
64	Trying to write E90932: Value is out of limits
65	Trying to write E905xx: Value is out of limits
66	Trying to write E909xx: Value is out of limits
67	Trying to write E964xx when E962xx is active
68	Unknown E parameter
69	Unknown E parameter
70	Unknown E parameter
71	Unknown E parameter
72	Unknown E parameter
73	Unknown E parameter
74	Unknown E parameter
75	Unknown E parameter
76	Unknown E parameter
77	Unknown E parameter
78	Unknown E parameter
79	Trying to write out of range value of L../E... with G76 function
80	Trying to write out of range value of L../E... with G76 function
81	Trying to write out of range value of L../E... with G76 function
82	Trying to write out of range value of L../E... with G76 function

Error 93	Dynamic operators (Error in the definition of a parameter)
Extension	**Meaning of the error**
7	Dynamic operator 11 : missing number of bytes to be transmitted
8	Missing '=' char after the definition of a new operator
9	Unknown dynamic operator function code
10	Dynamic operator 11 : Storage at a not declared axis address
11	Dynamic operator 11 : Storage at a not-declared handwheel address
12	Dynamic operator 11 : Not type 3 parameter
13	Dynamic operator using an E parameter not of type 1
14	Dynamic operator : Trying to write in the correction of an inclined plane or RTCP involved axis
15	Deletion of a dynamic operator which wrote the correction of an axis involved in inclined plane or RTCP
16	Dynamic operator 22 : Graphic channel : E parameter not admitted
17	Dynamic operator 22 : E parameter not admitted
18	Dynamic operator 5 : Missing '/' char before second E parameter
19	Dynamic operator 12 : Missing '/' char before the modulo I

20	Missing first E parameter (Ea)
21	Dynamic operator 22 : L variable not admitted
22	Dynamic operator 6/22 : Axis address not declared
23	Dynamic operator 6 : Handwheel address not declared
24	Dynamic operator 6 : Not type 3 parameter
25	Dynamic operator 6 : K programed with wrong value
26	Dynamic operator 9: wrong value for Immediate parameter
27	Dynamic operator 10: missing E parameter after char '&'
28	Dynamic operator 9: missing E parameter after char '&'
29	Dynamic operator 25: missing argument address
30	Dynamic operator 5 : Second parameter is not an E parameter or a '$' value
31	Missing '/' char after second E parameter
32	Missing third parameter (Ec)
33	Dynamic operator try to use an E parameter not of type 2
34	Dynamic operator 4/14 : Immediate value of modulo negative or nil
35	Dynamic operator 16 : Missing '/' char before the L variable
36	Dynamic operator 6 : Missing '/' char before the second E parameter
37	Dynamic operator 13 : Missing '/' char before the second E parameter
38	Wrong axis address (negative or higher than 32)
39	Dynamic operator 25: no symbol argument
40	Dynamic Operators : @ handwheel negative or higher than 3
41	Dynamic operator 25: missing symbol identifier
42	Dynamic operator 25: symbol length null
43	Dynamic operator 25: no index argument
44	Dynamic operator 25: missing index identifier
45	Dynamic operator 25: wrong symbol
46	Dynamic operator 7 or 8 : Maximum index value negative
47	Dynamic operator 11 : Programed immediate value when storage at axis address
48	Dynamic operator 11 : Number of bytes to be transmitted negative or nil
49	Dynamic operator 11 : Number of bytes to be transmitted different of 2 with parameters E3410x, E3600x
50	Dynamic operator 11 : Number of bytes to be transmitted different of 1 with parameters E34000, E33000
51	Dynamic operator 11 : Number of bytes to be transmitted + index of E37xxx higher than 128
52	Dynamic operator 11 : Number of bytes to be transmitted different of 1, 2 or 4 with parameters E37000
53	Dynamic operator 11 : Number of bits to be transmitted + index of E100xx higher than 32
54	Dynamic operator 12 : Negative or nil modulo
55	Dynamic operator 17: @ char missing
56	Dynamic operator 13 : Missing '/' char before the L variable
57	Dynamic operator 13 : Missing L variable
58	Dynamic operator 13 : Index of L variable is negative or higher than 99
59	Dynamic operator 25: index too large
60	Dynamic operator 16 : Missing L variable

61	Dynamic operator 16 : Index of L variable is negative or higher than 96
63	Dynamic operator 16 : Last '/' not followed by '*'
64	Dynamic operator 10 : Missing '/' char after Ec parameter
65	Dynamic operator 10 : Automatic parameter less than E37000 or higher than E37127
66	Dynamic operator 11 : First parameter is not axis ('@'), nor a handwheel ('W'), neither an E parameter
67	Dynamic operator 15 : @ char missing
68	Dynamic operator 15 : Missing '/' char after the axis address
69	Dynamic operator 15 : Missing E47xxx parameter
70	Dynamic operator 15 : Parameter used different from E47xxx
71	Dynamic operator 9 : Missing '/' char after Eb parameter
72	Dynamic operator 17 : Missing '/' char before second E parameter
74	Dynamic operator 15 : Immediate value of speed negative or nil
75	Dynamic operator 20 : Missing char '@' for slave address
76	Dynamic operator 20 : Missing char '/' after slave address
77	Dynamic operator 20 : Missing char '@' for master address
78	Dynamic operator 20 : Missing char '/' after master address
79	Dynamic operator 20 : Wrong E200xx parameter
80	Dynamic operator 20 : Missing '/' char after E200xx parameter
81	Dynamic operator 20 : Wrong E37xxx parameter
82	Dynamic operator 20 : Missing '/' char after E37xxx parameter
83	Dynamic operator 21 : 4th argument is not an E parameter
84	Dynamic operator : All auxiliary tables are used
85	Dynamic operator 21 : Wrong E100xx parameter
86	Dynamic operator 21 : Missing '=' char after axis address
87	Dynamic operator 21 : Request time base modulation with wrong E47xxx parameter
88	Dynamic operator 21 : Wrong axes definition: Not @ char nor an E parameter
89	Dynamic operator 22 : Graphic channel : Missing function name
90	Dynamic operator 22 : Graphic channel : Wrong parameter type
91	Dynamic operator 22 : Missing function name
92	Dynamic operator 15 : Missing '/' char after E47xxx parameter
93	Dynamic operator 22 : Handwheel is not declared
94	Dynamic operator 22 : Wrong parameter type
95	Dynamic operator 22 : More than 16 parameters
96	Dynamic operator 24 : More than 9 axis to be controlled
97	Dynamic operator 24 : Missing '/' char after axis address
98	Dynamic operator 24 : No axis programed
99	Dynamic operator 24 : No handwheel programed
100	Dynamic operator 24 : Programed handwheel not declared
101	Dynamic operator 24 : Handwheel multiplication coefficient for rotary axes is zero
102	Dynamic operator 24 : Handwheel division coefficient for rotary axes is zero
103	Dynamic operator 24 : Handwheel multiplication coefficient is zero

104	Dynamic operator 24 : Handwheel division coefficient is zero
105	Dynamic operator 24 : Wrong char for a slave axis
106	UNUSED
107	Dynamic operator 23 : Missing '@' char for the first axis
108	Dynamic operator 23 : More than 32 dimensions programed
109	Dynamic operator 23 : Missing E37xxx parameter
110	Dynamic operator 23 : Only E37xx parameters admitted
111	Dynamic operator 23 : Missing '=' char after axis number
112	Dynamic operator 23 : The number of dimensions required for each axis is 2 at minimum
113	Dynamic operator 23 : Unknown char
114	Dynamic operator 23: The number of physical axes is limited to 3.
115	UNUSED
116	Dynamic operator 15 : Missing 'E' char for travel range

Error 94	Forbidden operation
Extension	**Meaning of the error**
1	Square root calculation is not possible
2	Division not possible
3	Trying to write E56xxx : Value is out of limits
4	Trying to write E580xx by Part Program or MDI
5	Trying to write a wrong value in E3100x
6	Trying to write a wrong value in E31002
7	Trying to write E32002 with a value higher than 65535
8	Trying to write E32003 : Value is out of limits (0 ... 360)
9	Trying to write E32004 : Value is negative
10	Trying to write E32005 : Value is out of limits (0 ... 13)
11	Trying to write E32011 : Value is negative
12	Trying to write E3410x : Value is negative
13	Trying to write E206xxx : Value is out of limits
14	Trying to write E580xx : Index is out of limits (1 ... 32)
15	Trying to write E205xxxx : Value is out of limits (0 ... 8)
16	Trying to write E55xxx : Value is out of limits (0 ... 8)
17	Trying to write E57xxx : Value is out of limits (0 ... 2)
18	Trying to write E207xxxx : Value is out of limits (0 ... 2)
19	Trying to write E69000 : Value is out of limits (1 ... 9999)
20	Trying to write E69001 : Value is out of limits (-80 ... 80)
21	Trying to write E69002 : Value is out of limits (-80 ... 80)
22	Trying to write E90700 : Value is out of limits (0 ... 99999999)
23	Unknown function
24	Undefined operand
25	Writing E69003 with a wrong value
26	Writing E69003 with a wrong value

27	Writing E69003 with a wrong value
28	Writing E69003 : Axes different from X Y Z
29	Writing E91032 : Option not active
30	Writing E91032 without indexation
31	Writing E91032 for a not-homed spindle
32	Writing E941xx for a not-homed master spindle
33	Writing E941xx for a not-homed slave spindle
34	Writing E985xx : Value is out of limits (0 ... 256)
35	Writing E985xx : With xx higher than 32
36	Parameters E90400-E90900 : Written value is negative or higher than 65535
37	Trying to write E59000 : Value is out of limits (1 ... 5)
38	Trying to write E211xxxx : Value is out of limits (0 ... 3)
39	Function LOG with negative or null input value
40	Function LN with negative or null input value
41	Modulo calculation is not possible
42	Trying to write E32001 : Value is out of limits (0 ... 65535)
43	Missing input parameter
44	SCMP: Missing second input parameter
45	Unhandled function
46	SISTR: Missing second input parameter
47	SNDEL: value of index or number of char is negative
48	SNDEL: value of index greater than string length
49	SNINS: value of index negative or greater than string length
50	SCPY: size of destination parameter is insufficient
51	SCAT: size of destination parameter is insufficient
52	SNINS: size of destination parameter is insufficient
53	GETSTR: size of destination parameter is insufficient
54	Programed movement of axis controlled by Op. Dyn 15
55	SSUB: value of index negative or greater than string length
56	SSUB: size of destination string is insufficient
57	SLEN: Missing input parameter
58	ATAN2 / ATAN3 domain error (sin=0 and cos=0)
59	Input value out of domain
60	Result too large
61	GETSPLC: size of destination parameter is insufficient
62	SSUB: number of characters negative
63	Result of operation is not a finite valid number
64	Result of operation is not a finite valid number
65	Trying to write E32059 : Value is out of limits (0 ... 360)

Error 95	Trying to write to a read only parameter
Extension	**Meaning of the error**
2	Trying to write E914xx
3	Writing E352xx without authorization (P7 N2.1)
4	Writing E6x007 with incline plane or RTCP active
5	[.xxx] not allowed for writing or keyword BSET not expected
6	Writing E940xx : Wrong limits in correction table
7	Writing E940xx : Wrong order in correction table
8	Writing E940xx : Difference between two consecutive corrections is higher than 65535
9	Trying to write E49xxx
10	Dynamic operators : First E parameter not allowed (not type1)
11	Trying to write E320xx
12	Dynamic operator 15, the axis must be declared as non-interpolated
13	Trying to write E20xxx
14	Trying to write E21xxx
15	Trying to write E37xxx
16	Trying to write E40xxx
17	Trying to write E41xxx
18	Trying to write E43xxx
19	Trying to write E44xxx
20	Trying to write E46xxx
21	Trying to write E47xxx
22	Trying to write E984xx
23	Trying to write E11012=1 with speed averaging filter active. Disable filter with E11021=0
24	Trying to write E11017 or E11018
25	Trying to write E11021 with HSC active
26	Trying to write parameter E32005 with HSC not enabled
27	Trying to write parameter E32011 with filter active
28	Trying to write E351xx
29	Trying to write parameter E5Yxxx with xxx higher than 255 or xxx = 0
30	Trying to write parameter E57xxx = 2 without Boring/Radial axis option
31	E69001: The Inclined axis and Inclined wheel states cannot be declared together
32	E69002: The Inclined axis and Inclined wheel states cannot be declared together
33	Trying to write E69003 with a turning plane active
34	Trying to write E492xx
35	Trying to write E90032
36	Trying to write E7x005 of an axis in another channel
37	Trying to write a wrong E7xxxx
38	Trying to write an E81xxx or E82xxx parameter already used for inter axes calibration
39	Trying to write E900xx on servo axis
40	Trying to write E923xx with axes involved in HSC

41	Trying to write E930xx
42	Trying to write E931xx
43	Trying to write E933xx
44	Trying to write E934xx
45	Trying to write E935xx
46	Trying to write E936xx
47	Trying to write E937xx
48	Trying to write E938xx
49	Trying to write E95xxx
50	UNUSED
51	Trying to write E3411x
52	Trying to write a wrong E90xxx parameter
53	Programed value of E790xx with xx less than 5 or higher than 8
54	Request to write a parameter outside the range E50000 / E70000
55	Trying to write E342xx
56	Trying to write E343xx
57	Trying to write E940xx with inter-axis calibration locked P7N0
58	Request to write E69xxx not accepted at this stage
59	Request to write E69005 parameter
60	Request to write E69xxx not accepted at this stage
61	Tool Manager E208xxxx - Life value is read-only
62	Trying to write parameter E207xxxx = 2 without Boring/Radial axis option
63	Parameter not allowed for string operations
64	Writing of E12000 is allowed only in PcP + PP 9999.0
65	Trying to read E915xx for a non-existent axis
66	Trying to write E915xx for a non-existent axis
67	Trying to read E916xx for a non-existent axis
68	Trying to write E916xx
69	Trying to write E91164-E91195

Error 96	No E parameter access because the previous block is incomplete
Extension	**Meaning of the error**
1	The block preceding the external parameter declaration is incomplete
2	
3	Internal error during the storage of the profile

Error 97	Assignment of a parameter in G76 not possible
Extension	**Meaning of the error**
1	Unknown character < 'A'
2	G76 : '=' expected after L variable or E parameter
3	G76 : character is not numerical
4	G76 : character is not numerical

5	Symbolic variables not in the stack
6	G76+ S[..] [..] Additional programing forbidden
7	G76- S[..] [..] Additional programing forbidden
8	G76- S[..] [..] Inconsistent structure number
9	G76- S[..] [..] Flag writing protection enable
10	G76+ S[..] [..] Inconsistent structure number
11	G76+ S[..] [..] Length structure null
12	G76+ S[..] [..] Structure not empty
13	G76+ S[..] [..] Structure length too small
14	G76+ S[..] [..] Flag writing protection enable

Error 98	Error declaring a dynamic operator
Extension	**Meaning of the error**
1	Operator declared on another channel

Error 99	Error associated with the N/M AUTO function
Extension	**Meaning of the error**
1	More than 5 axes defined N/M AUTO
2	Non-servo axis defined N/M AUTO
3	Definition of one N/M AUTO axis from another axes channel

Error 101	PGP: Insufficient data in the programing of a circle
Extension	**Meaning of the error**
1	G23 : Wrong syntax : Not all 4 coordinates are programed
2	G2/G3 wrong axis programed
3	G23: two programed points are equal
4	G23: circle not correctly defined

Error 102	PGP: The missing coordinate cannot be calculated
Extension	**Meaning of the error**
1	Starting point and ending point equal
2	Angle is too small

Error 103	PGP : Chamfer or Fillet
Extension	**Meaning of the error**
1	Starting point outside the segment programed before the chamfer or the fillet
2	Ending point outside the segment after the chamfer or the fillet
3	Starting point outside the arc before the fillet
4	Ending point outside the arc after the fillet
5	Segment<->Arc: radii arc and fillet inconsistent each other
6	Arc<->Arc: radii arcs and fillet inconsistent each other

Error 107	PGP: Circle programed
Extension	**Meaning of the error**
1	Ending radius too small

2	The distance between the start point and end point is greater than twice the programed radius
3	Undefined starting angle
4	Radius difference greater then tolerance

Error 110	PGP: Syntax error in the first of 2 blocks
Extension	**Meaning of the error**
1	PGP: Syntax error, missing EA/ES
2	PGP: Syntax error, missing EA/ES/ET
3	PGP: Syntax error, missing ES/I/J/R
4	PGP: Syntax error, missing ES/I/J/R

Error 111	PGP: Syntax error in the second block
Extension	**Meaning of the error**
1	PGP: Syntax error, missing ET/…
2	PGP: Syntax error, missing I/K/ES

Error 112	PGP: Line-line intersection
Extension	**Meaning of the error**
1	Starting point and ending point equal
2	Angle is too small

Error 113	PGP : Impossible to calculate the point of intersection
Extension	**Meaning of the error**
1	PGP: Calculated intermediate values are incorrect
2	PGP: No intersection
3	PGP: Starting point inside the circle
4	PGP: The end point of the 2nd block is internal to the 1st circle
5	PGP: No intersection
6	PGP: No tangency point
7	PGP: The 2 circles don't have the same direction
8	PGP: No tangency point
9	PGP: Curves are not intersecting
10	PGP: No tangency point

Error 114	PGP: Intersection or tangent point not defined by ET+, ET-, ES+ or ES-
Extension	**Meaning of the error**
1	PGP: Missing signs + / -
2	PGP: Intersection point not specified
3	PGP: Intersection point not specified

Error 121	PGP: Syntax error in last of 3 blocks
Extension	**Meaning of the error**
1	PGP : Syntax error in I/J/R or X/Z/I/J
2	PGP : Syntax error in I/J/R or X/Z/I/J

3	PGP : Syntax error in I/J/R or X/Z/I/J
4	PGP : Syntax error in I/J/R or X/Z/I/J
5	PGP : Syntax error in I/J/R or X/Z/I/J

Error 122 PGP : The first two blocks are non-secant lines	
Extension	**Meaning of the error**
1	PGP: Circle center = starting point
2	PGP: The angles of the two lines are equal
3	PGP: No line segment
4	PGP: Parallel lines

Error 123 PGP : Data in the 3 blocks doesn't allow the tangent points to be calculated	
Extension	**Meaning of the error**
1	PGP: Determination tangent point impossible between segment and circle
2	PGP: Backtracking point between the last two circles
3	PGP: No tangent segment between the two circles
4	PGP: Determination center of the 2nd circle impossible

Error 124 PGP: Tangent point second-third block not defined by ET+ or ET-	
Extension	**Meaning of the error**
1	PGP: Missing ET+ /ET-

Error 130 No axis movement in 1 of the 2 blocks connected by a fillet or chamfer	
Extension	**Meaning of the error**
1	Too small Segment or null

Error 131 M0, M1 or M2 is not allowed in a block with a fillet or chamfer	
Extension	**Meaning of the error**
1	Program stop after an incomplete block
2	PGP in progress
3	Fillet is on hold

Error 133 Curve programing error in Spline and Angular shift	
Extension	**Meaning of the error**
1	Spline buffer is full, impossible to complete the number of coefficients
2	Spline buffer is full, impossible to complete the number of coefficients

Error 135 A chamfer can only be between 2 lines	
Extension	**Meaning of the error**
1	Line-line intersection is required

Error 136 More than 2 non movement blocks between 2 incompletely defined elements	
Extension	**Meaning of the error**
1	Too many blocks without move.
2	Too many undefined blocks.

Error 137	Change of interpolation plane even though the block is not valid
Extension	**Meaning of the error**
1	A valid block is required

Error 138	Change of interpolation plane must be in G40
Extension	**Meaning of the error**
1	Transition from and to G17/G18/G18 must be in G40
2	Transition from G21 to G20 must be in G40
3	Transition from G20 to G21 must be in G40

Error 139	Two parallel [carried] axes programed in the same block
Extension	**Meaning of the error**
1	Outside G0
2	Outside G52

Error 140	Programing error in radius correction
Extension	**Meaning of the error**
1	Function cannot be programed when radius offset is active
2	Analysis of PP interrupted
3	Invalid block
4	Out of memory (BFF)

Error 141	Programing error in G41/G42 with collision avoidance
Extension	**Meaning of the error**
1	Setting number of blocks of anticipation and previous block not in G40
2	Number of blocks of anticipation outside the range allowed
3	Conditions not fulfilled to start with the first block from G40
4	Conditions not fulfilled to keep on with the first block from G41/G42
5	Conditions not fulfilled to keep on with the subsequent block from G40
6	Conditions not fulfilled to keep on with the subsequent block from G41/G42
7	Wrong block at beginning of corrected path
8	Not enough block for corrected path
9	Collision on first block programed
10	Collision on second block programed
11	Starting offset dimensions undetermined

Error 142	Error in jet cutting activation
Extension	**Meaning of the error**
1	Tool angle compensation option missing
2	Wrong head type (E69008)
3	Tool angle too small

Error 143	Error in tool vector programing or re-orientation
Extension	**Meaning of the error**
1	Rotary axes programing and tool vector programing incompatible

2	Tool vector programing without RTCP
3	Tool vector programing and RTCP rotary axis programing incompatible
4	Type kinematic not supported
5	No twist in RTCP kinematic
6	No RTCP rotary axes solution for kinematic type 2
7	Modulo tool vector orientation too small
8	Tool vector reference not aligned on axis X, Y or Z
9	Tool vector reference inconsistent with kinematic
10	RTCP rotary axes not programed yet (kinematic 2)
11	Programing all RTCP rotary axes requested
12	Type of tool vector programing inconsistent
13	Polynomial on RTCP rotary axes forbidden in tool vector re-orientation by RTCP rotary axes programing
14	Tool vector re-orientation by RTCP rotary axes programing and tool vector programing incompatible
15	Tool vector programing and RTCP rotary axes programing incompatible
16	No RTCP rotary axes solution for kinematic type 4
17	RTCP rotary axes not programed yet (kinematic 4)
18	Tool orientation programing: option is not activated

Error 144 Quantified axis displacement is different from the increment

Extension	Meaning of the error
1	Displacement is not multiple of the increment

Error 145 G29 : Absolute value of [P*P + Q*Q + R*R - 1000mm] higher than 1 mm

Extension	Meaning of the error
1	Normal vector length greater than 1000 mm within 1 mm in 3-axis correction
2	Normal vector length lower than 1000 mm within 1 mm in 3-axis correction

Error 146 3D Tool offsets, with G29 or G43

Extension	Meaning of the error
1	Missing option for 5-axes tool correction
2	All components P,Q,R,I,J,K must be programed
3	Components P,Q,R not programed
4	G43: I,J,K not fully programed
5	G43-/+: P,Q,R should not be programed
6	G43-/+: I,J,K should not be programed
7	G43-/+: displacement too short
8	G43-/+: angle tangent movement/tool direction too small

Error 147 In jet cutting no path possible

Extension	Meaning of the error
1	Direction of tool radius correction has changed
2	Jet cutting: Circle-Circle intersection
3	Jet cutting: Line-Line intersection

Error 148	Number of interpolated axes higher than the maximum allowed
Extension	**Meaning of the error**
1	System configuration mismatch

Error 149	Tool radius too large in relation to the programed trajectory
Extension	**Meaning of the error**
1	Tool radius too large > 20 IU
2	Jet cutting: Error in tool radius correction
3	Tool radius > Programed radius
4	Jet cutting: Radius at top < 50 increments IU
5	Jet cutting: Radius at bottom < 50 increments IU
6	In jet cutting junction not possible
7	Line-Circle intersection
8	Jet cutting: Circle-Circle intersection
9	Jet cutting: Circle-Circle intersection

Error 150	X axis out of limits
Extension	**Meaning of the error**
1	Circular interpolation
2	Circular interpolation
3	Circular interpolation
4	Circular interpolation
5	Inclined axis
6	Inclined axis
7	Inclined axis
8	Inclined axis
11	Dynamic operator 21
14	Over-travel on tool tip in RTCP (end of block)
15	Over-travel on tool tip in Tilted Plane (end of block)
16	Over-travel on axis outside RTCP and Tilted Plane (end of block)
17	Over-travel on tool tip outside RTCP and Tilted Plane (end of block)
18	Over-travel negative on linear axis in RTCP (end of block)
19	Over-travel positive on linear axis in RTCP (end of block)
20	Over-travel on programed axis (end of block)
21	RTCP : Position less than travel min
22	RTCP : Position higher than travel max
23	RTCP : Position less than travel min
24	RTCP : Position higher than travel max
25	RTCP : Circle programed - position less than travel min
26	RTCP : Circle programed - position higher than travel max
27	RTCP : Position less than travel min
28	RTCP : Position higher than travel max

29	Dynamic operator 21 : Total cycle displacement exceeds limits
30	Spline execution exceeds limits
31	Flexium Tools Frequency Analysis / Stimulus Over-travel

Error 151 Y axis out of limits	
Extension	**Meaning of the error**
Please refer to the Error 150 extensions.	

Error 152 Z axis out of limits	
Extension	**Meaning of the error**
Please refer to the Error 150 extensions.	

Error 153 U axis out of limits	
Extension	**Meaning of the error**
Please refer to the Error 150 extensions.	

Error 154 V axis out of limits	
Extension	**Meaning of the error**
Please refer to the Error 150 extensions.	

Error 155 W axis out of limits	
Extension	**Meaning of the error**
Please refer to the Error 150 extensions.	

Error 156 A axis out of limits	
Extension	**Meaning of the error**
Please refer to the Error 150 extensions.	

Error 157 B axis out of limits	
Extension	**Meaning of the error**
Please refer to the Error 150 extensions.	

Error 158 C axis out of limits	
Extension	**Meaning of the error**
Please refer to the Error 150 extensions.	

Error 159 Move demanded on a not-homed axis	
Extension	**Meaning of the error**
1	G26 : Axes not homed
2	Linear interpolation : Programed movement on a non-homed axis
3	Programed movement on a non-homed axis

Error 160 Slave axis homing error	
Extension	**Meaning of the error**
1	Axis already on stop

Error 161	Synchronized axis referencing error
Extension	**Meaning of the error**
1	Home cycle not complete

Error 162	Emergency stop: G75 not declared
Extension	**Meaning of the error**
1	G75 not declared in common channels
2	G75 not declared in independent channels

Error 163	Torque and Reference management error
Extension	**Meaning of the error**
1	Warning Torque and Reference under Flexium Tools / FXServer control

Error 164	Axis out of limits
Extension	**Meaning of the error**
1	Flexium Tools Frequency Analysis / Stimulus Over-travel

Error 165	Homing absolute encoder
Extension	**Meaning of the error**
1	Difference between new and stored position greater than tolerance

Error 190	Too many nested jumps or loops [15 max]
Extension	**Meaning of the error**
1	Too many branch or loop nesting levels (maximum 15) WHILE / IF loops
2	Too many branch or loop nesting levels (maximum 15) FOR loops

Error 191	Syntax error in structured programing
Extension	**Meaning of the error**
1	CUT: Missing '[' character
2	THEN: Forbidden key word in this programing context.
3	Structured programing is forbidden in MDI mode.
4	FOR: Missing '=' character
5	FOR: Missing key word TO/DOWNTO
6	FOR: Missing key word DO
7	FOR: Syntax error after key word 'TO'
8	FOR: Exxx parameter forbidden. Must be >= E80000
9	FOR: Exxx parameter forbidden. Must be < E90000
10	PUSH/PULL: Missing first 'L' argument
11	PUSH/PULL: Missing '-' character
12	PUSH/PULL: Missing second 'L' argument
13	PUSH/PULL: L parameter out of range L100-L199
14	PUSH/PULL: L parameter out of range L900-L959
15	BCLR: Missing '[' character

16	BCLR: Missing '.' character
17	SEARCH: Missing '[' character
18	SEARCH: Missing argument 'N'
19	R.OFF: Missing '[' character
20	MOVE: Missing '[' character
21	DELETE: Missing '[' character
22	SAVE: Missing '[' character
23	IF/WHILE/UNTIL: Symbol missing among 'A-'Z in an expression (character<'A')
24	IF/WHILE/UNTIL: Symbol missing among 'A-'Z in an expression (character>'Z')
25	IF/WHILE/UNTIL: Wrong variable after '<' ,'=', '>' operator
26	FOR/IF/WHILE/UNTIL: Wrong variable. Lnn/Exx/[Symbolic variable] required
27	DO: Forbidden key word in this programing context.
28	Unknown [or forbidden] key word in this program context
29	IF: Missing key word THEN
30	WHILE: Missing key word DO
31	FOR: Missing key word DO
32	FOR: Missing key word DO
33	ENDI: ISO codes programed in the same block as ENDI are forbidden
34	ENDW: ISO codes programed in the same block as ENDW are forbidden
35	ENDF: ISO codes programed in the same block as ENDF are forbidden
36	EXIT: ISO codes programed in the same block as EXIT are forbidden
37	UNTIL: ISO codes programed in the same block as UNTIL are forbidden
38	SWITCH: Expression value out of range]-32767 , 32767[
39	ENDS: ISO codes programed in the same block as ENDS are forbidden
40	BREAK: ISO codes programed in the same block as BREAK are forbidden
41	CONT: ISO codes programed in the same block as CONT are forbidden
42	Unknown memory area in the syntax SEARCH H.. :s (0<s<4)
43	SEARCH: L/E Parameter out of range (L0-L99, E80000-E84999)
44	SWITCH: ISO codes programed in the same block as SWITCH are forbidden

Error 192 Unknown [or forbidden] key word in this program context

Extension	Meaning of the error
1	Unknown key word in this program context
2	THEN: Forbidden key word in this programing context.
3	DO: Forbidden key word in this programing context.
4	AND: Forbidden key word in this programing context.
5	OR: Forbidden key word in this programing context.
6	TO: Forbidden key word in this programing context.
7	DOWNTO: Forbidden key word in this programing context.
8	STEP: Forbidden key word in this programing context.

Error 193 Structuration error	
Extension	**Meaning of the error**
1	Structuration error
2	
3	VAR: Missing key word ENDV or not the first word in the block
4	VAR: Missing key word ENDV. End of file reached
5	VAR: Missing key word ENDV.
6	VAR: Missing key word ENDV.
7	IF: Missing key word ELSE/ENDI
8	WHILE: Missing key word ENDW
9	IF: Missing key word THEN
10	IF: Missing key word ENDI
11	ENDW: Forbidden key word in this programing context.
12	ENDW: Forbidden key word in this programing context.
13	ENDI: Forbidden key word in this programing context.
14	UNTIL: Forbidden key word in this programing context.
15	
16	
17	ENDF: Forbidden key word in this programing context.
18	FOR: Missing key word ENDF
19	WHILE: Missing key word ENDW
20	REPEAT: Missing key word UNTIL
21	FOR: Missing key word ENDF
22	WHILE: Missing key word ENDW
23	FOR: Missing key word ENDF
24	FOR: Missing key word ENDF
25	IF: Missing key word ENDI
26	IF: Missing key word ENDI
27	IF: Potential error on key word ELSE if programed
28	SWITCH: Missing key word CASE/DEFAULT
29	SWITCH: Structuration error
30	ENDS: Forbidden key word in this programing context.
31	BREAK: Missing key word ENDS
32	BREAK: Structuration error
33	CASE: Missing key word ENDS
34	CASE: Structuration error
35	DEFAULT: Structuration error
36	CONT keyword used outside a loop
37	EXIT keyword used outside a loop
38	CASE: Invalid constant expression
39	SWITCH: Missing key word CASE/DEFAULT

Error 195 Stack overflow	
Extension	**Meaning of the error**
1	Stack overflow when saving L variables with PUSH function.
2	Stack overflow when saving symbolic variables with SAVE function.
3	Stack overflow when adding a new symbolic variable.
4	Number of defined constants exceeds the reservation
5	Insufficient memory on the stack for the storage of the profile of BUILD/P.BUILD function.
6	Insufficient memory on the stack for the storage of the profile of Fillets and Chamfers.

Error 196 Error in the declaration of the table index	
Extension	**Meaning of the error**
1	Symbolic variable length is higher than the maximum allowed (32)
2	Table dimension or index is negative or zero
3	Table dimension or index > 65535
4	First table dimension or index > 255
5	Missing ',' in the expression of the table dimension or index
6	Table dimension or index out of range [1..255]
7	Table dimension or index is negative
8	BUILD: Missing 'P' character
9	BUILD: Exceeding the maximum number of items (16)
10	BUILD/P.BUILD: Non existent item
11	BUILD: Exceeding the maximum number of items (16)
12	CUT: First dimension must be equal to 7
13	CUT: Second dimension is higher than the maximum allowed (255)
14	CUT: Second dimension is higher than the maximum allowed (255)
15	R.OFF: First dimension must be between 7 and 15
16	R.OFF: Second dimension is higher than the maximum allowed
17	R.OFF: The first column of the table doesn't contain the code of G1/G2/G3 (0/-1/+1)
18	R.OFF: Second dimension is higher than the maximum allowed (255)
19	R.OFF: Second dimension is higher than the maximum allowed (255)
20	R.OFF: Second dimension is higher than the maximum allowed (255)
21	MOVE: Invalid end index of destination table.
22	MOVE: Invalid start index of destination table.
23	MOVE: Invalid end index of destination table.
24	MOVE: Invalid start index of destination table.
25	MOVE: Invalid end index of source table.
26	MOVE: Invalid start index of source table
27	MOVE: Invalid end index of source table
28	MOVE: Invalid index
29	MOVE: Invalid dimension of destination/source table
30	MOVE: Invalid index

31	MOVE: Invalid destination index
32	MOVE: Invalid index
33	MOVE: Invalid source index
34	MOVE: Invalid index
35	MOVE: Invalid index
36	MOVE: Invalid first dimension of source table
37	MOVE: Invalid first dimension of destination table
38	Invalid index reading a table element
39	MOVE: Invalid dimension of source table
40	MOVE: Invalid dimension of destination table

Error 197 Use of an undeclared symbol	
Extension	**Meaning of the error**
1	FOR: [. XXX] Is prohibited as a variable.
2	PULL: L variable not saved
3	DELETE: Symbolic variable not found
4	SAVE: Symbolic variable not found
5	Invalid table [.xxx(y)]
6	Table index out of range [.Ixx(p)]
7	Invalid variable [.Rxx]
8	Invalid variable [.BGxx]/[.BMxx]
9	Invalid variable [.xxx]
10	Table index out of range [.Ixx(p)]
11	Invalid variable [.BGxx]/[.BMxx]
12	Invalid variable [.BGxx]/[.BMxx]
13	Variable not found
14	NC symbolic variable with name length null
15	NC symbolic variable with unknown location
16	NC symbolic variable with unknown key word
17	NC symbolic variable with unknown name
18	Missing ')' in NC symbolic variable
19	Missing ']' in NC symbolic variable
20	Unknown indexed symbolic variable
21	Unknown symbolic variable

Error 198 Syntax error in declaring the symbol for a variable	
Extension	**Meaning of the error**
1	R.OFF: Missing ',' character
2	R.OFF: Missing ')' character
3	CUT: Missing ',' character
4	CUT: Missing ')' character
5	Wrong symbol. First character must be in [@,A/a-Z/z]

6	Symbolic variable length is higher than the maximum allowed (32)
7	Indirect addressing forbidden from NC symbolic variable
8	R.OFF: Missing ',' or '/' character
9	R.OFF: Missing '(' character
10	R.OFF: Missing ')' character
11	Access to a type not allowed of symbolic variable
12	C.INIT: Too many columns in input contour
13	C.INIT: Missing "]" character
14	C.INIT: Unexpected end of command
15	C.INIT: Expected ")"
16	C.INIT: Expected "="
17	C.INIT: Expected "="
18	C.INIT: Expected "]"
19	C.OFF Unexpected argument
20	C.OFF: Invalid parameter
21	C.INIT: Too few input columns for polar contour
22	C.INIT: Too many points for polar contour
23	C:INIT: Too few input columns for Cartesian contour
24	C.INIT / C.OFF: Expected value
25	C.INIT: Expected "="
26	C.INIT: Expected "["
27	C.INIT: Invalid offset side
28	C.OFF: Argument without value
29	Bad values in the input contour

Error 199	Syntax error in declaring a variable
Extension	**Meaning of the error**
1	R.OFF: Missing '/' character before first argument
2	R.OFF: Wrong argument. Must be +1/-1
3	R.OFF: Missing '/' character before second argument
4	MOVE: Missing '=' character
5	MOVE: Missing '[' character
6	Internal error. Current Part Program corrupted.
7	Syntax error in declaring a variable. Missing '[' character.
8	BSET/BCLR: Wrong list of symbols to be enabled or inhibited
9	BUILD/P.BUILD: Syntax error in this programing context.
10	BUILD/P.BUILD: Missing ',' character
11	BUILD/P.BUILD: Syntax error. ']' / '\n' forbidden in this programing context.
12	UNUSED
13	BUILD/P.BUILD: Missing ']' character
14	Dynamic operator 21 : Missing '[' char for symbolic variable array
15	Dynamic operator 21 : Dimension of symbolic variable must be [5,2]

16	Dynamic operator 21 : Values Ta0 of symbolic table out of limits
17	Dynamic operator 21 : Values Ta1 of symbolic table out of limits
18	Dynamic operator 21 : Values Ta2 of symbolic table out of limits
19	Dynamic operator 21 : Values Ra of symbolic table out of limits
20	Dynamic operator 21 : Values Tr0 of symbolic table out of limits
21	Dynamic operator 21 : Values Tr1 of symbolic table out of limits
22	Dynamic operator 21 : Values Tr2 of symbolic table out of limits
23	Dynamic operator 21 : Values Tr of symbolic table out of limits
24	Syntax error in declaring a variable. Missing ']' character.
25	BSET/BCLR: M3, M4 and M5 cannot be used
26	Symbolic variable size too big
27	Expected Symbolic variable separator "/"
28	Expected Symbolic variable separator "/"
29	R.OFF: Discarded blocks or connection inserted on polar contour
30	R.OFF: Discarded blocks or connection inserted on polar contour
31	R.OFF optional angle out of range (0°-120°)
32	C.OFF optional angle out of range (0°-120°)

Error 200 Poor signal or complementarity error on handwheel 0

Extension	Meaning of the error
Please refer to the Error 203 extensions.	

Error 201 Poor signal or complementarity error on handwheel 1

Extension	Meaning of the error
Please refer to the Error 203 extensions.	

Error 202 Poor signal or complementarity error on handwheel 2

Extension	Meaning of the error
Please refer to the Error 203 extensions.	

Error 203 Poor signal or complementarity error on handwheel 3

Extension	Meaning of the error
1	Handwheel in failure
2	Sensor complementarity error
3	Sensor dirty signal
4	Counter error

Error 209 Reboot NC

Extension	Meaning of the error
1	Reset parameter memory
2	Reset DRAM
3	Reset NVRAM

Error 210 Feedback fault on axis 0

Extension	Meaning of the error

1	Sensor failure (axes / spindles)
2	Sensor complementarity error
3	Sensor dirty signal
4	Counter error
5	Wrong PLCAAI_x function call
6	SSI or EnDat: CounterError from PLC
7	EnDat: wrong LinePer Rev value from PLC

Error 211 Feedback fault on axis 1	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 212 Feedback fault on axis 2	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 213 Feedback fault on axis 3	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 215 Feedback fault on axis 5	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 216 Feedback fault on axis 6	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 217 Feedback fault on axis 7	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 218 Feedback fault on axis 8	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 219 Feedback fault on axis 9	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 220 Feedback fault on axis 10	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 221 Feedback fault on axis 11	
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 222	Feedback fault on axis 12
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 223	Feedback fault on axis 13
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 224	Feedback fault on axis 14
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 225	Feedback fault on axis 15
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 226	Feedback fault on axis 16
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 227	Feedback fault on axis 17
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 228	Feedback fault on axis 18
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 229	Feedback fault on axis 19
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 230	Feedback fault on axis 20
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 231	Feedback fault on axis 21
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 232	Feedback fault on axis 22
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 233	Feedback fault on axis 23
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 234	Feedback fault on axis 24
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 235	Feedback fault on axis 25
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 236	Feedback fault on axis 26
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 237	Feedback fault on axis 27
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 238	Feedback fault on axis 28
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 239	Feedback fault on axis 29
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 240	Feedback fault on axis 30
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 241	Feedback fault on axis 31
Extension	**Meaning of the error**
Please refer to the Error 210 extensions.	

Error 245	Digital servo loop fault
Extension	**Meaning of the error**
1	Drive alarm reset
2	Measure not available
3	Overload communication error
4	Drive in alarm or out of synchronization
5	Sequence communication error
6	Communication error
7	Drive in HALT
8	Drive-DSP life bit fault

Error 246	Emergency Request
Extension	**Meaning of the error**
1	Emergency Req Active

Error 247	Drive parameter fault DISC NT+
Extension	**Meaning of the error**
1	Internal error generated to be able to reset a lag error

Error 248	Axis drive or digital spindle fault
Extension	**Meaning of the error**
1	Drive alarm is active
2	Actual measure value not available
3	Drive out of sync

Error 249	Communication error between CNC and PLC
Extension	**Meaning of the error**
1	Watch Dog is down, PLC stopped or disconnected
2	PLC stopped or disconnected

Error 250	Invalid PERSOChip
Extension	**Meaning of the error**
1	NCK dongle not found

Error 251	Inconsistent PERSOChip Data
Extension	**Meaning of the error**
1	Invalid dongle

Error 252	Invalid license File
Extension	**Meaning of the error**
1	No valid license
2	Flexium+ option is mandatory
3	Wrong temporary license
4	Temporary license expired
5	Temporary license expired
6	System date changed

Error 253	Nesting of subroutines called by M function not allowed
Extension	**Meaning of the error**
1	A subroutine called by M function can't call itself another subroutine by M function

Error 254	Floating point unit exception error
Extension	**Meaning of the error**
1	Unrecoverable exception

Error 266	Insufficient memory
Extension	**Meaning of the error**
1	Invalid file address: address out of range
2	G76+ Insufficient memory
3	G76+ Insufficient memory
4	G76+ Insufficient memory

5	G76+ Insufficient memory
6	G76+ cannot create file
7	G76+ cannot create file
8	G76- cannot remove file
9	G76- cannot remove file

Error 267 Please get in contact with NUM Service. The system has a major problem	
Extension	**Meaning of the error**
1	Hardware issue

Error 300 Axis/Spindle 0 configuration fault	
Extension	**Meaning of the error**
1	Axis/Spindle DISC-NT+ declared in P120 but not detected on the bus
2	Axis/Spindle DISC-NT+detected on the bus but not correctly declared in P120
3	Analog Axis/Spindle not correctly declared in P120
4	Spindle analog port not correctly declared in P70

Error 301 Axis/Spindle 1 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 302 Axis/Spindle 2 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 303 Axis/Spindle 3 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 304 Axis/Spindle 4 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 305 Axis/Spindle 5 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 306 Axis/Spindle 6 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 307 Axis/Spindle 7 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 308 Axis/Spindle 8 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 309 Axis/Spindle 9 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 310 Axis/Spindle 10 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 311 Axis/Spindle 11 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 312 Axis/Spindle 12 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 313 Axis/Spindle 13 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 314 Axis/Spindle 14 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 315 Axis/Spindle 15 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 316 Axis/Spindle 16 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 317 Axis/Spindle 17 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 318 Axis/Spindle 18 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 319 Axis/Spindle 19 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 320 Axis/Spindle 20 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 321 Axis/Spindle 21 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 322 Axis/Spindle 22 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 323 Axis/Spindle 23 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 324 Axis/Spindle 24 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 325 Axis/Spindle 25 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 326 Axis/Spindle 26 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 327 Axis/Spindle 27 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 328 Axis/Spindle 28configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 329 Axis/Spindle 29 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 330 Axis/Spindle 30 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 331 Axis/Spindle 31 configuration fault	
Extension	**Meaning of the error**
Please refer to the Error 300 extensions.	

Error 400	Loading dynamic operators in C: The size of user code is too big
Extension	**Meaning of the error**
1	File to big or corrupted

Error 401	Loading dynamic operators in C: Format error
Extension	**Meaning of the error**
1	File corrupted

Error 402	Loading dynamic operators in C: Checksum error
Extension	**Meaning of the error**
1	File corrupted

Error 403	The system has insufficient memory for dynamic operators in C
Extension	**Meaning of the error**
1	Out of memory (Dynop C area)

Error 404	Loading dynamic operators in C: Open error
Extension	**Meaning of the error**
1	File not found

Error 405	Loading dynamic operators in C: Read error
Extension	**Meaning of the error**
1	File not found or corrupted

Error 406	Loading dynamic operators in C: Close error
Extension	**Meaning of the error**
1	File not found or corrupted

Error 407	Loading dynamic operators in C: The directory is empty
Extension	**Meaning of the error**
1	File not found

Error 410	Dynamic operators in C: Number of parameters passed does not match
Extension	**Meaning of the error**
1	Too much or not enough parameters

Error 411	Dynamic operators in C: User error in INIT
Extension	**Meaning of the error**
1	User error declared in Dynamic operator in C

Error 413	Unrecognised dynamic operators in C
Extension	**Meaning of the error**
1	Operator not found

Error 414	Dynamic operators in C without MAIN
Extension	**Meaning of the error**
1	Main function is missing

Error 420	Dynamic operators in C: User error in the QUIT function
Extension	**Meaning of the error**
1	User defined error

Error 421	Dynamic operators in C: User Error in the QUIT function
Extension	**Meaning of the error**
1	User defined error

Error 423	Dynamic operators in C: Range of function in C not from [0..100]
Extension	**Meaning of the error**
1	Out of range

Error 424	Internal system error
Extension	**Meaning of the error**
1	Wrong internal case

Error 500	Tilted Planed configuration : Invalid option FXSO000914
Extension	**Meaning of the error**
-	Not present extension

Error 501	Invalid T argument given the 5 axes kinematic configuration
Extension	**Meaning of the error**
-	Not present extension

Error 502	T arg.: positive index of setting table of RTCP function : T1 up to T8
Extension	**Meaning of the error**
-	Not present extension

Error 503	T arg.: index limited to 8 configurations
Extension	**Meaning of the error**
-	Not present extension

Error 504	Non authorized secondary axis
Extension	**Meaning of the error**
-	Not present extension

Error 510	RTCP configuration : Invalid option FXSO000154
Extension	**Meaning of the error**
-	Not present extension

Error 511	Syntax error in the current block G151Sn or G150Sn :S value is erroneous (S0 or S1 or S2 for RTCP) or (S0 or S1 for iPLANE)
Extension	**Meaning of the error**
-	Not present extension

Error 512	Angular measurement by encoder is required by RTCP control
Extension	**Meaning of the error**
-	Not present extension

Error 513	Error in the call conditions of RTCP by G77 statement: see the process in state G999
Extension	**Meaning of the error**
-	Not present extension

Error 514	Programing error of RTCP or iPLANE function : The value of an argument is not allowed
Extension	**Meaning of the error**
-	Not present extension

Error 515	Programing error of tool retraction
Extension	**Meaning of the error**
1	R argument value is erroneous (allowed R1)
2	Tool frame computation doesn't support the 5 axis kinematics

Error 517	Programing Error in the cycle248 of 5-axis kinematics calibrating
Extension	**Meaning of the error**
1	Current CNC mode prohibited in a touch-probing cycle
2	T arg.: positive index of setting table of RTCP function : T1 up to T8
3	T arg.: index limited to 8 configuration table : T1 up to T8
4	T argument is missing: Add the argument T to select the table of settings for your machine kinematics
5	Cycle248 configuration: Invalid option FXSO000153
6	No tool corrector programed : program a tool corrector conforming to the 3D probe of the customization file
7	The 5-axis machine not managed by the current version of kinematics measuring/calibrating cycle
8	Exceeding the allowed size of swivelling S axis: nbS <=18 indexing positions
9	Exceeding the allowed size of rotary R axis: nbR <=18 indexing positions
10	The kinematic measuring cycle must be done separately on one rotary axis only, the other axis being at zero position
11	The customization file of RTCP function is incompatible with the kinematic measuring cycle248
12	The programed rotary axes of the first block are mandatorily the reference zero angles

Error 518	Error occurred while execution of kinematic measurement cycle
Extension	**Meaning of the error**
1	IT error : no interrupt is detected on a descent movement towards the top of the ball
2	The measuring cycle doesn't currently support this kinematics
3	Invalid value of EI arg. : the allowed values: (0) for the first measuring cycle (STEP1) ; (1) for residual errors measuring (STEP2)
4	Error domain of arc tangent computation : the both arguments sin/cos are null
5	Collision detected during the protected positioning

Error 519	Sphere solving error
Extension	**Meaning of the error**
1	Ill conditioned problem in the sphere solving subroutine

Error 530	Failure of part features measuring cycle or failure of the probe calibration cycle
Extension	**Meaning of the error**
1	Required missing options: Option bit 46/bit 47/bit 61/bit 71

2	No tool corrector
3	Definition error of the workplane and the tool orientation
4	Feed rate set to zero
5	Probe Integrity Failure (Contact Status)
6	Status of probing input :In progress or Busy
7	Error of the rotary table definition : check the variable [TAV.g] in the customization file PERSO.XPI
8	Bore diameter not programed (ED argument)
9	EA Angle of a shoulder or of an oblique surface is out of range
10	Erroneous position of the probe with respect to an oblique surface (EG value)
11	Gauging correctors not programed
12	G147 Cycle: Erroneous value EQ defining the axis measurement
13	G149 EF calibration function doesn't exist (EF value is wrong)
14	Programing error: EF isn't programed
15	G149 EF The current version of the calibration cycle is limited to milling
16	G149 EF3 missing argument in the calling of probe calibration cycle in a bored hole (EX EY EZ ou ED)
17	G149 EF4 cycle doesn't work if the probe length refers to the ball-center (i.e. the variable [Len_ref]=0 in the PROBE file)
18	Erroneous value EQ defining the axis measurement
19	G145 Cycle: Erroneous value EQ defining the axis measurement
20	Direction of measurement, programed by ES, is incorrect
21	Erroneous value EQ to set or restore DAT2 shifts of linear axes (Part Balancing)
22	Rotary table is missing: check the variable [TAV.g] in PROBE file
23	Erroneous value EQ defining the axis measurement
24	G145 Cycle: Erroneous value ED defining the distance between the points A and B (ED>0).
25	Erroneous value EQ defining the axis on which DAT3 off-centring is to be adjusted
26	Erroneous value of the spindle indexing angle EA=xxx for the 1st tooth of milling tool
27	Erroneous value of the highest tooth DE = xxx for the teeth number EZ of a milling tool EZ>0
28	Spindle probe number EP not present in the PROBE.XPI file
29	Measurement number EN not present in the MEASURES file
30	The dimension is outside the tolerance programed by ET
31	Programing Error: non-existent calibrating function: permitted values EF1 or EF2
32	Programing Error: non-existent calibrating function: permitted values are EF1 up to EF4
33	Error while the spindle indexing control (M03/M04 or spindle indexing speed)
34	Wrong value of number of teeth EZ=z of reference tool
35	Wrong value of tool radius: tool corrector E52xxx>0
36	Wrong value of the programed width EW for measuring the features of a groove or rib
37	Wrong value of the height ED>0 for measuring the features of a rib or web
38	Wrong value of the clearance distance EC in front a rib or web
39	The safety travel along the abscissa axis is too small to reach the surface
40	The safety travel along the ordinate axis is too small to reach the surface
41	Wrong value of the diameter EW for measuring the features of a boss

42	Wrong value of the height ED>0 for measuring the features of a boss
43	Wrong value of the clearance distance EL in front the surface to be measured
44	Wrong number EN assigned to the bore or to the boss,the permitted value is between [1 25]
45	Wrong number EK assigned to the 2nd bore or to the 2nd boss, the permitted value is between [1 25]
46	Impossibility to evaluate the dimensions between two features (bores or bosses)
47	Wrong value of the maximum gap ED>0,025mm between two measuring points
48	Incorrect offset EX along the abscissa axis to perform both measurements
49	Incorrect offset EY along the ordinate axis to perform both measurements
50	The zero radius of the probe ball: Set parameter E51xxx or the variable[PrbDiam] or adjust the dynamic radius by calibrating the probe
51	Programing error in Cycle G158 EF, an argument is missing or a transmitted value is incorrect
52	Incorrect use of part measurement cycles. These cycles can only be used with a milling spindle probe
53	Collision detected during the protected positioning
54	Angular increment EI cannot exceed 90° for a 4-point circle measurement or 120° for a 3-point circle measurement
55	Incorrect distribution of points on the circle for its characterization
56	The recording address EI of the 1st boss compared to the 2nd boss must refer to the block numbered N600 or N750
57	The recording address EJ of the 2nd boss compared to the 1st boss must refer to the block numbered N600 or N750
58	The recording address EI of the 1st bore compared to the 2nd bore must refer to the block numbered N500 or N700
59	The recording address EJ of the 2nd bore compared to the 1st bore must refer to the block numbered N500 or N700

Error 600 Curve number must be different of zero	
Extension	**Meaning of the error**
-	Not present extension

Error 601 N... N... must be programed	
Extension	**Meaning of the error**
-	Not present extension

Error 602 No axes programed in the first block	
Extension	**Meaning of the error**
-	Not present extension

Error 603 Slope of the curve undefined	
Extension	**Meaning of the error**
-	Not present extension

Error 604 Less than three blocks in the profile	
Extension	**Meaning of the error**
-	Not present extension

Error 605 Unknown curve number

Extension	Meaning of the error
-	Not present extension

Error 700 Configuration error detected by VEComp G246 cycle	
Extension	**Meaning of the error**
1	Missing option 99 or 100 requested by VEComp
2	Missing option 102 needed for ROTARY VEComp
3	Data Segment is probably not initialized by VEComp

Error 701 Programing error detected by VEComp G246 cycle	
Extension	**Meaning of the error**
1	VEComp: Wrong value of T argument to specify the machine kinematic
2	VEComp:Inconsistency between T argument and numbers of data files
3	VEComp: Keywords missing in the kinematic data structure
4	VEComp:Deviation symbol missing in X axis error motions file passed by EX arg.
5	VEComp:Deviation symbol missing in Y axis error motions file passed by EY arg.
6	VEComp:Deviation symbol missing in Z axis error motions file passed by EZ arg.
7	VEComp:Deviation symbol missing in U axis error motions file passed by EU arg.
8	VEComp:Deviation symbol missing in V axis error motions file passed by EV arg.
9	VEComp:Deviation symbol missing in W axis error motions file passed by EW arg.
10	VEComp:Deviation symbol missing in A axis error motions file passed by EA arg.
11	VEComp:Deviation symbol missing in B axis error motions file passed by EB arg.
12	VEComp:Deviation symbol missing in C axis error motions file passed by EC arg.
13	VEComp:Deviation symbol missing in X2 axis error motions file passed by ER arg.
14	VEComp:Deviation symbol missing in Y2 axis error motions file passed by ES arg.
15	VEComp:Inconsistency between the location & orientation error file and the kinematic structure
16	VEComp: wrong value of ET arg. giving the spindle axis direction
17	Internal unit unprocessed by VEComp function
18	VEComp:Incorrect unit for linear deviations
19	VEComp:Incorrect unit for the angular deviations
20	VEComp:Inconsistency between X axis addr. and the deviation data file
21	VEComp:Inconsistency between Y axis addr. and the deviation data file
22	VEComp:Inconsistency between Z axis addr. and the deviation data file
23	VEComp:Inconsistency between U axis addr. and the deviation data file
24	VEComp:Inconsistency between V axis addr. and the deviation data file
25	VEComp:Inconsistency between W axis addr. and the deviation data file
26	VEComp:Inconsistency between A axis addr. and the deviation data file
27	VEComp:Inconsistency between B axis addr. and the deviation data file
28	VEComp:Inconsistency between C axis addr. and the deviation data file
29	VEComp:Inconsistency between X2 slave axis addr. and the deviation data file
30	VEComp:Inconsistency between Y2 slave axis addr. and the deviation data file
31	VEComp:Inhomogeneous unit of the linear deviations between data files

32	VEComp:Inhomogeneous unit of angular deviations between data files
33	VEComp: Wrong index EK for selecting an 5-axis configuration parameters table, permitted values 1 up to 8
34	VEComp: Wrong Index EK, limited to 8 configuration parameters tables

Error 702 VEComp G246 cycle Error detected in the initialisation routine	
Extension	**Meaning of the error**
1	VEComp:Invalid address of DRAM segment
2	VEComp:Invalid address of sub data segment
3	VEComp: Kinematic number not found or kinematic model not handled by the function
4	VEComp:DRAM segment reading error
5	VEComp:Code number of Kinematic not found in data segment
6	Internal unit unprocessed by VEComp function
7	VEComp: Unassigned pointer detected in routine of spatial errors computation
8	VEComp: Null pointer in routine of DRAM copy
9	VEComp: A compensated position of an axis exceeds the axis stroke limits
10	VEComp: Wrong size of the X axis error motions tables
11	VEComp:Wrong size of the Y axis error motions tables
12	VEComp:Wrong size of the Z axis error motions tables
13	VEComp:Wrong size of the U axis error motions tables
14	VEComp:Wrong size of the V axis error motions tables
15	VEComp:Wrong size of the W axis error motions tables
16	VEComp:Wrong size of the A axis error motions tables
17	VEComp:Wrong size of the B axis error motions tables
18	VEComp:Wrong size of the C axis error motions tables
19	VEComp:Wrong size of the X2 axis error motions tables
20	VEComp:Wrong size of the Y2 axis error motion tables
21	VEComp:Invalid option 99 by redundant checking
22	VEComp:Invalid option 100 by redundant checking
23	VEComp:Sorting failure of X2 axis error motion tables
24	VEComp:Sorting failure of Y2 axis error motion tables
25	VEComp:Sorting failure of X axis error motion tables
26	VEComp:Sorting failure of Y axis error motion tables
27	VEComp:Sorting failure of Z axis error motion tables
28	VEComp:Sorting failure of U axis error motion tables
29	VEComp:Sorting failure of V axis error motion tables
30	VEComp:Sorting failure of W axis error motion tables
31	VEComp:Sorting failure of A axis error motion tables
32	VEComp:Sorting failure of B axis error motion tables
33	VEComp:Sorting failure of C axis error motion tables
34	VEComp:Incorrect unit of linear deviations
35	VEComp:Incorrect unit of angular deviations

36	VEComp: Null pointer in the routine to reset the motion errors
37	VEComp: Null pointer in the routine to reset the location errors

Error 703	VEComp: Error detected in the RTS Opdyn task
Extension	**Meaning of the error**
1	VEComp: Out of range of the inner states of filter
2	VEComp: Out of range of the states of filter in initialization step
3	VEComp: Computed exceeding of stroke limits of X axis
4	VEComp:Computed exceeding of stroke limits of Y axis
5	VEComp:Computed exceeding of stroke limits of Z axis
6	VEComp:Unassigned pointer in routine of spatial errors computation
7	VEComp:Unable to compute a weighted compensation of spatial errors for the gantry
8	VEComp:Compensation value of X axis error motion exceeds the max tolerance: see the data files of axis error motions
9	VEComp:Compensation value of Y axis error motion exceeds the max tolerance: see the data files of axis error motions
10	VEComp:Compensation value of Z axis error motion exceeds the max tolerance: see the data files of axis error motions

Error 704	VEComp ROTARY G247 cycle Programing Error
Extension	**Meaning of the error**
1	VEComp Rotary:Invalid value of T argument to specify the machine kinematic
2	VEComp Rotary:Kinematic number T no found or kinematic model not handled by ROTARY VEComp
3	VEComp Rotary:Invalid set of kinematic dimensions EK
4	VEComp Rotary:Invalid file number of deviations EF
6	VEComp Rotary:Inconsistency between A axis addr. and the deviation data file
7	VEComp Rotary:Inconsistency between B axis addr. and the deviation data file
8	VEComp Rotary:Inconsistency between C axis addr. and the deviation data file

Error 705	VEComp ROTARY G247 cycle Error at initializing process
Extension	**Meaning of the error**
1	VEComp Rotary:Invalid address of DRAM Segment Sn
2	VEComp Rotary:Invalid address of Data sub-Segment
3	VEComp Rotary:Data Segment is empty
4	VEComp Rotary:The size of used segment is not allocated to the kinematic object
5	VEComp Rotary:Kinematic number no found or kinematic model not handled by Rotary VEComp fct
6	VEComp Rotary:Reading error in DRAM segment
7	VEComp Rotary:Kinematic code number not found in data segment
8	VEComp Rotary:Internal unit unprocessed by ROTARY VEComp function
9	VEComp Rotary:Invalid option 99 by redundant checking
10	VEComp Rotary:Null pointer in routine of DRAM copy
11	VEComp Rotary:Sorting failure of data arrays of dual axes : invalid address
12	VEComp Rotary:Sorting failure of data arrays of the swivel head: invalid address
13	VEComp Rotary:Sorting failure of data arrays of the rotary table: invalid address

14	VEComp Rotary:Sorting failure: wrong size of a data array
15	VEComp Rotary:Sorting failure: invalid address of one of data arrays
16	VEComp Rotary:Sorting failure: invalid type of kinematic chain
18	VEComp Rotary:Size of data array is out of range : 180 maxi
19	VEComp Rotary:Size of data array is out of range : 72 maxi
20	VEComp Rotary:Size of data array is out of range : 36 maxi
21	VEComp Rotary:Incorrect unit of linear deviations
22	VEComp Rotary:Incorrect unit of angular deviations

Error 706 VEComp ROTARY G247 cycle Error in the RTS task	
Extension	**Meaning of the error**
1	VEComp Rotary:Unassigned pointer in routine of Cartesian errors computation
2	VEComp Rotary:Computed exceeding of stroke limits of X axis
3	VEComp Rotary:Computed exceeding of stroke limits of Y axis
4	VEComp Rotary:Computed exceeding of stroke limits of Z axis

Error 750 G725 Programing Error	
Extension	**Meaning of the error**
1	No width for the trochoidal path
2	No path
3	Too many lines in the path (max.128)
4	Discontinuity on trochoidal path: possible intersection of adjacent segments
5	Different nbr of segments between trochoidal and center-tool path: possible path intersection

Error 751 G725 Width of the trochoidal path too small for the tool programed	
Extension	**Meaning of the error**
-	Not present extension

Error 752 G725 Width of the trochoidal path larger than the diameter of at least one arc in the profile	
Extension	**Meaning of the error**
-	Not present extension

Error 753 G725 Illegal value for EG2/EG3 argument for trochoidal machining	
Extension	**Meaning of the error**
-	Not present extension

Error 754 G725 Illegal value for EE Finishing argument for trochoidal machining	
Extension	**Meaning of the error**
-	Not present extension

Error 755 G725 Invalid option 106.	
Extension	**Meaning of the error**
-	Not present extension

Error 760 G730 Programing Error	
Extension	**Meaning of the error**
1	Height of the letters missing
2	Space between letters missing
3	Depth of the engraving missing
4	Text missing
5	Character not handled

Error 761 G730 Axial path less than engraving depth	
Extension	**Meaning of the error**
-	Not present extension

Error 762 G730 Height of the letters less or equal to 0	
Extension	**Meaning of the error**
-	Not present extension

Error 763 G730 Space between letters less or equal to 0	
Extension	**Meaning of the error**
-	Not present extension

Error 764 G730 invalid option 107	
Extension	**Meaning of the error**
-	Not present extension

Error 830 Positioning not carried out	
Extension	**Meaning of the error**
-	Not present extension

Error 831 Spindle not turning	
Extension	**Meaning of the error**
-	Not present extension

Error 832 End point, P and K must be programed	
Extension	**Meaning of the error**
-	Not present extension

Error 833 Retract value inadequate	
Extension	**Meaning of the error**
-	Not present extension

Error 834 EB value: -90 < EB < +90	
Extension	**Meaning of the error**
-	Not present extension

Error 835 No negative values for P, Q, R and K	
Extension	**Meaning of the error**
-	Not present extension

Error 836	The interpolation plane must be G18 or G20
Extension	**Meaning of the error**
-	Not present extension

Error 837	Illegal Value for F or S
Extension	**Meaning of the error**
-	Not present extension

Error 838	Re-Thread: incoherent data
Extension	**Meaning of the error**
-	Not present extension

Error 839	Search In Block error
Extension	**Meaning of the error**
1	Plane G20, G21, G22 forbidden
2	Retract dimension NOT sent by HMI
3	Plane axes NOT sent by HMI
4	Tool radius offset > trajectory radius
5	G01 and no plane axis programed in block to re-engage
6	One axis plane dimension missing in block to re-engage or in its previous one
7	Dimension of working axis missing in block to re-engage
8	Variation of dimension of axis programed in block to re-engage NOT computable
9	Segment length too short
10	Difference too large between starting and ending radius
11	Radius too short
12	Abscissa dimension of center NOT null
13	Positioning dimensions of axes planes NOT sent by HMI
14	Dimension of circle center NOT sent by HMI
15	Previous block to the one to re-engage NOT in G40, G41 or G42
16	Block to re-engage NOT in G40, G41, G42
17	Block to re-engage and its previous one NOT in the same status G40, G41 or G42
18	No sub-routine for arguments of the sub-program by G function
19	Mirroring changed between block to re-engage and its previous one
20	G70 programed in block to re-engage or in its previous one
21	G74 programed in block to re-engage or in its previous one
22	Axes re-assignment programed in block to re-engage or in its previous one
23	Direction rotation NOT sent by HMI
24	Direction rotation inconsistent between HMI and NCK
25	Programing G1 EA.. with EA inconsistent
26	Programing G1 EA.. with distance null
27	Programing G2/G3 with coordinate center missing OR EA.. missing
28	Programing G2/G3 with coordinate missing OR R.. missing
29	Programing G2/G3 with R.. : point not reachable

30	Start fillet or chamfer in block to re-engage

Error 840	G38+ NOT allowed with argument ER
Extension	**Meaning of the error**
-	Not present extension

Error 841	ER value: 0 < ER < +90
Extension	**Meaning of the error**
-	Not present extension

Error 842	Thread feed X axis exceeds speed limit
Extension	**Meaning of the error**
-	Not present extension

Error 843	End of thread feed Z axis exceeds speed limit
Extension	**Meaning of the error**
-	Not present extension

Error 844	Thread feed Z axis exceeds speed limit
Extension	**Meaning of the error**
-	Not present extension

Error 845	End of thread feed X axis exceeds speed limit
Extension	**Meaning of the error**
-	Not present extension

Error 846	Start of thread feed Z axis exceeds speed limit
Extension	**Meaning of the error**
-	Not present extension

Error 847	Start of thread feed X axis exceeds speed limit
Extension	**Meaning of the error**
-	Not present extension

Error 848	End of Thread movement argument R too long
Extension	**Meaning of the error**
-	Not present extension

Error 849	End of Thread movement argument ER too big
Extension	**Meaning of the error**
-	Not present extension

Error 850	Reduced retract value inadequate
Extension	**Meaning of the error**
-	Not present extension

Error 851	EK < 1 in G84
Extension	**Meaning of the error**
-	Not present extension

Error 852	ER Mandatory in MDI mode
Extension	**Meaning of the error**
-	Not present extension

Error 853	Cycle programed without tool axis pre-positioning
Extension	**Meaning of the error**
-	Not present extension

Error 862	End point, P or R must be programed
Extension	**Meaning of the error**
-	Not present extension

Error 863	End point incompatible with EA
Extension	**Meaning of the error**
-	Not present extension

Error 864	Milling tool prohibited in G66
Extension	**Meaning of the error**
-	Not present extension

Error 871	Finish profile limits undefined
Extension	**Meaning of the error**
-	Not present extension

Error 872	Dimension(s) missing in blank definition
Extension	**Meaning of the error**
-	Not present extension

Error 873	P or R not programed
Extension	**Meaning of the error**
-	Not present extension

Error 874	Blank incompatible with finish profile
Extension	**Meaning of the error**
-	Not present extension

Error 875	No intersection of EA with profile
Extension	**Meaning of the error**
-	Not present extension

Error 876	Wrong rake angle EB
Extension	**Meaning of the error**
-	Not present extension

Error 880	Cycle axis not known
Extension	**Meaning of the error**
-	Not present extension

Error 881	Parameter value not compatible
Extension	**Meaning of the error**
-	Not present extension

Error 882	Bottom of hole dimension missing
Extension	**Meaning of the error**
-	Not present extension

Error 883	Pitch (I J K) or retract (P) missing
Extension	**Meaning of the error**
-	Not present extension

Error 884	More than 9 thread starts
Extension	**Meaning of the error**
-	Not present extension

Error 885	Pocket not compatible with selected plane
Extension	**Meaning of the error**
-	Not present extension

Error 886	Tool incompatible with programed radius
Extension	**Meaning of the error**
-	Not present extension

Error 887	Cut > tool diameter
Extension	**Meaning of the error**
-	Not present extension

Error 888	Dwells prohibited in this cycle
Extension	**Meaning of the error**
-	Not present extension

Error 889	Syntax error
Extension	**Meaning of the error**
-	Not present extension

Error 890	Incompatible tool orientation
Extension	**Meaning of the error**
-	Not present extension

Error 891	Return plane = bottom of hole
Extension	**Meaning of the error**
-	Not present extension

Error 892	Axial feed missing
Extension	**Meaning of the error**
-	Not present extension

Error 893	Lateral feed missing
Extension	**Meaning of the error**
-	Not present extension

Error 894	ER is prohibited in G20
Extension	**Meaning of the error**
-	Not present extension

Error 895	G21,G22 is prohibited in cycle
Extension	**Meaning of the error**
-	Not present extension

Error 896	Dimension incompatible with tool radius
Extension	**Meaning of the error**
-	Not present extension

Error 897	Length of oblong pocket < diameter
Extension	**Meaning of the error**
-	Not present extension

Error 898	Tool corrector missing
Extension	**Meaning of the error**
-	Not present extension

Error 899	Spindle not assigned to this channel or spindle/channel incompatibility
Extension	**Meaning of the error**
-	Not present extension

9.2 CNC Warning (HMI messages)

Warning	Description
1	**CN_INIT Drives not found** Warning during the boot phase, the drives initialization isn't correct. At the end of the boot, see the NC Error 245/248.
2	**CN_INIT All Drives HALT Timeout** Warning during the boot phase, the drives aren't able to go in the HALT state At the end of the boot, see the NC Error 245/248.
3	**CN_INIT All Drives RUN Timeout** Warning during the boot phase, the drives aren't able to go in the RUN state. At the end of the boot, see the NC Error 245/248.
4	**CN_INIT All Drives SYNCHRONIZED Timeout** Warning during the boot phase, the drives aren't able to go in the SYNCHRONIZED state At the end of the boot you can see the NC Error 245/248.
5	**CN_INIT All Drives MESDISPO Timeout** Warning during the boot phase, the MESDISPO bit is OFF At the end of the boot, see the NC Error 245/248.
6	**CN_INIT OPDYN Loading Format Error** The format of the Dynamic operator 22 written in C is wrong.
7 to 20	CN_INIT msg reserved
21	**DISCNT Writing parameter in Unknown Drive address: %1%2** The Drive at address xx isn't available.
22	**DISCNT Unknown Drive parameter: V %1%2%3%4** The Drive parameter xxxx isn't available.
23	**DISCNT Acyclic Request p_Bufian NULL** The Communication with Drives interrupted, see the NC Error 245/248.
24	**DISCNT Acyclic Request Error Drive address: %1%2** The Communication with Drives address xx interrupted, see the NC Error 245/248.
25	**DISCNT Acyclic Request List Frame Error** The Communication with Drives interrupted, see the NC Error 245/248.
26	**DISCNT Acyclic Request Unknown** The Communication with Drives interrupted, see the NC Error 245/248.
27	**DISCNT Timeout Drive Detection** The Communication with Drives interrupted, see the NC Error 245/248.
28	**DISCNT Acyclic Request Timeout** The Communication with Drives interrupted, see the NC Error 245/248.
29	**DISCNT Repetition communication error Ring %1** The Communication with Drives interrupted Ring x, see the NC Error 245/248.
30	**DISCNT Overload communication error Ring %1** The Communication with Drives interrupted Ring x, see the NC Error 245/248.
31	**DISCNT Sequence communication error Ring %1** The Communication with Drives interrupted Ring x, see the NC Error 245/248.
32	**Drive-DSP life bit fault address %1%2** Life bit Drive address xx is OFF, see the NC Error 245/248.
33 to 50	**DISCNT msg reserved**
51	**ETH T Code Function not Acknowledged**
52	**ETH E30000 Read not Acknowledged**
53	**ETH E40000 Read not Acknowledged**
54	**ETH E30000 Write not Acknowledged**

55	**ETH REQ UNITE pRMG NULL** Communication with the PLC interrupted, see the Comm 1009 HMI message.
56	**ETH Unknown Message Received from RTS** See the Comm 1009 HMI message.
57	**ETH REQ UNITE Buffers full** Communication with the PLC interrupted, see the Comm 1009 HMI message.
58	**ETH RTCSPCB Alloc Send Error** Communication with the PLC interrupted, see the Comm 1009 HMI message.
59	**ETH M Code Function not Acknowledged**
60	**ETH Decoded M Function not Acknowledged**
61	**ETH Wrong Mac address of destination**
62 to 70	**ETH msg reserved**
71	**PP Loading out of order** Error during file download (see others HMI message)
72	**PP Loading no free buffer** Error during file download (see others HMI message)
73	**Axis %1 Channel %2 Axis travel limit reached with Jog %3**
74	**Axis %1 Channel %2 Axis travel limit reached with Handwheel %3**
75	**G751: Cannot open file in CNC MFS partition** Error in the G751 function.
76	**G751: MFS partition is full** Error in the G751 function.
77	**G750: Cannot open file in CNC MFS partition** Error in the G750 function.
78	**G750: XM0 memory area full** Error in the G750 function.
79	**G750: Download failed** Error in the G750 function.
80	**G750: Access denied to XM0 area** Error in the G750 function.
81	**G750: not possible to save destination file** Error in the G750 function.
82	**G750: not possible to overwrite destination file** Error in the G750 function.
83	**Axis %1 Channel %2 Axis travel limit %3 reached**
84	**OPDYN attempts to move stopped axis %1**
85	**Temporary license expired**
86	**System date changed**
87	**No Limits for the Collision detection are available**
88	**Collision detected Axis %1 Channel %2 Axis travel limit reached with Jog %3**
89	**Collision detected Axis %1 Channel %2 Axis travel limit reached with Handwheel %3**
90	**Collision detected Axis %1 Channel %2 Axis travel limit %3 reached**
91	**Online collision detection Option not available**
92	**Quantified Axis %1 Channel %2 Axis travel limit reached with Jog %3**
164	**Axis @%1%2 out of limits**
165	**Slave axis movement not allowed**

166	**Manual Homing not allowed for a master axis when E961xx are active**
1000	**EGB follower axis @%1 failure (lost origin)**
1001	**EGB follower axis @%1 failure**
1002	**EGB follower axis @%1 failure**
1003	**EGB follower axis @%1 failure (over travel)**
1004	**EGB follower axis @%1 failure (synchro error)**
1005	**PLC DisableServoCntrl requested on the axis @%1%2 transformed by inclined plane or RTCP**
1006	**PLC DisableServoCntrl requested while axes are moving in the channel %1**
1007	**E910xx = 1 requested on the axis @%1%2 while DisableServoCntrl is active from PLC**
1008	**Display message with parametric value: value too large**
1999	**Warning by PP out of range (< 2000)**
2000	**The position deviation along %1 is out of fixed tolerance**
2001	**The width measured along %1 is out of fixed tolerance**
2002	**The width measured along the normal axis to the shoulder is outside the fixed tolerance**
2003	**The measured diameter of the bore is outside the fixed tolerance.**
2004	**The measured diameter of the boss is outside the fixed tolerance.**
4999	Reserved for NUM MACRO

9.3 Drives warning

1	Motor I2t intervention
2	Motor over-temperature
3	Drive over-temperature
4	Braking resistor over-temperature
5	EPSM will cause phase offset switching after POM/APEM
6	Error UGA sensor S1
7	Error UGA sensor S2
8	Drive I^2t intervention
9	Following-error overflow in conversion from Phys. Unit to Internal Unit.
10	S1 ENDAT 2.2 CRC error
11	S2 ENDAT 2.2 CRC error
12	NUM-STOX over time 35 hours
13	NUM-SAMX test required
14	S1 Sensor battery low
15	S2 Sensor battery low
16	DSL cable near breakage: substitute as soon as possible
17	S1 "bad signal": sensor scale should be cleaned
18	S2 "bad signal": sensor scale should be cleaned

Page deliberately empty

Page deliberately empty

Published by NUM Spa

Via F. Somma 62
20012 Cuggiono (MI) (Italy)
Tel. : +39 02 979 691
website: www.num.com

Printed October 2022

CNC HighEnd Applications