Hợp tác Đức
DEUTSCHE ZUSAMMENARBEIT

Funding programme
develoPPP
Where business meets development.

Implemented by
giz Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH

SIEMENS

LILAMA2 COLLEGE

Development Partnership with the Private Sector

Vocational Training for Smart Manufacturing in Machine Tools

Training Module

Advanced DIN/ISO Programming and Operation of CNC Machines - Turning and Milling

develoPPP.de programme of the German Federal Ministry for Economic Cooperation and Development

Hợp tác Đức
DEUTSCHE ZUSAMMENARBEIT

Funding programme
develoPPP
Where business meets development.

Implemented by
giz Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH

SIEMENS

LILAMA2 COLLEGE

Development Partnership with the Private Sector

Vocational Training for Smart Manufacturing in Machine Tools

Siemens Vietnam - GIZ - LILAMA 2

Training Module Advanced DIN/ISO Programming and Operation of CNC Machines - Turning and Milling

Development Partnership with the Private Sector
Vocational Training for Smart Manufacturing in Machine Tools
Siemens Vietnam - GIZ - LILAMA 2

Training Module
Advanced DIN/ISO Programming and Operation of CNC Machines
– Turning and Milling

Training Materials – Exercises – Project Works – Solutions

Editorial Board

Hartmut Stein, Phan Thi Anh Tu, Le Tuyen Giao
Nguyen Xuan Huy, Nguyen Trong Luc
Ralf Hill

Contributions

LILAMA 2 International Technology College
Mechanical Engineering Faculty
Training Department
Quality Assurance Department

GIZ Programme Reform of TVET in Viet Nam
Pham Thi Thanh Truc
Nguyen Minh Cong

Translation **English language to Vietnamese**
Tran Simon Trung Hieu
Vu Trinh Mai Tram

The book ***Training Module Advanced DIN/ISO Programming and Operation of CNC Machines – Turning and Milling*** *is officially published by the cooperation partners of the development partnership* ***"Vocational Training for Smart Manufacturing in Machine Tools".***

It was disseminated to eleven Institutes of the ***Vietnamese-German Programme Reform of Technical and Vocational Education and Training (TVET) in Viet Nam****, and other relevant stakeholders in TVET and in the manufacturing industry.*

The book can be copied or downloaded on ***www.tvet-vietnam.org*** *free for academic purpose and research without commercial interest. For any other purposes of use and duplication, inquire* ***Programme Reform of TVET in Viet Nam*** *for further information and permission.*

Address: *#1, Lane 17, Ta Quang Buu Street, Bach Khoa Ward, Hai Ba Trung District, Hanoi, Viet Nam*
Tel: *+84 (0) 24 39 74 64 71*
Fax: *+84 (0) 24 39 74 65 70*
Website: *www.tvet-vietnam.org*
www.giz.de/vietnam

FOREWORD

Dear Industry 4.0 stakeholders in Technical and Vocational Education and Training and manufacturing industry

Technical and Vocational Education and Training (TVET) plays a crucial role in the development of a skilled and competent workforce for advanced industries. The adaptation of knowledge and skills of the Vietnamese workforce to the new competence requirements due to digitalisation and Industry 4.0 (I 4.0) is a challenge that needs to be addressed by TVET institutes in close cooperation with the business sector.

With support of the Vietnamese Directorate of Vocational Education and Training (DVET) under the Ministry of Labour, Invalids and Social Affairs (MoLISA), the partners Siemens, LILAMA 2 International Technology College (LILAMA 2) and the Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ), established the development partnership ***"Vocational Training for Smart Manufacturing in Machine Tools".*** The development partnership qualifies teachers of LILAMA 2 and other TVET institutes, technicians, and engineers from the manufacturing industry in Viet Nam to meet the requirements of I 4.0

Development partnerships with the private sector promote activities where entrepreneurial opportunities and development policy potential meet and are supported by the ***www.develoPPP.de*** programme of the German Federal Ministry for Economic Cooperation and Development (BMZ).

The main activities of the partners comprised: establishing the "Digital Technology Education Centre" for smart manufacturing at LILAMA 2; developing inclusive training modules and teaching and learning materials on higher digitalization and I 4.0 competencies based on German training standards and on the industry needs; improving technical and didactical competencies of TVET teachers and in-company trainers for smart manufacturing for metal processing occupations; disseminating developed training modules and multiplying knowledge and skills to the Vietnamese TVET system; promoting gender equality and raising awareness about the employment potential of people of disadvantaged groups and their career options in industrial occupations and in TVET.

National and international smart manufacturing experts and curriculum designers developed the training module "Advanced DIN/ISO Programming and Operation of CNC Machines – Turning and Milling" and further trained teachers of LILAMA 2 and other partner TVET institutes of the Vietnamese-German Programme Reform of TVET in Viet Nam as master trainers and multipliers for the TVET system.

The newly developed training module is based on German standard and fulfils the qualification needs of national and international companies in Viet Nam. It complies to Circular 03/2017 of the MoLISA and resembles the structure of the initial training programmes for intermediate and college levels, developed in the frame of the Programme Reform of TVET in Viet Nam. They are available for free download at ***www.tvet-vietnam.org.*** The module and the teaching and learning material build up on these programmes and provide additional qualification in sophisticated smart manufacturing applications. The module can be offered as an elective training module for college level graduates of the three-years training programme Metal Cutting Technician but also as further training to industry technicians and engineers in the field of mechanical engineering and machine tools.

TVET teachers are recommended to offer the training module as hybrid learning or blended learning, in which trainees learn via electronic and online media, as well as traditional face-to-face teaching and training.

On behalf of the partners, we would like to express our utmost gratitude for the assistance of all parties to develop the training module "Advanced DIN/ISO Programming and Operation of CNC Machines - Turning and Milling" and wish a successful implementation and dissemination.

Dr Juergen Hartwig
Director
Programme Reform of TVET in Viet Nam

ABBREVIATIONS

2D	Two-Dimensional
3D	Three-Dimensional
BMZ	Federal Ministry for Economic Cooperation and Development
CLDATA	Cutter Location Data
CNC	Computer Numerical Control
DIN	Deutsches Institut für Normung e.V. (German Institute for Standardization)
DPP	Development Partnerships with the Private Sector
DT	Driven Tool
DVET	Directorate of Vocational Education and Training
GIZ	Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH
HLL	High Level Language
HSK	Horizontal Softkey bar
ISO	International Standardization Organization
IT	Information Technology
I.4.0	Industry 4.0
LED	Light Emitting Diode
LILAMA 2	LILAMA 2 International Technology College
MCS	Machine Coordinate System
MoLISA	Ministry of Labour, Invalids and Social Affairs
NC	Numerical Control
PLC	Programmable Logic Controller
PPE	Personal Protective Equipment
SZS	Settable Zero System
TC	Tool Carrier
ToRs	Terms of reference
TVET	Technical Vocational Education and Training
USB	Universal Serial Bus
VSK	Vertical Softkey bar
W	Workpiece
WCS	Workpiece Coordinate System
WPD	Working Programme Directory

TABLE OF CONTENTS

ABBREVIATIONS 08

A. TRAINING MODULE 12

I. MODULE CLASSIFICATION AND CHARACTERISTICS: 12

II. MODULE OBJECTIVES: 12

III. MODULE CONTENT: 14

1. General content and time allocation 14

2. Detailed content 17

IV. REQUIREMENTS FOR MODULE IMPLEMENTATION 26

1. Vocational classroom/ Training workshop 26

2. Equipment and machinery 26

3. Teaching and learning materials, tools, consumption materials 26

4. Other requirements and conditions: None 27

V. ASSESSMENT CONTENT AND METHODOLOGY 28

1. Assessment content: 28

2. Assessment methods 30

VI. GUIDELINES FOR PROFESSIONAL MODULE IMPLEMENTATION: 31

1. Scope of application 31

2. Guidelines on teaching and learning methods 31

3. Aspects to be considered 32

4. Reference materials 32

5. Further notes and explanations (if any) 32

B. TRAINING UNITS 33

UNIT 1: FUNDAMENTAL DIN/ISO PROGRAMMING – CNC – TURNING AND MILLING PROGRAMMING BASICS 33

1. Introduction: 33

2. Objective: 33

3. Content: 33

3.1. Program management and basic operations 33

3.2. Operation area for parameter offset 41

3.3. Kiến thức cơ bản về lập trình hướng đối tượng 47

3.4. Contour definition and contour call in the program 50

3.5. Programme turning cycles 53

3.6. Programme milling cycles 60

3.7. Programme drilling cycles 73

3.8. Programme jump and branches, check structures 78

3.9. Plan and realise operational work order of the industry 78

TABLE OF CONTENTS

UNIT 2: ADVANCED DIN/ISO PROGRAMMING – CNC TURNING WITH DRIVEN TOOLS (C AND Y AXIS) AND COUNTER SPINDLE 127
1. Introduction: 127
2. Objective: 127
3. Content: 127
3.1. Planes when working with driven tools 127
3.2. Setup the tool offset, spindle chuck data and work offset 134
3.3. Programming of driven tools (DT) on the front surface 149
3.4. Programming of driven tools on the circumference surface 162
3.5. Applying of milling and drilling operations with C and Y-axis 195
3.6. Working areas of machines regard to the frames and how to calcu late them for programming 213
3.7. Advanced programming of exercises with counter spindle and driven tools with the cycles and G-codes 214
3.8. Plan and realise operational work order of the industry 235

UNIT 3: ADVANCED DIN/ISO PROGRAMMING – CNC MILLING 3+2 AXES – SWIVELLING IN ACCORDANCE WITH DIN/ISO 243
1. Introduction: 243
2. Objective: 243
3. Content: 243
3.1. Simplifying programming for 3+2 axes with programGUIDE 243
3.2. Tool carrier of a 5-axis milling machine 245
3.3. Kinematics of 5-axis machines 249
3.4. Functional explanation for the swivel cycle (CYCLE800) 251
3.5. Swivel modes by the programming 253
3.6. Advantages of programming with the CYCLE800 255
3.7. Advanced programming of exercises with swivelled planes, the cycles and G-codes 256
3.8. Plan and realise operational work order of the industry 304

UNIT 4: CNC PROGRAMMING OF HIGH-LEVEL LANGUAGE FOREXTREMELY DEMANDING CNC TURNING AND MILLING APPLICATIONS OF THE INDUSTRY 317
1. Introduction: 317
2. Objective: 317
3. Content: 317
3.1. Introduction and overview of the basics and functionality 317
3.2. Subprogram 328
3.3. R Parameter 338
3.4. Jumps and user variables 351
3.5. System variables and subprogram 363
3.6. System variables and safety query 377

TABLE OF CONTENTS

3.7. Frame concept 391
3.8. Programming of arithmetic functions 404
3.9. USER CYCLES 418
3.10. Plan and realise operational work order of the industry 454

UNIT 5: MACHINE INTEGRATED MEASURING OF WORKPIECES AND TOOLS WITH 3D TOUCH PROBES 459
1. Introduction: 459
2. Objective: 459
3. Content: 459
3.1. Measuring instruments for measuring workpieces 459
3.2. Calibration of measuring instruments for measuring accuracy 463
3.3. Measuring in the Setup mode 468
3.4. Measuring in Automatic mode 472
3.5. Measuring results through logging 483
3.6. Log measurement results with using the high-level-language 501

APPENDIX 512

1. Project regions and partnering tvet institutes 512
2. List of charts 513
3. List of tables 514
4. List of figures 515

REFERENCES 518

A. TRAINING MODULE

(According circular No. 03/2017/TT-BLĐTBXH March 1, 2017 of the Ministry of Labour, Invalids and Social Affairs)

Module name: Advanced DIN/ISO Programming and Operation of CNC Machines - Turning and Milling

Module code: MD01

Scheduled time: 320 hours (Theory: 119h; Practice, Laboratory, Discussion, Assignments: 189h; Examinations / Assessments: 12h)

I. Module classification and characteristics:

- **Classification:**

  Training module for higher Industry 4.0 qualification – International College Level. Admission requirement: National College Diploma in Metal Cutting.

- **Characteristics:**
  - + The module orients on German training and examination standard. It mediates digtalisation topics and advanced Industry 4.0 competencies as higher qualification in the fields of metal cutting. The module is structured in 5 training units that contents build up on each other's. It can be offered as supplementary training module for for college graduades of initial training programmes as well as offered to the industry in hybrid short-term trainings for technicians, engineers, and in-company trainers.
  - + CNC programmers increasingly face special tasks that take them to the limits of classic (DIN 66025) and graphical CNC programming. The CNC programming language, used in the training module, is internationally standardised applied. The trainees can build up on their already aquired competencies and further develop the necessary understanding for highly demanding machining processes with multi-axis CNC machines. They plan and realise complex industrial work assigments on 5-axis CNC milling centres as well on CNC turning machines with C and Y axes and with counter spindle.
  - + The trainees know and observe the regulations on IT security and occupational safety as well as health and environmental protection.

II. Module objectives:

By the end of this module, the trainees will be able to

- **Knowledge:**

- know the regulations of IT security, occupational safety, health and environmental protection when dealing with multi axis CNC machine tools
- describe the structure, functional units and safety devices of 5-axis CNC milling machines
- describe the structure, functional units and safety devices of CNC lathes with C and Y axes and with a counter spindle
- apply of DIN/ISO programming for special applications that go beyond the limits of classic G code and graphic CNC programming
- know and apply the benefit of rogramme-integrated measuring to reduce set-up times and to ensure quality features.

- **Skills:**
  - consolidate and deepen the fundamental DIN/ISO programming CNC turning and milling
  - programme and operate turning machines with driven tools and counter spindle in G code on advanced level
  - programme and operate milling machines with 3+2 axes in G code by swivel cycles on advanced level
  - programme high-level languages for extremely demanding CNC turning and milling applications of the industry
  - apply quality assurance through machine integrated measuring of work pieces and tools with CNC turning and milling machines
  - plan and realise complex industrial work assigmentsion system.
- **Autonomy and responsibility:** the trainees are able to work independently and in teams according to the work assignment, to
  - determine work steps and processes according to functional, organizational and manufacturing criteria
  - design workflow diagrams for custom tasks according to the order
  - research and evaluate information for order planning in digital networks
  - carry out complex manufacturing processes in compliance with regulations on IT security and occupational safety as well as environmental protection
  - arrange, monitor and control sub orders
  - develop special task solutions in terms of customer requests
  - apply high-level-language programming for highest flexibility in processing
  - secure quality assurance via program-integrated work steps
  - create measure protocols via the program to archive the quality features.

III. Module Content:

1. General content and time allocation:

Nr.	Modular Training Units	Scheduled time (hours)			
		Total	Theorie	Practice, Laboratory, Discussion, Assignments	Examination/ Assessment
1	**1. Fundamental DIN/ISO programming – CNC-turning and milling Programming basics** 1.1. Program management and basic operations 1.2. Operation area for parameter offset 1.3. General basics for programGUIDE 1.4. Contour definition and contour call in the program 1.5. Programme turning cycles 1.6. Programme milling cycles 1.7. Programme drilling cycles 1.8. Programme jump and branches, check structures 1.9. Plan and realise operational work order of the industry	40	15	24	1
2	**2. Advanced DIN/ISO programming – CNC turning with driven tools (C and Y axis) and counter spindle** 2.1. Planes when working with driven tools 2.2. Setup the tool offset, spindle chuck data and work offset 2.3. Programming of driven tools (DT) on the front surface 2.4. Programming of driven tools on the circumference surface 2.5. Applying of milling and drilling operations with C and Y-axis	60	20	38	2

	2.6. Working areas of machines regard to the frames and how to calculate them for programming 2.7. Advanced programming of exercises with counter spindle and driven tools with the cycles and G-codes 2.8. Plan and realise operational work order of the industry 2.9. Practice your English				
3	**3. Advanced DIN/ISO programming – CNC milling 3+2 axes – Swivelling in accordance with DIN/ISO** 3.1. Simplifying programming for 3+2 axes with programGUIDE 3.2. Tool carrier of a 5-axis milling machine 3.3. Kinematics of 5-axis machines 3.4. Functional explanation for the swivel cycle (CYCLE800) 3.5. Swivel modes by the programming 3.6. Advantages of programming with the CYCLE800 3.7. Advanced programming of exercises with swivelled planes, the cycles and G-codes 3.8. Plan and realise operational work order of the industry 3.9. Practice your English	60	20	38	2
4	**4. CNC programming of high-level language for extremely demanding CNC turning and milling applications of the industry** 4.1. Introduction and overview of the basics and functionality 4.2. Subprogram 4.3. R Parameter 4.4. Jumps and user variables	120	50	65	5

	4.5. System variables and subprogram 4.6. System variables and safety query 4.7. Frame concept 4.8. Programming of arithmetic functions 4.9. User cycles 4.10. Plan and realise operational work order of the industry 4.11. Practice your English				
5	**5. Machine integrated measuring of workpieces and machine tools with 3D touch probes** 5.1. Measuring instruments for measuring workpieces 5.2. Calibration of measuring instruments for measuring accuracy 5.3. Measuring in the Setup mode 5.4. Measuring in Automatic mode 5.5. Measuring results through logging 5.6. Log measurement results with using the high level language (HLL)	40	14	24	2
	Total hours:	**320**	**119**	**189**	**12**

2. Detailed content

Training Unit 1: FUNDAMENTAL DIN/ISO PROGRAMMING – CNC TURNING AND MILLING Time: 40 hours

1. Objective: By the end of this training unit, the trainees will be able to

- Present the required programming parameters of the integrated machining cycles in the Sinumerik 840Dsl Operate version 4.8
- Analyze detail drawings and customer's technical requirements
- Complete the work plan for each mechanical part in the order
- Calculate and select machining parameters for required mechanical parts
- Use mechanical manuals to look up relevant technical information
- Use G-code instructions and suitable cycles to program machining for specific applications
- Optimize specified programming files.

2. Content

2.1. Program management and basic operations
- 2.1.1. Main screen
- 2.1.2. Program manager
- 2.1.3. Softkey bars
- 2.1.4. Basic operations of programming

2.2. Operation area for parameter offset
- 2.2.1. Administration of tool offset
- 2.2.2. Administration of the work offset
- 2.2.3. Archiving setup data

2.3. General basics for programGUIDE
- 2.3.1. Overview G codes with linked addresses
- 2.3.2. Tool call-up
- 2.3.3. Overview M codes
- 2.3.4. Overview high level language commands

2.4. Contour definition and contour call in the program
- 2.4.1. Contour definition
- 2.4.2. Contour call in the program

2.5. Programme turning cycles
- 2.5.1. Stock removal and cut residual stock cycles
- 2.5.2. Groove cycles
- 2.5.3. Undercut cycles

2.5.4. Thread cycles
2.5.5. Cutoff cycle
2.6. Programme milling cycles
2.6.1. Path milling cycle
2.6.2. Pocket cycles
2.6.3. Spigot cycles
2.6.4. Slot cycles
2.6.5. Arbitrary contour pocket cycles
2.6.7. Arbitraty contour spigot cycles
2.7. Programme drilling cycles
2.7.1. Drilling cycles
2.7.2. Reaming cycle
2.7.3. Tapping Cycle
2.8. Programme jump and branches, check structures
2.8.1. Programme jump and branches
2.8.2. Check structures
2.9. Plan and realise operational work order of the industry
2.9.1. Turning application using ProgramGUIDE
2.9.2. Milling application using ProgramGUIDE

Training Unit 2: ADVANCED DIN/ISO PROGRAMMING – CNC TURNING WITH DRIVEN TOOLS (C AND Y AXIS) AND COUNTER SPINDLE Time: 60 hours

1. Objective: By the end of this training unit, the trainees will be able to

- Understand planes in CNC programming with driving tools;
- Understand the process of preparing, programming, simulating, machining, editing and optimizing in CNC turning programs;
- Setup the tool offset, spindle chuck data and work offset;
- Program with machining control tools on the front surface of the workpiece;
- Program with machining control tools on the cylindrical surface of the workpiece;
- Program for milling, drilling with C and Y axis;
- Understand the working of CNC lathes with reverse axis and how to calculate for programming;
- Program with main spindles on lathes with counter spindle;
- Use Mechanical Trade and Metal Handbook.

2. Content:

2.1. Planes when working with driven tools

2.1.1. G18

2.1.2. G17

2.1.3. G19

2.2. Setup the tool offset, spindle chuck data and work offset

2.2.1. Setup tools for main and counter spindle

2.2.2. Setup chuck dimensions for main and counter spindle

2.2.3. Setup jaw dimensions for main and counter spindle

2.2.4. Setup the work offsets for main and counter spindle

2.3. Programming of driven tools (DT) on the front surface

2.3.1. Drilling operations in G17 for main and counter spindle with C axis

2.3.2. Drilling operations in G17 for main and counter spindle with Y axis

2.3.3. Milling operations in G17 for main and counter spindle with C axis

2.3.4. Milling operations in G17 for main and counter spindle with Y axis

2.4. Programming of driven tools on the circumference surface

2.4.1. Drilling operations in G19 for main and counter spindle with C axis

2.4.2. Drilling operations in G19 for main and counter spindle with Y axis

2.4.3. Milling operations in G19 for main and counter spindle with C axis

2.4.4. Milling operations in G19 for main and counter spindle with Y axis

2.5. Applying of milling and drilling operations with C and Y-axis

2.5.1. Programming with Contour name (graphic editor)

2.5.2. Programming with Labels (G code)

2.5.3. Programming with Subprogram

2.5.4. Programming with Labels in subprogram

2.6. Working areas of machines regard to the frames and how to calculate them for programming

2.6.1. Working with two spindles

2.6.2. Working ares and their calculations

2.6.3. Commands required for programming of the counter spindle

2.7. Advanced programming of exercises with counter spindle and driven tools with the cycles and G-codes

2.7.1. Programming of coupling parameters, position control and synchronism

2.7.2. Calculations of counter spindle transfer preposition

2.7.3. Calculation of counter spindle transfer position for workpiece clamping

2.8. Plan and realise operational work order of the industry

2.8.1. Using programGUIDE with the cycles and G-codes for advanced turning application 1

2.8.2. Using programGUIDE with the cycles and G-codes for advanced turning application 2

2.9. Practice your English

Training Unit 3: ADVANCED DIN/ISO PROGRAMMING – CNC MILLING 3+2 AXES – SWIVELLING IN ACCORDANCE WITH DIN/ISO Time: 60 hours

1. Objective: By the end of this training unit, the trainees will be able to

- Explain function of swivel cycle (CYCLE800) and the advantages of this cycle
- Set and select control configuration in Sinutrain software for AC and BC kinematics for machining on 5-axis CNC milling machines
- Analyze detail drawings and customer's technical requirements
- Complete the work plan for each mechanical part in the order
- Calculate and select machining parameters for required mechanical parts
- Use mechanical manuals to look up relevant technical information
- Use swivel cycle (CYCLE800), G-code instructions and suitable cycles to program machining for advanced milling applications (3+2 axes)
- Optimize specified programming files.

2. Content:

2.1. Simplifying programming for 3+2 axes with programGUIDE
  2.1.1. General introduction
  2.1.2. Programming CYCLE800
2.2. Tool carrier of a 5-axis milling machine
2.3. Kinematics of 5-axis machines
  2.3.1. Rotary/swivel table
  2.3.2. Articulated rotary/ swivel table
  2.3.3. Fork head
  2.3.4. Articulated head kinematics
  2.3.5. Mixed kinematics
2.4. Functional explanation for the swivel cycle (CYCLE800)
2.5. Swivel modes by the programming
  2.5.1. Axis by axis
  2.5.2. Projection angle
  2.5.3. Directly
2.6. Advantages of programming with the CYCLE800
2.7. Advanced programming of exercises with swivelled planes, the cycles and G-codes
  2.7.1. The first programGuide exercise is to be programmed with the help of the swivel CYCLE800
  2.7.2. Complex milling exercise with 3+2 axes with programGUIDE
2.8. Plan and realise operational work order of the industry

2.8.1. Using programGUIDE with swivelled planes, the cycles and G-codes for advanced milling application 1 with 3+2 axes

2.8.2. Using programGUIDE with swivelled planes, the cycles and G-codes for advanced milling application 2 with 3+2 axes

2.9. Practice your English

Training Unit 4: CNC PROGRAMMING OF HIGH-LEVEL LANGUAGE FOR EXTREMELY DEMANDING CNC TURNING AND MILLING APPLICATIONS OF THE INDUSTRY

Time: 80 hours

1. Objective: By the end of this training unit, the trainees will be able to

- Develop configurable CNC programmes for component families that differ only in a few geometric sizes.
- Apply the higher level language such as R parameter, arithmetic parameters, system variables, user variables and loops for programs that require their own technology cycle because not all special tasks can be handled using standard cycles.
- Programme a simple operator message to be generated from an ongoing CNC program that cannot be fully implemented using the instruction set offered by DIN 66025 or graphical CNC programming.
- Develop comprehensive programmes logging of machining parameters and measurement results.
- Know and observe the regulations on IT security and occupational safety as well as health and environmental protection.
- Optimize specified programming files.

2. Content:

2.1. Introduction and overview of the basics and functionality

2.1.1. Rules for naming of a NC program

2.1.2. Elements of the NC-high level language

2.1.3. Value assignments

2.1.4. Programme the basic program using programGUIDE.

2.2. Subprogram

2.2.1. Working with subprogram

2.2.2. Stucture of subprogram

2.2.3. Commands M17, RET, M30

2.2.4. Programme the basic program using programGUIDE.

2.3. R Parameter

2.3.1. Predefined arithmetic parameters

2.3.2. R variable requirements

2.3.3. Calculated parameters

2.3.4. Calculated R parameter displayed by message text

2.3.5. Programme the basic program according to technical drawings.

2.4. Jumps and user variables

2.4.1. Explanation of variables
2.4.2. Explanation of jumps
2.4.3. Jump commands GOTO/ GOTOF/ GOTOB
2.4.4. Programming of error queries and error loops.

2.5. System variables and subprogram
2.5.1. Function of system variables
2.5.2. Specific notation by the machine manufacturer
2.5.3. Programme the program using system variables and subprogram.

2.6. System variables and safety query
2.6.1. Commands
2.6.2. Function of system variables and safety query
2.6.3. Programming with system variables, tool queries and error loops.

2.7. Frame concept
2.7.1. Function of frames
2.7.2. Explanation of the commands and variables used
2.7.3. Classification of the G-functions into so-called function groups
2.7.4. Programming with R parameters, user and system variables, frames

2.8. Programming of arithmetic functions
2.8.1. Types of variables
2.8.2. Function of variables
2.8.3. Programming of arithmetic functions

2.9. User cycles
2.9.1. Configuration and management of a cycle on SINUMERIK
2.9.2. Project work: Writing of an own drilling cycle step by step

2.10. Plan and realise operational work order of the industry
2.10.1. Use programGUIDE with cycles and G-codes to program the machining of a simple drill piece using R parameters, user variables, system variables, loops, calculations, etc.

2.11. Practice your English

Training Unit 5: MACHINE INTEGRATED MEASURING OF WORKPIECES AND TOOLS WITH 3D TOUCH PROBES Time: 40 hours

1. Objective: By the end of this training unit, the trainees will be able to

- Understand the structure and operating principle of measuring devices used when measuring workpieces and products on CNC milling machines;
- Repair the instruments for high accuracy before measuring;
- Measure the dimensions of various types of part shapes in manual and automatic modes on CNC milling machines;
- Measure the position of common workpiece standards before machining;
- Practice well in the work of redefining the standard and accuracy of CNC milling machines;
- Export the measurement result report as text;
- Use Mechanical Trade and Metal Handbook.

2. Content:

2.1. Measuring instruments for measuring workpieces

2.1.1. Introduction to workpiece measurement with SINUMERIK

2.1.2. Measuring instruments for measuring workpieces

2.2. Calibration of measuring instruments for measuring accuracy

2.2.1. Required measuring equipment

2.2.2. Method of calibration

2.2.3. Define the measuring probe in the tool list

2.2.4. Adjust the length and radius of the probe

2.2.5. Measurements conditions with touch probe

2.3. Measuring in the Setup mode

2.3.1. Modes information

2.3.2. Setup in the JOG mode

2.3.3. Measure workpieces in the Setup mode

2.4. Measuring in Automatic mode

2.4.1. Workpiece measurement in AUTOMATIC mode

2.4.2. Overview of the measurement functions in AUTOMATIC

2.4.3. Measure workpieces in Automatic mode

2.5. Measuring results through logging

2.6. Log measurement results with using the high level language (HLL)

IV. Requirements for module implementation

1. Vocational classroom/ Training workshop

a. Vocational Classroom:

- Offers barrier-free access and workplaces, complies with occupational safety regulations, technical directions, and legal regulations.
- Provides sufficient workspace for the number of trainees as well as operational IT infrastructure with PC workstations with appropriate programming and simulating softwares and internet connection

b. Training workshop:

- CNC cutting workshop and machine tool workshop with 2-axis and 3-axis CNC lathes, 3-axis and 5-axis CNC milling machines as well as internal and external tool measuring equipments
- Offering barrier-free access and workplaces, complies with occupational safety regulations, technical directions, and legal regulations.
- Providing sufficient workspace and machine workplaces for the number of trainees
- Having a warehouse for raw materials
- Having a warehouse for semi-finished and finished products
- Barrier-free toilet rooms, bathrooms, and changing rooms, separately for women and men.

2. Equipment and machinery

a. Permanently installed conventional machine tools (including standard accessories and tools)

- Sawing machines
- Grinding machines (2 – wheel and belt grinders)

b. CNC machines and equipment (including standard accessories and tools)

- 2-axis and 3-axis CNC lathes, measuring probes and measuring device for cutting tool parameters integrated in the machines.
- 3-axis and 5-axis CNC milling machines with AC or BC kinematics, measuring probes and measuring device for cutting tool parameters integrated in the machines.
- Preliminary cutting tool parameter measuring device.

3. Teaching and learning materials, tools, consumption materials

a. Teaching and learning materials:

- Detailed drawings, detail assemblies and general drawings
- Assembly instructions, maintenance and repair plans
- Machining plan, arrangement, and work plan

- Table of safety data
- Table of characteristic values, measurement checklist, assessment record
- Textbooks, reference manuals
- Pocket calculator, drawing materials.

b. Tools:

- Analog measuring instruments, digital display and precision measuring instruments
- Measuring gauges
- Hand – operated tools.

c. Support materials:

- Cooling and lubricating materials
- Lubricating and cutting oils
- Detergents

d. Safety equipment:

- Personal protective equipment (PSA): safety clothing, safety shoes, goggles and noise protection.

e. Consumption materials:

- Materials needed for machining parts are compatible with practice exercises, orders and exams.

4. **Other requirements and conditions:** None

V. Assessment content and methodology

1. Assessment content:

- **Knowledge:**
  - + Know the regulations on information technology security, labor safety, health protection and environmental protection when working with CNC lathes and milling machines.
  - + Describe the structure, function and safety devices of CNC lathes and milling machines.
  - + Understand the operation of CNC lathes and milling machines and applicable safety regulations.
  - + Know how to program with ProgramGUIDE on CNC lathes and milling machines.
  - + Specify input and output data devices according to the intended use

- **Skills**
  - + Program and test programs by DIN/ISO on CNC lathes and milling machines (programmable in the workshop) and on the computer.
  - + Import, test, modify and optimize programs on CNC lathes and milling machines.
  - + Set and select control configuration in simulation software for machining on CNC lathes with C and Y axes.
  - + Set and select control configuration in simulation software for AC and BC kinematics for machining on 5-axis CNC milling machines.
  - + Transfer and secure data in accordance with legal and business regulations.
  - + Adjust, operate and maintain CNC lathes and milling machines.
  - + Select, prepare, mount and adjust of clamping tools and fixtures.
  - + Adjust and clamp workpieces.
  - + Select, mount and measure parameters of cutting tools and standard workpieces.
  - + Define, control and optimize machining methods and steps.
  - + Define and enter production parameters with attention to workpieces, workpieces, tools, and cutting tool materials.
  - + Identify and fix errors in production.
  - + Systematic search and eliminate the causes of poor quality defects.
  - + Identify and correct safety deficiencies and malfunctions on the machine.
  - + Perform and document maintenance work, preventive maintenance for CNC lathes and milling machines as planned.
  - + Troubleshoot basic mechanical and electrical problems or plan for diagnosis.

- **Autonomy and responsibility:**
  - + Observe and apply general regulations of occupational safety, health, fire and environmental protection (observation, checklist with 90% correct answers)
  - + Analyse work orders of internal and external customers and to assess the technical and economic feasibility in compliance with safety and environmental protection regulations
  - + Take into account customer-specific requirements and deadlines and arrange for partial orders
  - + To research and evaluate information for job planning in digital networks
  - + Initiate, monitor and control partial orders
  - + To hand over products to external customers or to the subsequent production area (internal customer) and to present work results also with the help of digital media
  - + To assume responsibility in the manufacturing process and to be aware of product liability in the context of business relations with customers.
  - + To communicate and cooperate in interdisciplinary teams
  - + To use energy and material under economic and environmental aspects and to dispose of materials and substances in an environmentally friendly way
  - + Ensure learning time and learning creativity (observation, checklist). To participate actively in the lessons (more than 80% in theoretical and 100% in practical lessons).

2. Assessment methods

The assessment is based on the project work carried out and products manufactured by the trainee/learner and is carried out in accordance with the provisions on the minimum knowledge, skills, autonomy and responsibility required for module.

- **Knowledge:**

  The trainee's/learner's skills and behaviour are determined based on oral and written tests such as quizzes, technical discussions, and multiple-choice questions, as well as through integrated theory - practice exercises or practical exercises during the implementation of the teaching units of the module. The evaluations are calculated according to the valid point rules.

- **Skills:**

  On the basis of practical exercises, project work and company work orders, the practical performance of the trainee/learner is assessed with regard to the following criteria with the aid of evaluation sheets/scales:

  + Occupational safety
  + Organization of the workplace
  + Technical standards
  + Planning and implementation
  + Target time
  + Self - assessment

- **Autonomy and responsibility:**

  Following attitudes and characters of the trainees/learners are determined and evaluated by observation over the entire training period: work, learning and cooperation ethics, regulation and regulation morale, diligence, conscientiousness, discipline, ability to work in a team, punctuality, independence, sense of responsibility, prudence, initiative, active participation in lessons and support/motivation of others in the steering process.

VI. Guidelines for professional module implementation:

1. Scope of application

- Training module for higher Industry 4.0 qualification – International College Level. Admission requirement: National College Diploma in Metal Cutting.

2. Guidelines on teaching and learning methods

- For teachers, lecturers and in-company trainers: The responsible teachers (at the TVET institute)/in-company trainers (in the company) observe the following guidelines for the professional implementation of the theory lessons and practice training of the whole module and each single training units:
  - + The trainees/learners shall be instructed in detail and comply with regulations on occupational safety, health and environmental protection as well as fire protection and security. Compliance with the regulations must be continuously monitored by the responsible teacher/responsible In-company trainer. The trainee/learner is to be expressly informed and made aware of the appropriate measures and consequences of non-compliance with the regulations.
  - + The learning process and learning progress of the trainees/learners shall be continuously monitored and regularly evaluated, in particular the consistent compliance with occupational safety regulations and environmental protection conditions.
  - + To ensure the highest possible quality of teaching and training through the reference to the corresponding teaching unit in the planning and implementation of lessons.
  - + Within the framework of the practice training units, the necessary work steps shall be carefully explained to the trainee/learner and demonstrated correctly.
  - + The personal level of knowledge and skills shall be checked and assessed individually for each practical teaching unit on the basis of regular work reports drawn up by the trainee.
  - + The quality of teaching is increased and ensured by the increased use of different teaching and learning methods such as the 4-step method, project method, guiding text, self-study and group work as well as by the efficient use of teaching and learning materials and other aids.
  - + The work results of the trainees/learners are to be evaluated and discussed transparently and together with the trainees/learners by the responsible vocational teacher or by the company trainer.
- For trainees/learners: The trainees/learners are instructed to
  - + Strictly follow the instructions of the vocational teachers or the company trainers

+ Participate regularly and actively in the lessons and each teaching unit of the training module
+ Observe the regulations on occupational safety and health, fire and environmental protection
+ Actively contribute to environmental protection
+ Observe the teaching and workshop regulations
+ Participate attentively in class, take notes and ask questions in case of uncertainty
+ Ask questions to the vocational school teachers or the company trainers or to other trainees/learners to ask for support with difficult tasks and to name problems
+ Prepare the workplace and keep it clean and tidy
+ Prepare, properly handle and maintain the equipment
+ Prepare daily and weekly work reports on the theoretical and practical lessons of the module.

3. Aspects to be considered

- The training focus of the training module is on training units: 1, 2, 3, 4 and 5.

4. Reference materials

- Metal Engineering, Young Publishing House, Vietnam
- Mechanical and Metal Trades Handbook (Tabellen Buch Metall), 3rd English Edition, EUROPA –LEHRMITTEL, Germany.
- Chau Manh Luc, Textbook of machining technology on CNC machines, Danang University of Science and Technology.
- Tran The San, Nguyen Ngoc Phuong, CNC Programming Manual, Da Nang Publishing House.
- Tran Van Dich, Technology on CNC machines, Hanoi Science and Technology Publishing House.
- Ta Duy Liem, Bui Duc Anh, Phan Van, Le Duc Bao, Fundamental of CNC machines, Hanoi Bach Khoa Publishing House.
- Training materials related to Sinumerik Operate 840Dsl, Siemens Corp., 2019.

5. Further notes and explanations (if any)

B. TRAINING UNITS

UNIT 1: FUNDAMENTAL DIN/ISO PROGRAMMING – CNC – TURNING AND MILLING PROGRAMMING BASICS

1. **Introduction:** As the first unit of the module, the trainees will be familiar with programming method using programGUIDE to solve applications at a basic level.
2. **Objective:** By the end of this training unit, the trainees will be able to

- Present the required programming parameters of the integrated machining cycles in the Sinumerik 840Dsl Operate version 4.8
- Analyze detail drawings and customer's technical requirements
- Complete the work plan for each mechanical part in the order
- Calculate and select machining parameters for required mechanical parts
- Use mechanical manuals to look up relevant technical information
- Use G-code instructions and suitable cycles to program machining for specific applications
- Optimize specified programming files.

3. **Content:**

3.1. Program management and basic operations

3.1.1. Main screen

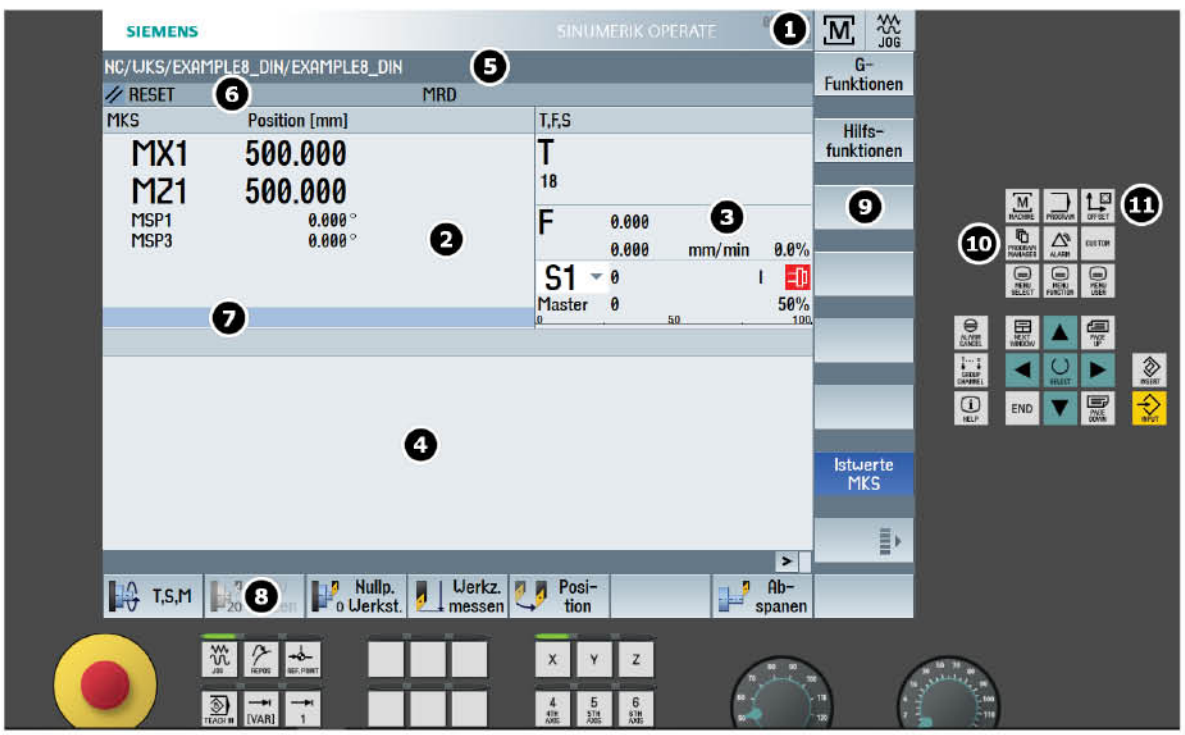
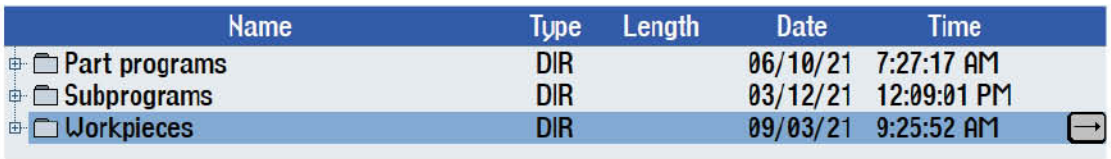


Figure 1.1: Main screen of Sinumerik Operate

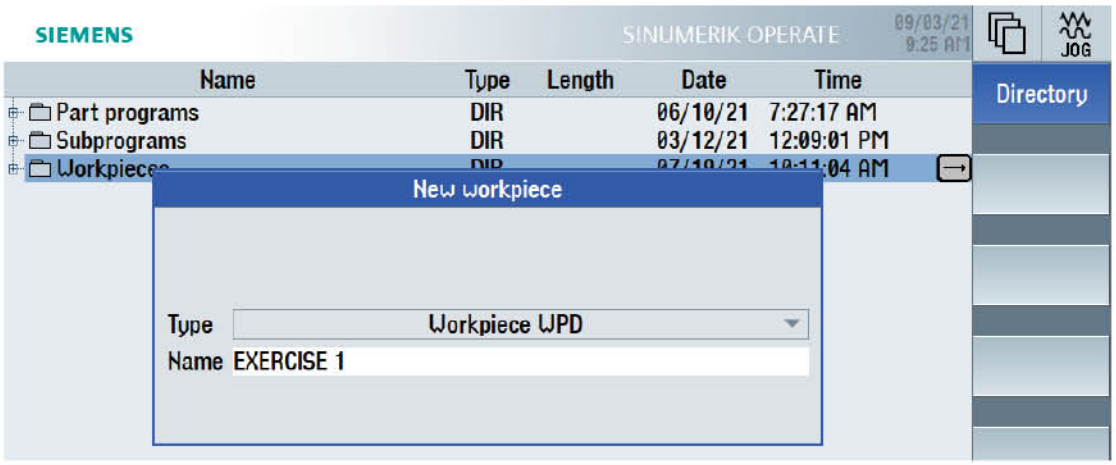
1. Active operating area and mode
2. Position readout for the axes
3. Display of:
    a. T = active tool
    b. F = present feedrate
    c. S = spindle
    d. Spindle load factor in percent
4. Working window and program display
5. Program path and name
6. Status, program influence and channel name
7. Display of the active zero point, rotation, mirroring and scaling
8. Horizontal softkey bar (HSK)
9. Vertical softkey bar (VSK)
10. Program manager
11. Tool and zero point offset

3.1.2. Program manager (part programs, workpieces, subprograms)

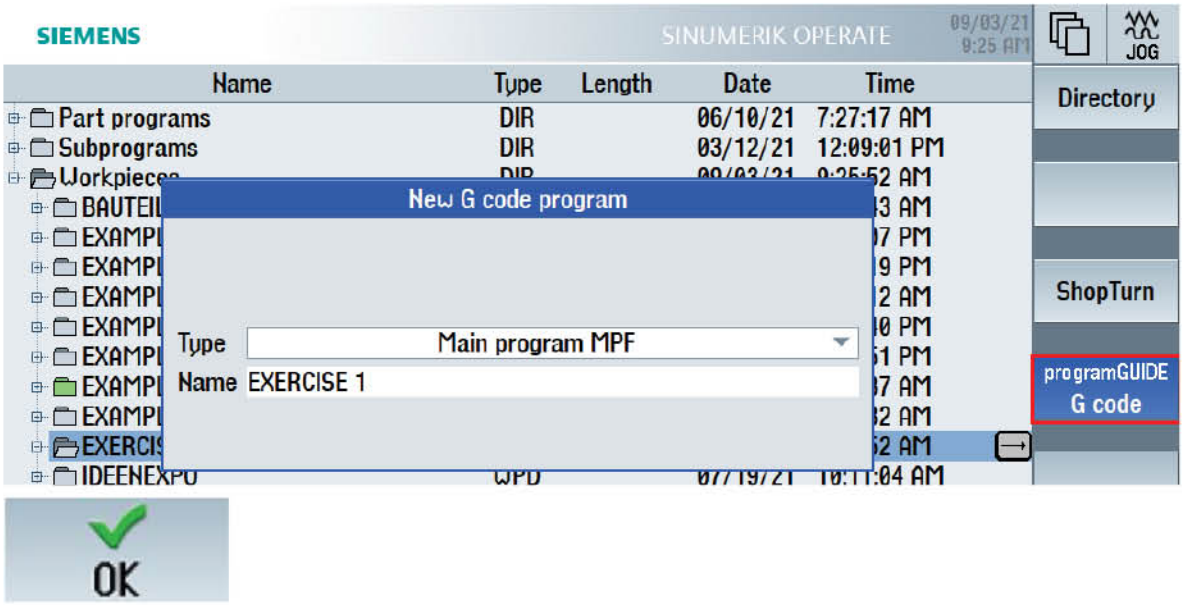
To input a new sequential program in DIN/ISO with programGuide select a drive where you want to create a new program (NC, USB, Local drive)



Here we create a new workpiece directory and accept with VSK8 “OK”



Choose “programGuide G code” with VSK4 and input a new program name as MPF



3.1.3. Sofkey bars

a. Horizontal softkey bar (HSK)

The user interface consists of different subsections. At the bottom of the screen is the horizontal softkey bar (HSK) containing 8 softkeys. The selection of a new window is made by pressing the buttons just under the softkeys.

If the number of functions exceeds the representation capacities of the maximum of 8 softkeys than a partitioning in two different horizontal softkey bars occurs.

The change over forth and back takes place with the “Menu extend“-key on the operator panel.

b. Vertical softkey bar (VSK)

The available functions and operating modes can be selected from the keys right beside the vertical softkey bar (VSK) on the right hand side of the screen. If the number of functions exceeds the representation possibility capacity of the maximum of 8 softkeys than a partitioning in two different vertical softkey bars occurs.

The changeover takes place with the:

“Forward“-key	
“Backward“-key (VSK 8)	

3.1.4. Basic operations of programming (program header, blank definition, tool call, technology blocks, machining steps)

Certain principles should be followed during the creation of a G code program with programGuide:

The program should exhibit a clear and concise structure and should contain comments wherever possible to ensure that other users can understand the layout easily at later stages.

The following flow chart represents a possible suggestion for a suitable structure of the main program:

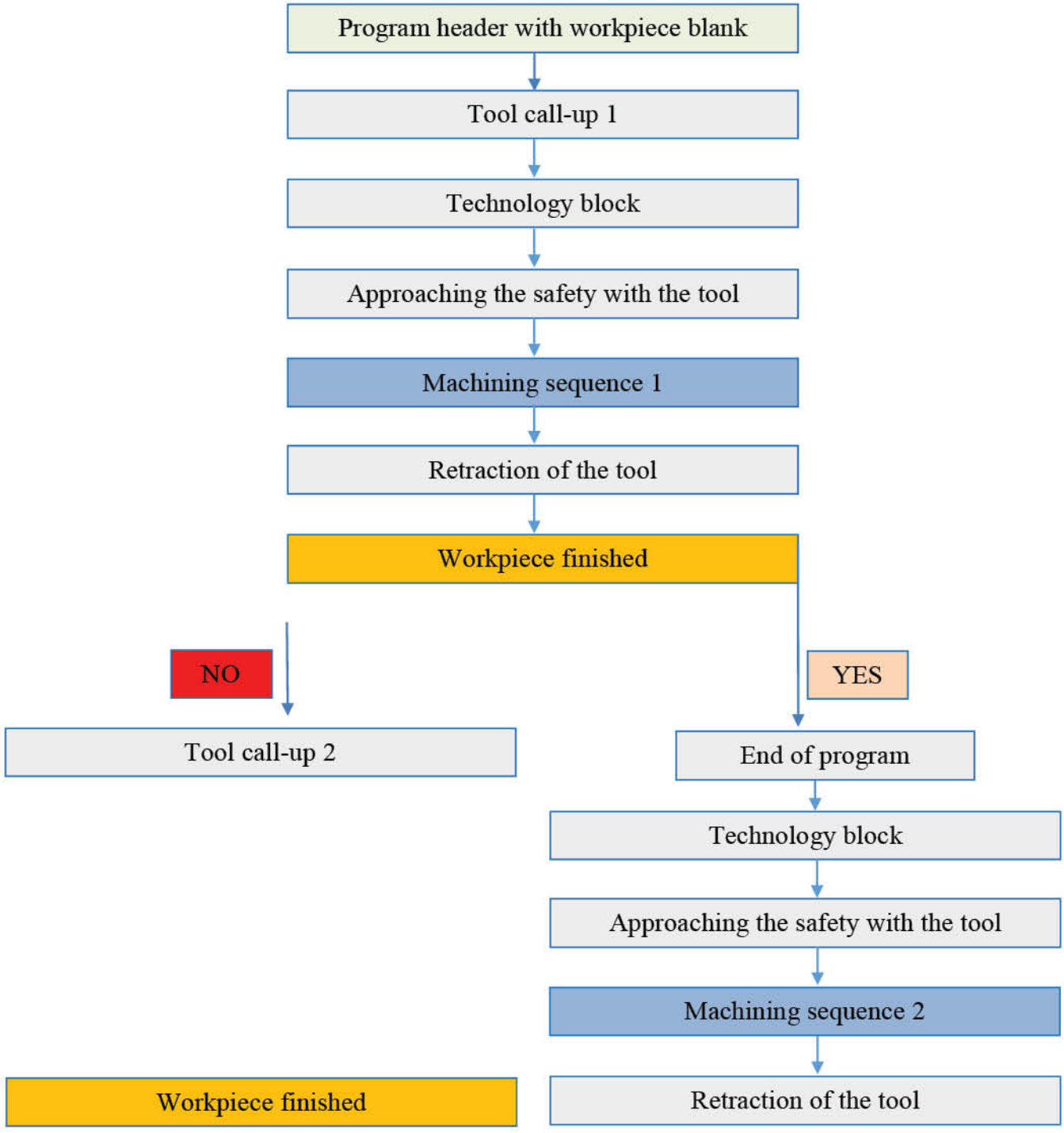


Chart 1.1: Structure of main program

The general structure of a G code program can be programmed freely. For a good legibility however, the following structure is recommended:

1. Zero point selection, plane selection, absolute dimensioning
2. Blank attribution for the simulation
3. Tool call-up and tool change
4. Technology data, path commands
5. Programming of the technologies (cycles)
6. Program end

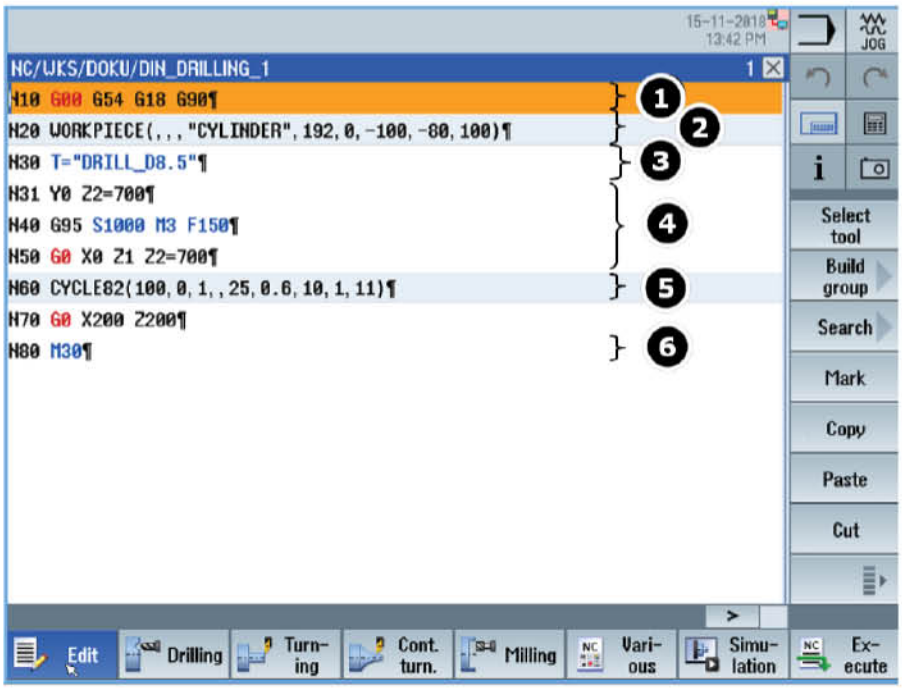


Figure 1.2: General structure of a G code program

If the programmer wants to program with the DIN/ISO functionality, you can generate the programs with G code commands in the ShopTurn or ShopMill user interface.

G code commands can be programmed as per DIN 66025.

With the G code programming with the programGUIDE, parameter masks guide you in the process of measuring, programming of contours, as well as drilling, turning and milling cycles.

From within the parameter masks, G code will be generated, which can also be translated back into the parameter masks to change and optimize the current program.

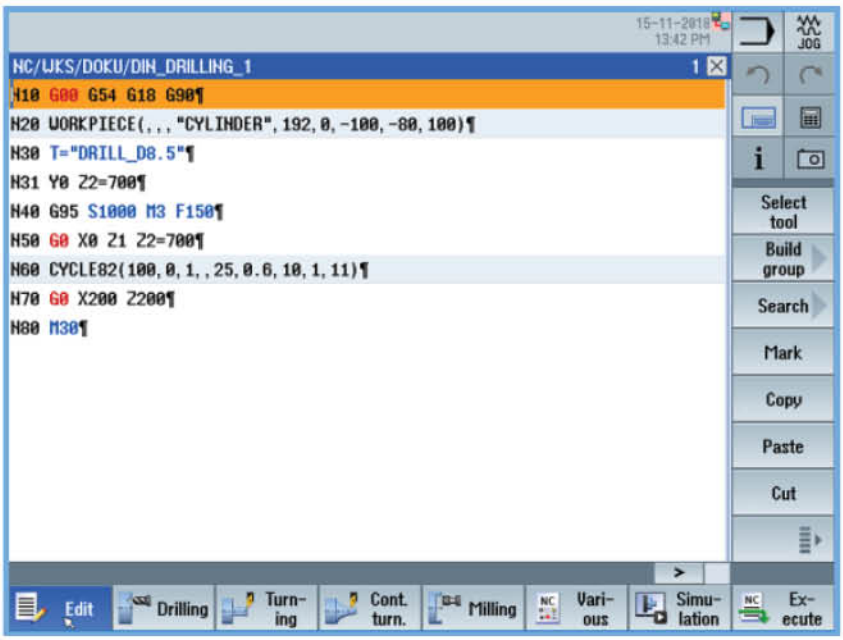
The following functions simplify the programming of G code programs:

- Technology oriented program step selection (cycles) using softkeys
- Input masks and windows for parameters, with animated help graphics

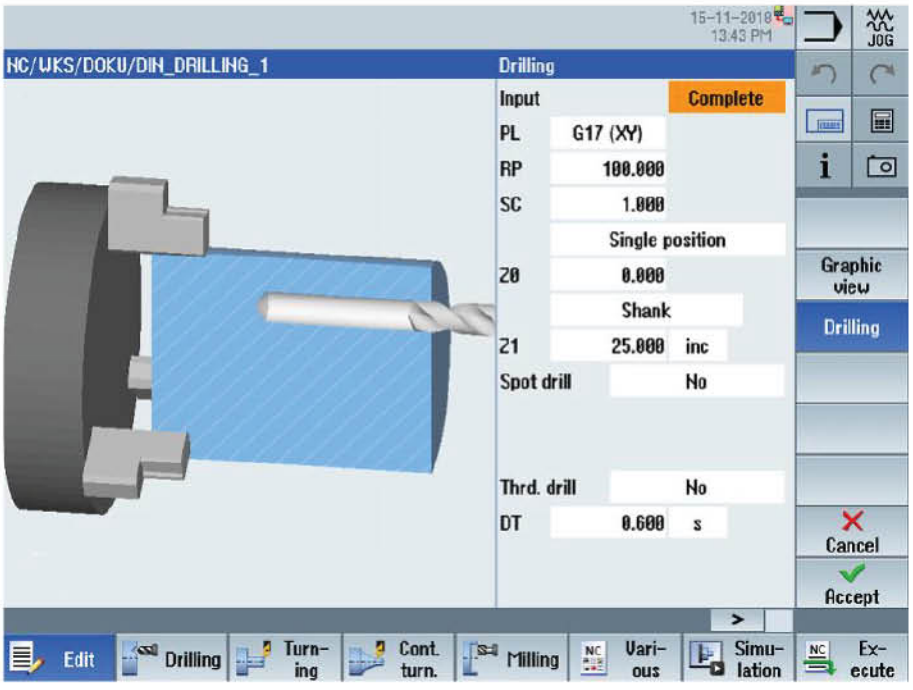
- Context sensitive online help for every input mask and window
- Comfortable definition of the blank
- Support for the contour input with the geometric processor (optional as G code)

ProgramGuide programs can be represented in different views:

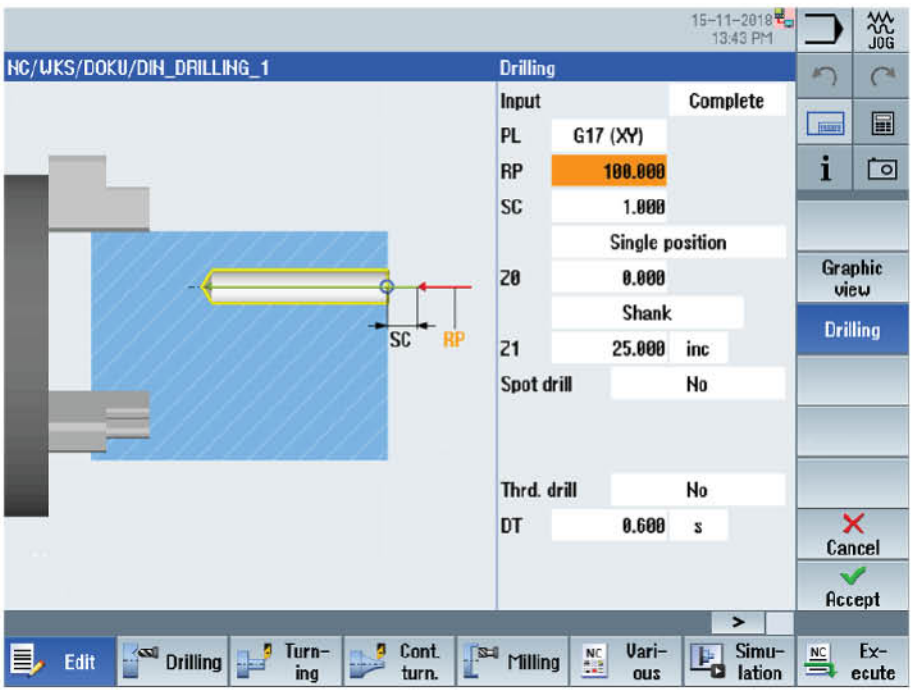
- As a work plan showing the call up of the tool, path commands, revolution settings, spindle data, feed, cycle calls, the program end, etc.



- As a parameter mask, with 3D animations during parameter input of cycles and blank definition.



- As a parameter mask, with help graphics during parameter input of cycles and blank definition.



3.2. Operation area for parameter offset

3.2.1. Administration of tool offset

In the Operation area “Parameter” you have a choice of selecting between various lists (e.g. Tool list, tool wear, magazine list, offsets, user variables and setting data).

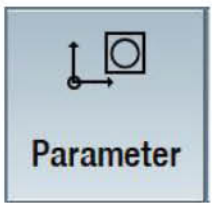
For example, in the tool management area, all tools and if configured also the magazine locations are being displayed.

Both lists display the same tools in the same order. When switching between lists, the position of the cursor on a particular tool in the current screen is carried over to the same tool in a new screen. The lists differ from each other by the displayed parameters and the Softkey functions

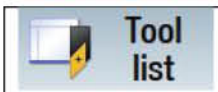
Switching between lists is a specific change from one topic to the next.

- Tool list: All parameters and functions required to create and set up tools are displayed.
- Tool wear: All parameters and functions that are required during operation, e.g. wear and monitoring functions, are listed here.
- Magazine: Magazine and magazine location-related parameters and functions for the tools and magazine locations are listed here.

The operating area “Parameter“ can be opened from every operating modes (“JOG”, “MDA”, “AUTO”).

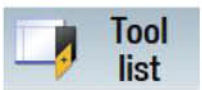
	Press the “OFFSET”-key on the keyboard. The operating area “Parameter“ respectively the “Tool list” opens directly.
	The operating area “Parameter” opens, with the “Tool list”, “Tool wear”, “Magazine”, “Work offset”, “User variables” and the “Setting data”. These functions are made available in the following described horizontal softkey bar.

Horizontal softkey bar (HSK) of the operating area “Parameter”

	By pressing the HSK 1 “Tool list” the tool list window opens for creating new tools or cutting edges (D1 – D9), delete tools, load or unload.
	By pressing the HSK 2 “Tool wear“ the tool wear list opens. Here it´s possible to input correction values of tool wear and “Tool data”.

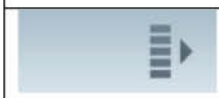
Maga-zine	By pressing the HSK 4 “Magazine“ the Magazine management is opened. Here it´s possible to delete, load or unload all tools at the same time in one work step
Work offset	By pressing the HSK 5 “Work offset“ a list with all Work offsets is opened. This window is generally used for monitoring: • Active work offset • Work offset overview • Base offset (G500) • Settable zero offset (Work offset - G54...G57)
R User variable	By pressing the HSK 6 “User variable“ a list with all R variables is opened.
SD Setting data	By pressing the HSK 8 “Setting data” a list with all setting data is opened. Working area limit, spindle data and spindle chuck data are required set up here.

Vertical softkey bar (VSK) of the “Tool list” in the operating area “Parameter”



Tool measure	By pressing the VSK 1.1 “Tool measure” the “Measuring tool” window opens. With this function the correction values of tool can be determined.
New tool	By pressing the VSK 1.2 “New tool” a new tool can be created. This function is the cursor is positioned on a field that does not yet has a tool assigned to it.
Edges	By pressing the VSK 1.3 “Edges” the vertical softkey bar for assigning new cutting edges and deleting existing cutting edges opens to the right side of the screen. If a tool has several cutting edges each edge gets its own set of correction data.
Further data	By pressing the VSK 1.4 “Further data” more information about a tool will be displayed. This function is only available for tools which have additional information.
Unload Load	By pressing the VSKs 1.5 ”Unload” or “Load” the actual selected tool will be unloaded from or loaded to the magazine. Unloaded tools are being displayed on the bottom of the magazine list.

Delete tool	By pressing the VSK 1.6 “Delete tool” the selected tool will be deleted from the tool list.

Magazine selection	By pressing the VSK 1.7 “Magazine selection” Ssoftkey multiple times you can jump between buffer location (spindle and gripper), Magazine and NC-memory (unloaded tools) and back to the buffer location. The cursor is always positioned at the beginning of each group.
	By pressing the VSK 1.8 “Extend” on the operator panel the extended vertical softkey bar 2 opens on the right hand side of the screen.

Vertical softkey bar (VSK) of the “Extended” in the “Tool list”

Sort	By pressing the VSK 2.1 “Sort” you can sort the tools in the tool list according to the following criteria: •Magazine •Name •Type •T-Number The suitable softkeys are offered in the vertical softkey bar.
Filter	Pressing the VSK 2.2 “Filter” opens the screen to set the filter options.
Search	Pressing the VSK 2.3 “Search“ opens a new softkey bar with the functions • Tool • Magazine location • Empty location The type of search is selected by the sotfkey selection.
Details	Pressing the VSK 2.4 „Details“ opens a new softkey bar with the functions • Tool Data • Cutting edge data • Monitoring data The details of the tool which is selected by the cursor position are listed in the screen.
	Pressing the VSK 2.8 “Back“ on the operator panel you switch back to the vertical softkey bar 1.

3.2.2. Administration of the work offset

Following reference point approach, the actual value display for the axis coordinates is based on the machine zero (M) of the machine coordinate system (MCS).

The program for machining the workpiece, however, is based on the workpiece zero (W) of the workpiece coordinate system (WCS). The machine zero and workpiece zero are not necessarily identical.

The distance between the machine zero and workpiece vary in accordance with the type of tool and the way it is clamped. This zero offset is taken into account during execution of the program and can be a combination of different offsets.

On the Sinumerik Operate, the position actual value display refers to the SZS-coordinate system (settable zero system). The position of the active tool relative to the workpiece zero is displayed.

The offsets are added as follows:

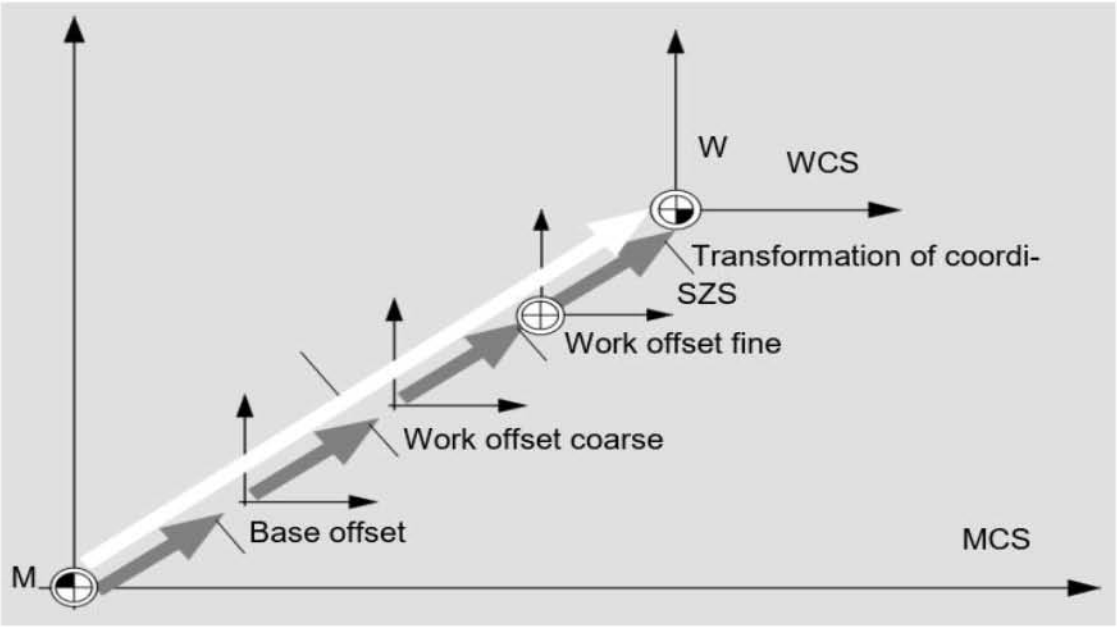


Figure 1.3: Types of offsets

- Base offset:

The base offset is a zero offset that is always active. If you have not defined a base offset, its value will be zero. You determine the base offset via "Measure workpiece zero".

- Zero offsets:

Every zero offset (G54 to G57, G505 to G599) consists of a coarse offset and a fine offset. You can call the work offsets from any sequence program (coarse and fine offsets are added together).

You can save the workpiece zero, for example, in the coarse offset, and then store the offset that occurs when a new workpiece is clamped between the old and the new workpiece zero in the fine offset.

- Coordinate transformations:

You always program coordinate transformations for a specific sequence program. They are defined by:

- Offset
- Rotation
- Scaling
- Mirroring

These transformations can work as "new" or they can work "additive" to the active zero point offset.

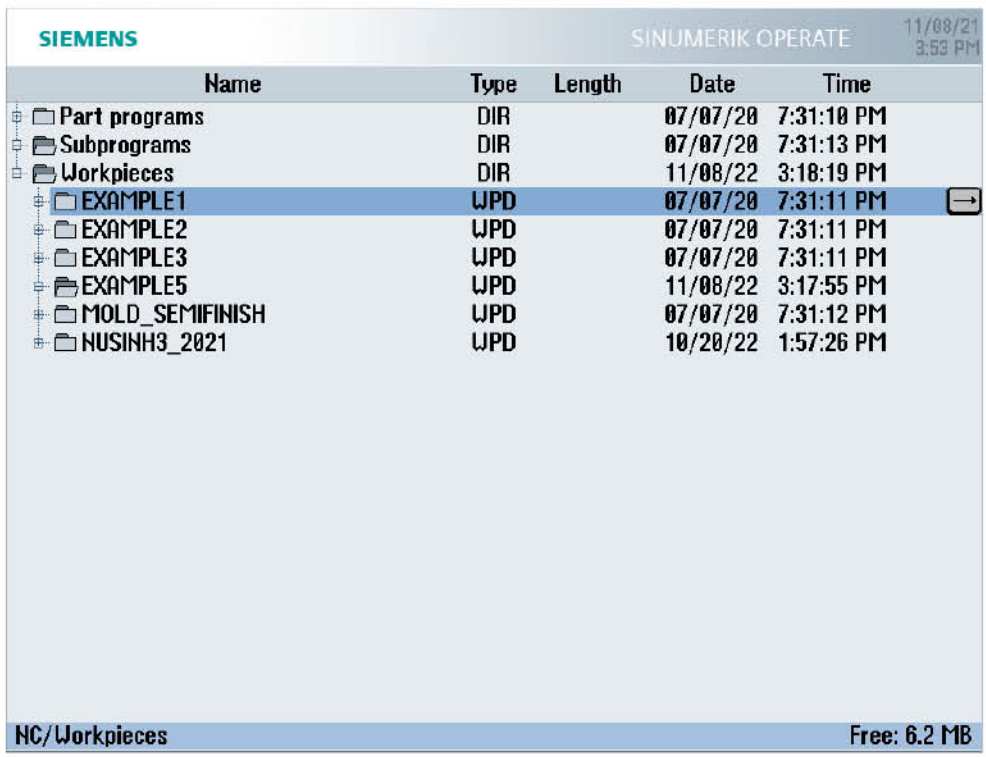
- Total offset:

The total offset is calculated from the sum of all offsets and coordinate transformations.

3.2.3. Archiving setup data

Setup data contains tool data, magazine assignment and zeropoints, it is used to recover lost data by user/ operator or the operator just wants to use restored setup data of orders of customers again.

To achieve setup data of specific part program, we do the following steps:

1.	Open **Workpieces** folder and choose the part program to create archieve, inhere we choose **EXAMPLE1** 

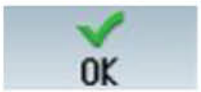
2.

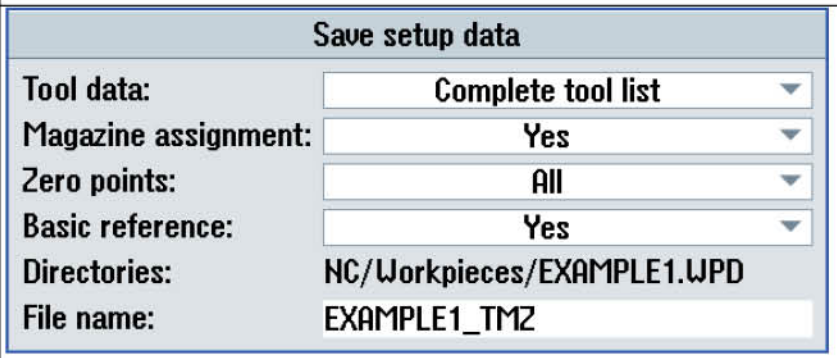
Press the VSK 2.1 “Archieve”.

The work area with the functions of “Archive” opens.

Press the VSK 3 “Save setup data”, to open the input mask for “Save setup data”.

3.



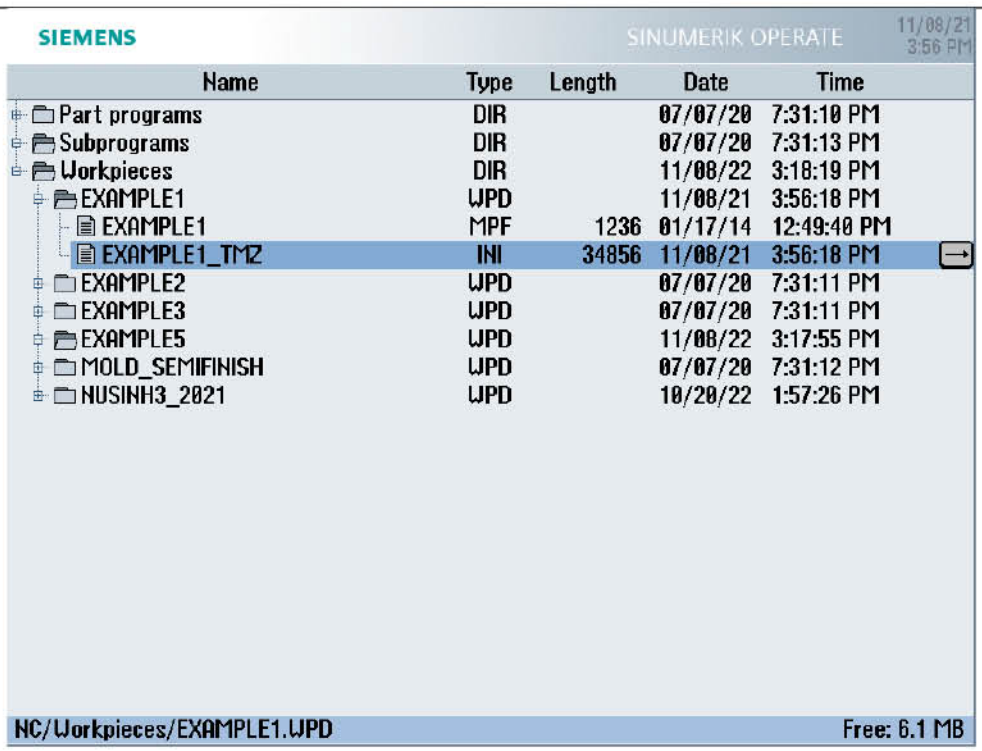


Choose the following values into the input mask

Confirm your inputs by pressing the VSK 8 “OK”.

Note: Directories for saving file and name of file, inhere we have **EXAMPLE1_TMZ**

4.



In **EXAMPLE1** folder, we have 2 files: one file of original part program (**EXAMPLE1.MPF)** and the other of setup data (**EXAMPLE1_TMZ.INI)**. And

	then, for IT sercurity policy with customers we can copy and save these files on secured disk or directories of machine memory.

3.3. General basics for programGUIDE

3.3.1. Overview G codes with linked addresses

G0	X… Y... Z…	rapid traverse command
G1	X... Y... Z...	linear interpolation in feed motion
G2	X… Y... Z... I... J... K... oder CR...	circular interpolation counter clock wise
G3	X... Y... Z... I... J... K... oder CR...	circular interpolation clock wise

CR=	circular radius until 180° positive over 180° negative
ANG=	angle absolute
RND=	rounding radius to the next contour element
CHR=	chamfer to the next contour element (width of chamfer)
CHF=	chamfer to the next contour element (length of chamfer)
AR=	opening angle absolute
X/Y/Z=AC(…)	absolute coordinate programming
X/Y/Z=IC(…)	incremental coordinate programming
G110	pole coordinates refer to the last position reached
G111	pole coordinates refer to the zero point of the current WCS
G112	pole coordinates refer to the last valid pole
AP=	polar angle
RP=	polar radius
G40	deactivate tool radius compensation (TRC)
G41	activate TRC with machining direction left of the contour
G42	activate TRC with machining direction right of the contour
G90	absolute dimensions
G91	incremental dimensions
G500	deactivation of the current settable zero offset
G53	suppresses the settable zero offset and the programmable zero offset non-modally

G153	has the same effect as G53 and also suppresses the entire basic frame
G54 - G57	settable zero offset
G17	Working plane X/Y
G18	Working plane X/Z
G19	Working plane Y/Z
G97 S...	constant spindle speed in min^{-1}
G96 S...	constant cutting speed in m/min
G94 F...	linear feedrate [mm/min
G95 F...	revolutional feedrate in mm/revolution
G95 FZ=…	revolutional feedrate per tooth in mm/rev
G147	approach with a straight line
G148	retraction with a straight line
G247	approach with a quadrant circle
G248	retraction with a quadrant circle
G347	approach with a semicircle
G348	retraction with a semicircle
DISR=	1. For approach and retraction with straight lines (G147/G148): distance of the cutter edge from the starting point of the contour 2. For approach and retraction with circles (G247, G347/G248, G348): radius of the tool centre point path

3.3.2. Tool call-up

T="…"	tool name
D	cutting edge number (D1 – D9)

3.3.3. Overview M codes

M00	Programmed Halt
M01	Optional stop, see also M00
M02	End of Program
M03	Spindle Start clockwise
M04	Spindle Start anti-clockwise
M05	Spindle Stop

M06	Tool change
M08	Coolant ON
M09	Coolant OFF
M19	Defined spindle stop
M30	End of Program (see also M02)

3.3.4. Overview high level language commands

SUPA	has the same effect as G153 and also suppresses all other offsets (hand wheel, overlaid movements, external zero offset, PRESET offset)
TRANS	Absolute offset of the WCS with reference to the workpiece zero
ATRANS	Additive zero offset of the WCS with reference to the parameterized workpiece zero set with TRANS
MIRROR	mirror absolute in relation to the currently valid coordinate system
AMIROR	additive mirror image with reference to the currently valid set or programmed coordinate system
ROT RPL=	absolute rotation
AROT RPL=	additive rotation
SCALE	scale up/down absolute in relation to the currently valid coordinate system
ASCALE	scale up/down additive in relation to the currently valid set or programmed coordinate system
OFFN=	allowance on the programmed contour
OFFN=0	deactivate allowance on the programmed contour
MCALL L...	modal subprogram call
MCALL	deactivate modal call
SPOS[1]=0	set spindle to specified angle (0°)
SETMS(1)	set spindle <n> as master spindle (here main spindle)
SETMS(2)	set spindle <n> as master spindle (here spindle of driven tool)
TRANSMIT	activate face end transformation
TRACYL(32)	activate cylinder surface transformation (here Ø32mm)
TRAFOOF	deactivates all active transformations (TRANSMIT, TRACYL)
R=	variable for parameter programming

CFC	constant feedrate at the contour (tool cutting edge)
CFTCP	constant feedrate on the milling cutter center path
CFIN	constant feedrate at the tool cutting edge only at concave contours, otherwise on the milling cutter centre path - the feedrate is reduced for inside radii
LAL:	jump marker
REPEAT	repeat program section between jump marker (he start of the program section to be repeated)
REPEATB	the program line to be repeated
DIAMON	diameter programming
DIAMOF	diameter programming off
DIAM90	diameter programming for G90 - radius programming for G91

Command	Destination point of X coordinate	
	with G90	with G91
DIAMON*	Diameter	Diameter
DIAMOF**	Radius	Radius
DIAM90	Diameter	Radius

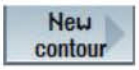
* switching-ON condition for turning machines

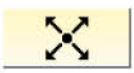
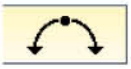
** switching-ON condition for milling machines

3.4. Contour definition and contour call in the program

3.4.1. Contour definition

a. Define contour using graphic editor

To use graphic editor to define contour elements (line, arc, circle etc.), we press VSK1 

and then enter the new name of contour. In the interface of graphic editor, we use relevant VSKs (straight line X), (straight line Y),  (straight XY),  (Circle) to draw each element of contour.

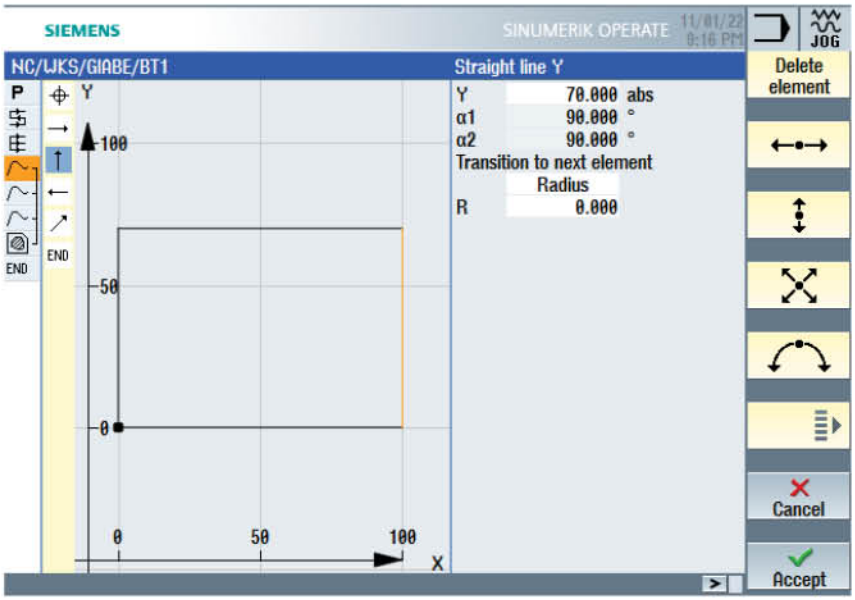


Figure 1.4: Detail contour created by graphic editor

b. Define contour using labels (G code)

Contour **can** be defined by using function labels, elements of contour are programed by G codes and always begin by G0.

```
N166 BORDER_ST:
;attention: allways first NC-block with G0
N167 G0 X0 Y0
N168 G1 Y90
N169 G1 X120
N170 G1 Y0
N171 G1 X0 ;contour must be closed
N172 BORDER_END:
```

Figure 1.5: The programmed contour is limited by the start label: BORDER_ST and the end label: BORDER_ST

c. Define contour in Subprogram

Normally the subprogram has the extension *.SPF. When we create new G code program, we can choose one of two types: Mainprogram MPF or Subprogram SPF.

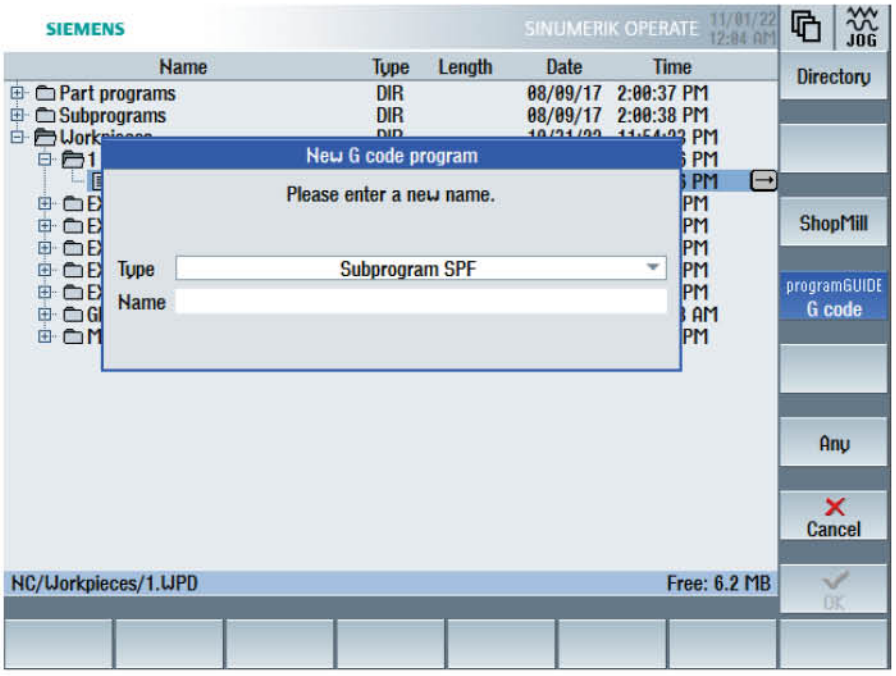


Figure 1.6: Creating subprogram with programGUIDE G code

```
NC/WKS/EXERCISE_1T_DIN/WUP.SPF
N1 G0 G18 G40 SUPA X350 Z450 D0¶
N2 M17¶
```

Figure 1.7: Subprogram with G/M code for tool change

d. Define contour using labels in subprogram

We can program using labels in subprogram, programming structure will be same as in main program.

3.4.2. Contour call in the program

Using **Cycle62** to call contour description in the program with 4 types as follows:

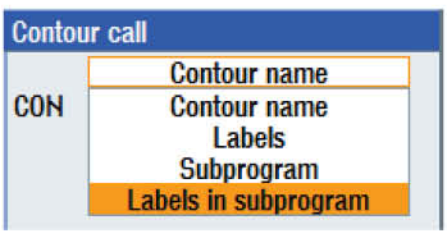


Figure 1.8: Four types to call contour in the program

3.5. Programme turning cycles

3.5.1. Stock removal and cut residual stock cycles

a. Stock removal

Softkey	Description
	Simple shoulder.
	Shoulder with radii/chamfer.
	Shoulder with bevel at an angle and radii/chamfers.

Parameters	Description
SC	Safety clearance
F	Feedrate
Machining	Roughing/finishing
Position	Position of the stock removal cycle, face/ longitudinal and inside/ outside
X0	Reference diameter in X
Z0	Reference point in Z
X1	End point in X
Z1	End point in Z
D	Maximum infeed
UX	Finishing allowance in X
UZ	Finishing allowance in Z

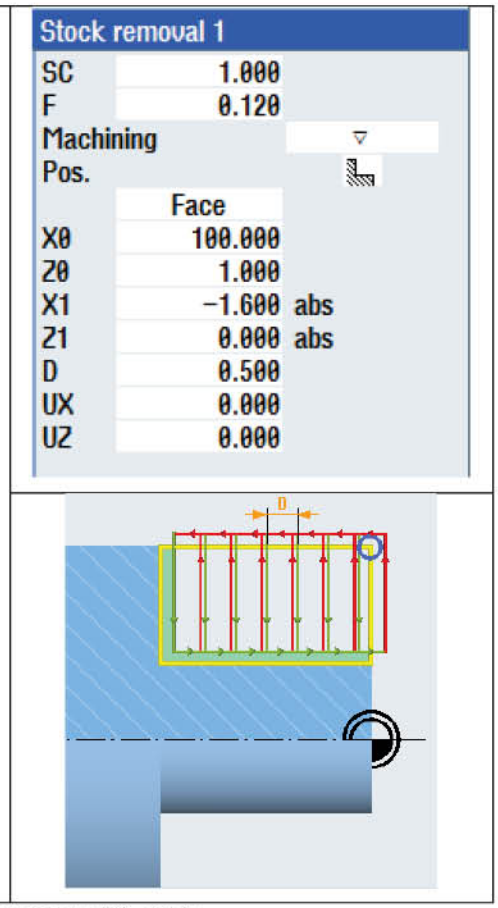


Table 1.1: Parameters of stock removal 1 cycle

b. Stock removal for contour

Parameters	Description
PRG	Name of program to be generated
Residual mat.	With subsequence residual material machining
CONR	Name for saving updated blank contour for resid. Material machining
SC	Safety clearance
F	Feedrate
Machining	Roughing/finishing
Position	Position of the stock removal cycle, face/ longitudinal and inside/ outside
D	Maximum infeed
UX	Finishing allowance in X
UZ	Finishing allowance in Z
DI	Continous cuts
BL	Blank description
XD	Allowances or cylinder dimensions
ZD	
Relief cuts	Machine relief cuts
Limit	Limit machining area

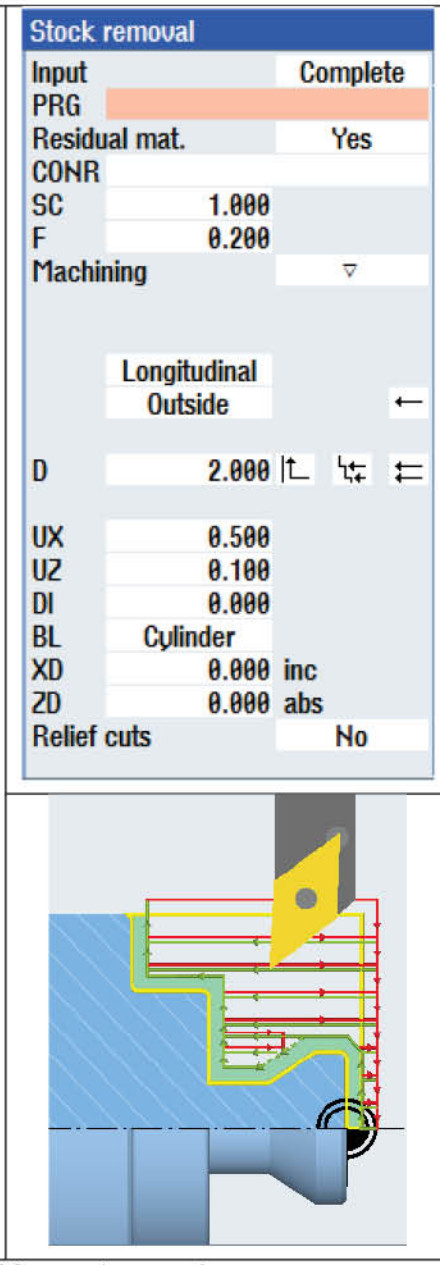


Table 1.2: Parameters of stock removal for contour cycle

c. Cut residual stock

Parameters	Description
PRG	Name of program to be generated
CON	Name for saving updated blank contour for resid. Material machining
SC	Safety clearance
F	Feedrate
Machining	Roughing/finishing
Position	Position of the stock removal cycle, face/ longitudinal and inside/ outside
D	Maximum infeed
UX	Finishing allowance in X
UZ	Finishing allowance in Z
DI	Continous cuts
Relief cuts	Machine relief cuts
FR	Insertion feed relief cuts
Limit	Limit machining area

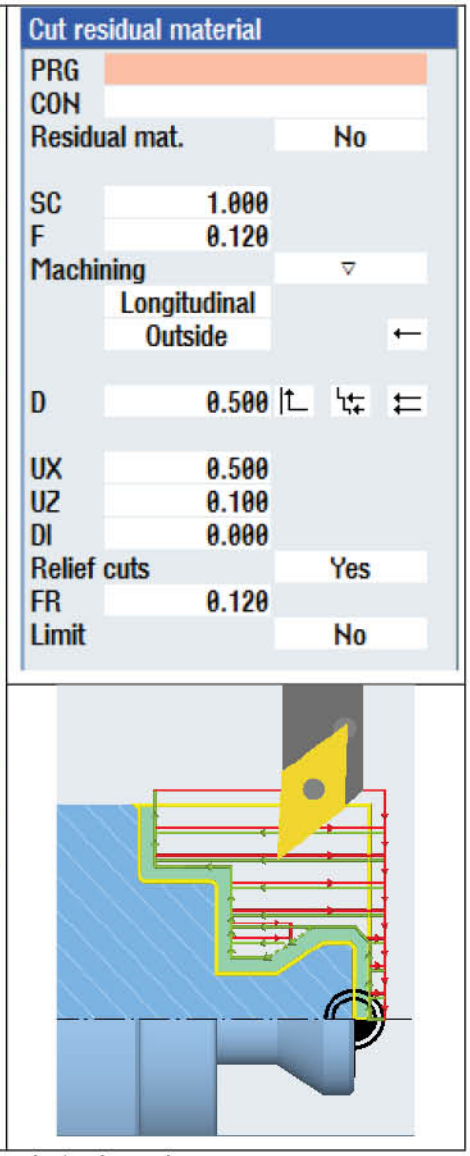


Table 1.3: Parameters of cut residual stock cycle

3.5.2. Groove cycles

Softkey	Description
	Straight groove
	Groove with inclines and chamfers/radii
	Groove with inclined edges, chamfers/radii on a cone

Parameters	Description
SC	Safety clearance
F	Feedrate
Machining	Roughing, finishing, roughing +finishing
Pos.	Inside or outside radial or axial
Reference point	Edge top left or right Edge inside left or right
X0 / Z0	Reference points in the X and Z axes
B1 **B2**	Width at bottom of groove Width at base of groove
T1	Groove depth/groove diameter
α1 and α2	Flank angle
FS1-4	Chamfer and/or radii 1-4
D	Maximum infeed
UX	Finishing allowance in X
UZ	Finishing allowance in Z
N	Number of grooves
DP	Distance between grooves

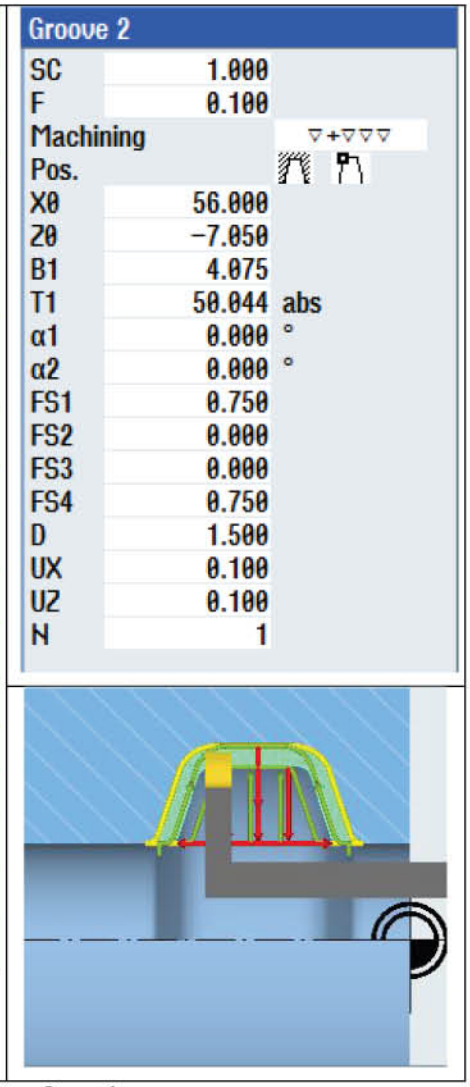


Table 1.4: Parameters of groove 2 cycle

3.5.3. Undercut cycles

Softkey	Description
Undercut form E	Undercut form E for grinding
Undercut form F	Undercut form F for grinding
Undercut thrd DIN	Undercut for DIN thread cutting
Undercut thread	Undercut for thread cutting

Parameters	Description
SC	Safety clearance
F	Feedrate
Machining	Roughing / finishing / roughing & finishing
Pos.	Front / rear / inside / outside
Machining type	Longitudinal or parallel to the contour
Form	Normal / short
P	Pitch
X0/ Z0	Reference X and Z axes
α	Insertion angle
D	Maximum infeed
UX	Finishing allowance in X
UZ	Finishing allowance in Z

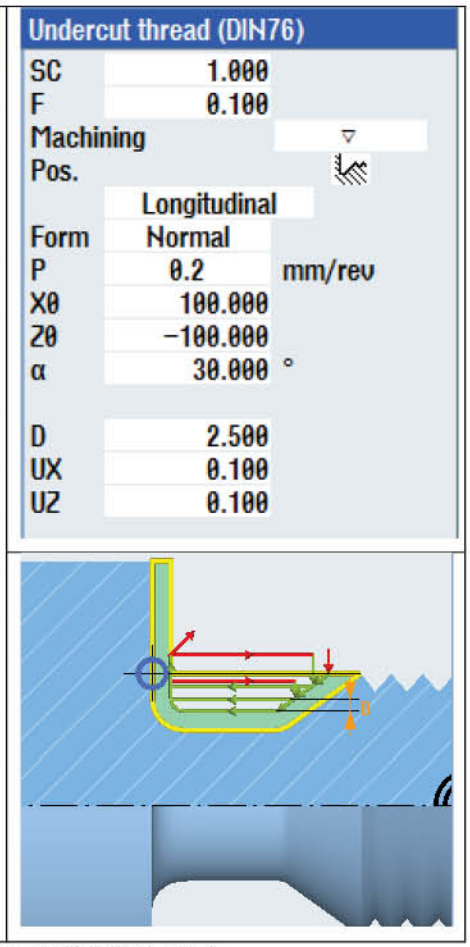


Table 1.5: Parameters of undercut thread (DIN76) cycle

3.5.4. Thread cycles

Types of thread cycles: thread longitudinal, thread taper, thread face, thread chain.

Parameters	Description
Table	Without/ ISO-metric/ Whitworth BSW/ Whit-worth
P	Thread pitch
G	Pitch change
Machining	Roughing / finishing / Roughing +finishing
Infeed	Degressive or linear
Thread	Male or female thread
X0 / Z0	Reference points in the X and Z axes
Z1	Thread length
LW, LW2, LW2=R	Thread advance, thread run-in
LR	Thread run-out
H1	Thread depth
DP, αP	Infeed slope (flank or
Flank infeed	Along a flank, with alter- nating flank
D1/ ND	First infeed/ number of roughing cuts
U	Finishing allowance
NN	Number of idle passes
VR	Return distance
Multiple	1 thread or multiple thread
α0	Starting angle offset

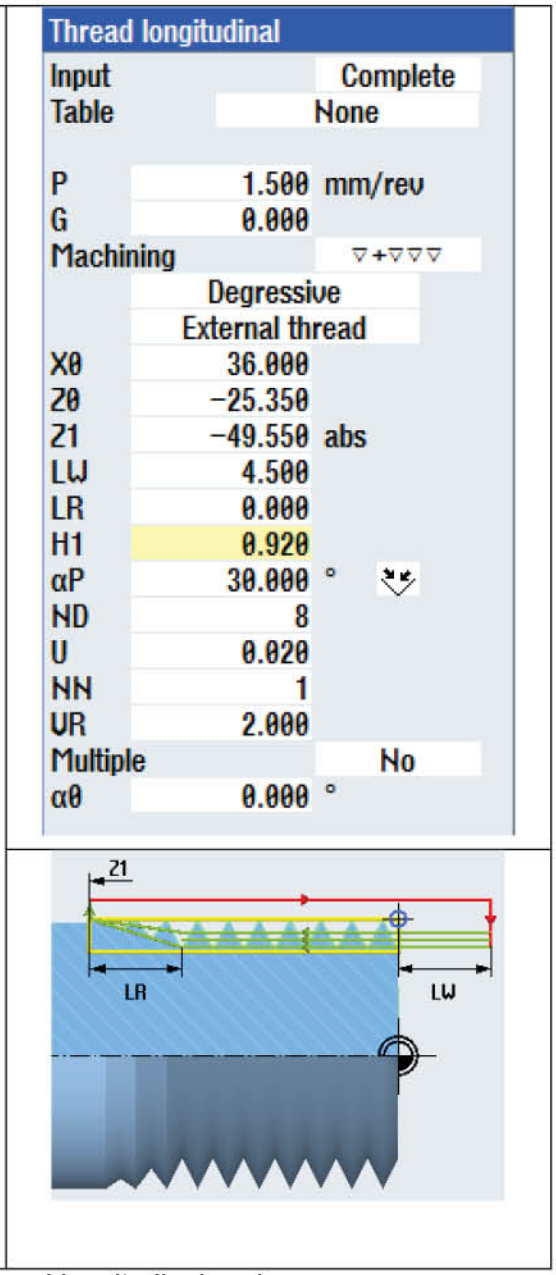


Table 1.6: Parameters of thread longitudinal cycle

3.5.5. 3.5.5. Cutoff cycle

Parameters	Description
SC	Safety clearance
F	Feedrate
DIR	Direction of rotation
S	Spindle speed
X0 / Z0	Reference points in X and
FS / R	Chamfer or radius
X1	Depth for speed reduction
FR	Reduced feedrate
SR	Reduced speed
X2	Groove depth X axis

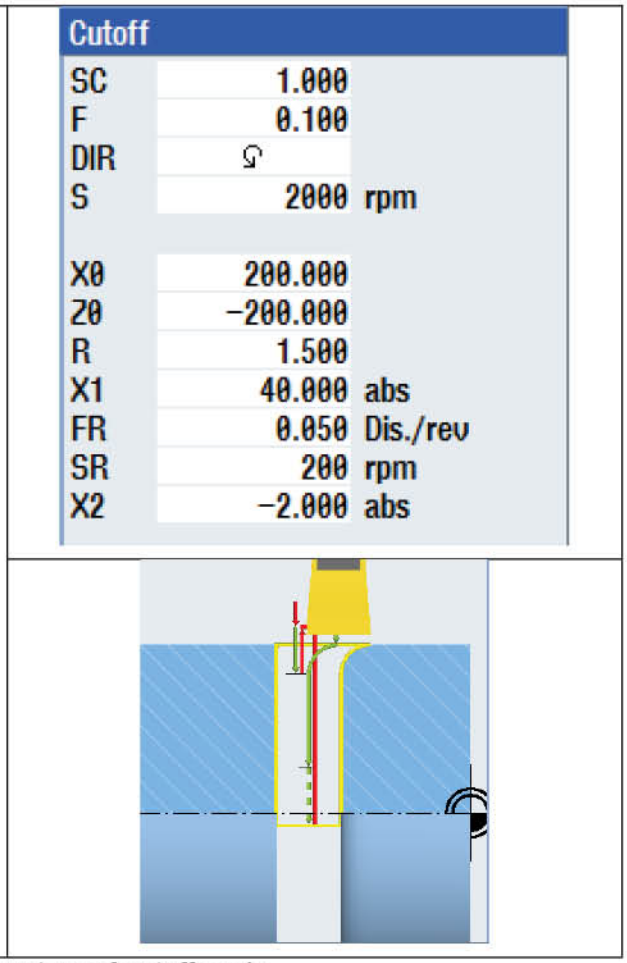


Table 1.7: Parameters of cutoff cycle

3.6. Programme milling cycles

3.6.1. Path milling cycle

Parameters	Description
PL	Machining plane
RP	Retraction plane
SC	Safety distance
F	Feedrate
Machining	Roughing/finishing/chamfer
Machine direction	Forward/ backward
Radius comp.	Left/ right/ off
Z0	Reference point
Z1	Machining depth
DZ	Maximum infeed per cut
UZ	Finishing allowance depth
UXY	Finishing allowance plane
Approach, retract:	Quarter circle, semicircle, straight line, vertical
L1	Approach length
FZ	Infeed feedrate
L2	Retract length
Lift mode	No retract, to RP, Z0+SC, by safety distance

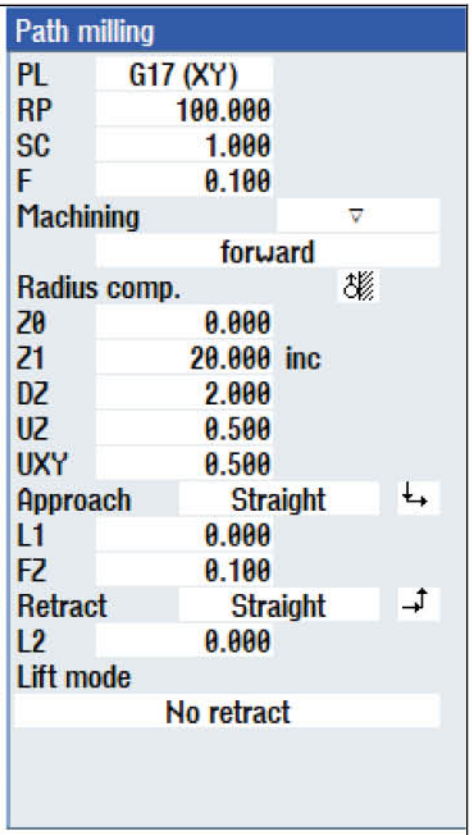


Table 1.8: Parameters of path milling cycle

3.6.2. Pocket cycles

a. Rectangular pocket

Parameters	Description
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
Reference point	Position of reference point
Machining	Roughing/finishing/ chamfer
Machine position	Single position, position pattern (MCALL)
X0, Y0, Z0	Reference points
W	Pocket width
L	Pocket length
R	Corner radius
α0	Rotation angle
Z1	Depth related to Z0
DXY	Maximum plane infeed
DZ	Maximum infeed depth
UXY	Finishing allowance plane
UZ	Finishing allowance depth
Insertion	Predrilled, vertical, helical, oscilat
FZ	Infeed feedrate
Removing	Complete machining, remachining

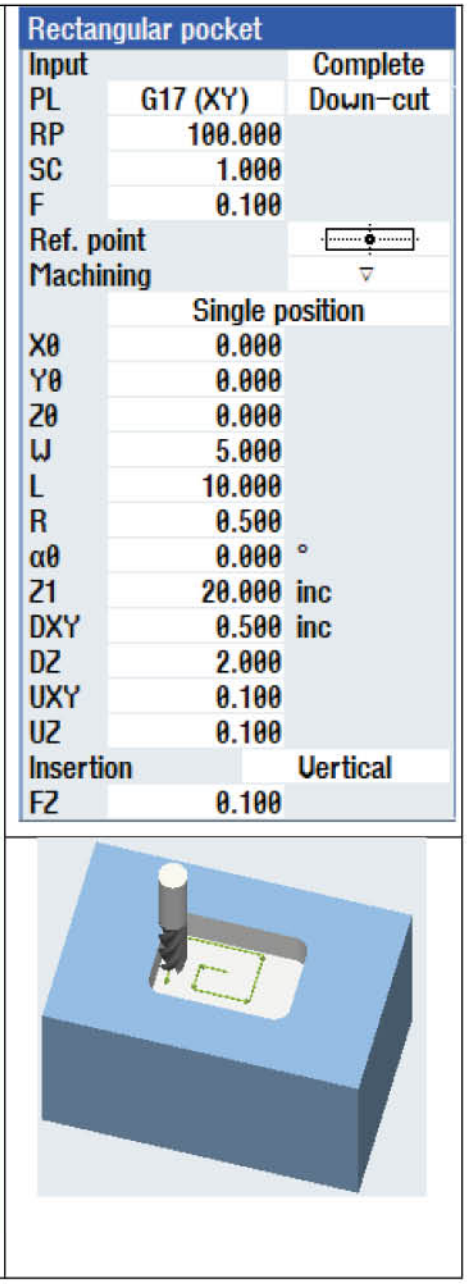


Table 1.9: Parameters of rectangular pocket cycle

b. Circular pocket

Parameters	Description
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
Machining	Roughing/finishing/chamfer
Machine type	Centric, helical
Machine position	Single position, position pattern (MCALL)
X0, Y0, Z0	Reference points
Ø	Diameter of pocket
Z1	Depth related to Z0
DXY	Maximum plane infeed
DZ	Maximum infeed depth
UXY	Finishing allowance plane
UZ	Finishing allowance depth
Insertion	Predrilled, vertical, helical, oscilat
FZ	Infeed feedrate
Removing	Complete machining, remachining

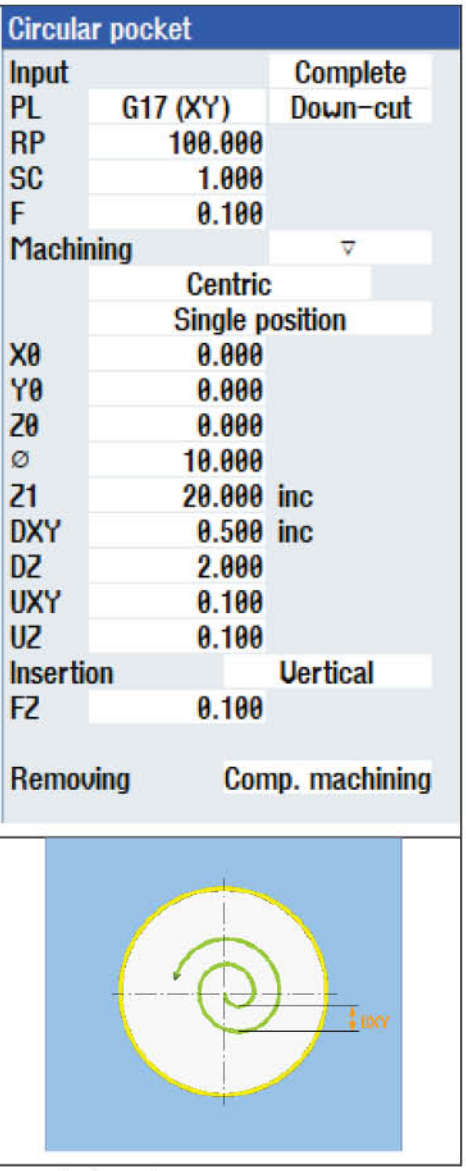


Table 1.10: Parameters of circular pocket cycle

3.6.3. Spigot cycles

a. Rectangular spigot

Parameters	Description
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
FZ	Infeed feedrate
Reference point	Position of reference point
Machining	Roughing/finishing/chamfer
Machine position	Single position, position pattern (MCALL)
X0, Y0, Z0	Reference points
W1	Width of blank spigot
L1	Length of blank spigot
W	Width of spigot
L	Length of spigot
R	Corner radius
α0	Rotation angle
Z1	Depth related to Z0
DZ	Maximum infeed depth
DXY	Maximum plane infeed
UZ	Finishing allowance depth

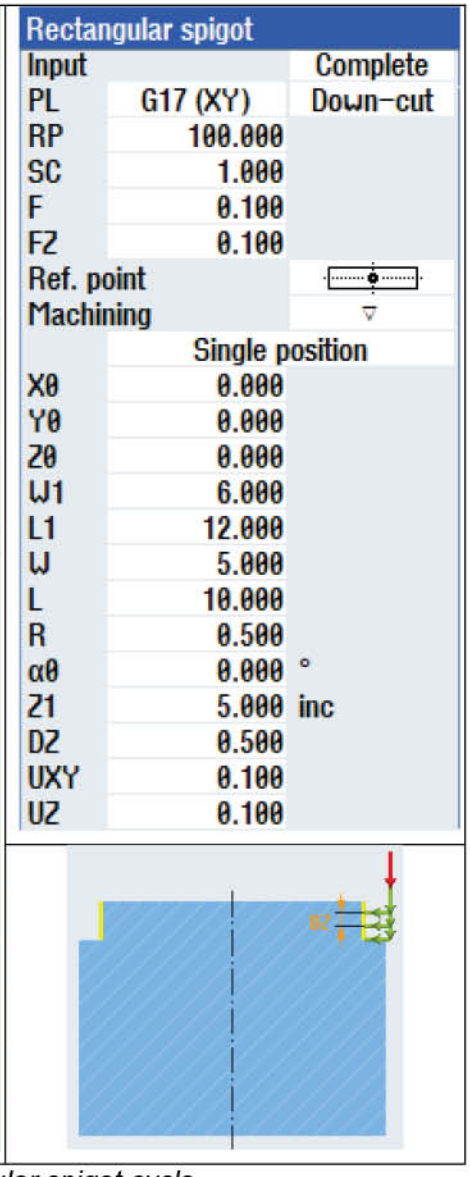


Table 1.11: Parameters of rectangular spigot cycle

b. Circular spigot

Parameters	Description
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
FZ	Infeed feedrate
Machining	Roughing/finishing/chamfer
Machine position	Single position, position pattern (MCALL)
X0, Y0, Z0	Reference points
Ø1	Diameter of blank spigot
Ø	Diameter of spigot
Z1	Depth related to Z0
DZ	Maximum infeed depth
DXY	Maximum plane infeed
UXY	Finishing allowance plane
UZ	Finishing allowance depth

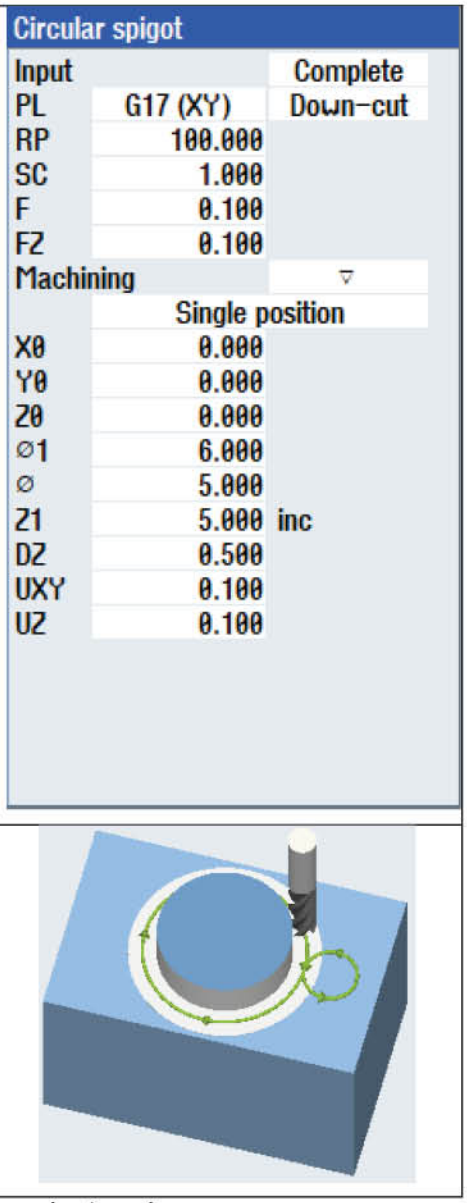


Table 1.12: Parameters of circular spigot cycle

c. Multi-edge spigot

Parameters	Description
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
Machining	Roughing/finishing/chamfer
Machine position	Single position, position pattern (MCALL)
X0, Y0, Z0	Reference points
Ø	Diameter of blank spigot
N	Number of edges
α0	Rotation angle
R	Rounding radius
Z1	Depth related to Z0
DXY	Maximum plane infeed
DZ	Maximum infeed depth
UXY	Finishing allowance plane
UZ	Finishing allowance depth

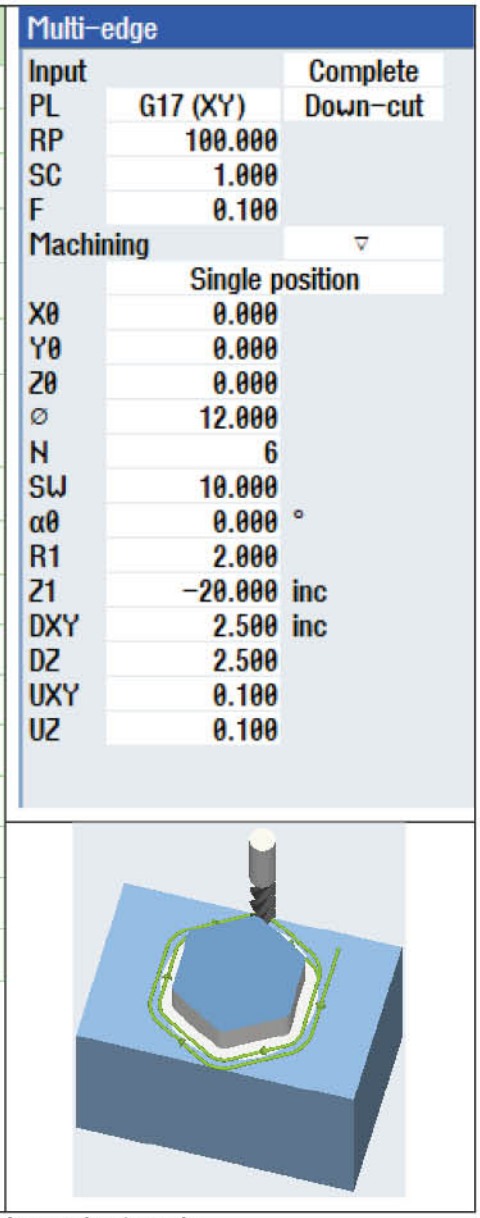


Table 1.13: Parameters of multi-edge spigot cycle

3.6.4. Slot cycles

a. Longitudinal slot

Parameters	Description
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
Reference point	Position of reference point
Machining	Roughing/finishing/chamfer
Machine position	Single position, position pattern (MCALL)
X0, Y0, Z0	Reference points
W	Slot width
L	Slot length
α0	Rotation angle
Z1	Depth related to Z0
DZ	Maximum infeed depth
DXY	Maximum plane infeed
UXY	Finishing allowance plane
UZ	Finishing allowance depth
Insertion	Predrilled, vertical, helical, oscilat
FZ	Infeed feedrate

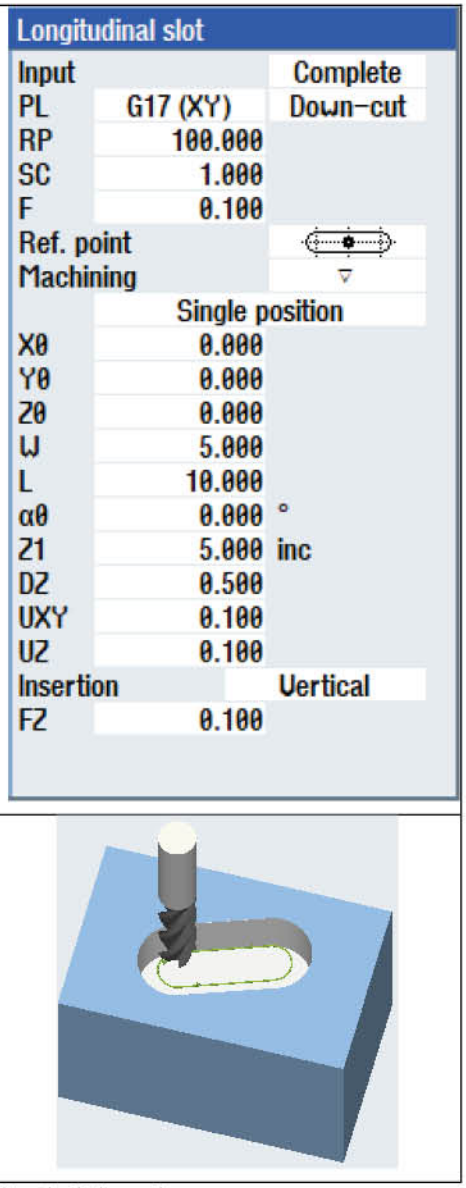


Table 1.14: Parameters of longitudinal slot cycle

b. Circumferential slot

Parameters	Description
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
Machining	Roughing/finishing/chamfer
FZ	Infeed feedrate
Circular pattern	Full circle, pitch circle
X0, Y0, Z0	Reference points
N	Numbers of slots
R	Radius of circum. slot
α0	Starting angle
α1	Opening angle of slot
W	Slot width
Z1	Depth related to Z0
DZ	Maximum infeed depth
UXY	Finishing allowance plane
Positioning	Straight, circle

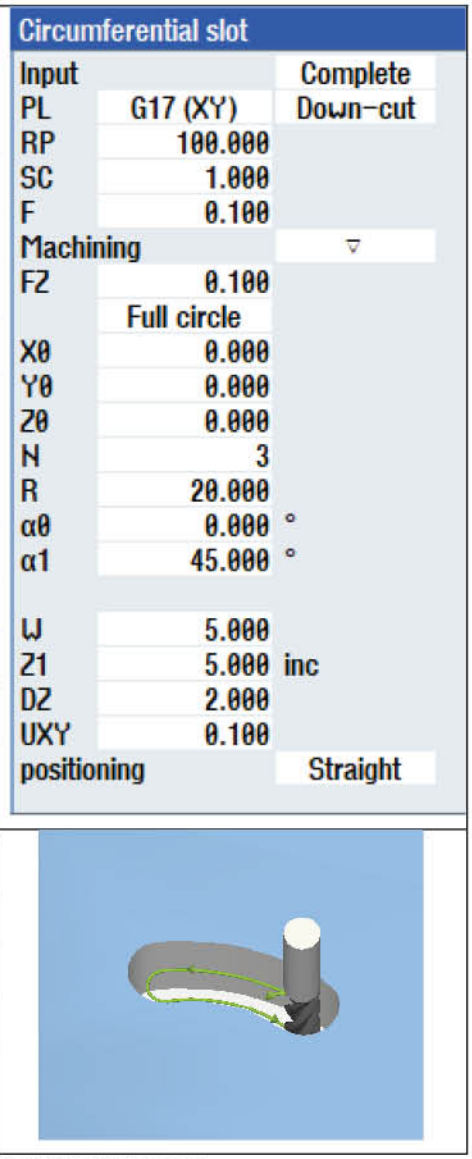


Table 1.15: Parameters of circumferential slot cycle

c. Open slot

Parameters	Description
PL	Machining plane
RP	Retraction plane
SC	Safety distance
F	Feedrate
Reference point	Position of reference point
Machining	Roughing/finishing/chamfer
Technology	Spiral milling, plunge cutting
Mill direction	Down-cut, Up-cut
Machine position	Single position, position pattern (MCALL)
X0, Y0, Z0	Reference points
W	Slot width
L	Slot length
α0	Rotation angle
Z1	Depth related to Z0
DXY	Maximum plane infeed
DZ	Maximum infeed depth
UXY	Finishing allowance plane
UZ	Finishing allowance depth

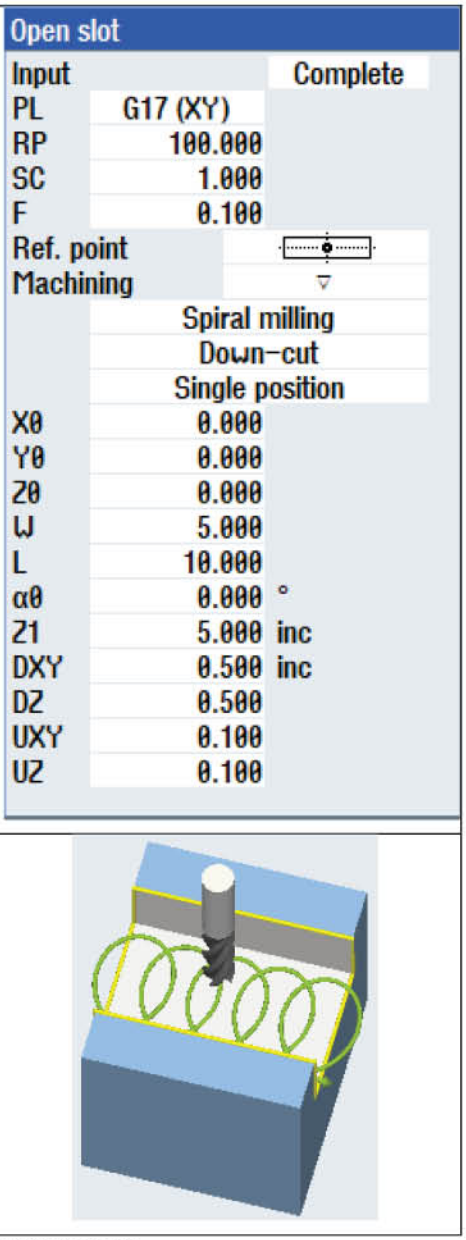


Table 1.16: Parameters of open slot cycle

3.6.5. Arbitrary contour pocket cycles

a. Mill pocket

Parameters	Description
PRG	Name of the program to be generated
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
Machining	Roughing/finishing/chamfer
Z0	Reference point Z
Z1	Depth related to Z0
DXY	Maximum plane infeed
DZ	Maximum infeed depth
UXY	Finishing allowance plane
UZ	Finishing allowance depth
Starting point	Automatically, manually
Insertion	Vertical, helical, oscilat
FZ	Infeed feedrate
Lift mode	To RP, Z0+SC

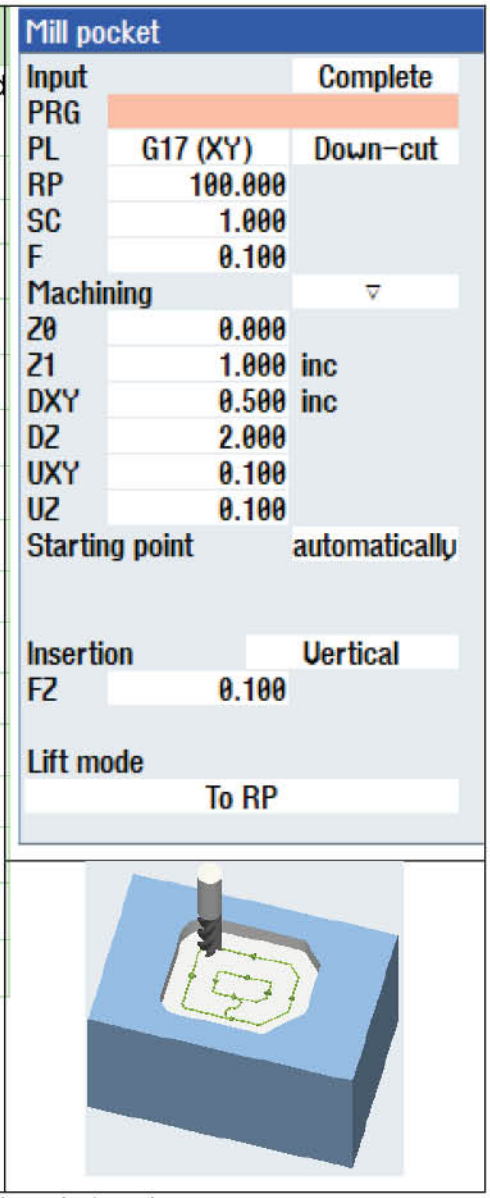


Table 1.17: Parameters of mill pocket cycle

b. Pocket residual material

Parameters	Description
PRG	Name of the program to be generated
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
Machining	Roughing/finishing/chamfer
TR	Reference tool milling from pocket
D	Edge number
Z0	Reference point Z
Z1	Depth related to Z0
DXY	Maximum plane infeed
DZ	Maximum infeed depth
UXY	Finishing allowance plane
UZ	Finishing allowance depth
Lift mode	To RP, Z0+SC

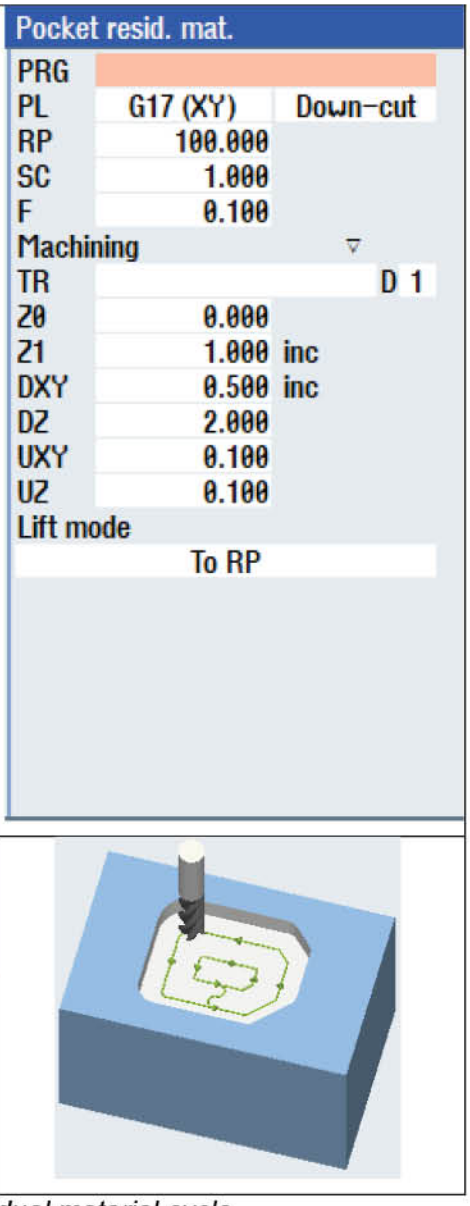
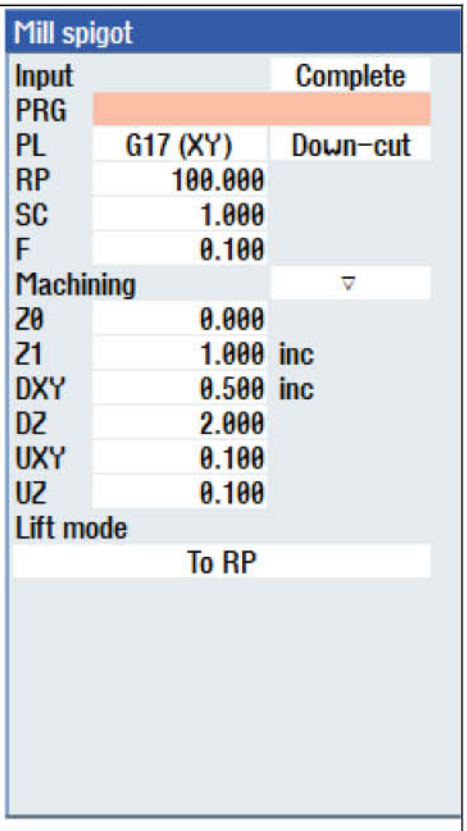


Table 1.18: Parameters of pocket residual material cycle

3.6.6. Arbitraty contour spigot cycles

a. Mill spigot

Parameters	Description
PRG	Name of the program to be generated
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
Machining	Roughing/finishing/chamfer
Z0	Reference point Z
Z1	Depth related to Z0
DXY	Maximum plane infeed
DZ	Maximum infeed depth
UXY	Finishing allowance plane
UZ	Finishing allowance depth
Lift mode	To RP, Z0+SC



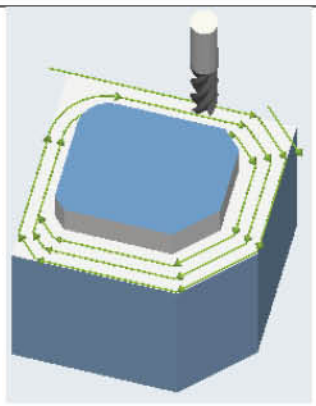


Table 1.19: Parameters of milling spigot cycle

b. Spigot residual material

Parameters	Description
PRG	Name of the program to be generated
PL	Machining plane
Mill direction	Down-cut, Up-cut
RP	Retraction plane
SC	Safety distance
F	Feedrate
Machining	Roughing/finishing/chamfer
TR	Reference tool milling from pocket
D	Edge number
Z0	Reference point Z
Z1	Depth related to Z0
DXY	Maximum plane infeed
DZ	Maximum infeed depth
UXY	Finishing allowance plane
UZ	Finishing allowance depth
Lift mode	To RP, Z0+SC

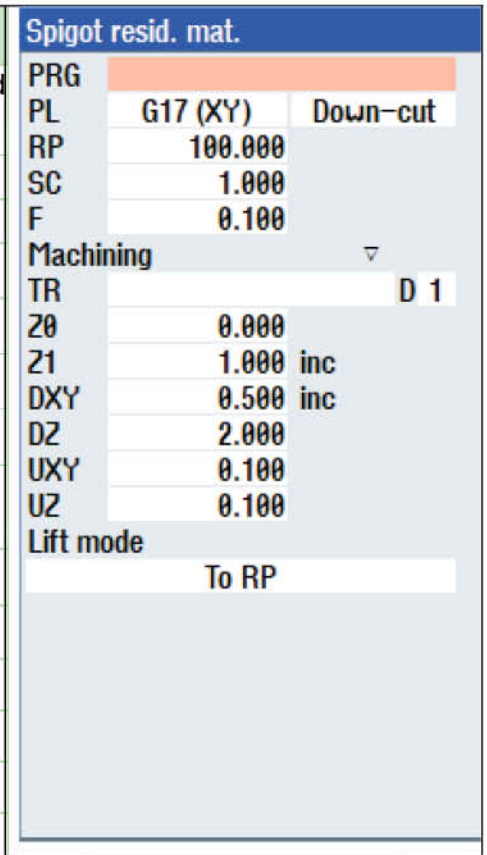


Table 1.20: Parameters of spigot residual material cycle

3.7. Programme drilling cycles

3.7.1. Drilling cycles

a. Centering

Parameters	Description
PL	Machining plane
RP	Retraction plane
SC	Safety distance
F	Feedrate
Machining position	Single position, position pattern (MCALL)
Z0	Reference point Z
Centering in relation to diameter	Diameter, tip
Ø	Diameter of centering
DT	Dwell time at final depth in seconds

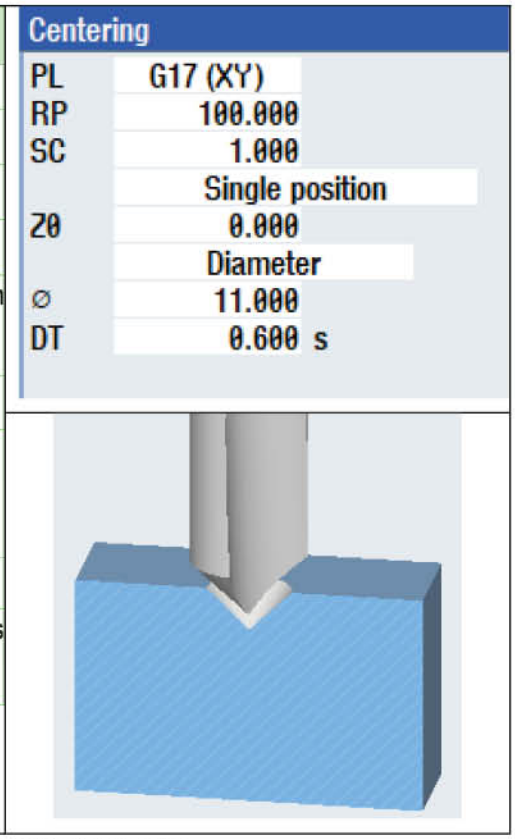


Table 1.21: Parameters of centering cycle

b. Drilling

Parameters	Description
PL	Machining plane
RP	Retraction plane
SC	Safety distance
Machining position	Single position, position pattern (MCALL)
Z0	Reference point Z
Centering in relation to diameter	Diameter, tip
Z1	Depth related to Z0
Spot drill	Yes, no
Thrd. drill	Drilling with constant feed
DT	Dwell time at final depth in seconds

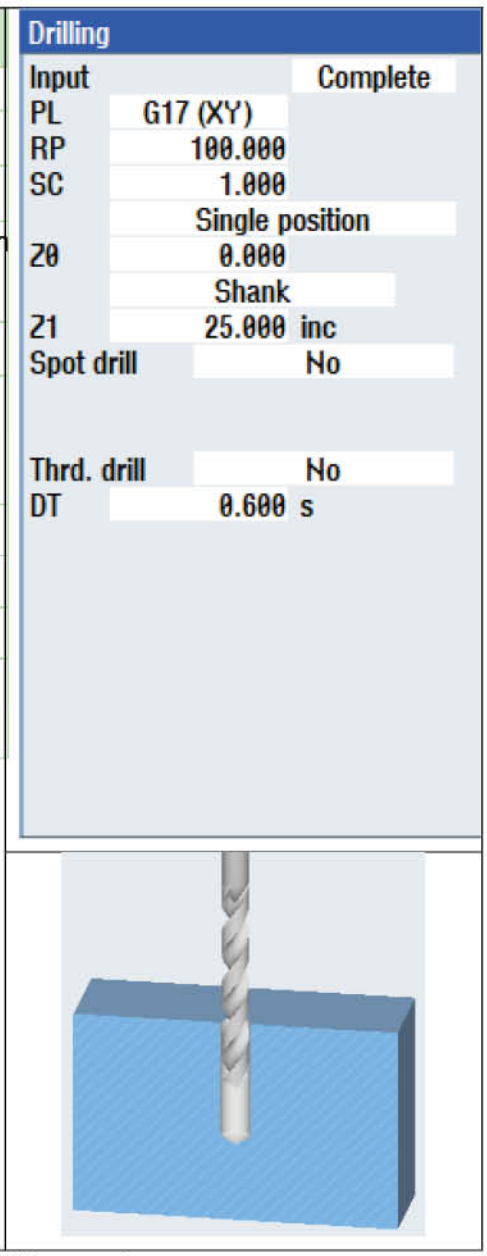


Table 1.22: Parameters of drilling cycle

c. Deep hole drilling 1

Parameters	Description
PL	Machining plane
RP	Retraction plane
SC	Safety distance
Machining position	Single position, position pattern (MCALL)
Machining	Chipbreaking, chipremoval
Z0	Reference point Z
Centering in relation to diameter	Diameter, tip
Z1	Depth related to Z0
FD1	Feed percentage for first infeed
D	1st drilling depth related to Z0
DF	Percentage for every further infeed
V1	Minimum depth infeed
V2	Retraction value: after each machining
DTB	Dwell time at drilling depth in seconds
DT	Dwell time at final depth in seconds

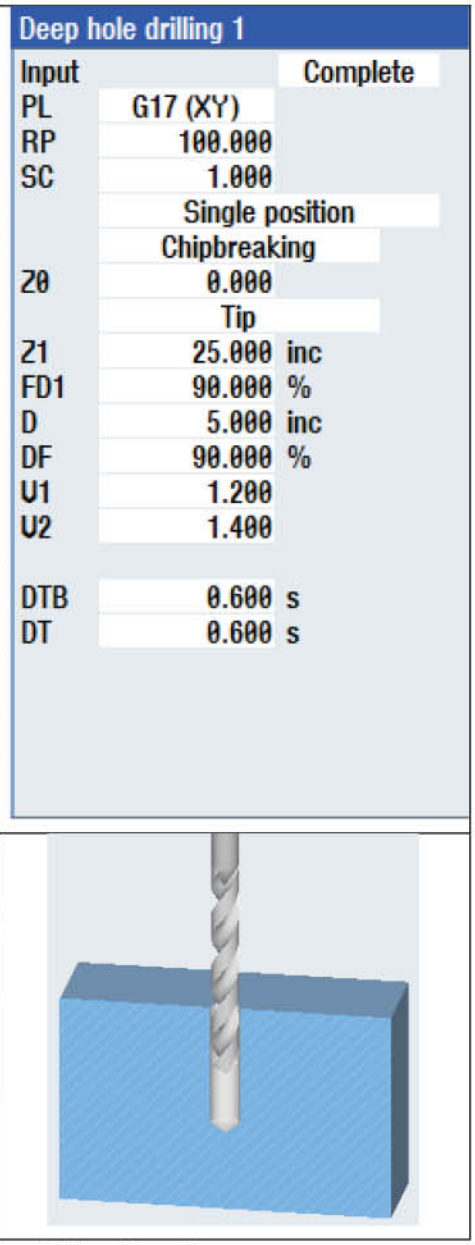


Table 1.23: Parameters of deep hole drilling 1 cycle

3.7.2. Reaming cycle

Parameters	Description
PL	Machining plane
RP	Retraction plane
SC	Safety distance
Machining position	Single position, position pattern (MCALL)
F	Feedrate
FR	Feed for retract
Z0	Reference point Z
Z1	Depth related to Z0
DT	Dwell time at final depth in seconds

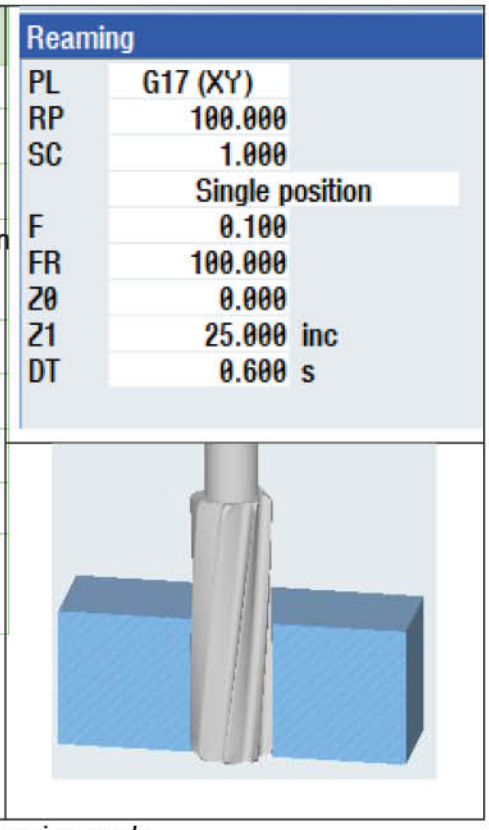


Table 1.24: Parameters of reaming cycle

3.7.3. Tapping cycle

Parameters	Description
PL	Machining plane
RP	Retraction plane
SC	Safety distance
Compensating chuck mode	With compensat. chuck, w/o compensat. chuck
Machining position	Single position, position pattern (MCALL)
Z0	Reference point Z
Z1	Depth related to Z0
Direction of rotation of thread	RH thread, LH thread
Table	None, ISO metric, Whitworth, UNC
P	Thread pitch in mm/rev
αS	Starting angle offset
S	Spindle speed
Machining	1 cut, chipbreaking, chipremoval
DT	Dwell time at final depth in seconds
SR	Spindle speed for retraction
SDE	Direction of rotation after end of cycle

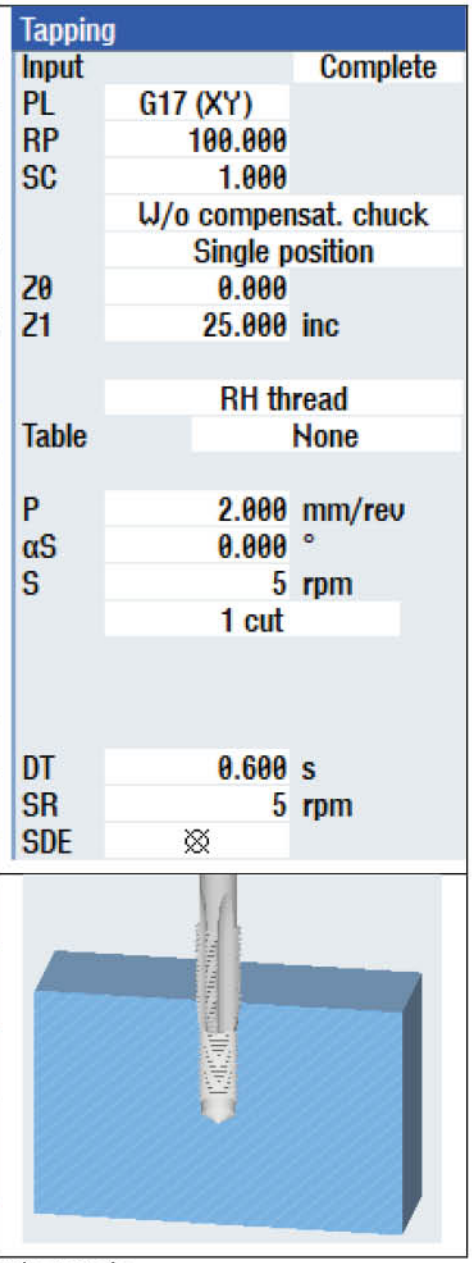


Table 1.25: Parameters of tapping cycle

3.8. Programme jump and branches, check structures

3.8.1. Programme jump and branches

- Return jump to the start of the program (GOTOS)
- Program jumps to jump markers (GOTOB, GOTOF, GOTO, GOTOC)
  - + GOTOB: Jump instruction with jump destination towards the beginning of the program.
  - + GOTOOF: Jump instruction with jump destination towards the end of the program
  - + GOTO: Jump instruction with jump destination search. The search is first made in the direction of the end of the program, then in the direction of the beginning of the program.
  - + GOTOC: Same effect as for GOTO with the difference that Alarm 14080 "Jump designation not found" is suppressed.

This means that program execution is not interrupted in the case that the jump destination search is unsuccessful – but is continued with the program line following the GOTOC command.

- Program branch (CASE ... OF ... DEFAULT ...)
- Repeat program section (REPEAT, REPEATB, ENDLABEL, P)
  - + REPEAT: Command for repeating a program line
  - + REPEATB: Command for repeating a program section
  - + ENDLABEL: Keyword marking the end of a program section to be repeated.
  - + If the line with the ENDLABEL contains further operations, these are executed again on each repetition. ENDLABEL can be used more than once in the program.
  - + P: Address for specifying the number of repetitions.

3.8.2. Check structures

- Program loop with alternative (IF, ELSE, ENDIF)
- Continuous program loop (LOOP, ENDLOOP)
- Count loop (FOR ... TO ..., ENDFOR)
- Program loop with condition at start of loop (WHILE, ENDWHILE)
- Program loop with condition at the end of the loop (REPEAT, UNTIL).

3.9. Plan and realise operational work order of the industry

3.9.1. Program turning application using ProgramGUIDE

a. Description:

A first programGuide exercise is to be programmed with the help of stock removal

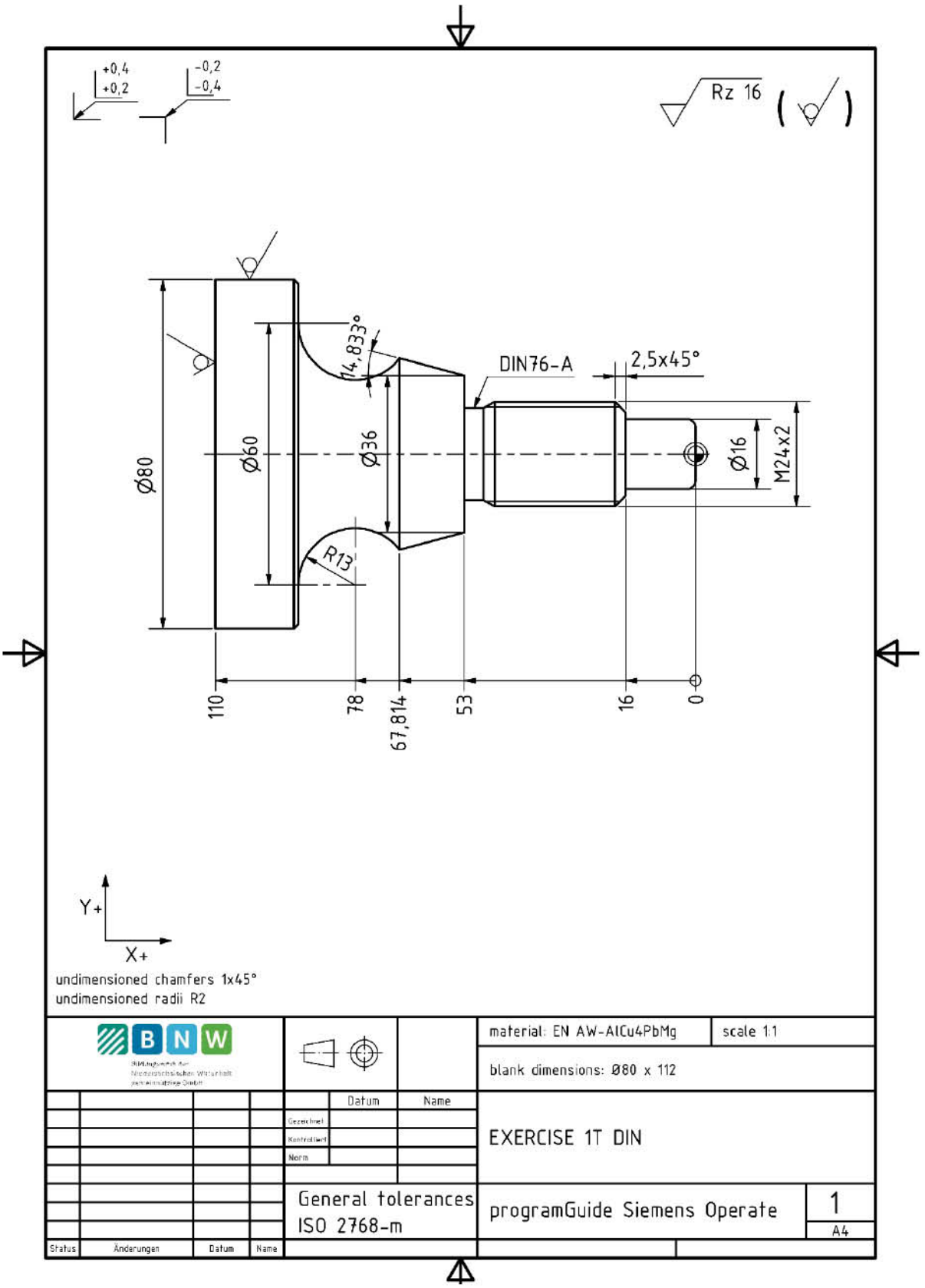
and threading cycle calls.

b. Objective:

A new EXERCISE_1T_DIN program should be created and opened in the editor.

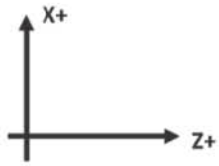
The program header, stock removal without relief cuts, stock removal residual material and stock removal finishing, threading and end of the program to be programmed and subsequently simulated. All tools are used as economically as possible.

c. The technical drawing EXERCISE_1T_DIN:

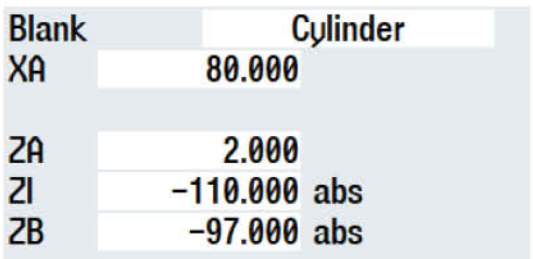
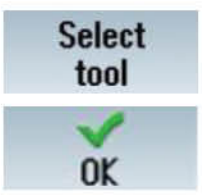

+0,4
+0,2
-0,2
-0,4
Rz 16
14,833°
DIN76-A
2,5x45°
Ø80
Ø60
Ø36
Ø16
M24x2
R13
110
78
67,814
53
16
0
Y+
X+
undimensioned chamfers 1x45°
undimensioned radii R2
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Ø80 x 112
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE 1T DIN
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name

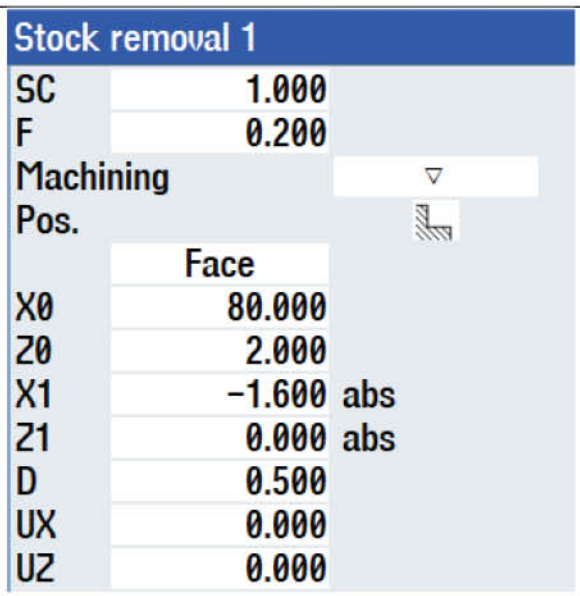
d. The following tools and technology data are required for programming:

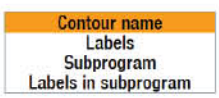
Workpiece: EXERCISE 1T DIN	Blank material: EN-AW AlCu4PbMg	Machine configuration:
Drawing: 1(1)	Blank dimensions: Ø80 x 112	Turning 2 axes

	Tools for outer machining							
	Technological data							
Tool name in OFFSET	ROUGHING_ T80 A	ROUGHING_ T55 A	FINISHING _T35	THREAD-ING_2				
Cutting radius (mm)	0,8	0,4	0,4					
Cutting speed (m/min)	200	200	200 / 250	120				
Cutting depth a_p = max.	1 / 2	2	2 / 0,5					
Cutting material	SC	SC	SC	SC				
Feed rate per rev./pitch	0,3 / 0,12	0,25 / 0,12	0,2 / 0,1					
Angle/pitch/diameter Ds	80°	55°	35°	P 2				
								

e. Create a new G code programGuide program, like described in this unit.
Create a new program directory and call the program for example "EXERCISE_1T_DIN.MPF".

1.	Program the first lines of the program by inserting the following G code commands: **N1 G18 G64 G71 G90**
2. 	Insert now a blank for the simulation into the program. Press the HSK 1.6 “Various” to open the work area “Various”. The following input mask for the blank input opens.
3.	Enter following values/parameters for the blank: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N2 WORKPIECE(,,,"CYLINDER",192,2,-110,-97,80)**
4.	Now insert the following program lines for the programming conditions and tool changing point. **N3 DIAMON LIMS=3000** **N4 G54** **N5 WWP** Note: Find the subprogram “WWP” in this document at pages 86, 87 and 88.
5. 	Now, insert a new tool into the program. Press the VSK 1.1 “Select tool“ and the tool list window opens. Mark the desired tool “ROUGHING_T80 A” in the tool list window with the orange selection cursor, by using the blue cursor keys on the keyboard. The following program line is inserted into the program with “OK”: **N6 T="ROUGHING_T80 A"** Alternatively you can program the tool by hand.

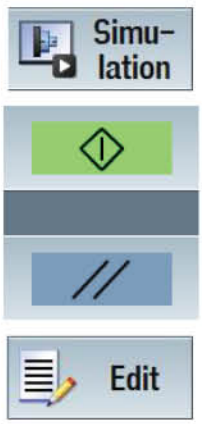
6.	Insert the following G code commands into the program: **N7 G96 S200 M4** **N8 G95 F0.2**
7.	Insert the following information message into the program: **N9 MSG("FACE CUTTING LENGTH 110 mm")**
8.	Insert the start position of the stock removal with rapid traverse: **N10 G0 X82 Z5 M8**
9. 	Insert a “Stock removal” cycle into the program. (Stock removal 1, Face) Press the HSK 1.3 “Turning”, to open the technology “Turning”. Press the VSK 1 “Stock removal”. The work area with the functions for the technology “Stock removal” opens. Press the VSK 3 “Stock removal 1”, to open the input mask for the “Stock removal 1” cycle.
10. 	 Enter the following values into the input mask. Use the “Select” key where indicated. Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N12 CYCLE951(80,2,-1.6,0,-1.6,0,1,0.5,0,0,12,0,0,0,1,0.2,0,2,1110000)**
11.	Insert the following information message into the program: **N13 MSG("ROUGHING OUTSIDE UP CUT CONTOUR")**
12.	Insert the start position of the stock removal for contour turning with rapid traverse: **N14 G0 X82 Z5 M8**

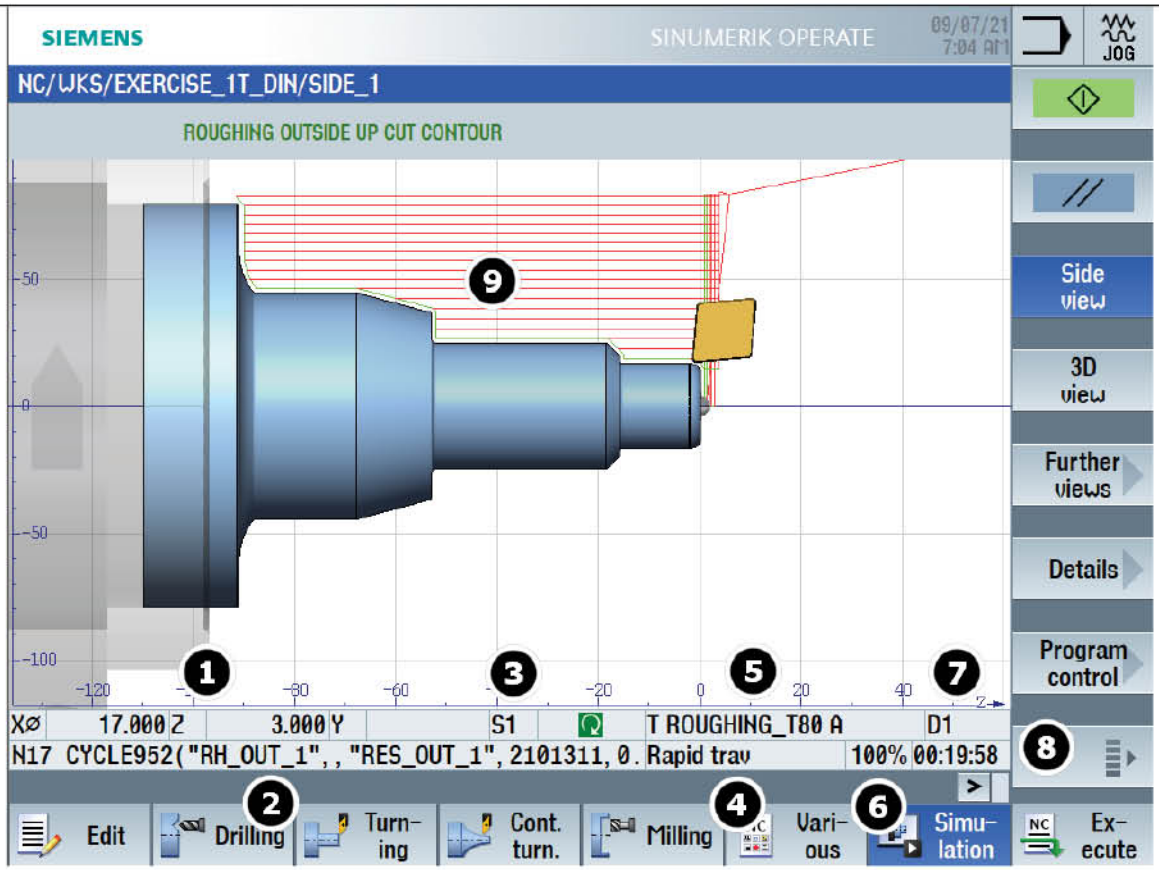
13. 	Program now the call up of the contour “CYCLE62”. Note: The contour itself is programmed at the end of the program. For this, press the HSK 1.4 “Cont. turn.” to open the technology “Contour turning”. Press now the VSK 1 “Contour”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
14.	Insert the following “Labels” name into the input mask:
  	Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N16 CYCLE62(,2,"OUTS_ST","OUTS_END")** Note 1: The contour selection for the “Contour call” you call up here has four different choices for programming: • Contour name (graphic editor) • Labels (G code) • Subprogram • Labels in subprogram Note 2: The “Labels” with the contour description you called up here, will be inputted behind the” M30”.
15. 	Program now the stock removal cycle “CYCLE952”. Press the HSK 1.4 “Cont. turn.” to open the technology “Contour turning”. Press the VSK 2 "Stock removal" to the input screen for the technology "removal" to open.

16.	Enter the following values into the input mask. Use the “Select” key where indicated. PRG: Name of program to be generated CONR: Name for saving updated blank contour for residual material machining
17.	Confirm your inputs by pressing the VSK 8 “Accept”. The following program code is inserted into the program: **N17** **CYCLE952("RH_OUT_1",,"RES_OUT_1",2101311,0.3,0,23,2,0.1,0.1,0.5,0.1,** **0.1,0,1,0,0,,,,,2,2,,,0,1,,0,12,1110110,1,0)** Add the tool changing point into the program: **N18 WWP**
18.	Enter now the finished contour behind the "M30" between the "Labels" you called in the “CYCLE62” by using DIN/ISO G code programming: **N41 OUTS_ST:** **N42 G0 X0 Z2** **N43 G1 Z0** **N44 G1 X16 RND=2** **N45 G1 Z-16** **N46 G1 X24 CHR=2.5** **N47 G1 Z-46**

Stock removal

Input	Complete	
PRG	RH_OUT_1	
Residual mat.	Yes	
CONR	RES_OUT_1	
SC	1.000	
F	0.300	
Machining	▽	
	Longitudinal	
	Outside	←
D	2.000	
UX	0.500	
UZ	0.100	
DI	0.000	
BL	Cylinder	
XD	0.000	inc
ZD	0.000	inc
Relief cuts	No	
Limit	No	

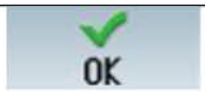
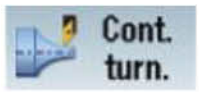
	N48 G1 X21 ANG=210 RND=1 **N49 G1 Z-53 RND=1** **N50 G1 X36 RND=0.4** **N51 G1 Z-67.814 ANG=165.167** **N52 G2 X60 Z-91 I=AC(60) K=AC(-78) ;CR=13** **N53 G1 X78** **N54 G1 X81 ANG=135** **N55 OUTS_END:**
19. 	Start the simulation for the work sequences stock removal face and stock removal outside up cut contour: Press the HSK 1.7 “Simulation” to open the program check. Note 1: With the “Reset” and “NC-Start” buttons you can repeat your program check with the simulation several times. Press the HSK 1.1 “Edit” to open the edit mode again. Note 2: It´s important to leave the simulation screen with the function “Edit”.

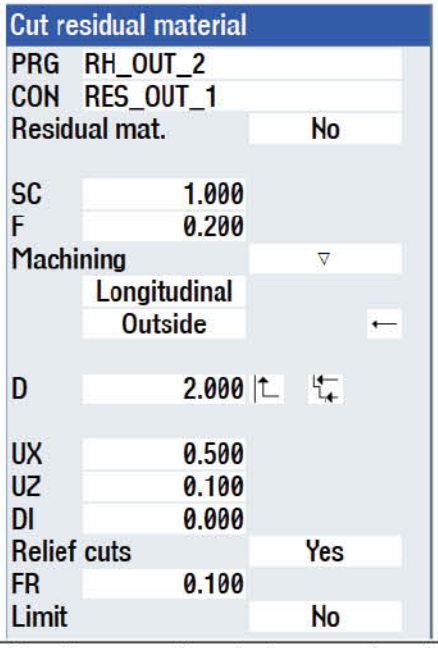


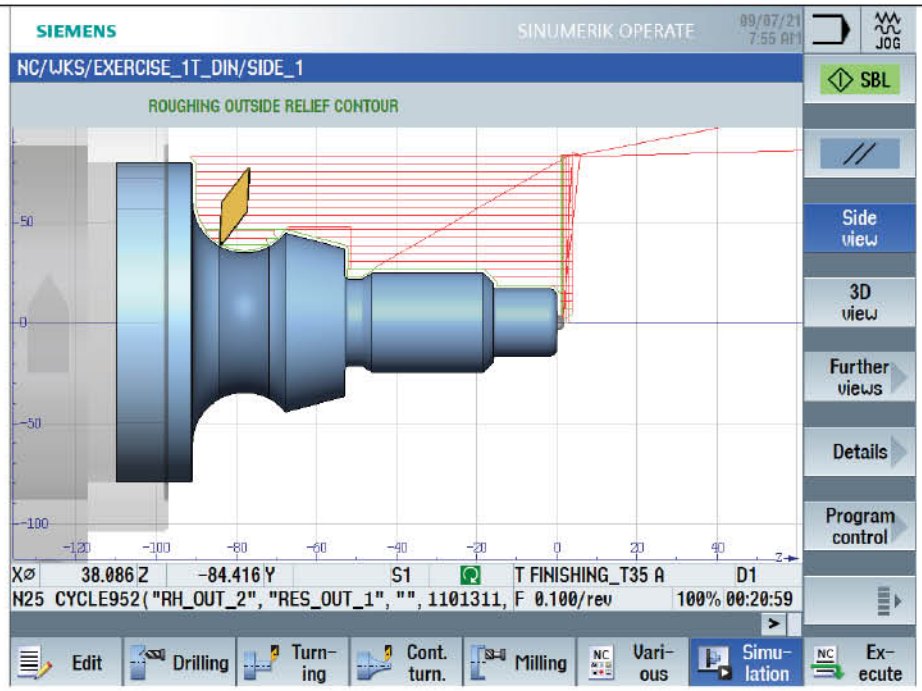
The simulation screen shows following details:

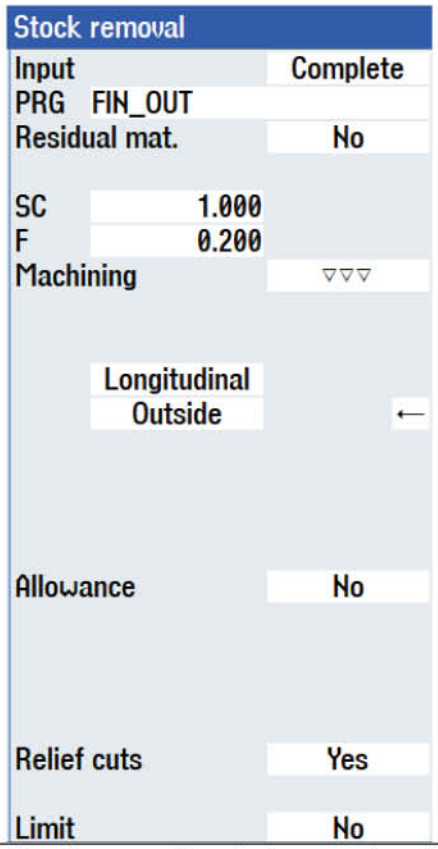
1. Current tool path position
2. Current NC block in the program
3. Spindle with rotation direction
4. Feedrate or rapid traverse
5. Active tool in processing
6. Override speed for simulation (max. 120%)
7. Active cut of the current tool
8. Processing time
9. Green lines for feedrate and red lines for rapid traverse

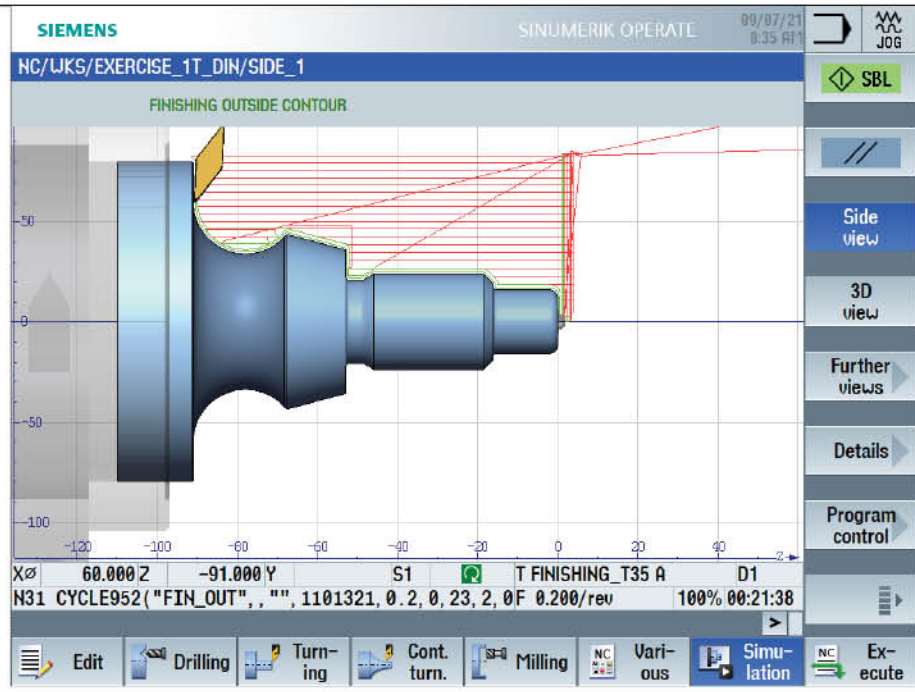
20.	Insert the following information message into the program: **N19 MSG("ROUGHING OUTSIDE RELIEF CONTOUR")**
21. **Select tool**	Now, insert a new tool into the program. Press the VSK 1.1 “Select tool“ and the tool list window opens. Mark the desired tool “**FINISHING_T35 A"** in the tool list window with the orange selection cursor, by using the blue cursor keys on the keyboard.

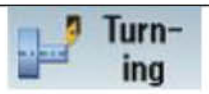
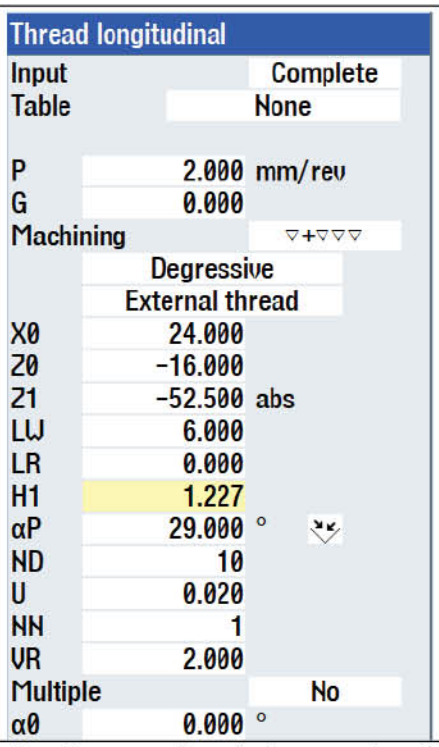
	The following program line is inserted into the program with “OK”: **N20 T="FINISHING_T35 A”** Alternatively you can program the tool by hand.
22.	Insert the following G code commands into the program: **N21 G96 S200 M4** **N22 G95 F0.2**
23.	Insert the start position of the stock removal with rapid traverse: **N23 G0 X82 Z5 M8**
24.  	Program now the stock removal cycle “CYCLE952”. Press the HSK 1.4 “Cont. turn.” to open the technology “Contour turning”. Press the VSK 3 "Cut residual stock" to the input screen for the technology "Cut residual material" to open.
25.	Enter the following values into the input mask. Use the “Select” key where indicated. PRG: Name of program to be generated CON: Name for updated blank contour for residual material machining Note: “CON” was created by the CNC control in the work sequence 16 in NC block “N17”.
26. 	Confirm your inputs by pressing the VSK 8 “Accept”. The following program code is inserted into the program: **N25** **CYCLE952("RH_OUT_2","RES_OUT_1","",1101311,0.2,0.1,0,2,0.1,0.1,0.5,0.1,0.1,0,1,0,,,,,,2,2,,,0,1,,0,112,1100010,1,0)**
27.	Start the simulation for the work sequences "Cut residual stock": Press the HSK 1.7 “Simulation” to open the program check.

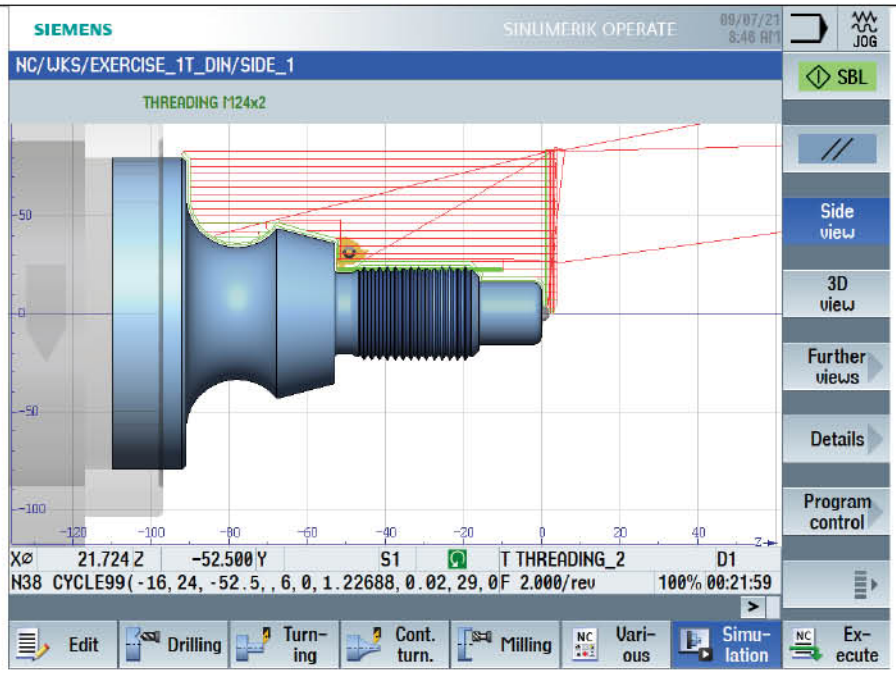


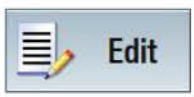
Simu-lation	
28. Edit	Press the HSK 1.1 “Edit” to open the edit mode again.
29.	Insert the following information message into the program: **N27 MSG("FINISHING OUTSIDE CONTOUR")**
30.	Insert the following G code commands into the program: **N28 G96 S250 M4** **N29 G95 F0.1**
31.	Insert the start position of the stock removal with rapid traverse: **N30 G0 X82 Z5 M8**
32. Cont. turn. Stock removal	Program now the stock removal cycle “CYCLE952”. Press the HSK 1.4 “Cont. turn.” to open the technology “Contour turning”. Press the VSK 2 "Stock removal" to the input screen for the technology "removal" to open.

33.	Enter the following values into the input mask. Use the “Select” key where indicated. PRG: Name of program to be generated 
34. 	Confirm your inputs by pressing the VSK 8 “Accept”. The following program code is inserted into the program: **N31** **CYCLE952("FIN_OUT",,"",1101321,0.2,0,23,2,0.1,0.1,0.5,0.1,0.1,0,1,0,0,,,,,2,2,,,0,1,,0,12,1100010,1,0)** Add the tool changing point into the program: **N32 WWP**
35. 	Start the simulation for the work sequences "Cut residual stock" for finishing: Press the HSK 1.7 “Simulation” to open the program check.

	
36. 	Press the HSK 1.1 “Edit” to open the edit mode again.
37.	Insert the following information message into the program: **N33 MSG("THREADING M24x2")**
38. 	Now, insert a new tool into the program. Press the VSK 1.1 “Select tool“ and the tool list window opens. Mark the desired tool **"THREADING_2"**in the tool list window with the orange selection cursor, by using the blue cursor keys on the keyboard. The following program line is inserted into the program with “OK”: **N34 T="THREADING_2"** Alternatively you can program the tool by hand.
39.	Insert the following G code commands into the program: **N28 G96 S250 M4** **N29 G95 F0.1**
40.	Insert the start position of the thread longitudinal with rapid traverse: **N35 G97 S1592 M3** **N36 G0 X26 Z5 M8**
41.	Program now the thread longitudinal cycle “CYCLE99”. Press the HSK 1.3 “Turning” to open the technology “Thread”.

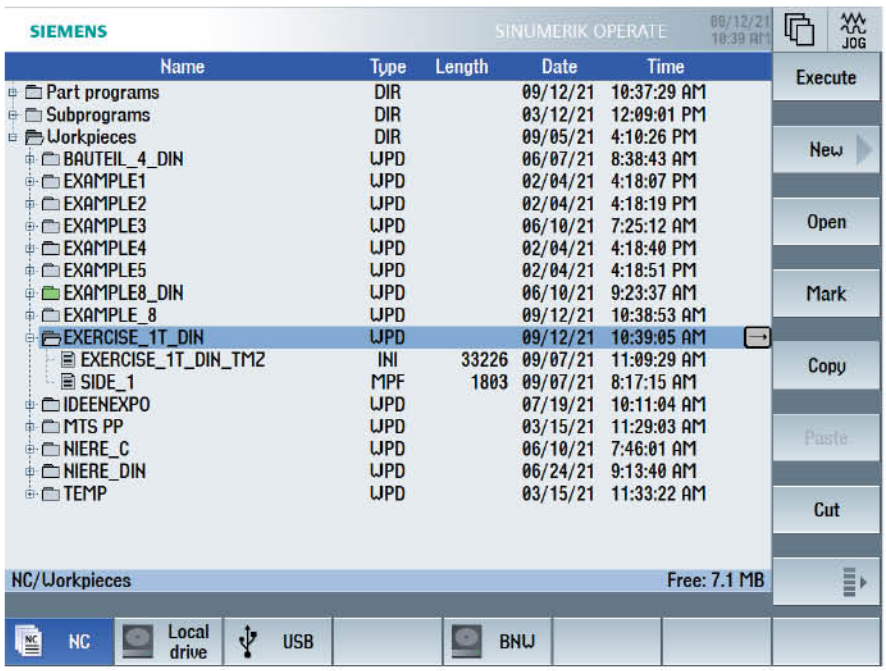
 Thread Thread long.	Press the VSK 4 "Thread" to the input screen. For the technology "Thread longitudinal" press VSK 3" to open.
42.	 Enter the following values into the input mask. Use the "Select" key where indicated.
43. 	Confirm your inputs by pressing the VSK 8 "Accept". The following program code is inserted into the program: **N38 CYCLE99(-16,24,-52.5,,6,0,1.22688,0.02,29,0,10,1,2,1300203,4,2,0.201147,0.5,0,0,1,0,0.680071,1,,,,102,0)** Add the tool changing point into the program: **N39 WWP** Note: The end of program command "M30" already existed with the input of the "Labels" for the cotour description. **N40 M30**
44. 	Start the simulation for the work sequences " Thread longitudinal ": Press the HSK 1.7 "Simulation" to open the program check.

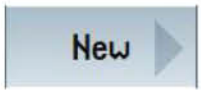
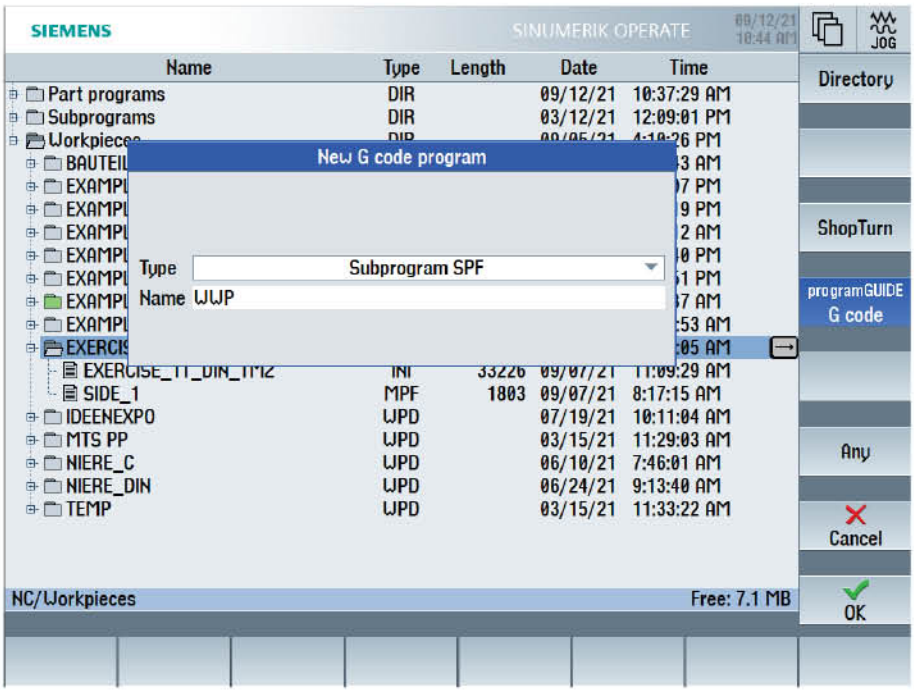
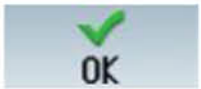


45. 	Press the HSK 1.1 “Edit” to open the edit mode again.

f. Subprogram “WWP” for the tool changing point

46.	Open the “Workpiece” drive the “EXERCISE_1_T_DIN.WPD”.



47. 	Press the VSK 1.2 “NEW” to open the input dialog for a new program.
48.	Press the VSK 1.4 “programGuide” and chose in the pull down menu “Type” the “Subprogram SPF” and enter the name “WWP”. 
49. 	Now, open the new program. The “WWP” program opens automatically with “OK”:
50.	Now, insert the following NC blocks into the program. NC/WKS/EXERCISE_1T_DIN/WWP.SPF N1 G0 G18 G40 SUPA X350 Z450 D0¶ N2 M17¶

3.9.2. Milling application using ProgramGUIDE

a. Description:

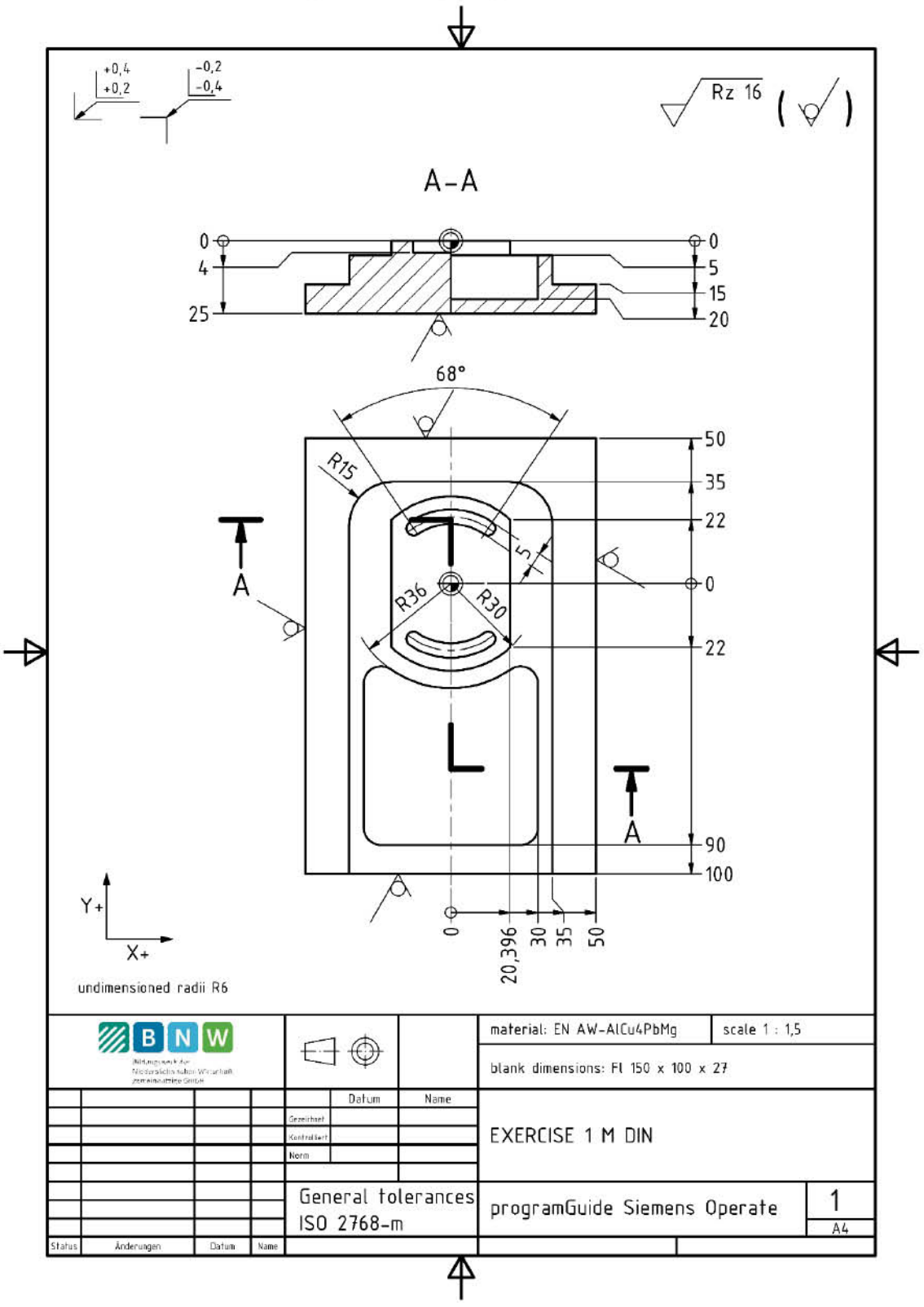
A first programGuide exercise is to be programmed with the help of roughing and finishing modes with different cycle calls.

b. Objective:

A new EXERCISE_1M_DIN program should be created and opened in the editor.

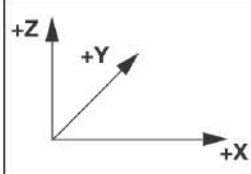
The program header, face milling, path milling, spigot milling with blank and finished contours, pocket milling, slot milling as well chamfer milling and end of the program to be programmed and subsequently simulated.

c. The technical drawing EXERCISE_1M_DIN:



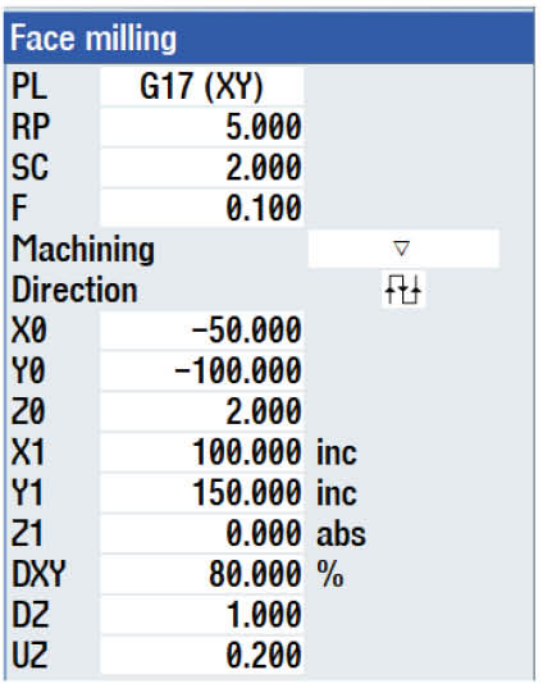
d. The following tools and technology data are required for programming:

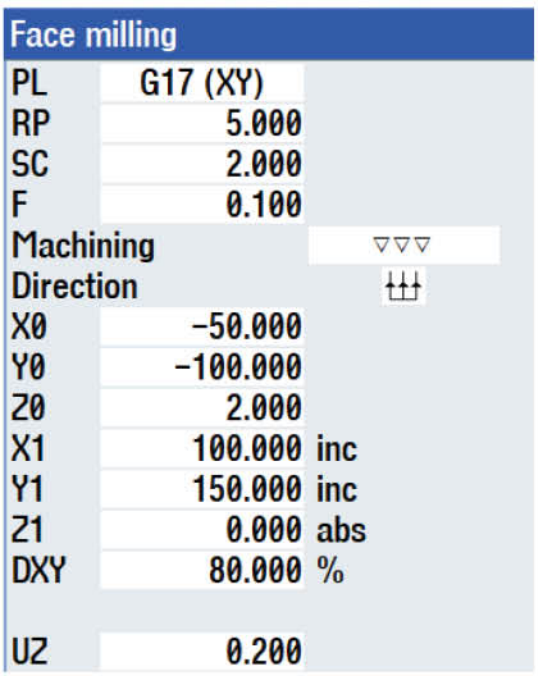
Workpiece: EXERCISE 1M DIN	Blank material: EN AW-AlCu4PbMg	Machine configuration:
Drawing: 1(1)	Blank dimensions: Fl 150 x 100 x 27	Milling 3 axes

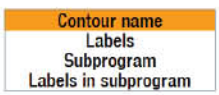
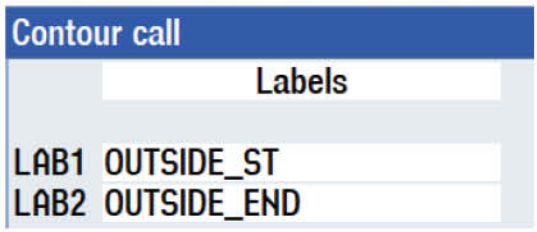
	Technological data							
Tool name in OFFSET	Spot_drill_12	Deburrer_8	Cutter_32	Cutter_16	Cutter_10	Cutter_4	Drill_8,5	Tap_10
Tool Ø (mm)	12 x 90°	8 x 90°	32	16	10	4	8,5	M10
Cutting speed (m/min)	80	105	150 / 175	150 / 175	150 / 175	150 / 175	60	10
Cutting depth a_p = max.	5	3	5	8	5	2		
Cutting material	HSS	SC	SC	SC	SC	SC	HSS	HSS
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,05 / 0,04	0,03 / 0,02	0,02 / 0,01	0,1	F = P
Number of teeth N		4	2	4	4	4		
								
				✓	✓	✓		

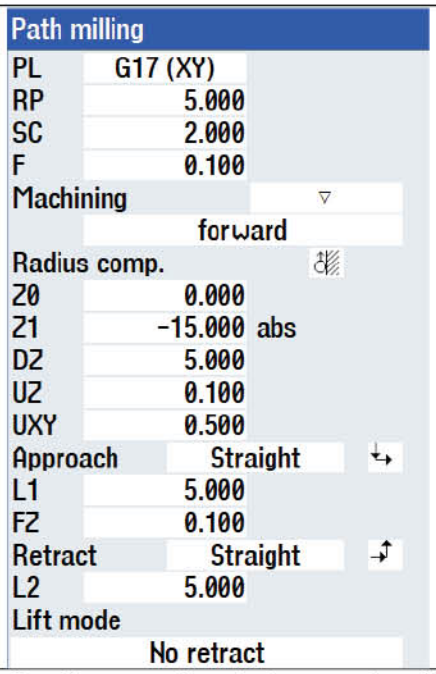
e. Create a new G code programGuide program, like described in this unit.
Create a new program directory and call the program for example "EXERCISE_1M_DIN.MPF".

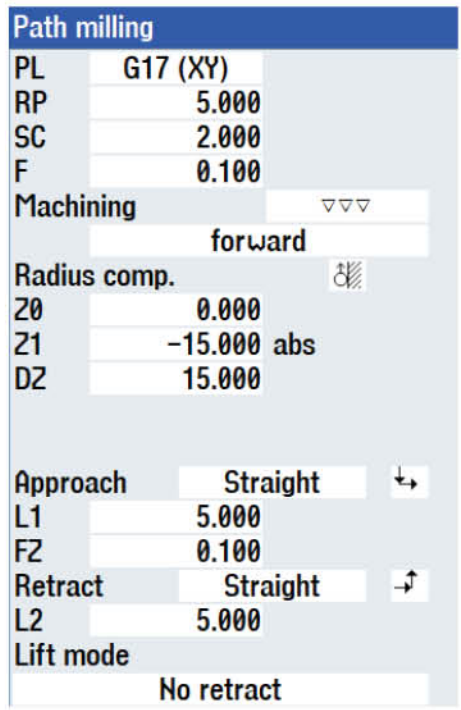
1.	Program the first lines of the program by inserting the following G code commands: **N1 G17 G64 G71 G90**
2. 	Insert now a blank for the simulation into the program. Press the HSK 1.6 “Various” to open the work area “Various”. The following input mask for the blank input opens.
3.	Enter following values/parameters for the blank: Blank Block X0 -50.000 Y0 -100.000 X1 50.000 abs Y1 50.000 abs ZA 2.000 ZI -25.000 abs
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N2 WORKPIECE(,"",,"BOX",112,2,-25,-80,-50,-100,50,50)**
4.	Now insert the following program lines for the programming conditions and to create a jump destination for the simulation. **N3 G54** **N4 ;GOTOF SIM_ST ;name of jump destination without double point**
5. 	Now, insert a new tool into the program. Press the VSK 1.1 “Select tool“ and the tool list window opens. Mark the desired tool “T="CUTTER 32” in the tool list window with the orange selection cursor, by using the blue cursor keys on the keyboard. The following program line is inserted into the program with “OK”: **N5 “T="CUTTER 32”** To carry out the tool change inserted into the program: **N6 M6**
6.	Insert the following G code commands into the program: **N7 G97 S1492 M3**

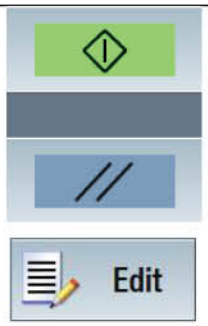
	N8 G95 FZ=0.1
7.	Insert the following information message into the program: **N9 MSG("FACE ROUGH HIGHT 25mm")**
8.	Insert the start position face milling with rapid traverse: **N10 G0 X-50 Y-25 Z2**
9. 	Insert a “Face milling” cycle into the program. Press the HSK 1.3 “Milling”, to open the technology “Milling”. Press the VSK 1 “Face milling”. The work area with the functions for the technology “Face milling” opens.
10. 	Enter the following values into the input mask. Use the “Select” key where indicated.  Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N11 CYCLE61(5,2,2,0,-50,-100,100,150,1,80,0.2,0.1,41,0,1,10)**
11.	Insert the following space block and information message into the program: **N12 ;*** N13 MSG("FACE FINISH")**

12.	Insert the following technology commands into the program: **N14 S1741 FZ=0.08**
13. 	Insert a second “Face milling” cycle for finishing into the program. Press the HSK 1.3 “Milling”, to open the technology “Milling”. Press the VSK 1 “Face milling”. The work area with the functions for the technology “Face milling” opens. Note: For the finishing of the surface we enter the “Same machining direction” different than before with “Changing machining direction”.
14. 	Enter the following values into the input mask. Use the “Select” key where indicated.  Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N15 CYCLE61(5,2,2,0,-50,-100,100,150,1,80,0.2,0.1,22,0,1,10)**
15.	Insert the following space block and information message into the program: **N16 ;*** N17 MSG("OUTSIDE ROUGH")**

16.	Insert the following technology commands into the program: **N18 S1492 FZ=0.1**
17. 	Program now the call up of the contour “CYCLE62”. Note: The contour itself is programmed at the end of the program. For this, press the HSK 1.4 “Cont. mill.” to open the technology “Contour milling”. Press now the VSK 1 “Contour”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
18. 	Insert the following “Labels” name into the input mask:  Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N19 CYCLE62(,2,"OUTSIDE_ST","OUTSIDE_END")** Note 1: The contour selection for the “Contour call” you call up here has four different choices for programming: • Contour name (graphic editor) • Labels (G code) • Subprogram • Labels in subprogram Note 2: The “Labels” with the contour description you called up here, will be inputted behind the” M30”.
19. 	Program now the path milling cycle “CYCLE72”. Press the HSK 1.4 “Cont. mill.” to open the technology “Contour milling”. Press the VSK 2 "Path milling" to the input screen for the technology to open.

20.		Enter the following values into the input mask: Use the “Select” key where indicated.
21.		Confirm your inputs by pressing the VSK 8 “Accept”. The following program code is inserted into the program: **N20 CYCLE72("",5,0,2,-** **15,5,0.5,0.1,0.1,0.1,301,41,1,5,0.1,1,5,0,1,2,101,1011,100)**
22.		Enter now the finished contour behind the "M30" between the "Labels" you called in the “CYCLE62” by using DIN/ISO G code programming: **N98 OUTSIDE_ST:** **N99 G0 X-35 Y-100** **N100 G1 Y35 RND=15** **N101 G1 X35 RND=15** **N102 G1 Y-100** **N103 OUTSIDE_END:** Note: For using the “Path milling” cycle the contour description can be an open or closed one.
23.		Insert the following space block and information message into the program: **N21 ;***

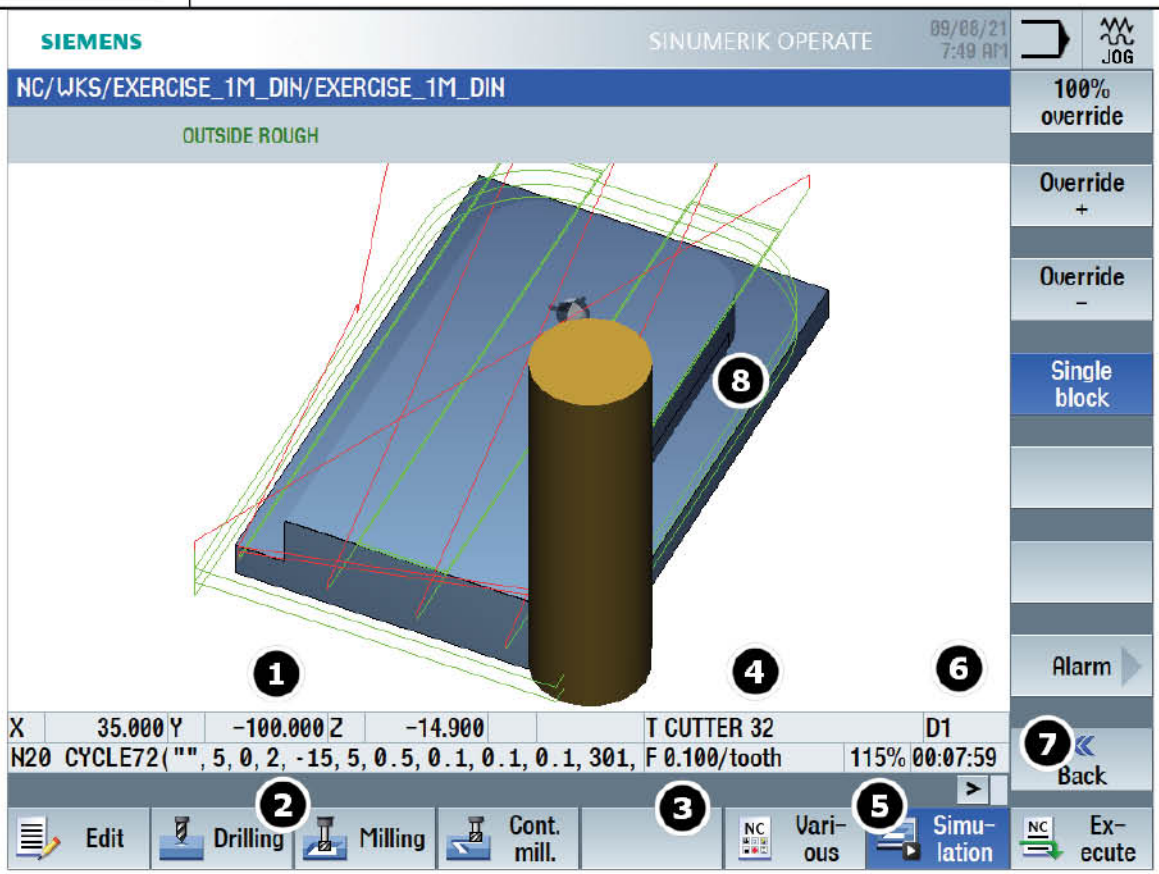
25. 	Program now the path milling cycle “CYCLE72”. Press the HSK 1.4 “Cont. mill.” to open the technology “Contour milling”. Press the VSK 2 "Path milling" to the input screen for the technology to open. Note: To continue “Path milling” in finishing mode following it´s not necessary to call the contour with CYCLE62 again. If a contour call before the cycle is not programmed programGuide is automatically using the last programmed CYCLE62 in the program.
26.	Enter the following values into the input mask. Use the “Select” key where indicated. 
27. 	Confirm your inputs by pressing the VSK 8 “Accept”. The following program code is inserted into the program: **N24 CYCLE72("",5,0,2,-** **15,15,0.5,0.1,0.1,0.1,302,41,1,5,0.1,1,5,0,1,2,101,1011,100)**
28. 	Start the simulation for the work sequences face and path milling: Press the HSK 1.7 “Simulation” to open the program check. Note 1: With the “Reset” and “NC-Start” buttons you can repeat your program



check with the simulation several times.

Press the HSK 1.1 “Edit” to open the edit mode again.

Note 2: It´s important to leave the simulation screen with the function “Edit”.



The simulation screen shows following details:

1. Current tool path position
2. Current NC block in the program
3. Feedrate or rapid traverse
4. Active tool in processing
5. Override speed for simulation (max. 120%)
6. Active cut of the current tool
7. Processing time
8. Green lines for feedrate and red lines for rapid traverse

29.	Insert the following space block and information message into the program: **N25 ;*****

Accept	Contour call Labels LAB1 PIN_ST LAB2 PIN_END Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N29 CYCLE62(,2,"PIN_ST","PIN_END")** Note: The contour for “Spigot milling” needs to be always a closed one.
35. Cont. mill. Spigot	Program now the spigot milling cycle “CYCLE63”. Press the HSK 1.4 “Cont. mill.” to open the technology “Contour milling”. Press the VSK 6 "Spigot" to the input screen for the technology to open.
36.	Enter the following values into the input mask. Use the “Select” key where indicated. PRG: Name of program to be generated
37. Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following program code is inserted into the program: **N30 CYCLE63("PIN_ROUGH",1,5,0,2,-5,0.1,,40,5,0.5,0.1,0,,,,,,1,2,,,,0,201,110)**

Mill spigot

Input		Complete
PRG	PIN_ROUGH	
PL	G17 (XY)	Down-cut
RP	5.000	
SC	2.000	
F	0.100	
Machining		▽
Z0	0.000	
Z1	-5.000	abs
DXY	40.000	%
DZ	5.000	
UXY	0.500	
UZ	0.100	
Lift mode		
	To RP	

38.	Enter now the blank contour behind the "M30" between the "Labels" you called in the “CYCLE62” by using DIN/ISO G code programming: **N104 BORDER_ST:** **N105 G0 X-35 Y-100** **N106 G1 Y35 ;RND=15** **N107 G1 X35 ;RND=15** **N108 G1 Y-100** **N109 G1 X-35 Y-100** **N110 BORDER_END:** Note: In order to get no error messages the first NC block needs to be programmed with “G0”.
39.	Enter now the spigot contour behind the "M30" between the "Labels" you called in the “CYCLE62” by using DIN/ISO G code programming: **N111 PIN_ST:** **;attention: always first NC-block with G0** **N112 G0 X-20.369 Y0** **N113 G1 Y22 RND=0.4** **N114 G2 X20.369 Y22 CR=30 RND=0.4** **N115 G1 Y-22 RND=0.4** **N116 G2 X-20.369 Y-22 CR=30 RND=0.4** **N117 G1 Y0** **N118 PIN_END:**
40.	Insert the following space block and information message into the program: **N31 ;*** **N32 MSG("PIN FINISH BASE")**
41.	Insert the following technology commands into the program: **N33 S1741 FZ=0.08**
42. 	Program now the spigot milling cycle “CYCLE63” for finishing the base. Press the HSK 1.4 “Cont. mill.” to open the technology “Contour milling”. Press the VSK 6 "Spigot" to the input screen for the technology to open.

43.	**Mill spigot** Input — Complete PRG PIN_FIN_BA PL G17 (XY) — Down-cut RP 5.000 SC 2.000 F 0.100 Machining — ▽▽▽Base Z0 0.000 Z1 -5.000 abs DXY 80.000 % UXY 0.500 UZ 0.100 Lift mode To RP	Enter the following values into the input mask. Use the "Select" key where indicated. PRG: Name of program to be generated
44. 	Confirm your inputs by pressing the VSK 8 "Accept". The following program code is inserted into the program: **N34 CYCLE63("PIN_FIN_BA",3,5,0,2,-5,0.1,,80,5,0.5,0.1,0,,,,,,1,2,,,,0,201,110)**	
45.	Insert the following space block and information message into the program: **N35 ;***

47. Enter the following values into the input mask. Use the “Select” key where indicated.

Mill spigot		
Input		Complete
PRG	PIN_FIN_WA	
PL	G17 (XY)	Down-cut
RP	5.000	
SC	2.000	
F	0.100	
Machining		▽▽▽Wall
Z0	0.000	
Z1	-5.000	abs
DZ	5.000	
UXY	0.500	
Lift mode		
	To RP	

PRG: Name of program to be generated

48.

Confirm your inputs by pressing the VSK 8 “Accept”.

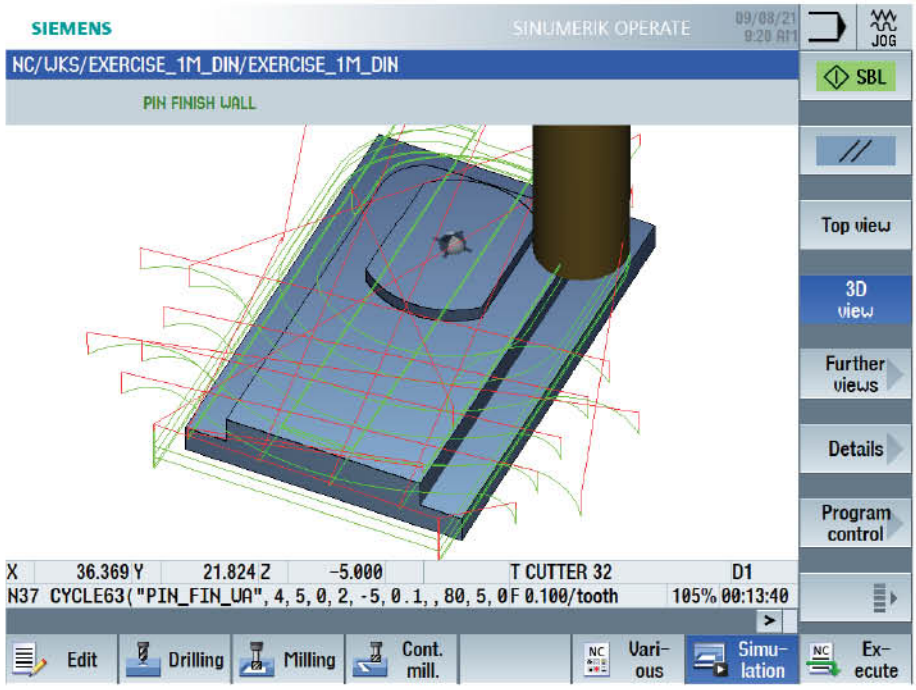
The following program code is inserted into the program:

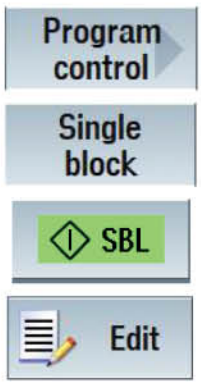
N37 CYCLE63("PIN_FIN_WA",4,5,0,2,-5,0.1,,80,5,0.5,0.1,0,,,,,,1,2,,,,0,201,110)

49.

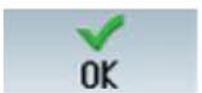
Start the simulation for the work sequences "Spigot milling":

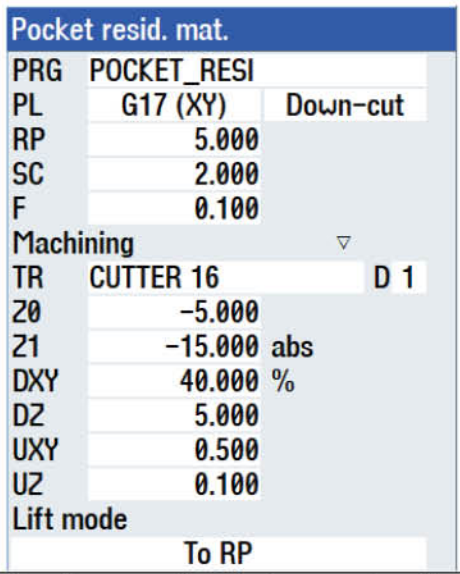
Press the HSK 1.7 “Simulation” to open the program check.

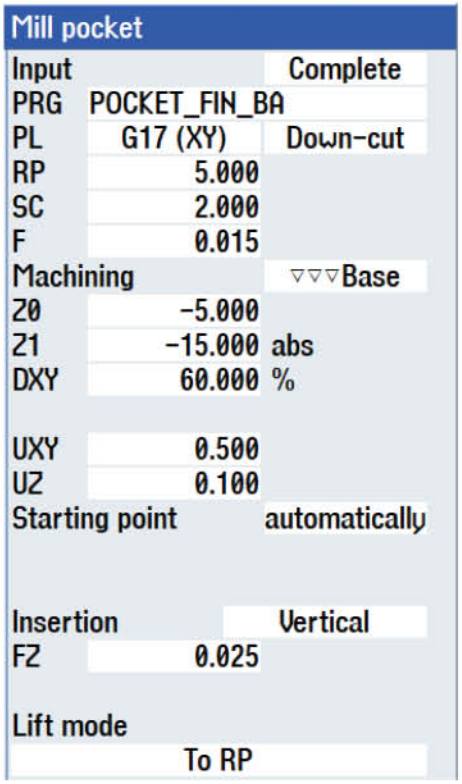


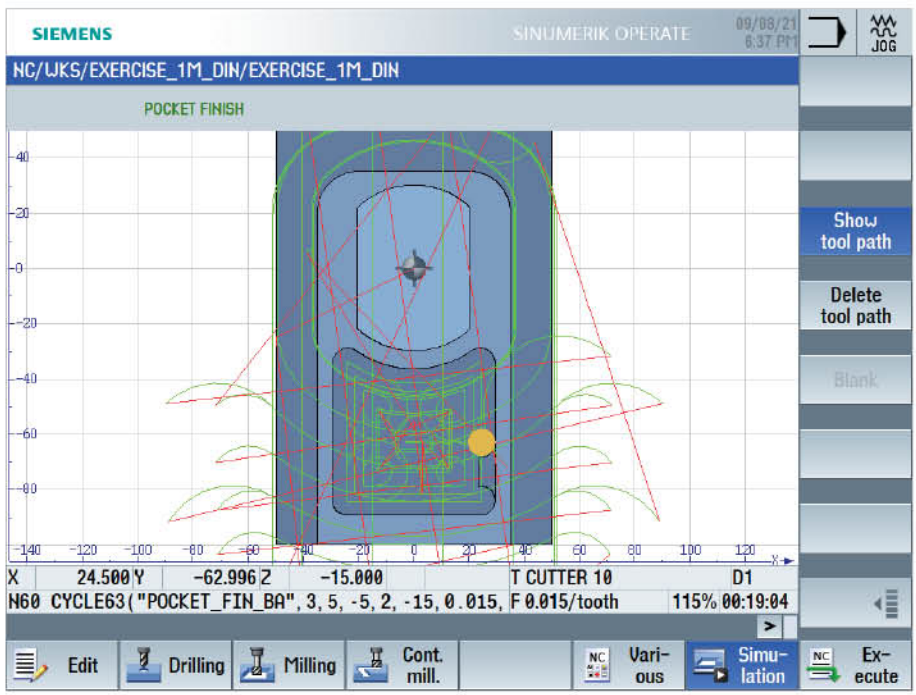
	Note 1: With the VSK 1.7 “Program control” and following VSK 2.4 “Single block” buttons you can control your program check during the simulation. Note 2: Press VSK 1.1 "SBL" to run the simulation NC block by NC block. Press the HSK 1.1 “Edit” to open the edit mode again. Note 2: It´s important to leave the simulation screen with the function “Edit”.
50.	Insert the following space block and jump destination into the program: **N38 ;***

Contour call	
56. Accept	Insert the following “Labels” name into the input mask: **Contour call** Labels LAB1 POCKET_ST LAB2 POCKET_END Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N46 CYCLE62(,2,"POCKET_ST","POCKET_END")** Note: The contour for “Pocket milling” needs to be always a closed one.
57. Back Pocket	Press the VSK 1.8 “Back”. Press the VSK 4 “Pocket” to open the input mask for the CYCLE63 “Mill pocket”.
58.	**Mill pocket** Input Complete PRG POCK_ROUGH PL G17 (XY) Down-cut RP 5.000 SC 2.000 F 0.050 Machining ▽ Z0 -5.000 Z1 -15.000 abs DXY 50.000 % DZ 5.000 UXY 0.500 UZ 0.100 Starting point automatically Insertion Helical EP 2.000 ER 6.000 Lift mode To RP Enter following values into the input mask. Use the “Select” key where indicated. PRG: Name of program to be generated

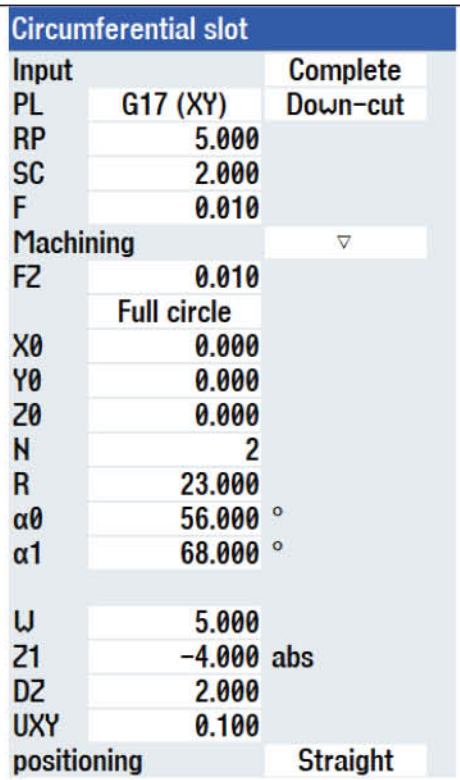
59. 	Confirm your inputs by pressing the VSK 8 “Accept”. The following program code is inserted into the program: **N47 CYCLE63("POCK_ROUGH",11,5,-5,2,-15,0.05,0.1,50,5,0.5,0.1,0,0,0,10,2,15,1,2,"",1,,0,101,110)**
60.	Enter now the pocket contour behind the "M30" between the "Labels" you called in the “CYCLE62” by using DIN/ISO G code programming: **N120 POCKET_ST:** **N121 G0 X0 Y-90** **N122 G1 X30 RND=6** **N123 G1 Y-19.9 RND=6** **N124 G2 X-30 Y-19.9 CR=36 RND=6** **N125 G1 Y-90 RND=6** **N126 G1 X0 Y-90** **N127 POCKET_END:** Note: In order to get no error messages the first NC block needs to be programmed with “G0”.
61.	Insert the following space block and information message into the program: **N48 ;********** **N49 MSG("POCKET RESIDUAL")**
62. 	First Press HSK 1 “Edit”. Vertical softkey bar with additional functions opens. Press the VSK 1.1 “Select tool”. The “Tool list” window opens. Use the blue cursor keys on the keyboard to select the tool “CUTTER 10”. Press the VSK 1.8 “OK”. The following program line is inserted into the program with “OK”: **N50 “T="CUTTER 10”** To carry out the tool change inserted into the program: **N51 M6**
63.	Insert the following G code commands into the program: **N52 S4775 M3** **N53 G95 FZ=0.03**
64.	Insert the start position for pocket milling with rapid traverse: **N54 G0 X0 Y-55 Z2**
65.	Insert now a “Pocket residual material” cycle (CYCLE63) for residual material “roughing” of the pocket into the program.

  	To do this, press the HSK 1.4 “Cont. Mill” to open the technology “Contour milling”. Press the VSK 5 “Pocket res.mat.” to open the input mask for the CYCLE63 “Pocket residual material”.
66.	Enter following values into the input mask. Use the “Select” key where indicated.  PRG: Name of program to be generated Note: With the blue “Select”-key the machining depth “Z1” can be set to “inc” or “abs” and the tool path step over “DXY” can be set to % of tool Ø or a value in mm of the tool Ø.
67. 	Confirm your inputs by pressing the VSK 8 “Accept”. The following line will be inserted into the program: **N55 CYCLE63("POCKET_RESI",1,5,-5,2,-15,0.1,,60,5,0.5,0.1,0,0,0,,,,,,"CUTTER 32",1,,0,1101,10)**
68.	Insert the following space block and information message into the program: **N56 ;********** **N57 MSG("POCKET FINISH")**
69.	Insert the following G code commands into the program: **N57 S5570 M3** **N58 G95 FZ=0.015**
70.	Insert the start position for pocket milling with rapid traverse: **N59 G0 X0 Y-55 Z2**
71. 	Press the VSK 4 “Pocket” to open the input mask for the CYCLE63 “Mill pocket”.

72.		Enter following values into the input mask. Use the "Select" key where indicated. PRG: Name of program to be generated
73.	Confirm your inputs by pressing the VSK 8 "Accept". The following line will be inserted into the program: **N60 CYCLE63("POCKET_FIN_BA",3,5,-5,2,-15,0.015,0.025,60,5,0.5,0.1,0,0,0,10,2,15,1,2,"",1,,0,101,110)**	
74.	Press the VSK 4 "Pocket" to open the input mask for the CYCLE63 "Mill pocket".	

75.	**Mill pocket** Input Complete PRG POCKET_FIN_WA PL G17 (XY) Down-cut RP 5.000 SC 2.000 F 0.015 Machining ▽▽▽Wall Z0 -5.000 Z1 -15.000 abs DZ 10.000 UXY 0.500 Lift mode To RP	Enter following values into the input mask. Use the "Select" key where indicated. PRG: Name of program to be generated
76.		Confirm your inputs by pressing the VSK 8 "Accept". The following line will be inserted into the program: **N61 CYCLE63("POCKET_FIN_WA",4,5,-5,2,-15,0.015,0.025,60,10,0.5,0.1,0,0,0,10,2,15,1,2,"",1,,0,101,110)**
77.		Start the simulation for the work sequences "Pocket milling": Press the HSK 1.7 "Simulation" to open the program check. 
78.	Insert the following space block, jump destination and information message into	

	the program: **N62 ;***

	
84. 	Confirm your inputs by pressing the VSK 8 “Accept”. The following line will be inserted into the program: **N71 SLOT2(5,0,2,-4,,2,68,5,0,0,23,56,90,0.005,0.01,2,0,0.1,1001,0,0,0,,0,1,2,100,1001,102)**
85.	Insert the following space block and information message into the program: **N72 ;***

Circumferential slot		
Input		Complete
PL	G17 (XY)	Down-cut
RP	5.000	
SC	2.000	
F	0.008	
Machining		▽▽▽
FZ	0.008	
	Full circle	
X0	0.000	
Y0	0.000	
Z0	0.000	
N	2	
R	23.000	
α0	56.000	°
α1	68.000	°
W	5.000	
Z1	-4.000	abs
DZ	2.000	
UXY	0.100	
positioning		Straight

Note: For “Slot milling” are four “Machining” modes available:

- Roughing
- Finishing base and wall together
- Finishing wall
- Chamfer milling

89.

Confirm your inputs by pressing the VSK 8 “Accept”.

The following line will be inserted into the program:

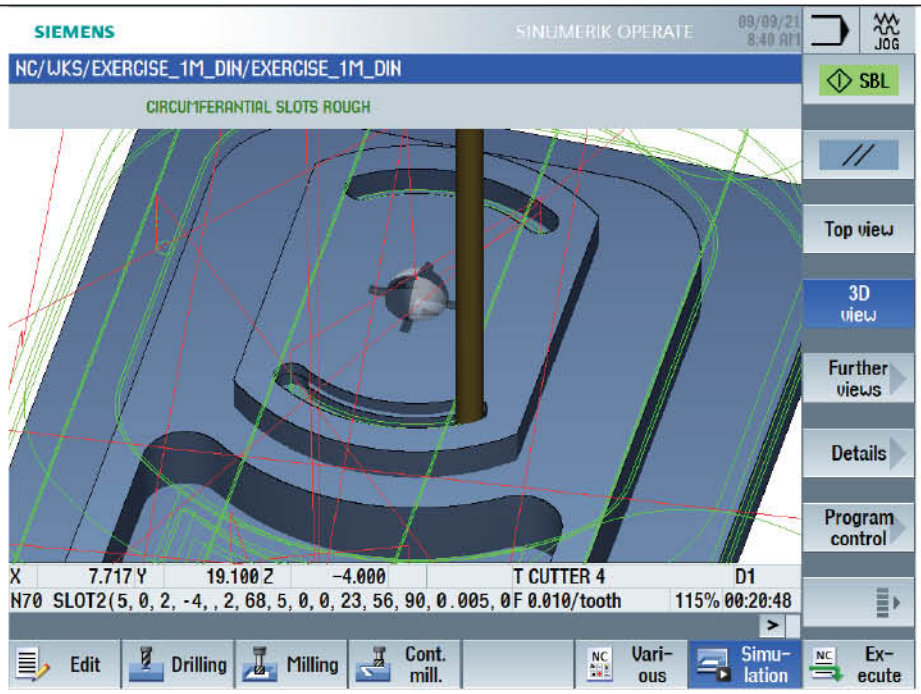
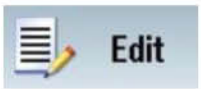
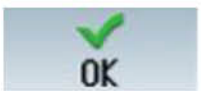
N76 SLOT2(5,0,2,-

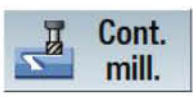
4,,2,68,5,0,0,23,56,90,0.008,0.008,2,0,0.1,1002,0,0,0,,0,1,2,100,1001,102)

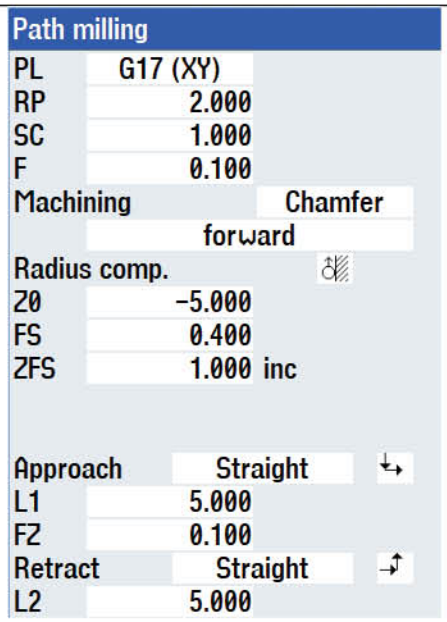
90.

Start the simulation for the work sequences "Slot milling":

Press the HSK 1.7 “Simulation” to open the program check.

	
91.  	First Press HSK 1 “Edit”. Vertical softkey bar with additional functions opens. Press the VSK 1.1 “Select tool”. The “Tool list” window opens. Use the blue cursor keys on the keyboard to select the tool “DEBURRER 8”. Press the VSK 1.8 “OK”. The following program line is inserted into the program with “OK”: **N78 “T="DEBURRER 8”** To carry out the tool change inserted into the program: **N79 M6**
92.	Insert the following G code commands into the program: **N80 S4178 F=0.1 M3**
93.	Insert the start position for chamfer milling with rapid traverse: **N81 G0 X-15 Y15 Z2**
94.	Insert the following information message into the program: **N82 MSG("OUTSIDE 2 CHAMFER 0,4x45°")**
95. 	Program now the call up of the contour “CYCLE62”. Note: The contour itself is programmed at the end of the program. For this, press the HSK 1.4 “Cont. mill.” to open the technology “Contour milling”. Press now the VSK 1 “Contour”.

 	The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
96. 	Insert the following “Labels” name into the input mask: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N83 CYCLE62(,2,"OUTSIDE_2_ST","OUTSIDE_2_END")**
97.	Enter now the path contour behind the "M30" between the "Labels" you called in the “CYCLE62” by using DIN/ISO G code programming: **N128 OUTSIDE_2_ST:** **N129 G0 X-35 Y-100** **N130 G1 Y35 RND=15** **N131 G1 X35 RND=15** **N132 G1 Y-100** **N133 G1 X-35** **N134 OUTSIDE_2_END:** Note: The order of the input of the “Path milling” and of the “Labels” behind the "M30" is arbitrary.
98.  	Program now the path milling cycle “CYCLE72”. Press the HSK 1.4 “Cont. mill.” to open the technology “Contour milling”. Press the VSK 2 "Path milling" to the input screen for the technology to open.
99.	Enter the following values into the input mask. Use the “Select” key where indicated.



100. Accept

Confirm your inputs by pressing the VSK 8 “Accept”.

The following program code is inserted into the program:

N84 CYCLE72("",2,-5,1,-

15,15,0.5,0.1,0.1,0.1,305,41,1,5,0.1,1,5,0,0.4,1,101,1011,100)

101. Insert the following information message into the program:

N85 MSG("SPIGOT CHAMFER 0,4x45°")

102. Contour

Contour call

Program now the call up of the contour “CYCLE62”.

Press now the VSK 1 “Contour”.

The vertical softkey bar with functions for creating or calling up a contour opens.

Press the VSK 2 “Contour call” to open the input mask for the contour call up.

103. Accept

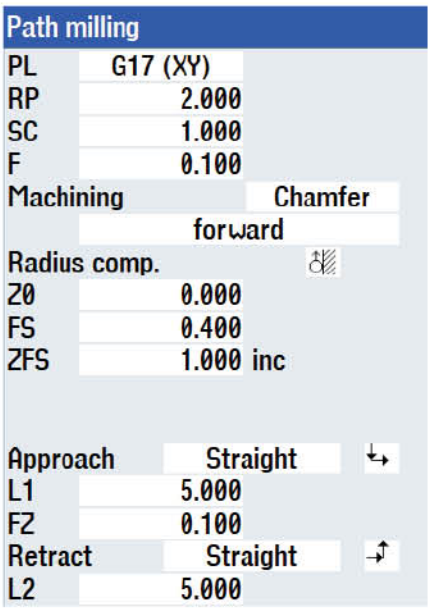
Insert the following “Labels” name into the input mask:

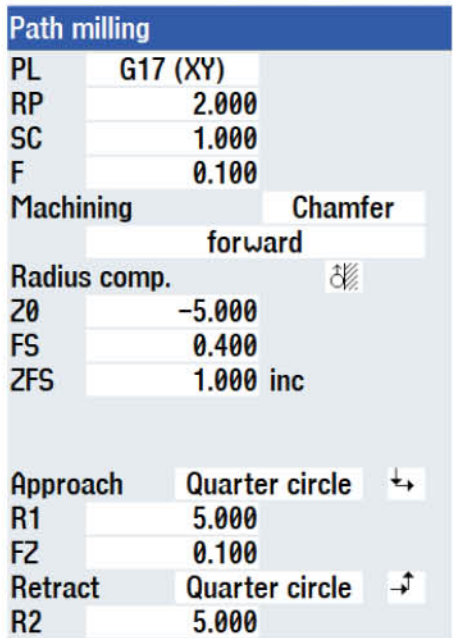
Confirm your input by pressing the VSK 8 “Accept”.

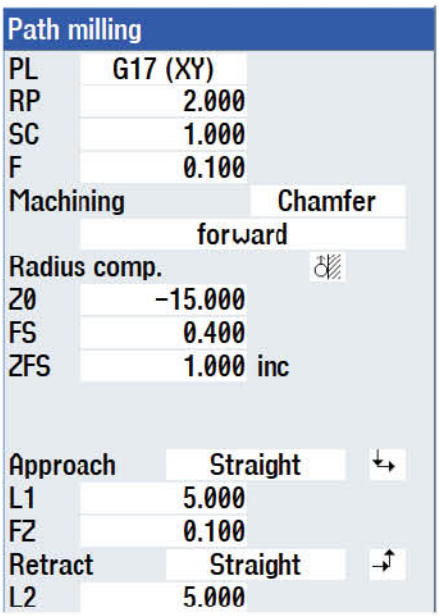
The following line is inserted into the program:

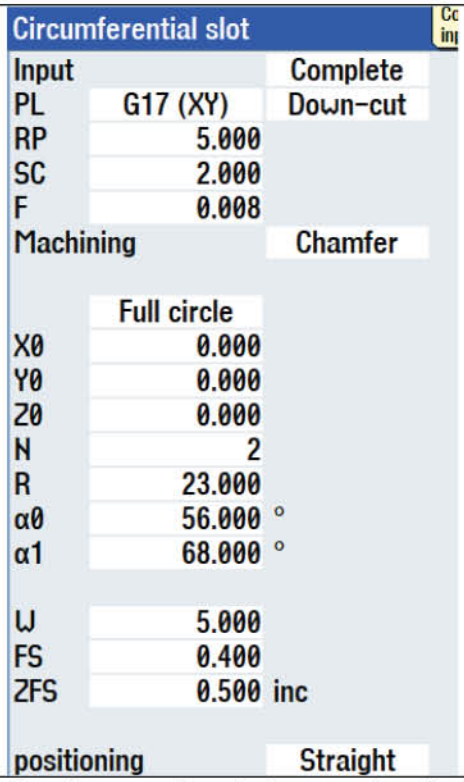
N86 CYCLE62(,2,"PIN_ST","PIN_END")

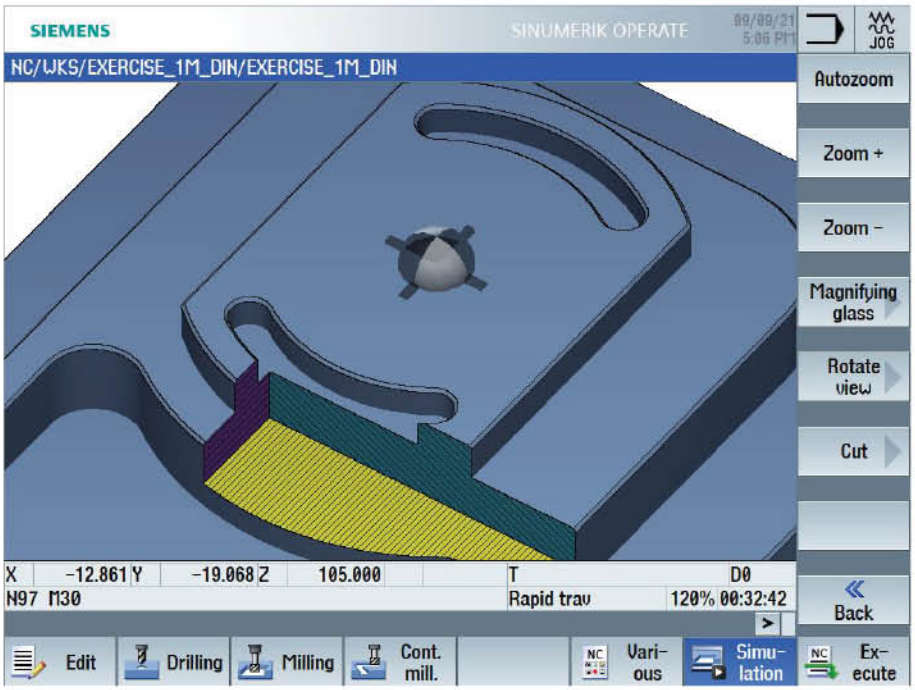
Note: This contour is already programmed.

104.	Enter the following values into the input mask. Use the "Select" key where indicated. 
105. 	Confirm your inputs by pressing the VSK 8 "Accept". The following program code is inserted into the program: **N87 CYCLE72("",2,0,1,-** **15,15,0.5,0.1,0.1,0.1,305,41,1,5,0.1,1,5,0,0.4,1,101,1011,100)**
106.	Insert the following information message into the program: **N88 MSG("POCKET CHAMFER 0,4x45°")**
107. 	Program now the call up of the contour "CYCLE62". Press now the VSK 1 "Contour". The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 "Contour call" to open the input mask for the contour call up.
108.	Insert the following "Labels" name into the input mask: Confirm your input by pressing the VSK 8 "Accept". The following line is inserted into the program:

Accept	**N89 CYCLE62(,2,"POCKET_ST","POCKET_END")** Note: This contour is already programmed.
109.	Enter the following values into the input mask. Use the "Select" key where indicated. 
110. Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following program code is inserted into the program: **N90 CYCLE72("",2,-5,1,-15,15,0.5,0.1,0.1,0.1,305,41,2,5,0.1,2,5,0,0.4,1,101,1011,100)**
111.	Insert the following information message into the program: **N91 MSG("BLANK RECTANGLE CHAMFER 0,4x45°")**
112. Contour Contour call	Program now the call up of the contour "CYCLE62". Press now the VSK 1 "Contour". The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 "Contour call" to open the input mask for the contour call up.
113.	Insert the following "Labels" name into the input mask:

Accept	 Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N92 CYCLE62(,2,"BLANK_ST","BLANK_END")**
114.	Enter the following values into the input mask. Use the “Select” key where indicated. 
115. Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following program code is inserted into the program: **N93 CYCLE72("",2,-15,1,-15,15,0.5,0.1,0.1,0.1,305,41,1,5,0.1,1,5,0,0.4,1,101,1011,100)**
116.	Enter now the path contour behind the "M30" between the "Labels" you called in the “CYCLE62” by using DIN/ISO G code programming: **N134 BLANK_ST:** **N135 G0 X-50 Y-100** **N136 G1 Y50** **N137 G1 X50** **N138 G1 Y-100** **N139 BLANK_END:**

117.	Insert the following information message into the program: **N94 MSG("CIRCUMFERANTIAL SLOTS CHAMFER 0,4x45°)**
118. 	Insert now a "circumferential slots" cycle (SLOT2) for "chamfering" of the two slots into the program. To do this, press the HSK 1.3 "Milling" to open the technology "Milling". Press the VSK 4 "Slot" to open the input mask for the "Slot milling cycles". Press the VSK 4 "Circumferential slot" to open the input mask for the "Circumferential slot cycle".
119.	 Enter the following values into the input mask. Use the "Select" key where indicated.
120. 	Confirm your inputs by pressing the VSK 8 "Accept". The following program code is inserted into the program: **N95 SLOT2(5,0,2,-4,,2,68,5,0,0,23,56,90,0.008,0.008,2,0,0.1,1005,0,0,0,,0,0.4,0.5,100,1001,102)**
121.	Finally, we enter the commands to empty the tool from the spindle above the already inputted "Program end". **N96 T0 M6**

	N97 M30
122. 	Start the simulation for the work sequences "Chamfer milling": Press the HSK 1.7 “Simulation” to open the program check. 

UNIT 2: ADVANCED DIN/ISO PROGRAMMING – CNC TURNING WITH DRIVEN TOOLS (C AND Y AXIS) AND COUNTER SPINDLE

1. **Introduction:** As the second unit of the module, the trainees will e familiar with programming method using programGUIDE to solve advanced applications regarding CNC turning with driven tools (C and Y axis) and counter spindle
2. **Objective:** By the end of this training unit, the trainees will be able to

- Understand planes in CNC programming with driving tools;
- Understand the process of preparing, programming, simulating, machining, editing and optimizing in CNC turning programs;
- Setup the tool offset, spindle chuck data and work offset;
- Program with machining control tools on the front surface of the workpiece;
- Program with machining control tools on the cylindrical surface of the workpiece;
- Program for milling, drilling with C and Y axis;
- Understand the working of CNC lathes with reverse axis and how to calculate for programming;
- Program with main spindles on lathes with counter spindle;
- Use Mechanical Trade and Metal Handbook.

3. **Content:**

3.1. Planes when working with driven tools

In a CNC programming there are 3 G-codes for plane selection during the NC programming that are used to define the two axes of either X, Y or Z. The plane selection is modal and stays in effect for all until you command another circular plane.

The 3 Plane selection G-Codes are:

G17 for XY Plane

G18 for XZ Plane

G19 for YZ Plane

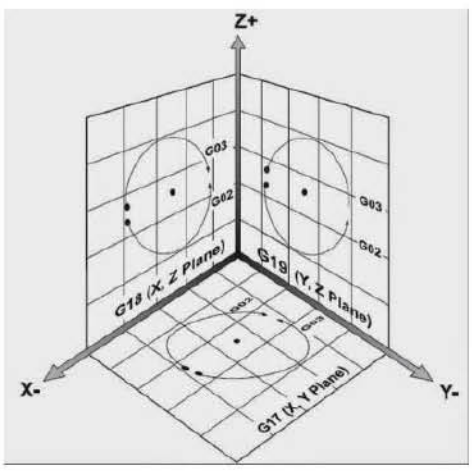


Figure 2.1: The 3 Plane selection G-Codes are G17, G18, G19

When machine is powered on, the default for plane selection is G17 for the XY plane. This means that the XY plane is already selected without programming in a G17, so the G17 XY-plane will always be active when you Power On the machine.

G18: Turning plane selection

Function

The command G18 defines the ZX plane as the turning plane with the 1st geometry axis Z and the 2nd geometry axis X between which circular interpolation takes place. The workpiece zero points remain unchanged. A rotating tool spindle will be stopped. The commands S and M for rotational speed and rotational direction are activated for the selected workpiece spindle after programming of the plane selection.

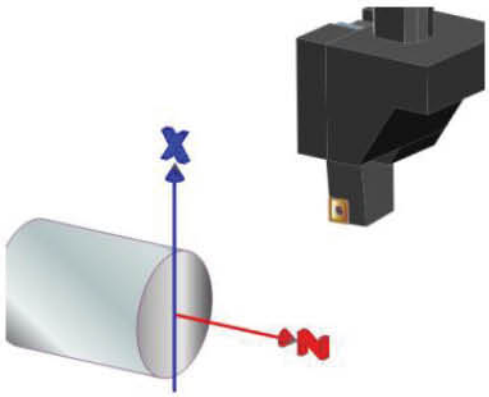


Figure 2.2: G18 - Turning plane selection

G17: Front face with polar coordinates

Function

Programming of the front face in pseudo polar coordinates X, C and infeed in Z.

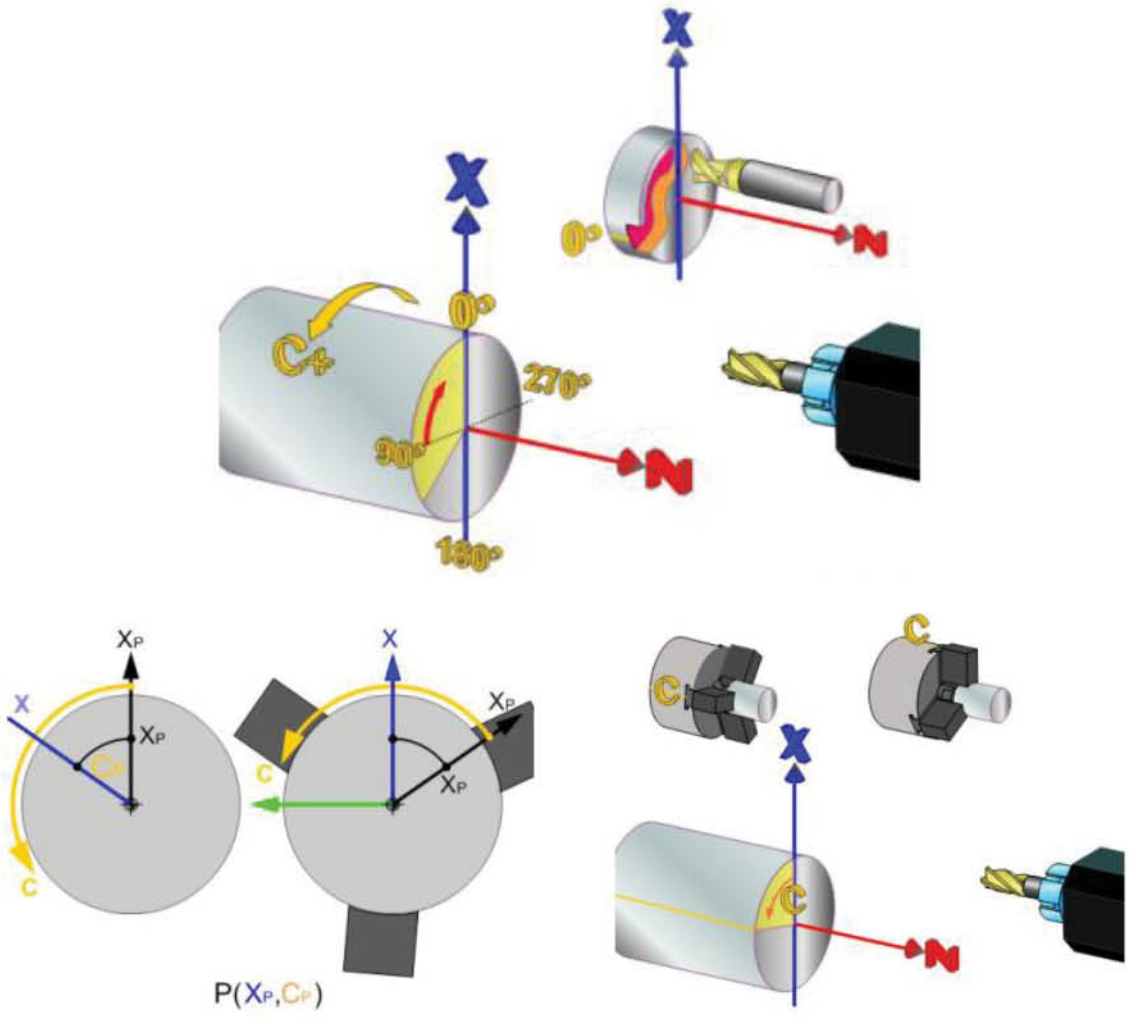


Figure 2.3: G17- Front face with polar coordinates

G17: Front face with virtual Y axis

Function

Programming of the front face machining with virtual Y-axis in a G17 coordinate system XYZ with infeed in Z and internal conversion of the XY coordinates to polar coordinates X, C .

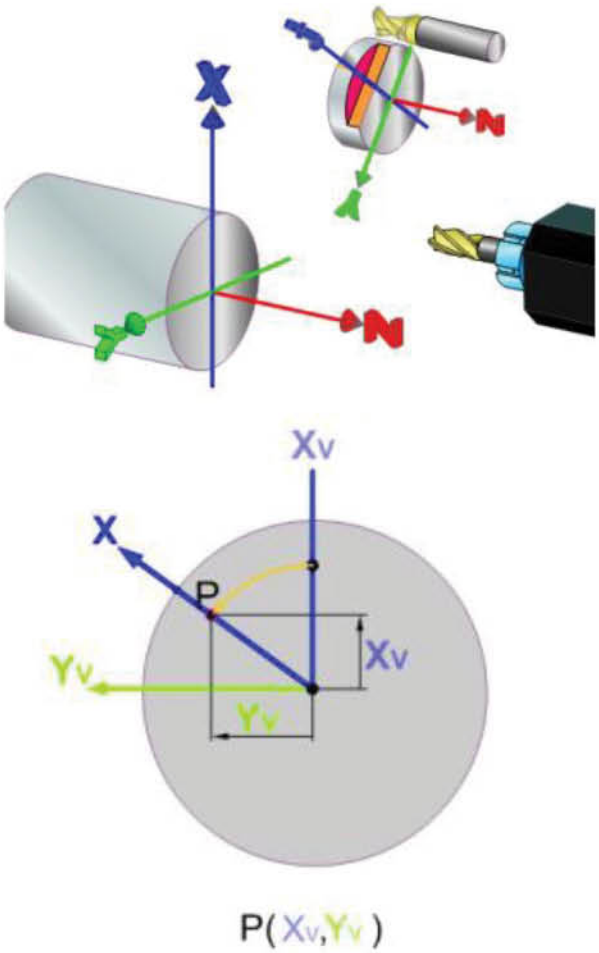


Figure 2.4: G17- Front face with virtual Y axis

G17: Front face machining with real Y axis

Function

Programming of the front face machining with real existing Y-axis in a G17 coordinate system while holding the C-axis on a fixed value

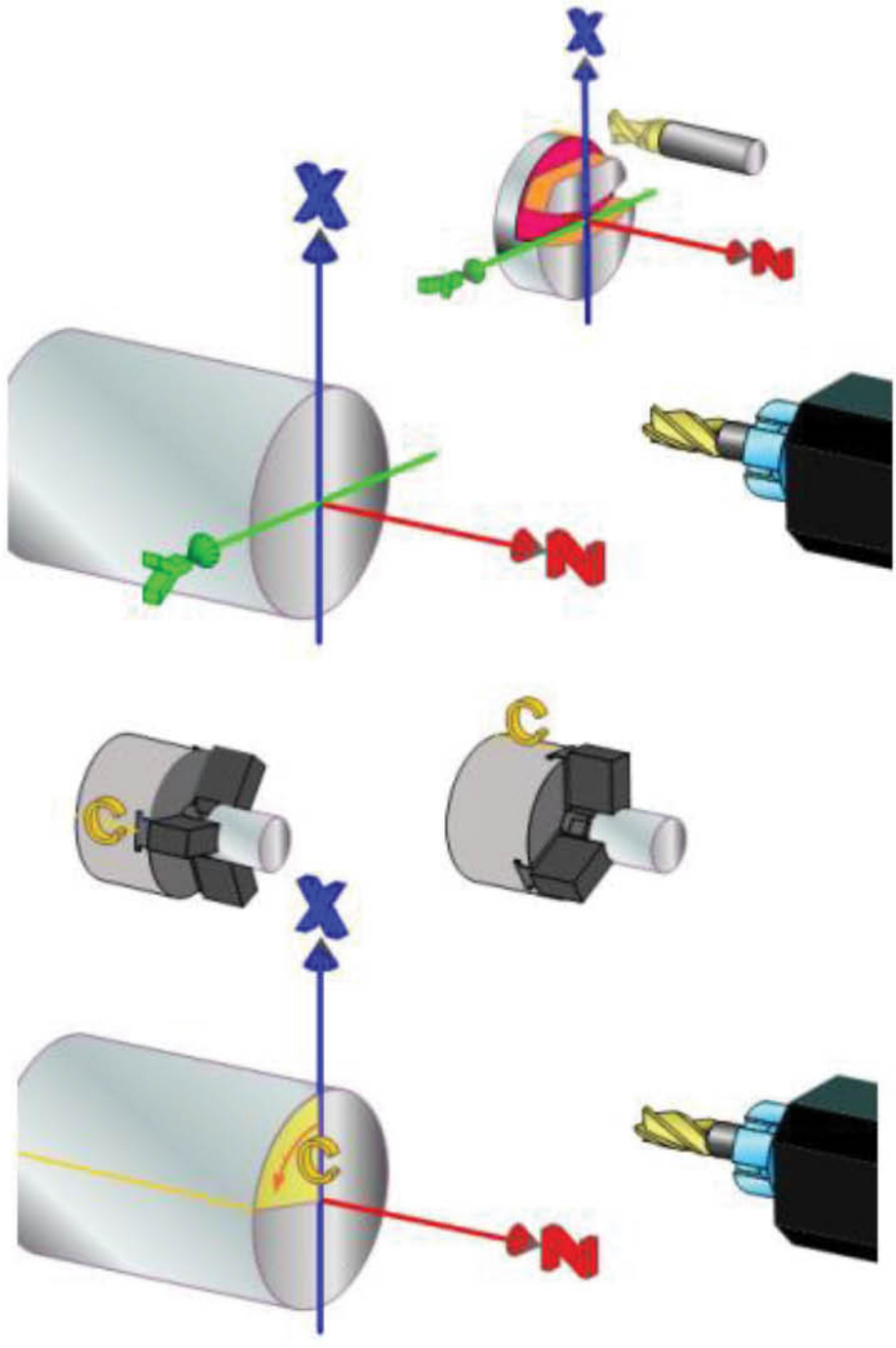


Figure 2.5: G17- Front face machining with real Y axis

G19: Cylinder surface with polar coordinates

Function

Programming the cylinder surface in cylinder coordinates Z, C and infeed in X

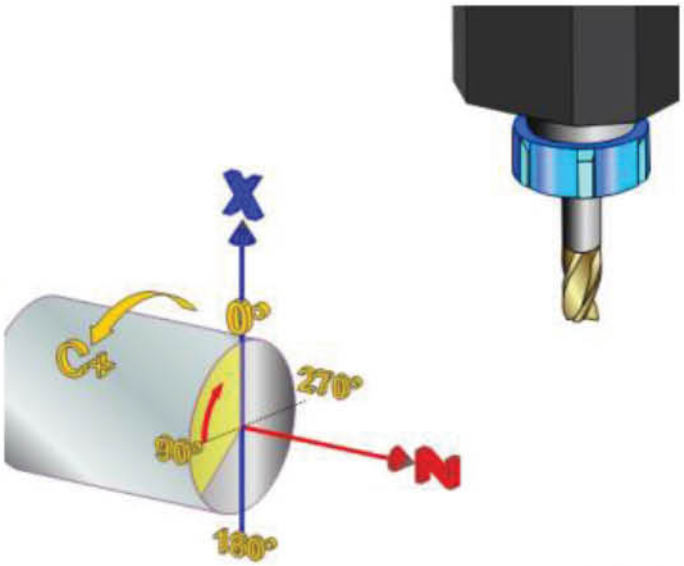


Figure 2.6: G19 - Cylinder surface with polar coordinates

G19: Cylinder surface with virtual Y axis

Function

Programming of the cylinder surface machining on the cylinder surface unwound in diameter X with virtual Y-axis in the G19 coordinate system YZX and internal conversion of the YZ coordinates in the cylindrical coordinates Z, C .

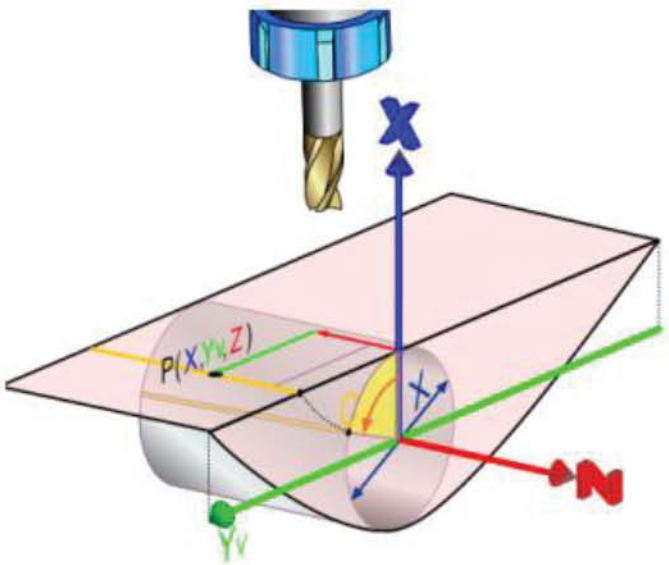


Figure 2.7: G19 - Cylinder surface with virtual Y axis

G19: Chord plane

Function

Programming of a chord surface with real existing Y-axis in a G19 coordinate system while holding the C-axis on a fixed value.

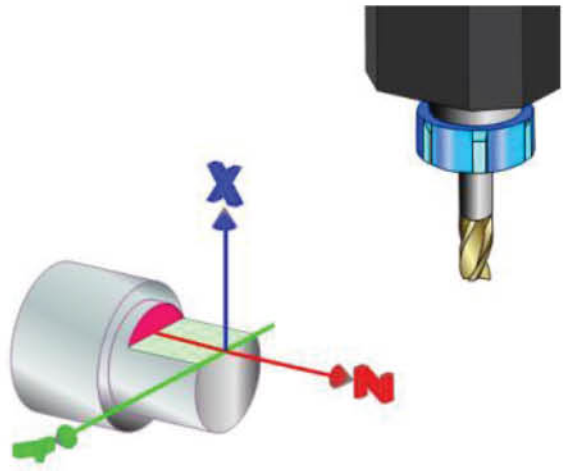


Figure 2.8: G19 - Chord plane

G19: Inclined chord plane

Function

Machining in a chord surface inclined by the angle B with real existing Y-axis and B axis (or in B adjustable tool) in the G19 coordinate system while holding the C-axis value. The workpiece coordinate system is rotated by the angle B. According to H the tool system (turret) is rotated around its B-axis so that the tool is perpendicular to the working plane. All the zero points remain thereby in their position. After selecting a different machining plane, the coordinate system rotation is cancelled.

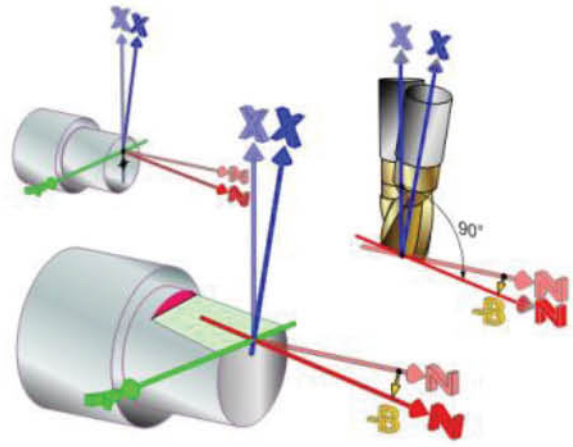


Figure 2.9: G19 - Inclined chord plane

3.2. Setup the tool offset, spindle chuck data and work offset

I. Administration of the tool offset

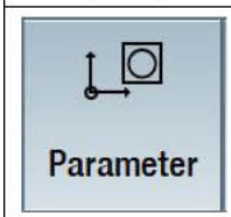
In the Operation area “Parameter” you have a choice of selecting between various lists (e.g. Tool list, tool wear, magazine list, offsets, user variables and setting data). For example, in the tool management area, all tools and if configured also the magazine locations are being displayed.

Both lists display the same tools in the same order. When switching between lists, the position of the cursor on a particular tool in the current screen is carried over to the same tool in a new screen. The lists differ from each other by the displayed parameters and the Softkey functions

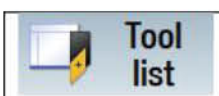
Switching between lists is a specific change from one topic to the next.

- Tool list: All parameters and functions required to create and set up tools are displayed.
- Tool wear: All parameters and functions that are required during operation, e.g. wear and monitoring functions, are listed here.
- Magazine: Magazine and magazine location-related parameters and functions for the tools and magazine locations are listed here.

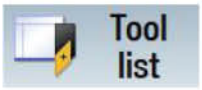
The operating area “Parameter“ can be opened from every operating modes (“JOG”, “MDA”, “AUTO”).

	Press the “OFFSET”-key on the keyboard. The operating area “Parameter“ respectively the “Tool list” opens directly.
	The operating area “Parameter” opens, with the “Tool list”, “Tool wear”, “Magazine”, “Work offset”, “User variables” and the “Setting data”. These functions are made available in the following described horizontal softkey bar.

Horizontal softkey bar (HSK) of the operating area “Parameter”

	By pressing the HSK 1 “Tool list” the tool list window opens for creating new tools or cutting edges (D1 – D9), delete tools, load or unload.
	By pressing the HSK 2 “Tool wear“ the tool wear list opens. Here it´s possible to input correction values of tool wear and “Tool data”.
	By pressing the HSK 4 “Magazine“ the Magazine management is opened. Here it´s possible to delete, load or unload all tools at the same time in one work step

Work offset	By pressing the HSK 5 "Work offset" a list with all Work offsets is opened. This window is generally used for monitoring: • Active work offset • Work offset overview • Base offset (G500) • Settable zero offset (Work offset - G54...G57)
R User variable	By pressing the HSK 6 "User variable" a list with all R variables is opened.
SD Setting data	By pressing the HSK 8 "Setting data" a list with all setting data is opened. Working area limit, spindle data and spindle chuck data are required set up here.

Vertical softkey bar (VSK) of the "Tool list" in the operating area 

"Parameter"

Tool measure	By pressing the VSK 1.1 "Tool measure" the "Measuring tool" window opens. With this function the correction values of tool can be determined.
New tool	By pressing the VSK 1.2 "New tool" a new tool can be created. This function is the cursor is positioned on a field that does not yet has a tool assigned to it.
Edges	By pressing the VSK 1.3 "Edges" the vertical softkey bar for assigning new cutting edges and deleting existing cutting edges opens to the right side of the screen. If a tool has several cutting edges each edge gets its own set of correction data.
Further data	By pressing the VSK 1.4 "Further data" more information about a tool will be displayed. This function is only available for tools which have additional information.
Unload Load	By pressing the VSKs 1.5 "Unload" or "Load" the actual selected tool will be unloaded from or loaded to the magazine. Unloaded tools are being displayed on the bottom of the magazine list.
Delete tool	By pressing the VSK 1.6 "Delete tool" the selected tool will be deleted from the tool list.

Magazine selection	By pressing the VSK 1.7 “Magazine selection” Ssoftkey multiple times you can jump between buffer location (spindle and gripper), Magazine and NC-memory (unloaded tools) and back to the buffer location. The cursor is always positioned at the beginning of each group.
	By pressing the VSK 1.8 “Extend” on the operator panel the extended vertical softkey bar 2 opens on the right hand side of the screen.

Vertical softkey bar (VSK) of the “Extendet” in the “Tool list”

Sort	By pressing the VSK 2.1 “Sort” you can sort the tools in the tool list according to the following criteria: • Magazine • Name • Type • T-Number The suitable softkeys are offered in the vertical softkey bar.
Filter	Pressing the VSK 2.2 “Filter” opens the screen to set the filter options.
Search	Pressing the VSK 2.3 “Search“ opens a new softkey bar with the functions • Tool • Magazine location • Empty location The type of search is selected by the sotfkey selection.
Details	Pressing the VSK 2.4 „Details“ opens a new softkey bar with the functions • Tool Data • Cutting edge data • Monitoring data The details of the tool which is selected by the cursor position are listed in the screen.
	Pressing the VSK 2.8 “Back“ on the operator panel you switch back to the vertical softkey bar 1.

II. Administration of the work offset

Following reference point approach, the actual value display for the axis coordinates is

based on the machine zero (M) of the machine coordinate system (MCS). The program for machining the workpiece, however, is based on the workpiece zero (W) of the workpiece coordinate system (WCS). The machine zero and workpiece zero are not necessarily identical.

The distance between the machine zero and workpiece vary in accordance with the type of tool and the way it is clamped. This zero offset is taken into account during execution of the program and can be a combination of different offsets.

On the Sinumerik Operate, the position actual value display refers to the SZS-coordinate system (settable zero system). The position of the active tool relative to the workpiece zero is displayed.

The offsets are added as follows:

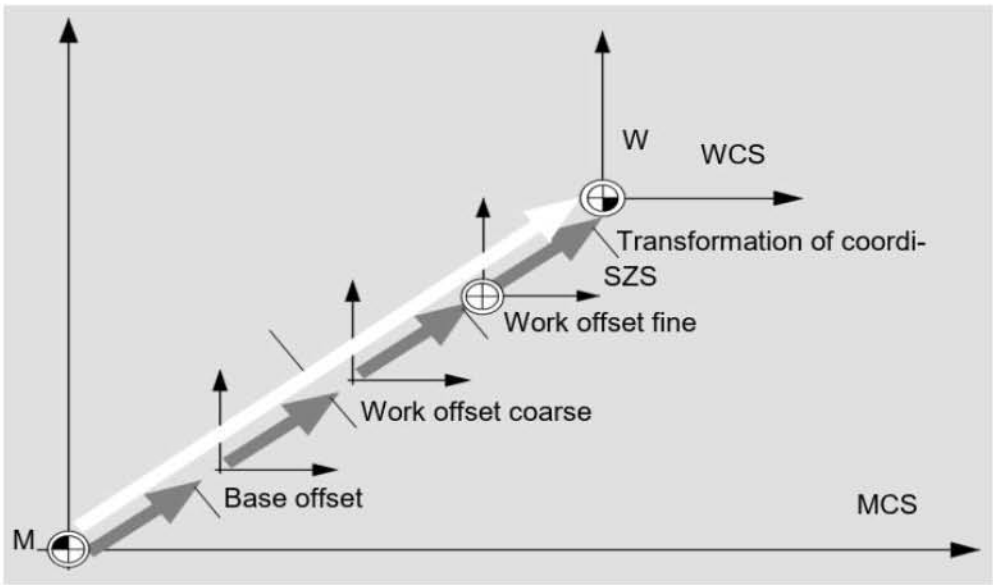


Figure 2.10: Administration of the work offset

Base offset:

The base offset is a zero offset that is always active. If you have not defined a base offset, its value will be zero. You determine the base offset via "Measure workpiece zero".

Zero offsets:

Every zero offset (G54 to G57, G505 to G599) consists of a coarse offset and a fine offset. You can call the work offsets from any sequence program (coarse and fine offsets are added together).

You can save the workpiece zero, for example, in the coarse offset, and then store the offset that occurs when a new workpiece is clamped between the old and the new workpiece zero in the fine offset.

Coordinate transformations:

You always program coordinate transformations for a specific sequence program.

They are defined by:

- Offset
- Rotation
- Scaling
- Mirroring

These transformations can work as “new” or they can work “additive” to the active zero point offset.

Total offset:

The total offset is calculated from the sum of all offsets and coordinate transformations.

Mirror, offset, rotate and scale

The commands in this module are known as frames (calculation instructions). They can influence, cancel or condition each other.

In the following description a known contour, which is described in a subprogram, is to be milled in various positions and sizes. This can thus be used for programming with driven tools on turning machines and as well in milling.

The commands from the range of themes “high-level languages” are used to show how with the aid of programmable frames the current zero point can be shifted to a suitable position in the work piece coordinate system. By shifting, rotating, scaling and mirroring the machining operations can be simplified.

If repetitive steps are laid down in subprograms, the programming extent can be reduced and the program can be produced with a greater flexibility.

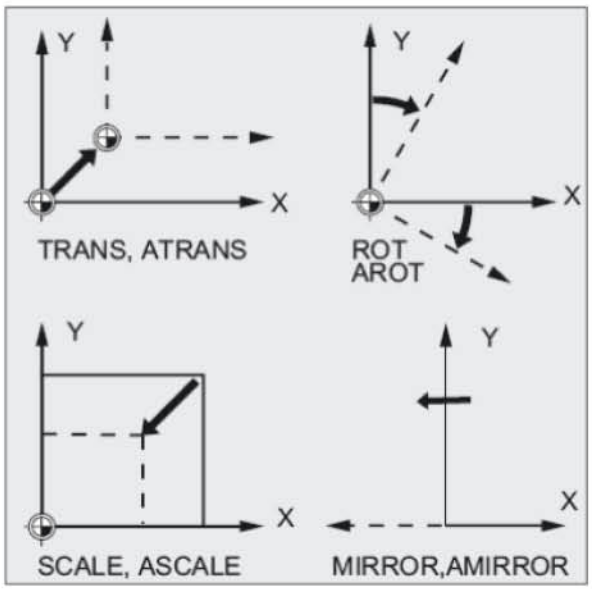


Figure 2.11: Mirror, offset, rotate and scale

The following guidelines apply to programming frames:

- Programmable frames always refer to the last active part zero point in the program.
- They are carried out in the order in which they appear in the program.
- Instructions without a preceding “A” (e.g. TRANS Z-10) are replacing frames, i.e. they delete all frame instructions previously in the program.
- This means that each of these instructions cancels all other previously programmed frame instructions and all frame instructions have a modal effect.
- The last called settable zero offset G54 to G599 is used as the reference.
- The preceding “A” identifies the additive instructions. They are based on all the frame instructions that were already in the program and therefore do not delete them (e.g. ATRANS Z-15).
- Active frames are reset (i.e. return to the set workpiece zero point) in the program with a replacement frame (e.g. TRANS), with "M30" or manually with "Reset".

Shifting with “TRANS” and “ATRANS”

The zero point can be shifted in the programmed axes by means of these commands. “ATRANS” (additive shift) works incrementally from the active frames.

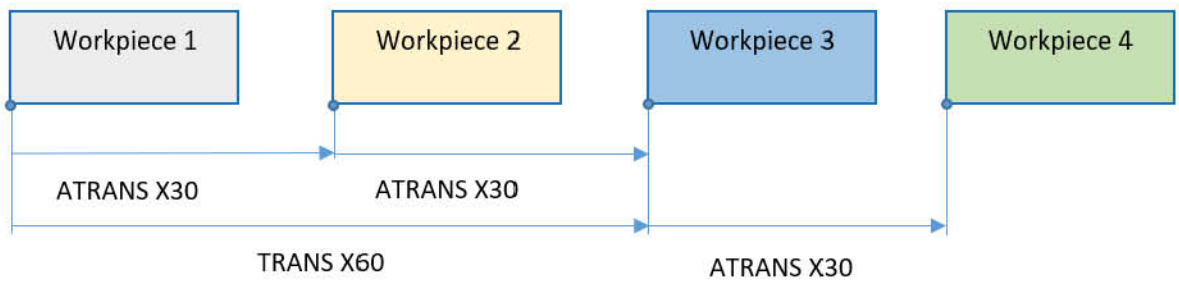
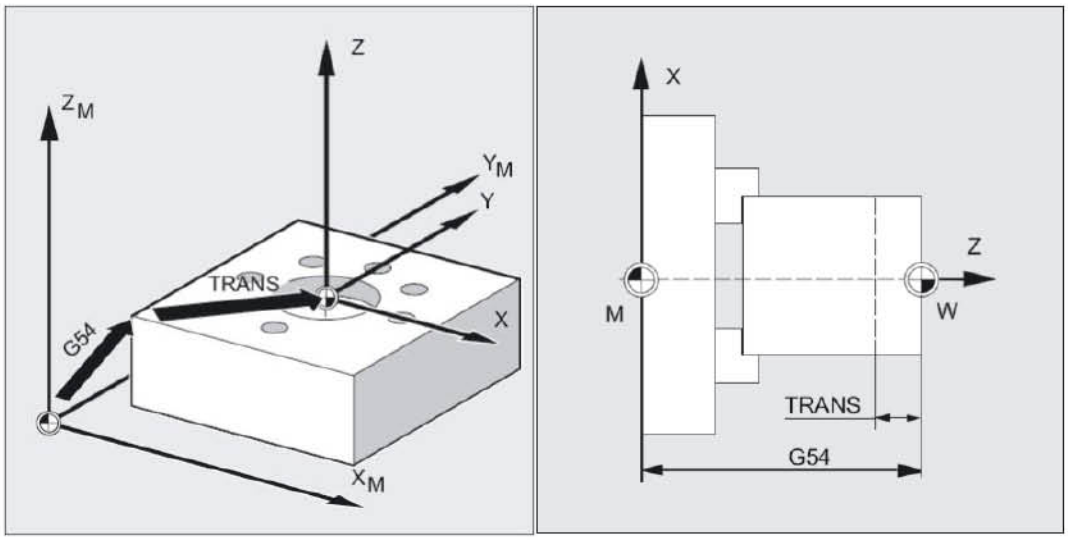


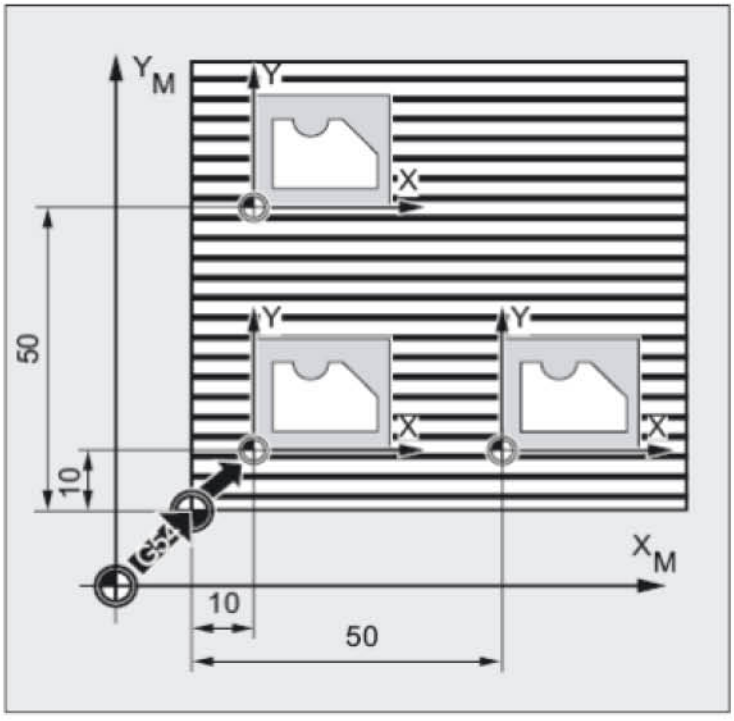
Figure 2.12: Shifting with “TRANS” and “ATRANS”

For instance, in case of multiple settings of work pieces the zero point can be shifted and the main program repeatedly executed. Instead, the “TRANS” and “ATRANS” can simplify individual machining operations by shifting the work piece zero point in the program.



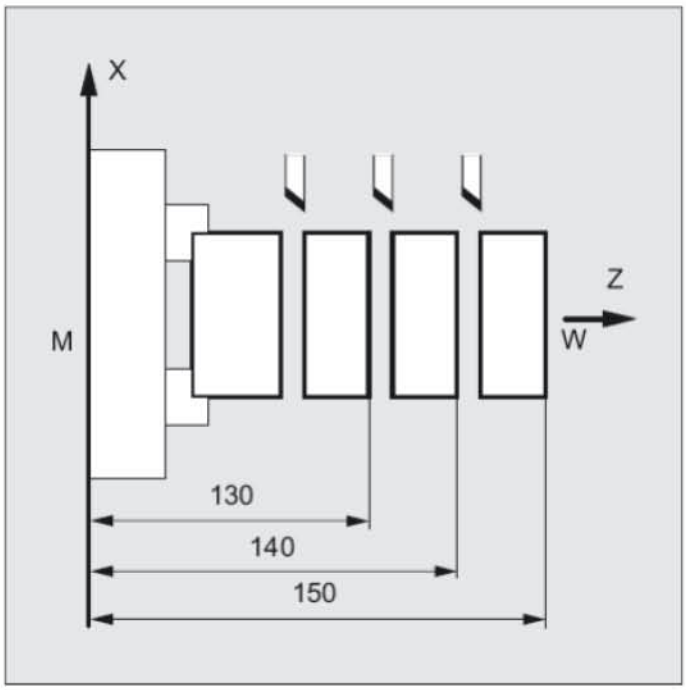
Example for shifting with milling and turning:

- The shapes shown occur several times on this work piece.
- The machining for these is programmed in a subprogram.
- All that is now necessary is to specify the respective work piece zero points by means of zero point offset and call up the subprogram.



Program code	**Comment**
N1 G17 G54	; Working plane X/Y, work piece zero point
N2 G0 X0 Y0 Z2	; Approach starting point

```
N3 TRANS X10 Y10          ; Absolute offset
N4 L10                    ; Subprogram call up
N5 TRANS X50 Y10          ; Absolute offset
N6 L10                    ; Subprogram call up
N7 TRANS X100 Y50         ; Absolute offset
N8 L10                    ; Subprogram call up
N9 M30                    ; End-of-program
```



Program code **Comment**

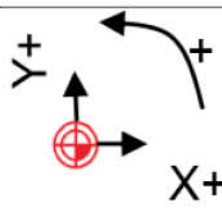
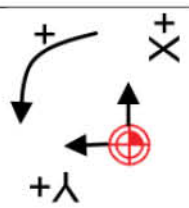
```
N1 G18
                ; Working plane X/Z
N2 G0 X… Z…                    ; Approach starting point
N3 TRANS X0 Z150               ; Absolute offset
N4 L20                         ; Subprogram call up
N5 TRANS X0 Z140 (or ATRANS Z-10)  ; Absolute offset
N6 L20                         ; Subprogram call up
N7 TRANS X0 Z130 (or ATRANS Z-10)  ; Subprogram call up
N8 L20                         ; End of program
```

Summary:

TRANS X…Y…Z…	Absolute programmable zero point offset as referred to the presently valid, with G54 to G57 and G505 to G599 selected work piece zero point
ATRANS X…Y…Z…	Additive programmable zero point offset as referred to the presently valid, selected or programmed zero point
X…Y…Z…	Shift value in direction of the stated axis
TRANS	Deactivation of programmable zero point offsets, previously programmed frames are cleared

Rotating with "ROT" and "AROT"

The coordinate system can be rotated by means of the command "ROT" or "AROT". The counter-clockwise rotation is taken to be positive. The positive X-axis represents the zero-degree position.

Variation 1		
	AROT Z=90	
Rotation about an axis AROT X… Y... Z...		
The zero point has been rotated by 90° about the Z-axis.		

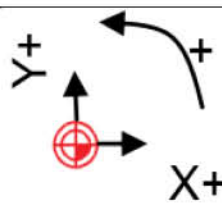
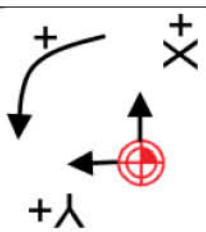
Variation 2		
	AROT RPL=90	
Rotation of the active plane AROT RPL...		
The active plane has been rotated by 90° about the Z-axis.		

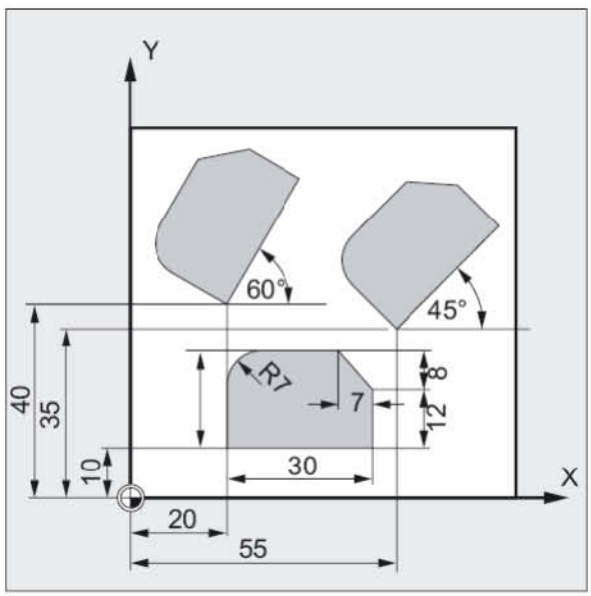
Figure 2.13: Rotating with "ROT" and "AROT"

The command "ROT" resets the coordinate system and all other active frames back to the original status. By means of ROT/AROT the work piece coordinate system can by choice be rotated about any of the three geometrical axes X, Y, Z or by an angle RPL in the selected working plane G17 to G19 (respectively about the vertical infeed axis).

This permits the machining of oblique faces or several work piece faces in one and the same setting.

Example for plane rotation:

- The shown elements occur several times on this work piece in one program.
- In addition to the zero point offset rotations must be carried out since the elements are not arranged parallel to the axes.



P

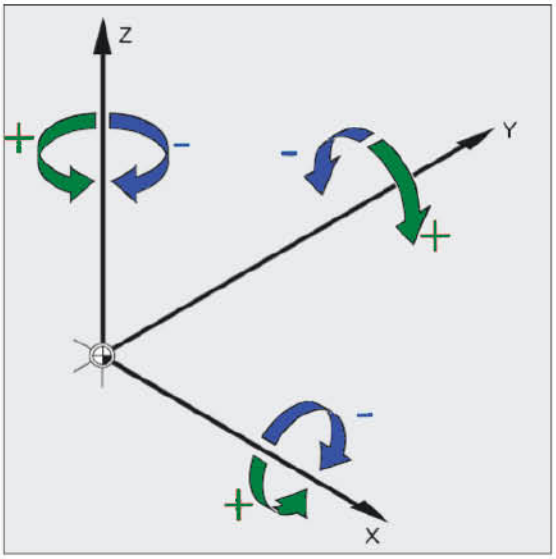
N

N

```
N3 L10                  ; Subprogram call up
N4 TRANS X55 Y35        ; Absolute offset
N50 AROT RPL=45         ; Rotation of the co-ordinate systems by 45°
N6 L10                  ; Subprogram call up
N7 TRANS X20 Y40        ; Absolute offset (resets all offsets used so far)
N8 AROT RPL=60          ; Additive rotation by 60°
N9 L10                  ; Sub program call up
N10 G0 X100 Y100        ; Retraction
N11 M30                 ; End of program
```

Direction of rotation:

By programming a rotation about an axis wit ROT/AROT X... Y... Z...the positive direction of rotation is clockwise when looking in the positive direction of the coordinate axis.



If you grip the axis with your fingers, the fingers point in the positive direction of rotation.

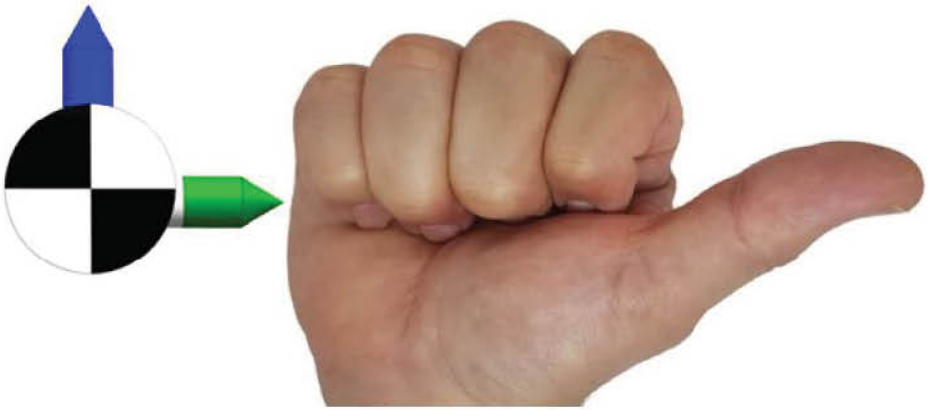


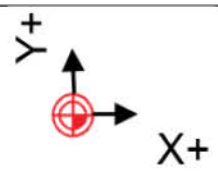
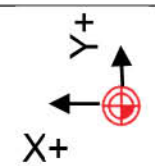
Figure 2.14: Direction of rotation

Summary:

ROT X…Y…Z… ROT RPL=…	Absolute programmable rotation as referred to the presently valid, with G54 to G57 and G505 to G599 selected work piece zero point
AROT X…Y…Z… AROT RPL=…	Additive programmable rotation as referred to the presently valid, selected or programmed zero point
X…Y…Z…	Rotation angle in space: Geometry axis that is being rotated
RPL=	Rotation angle in a plane: Angle by which the coordinate system Is rotated - (plane previously selected with G17 to G19)
ROT	Deactivation of programmable rotation, previously programmed frames are cleared

Mirroring with "MIRROR" and "AMIRROR"

With the command "MIRROR" or "AMIRROR" the coordinate system can be mirrored about the programmed axis or axes. In such a case the programmed coordinates are mirrored about the axis or axes by their sign.

Variation 1		
	AMIRROR X0	
Mirroring an axis		

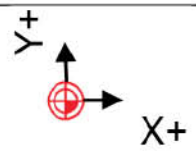
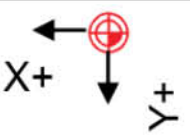
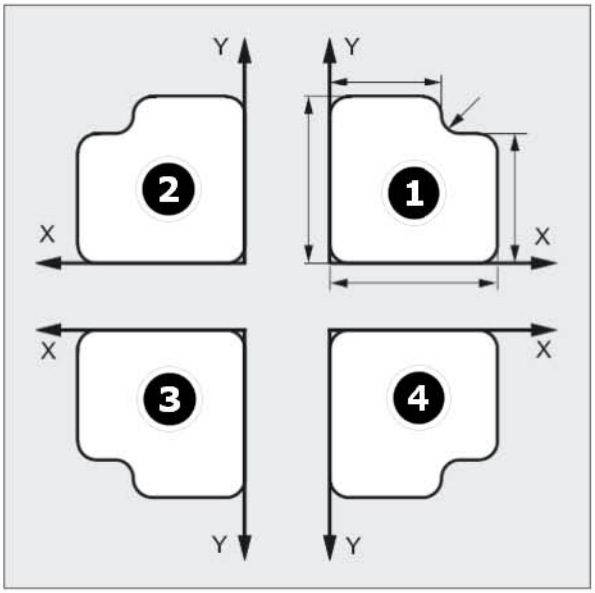
Variation 2		
	MIRROR X0 Y0	
Mirroring several axes		

Figure 2.15: Mirroring with "MIRROR" and "AMIRROR"

The command "MIRROR" resets the coordinate system and all other active frames back to the original status.With MIRROR/AMIRROR it is possible to mirror work piece shapes about coordinate axes. All feed motions that occur after the mirroring call up, for instance in a subprogram, are carried out in the mirrored fashion.

Example for mirroring:

The contour shown here is programmed just once in a sub-program. The three others are produced by mirroring. The work piece zero point is located centrally between the contours.



Program code	**Comment**
N1 G17 G54	; Working plane X/Y, work piece zero
N2 L10	; Machine first contour at top right
N3 MIRROR X0	; Mirroring X-axis (the direction is changed in X)
N4 L10	; Machine second contour at top right
N5 AMIRROR Y0	; Mirroring Y-axis (in Y the direction is; changed)
N6 L10	; Machine third contour at bottom left
N7 MIRROR Y0	; MIRROR resets previous frames
	; Mirror Y-axis (in Y the direction is changed)
N8 L10	; Machine fourth contour at bottom right
N9 MIRROR	; Deactivate mirroring
N10 G0 X300 Y100	; Retraction
N11 M30	; End of program

Summary:

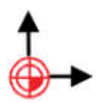
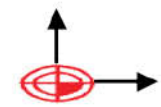
MIRROR X...Y...Z...	Absolute programmable mirroring as referred to the presently valid, with G54 to G57 and G505 to G599 selected work piece zero point
AMIRROR X..Y..Z..	Additive programmable mirroring as referred to the presently valid, selected or programmed zero point
X…Y…Z…	Coordinate axis, in which the signs are changed The value for X/Y or Z can be freely selected - e.g. X0/Y0/Z0
MIRROR	Deactivation of programmable mirroring, previously programmed frames are cleared

Scaling with "SCALE" and "ASCALE"

In some cases, the scaling of contour elements is quite sensible. The existing contours can be scaled up or down by a given factor.

The factor is defined following the command Scale for each individual axis SCALE/ASCALE can be used to program scale factors for all axes in the direction of the stated axis.

In this manner the size of an element can be changed arbitrary.

Variation 1		
	ASCALE X1.5	
Scaling an axis		

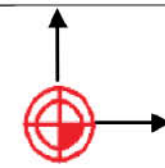
Variation 2		
	ASCALE X1.5 Y1.5	
Scaling several axes		

Figure 2.16: Scaling with "SCALE" and "ASCALE"

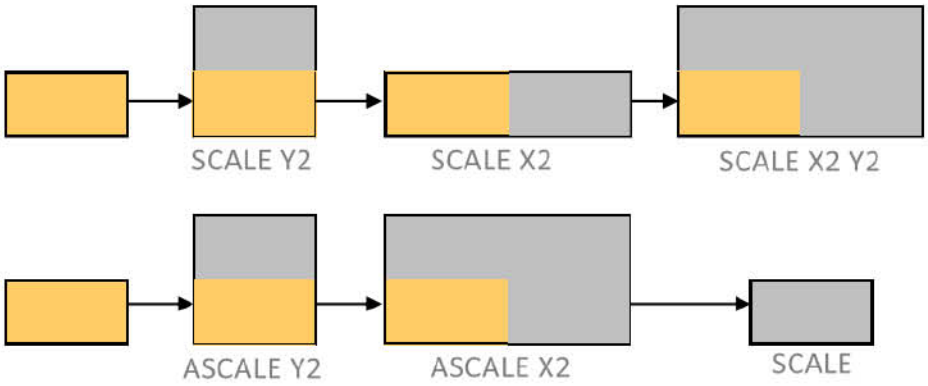


Figure 2.17: Diagram command ASCALE

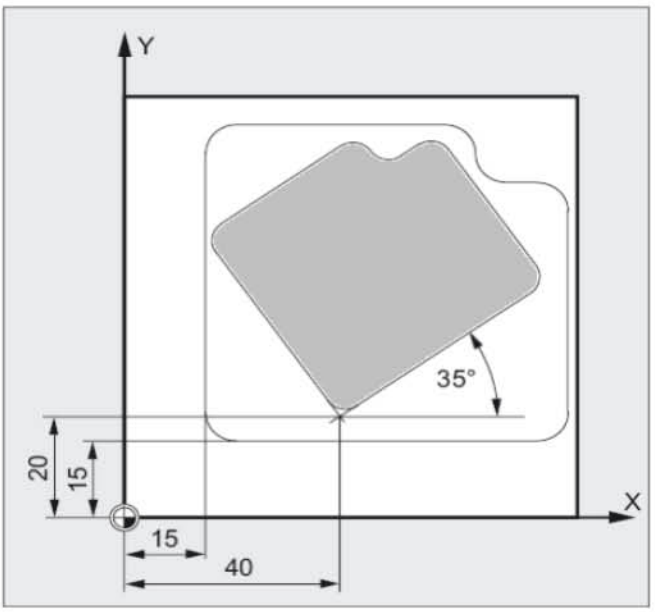
When programming with "ASCALE" the calculations are always referred to the presently valid coordinate system. With the command "SCALE" the values of the active zero point offset are taken as a basis. All presently active transformations (frames) like offset, mirror, scale are cancelled.

If no value follows the command "SCALE" all frames are cancelled. The same applies to the command "M30".

Example for scaling:

On this work piece there are two pockets, however, they are of different size and rotated relative to one another. The machining operation is programmed in a subprogram.

By zero point offset and rotation the respectively required work piece zero points are selected. By scaling the contour size is reduced before the subprogram is recalled.



Program code	Comment
N1 G17 G54	; Working plane X/Y, work piece zero point
N2 TRANS X15 Y15	; Absolute offset
N3 L10	; Mach. large pocket with L10 sub-program
N4 TRANS X40 Y20	; Absolute offset
N5 AROT RPL=35	; Rotation in the plane by 35°
N6 ASCALE X0.7 Y0.7	; Scale factor for the small pocket
N7 L10	; Machining small pocket
N8 G0 X300 Y100	; Retraction
N9 M30	, End-of-program

Summary:

SCALE X...Y...Z...	Absolute programmable enlarging or diminishing (scaling), as referred to the presently valid, with G54 to G57 and G505 to G599 selected work piece zero point
ASCALE X...Y...Z...	Additive programmable enlarging or diminishing (scaling) as referred to the presently valid, selected or programmed zero point
X…Y…Z…	Scaling factor (smaller / greater 1) in direction of stated axis
SCALE	Deactivation of programmable enlarging or diminishing, previously programmed frames are cleared

3.3. Programming of driven tools (DT) on the front surface

The C axis is an additional, programmable rotary axis around the Z axis. It enables the precise and correct positioning and interpolation of the respective spindle. It receives the positions via the address "C" or “X/Y” in the processing program. If a CNC turning machine has a C axis spindle, workpieces can be machined in the optimum time.

A further advantage is the quality: With regard to form and position tolerances, workpieces can be machined more precisely with drilling and milling tools in one clamping. The conventional programming of a C axis in G code requires more extensive knowledge of programming.

The following commands are required for the programming of the C axis spindle:

Command	Meaning
SETMS(1)	Activate the main spindle to master spindle (manufacturer-specific)
SETMS(3)	Activate the driven tool to master spindle (manufacturer-specific)
S1=	Programming the cutting speed / rotational speed MS (manufacturer-specific)
M1=3	Main spindle = clockwise rotation direction (manufacturer-specific)
M3=3	Driven tool spindle = clockwise rotation direction (manufacturer-specific)
M1=4	Main spindle = counter clockwise rotation direction (manufacturer-specific)
M3=4	Driven tool spindle = counter clockwise rotation direction (manufac.-specific)
M1=5	Main spindle stop (manufacturer-specific)
M3=5	Driven tool spindle stop (manufacturer-specific)
TRANSMIT	Front surface transformation (transformation milling into turning)
TRACYL(d)	Cylinder surface transformation with current working diameter
TRAFOOF	Deactivating all active transformations/frames
DIAMON	Dimensions in the diameter (G90 and G91)
DIAMOF	Dimensions in the radius (G90 and G91)
G17	Working plane X/Y, infeed direction Z - front surface
G18	Working plane Z/X, infeed direction Y - turning
G19	Working plane Y/Z, infeed direction X - cylinder surface

TRANSMIT

The TRANSMIT function enables the following services:

- Front surface drilling and milling of workpiece in one clamping situation.
- The Cartesian coordinate system can be used as a basis for programming these machining operations.
- The control transforms the programmed traversing movements of the Cartesian coordinate system to the traversing movements of the real machine axes X and C.

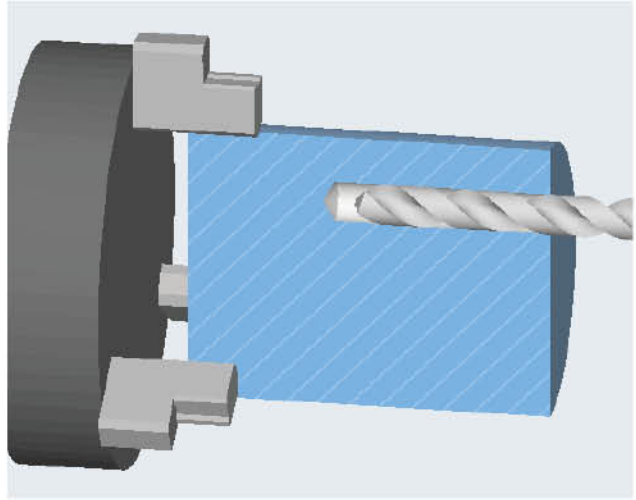


Figure 2.18: Drilling on front surface - G17

Description example for activating and deactivating the front surface machining:

Command	**Meaning**
SETMS(1)	Activate the main spindle to master spindle
G18	Working plane Z/X, infeed direction Y – turning
…..	Machining the turning operations
M1=5	Main spindle stop (here also “M5” possible)
SPOS[1]=0	Set master spindle to specified angle
SETMS(3)	Activate the driven tool spindle to master spindle
G17	Working plane X/Y
DIAMOF	Dimensions in the radius (G90 and G91)
TRANSMIT	Front face transformation (transformation milling into turning)
T=”…”	Tool call up
G0 X… Y… Z…	Start position machining
…	Machining the drilling operations
TRAFOOF	Deactivating all active transformations/frames
DIAMON	Dimensions in the diameter (G90 and G91)
M3=5	Driven tool spindle stop (here also “M5” possible)
SETMS(1)	Activate the main spindle to master spindle
G18	Working plane Z/X, infeed direction Y – turning
G0 SUPA X500	Tool changing point

Z500 D0 M9	

Front surface drilling example with program Guide

Description:

A first programGuide exercise is to be programmed with the help of centering, drilling and tapping modal cycle calls.

Objective:

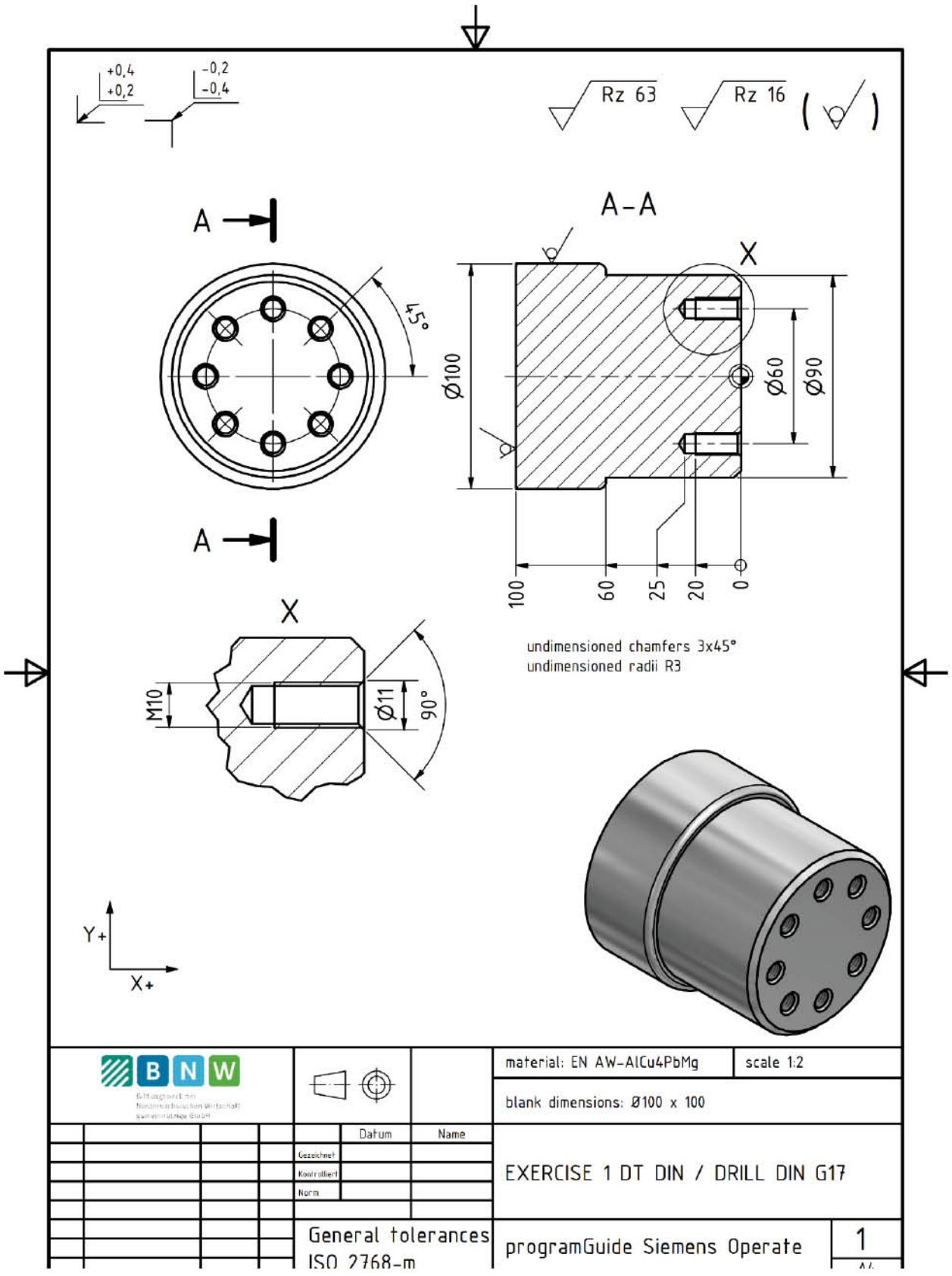
The program uses the above commands to drill into the face. Then switch to the turning machining level in order to manufacture the outer diameter Ø90.

The workflow sequences are to programmed:

- Program header
- Programming the commands for the face machining in G17
- Centering Ø11 for the eight drill holes
- Drilling the core holesØ8,5 mm for the eight drill holes
- Tapping M10 for the eight drill holes
- Switch to the turning machining plane in G18
- Longitudinal turning the Ø90
- Program end

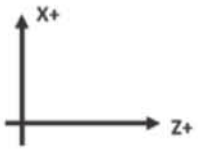
A new workpiece directory “EXERCISE_1_DT_DIN” should be created and opened in the editor. Create a new G code program in program Guide, like described in the previous modules.

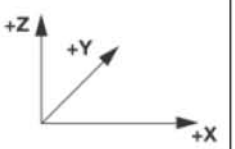
Give the program a name, for example “DRILL_DIN_G17”.


+0,4
+0,2
-0,2
-0,4
Rz 63
Rz 16
A
A
A–A
X
45°
Ø100
Ø60
Ø90
100
60
25
20
0
undimensioned chamfers 3x45°
undimensioned radii R3
X
M10
Ø11
90°
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:2
blank dimensions: Ø100 x 100
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE 1 DT DIN / DRILL DIN G17
General tolerances
ISO 2768-m
programGuide Siemens Operate
1

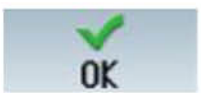
The following tools and technology data are required for programming:

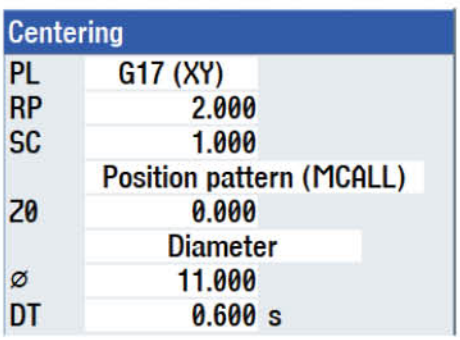
programGuide Siemens Operate VIETNAM 2021/2022		
Tool shape file	CNC Turning	
work piece: EXERCISE 1 DT DIN DIN_DRILLING_G17	blank material: EN-AW AlCu4PbMg	machine configuration:
drawing: 1(1)	blank dimensions: Ø100 x 100	X Z C

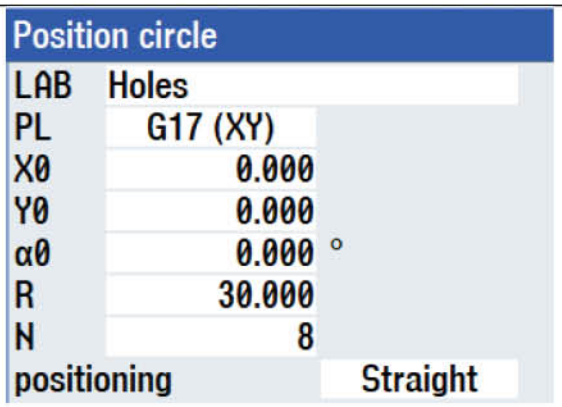
Tools for outer machining		
	Technological data	
Tool name in OFFSET	T="ROUGH T80 A"	
Cutting radius (mm)	0,8	
Cutting speed (m/min)	250	
Cutting depth ap = max.	2,5 / 1	
Cutting material	SC	
Feed rate per rev./pitch	0,3 / 0,12	
Angle/pitch/diameter Ds	80°	
		

Tools for drilling and milling machining			
	Technological data		
Tool name in OFFSET	T="CENTER DRILL 12"	T="DRILL 8,5"	T="TAP M10"
Tool Ø (mm)	12 x 90°	8,5	M10
Cutting speed (m/min)	80	60	10
Cutting depth ap = max.	5,5		
Cutting material	HSS	HSS	HSS
Feed tooth fz / rev. (mm)	0,12	0,1	F = P
Number of teeth N			
			
	G17	G17	G17

Open the new G code program Guide “DRILL_DIN_G17”:

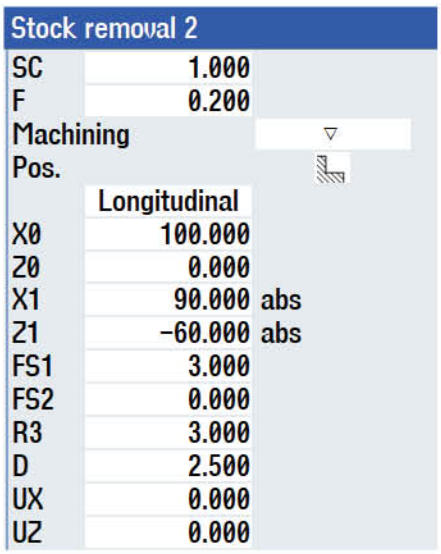
1.	Program the first lines of the program by inserting the following G code commands: **N1 G71 G90**
2. 	Insert now a blank for the simulation into the program. Press the HSK 1.6 “Various” to open the work area “Various”. The following input mask for the blank input opens.
3.	Enter following values/parameters for the blank: **Blank input** Blank: Cylinder XA: 100.000 ZA: 0.000 ZI: -100.000 abs ZB: -80.000 abs
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N2 WORKPIECE(,,,"CYLINDER",192,0,-100,-80,100)**
4.	Program the next G code and front surface machining lines into the program: **N3 G0 SUPA X500 Z500 D0 M9** **N4 G54** **N5 SPOS[1]=0** **N6 SETMS(3)** **N7 G17** **N8 DIAMOF** **N9 TRANSMIT**
5. 	Press the VSK 1.1 “Select tool”. The tool list window opens. Use the blue cursor keys to select the tool “CENTERDRILL 12”. Press the VSK 1.8 “OK”. The program line **N10 T="CENTERDRILL 12"**is inserted into the program.

6.	Program the following G code commands: **N11 G97 S2122 M3** **N12 G95 F0.12** **N13 G0 X30 Z5** **N14 M8**
7. 	Program the centre drill cycle “CYCLE81”. Press the HSK 1.2 “Drill.”, to open the technology “Drilling”. Press the VSK 1 “Centering”. The input mask for the centering cycle “CYCLE81” opens.
8.	Enter the following values into the input mask: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N15 MCALL CYCLE81(2,0,1,11,,0.6,10,1,11)**
9. 	Program now a position pattern (Position circle) for the following drillings. Press the VSK 7 “Positions” for selecting a position pattern. Press the VSK 5 “Position circle”. The parameter input mask for the position data of the bolt hole pattern opens.
10.	Insert “HOLES” into the “LAB” field to set a name for the jump mark for the repeat positions. Enter the following values into the input mask:

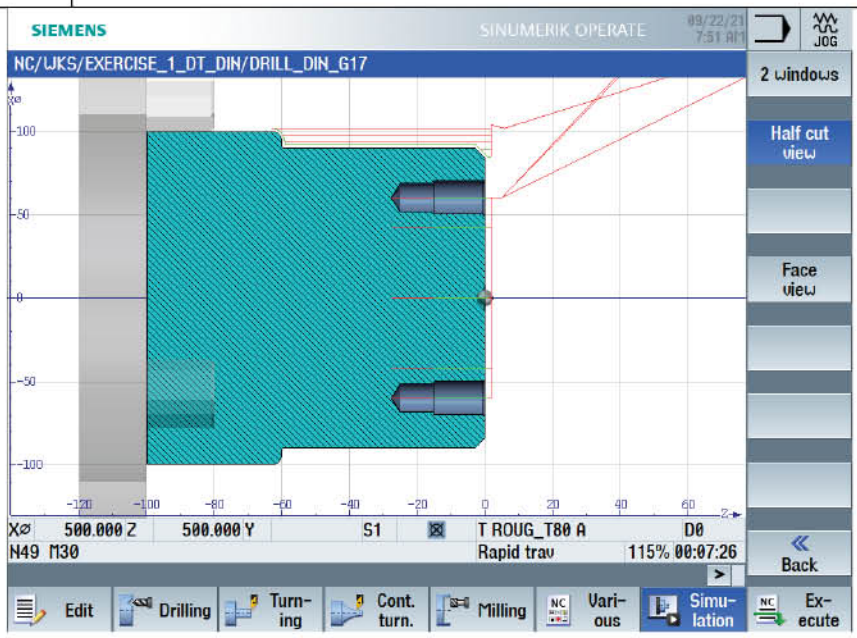
	
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N16 Holes: HOLES2(0,0,30,0,30,8,1000,0,,,1)**
11.	Program the following G code commands: **N17 MCALL** **N18 G0 SUPA X500 Z500 D0 M9**
12.	Insert the following tool “DRILL 8.5” into the program (see step 5) or program the following line by hand: **N19 T="DRILL 8.5"**
13.	Program the following G code commands: **N20 G97 S2247 M3** **N21 G95 F0.1** **N22 G0 X30 Z5** **N23 M8**
14. 	Press the VSK 2 “Drilling Reaming”. Press the VSK 3 “Drilling”. The input mask for the drilling cycle “CYCLE82” opens.
15.	Enter the following values into the input mask:

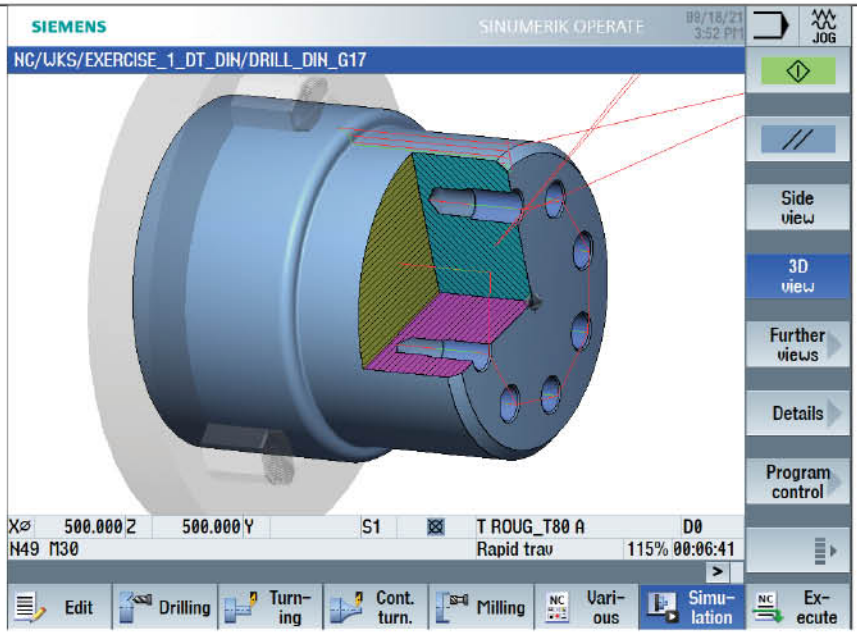
	Drilling Input Complete PL G17 (XY) RP 2.000 SC 1.000 Position pattern (MCALL) Z0 0.000 Shank Z1 -25.000 abs Spot drill No Thrd. drill No DT 0.600 s
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N24 MCALL CYCLE82(2,0,1,-25,,0.6,10,1,12)**
16. Position repetit.	Press the VSK 8 “Positions repeatit.” for call up a position pattern once again.
17.	Enter the following values into the input mask: **Repeat position** LAB Holes
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N25** **N26 REPEATB Holes ;#SM**
18.	Program the following G code commands: **N27 MCALL** **N28 G0 SUPA X500 Z500 D0 M9**
19.	Insert the tool “TAP M10” into the program (see step 5) or program the following line by hand: **N29 T="TAP M10"**
20.	Program the following G code commands: **N30 G97 S318 M3** **N31 G0 X30 Z5** **N32 M8**

21. Thread Tapping	Finally program the last cycle for the thread drilling (CYCLE84). Press the VSK 5 “Thread”. If not active by default, press the VSK 3 “Tapping” to open the input mask for the CYCLE84 “Tapping”.
22.	Enter the following values into the input mask: **Tapping** Input Complete PL G17 (XY) RP 2.000 SC 1.000 W/o compensat. chuck Position pattern (MCALL) Z0 0.000 Z1 -15.000 abs RH thread Table ISO metric Select M 10 P 1.500 mm/rev αS 0.000 ° S 316 rpm 1 cut DT 0.600 s SR 600 rpm SDE ☒
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N33 MCALL CYCLE84(2,0,1,-15,,0.6,5,,1.5,0,316,600,0,1,0,0,5,1.4,,"ISO_METRIC","M10",,1001,1001002)**
23. Position repetit.	Press the VSK 8 “Positions repeatit.” for call up a position pattern once again.
24.	Enter the following values into the input mask: **Repeat position** LAB Holes

	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N34 REPEATB Holes ;#SM**
25.	Program the following G code commands: **N35 MCALL** **N36 TRAFOOF** **N37 DIAMON** **N38 M3=5** **N39 SETMS(1)** **N40 G18** **N41 G0 SUPA X500 Z500 D0 M9**
26.	Insert the tool “ROUG_T80 A” into the program (see step 5) or program the following line by hand: **N42 T="ROUG_T80 A"**
27.	Program the following G code commands: **N43 G96 S250 M4** **N44 G95 F0.25** **N45 G0 X100 Z5 M8**
28.	Enter the following values into the input mask: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N46** **N47CYCLE951(100,0,90,-60,90,-60,1,2.5,0,0,11,3,0,3,1,0.2,1,2,110000)**

29.	Finish the program the following G code commands: **N48 G0 SUPA X500 Z500 D0 M9** **N49 M30**
30.	Finally check the machining with the simulation by using of different views.





Task:

Create blocks in the program in accordance with the work sequences and save the setup data for machining. Save the program on an external disc and transfer it to the machine control.

3.4. Programming of driven tools on the circumference surface

TRACYL

The TRACYL(d) function enables the following processing:

- Cylinder surface drilling and milling of workpiece in one clamping situation.
- The Cartesian coordinate system can be used as a basis for programming these machining operations.
- The control transforms the programmed traversing movements of the Cartesian coordinate system to the traversing movements of the real machine axes X and C.

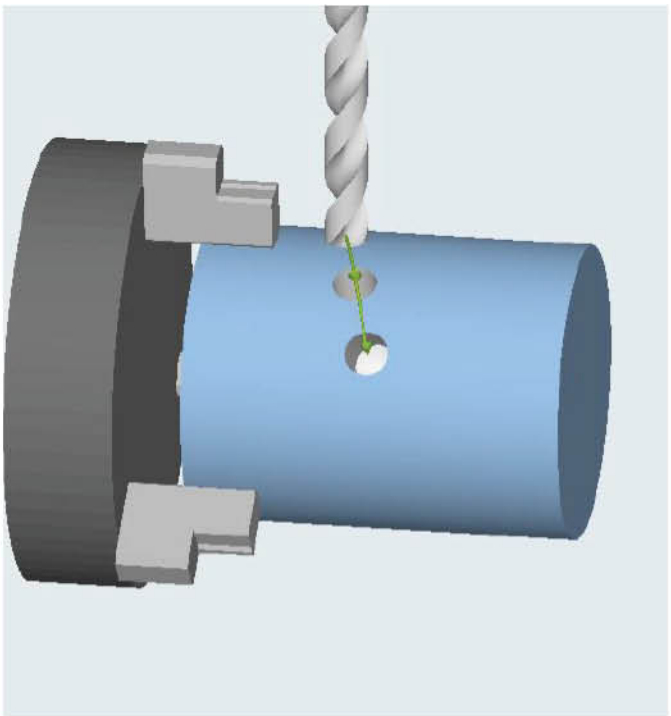


Figure 2.19: Drilling on cylindrical surface - G19

Description example for activating and deactivating the cylinder surface machining:

Command	**Meaning**
SETMS(1)	Activate the main spindle to master spindle
G18	Working plane Z/X, infeed direction Y – turning
…..	Machining the turning operations
M1=5	Main spindle stop (here also “M5” possible)
SPOS[1]=0	Set master spindle to specified angle
SETMS(3)	Activate the driven tool spindle to master spindle
G19	Working plane X/Y
DIAMOF DIAMON	Both programmable with “TRACYL(d)”
TRACYL(d)	Cylinder surface transformation with current working diameter
T=”…”	Tool call up
G0 X… Y… Z…	Start position machining
…	Machining the drilling operations
TRAFOOF	Deactivating all active transformations/frames
DIAMON	Dimensions in the diameter (G90 and G91)
M3=5	Driven tool spindle stop (here also “M5” possible)
SETMS(1)	Activate the main spindle to master spindle
G18	Working plane Z/X, infeed direction Y – turning
G0 SUPA X500 Z500 D0 M9	Tool changing point

Cylinder surface drilling example with program Guide

Description:

A first programGuide exercise is to be programmed with the help of centering, drilling and tapping modal cycle calls.

Objective:

Starting the program to the turning machining level in order to manufacture the outer diameter Ø90. Then uses the above commands to drill into the cylinder surface.

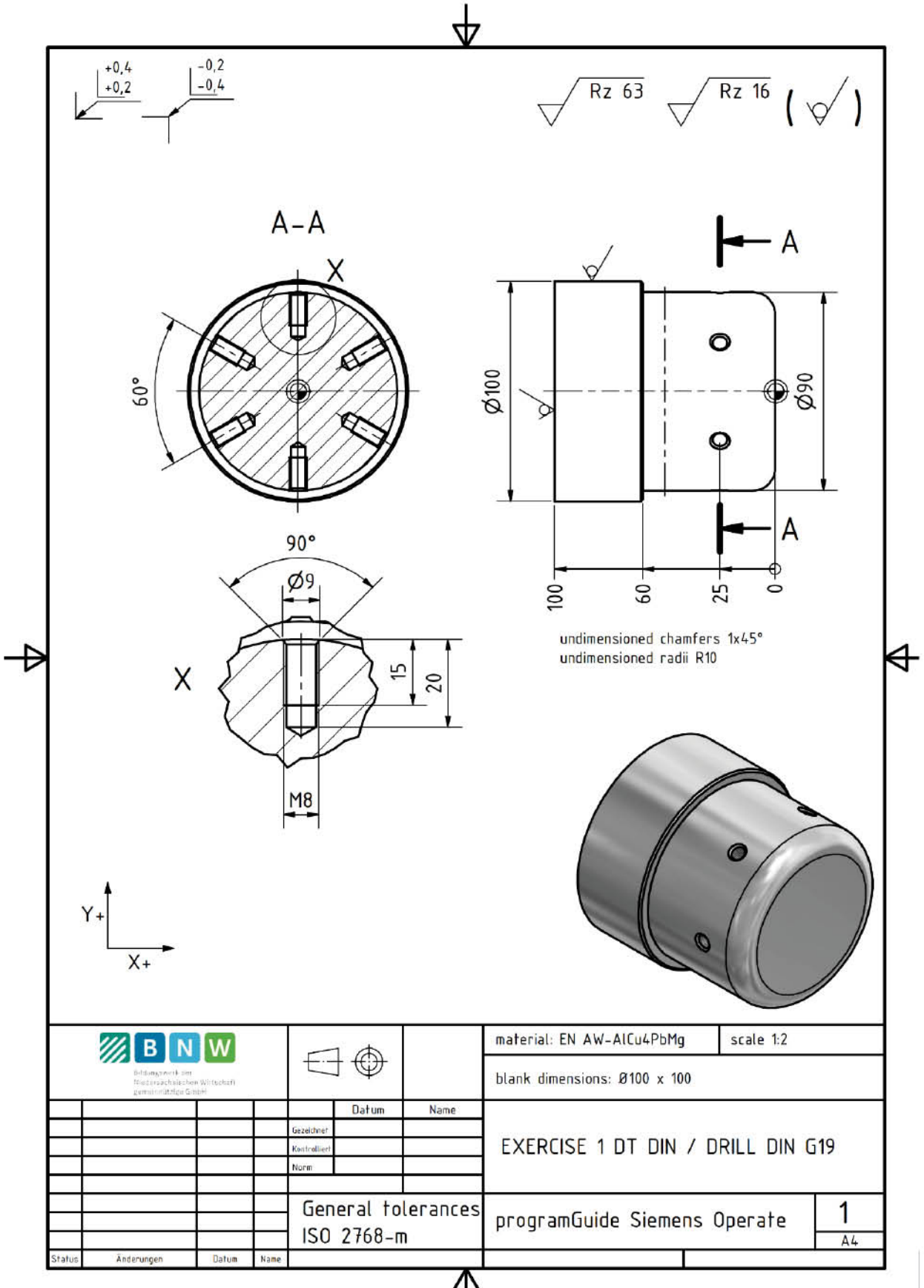
The workflow sequences are to programmed:

- Program header
- Longitudinal turning the Ø90
- Programming the commands for the cylinder surface machining in G19
- Centering Ø9 for the eight drill holes
- Drilling the core holesØ6,8 mm for the six drill holes
- Tapping M8 for the six drill holes
- Switch to the turning machining plane in G18
- Program end

We using the workpiece directory “EXERCISE_1_DT_DIN” once again.

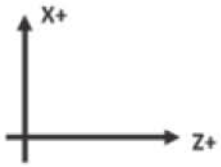
Create a new G code program in programGuide, like described in the previous modules.

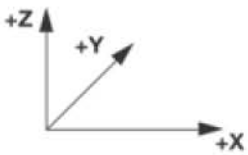
Give the program a name, for example “DRILL_DIN_G19”.


+0,4
+0,2
-0,2
-0,4
Rz 63
Rz 16
A-A
X
60°
A
Ø100
Ø90
A
100
60
25
0
90°
Ø9
X
15
20
M8
undimensioned chamfers 1x45°
undimensioned radii R10
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:2
blank dimensions: Ø100 x 100
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE 1 DT DIN / DRILL DIN G19
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name

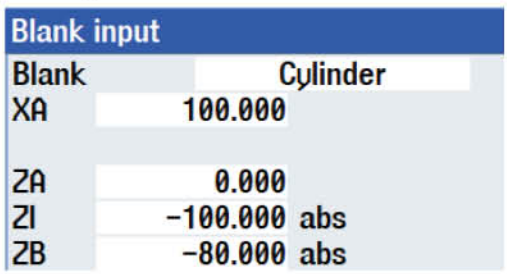
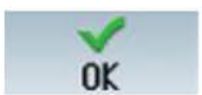
The following tools and technology data are required for programming:

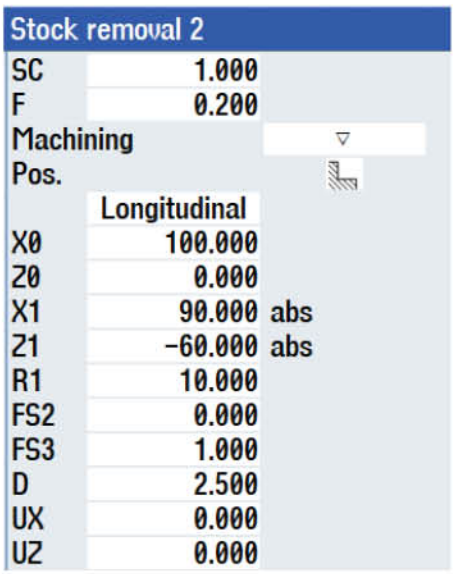
programGuide Siemens Operate VIETNAM 2021/2022		
Tool shape file	CNC Turning	
work piece: EXERCISE 1 DT DIN DIN_DRILLING_G19	blank material: EN-AW AlCu4PbMg	machine configuration:
drawing: 1(1)	blank dimensions: Ø100 x 100	X Z C

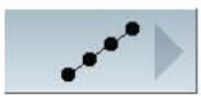
Tools for outer machining		
	Technological data	
Tool name in OFFSET	T="ROUGH T80 A"	
Cutting radius (mm)	0,8	
Cutting speed (m/min)	250	
Cutting depth ap = max.	2,5 / 1	
Cutting material	SC	
Feed rate per rev./pitch	0,3 / 0,12	
Angle/pitch/diameter Ds	80°	
		

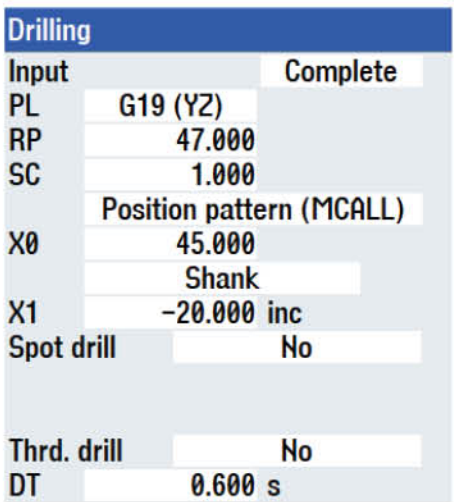
Tools for drilling and milling machining			
	Technological data		
Tool name in OFFSET	T="CENTER DRILL 12"	T="DRILL 6,8"	T="TAP M8"
Tool Ø (mm)	12 x 90°	6,8	M8
Cutting speed (m/min)	80	60	10
Cutting depth ap = max.	5,5		
Cutting material	HSS	HSS	HSS
Feed tooth fz / rev. (mm)	0,12	0,1	F = P
Number of teeth N			
			
	G19	G19	G19

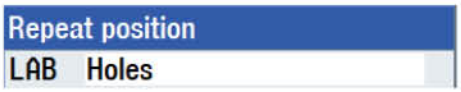
Open the new G code programGuide “DRILL_DIN_G17”:

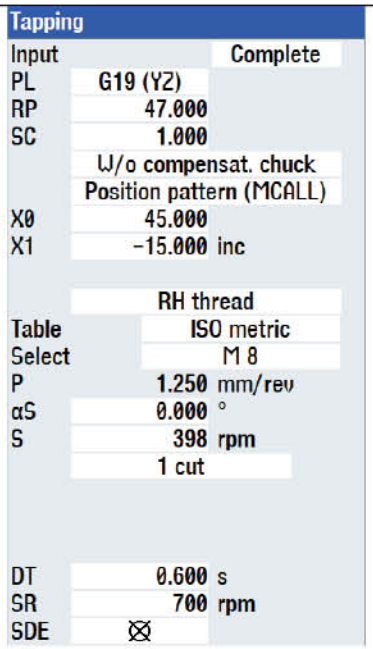
1.	Program the first lines of the program by inserting the following G code commands: **N1 G71 G90**
2. 	Insert now a blank for the simulation into the program. Press the HSK 1.6 “Various” to open the work area “Various”. The following input mask for the blank input opens.
3.	Enter following values/parameters for the blank: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N2 WORKPIECE(,,,"CYLINDER",192,0,-100,-80,100))**
4.	Program the next G code into the program: **N3 G0 SUPA X500 Z500 D0 M9** **N4 G54** **N5 G18** **N6 G0 SUPA X500 Z500 D0 M9**
5. 	Press the VSK 1.1 “Select tool”. The tool list window opens. Use the blue cursor keys to select the tool “T="ROUG_T80 A”. Press the VSK 1.8 “OK”. The program line **N7 T="ROUG_T80 A”** is inserted into the program.
6.	Program the following G code commands: **N8 G96 S250 M4** **N9 G95 F0.25** **N10 G0 X100 Z5 M8** **N11 M8**

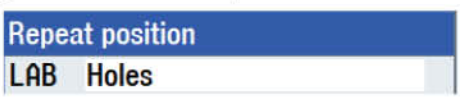
7.	Enter the following values into the input mask: 
 	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N12 CYCLE951(100,0,90,-60,90,-60,1,2.5,0,0,11,10,0,1,1,0.2,1,2,1100000)**
8.	Program the next G code and front surface machining lines into the program: **N13 G0 SUPA X500 Z500 D0 M9** **N14 M5** **N15 SPOS[1]=0** **N16 SETMS(3)** **N17 G19** **N18 DIAMOF** **N19 TRACYL(100)**
9.	Insert the following tool “CENTERDRILL_12” into the program (see step 5) or program the following line by hand: **N20 T="CENTERDRILL_12"**
10.	Program the following G code commands: **N21 G97 S2122 M3** **N22 G95 F0.12** **N23 G0 X50 Y0 Z-25** **N24 M8**

11. 	Program the centre drill cycle “CYCLE81”. Press the HSK 1.2 “Drill.”, to open the technology “Drilling”. Press the VSK 1 “Centering”. The input mask for the centering cycle “CYCLE81” opens.
12.	Enter the following values into the input mask: **Centering** PL G19 (YZ) RP 47.000 SC 1.000 Position pattern (MCALL) X0 45.000 Diameter ⌀ 9.000 DT 0.600 s
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N25 MCALL CYCLE81(47,45,1,9,,0.6,10,3,11)**
13. 	Program now a position pattern (Row of positions) for the following drillings. Press the VSK 7 “Positions” for selecting a position pattern. Press the VSK 2 “Row of positions”. The parameter input mask for the position data of the bolt hole pattern opens.
14.	Insert “HOLES” into the “LAB” field to set a name for the jump mark for the repeat positions. Enter the following values into the input mask: **Row of positions** LAB Holes PL G19 (YZ) Y0 0.000 Z0 -25.000 α0 0.000 ° L0 0.000 L 47.124 N 6
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program:

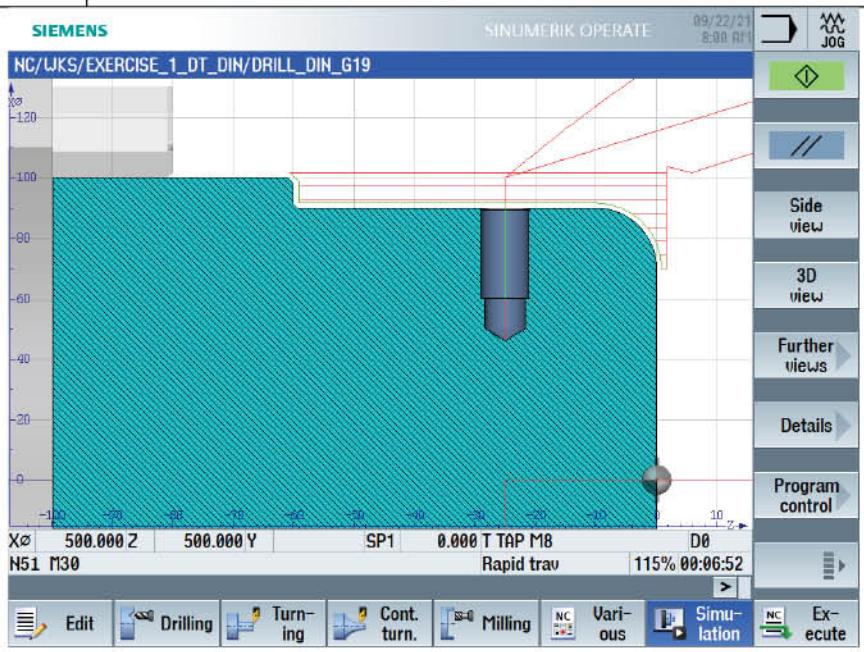
	N26 Holes: HOLES1(0,-25,0,0,47.124,6,,0,,,3)
15.	Program the following G code commands: **N27 MCALL** **N28 G0 SUPA X500 Z500 D0 M9**
16.	Insert the following tool "DRILL 6,8" into the program (see step 5) or program the following line by hand: **N29 T="DRILL 6,8"**
17.	Program the following G code commands: **N30 G97 S2808 M3** **N31 G95 F0.1** **N32 G0 X50 Y0 Z-25** **N33 M8**
18. 	Press the VSK 2 "Drilling Reaming". Press the VSK 3 "Drilling". The input mask for the drilling cycle "CYCLE82" opens.
19.	Enter the following values into the input mask: 
	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N34 MCALL CYCLE82(47,45,1,,-20,0.6,10,3,11)**
20. 	Press the VSK 8 "Positions repeatit." for call up a position pattern once again.

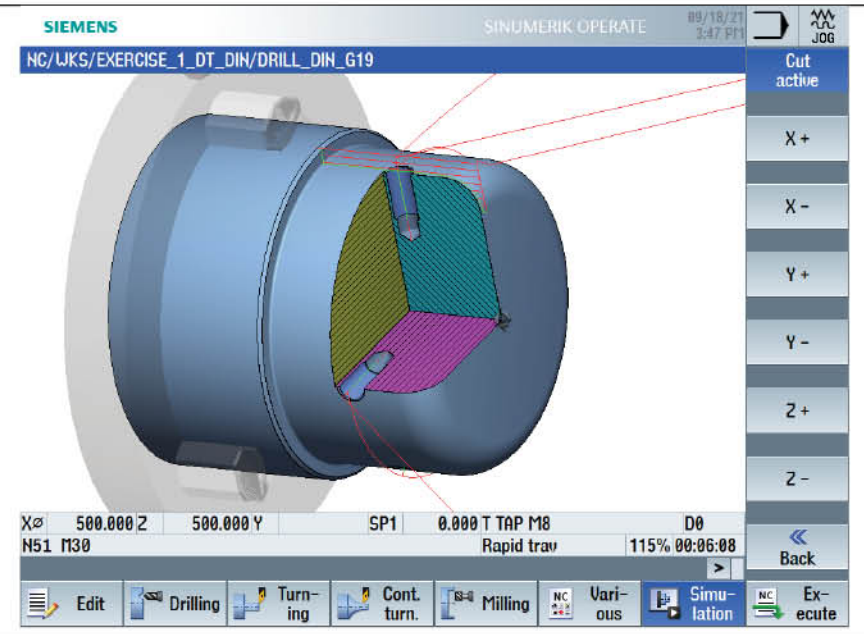
21.	Enter the following values into the input mask: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N35 REPEATB Holes ;#SM**
22.	Program the following G code commands: **N36 MCALL** **N37 G0 SUPA X500 Z500 D0 M9**
23.	Insert the tool “TAP M8” into the program (see step 5) or program the following line by hand: **N38 T="TAP M8"**
24.	Program the following G code commands: **N39 G97 S398 M3** **N40 G0 X50 Z-25** **N41 M8**
25. 	Finally program the last cycle for the thread drilling (CYCLE84). Press the VSK 5 “Thread”. If not active by default, press the VSK 3 “Tapping” to open the input mask for the CYCLE84 “Tapping”.
26.	Enter the following values into the input mask:

<table>
<tr><td></td><td>


</td></tr>
<tr><td>
</td><td>Confirm your inputs by pressing the VSK 8 “Accept”.
The following line is inserted into the program:
N42 MCALL CYCLE84(47,45,1,,-15,0.6,5,,1.25,0,398,700,0,1,0,0,5,1.4,,"ISO_METRIC","M8",,1003,1001001)</td></tr>
<tr><td>27.
</td><td>Press the VSK 8 “Positions repeatit.” for call up a position pattern once again.</td></tr>
<tr><td>28.</td><td>Enter the following values into the input mask:
 
</td></tr>
<tr><td>
</td><td>Confirm your inputs by pressing the VSK 8 “Accept”.
The following line is inserted into the program:
N43 REPEATB Holes ;#SM</td></tr>
<tr><td>29.</td><td>Program the following G code commands:
N44 MCALL
N45 TRAFOOF
N46 DIAMON
N47 M3=5
N48 SETMS(1)
N49 G18</td></tr>
</table>

30.	Finish the program the following G code commands: **N50 G0 SUPA X500 Z500 D0 M9** **N51 M30**
31.	Finally check the machining with the simulation by using of different views.





Task: Create blocks in the program in accordance with the work sequences and save the setup data for machining. Save the program on an external disc and transfer it to the machine

control.

Milling with driven tools – program guide

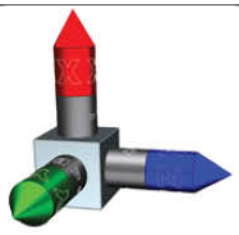
General introduction

The C axis is an additional, programmable rotary axis around the Z axis. It enables the precise and correct positioning and interpolation of the respective spindle. The axis receives the positions via the addresses for contour descriptions with the G codes from milling machines in the processing program to produce milling contours at a turning workpiece. If a CNC turning machine has a C axis spindle, workpieces can be machined in the optimum time.

A further advantage is the quality: With regard to form and position tolerances, workpieces can be machined more precisely with drilling and milling tools in one clamping. The conventional programming of a C axis in G code requires more extensive knowledge of programming.

The following commands are required for the programming of the C axis spindle:

Command	Meaning
SETMS(1)	Activate the main spindle to master spindle (manufacturer-specific)
SETMS(3)	Activate the driven tool to master spindle (manufacturer-specific)
S1=	Programming the cutting speed / rotational speed MS (manufacturer-specific)
M1=3	Main spindle = clockwise rotation direction (manufacturer-specific)
M3=3	Driven tool spindle = clockwise rotation direction (manufacturer-specific)
M1=4	Main spindle = counter clockwise rotation direction (manufacturer-specific)
M3=4	Driven tool spindle = counter clockwise rotation direction (manufac.-specific)
M1=5	Main spindle stop (manufacturer-specific)
M3=5	Driven tool spindle stop (manufacturer-specific)
TRANSMIT	Front surface transformation (transformation milling into turning)
TRACYL(d)	Cylinder surface transformation with current working diameter
TRAFOOF	Deactivating all active transformations/frames
DIAMON	Dimensions in the diameter (G90 and G91)
DIAMOF	Dimensions in the radius (G90 and G91)
G17	Working plane X/Y, infeed direction Z – front surface
G18	Working plane Z/X, infeed direction Y – turning
G19	Working plane Y/Z, infeed direction X - cylinder surface



TRANSMIT

The TRANSMIT function enables the following services:

- Front surface drilling and milling of workpiece in one clamping situation.
- The Cartesian coordinate system can be used as a basis for programming these machining operations.
- The control transforms the programmed traversing movements of the Cartesian coordinate system to the traversing movements of the real machine axes X and C.

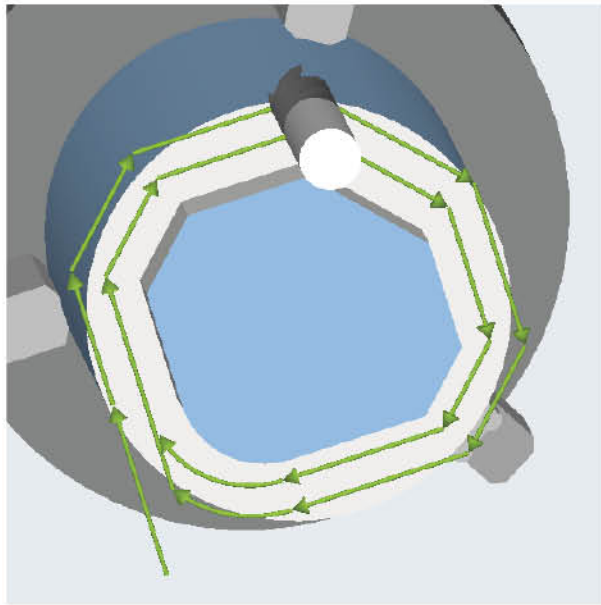


Figure 2.20: Milling on front face - G17

Description example for activating and deactivating the front surface machining:

Command	**Meaning**
SETMS(1)	Activate the main spindle to master spindle
G18	Working plane Z/X, infeed direction Y – turning
…..	Machining the turning operations
M1=5	Main spindle stop (here also “M5” possible)
SPOS[1]=0	Set master spindle to specified angle
SETMS(3)	Activate the driven tool spindle to master spindle
G17	Working plane X/Y
DIAMOF	Dimensions in the radius (G90 and G91)
TRANSMIT	Front face transformation (transformation milling into turning)
T=”…”	Tool call up
G0 X… Y… Z…	Start position machining
…	Machining the milling operations
TRAFOOF	Deactivating all active transformations/frames
DIAMON	Dimensions in the diameter (G90 and G91)
M3=5	Driven tool spindle stop (here also “M5” possible)
SETMS(1)	Activate the main spindle to master spindle
G18	Working plane Z/X, infeed direction Y – turning
G0 SUPA X500 Z500 D0 M9	Tool changing point

Front surface milling example with program Guide

Description:

A first programGuide milling exercise is to be programmed with the help of cycle calls for spigot milling.

Objective:

The program uses the above commands to mill contours at the front surface. At the end the program switches to the turning machining level in order to manufacture the outer chamfer at the diameter Ø100.

The exercise includes three milling contours. To produce the workpiece in the effectives way and economically as possible a pre milled contour is to use as blank contour for the next machining sequence. In order to focus the exercise to the new programming steps all work sequence be programmed in roughing mode without allowances.

The workflow sequences are to programmed:

- Program header
- Face cutting length 55 mm
- Programming the commands for the face machining in G17
- Milling contour WAF 80 mm
- Milling contour WAF 60 mm
- Milling contour rectangle 35 mm
- Chamfer milling on contours rectangle 35 mm, WAF 60 mm, WAF 80 mm
- Switch to the turning machining plane in G18
- Longitudinal turning the chamfer at the Ø100
- Program end

A new workpiece directory “EXERCISE_2_DT_DIN” should be created and opened in the editor. Create a new G code program in programGuide, like described in the previous modules.

Give the program a name, for example “EXERCISE_2_DT_DIN”.

This exercise explains the programming of contour milling machining with the program GUIDE by different means of contour descriptions, with polar coordinates, coordinates together with angles and incremental programming of the coordinate axes.

All contours are milled with down cut machining.

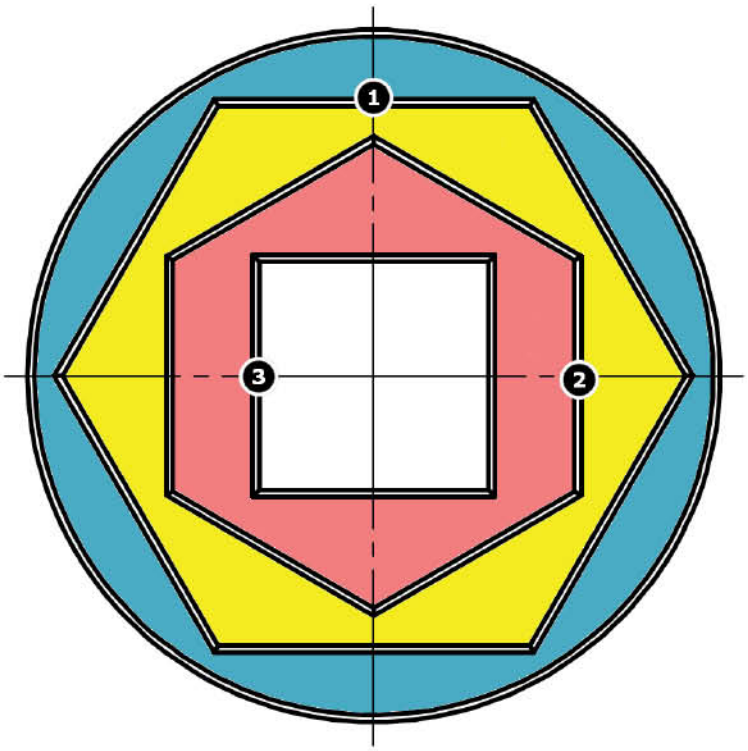
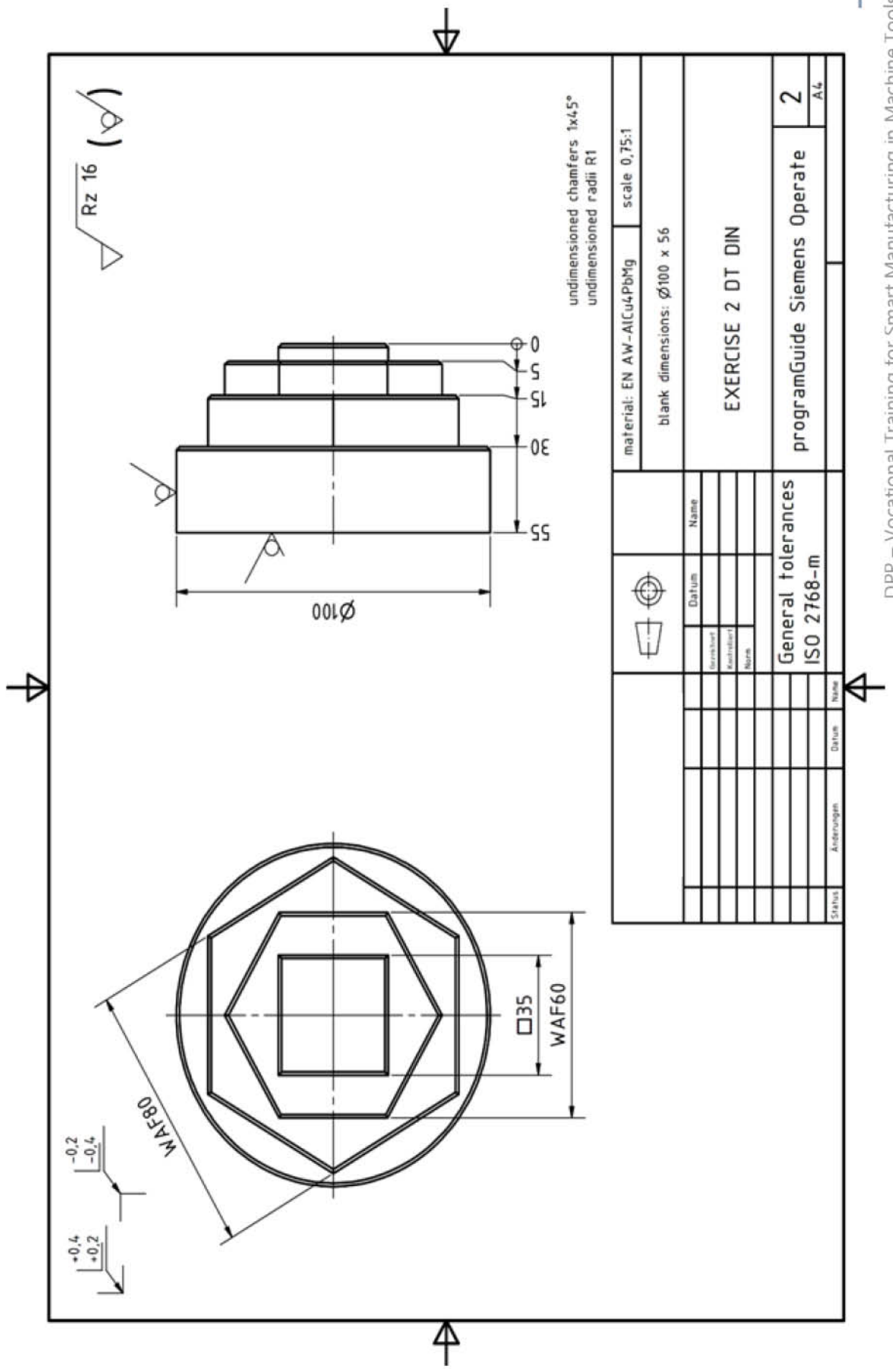


Figure 2.21: Milling on front face - G17

1. WAF 80 mm
   a. Blank contour Ø100 mm
   b. Milling contour hexagon 80 mm
   c. Contour description with polar coordinates

2. WAF 60 mm
   a. Blank contour hexagon 80 mm
   b. Milling contour hexagon 60 mm
   c. Contour description with X/Y coordinates and angle

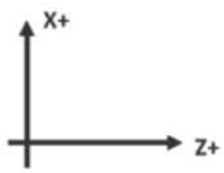
3. □ 35 mm
   a. Blank contour hexagon 60 mm
   b. Milling contour □ 35 mm
   c. Contour description with incremental X/Y coordinates

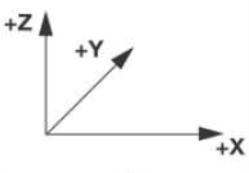


Training Module: Advanced DIN/ISO Programming and Operation of CNC Machines - Turning and Milling

The following tools and technology data are required for programming:

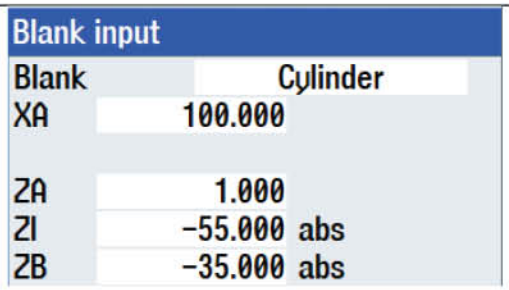
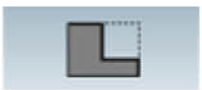
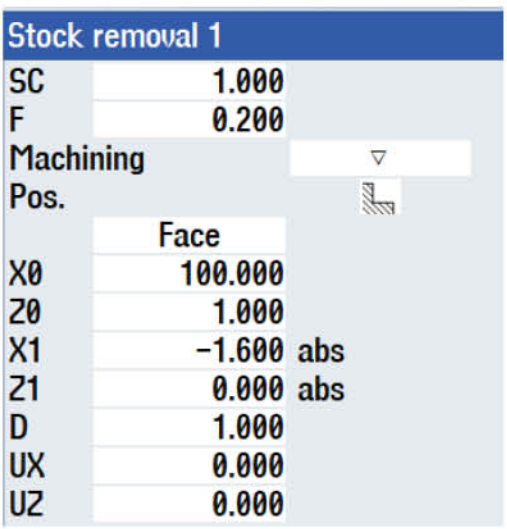
programGuide Siemens Operate VIETNAM 2021/2022		
Tool shape file	CNC Turning	
work piece: EXERCISE 2 DT DIN	blank material: EN-AW AlCu4PbMg	machine configuration:
drawing: 1(1)	blank dimensions: Ø100 x 55	X Z C

Tools for outer machining		
	Technological data	
Tool name in OFFSET	T="ROUGH T80 A"	
Cutting radius (mm)	0,8	
Cutting speed (m/min)	250	
Cutting depth ap = max.	2,5 / 1	
Cutting material	SC	
Feed rate per rev./pitch	0,3 / 0,12	
Angle/pitch/diameter Ds	80°	
		

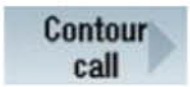
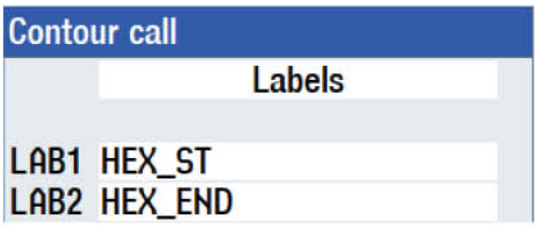
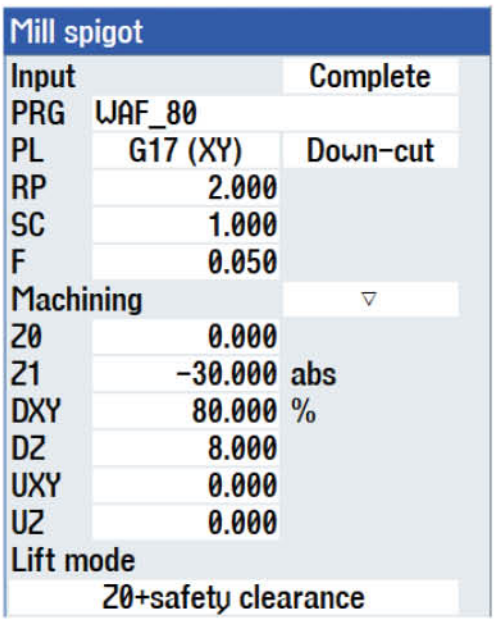
Tools for drilling and milling machining			
	Technological data		
Tool name in OFFSET	T="Cutter_16"	T="Deburrer_8"	
Tool Ø (mm)	16	8 x 90°	
Cutting speed (m/min)	150 / 175	105	
Cutting depth ap = max.	8	3	
Cutting material	SC	SC	
Feed tooth fz / rev. (mm)	0,05 / 0,04	0,025	
Number of teeth N	4	4	
			
	✓		
	G17	G17	

Open the new G code programGuide “EXERCISE_2_DT_DIN”:

1.	Program the first lines of the program by inserting the following G code commands: **N1 G18 G71 G90**
2. 	Insert now a blank for the simulation into the program. Press the HSK 1.6 “Various” to open the work area “Various”. The following input mask for the blank input opens.
3.	Enter following values/parameters for the blank:

	
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N2 WORKPIECE(,,,"CYLINDER",192,1,-55,-35,100)**
4.	Now insert the following program lines for the programming conditions and tool changing point: **N3 G0 SUPA X500 Z500 D0 M9** **N4 DIAMON** **N5 LIMS=3000** **N6 G54** **N7 MSG("FACING LENGTH 55 mm")**
5. 	Press the VSK 1.1 “Select tool”. The tool list window opens. Use the blue cursor keys to select the tool “ROUG_T80 A”. Press the VSK 1.8 “OK”. The program line **N8 T="ROUG_T80 A"** is inserted into the program.
6.	Program the following G code commands: **N9 G96 S250 M4** **N10 G95 F0.2**
7. 	Enter the following values into the input mask: 

	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N11** **N12 CYCLE951(100,1,-1.6,0,-1.6,0,1,1,0,0,12,0,0,0,1,0.2,0,2,1110000)**
8.	Program the following G code commands: **N13 G0 SUPA X500 Z500 D0 M9** **N14 M5** **N15 G54 G17 G97 G90** **N16 TRANSMIT** **N17 SETMS(3)** **N18 MSG("HEXAGON WAF80-POLARCOORD.")**
9.	Insert the following tool “CUTTER 16” into the program (see step 5) or program the following line by hand: **N19 T="CUTTER 16"**
10.	Program the following G code commands: **N20 G97 S2984 M3** **N21 G95 FZ=0.05** **N22 DIAMOF** **N23 G0 X0 Y60 Z2** **N24 G111 X0 Y0**
11. 	Program now the call up of the contour “CYCLE62”. Press now the VSK 1 “Contour”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
12. 	Insert the following “Labels” name into the input mask: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N25 CYCLE62(,2,"BLANK_ST","BLANK_END")**

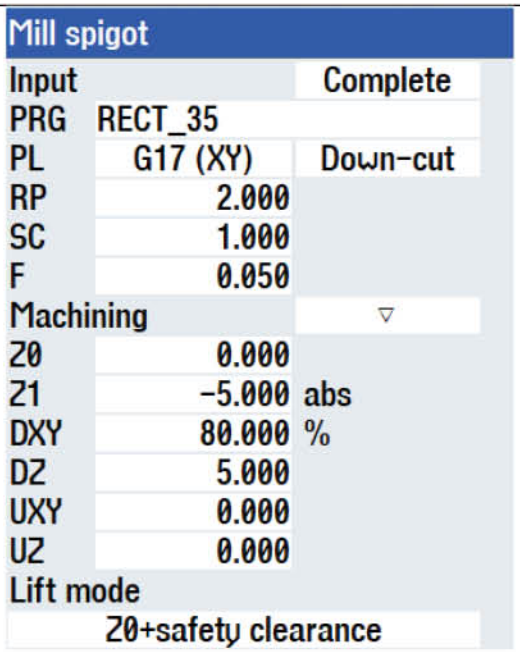
13. 	Program now the call up of the contour “CYCLE62”. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
14. 	Insert the following “Labels” name into the input mask:  Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N26 CYCLE62(,2,"HEX_ST","HEX_END")**
15. 	Now press HSK 1.5 “Milling” followed by VSK 8 “Cont. mill.” to call the “Milling” function. Now press VSK 6 “Spigot”. The input screen for the “Spigot” milling cycle opens.
16.	Enter the following values into the input mask: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N27 CYCLE63("WAF_80",1001,2,0,1,-30,0.05,,80,8,0,0,0,,,,,,1,2,,,,0,201,110)**
17.	Enter now the blank and hexagon contours behind the "M30" between the "Labels" you called in the “CYCLE62”. For the both “Labels” use the polar coordinates programming:

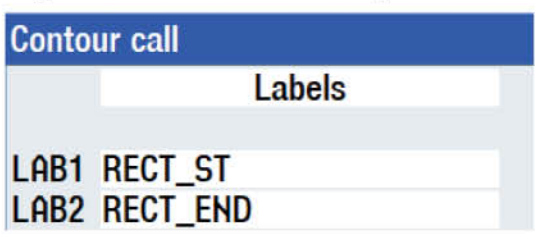
	N71 BLANK_ST: **N72 G17 G90 DIAMOF** **N73 G0 X50 Y0** **N74 G2 AP=360 RP=50** **N75 BLANK_END:** **N76 HEX_ST:** **N77 G17 G90 DIAMOF** **N78 G0 X0 Y40** **N79 G1 AP=60 RP=46.188** **N80 G1 AP=0 RP=46.188** **N81 G1 AP=-60 RP=46.188** **N82 G1 AP=-120 RP=46.188** **N83 G1 AP=-180 RP=46.188** **N84 G1 AP=-240 RP=46.188** **N85 G1 X0 Y40** **N86 HEX_END:**
18.	Continue to input the main program with the following G code commands: **N29 MSG("HEXAGON WAF60-DIN X/Y/ANG")** **N30 G0 X0 Y60 Z2**
19. 	Program now the call up of the contour “CYCLE62”. Press now the VSK 1 “Contour”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
20. 	Insert the following “Labels” name into the input mask: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N31 CYCLE62(,2,"BLANK2_ST","BLANK2_END")**

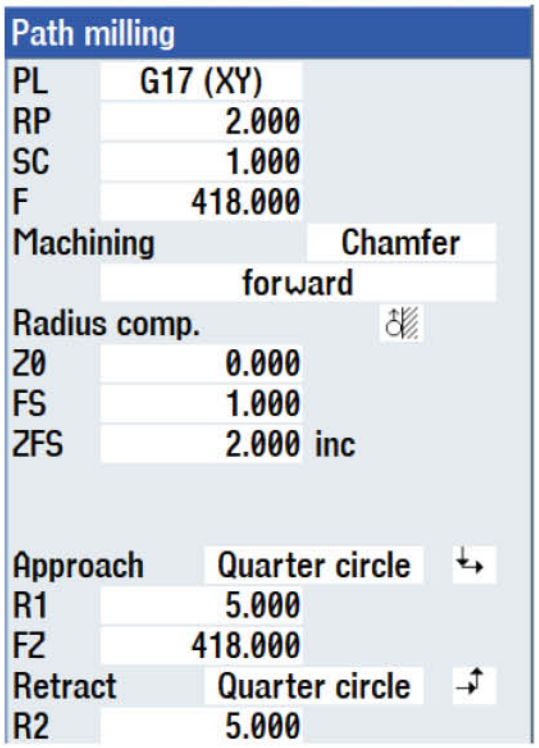
21. Contour call	Program now the call up of the contour “CYCLE62”. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
22. Accept	Insert the following “Labels” name into the input mask: **Contour call** Labels LAB1 HEX2_ST LAB2 HEX2_END Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N32 CYCLE62(,2,"HEX2_ST","HEX2_END")**
23. Milling Spigot	Now press HSK 1.5 “Milling” followed by VSK 8 “Cont. mill.” to call the “Milling” function. Now press VSK 6 “Spigot”. The input screen for the “Spigot” milling cycle opens.
24.	Enter the following values into the input mask: **Mill spigot** Input Complete PRG WAF_60 PL G17 (XY) Down-cut RP 2.000 SC 1.000 F 0.050 Machining ▽ Z0 0.000 Z1 -15.000 abs DXY 80.000 % DZ 8.000 UXY 0.000 UZ 0.000 Lift mode Z0+safety clearance
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N33 CYCLE63("WAF_60",1001,2,0,1,-15,0.05,,80,8,0,0,0,,,,,,1,2,,,,0,201,110)**

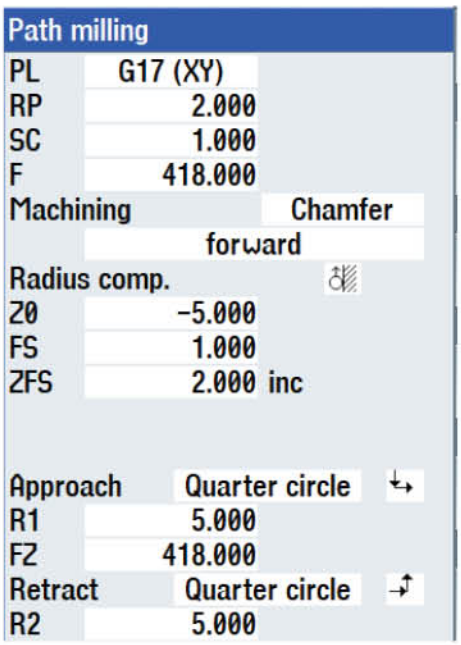
25.	Enter now the blank and hexagon contours behind the "M30" between the "Labels" you called in the “CYCLE62”. For the both “Labels” use the polar coordinates and X/Y coordinates together with angle programming: **N88 BLANK2_ST:** **N89 G17 G90 DIAMOF** **N90 G0 X0 Y40** **N91 G1 AP=60 RP=46.188** **N92 G1 AP=0 RP=46.188** **N93 G1 AP=-60 RP=46.188** **N94 G1 AP=-120 RP=46.188** **N95 G1 AP=-180 RP=46.188** **N96 G1 AP=-240 RP=46.188** **N97 G1 X0 Y40** **N98 BLANK2_END:** **N99 HEX2_ST:** **N100 G17 G90 DIAMOF** **N101 G0 X30 Y0** **N102 G1 Y-17.321 ANG=-90** **N103 G1 Y-34.641 ANG=-150** **N104 G1 X-30 ANG=-210** **N105 G1 Y17.321 ANG=90** **N106 G1 Y34.641 ANG=30** **N107 G1 X30 ANG=-30** **N108 G1 Y0 ANG=-90** **N109 HEX2_END:**
26.	Continue to input the main program with the following G code commands: **N34 MSG("RCTANGLE 35-DIN incemental X/Y")** **N35 G0 X0 Y60 Z2**
27. 	Program now the call up of the contour “CYCLE62”. Press now the VSK 1 “Contour”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.

28. 	Insert the following “Labels” name into the input mask: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N36 CYCLE62(,2,"BLANK3_ST","BLANK3_END")**
29. 	Program now the call up of the contour “CYCLE62”. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
30. 	Insert the following “Labels” name into the input mask: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N37 CYCLE62(,2,"RECT_ST","RECT_END")**
31. 	Now press HSK 1.5 “Milling” followed by VSK 8 “Cont. mill.” to call the “Milling” function. Now press VSK 6 “Spigot”. The input screen for the “Spigot” milling cycle opens.
32.	Enter the following values into the input mask:

	
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N38 CYCLE63("RECT_35",1001,2,0,1,-5,0.05,,80,5,0,0,0,,,,,,1,2,,,,0,201,110)**
33.	Enter now the blank and rectangle contours behind the "M30" between the "Labels" you called in the “CYCLE62”. For the both “Labels” use the X/Y coordinates together with angle and incremental programming for X/Y: **N111 BLANK3_ST:** **N112 G17 G90 DIAMOF** **N113 G0 X30 Y0** **N114 G1 Y-17.321 ANG=-90** **N115 G1 Y-34.641 ANG=-150** **N116 G1 X-30 ANG=-210** **N117 G1 Y17.321 ANG=90** **N118 G1 Y34.641 ANG=30** **N119 G1 X30 ANG=-30** **N120 G1 Y0 ANG=-90** **N121 BLANK3_END:** **N122 RECT_ST:** **N123 G17 G90 DIAMOF** **N124 G0 X17.5 Y0** **N125 G1 Y=IC(-17.5)** **N126 G1 X=IC(-35)** **N127 G1 Y=IC(35)**

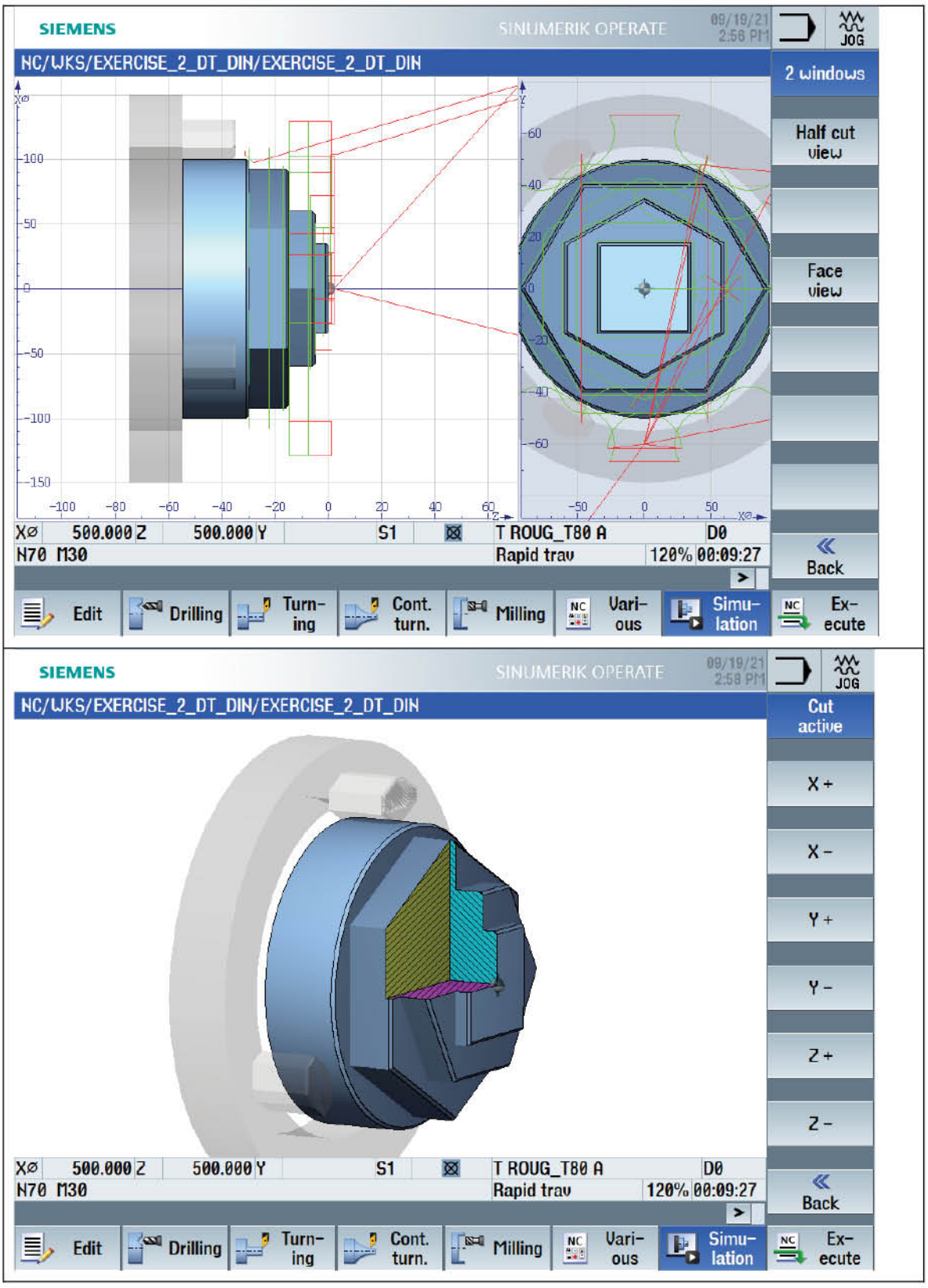
	N128 G1 X=IC(35) **N129 G1 Y=IC(-17.5)** **N130 RECT_END:**
34.	Continue to input the main program with the following G code commands: **N39 G0 SUPA X500 Z500 D0 M9** **N40 MSG("CHAMFER RECTANGLE 35")**
35.	Insert the following tool “DEBURRER 8” into the program (see step 5) or program the following line by hand: **N41 T="DEBURRER 8"**
36.	Program the following G code commands: **N42 G97 S4178** **N43 G94 F418 M3** **N44 G0 X0 Y60 Z2** **N45**
37. 	Program now the call up of the contour “CYCLE62”. Press now the VSK 1 “Contour”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
38. 	Insert the following “Labels” name into the input mask:  Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N46 CYCLE62(,2,"RECT_ST","RECT_END")**
39. 	Program now the path milling cycle “CYCLE72”. Press the HSK 1.5 “Milling” to open the technology “Contour milling”. Press the VSK 1.8 “Cont. mill.” to open the technology “Contour milling”. Press the VSK 2 "Path milling" to the input screen for the technology to open.

Path milling	
40.	Enter the following values into the input mask: 
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N47** **CYCLE72("",2,0,1,20,2,0.5,0.5,418,418,305,41,2,5,0.1,2,5,0,1,2,101,1011,101)**
41.	Program the following message: **N48 MSG("CHAMFER WAF60")**
42. Back Contour Contour call	Program now the call up of the contour “CYCLE62”. Press now the VSK 8 “Back”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
43.	Insert the following “Labels” name into the input mask:

Accept	 Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N49 CYCLE62(,2,"HEX2_ST","HEX2_END")**
44. Back Path milling	Program now the path milling cycle “CYCLE72”. Press now the VSK 8 “Back”. Press the VSK 2 "Path milling" to the input screen for the technology to open.
45.	Enter the following values into the input mask: 
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N50 CYCLE72("",2,-** **5,1,20,2,0.5,0.5,418,418,305,41,2,5,0.1,2,5,0,1,2,101,1011,101)**
46.	Program the following message: **N51 MSG("CHAMFER WAF80")**
47. Back	Program now the call up of the contour “CYCLE62”. Press now the VSK 8 “Back”. The vertical softkey bar with functions for creating or calling up a contour opens.

Contour Contour call	Press the VSK 2 “Contour call” to open the input mask for the contour call up.
48. Accept	Insert the following “Labels” name into the input mask: Contour call Labels LAB1 HEX_ST LAB2 HEX_END Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N52 CYCLE62(,2,"HEX_ST","HEX_END")**
49. Back Path milling	Program now the path milling cycle “CYCLE72”. Press now the VSK 8 “Back”. Press the VSK 2 "Path milling" to the input screen for the technology to open.
50.	Enter the following values into the input mask: Path milling PL G17 (XY) RP 2.000 SC 1.000 F 418.000 Machining Chamfer forward Radius comp. Z0 -15.000 FS 1.000 ZFS 2.000 inc Approach Quarter circle R1 5.000 FZ 418.000 Retract Quarter circle R2 5.000
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N53 CYCLE72("",2,-15,1,20,2,0.5,0.5,418,418,305,41,2,5,0.1,2,5,0,1,2,101,1011,101)**

51.	Program the following G code commands: **N54 G0 Z5** **N55 TRAFOOF** **N56 DIAMON** **N57 M3=5** **N58 SETMS(1)** **N59 G18** **N60 G0 SUPA X500 Z500 D0 M9** **N61 G54** **N62 MSG("CHAMFER Ø100")**
52.	Insert the tool "ROUG_T80 A" into the program (see step 5) or program the following line by hand: **N63 T="ROUG_T80 A"**
53.	Program the following G code commands: **N64 G96 S250 M4** **N65 G95 F0.2** **N66 G0 G42 X96 Z-29** **N67 G1 X102 ANG=135** **N68 G0 G40 X105** **N69 G0 SUPA X500 Z500 D0 M9** **N70 M30**
54.	Finally check the machining with the simulation by using of different views.



Task: Create blocks in the program in accordance with the work sequences and save the setup data for machining. Save the program on an external disc and transfer it to the machine control.

3.5. Applying of milling and drilling operations with C and Y-axis

General introduction

Contour programming is the solution for effectively programming complex contours in turned and milled workpieces. With the aid of the integrated contour editor and the corresponding machining cycles in SINUMERIK Operate, contours can be created and processed directly on the CNC control system. This module briefly demonstrates the possibilities using the example of contour programming during milling with driven tools in conjunction with program GUIDE.

The contour definition and machining are available in the program editor via the "Contour Milling" soft key. The "Contour" soft key defines the contour and programs the contour call up.The other soft keys represent the cycles for the defined contour, for example, the cycles for path milling and for pocket and stigot machining.

Front surface milling example with contour editor

Description:

A G code program (Kidney) is to be programmed with programGUIDE with the aid of various milling cycles (spigot).

Objective:

With the help of this example, we will get to know the essential functions for entering complex contours via graphical input.

The workflow sequences are to programmed:

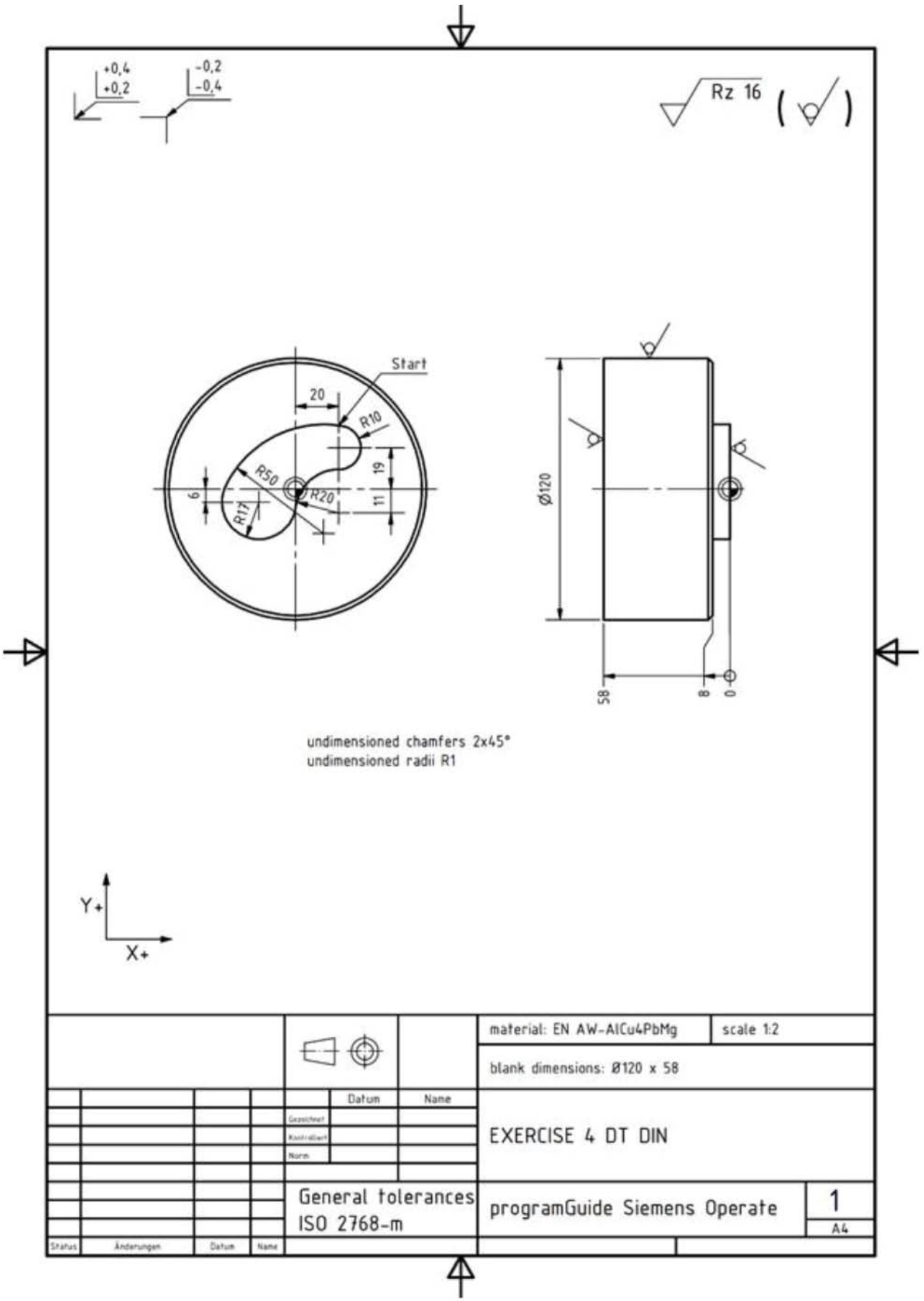
- Program header
- Programming the commands for the face machining in G17
- Pre milling the kidney contour
- Finish milling the kidney contour base
- Finish milling the kidney contour wall
- Chamfer milling on the kidney contour 0,4 mm x 45°
- Chamfer milling on the Ø120 with 2 mm x 45°
- Switch to the turning machining plane in G18
- Program end

A new workpiece directory "EXERCISE_4_DT_DIN" should be created and opened in the editor.

Create a new G code program in programGuide, like described in the previous modules.

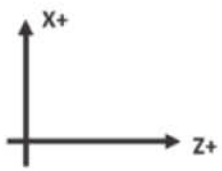
Give the program a name, for example "EXERCISE_4_DT_DIN".

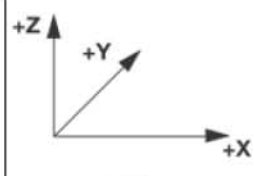
All contours are milled with down cut machining.


+0,4
+0,2
-0,2
-0,4
Rz 16
Start
20
R10
R50
R20
R17
19
11
6
Ø120
58
8
0
undimensioned chamfers 2x45°
undimensioned radii R1
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:2
blank dimensions: Ø120 x 58
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE 4 DT DIN
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name

The following tools and technology data are required for programming:

programGuide Siemens Operate VIETNAM 2021/2022		
Tool shape file	CNC Turning	
work piece: EXERCISE 4 DT DIN	blank material: EN-AW AlCu4PbMg	machine configuration:
drawing: 1(1)	blank dimensions: Ø120 x 58	X Z C

Tools for outer machining		
	Technological data	
Tool name in OFFSET	T="ROUGH T80 A"	
Cutting radius (mm)	0,8	
Cutting speed (m/min)	250	
Cutting depth ap = max.	2,5 / 1	
Cutting material	SC	
Feed rate per rev./pitch	0,3 / 0,12	
Angle/pitch/diameter Ds	80°	
		

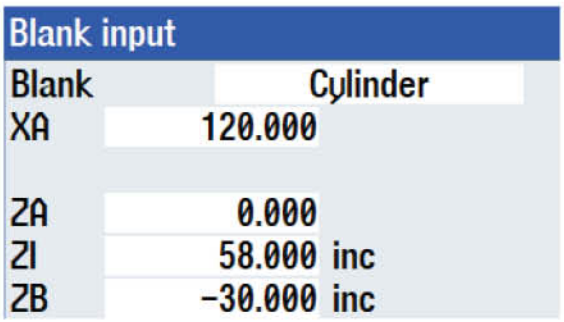
Tools for drilling and milling machining			
	Technological data		
Tool name in OFFSET	T="Cutter_ 16"	T="Deburre r_8"	
Tool Ø (mm)	16	8 x 90°	
Cutting speed (m/min)	150 / 175	105	
Cutting depth ap = max.	8	3	
Cutting material	SC	SC	
Feed tooth fz / rev. (mm)	0,05 / 0,04	0,025	
Number of teeth N	4	4	
			
	✓		
	G17	G17	

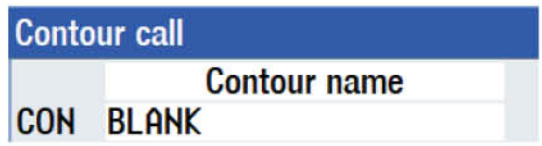
The following positions are programmed as the contour starting points:

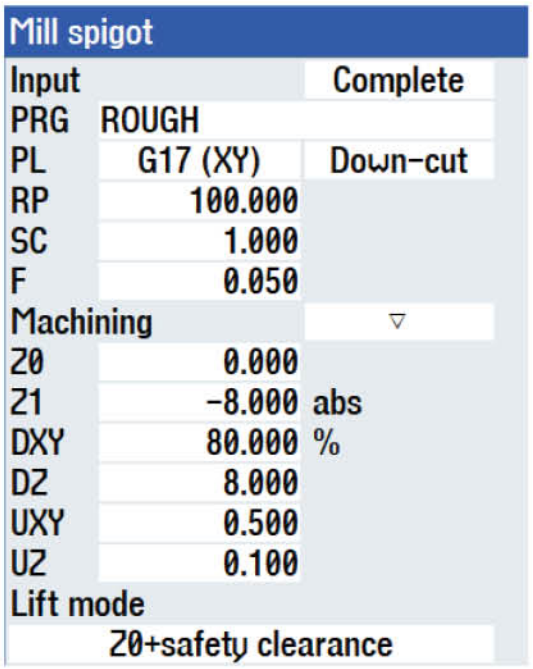
Contour 1 – the blank: X0 Y60

Contour 2 – the kidney: X20 Y29

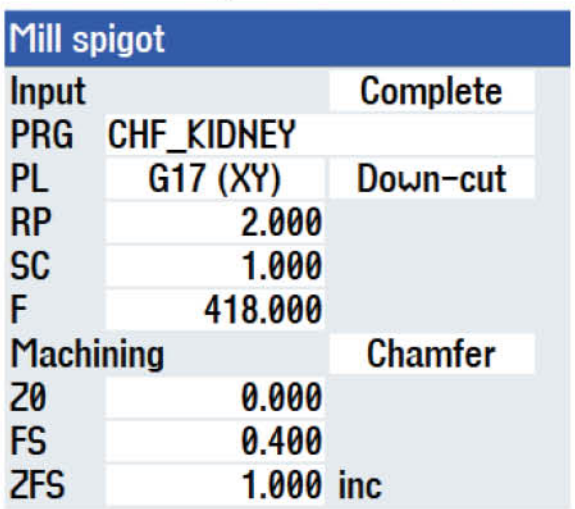
Open the new G code programGuide “EXERCISE_4_DT_DIN”:

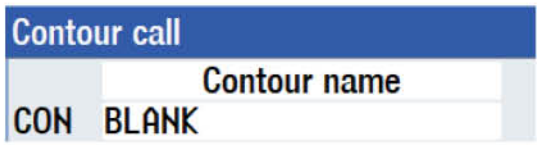
1.	Program the first lines of the program by inserting the following G code commands: **N1 G18 G71 G90**
2. 	Insert now a blank for the simulation into the program. Press the HSK 1.6 “Various” to open the work area “Various”. The following input mask for the blank input opens.
3.	Enter following values/parameters for the blank: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program:
4.	Now insert the following program lines for the programming conditions and tool changing point: **N3 G0 SUPA X500 Z500 D0 M9** **N4 G54** **N5 G17** **N6 DIAMOF** **N7 SPOS[1]=0** **N8 TRANSMIT** **N9 SETMS(3)**
5. 	Press the VSK 1.1 “Select tool”. The tool list window opens. Use the blue cursor keys to select the tool “CUTTER 16”. Press the VSK 1.8 “OK”. The program line **N10 T="CUTTER 16"**is inserted into the program.
6.	Program the following G code commands: **N11 G97 S2984 M3** **N12 G95 FZ=0.05** **N13 G0 X0 Y75 Z5** **N14 M8**

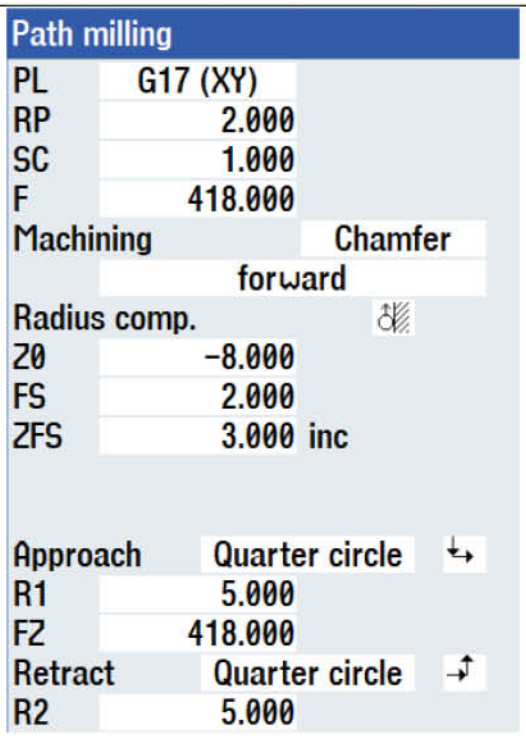
7. 	Program now the call up of the contour “CYCLE62”. Press now the VSK 1 “Contour”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
8. 	Insert the following “Contour name” into the input mask:  Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N15 CYCLE62("BLANK",1,,)**
9. 	Program now the call up of the contour “CYCLE62”. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
10. 	Insert the following “Contour name” into the input mask: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N16 CYCLE62("KIDNEY",1,,)**
11. 	Now press HSK 1.5 “Milling” followed by VSK 8 “Cont. mill.” to call the “Milling” function. Now press VSK 6 “Spigot”. The input screen for the “Spigot” milling cycle opens.

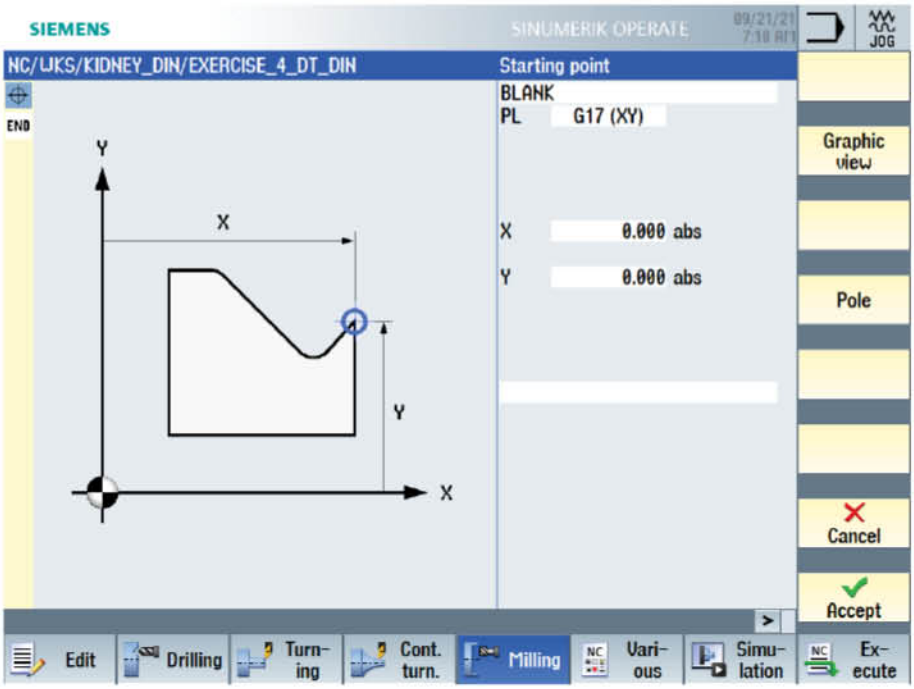
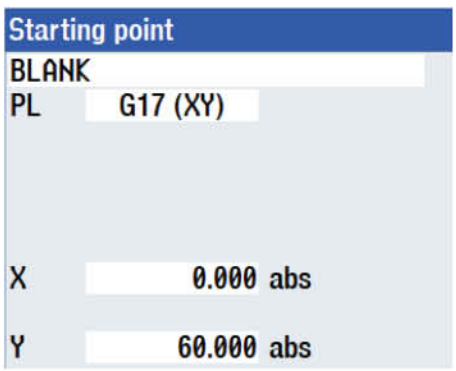
12.	Enter the following values into the input mask: 
 	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N17 CYCLE63("ROUGH",1001,100,0,1,-8,0.05,,80,8,0.5,0.1,0,,,,,,1,2,,,,0,201,110)** Note: It´s up to the programmer to enter now in the graphic editor the contours “BLANK” and “KIDNEY”. Alternative you can continue with the main program. After this is finished you can enter the contours “BLANK” and “KIDNEY” at the end of the spigot programming.
13. 	Now press HSK 1.5 “Milling” followed by VSK 8 “Cont. mill.” to call the “Milling” function. Now press VSK 6 “Spigot”. The input screen for the “Spigot” milling cycle opens.
14.	Enter the following values into the input mask:

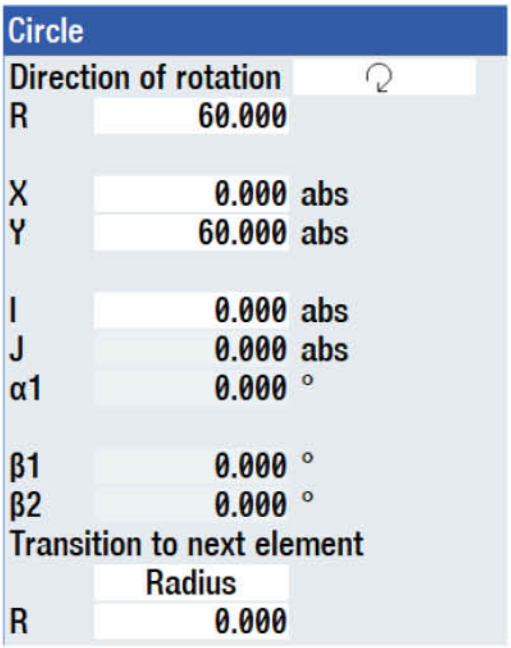
	Mill spigot Input Complete PRG FIN_BASE PL G17 (XY) Down-cut RP 2.000 SC 1.000 F 0.050 Machining ▽▽▽Base Z0 0.000 Z1 8.000 inc DXY 80.000 % UXY 0.500 UZ 0.100 Lift mode Z0+safety clearance
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N18** **CYCLE63("FIN_BASE",1003,2,0,1,8,0.05,,80,3,0.5,0.1,0,,,,,,1,2,,,,0,201,111)**
15. 	Now press HSK 1.5 “Milling” followed by VSK 8 “Cont. mill.” to call the “Milling” function. Now press VSK 6 “Spigot”. The input screen for the “Spigot” milling cycle opens.
16.	Enter the following values into the input mask: Mill spigot Input Complete PRG FIN_WALL PL G17 (XY) Down-cut RP 2.000 SC 1.000 F 0.050 Machining ▽▽▽Wall Z0 0.000 Z1 8.000 inc DZ 8.000 UXY 0.500 Lift mode Z0+safety clearance

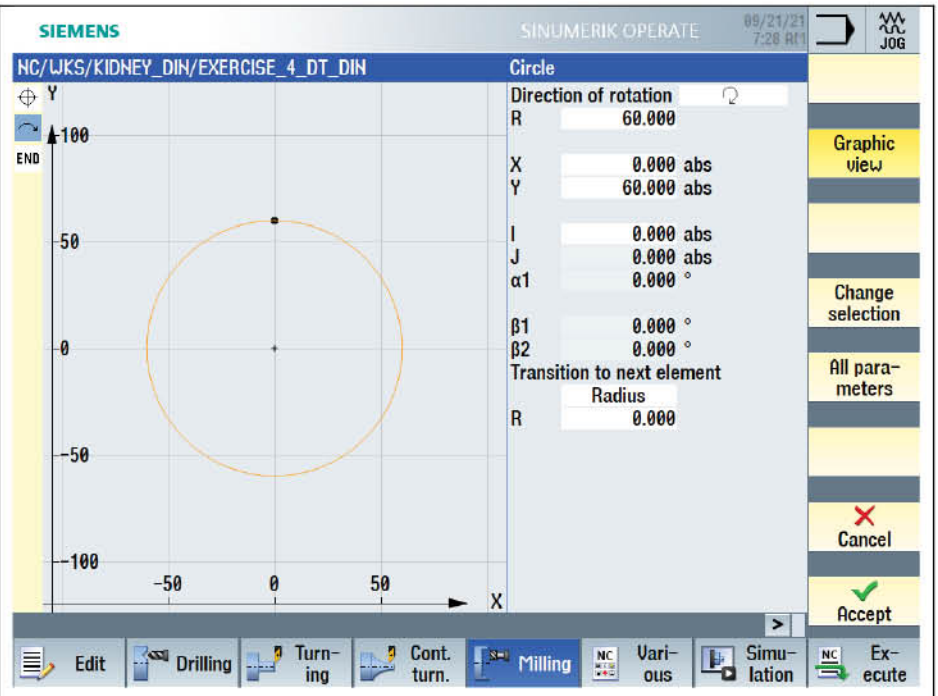
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N19** **CYCLE63("FIN_WALL",1004,2,0,1,8,0.05,,50,8,0.5,0.1,0,,,,,,1,2,,,,0,201,111)**
17.	Program the following G code commands: **N20 G0 SUPA X500 Z500 D0 M**
18.	Insert the following tool “DEBURRER 8” into the program (see step 5) or program the following line by hand: **N21 T="DEBURRER 8"**
19.	Program the following G code commands: **N22 G97 S4178** **N23 G94 F418 M3** **N24 G0 X0 Y55 Z5** **N25 M8**
20. 	Now press HSK 1.5 “Milling” followed by VSK 8 “Cont. mill.” to call the “Milling” function. Now press VSK 6 “Spigot”. The input screen for the “Spigot” milling cycle opens.
21.	Enter the following values into the input mask: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N26** **CYCLE63("CHF_KIDNEY",1005,2,0,1,5,418,,50,5,0.5,0.1,0,,,,,,0.4,1,,,,0,201,111)** **N27**

22. 	Program now the call up of the contour “CYCLE62”. Press now the VSK 1 “Contour”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.
23. 	Insert the following “Contour name” into the input mask:  Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N28 CYCLE62("BLANK",1,,)**
24. 	Program now the path milling cycle “CYCLE72”. Press the HSK 1.5 “Milling” to open the technology “Contour milling”. Press the VSK 1.8 “Cont. mill.” to open the technology “Contour milling”. Press the VSK 2 "Path milling" to the input screen for the technology to open.
25.	Enter the following values into the input mask:

	
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N29 CYCLE72("",2,-8,1,20,2,0.5,0.5,418,418,305,41,2,5,0.1,2,5,0,2,3,101,1011,101)**
26.	Program the following G code commands: **N30 G0 Z5** **N31 TRAFOOF** **N32 SETMS(1)** **N33 G18** **N34 DIAMON** **N35 G0 SUPA X500 Z500 D0 M9** **N36 M30** Note: Now we start to enter in the “Graphic editor” the contours “BLANK” and “KIDNEY”.
27. 	Program now the call up of the contour “CYCLE62” the first “BLANK MATERIAL” contour. Now press VSK 1 “Contour”. The vertical softkey bar to create and call a new contour opens. Press VSK 1 “New contour” to call a new contour. The input screen to assign a contour name opens. Assign the following name for the first contour

Create new	
Accept	The contour editor with the input screen for the contour definition opens automatically.  Enter the starting point of the contour:
28.	Enter the following coordinates as starting point: 
Accept	Confirm the entries with VSK 8 “Accept”.

29. 	Extend the contour with a circular movement in the clockwise direction. Press VSK 1.5 “Circle”. The input window for the contour element of a circular movement opens.
	Enter the coordinates as follows:  Note: When entering the circle center point in X (“I”), a dialog selection is opened. Note the input screen and the changing color of the graphic if you press the “Select dialog” softkey several times. If your entries in the input screen are correct, press the “Accept dialog” softkey to accept these entries.
	Confirm the entries with VSK 8 “Accept”.
30.	Now check your entries for the first contour. The following screen with the finished contour definition is displayed:



Finish the contour definition by pressing VSK 8 “Accept”.

31. The following G code lines are inserted in the program:

```
N36 M30¶
¶
N37 E_LAB_A_BLANK: ;#SM Z: 2¶
G17 G90 DIAMOF; *GP*¶
G0 X0 Y60 ; *GP*¶
G2 I=AC(0) J=AC(0) ; *GP*¶
E_LAB_E_BLANK: ¶
```

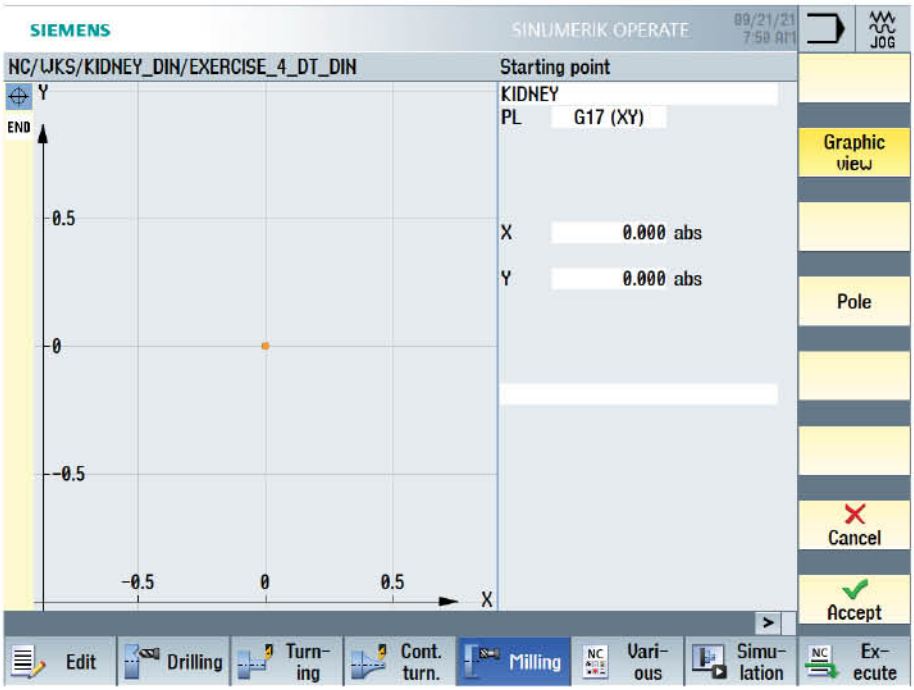
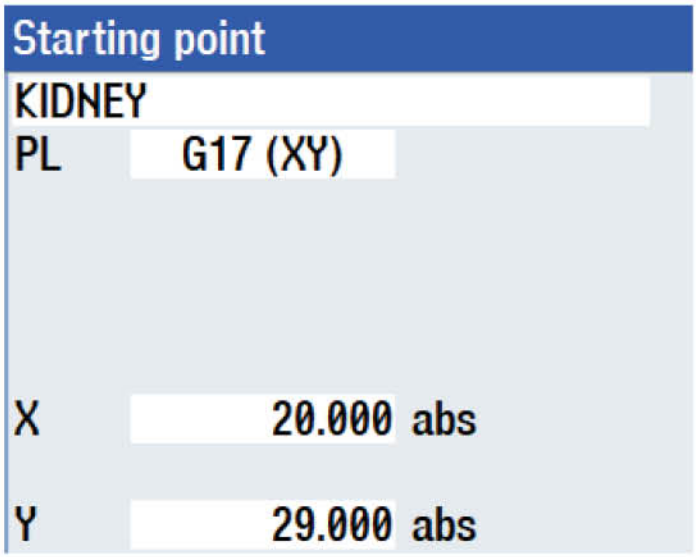
32. Now insert the contour definition for the second contour “KIDNEY”.

Now press VSK 1 “Contour”.

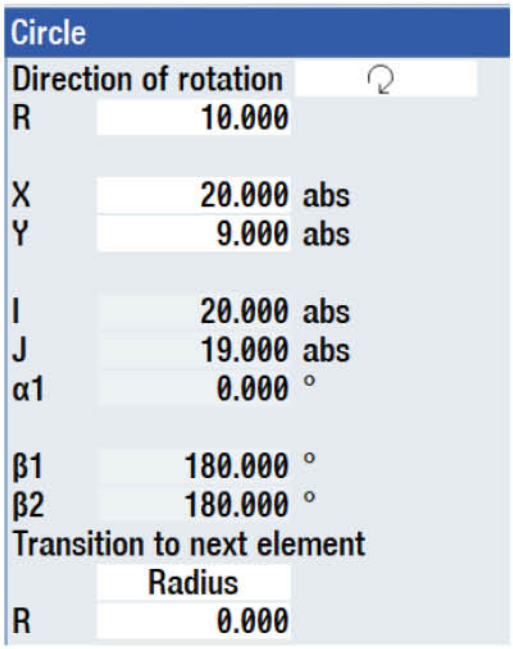
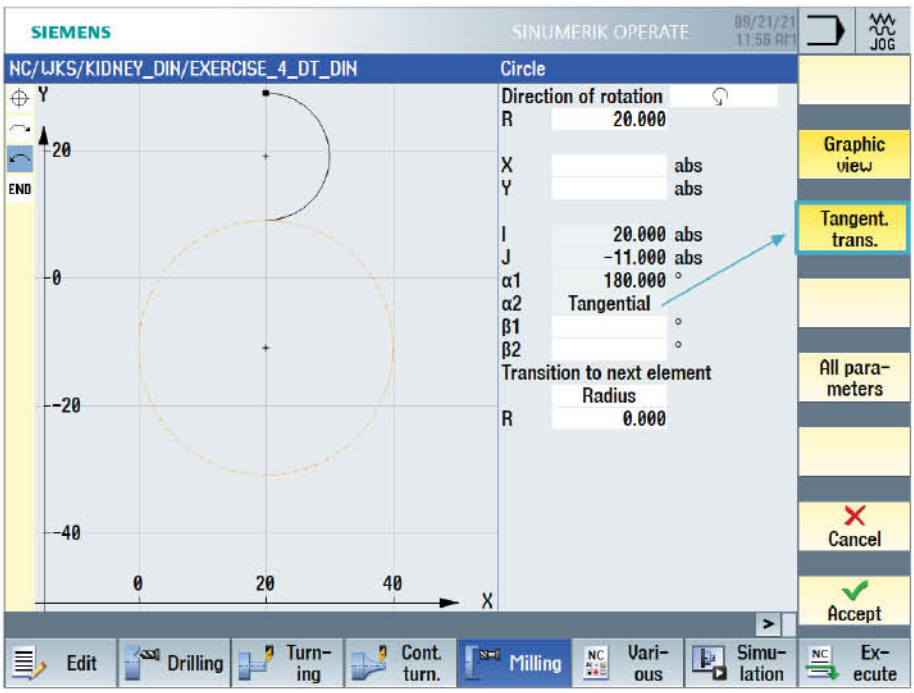
The vertical softkey bar to create and call a new contour opens.

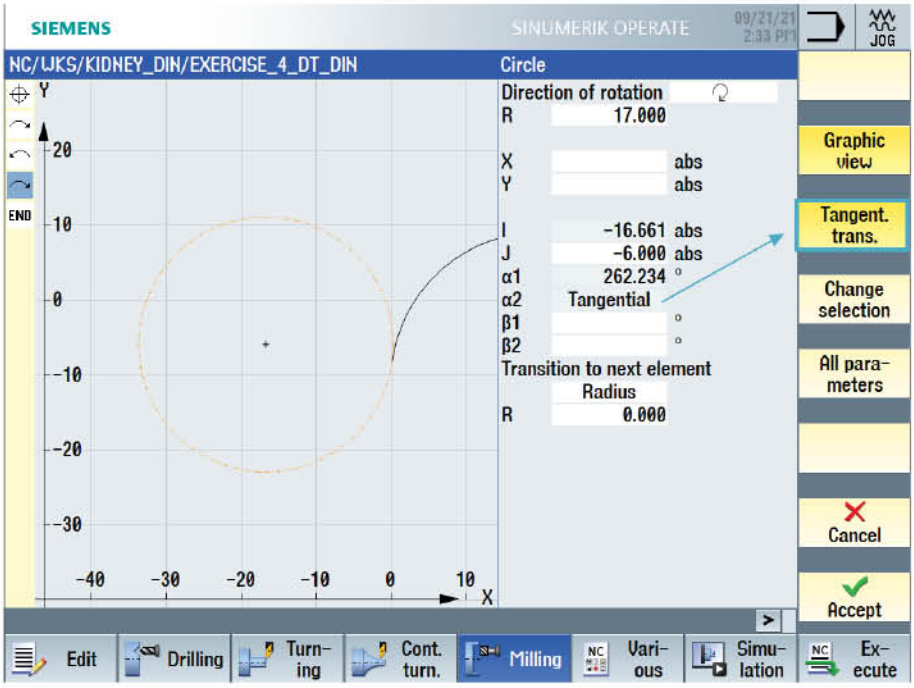
Press VSK 1 “New contour” to call a new contour.

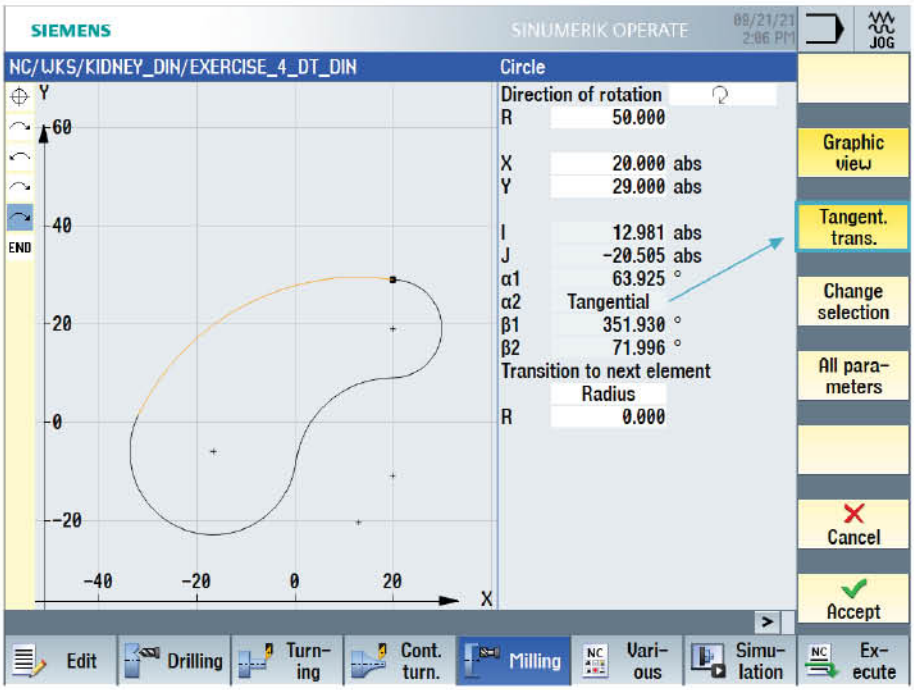
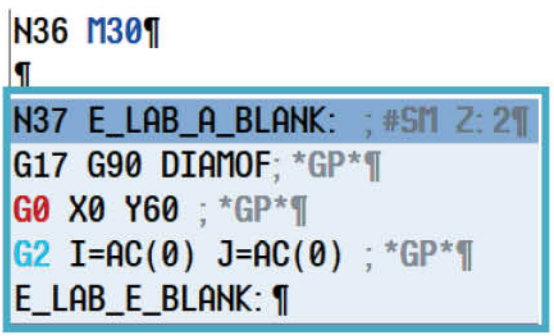
33. Assign the following name for the second contour:

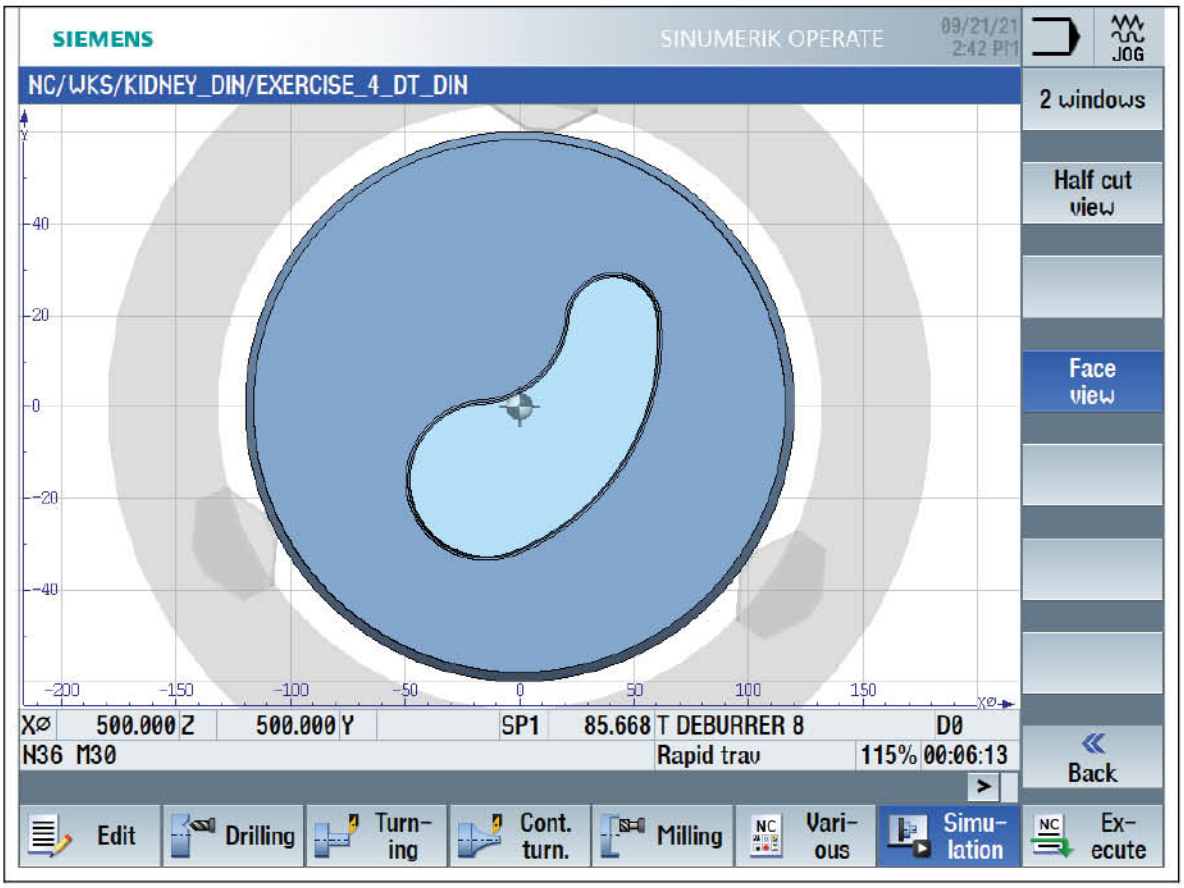
Accept	The contour editor with the input screen for the contour definition opens automatically.
	
34.	Enter the starting point of the contour: 
Accept	Confirm the entries with VSK 8 “Accept”.

35.	Extend the contour with a circular movement in the clockwise direction. Press VSK 1.5 “Circle”. The input window for the contour element of a circular movement opens.

Accept	Enter the coordinates as follows and confirm the entries with VSK 8 “Accept”. 
36.	Extend the contour with a circular movement in the counter clockwise direction. Press VSK 1.5 “Circle”. The input window for the contour element of a circular movement opens. Note: When entering the parameters, activate the “Tangent prev. elem.” softkey to receive the tangential transition to the previous element.
Accept	Enter the coordinates as follows and confirm the entries with VSK 8 “Accept”. 

37.	Extend the contour with a circular movement in the clockwise direction. Press VSK 1.5 “Circle”. The input window for the contour element of a circular movement opens. Note: When entering the parameters, activate the “Tangent prev. elem.” softkey to receive the tangential transition to the previous element.
	Enter the coordinates as follows and confirm the entries with VSK 8 “Accept”.  Note: When entering the circle center point in Y (“J”), a dialog selection is opened. Note the input screen and the changing color of the graphic if you press the “Select dialog” softkey several times. If your entries in the input screen are correct, press the “Accept dialog” softkey to accept these entries.
38.	Extend the contour with a circular movement in the clockwise direction. Press VSK 1.5 “Circle”. The input window for the contour element of a circular movement opens. Note: When entering the parameters, activate the “Tangent prev. elem.” softkey to receive the tangential transition to the previous element.

	Enter the coordinates as follows and confirm the entries with VSK 8 “Accept”.  Note: When entering the circle end point in X and Y, a dialog selection is opened. Note the input screen and the changing color of the graphic if you press the “Select dialog” softkey several times. If your entries in the input screen are correct, press the “Accept dialog” softkey to accept these entries.
	Finish the contour definition by pressing VSK 8 “Accept”.
39.	The following G code lines are inserted in the program: 
40.	Finally check the machining with the simulation by using of different views.


SIEMENS
SINUMERIK OPERATE
09/21/21
2:42 PM
JOG
NC/WKS/KIDNEY_DIN/EXERCISE_4_DT_DIN
2 windows
Half cut view
Face view
Back
XØ 500.000 Z 500.000 Y SP1 85.668 T DEBURRER 8 D0
N36 M30 Rapid trav 115% 00:06:13
Edit
Drilling
Turn-ing
Cont. turn.
Milling
Vari-ous
Simu-lation
Ex-ecute

3.6. Working areas of machines regard to the frames and how to calculate them for programming

Example calculations of the work piece zero points (WPZ)

Calculation parameters for G54 (-G57)	Main spindle – leading spindle
Chuck dimension ZC1	100
Jaw dimension ZS1	25
Outside chucking length to WPZ CL1	35
Chucking depth	25

Calculation parameters for G55 (-G57)	Counter spindle – following spindle
Chuck dimension ZC2	100
Jaw dimension ZS2	25
Outside chucking length to WPZ CL2	39
Processing position CS = Z2	1000
Chucking depth	17

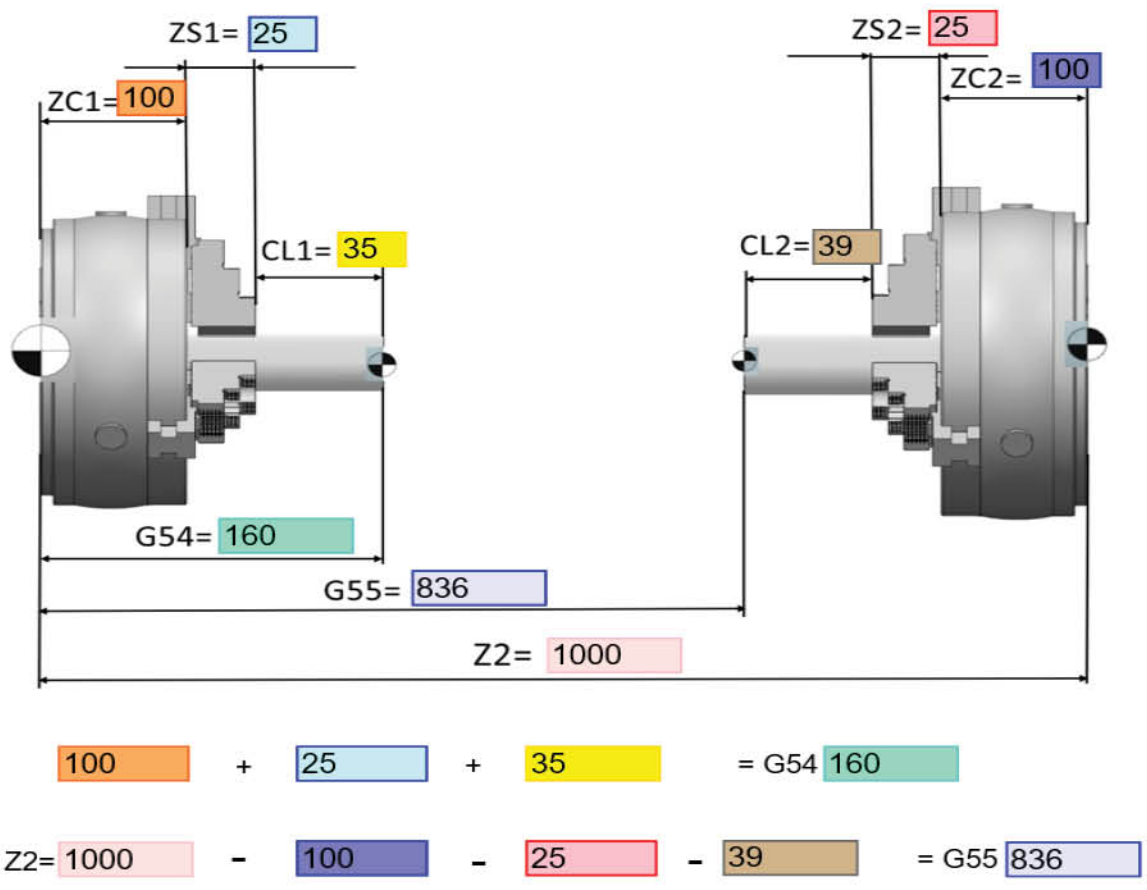


Figure 2.22: Calculations of the work piece zero points (WPZ)

3.7. Advanced programming of exercises with counter spindle and driven tools with the cycles and G-codes

General introduction

If a CNC turning machine has a counter spindle, workpieces can be machined in the optimum time. A further advantage is the quality: With regard to form and position tolerances, workpieces can be machined more precisely with two spindles.

The conventional programming of a counter spindle in G code requires more extensive knowledge of programming.

The following must be taken into account when creating a G code program for the counter spindle:

- Selection of the tool for the counter spindle machining
- Work offsets for the counter spindle machining
- Higher-level language commands, e.g. for the spindle coupling, must be programmed
- Machining on the counter spindle is performed in the direction of the Z axis in the positive direction

The counter spindle programming with program Guide facilitates working with two spindles.

This is described in detail on the following pages using an application example.

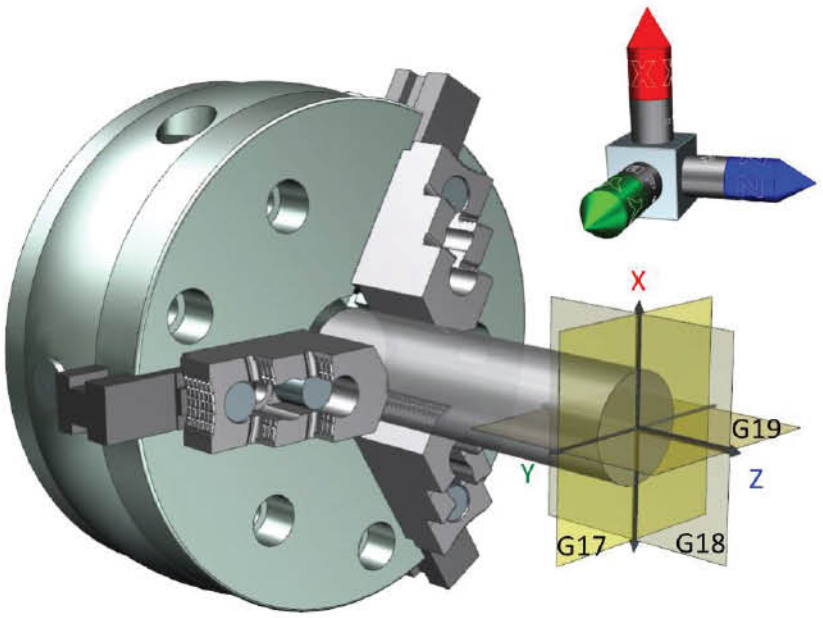


Figure 2.23: The workpiece is clamped on the self-centering 3-prong access plate

Working area

In order to implement working with the counter spindle, it is important to know how the working area of a machine is structured with regard to the frames, how these functions and how the counter spindle programming with program Guide uses and calculates them. This is shown here:

M = Machine zero point, defined by the machine manufacturer.

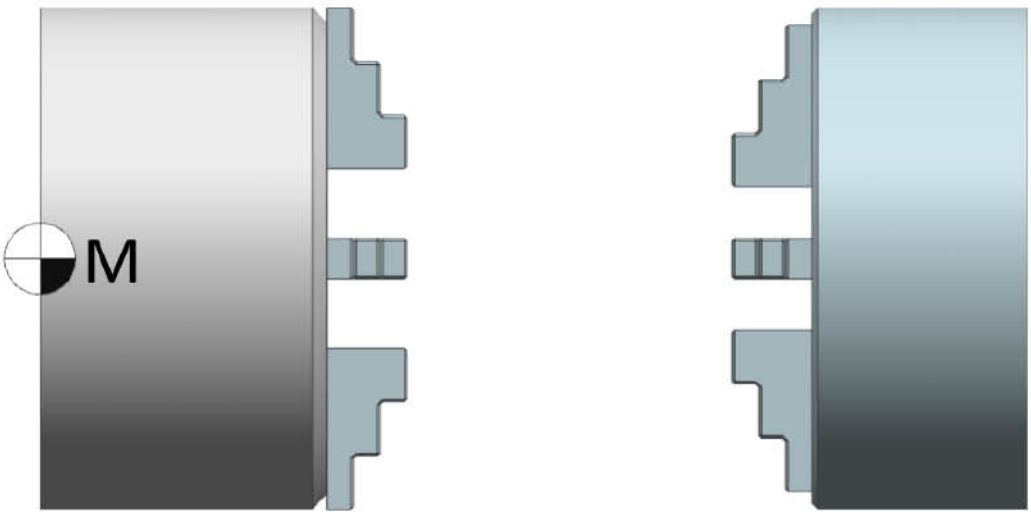


Figure 2.24: Main spindle chuck and opposite chuck (reverse chuck)

On a turning machine, the machine zero is on the surface behind the chuck of the main spindle. This is always the reference point for all other values and frames, such as work offsets.

The specification of the machining position for the counter spindle is the distance shown here. Designated as "ZS2W" in the calculation for programming with program Guide.

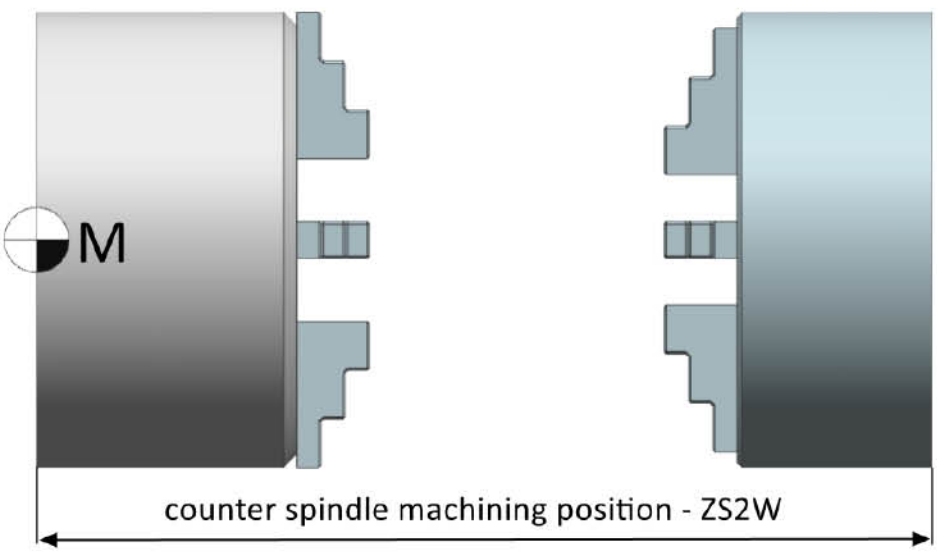


Figure 2.25: Opposite chuck position (reverse chuck)

If a value is not defined for the machining position in program Guide, the maximum machining position is approached automatically. In this way, it is not possible to travel to the machine zero with the counter spindle.

The chuck dimensions are described in the next step.

For the main spindle chuck, starting from the machine zero point “M”.

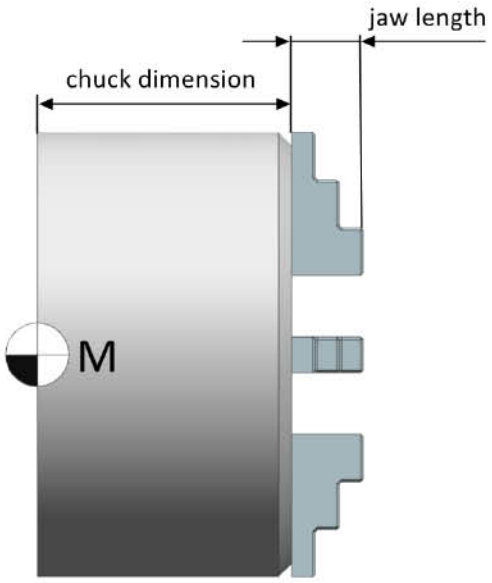


Figure 2.26: Spindle chuck size

For the counter spindle, starting from the current machining position in the program.

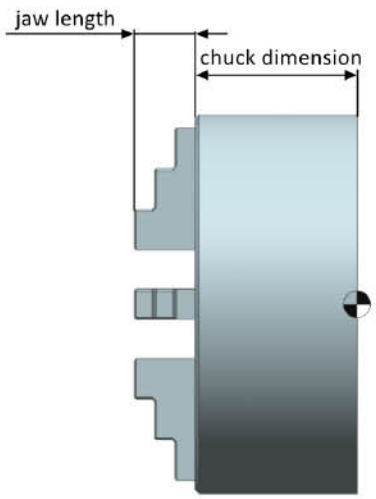


Figure 2.27: Dimension of chuck opposite chuck (reverse chuck)

The working area has now been defined. All values are included in the calculation of the current machining position of the counter spindle.

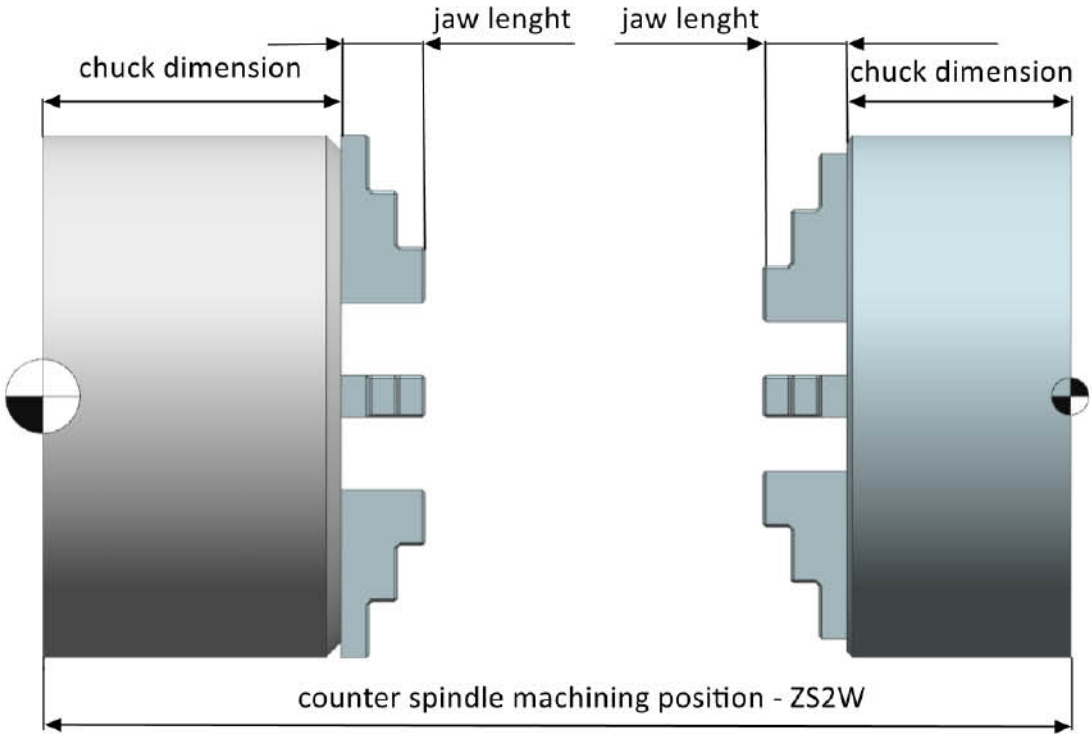
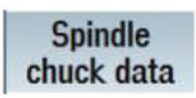
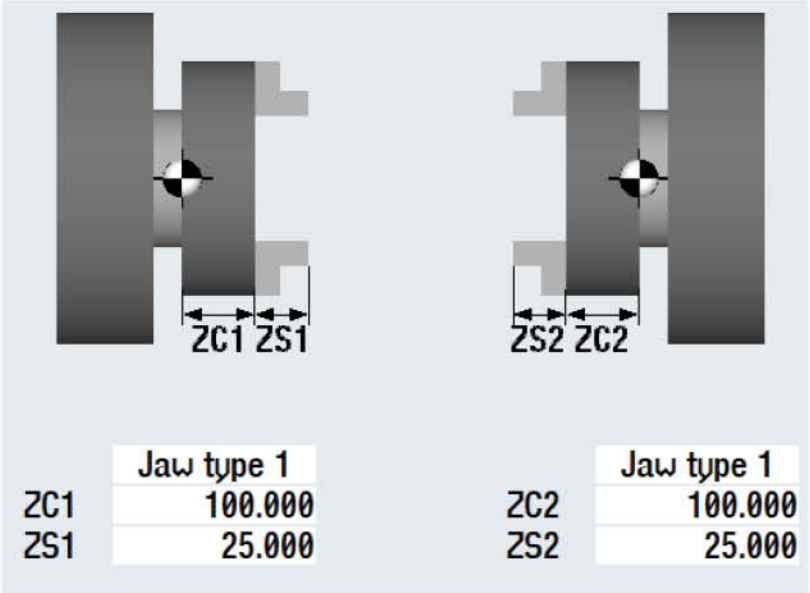
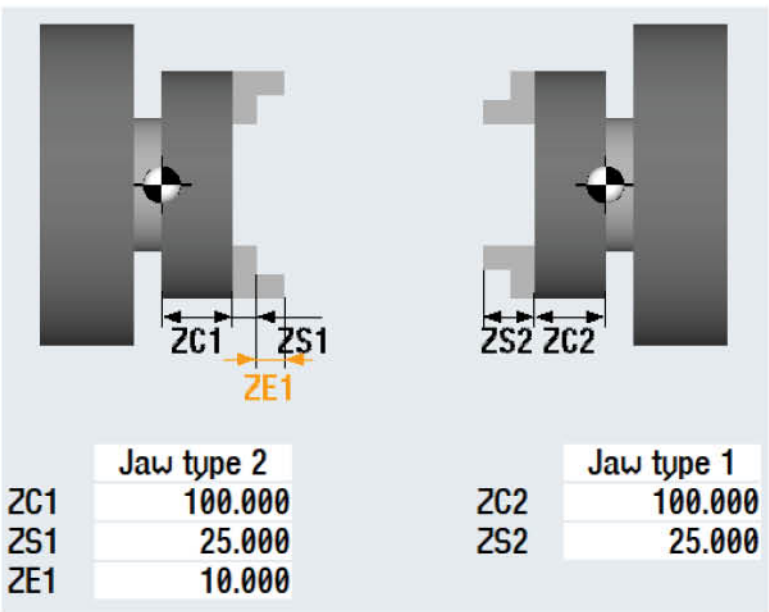


Figure 2.28: Dimension diagram of main chuck and opposite chuck (reverse chuck)

"OFFSET" adjustment of the real machine spindle chuck data

In order to program the counter spindle machining in the way of the real existing working area the spindle data have to be defined in the "Setting data" of the machine.

1.	Press the "OFFSET" button in the operating area "Parameter".
2.	Press the HSK 1.8 "Setting data" to find the "Spindle chuck data" menu.
3. 	Press the HSK 3 "Spindle chuck data" to open the dialog menu.

4.	The values entered here are included in the calculation for all further machining operations with the counter spindle in programGuide. 
5.	For the definition of the jaw types you have the possibility to change to “Jaw type 2” for main and counter spindle. You get a third input menu "ZE1" or “ZE2”. 
6. 	Press for example the “Program” or “Program manager” button to leave this menu.

If the workpiece is clamped in a collet, the value for the chuck dimension is the edge of the

collet.

The value for the jaw length (ZS2) is = 0

Figure 2.29: Collet and chuck

Counter spindle takeover example

Description:

The following workpiece is to be programmed with G code on a workpiece with a diameter of 100 mm and a finished length of 150 mm.

Following values for the machining are given:

Chuck dimension MS and CS = 100 mm

Jaw length MS and CS = 25 mm

Machining position of counter spindle = 1000 mm

Unclamping length of workpiece = 100 mm

Machining on the main spindle with G54 and LCT 80°:

- Face turning
- Outside longitudinal turning (roughing) Ø80 x 40 mm

Machining on the counter spindle with G55 and LCT 80°:

- Face turning
- Outside longitudinal turning (roughing) Ø90 x 50 mm

Work sequence diagram:

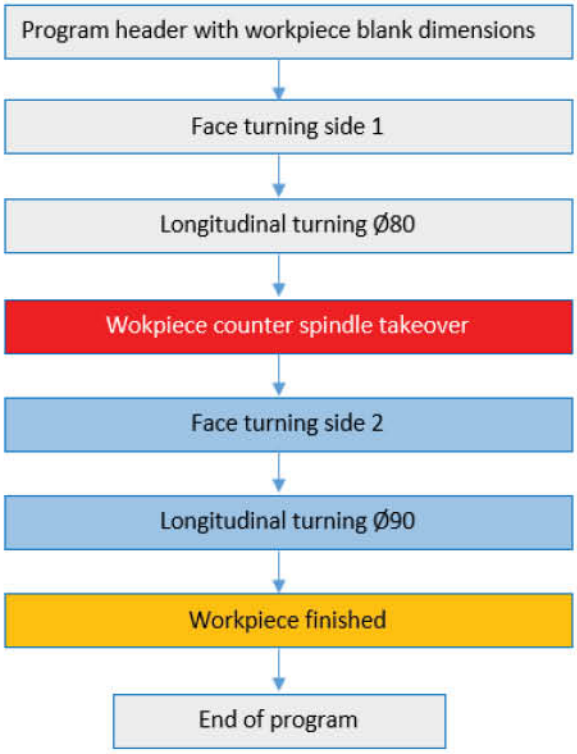
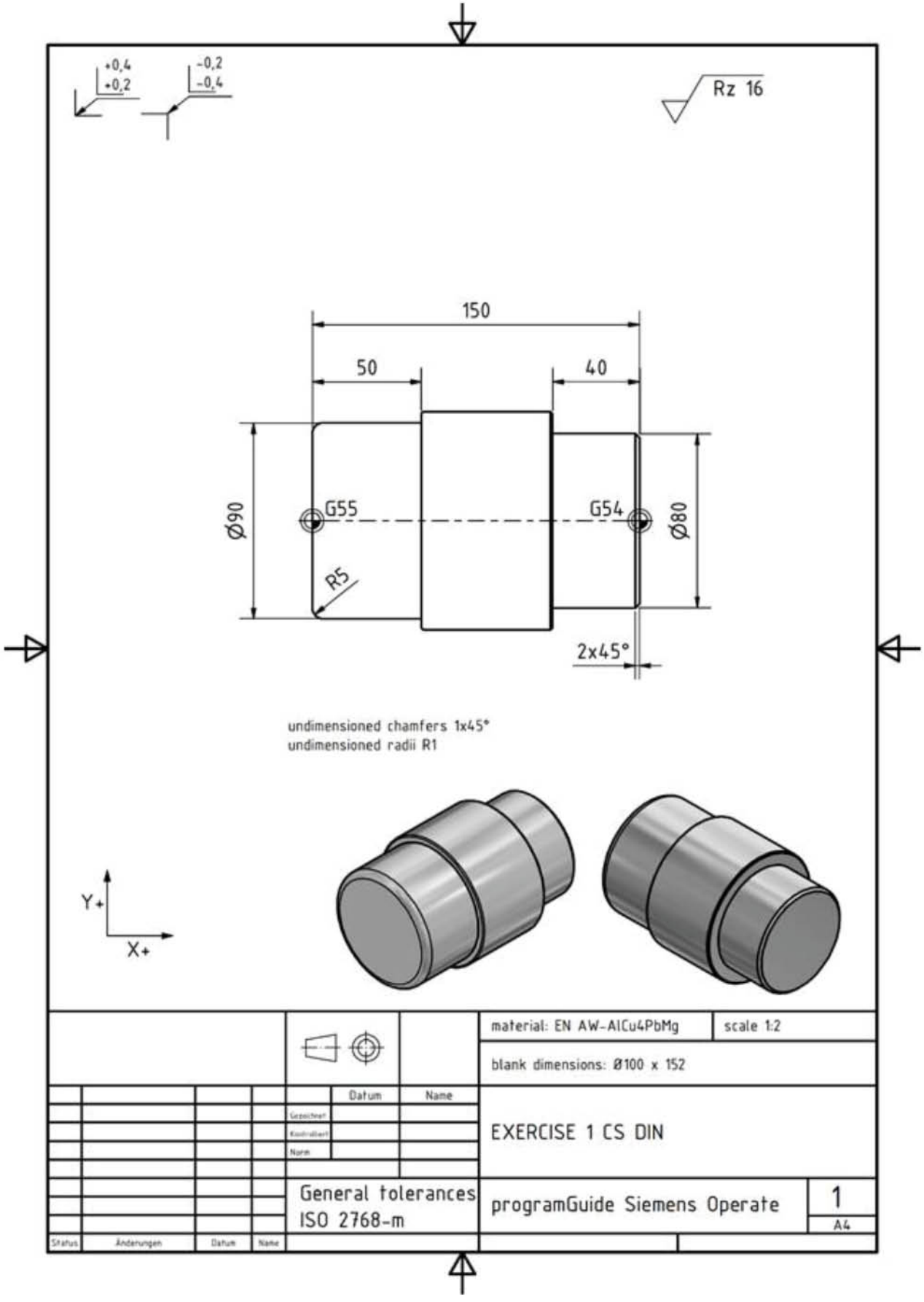


Chart 2.1: Product manufacturing process


+0,4
+0,2
-0,2
-0,4
Rz 16
150
50
40
Ø90
G55
G54
Ø80
R5
2x45°
undimensioned chamfers 1x45°
undimensioned radii R1
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:2
blank dimensions: Ø100 x 152
Datum
Name
EXERCISE 1 CS DIN
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name

The following commands are required for the programming of the counter spindle:

Command	Meaning
M132	Main spindle chuck open (manufacturer-specific)
M131	Counter spindle chuck open (manufacturer-specific)
M130	Counter spindle chuck close (manufacturer-specific)
M1=3	Main spindle = clockwise rotation direction
M2=3	Counter spindle = clockwise rotation direction
COUDEF	Definition of the coupling parameters
NOC	Immediate block change
DV	Set point coupling (default setting)
SPCON	Position control on
SPCOF	Position control off
COUPON	Synchronism on
COUPOF	Synchronism off
Z2 (or ZS2)	Designation of the counter spindle (manufacturer-specific)

A new program has to be programmed, the following blocks are created in the program:

1.	Create a new programGuide directory and main program called "CS_DIN_ISO".
2.	The "Edit" operation mode opens automatically.
3.	Press the VSK 2 "Build groups" to open the dialog menu.
4.	Input the first block name "START".
5.	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program:

6.	Create now following blocks in relation the work sequence: NC/WKS/CS_DIN_ISO/CS_DIN_ISO N1 ¶ N2 START N3 FACING SIDE 1 N4 LONGITUDINAL ROUGHING SIDE 1 N5 TRANSFER TO COUNTER SPINDLE N6 FACING SIDE 2 N7 LONGITUDINAL ROUGHING SIDE 2 N8 END

In order to continue the chuck dimensions are entered in the first step.

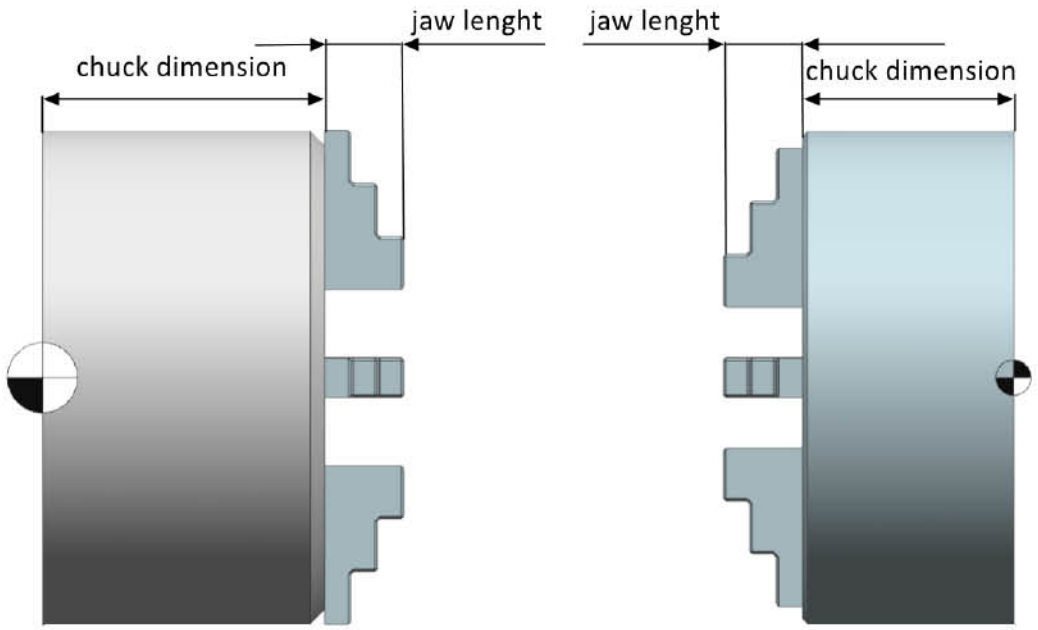
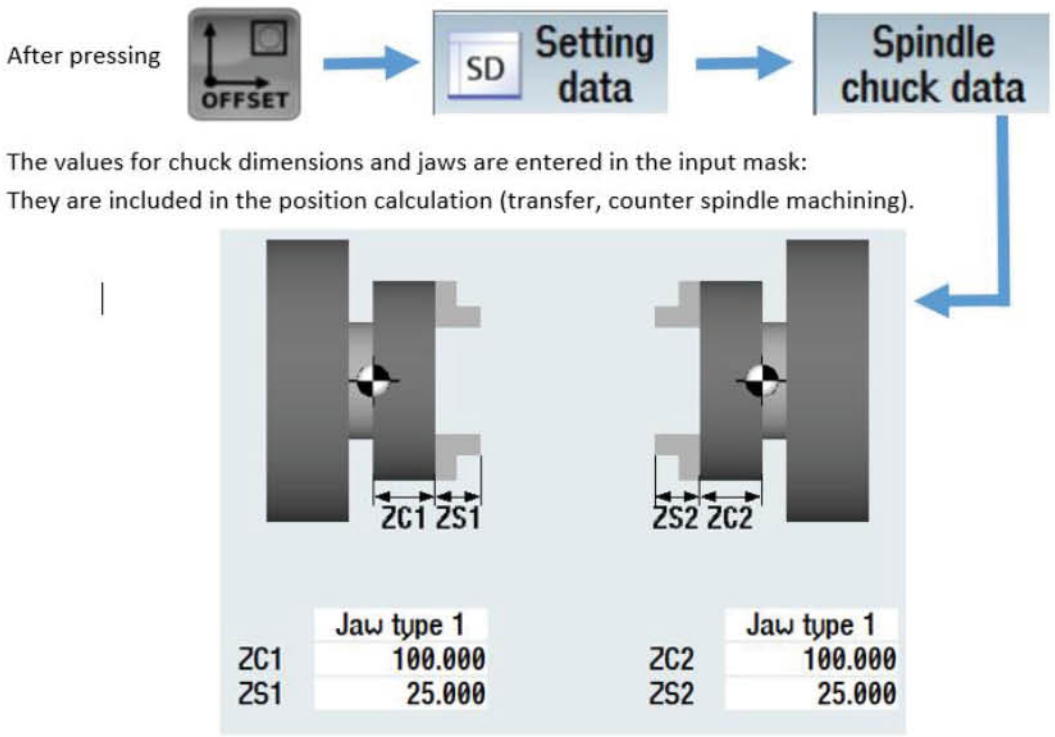


Figure 2.30: Dimension diagram of spindle chuck and opposite chuck (reverse chuck)



The following situation results from the workpiece, chuck and jaw length in combination with the machining position for the counter spindle:

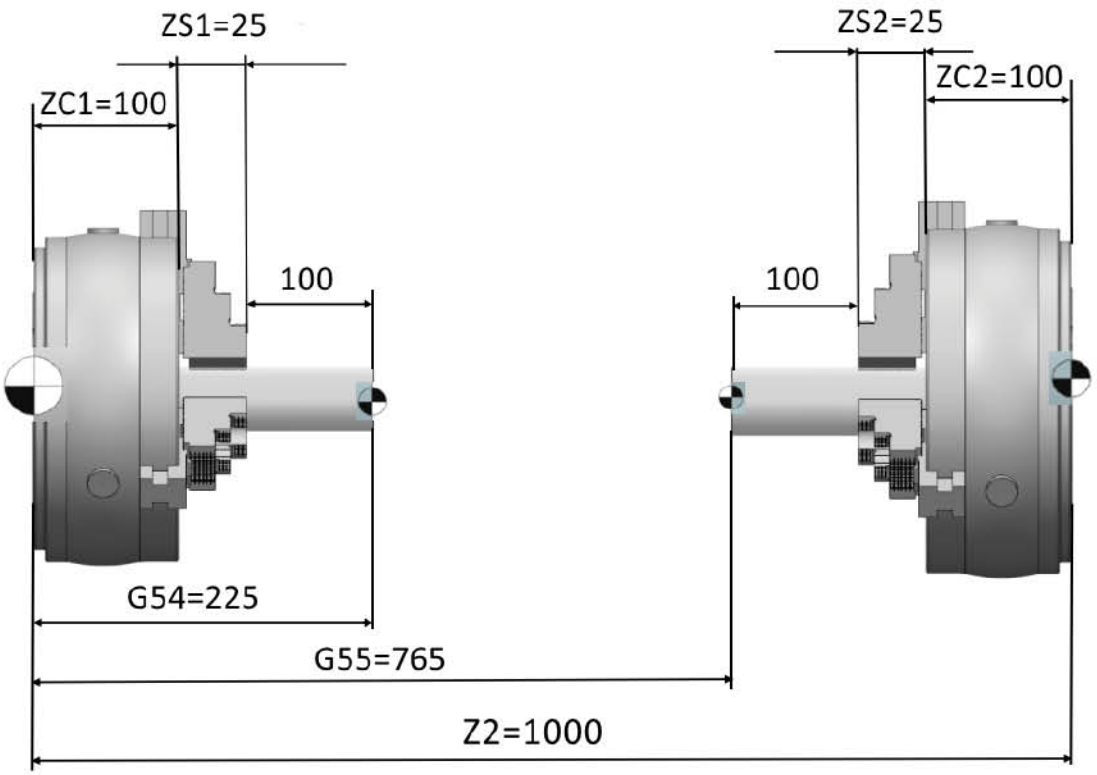
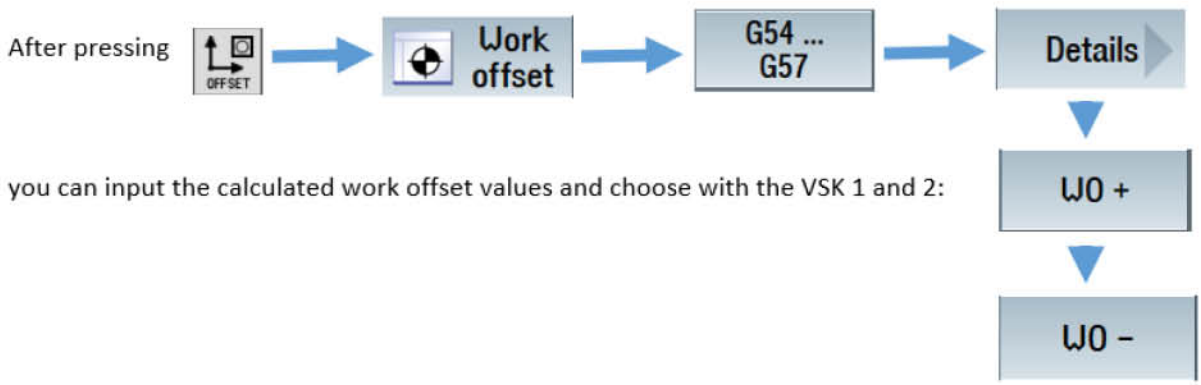


Figure 2.31: Example dimension of spindle chuck and opposite chuck (reverse chuck)

The calculated values for the work offsets for the Z axis in relation to the main spindle are entered in the work offset. They are not automatically entered in the simulation. The simulation and processing are therefore only displayed correctly then.

The mirroring with program Guide must be activated for the G55 work offset. The mirroring is not activated automatically as for ShopTurn programming.



Work offset G54:

SIEMENS SINUMERIK OPERATE 09/14/21 6:42 PM

Work offset – details: G54 [mm]

	Coarse	Fine			
X	0.000	0.000	0.000	1.000	☐
Y	0.000	0.000	0.000	1.000	☐
Z	225.000	0.000	0.000	1.000	☐
SP1	0.000	0.000		1.000	☐
SP3	0.000	0.000		1.000	☐
Z2	0.000	0.000		1.000	☐
SP2	0.000	0.000		1.000	☐

Work offset G55 with activated mirroring:

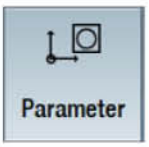
SIEMENS SINUMERIK OPERATE 09/15/21 7:58 AM

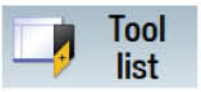
Work offset – details: G55 [mm]

	Coarse	Fine			
X	0.000	0.000	0.000	1.000	☐
Y	0.000	0.000	0.000	1.000	☐
Z	765.000	0.000	0.000	1.000	☑
SP1	0.000	0.000		1.000	☐
SP3	0.000	0.000		1.000	☐
Z2	0.000	0.000		1.000	☐
SP2	0.000	0.000		1.000	☐

Before we start the programming in the predefined blocks we have to check the tool offset.

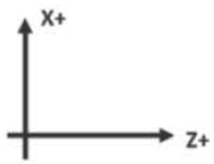
After pressing





You can input the following indicated tool offset values from this tool shape file:

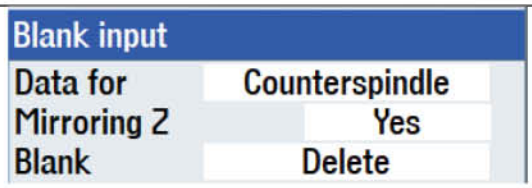
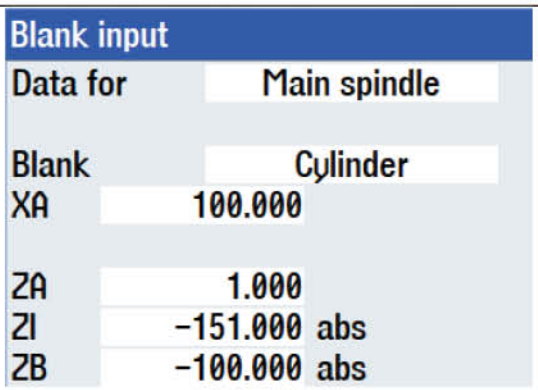
programGuide Siemens Operate VIETNAM 2021/2022		
Tool shape file	CNC Turning	
work piece: EXERCISE 1 CS DIN	blank material: EN-AW AlCu4PbMg	machine configuration:
drawing: 1(1)	blank dimensions: Ø100 x 152	X Z C Y with CS

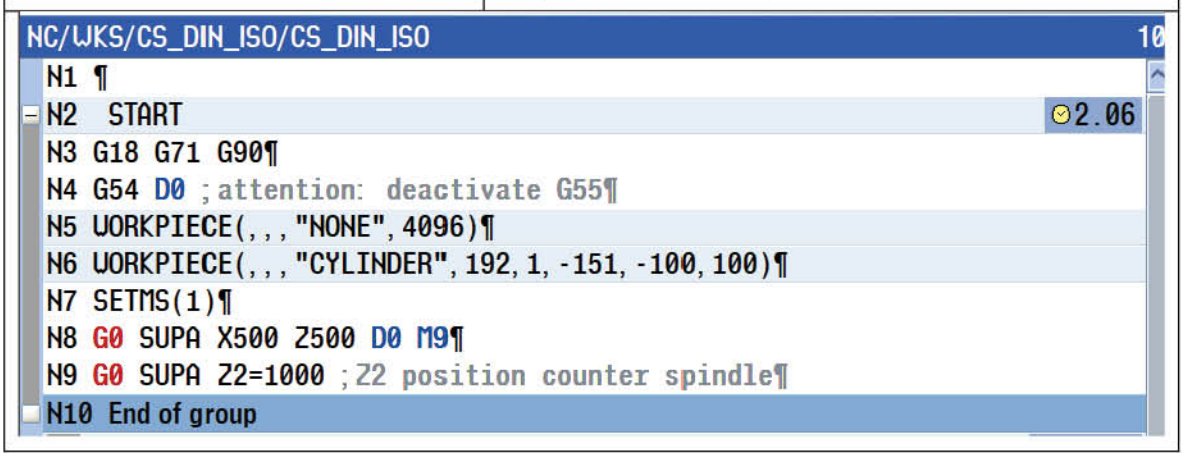
	Tools for outer machining							
	Technological data							
Tool name in OFFSET	T="ROUGH _A_80"	T="ROUG_ A_80_CS"	T="ROUG_ A_55"	T="ROUG_ A__55_CS"	T="FIN_A_ 35"	T="ROUGH _A_35_CS"	T="THRED_ 1,5"	T="FIN_A_ 35_CS"
Cutting radius (mm)	0,8	0,8	0,4	0,4	0,4	0,4		0,4
Cutting speed (m/min)	250	250	250	250	300	250	120	300
Cutting depth a_p = max.	1 / 2,5	1 / 2,5	2	2	0,5	2		0,5
Cutting material	SC	SC	SC	SC	SC	SC	SC	SC
Feed rate per rev./pitch	0,3 / 0,12	0,3 / 0,12	0,2 / 0,12	0,2 / 0,12	0,1	0,2 / 0,1	F = P	0,1
Angle/pitch/diameter Ds	80°	80°	55°	55°	35°	35°	P1,5	35°
								
		G18 CS		G18 CS		G18 CS		G18 CS

Note: The following description of program input describes the most important and new commands for programming and their meaning.

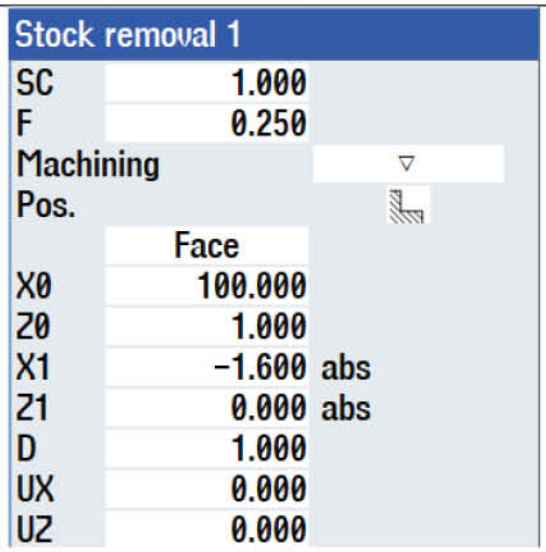
It´s up to you to complete the description to apply your knowledge.

The start commands are programmed in the "START" block.

Command	Meaning
N3 G18 G71 G90	
N4 G54 D0	
N5 WORKPIECE(,,,"NONE",4096)	
N6 WORKPIECE(,,,"CYLINDER",192,2,-152,-100,100)	
N7 SETMS(1)	Set master spindle to MS
N8 G0 SUPA X500 Z500 D0 M9	Tool changing point
N9 G0 SUPA Z2=1000	Z2 position counter spindle



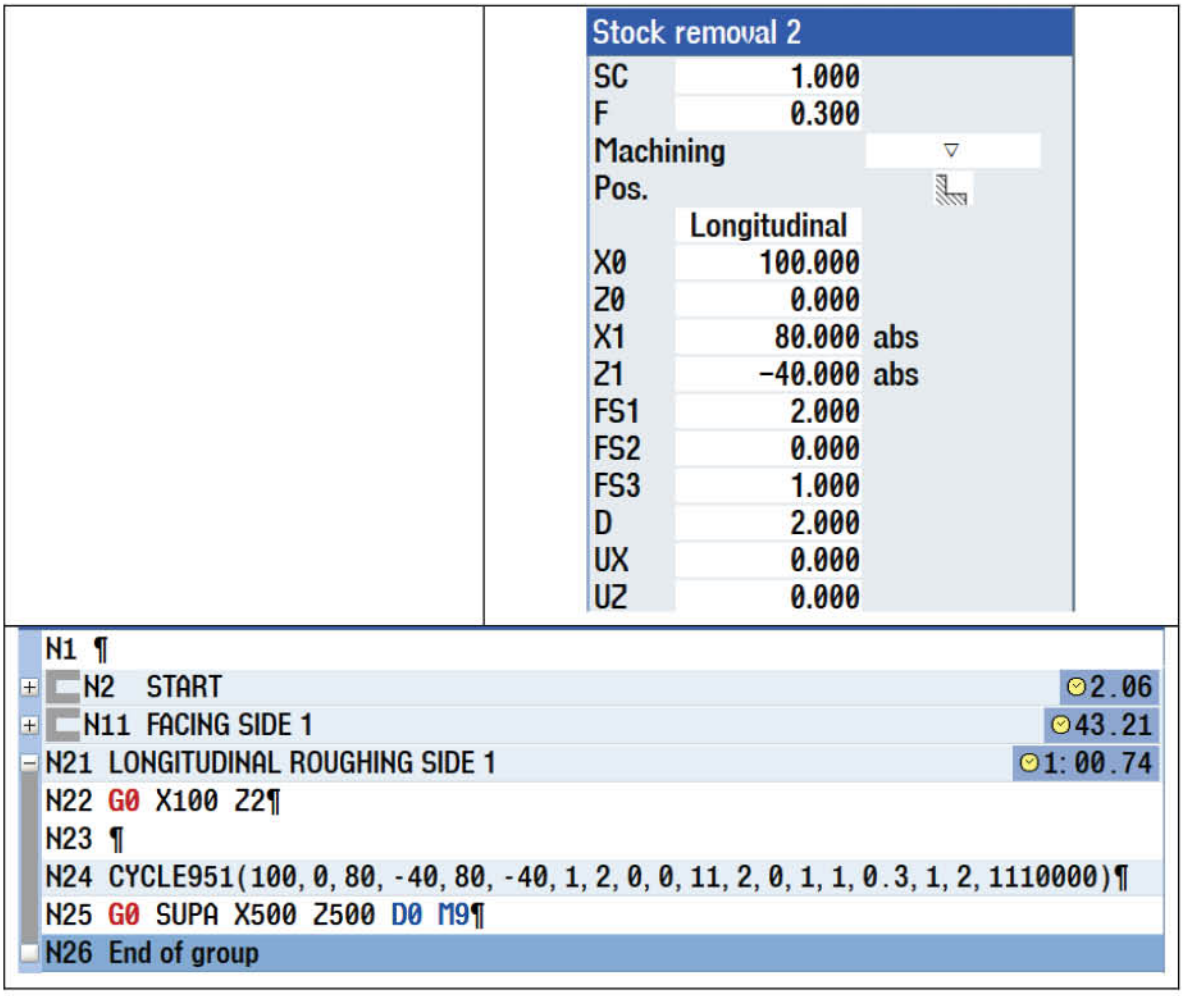
The face machining is programmed in the "FACING SIDE 1" block.

Command	Meaning
N12 DIAMON	
N13 LIMS=3000	
N14 T="ROUGH_A_80"	
N15 G96 S250 M4	
N16 G0 X105 Z2 M8	
N18 CYCLE951(100,1,-1.6,0,-1.6,0,1,1,0,0,12,0,0,0,1,0.25,0,2,1110000)	


```
N11 FACING SIDE 1                                                    43.21
N12 DIAMON¶
N13 LIMS=3000¶
N14 T="ROUGH_A_80"¶
N15 G96 S250 M4¶
N16 G0 X105 Z2 M8¶
N17 ¶
N18 CYCLE951(100, 2, -1.6, 0, -1.6, 0, 1, 1, 0, 0, 12, 0, 0, 0, 1, 0.25, 0, 2, 1110000)¶
N19 ¶
N20 End of group
```

The longitudinal machining is programmed in the “LONGITUDINAL ROUGHING SIDE 1” block.

Command	Meaning
N22 G0 X100 Z2 N23	
N24 CYCLE951(100,0,80,-40,80,-40,1,2,0,0,11,2,0,1,1,0.3,1,2,1110000)	

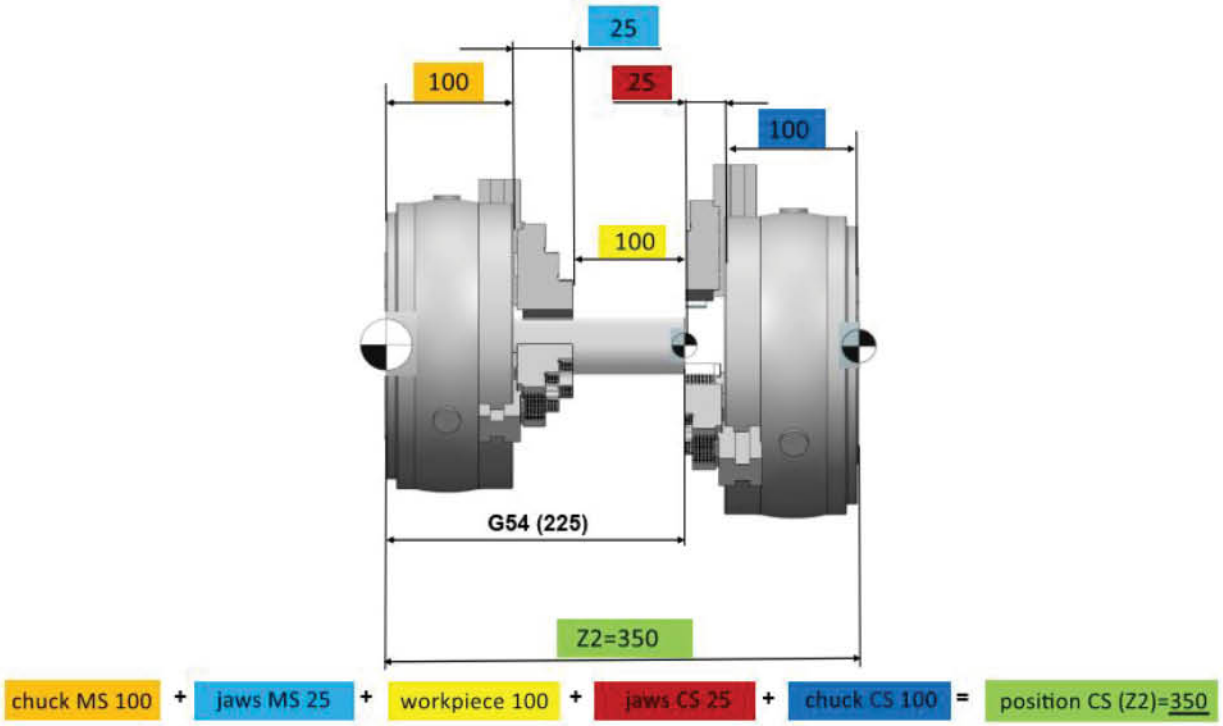

Stock removal 2
SC 1.000
F 0.300
Machining
Pos.
Longitudinal
X0 100.000
Z0 0.000
X1 80.000 abs
Z1 -40.000 abs
FS1 2.000
FS2 0.000
FS3 1.000
D 2.000
UX 0.000
UZ 0.000
N1 ¶
N2 START 2.06
N11 FACING SIDE 1 43.21
N21 LONGITUDINAL ROUGHING SIDE 1 1:00.74
N22 G0 X100 Z2¶
N23 ¶
N24 CYCLE951(100,0,80,-40,80,-40,1,2,0,0,11,2,0,1,1,0.3,1,2,1110000)¶
N25 G0 SUPA X500 Z500 D0 M9¶
N26 End of group

The transfer of the workpiece to the counter spindle is programmed in the "TRANSFER TO COUNTER SPINDLE" block.

Command	Meaning
N28 M231	Counter spindle chuck OPEN
N29 G95 S1=500 M1=4	Technology MS
N30 S2=500 M2=3	Technology CS
N31 G4 F3	Dwell time
N32 COUPDEF(S2,S1,1,-1,"NOC","DV")	Programming of the coupling parameters, S2=following spindle (CS), S1=leading spindle(MS)
N33 SPCON(2,1,0)	Activation of the position control for both spindles
N34 COUPON(S2,S1,0)	Activation of the synchronism
N35 G0 Z2=350	Transfer preposition (graphic 1 on the next page)
N36 G94 G1 Z2=310 F500	Transfer position (graphic 2 on the next page)
N37 M230	Counter spindle chuck CLOSE
N38 G4 F1	Dwell time
N39 M132	Main spindle chuck OPEN
N40 COUPOF(S2,S1)	Deactivation of the position control for both spindle
N41 SPCOF(2,1,0)	Deactivation of the synchronism
N42 G0 Z2=1000	Travel to machining position of the counter spindle
N43 M1=5	MS stop

```
N27 TRANSFER TO COUNTER SPINDLE                    17.40
N28 M231¶
N29 G95 S1=500 M1=4¶
N30 S2=500 M2=3¶
N31 G4 F3¶
N32 COUPDEF(S2, S1, 1, -1, "NOC", "DV")¶
N33 SPCON(2, 1, 0)¶
N34 COUPON(S2, S1, 0)¶
N35 G0 Z2=350¶
N36 G94 G1 Z2=310 F500¶
N37 M230¶
N38 G4 F1¶
N39 M132¶
N40 COUPOF(S2, S1)¶
N41 SPCOF(2, 1, 0)¶
N42 G0 Z2=1000¶
N43 M1=5¶
N44 End of group
```

Graphic 1 “N35 G0 Z2=350”:



Graphic 2 “N36 G94 G1 Z2=310 F500”:

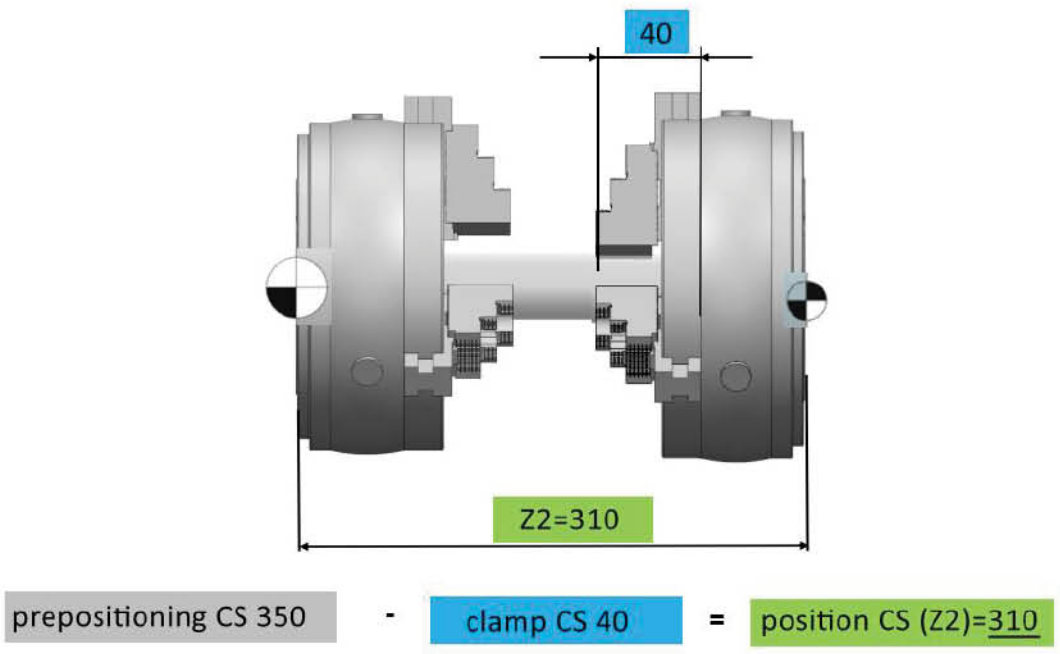
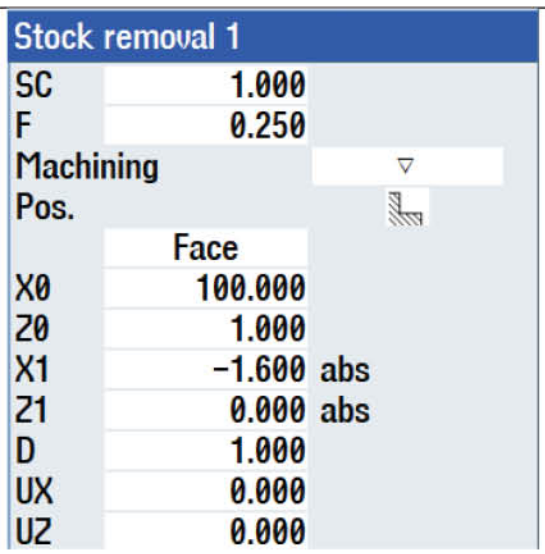


Figure 2.32: Example spindle chuck size and opposite chuck (reverse chuck)

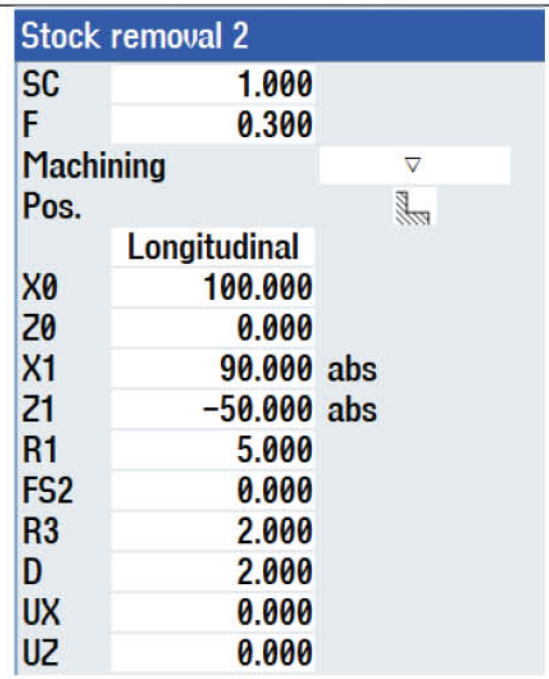
The second face machining is programmed in the "FACING SIDE 2" block.

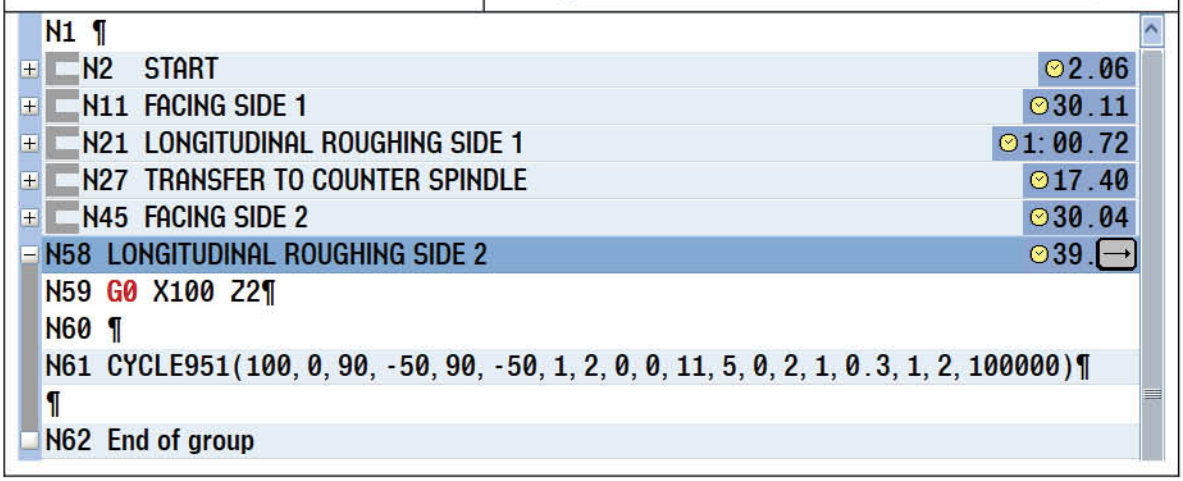
Command	Meaning
N46 G0 SUPA X500 Z500 D0 M9	
N47 T="ROUG_A_80_CS"	
N48 SETMS(2)	
N49 G18	
N50 G55 G90	
N51 DIAMON	
N52 LIMS=3000	
N53 G96 S250 M3	
N54 G0 X102 Z2	
N55 N56 CYCLE951(100,2,-1.6,0,-1.6,0,1,1,0,0,12,0,0,0,1,0.25,0,2,1110000)	


```
N45 FACING SIDE 2                                                    30.04
N46 G0 SUPA X500 Z500 D0 M9¶
N47 T="ROUG_A_80_CS"¶
N48 SETMS(2)¶
N49 G18¶
N50 G55 G90¶
N51 DIAMON¶
N52 LIMS=3000¶
N53 G96 S250 M3¶
N54 G0 X102 Z2¶
N55 ¶
N56 CYCLE951(100, 1, -1.6, 0, -1.6, 0, 1, 1, 0, 0, 12, 0, 0, 0, 1, 0.25, 0, 2, 1110000)¶
N57 End of group
```

The second longitudinal machining is programmed in the “LONGITUDINAL ROUGHING SIDE 2” block.

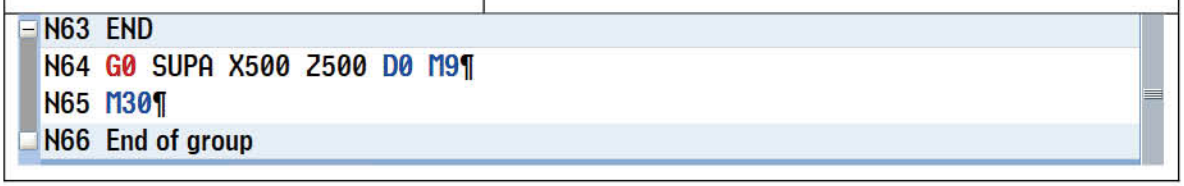
Command	Meaning
N59 G0 X100 Z2	
N60 N61 CYCLE951(100,0,90,-50,90,-50,1,2,0,0,11,5,0,2,1,0.3,1,2,100000)	





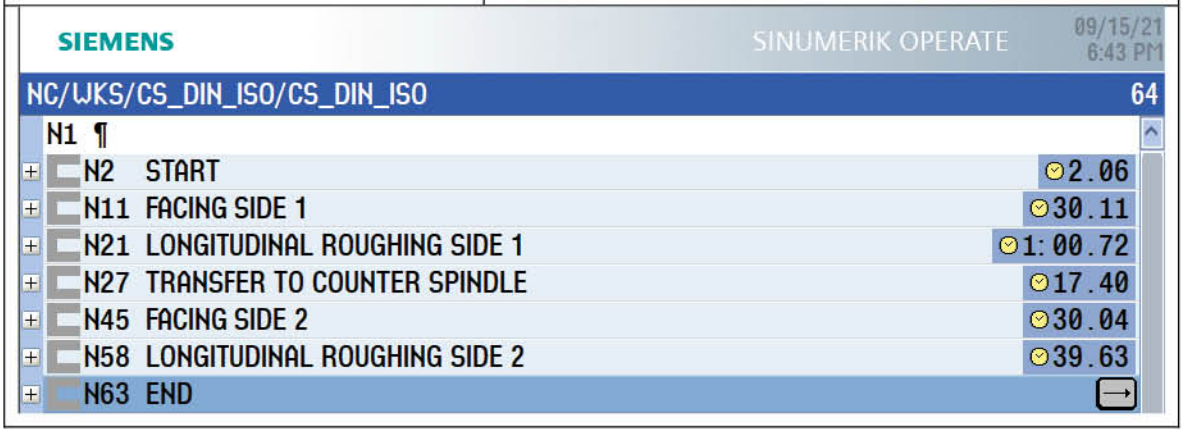
The end of the program is programmed in the "END" block:

Command	Meaning
N64 G0 SUPA X500 Z500 D0 M9	
N65 M30	

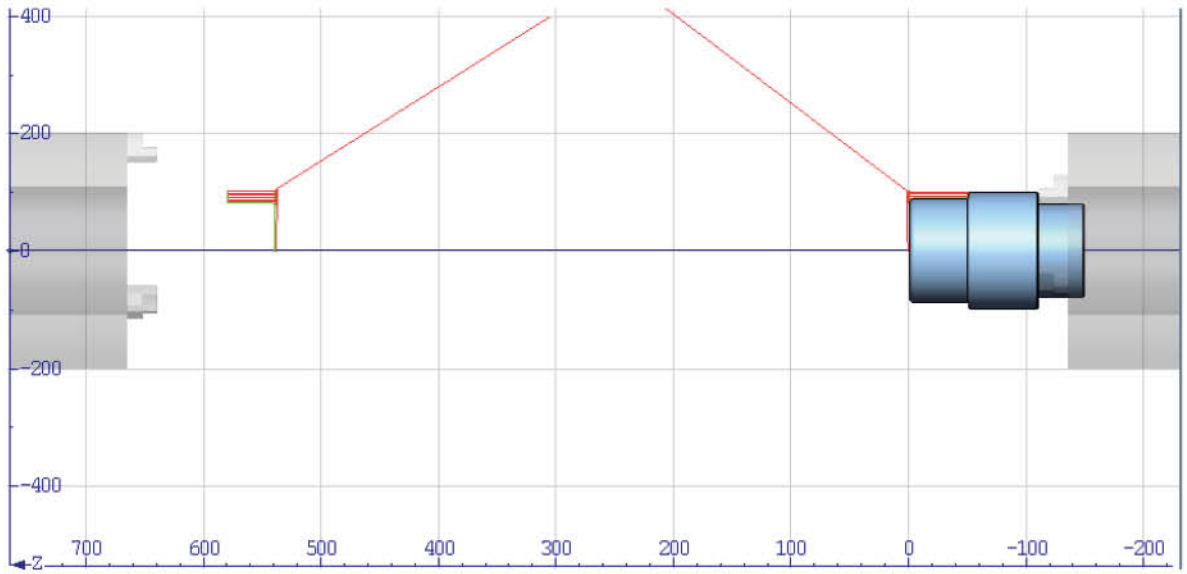


The program has been completed.

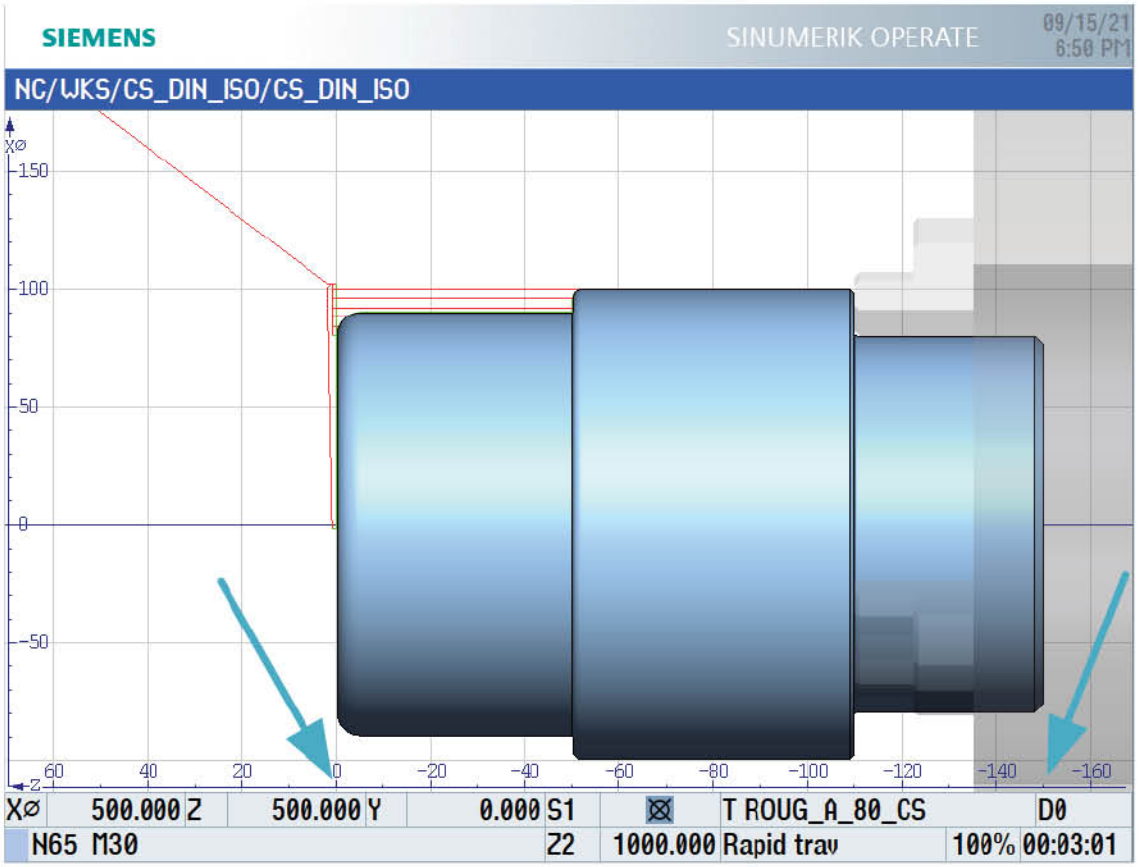
Command	Meaning



Check the dimensions with the help of the “Measuring scale” in the “Side view”:



Zoom around the counter spindle for measuring the workpiece length:



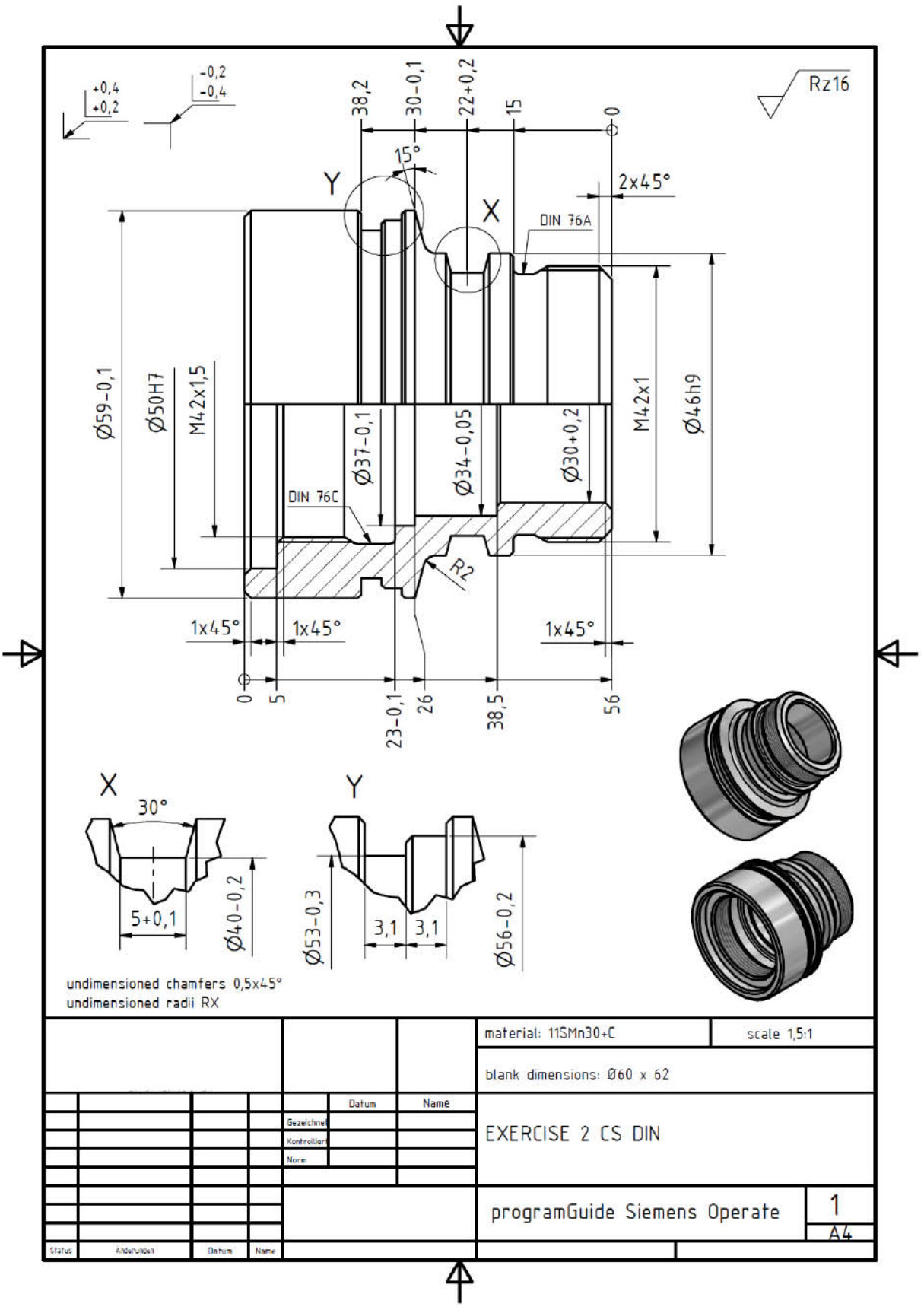
3.8. Plan and realise operational work order of the industry

3.8.1. Using programGUIDE with the cycles and G-codes for advanced turning application 1

a. Objective:

- Complete work plan based on the given template
- Program to machine the relevant application using the appropriate cycles and G-codes.

b. Technical drawing



c. The following tools and technology data are required for programming

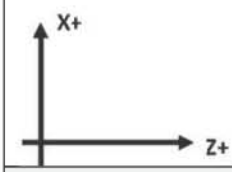
programGuide Siemens Operate

VIETNAM 2021/2022

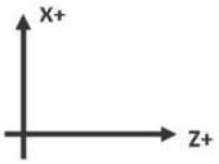
Note: The tool names are determined by the programmer and stored in an archive file at the end of programming

Tool shape file	CNC Turning

Work piece: EXERCISE2 CS	Blank material: 11SMn30+C	Machine configuration:
Drawing: 1(1)	Blank dimensions: Ø60 x 62	X Y Z C

	Tools for turning machining							
	Technological data							
Tool - Nr.	T1	T2	T3	T4	T5	T6	T7	T8
Cutting radius (mm)	0,8		0,4	0,4	0,4	0,4	0,2	
Cutting speed (m/min)	200	120	200	180	250	180	125	80
Cutting depth ap = max.	1 / 2,5		2,5	2	1	0,5	2	
Cutting material	SC		SC	SC	SC	SC	SC	SC
Feed rate per rev./ pitch	0,3 / 0,1	0,15	0,25 / 0,12	0,2 / 0,12	0,2 / 0,1	0,1 / 0,05	0,1	
Angle/pitch/diameter Ds	80°	Ø28	55°	55° - Ø16	35°	35° - Ø16	2,5 mm	Ø 16 – P1.5
								
			G18 CS				G18 CS	

	Tools for turning machining							
	Technological data							
Tool - Nr.	T9	T10	T11	T12	T13	T14		
Cutting radius (mm)		0,4	0,8	0,2	0,4	0,4		
Cutting speed (m/min)	100	150	200	125	250	180		
Cutting depth ap = max.		2	1 / 2,5	2	1	2		
Cutting material	SC	SC	SC	SC	SC	SC		

Feed rate per rev./ pitch	F = P	0,2 / 0,1	0,3 / 0,1	0,1	0,2 / 0,1	0,2 / 0,1		
Angle/pitch/diameter Ds	P1	80°	80°	2,5 mm	35°	55° - Ø16		
								
	G18 CS		G18 CS		G18 CS	G18 CS		

d. Template of work plan

PROJECT NAME:	
Surname:	
Given name:	Date:

Sinumerik Operate V4.8	Blank material:	CNC turning: X, Z
programGUIDE	Blank dimesions:	CNC milling: X, Y, Z

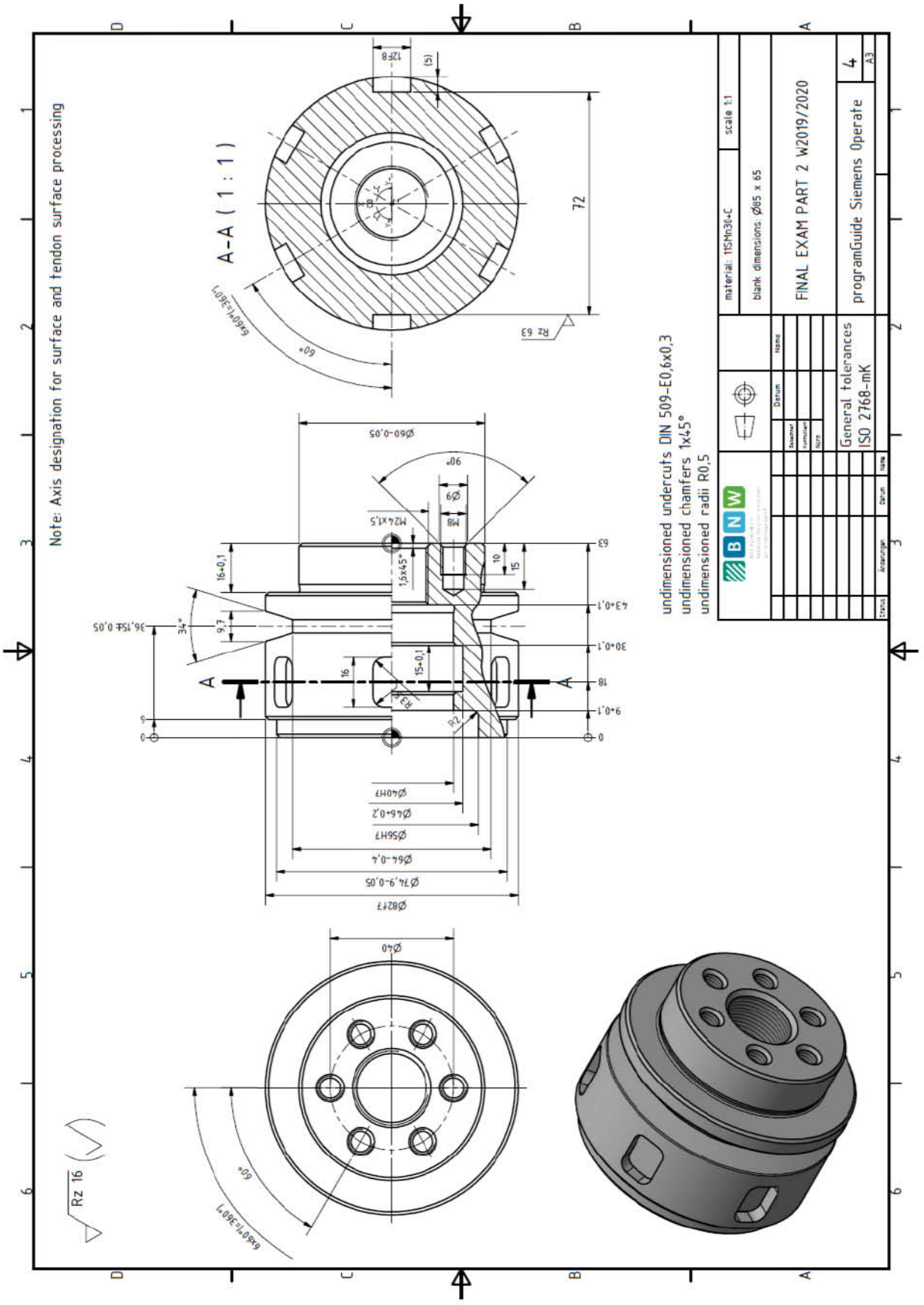
No.	**Work sequence**	**Tool**	**Remarks**

3.9.3. Using programGUIDE with the cycles and G-codes for advanced turning application 2

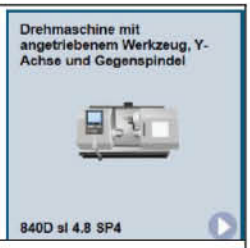
a. Objective:

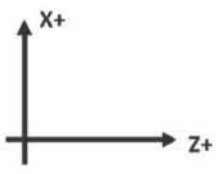
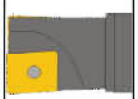
- Complete work plan (refer template at page 252)
- Program to machine the relevant application using the appropriate cycles and G-codes.

b. Technical drawing

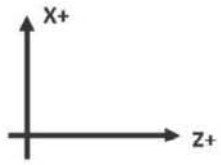
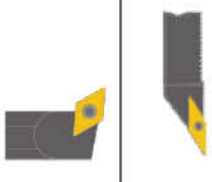


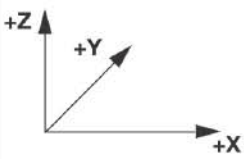
c. The following tools and technology data are required for programming

programGuide Siemens Operate VIETNAM 2021/2022 Note: The tool names are determined by the programmer and stored in an archive file at the end of programming.		
Tool shape file	CNC Turning	
Work piece: AP2 W2019-20 CS DIN DIN	Blank material: 11SMn30+C	Machine configuration:
Drawing: 1(1)	Blank dimensions: Ø85 x 65	X Y Z C

	Tools for turning machining							
	Technological data							
Tool - Nr.	T1	T2	T3	T4	T5	T6	T7	T8
Cutting radius (mm)	0,8		0,4	0,4	0,4	0,4	0,2	0,2
Cutting speed (m/min)	200	120	200	180	240	200	140	120
Cutting depth ap = max.	1 / 2,5		2,5	2	1	0,5	2,5	2,5
Cutting material	SC		SC	SC	SC	SC	SC	SC
Feed rate per rev./ pitch	0,3 / 0,1	0,1	0,3 / 0,1	0,2 / 0,1	0,2 / 0,1	0,1 / 0,05	0,15 / 0,05	0,15 / 0,05
Angle/pitch/diameter Ds	80°	Ø20	55°	55° - Ø12	35°	35° - Ø16	3 mm	Ø25 – 4mm
								
			G18 CS					

	Tools for turning machining							
	Technological data							
Tool - Nr.	T9	T10	T11	T12				
Cutting radius (mm)	0,8	0,4	0,4					
Cutting speed (m/min)	200	180	240	80				

Cutting depth ap = max	1 / 2,5	2	2					
Cutting material	SC	SC	SC	SC				
Feed rate per rev./ pitch	0,3 / 0,1	0,2 / 0,1	0,2 / 0,1					
Angle/pitch/diameter Ds	80°	55° - Ø16	35°	Ø 16 – P1,5				
								
	G18 CS	G18 CS	G18 CS	G18 CS				

	Driven tools for drilling and milling machining							
	Technological data							
Tool - Nr.	T13	T14	T15	T16				T20
Tool Ø (mm)	6	10 x 90°	6,8	8				8 x 90°
Cutting speed (m/min)	150 / 175	30	30	20				105
Cutting depth a_p = max.	3	5						3
Cutting material	SC	HSS	HSS	HSS				SC
Feed tooth fz / rev. (mm)	0,02 / 0,01	0,01	0,01	1,25				0,025
Number of teeth N	3							4
								
	✓							
	G19	G17	G17	G17				G17

UNIT 3: ADVANCED DIN/ISO PROGRAMMING – CNC MILLING 3+2 AXES – SWIVELLING IN ACCORDANCE WITH DIN/ISO

1. **Introduction:** As the third unit of the module, the trainees will be familiar with programming method using programGUIDE to solve advanced applications regarding CNC milling with 3+2 axes.
2. **Objective:** By the end of this training unit, the trainees will be able to

- Explain function of swivel cycle (CYCLE800) and the advantages of this cycle
- Set and select control configuration in Sinutrain software for AC and BC kinematics for machining on 5-axis CNC milling machines
- Analyze detail drawings and customer's technical requirements
- Complete the work plan for each mechanical part in the order
- Calculate and select machining parameters for required mechanical parts
- Use mechanical manuals to look up relevant technical information
- Use swivel cycle (CYCLE800), G-code instructions and suitable cycles to program machining for advanced milling applications (3+2 axes)
- Optimize specified programming files.

3. **Content:**

3.1. Simplifying programming for 3+2 axes with programGUIDE

3.1.1. General introduction

The Swivel CYCLE800 is a static plane transformation, which allows you on a 5-axis machine (e.g. swivel head or swivel table) to define a rotated working plane in space.

In this work plane you can than program a 2D or 3D machining operation.

This is also referred as 3+2 axis positioning, by means of the 2 rotary axes involved in the transformation being only positioned while the linear axes move during machining.

By calling the appropriate NC functions, the cycle converts the active workpiece zero (e.g. G54) and tool offsets to refer to the rotated surface, taking into account the kinematics chain offsets on the machine, and positions the physical rotary axes (optional) normal to the programmed working plane.

3.1.2. Programming CYCLE800

This cycle is also a subprogram that accesses data which was commissioned so that the programming and application are optimized and simplified for operators and programmers.

This unit shows examples for working with "CYCLE800".

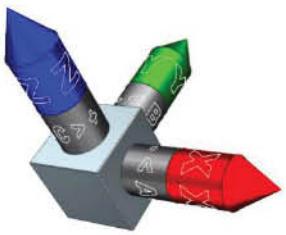
You can use swivel heads or swivel tables to set up and machine oblique planes.

Swiveling is possible in JOG mode as well as in AUTOMATIC mode. Swivel operation parameter assignment and programming are facilitated by the clearly laid out graphics.

You can either program all the swivel axes directly on the machine (A, B, C) or you can simply specify the rotations around the geometry axes (X, Y, Z) of the workpiece coordinate system as described in the relevant workpiece drawing. The rotation of the workpiece coordinate system in the program is then automatically converted to a rotation of the relevant swivel axis of the machine during machining.

The swivel axes are always rotated in such a way that the machining plane is perpendicular to the tool axis for machining. The machining plane is then fixed during the machining.

When the axes are swiveled, the active zero points and tool offsets are automatically converted for the swiveled state, resulting in a new coordinate system.



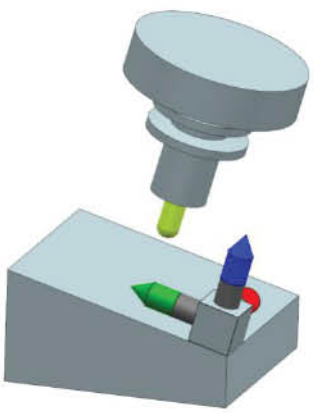
The resulting linear and rotary axes are shown in the right-hand graphic using a coordinate cube.

The machining plane is then fixed during the machining.

All this is implemented via "CYCLE800".

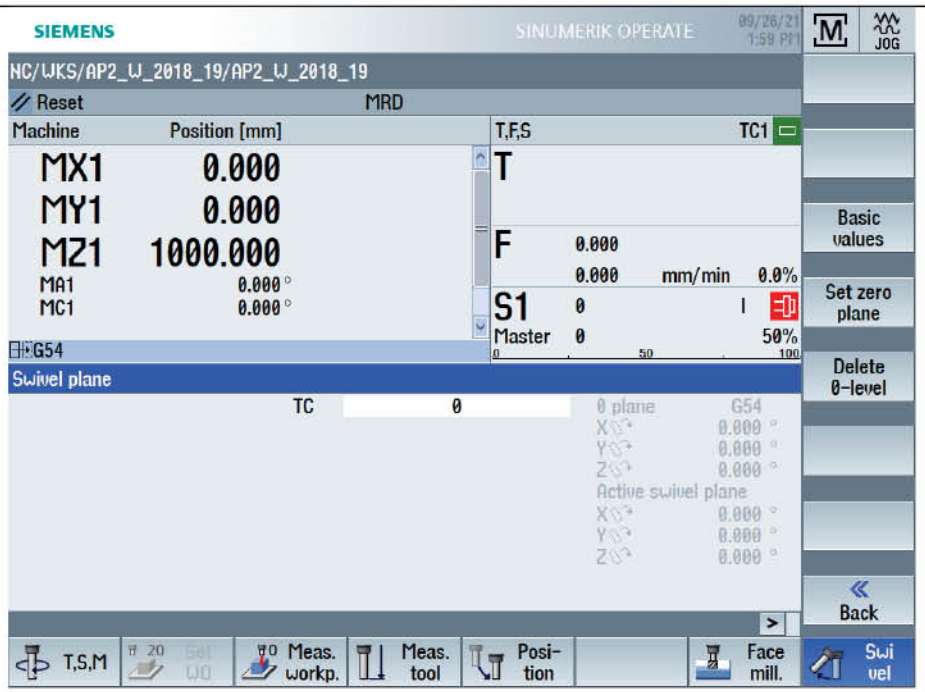
The term "CYCLE800" does not appear in ShopMill, it´s used with programGuide.

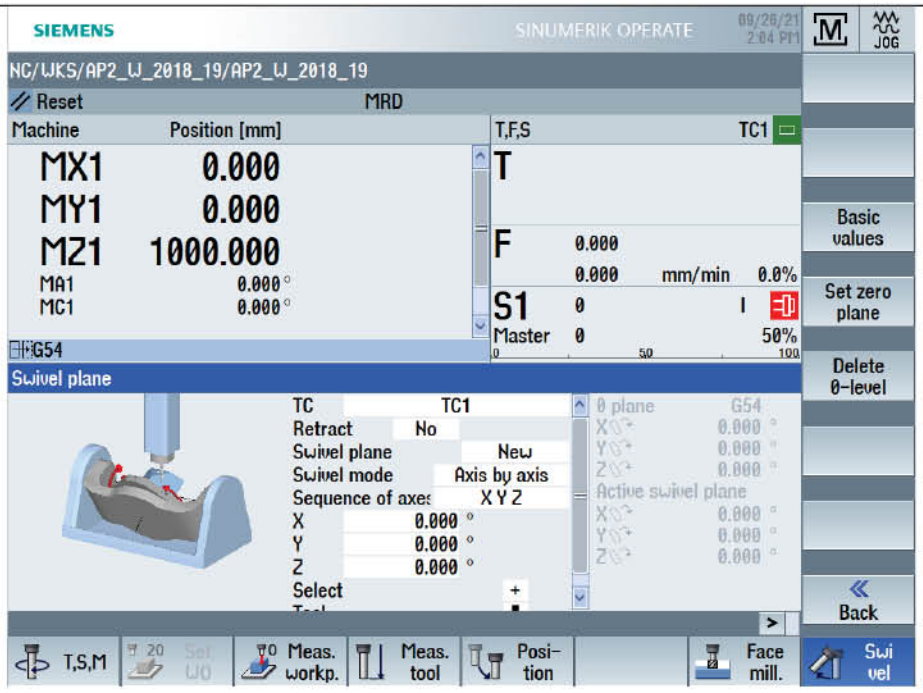
However, because this cycle permits swiveling, it is used completely for this purpose in all manuals, so for ShopMill and as well programGuide.



3.2. Tool carrier of a 5-axis milling machine

Open the new G code programGuide “DRILL_DIN_G17”:

1. 	Pressing “JOG” and HSK1.8 “Swivel” opens the input screen for swiveling in JOG. If no TC (Tool Carrier) that contains the machine kinematics for swiveling is selected, this field remains empty.
“TC0”	
2.	Only when a TC is selected does the input screen with the appropriate kinematics open.



Note:

The TC, the “Tool Carrier”, beginning with the name through to the parameterization, is normally commissioned by the manufacturer.

This is not the basis this module, but rather of the commissioning (setup by manufacturer).

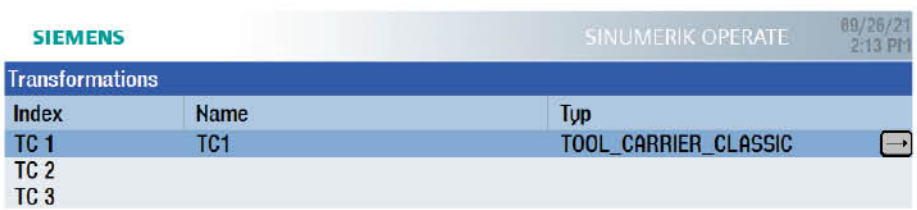
To simplify understanding, the function of a Tool Carrier as of version 4.8 is described in detail.

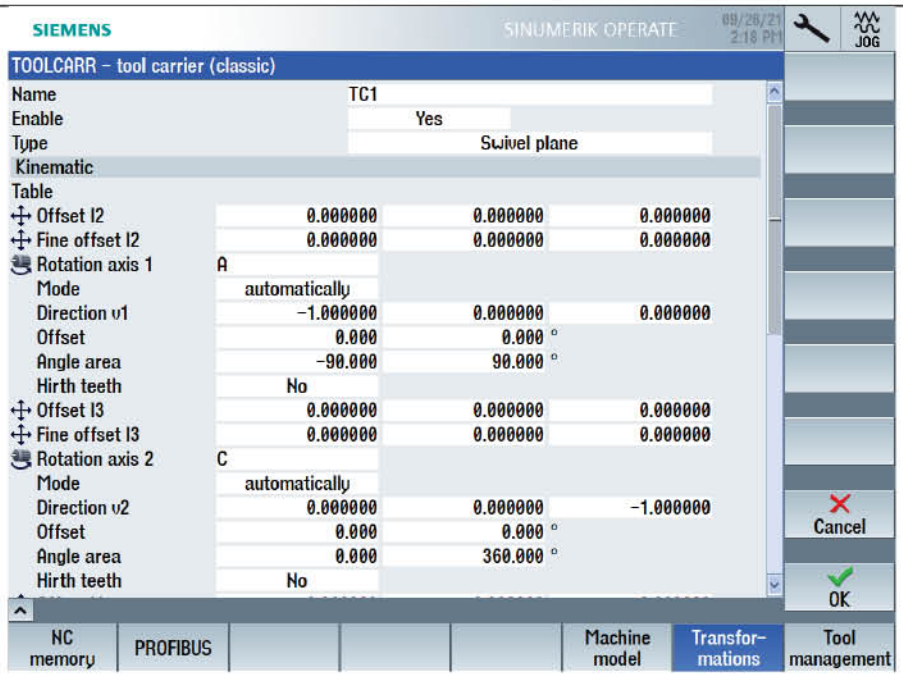
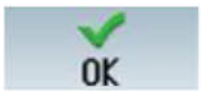
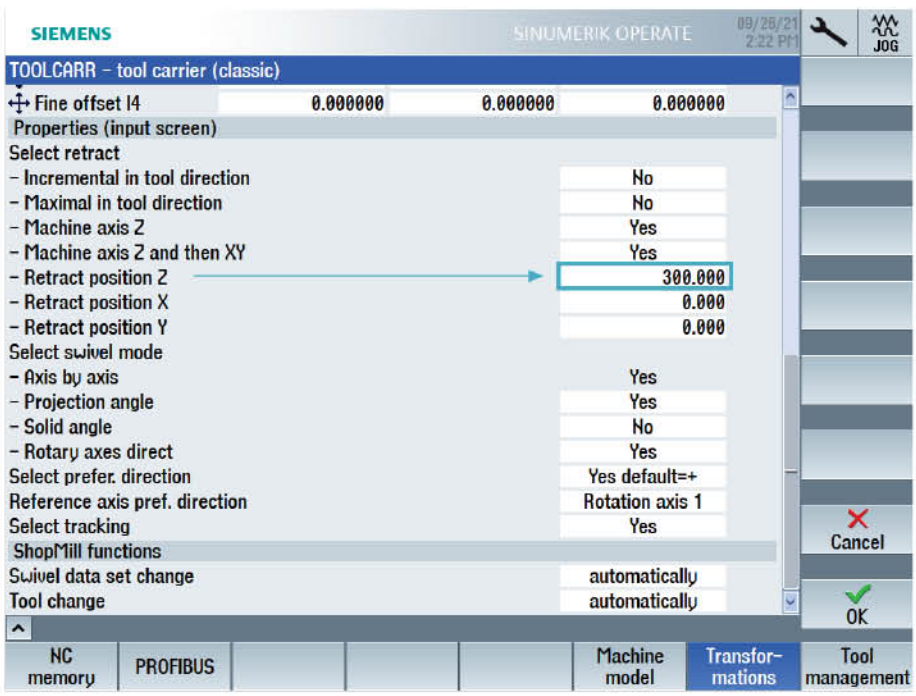
3.

Pressing “Menu select”, HSK1.2 “NC” and HSK2.7 “Transformation” opens the Tool Carrier. The name of the TC ("TABLE" or “TC1”) is arbitrary.

If necessary, several Tool Carriers can also be created, for example, when a replaceable head is used for machining.

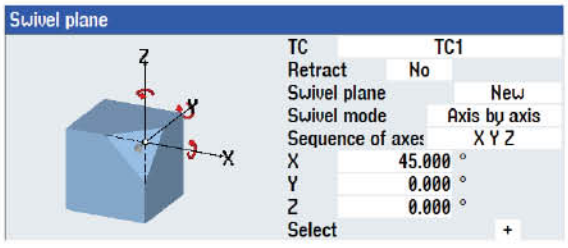
Default Tool Carriers of SinuTrain 4.8 Ed.2:

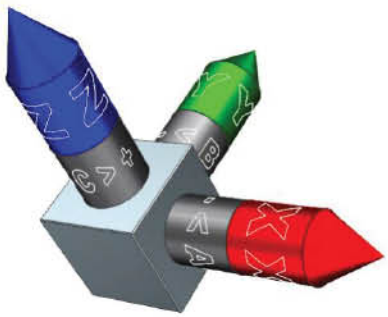


4.	Pressing “Open” to see the “TC1” parameter list. The complete mathematics of the machine axes are stored in the Tool Carrier “TC1”.
“TC1”	
5. 	Here change the valus “retract position Z” to 300 mm and confirm with VSK1.8 “OK”. 
	In addition to the values for the axis measurement, machine-related strategies are also specified in the Tool Carrier “TC1”. Such strategies are normally made only by the machine manufacturer.

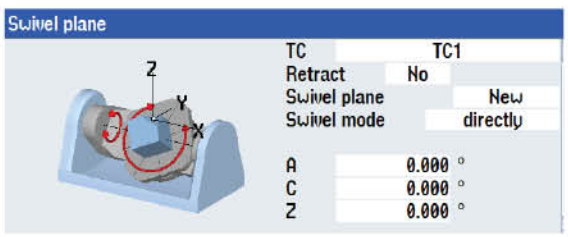
The axes can be swiveled in two ways.

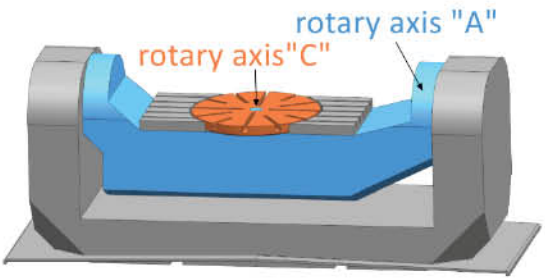
Specify the X, Y or Z axes in various sequences by entering a degree:



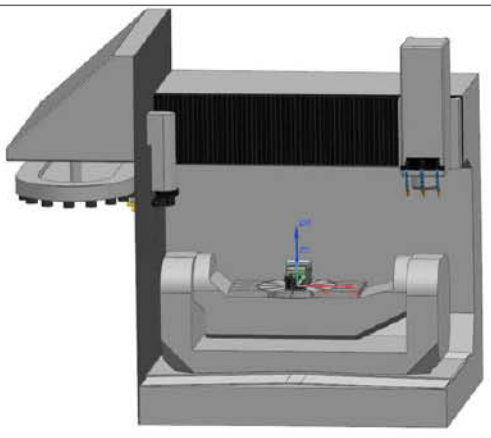


or by the direct selection of an axis (A, C) by entering a degree:





This manufacturer-neutral machine tool forms the basis of the 5-axis documentation with programGuide:



3.3. Kinematics of 5-axis machines

3.3.1. Rotary/swivel table

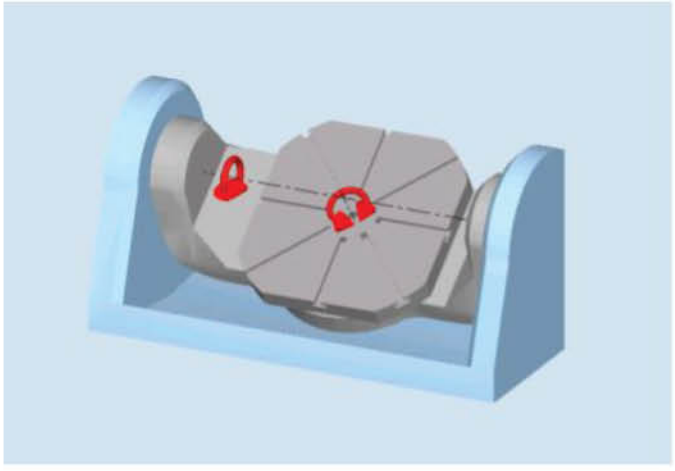


Figure 3.1: Rotary/swivel table

3.3.2. Articulated rotary/ swivel table

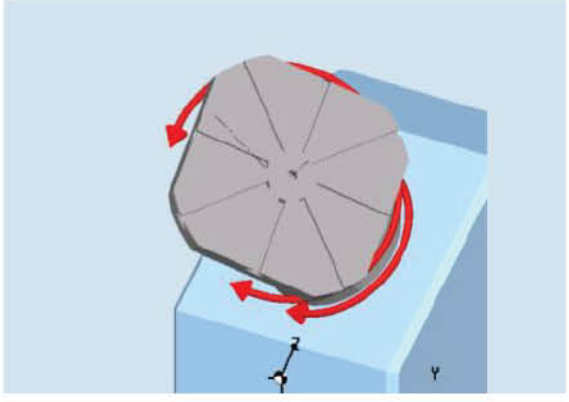


Figure 3.2: Articulated rotary/swivel table

3.3.3. Fork head

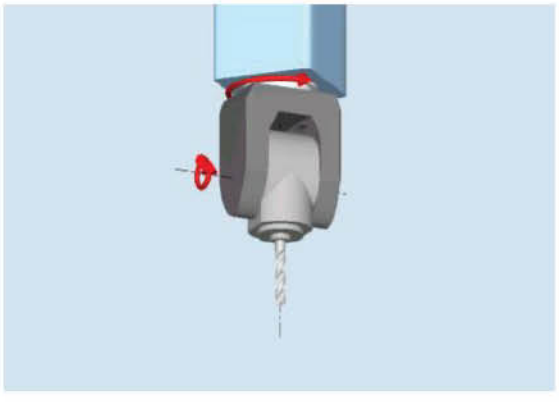


Figure 3.3: Fork head

3.3.4. Articulated head kinematics

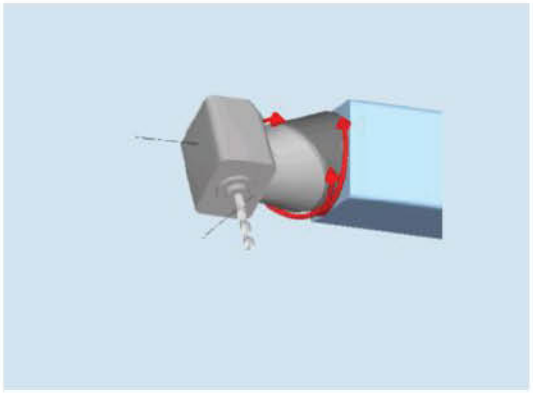


Figure 3.4: Articulated head kinematics

3.3.5. Mixed kinematics

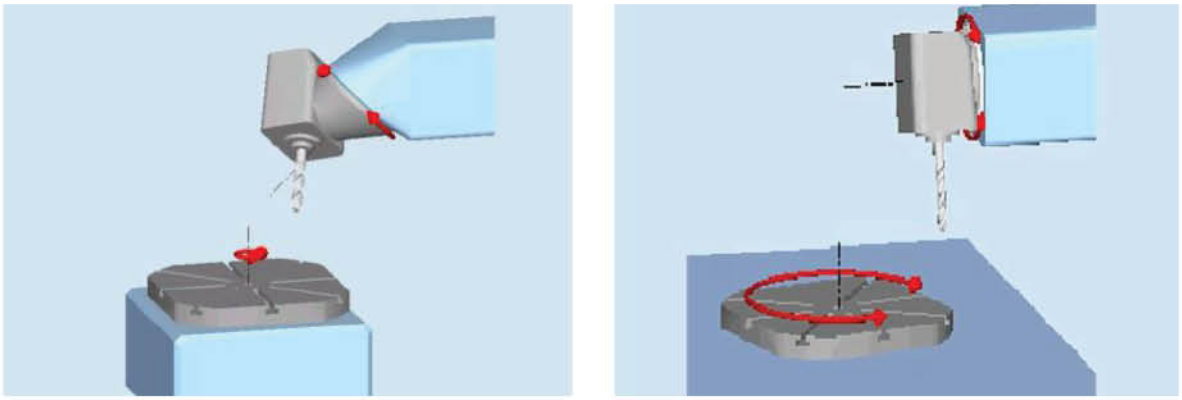
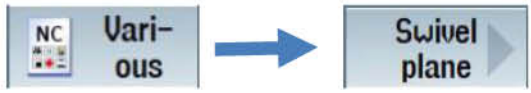


Figure 3.5: Mixed kinematics

3.4. Functional explanation for the swivel cycle (CYCLE800)

After pressing



The dialog of the machine with selected swivel data opens:

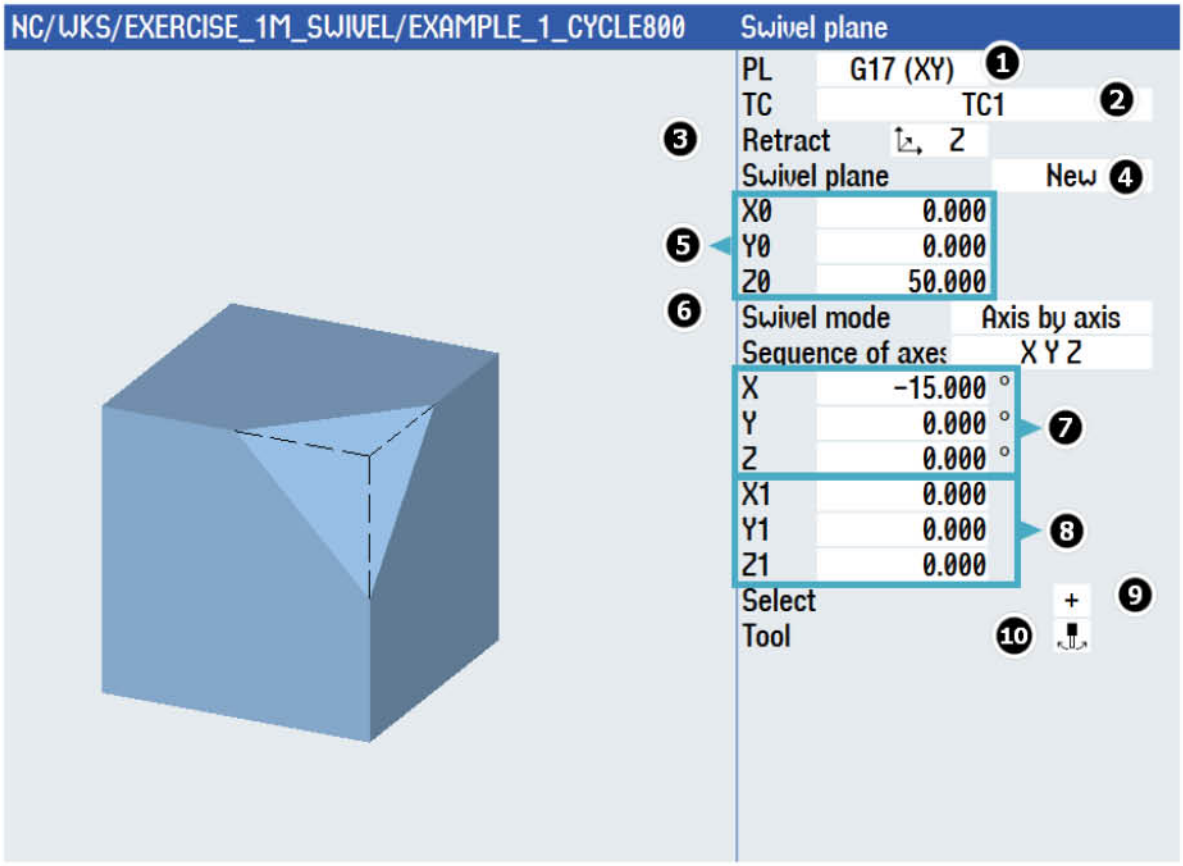


Figure 3.6: Swivel plane parameters

1. Active machining plane
2. Name of the swivel data set (Tool carrier)
3. Retraction in the direction of machine axis before the cycle starts, the retract possibilities are:

   "NO" No retract performed before swiveling

   "Z" Retract in the direction of machine axis Z

   "ZXY" Retract in the direction of machine axis Z and then in X and Y
4. New swivel plane selection with:

   "NEW" This implements a new physical swiveling on the machine.

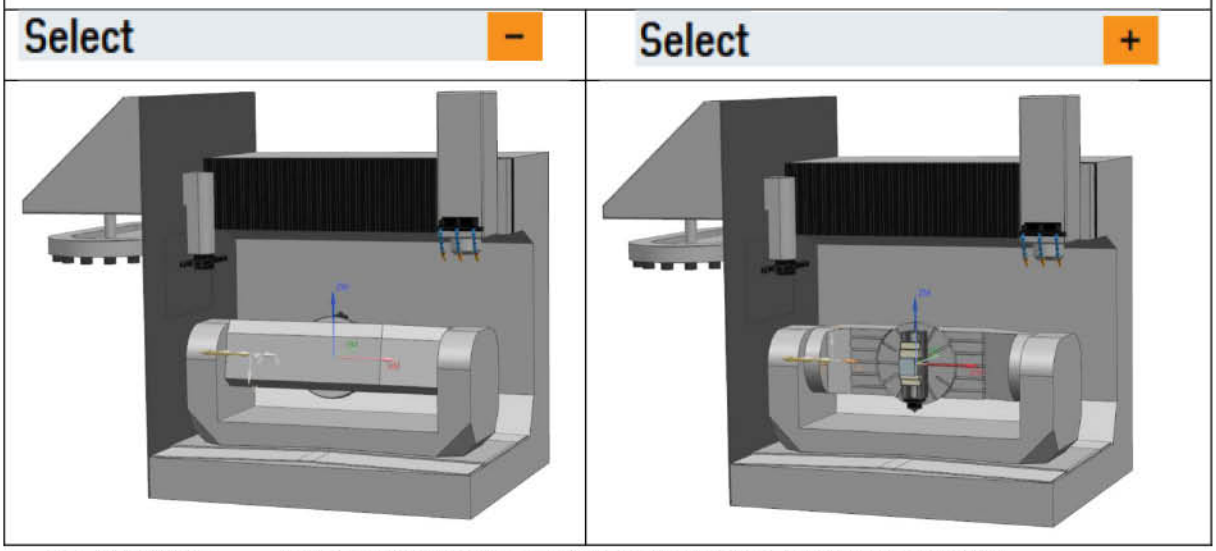
   "ADDITIVE" This is used for a further swiveling witch follows takes into account to a pre swiveld machining.
5. Reference point for X, Y and Z rotaion: If swiveling requires that the position of the coordinate system is moved, this is performed for each axis via the X0, Y0 and Z0 parame-ters. This is executed before the coordinate system rotation.

6. Swivel mode:

"Axis by Axis"	Swiveling is normally performed "axis-by-axis". Consequently, this is used in most examples and the axis sequence is generally arbitrary.
"Projection angle"	The projection angle is used for rectangular workpieces. The projection angle also consists of angles that build on each other. The angles "Xα"and "Yα" always refer to the non-rotated coordinate system. The angle "Zβ" refers to the rotated coordinate system.
"Directly"	The machine kinematics are now programmed directly here. The program depends on the real machine kinematics, as explained in this module for the "TC1" with axes A and C.

7. Rotation of workpiece coordinate system around the new reference point in X, Y and Z (see point "5").
8. Zero point move for rotated surface in X, Y and Z axes (after rotation around the axes)
9. Direction reference of traversing direction of rotary axis 1 or 2 "- " and "+" (not for direct swivel mode).

The swivel direction can be specified in the last input field of the cycle.
There are always two directions that mathematically lead to the result. Plus (+) and minus (-).
The operator can select to achieve an optimum view of the workpiece



10. Tracking: The position of the tool tip is maintained during swiveling.
    No tracking: The position of the tool tip changes during swiveling.

3.5. Swivel modes by the programming

3.5.1. Axis by axis

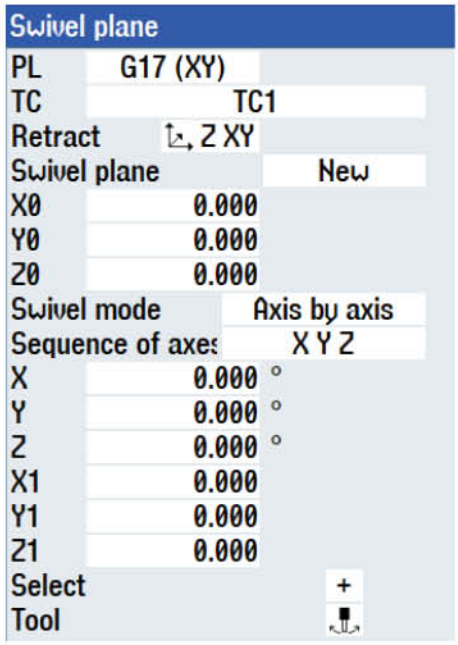


Figure 3.7: Programming with swivel mode Axis by axis

3.5.2. Projection angle

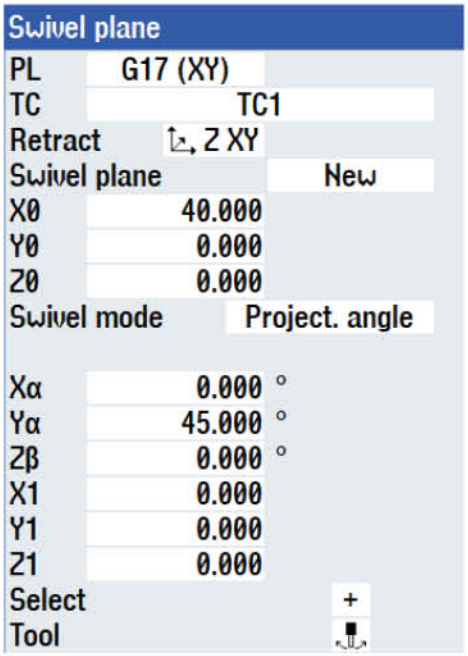


Figure 3.8: Programming with swivel mode Projection angle

3.5.3. Directly

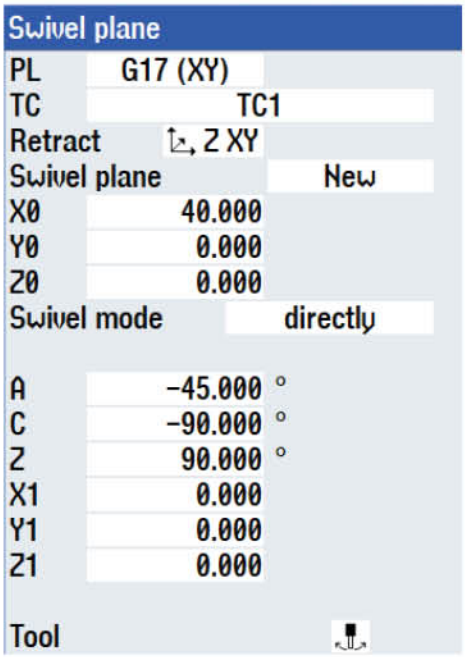


Figure 3.9: Programming with swivel mode Directly

3.6. Advantages of programming with the CYCLE800

- Quick and easy programming of machining operations in planes that can be swiveled as required without complicated calculations of rotary axis positions and transformations.
- Kinematics-independent program for “axis-by-axis” swiveling, i.e. the program runs on every machine with any kinematics type.
- The use of post processors for specific kinematics on machines is not necessary.
- The machine operator can change tool lengths in the HMI tool list at any time without making changes in the NC program.
- Zero point offsets can be defined and changed as required by the machine operator without making adjustments to the NC program (basic rotations or rotary axis offsets).
- Milling and drilling cycles can be programmed by aligning the tool with the machining plane (tool axis parallel to the Z axis in the G17 plane).
- Automatic retraction before swiveling, taking into account the software limit switches, is possible. Several retraction strategies are available for this.
- Swivel frames are retained after a power failure. Retraction from inclined workpiece contours and holes is therefore possible without any problems, taking the tool orientation into account.

3.7. Advanced programming of exercises with swivelled planes, the cycles and G-codes.

3.7.1. The first programGuide exercise is to be programmed with the help of the swivel CYCLE800.

a. Objective

A new "3+2_EXERCISE_1M_DIN.WPD" directory should be created and different programs have to be inputted in this directory.

The workpiece is to be programmed with CYCLE800 in three different swivelling modes:

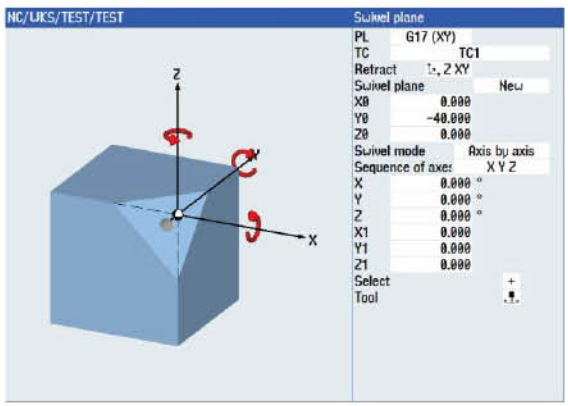
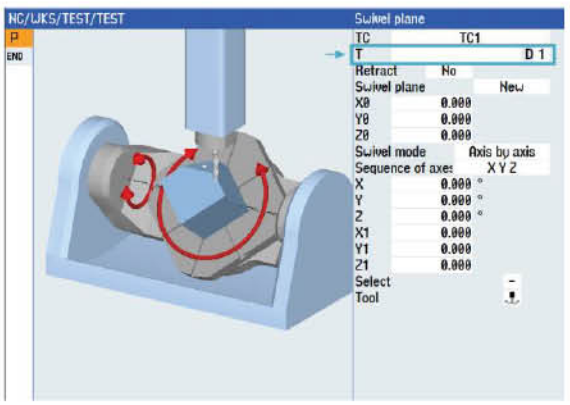
- "Axis by Axis"
- "Projection angle"
- "Directly"

In order to do that we enter three programs in this directory with the name extension of the swivelling mode.

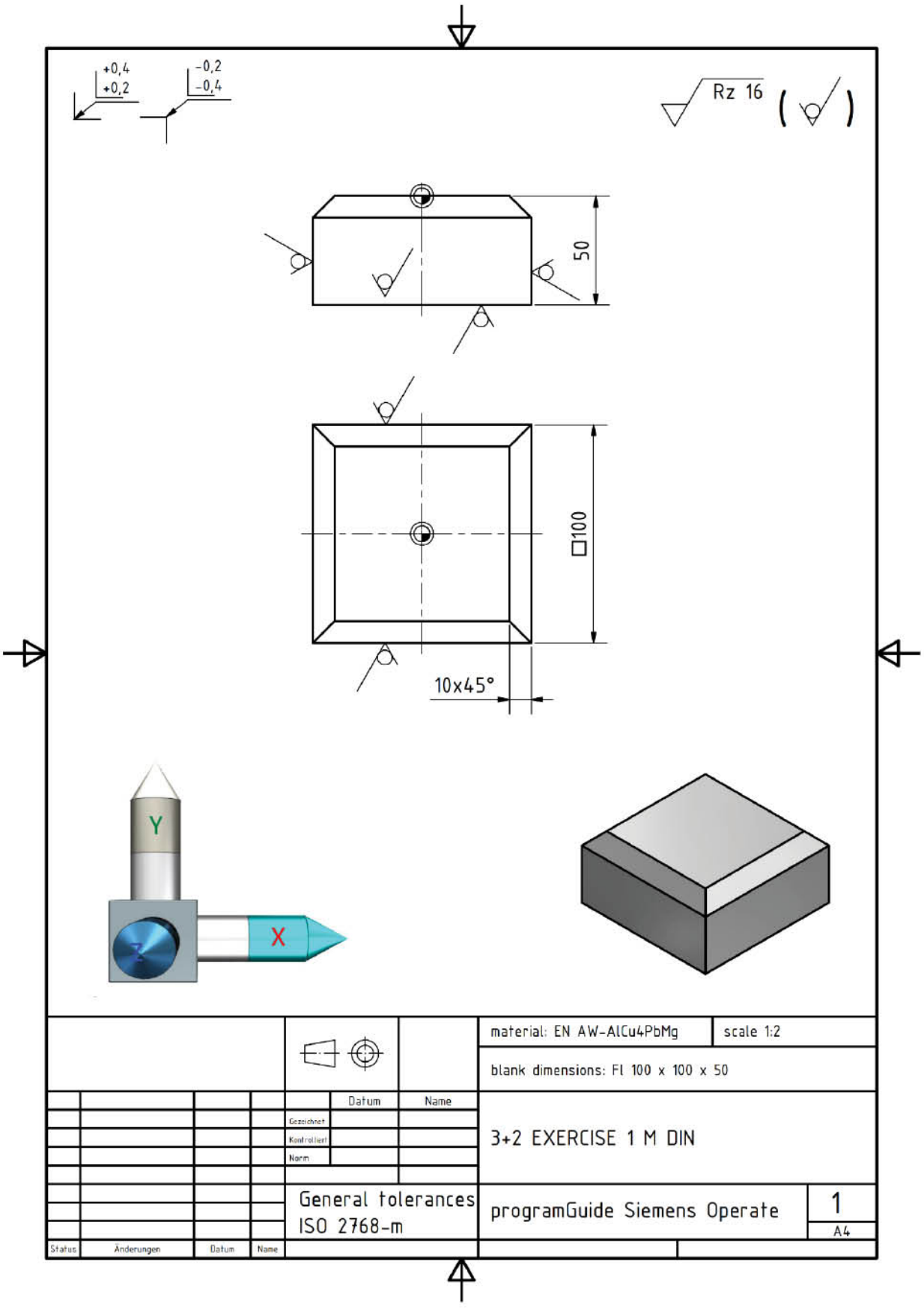
Preview of the parameters in ShopMill and programGuide for CYCLE800

Here it is shown that the only difference in the input masks of ShopMill and programGuide is that with ShopMill the tool call can also be entered in the input mask.

All other parameters are identical.

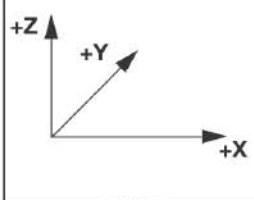
programGuide	ShopMill
	

b. The technical drawing 3+2_EXERCISE_1M_DIN:



c. The following tools and technology data are required for programming:

Workpiece: 3+2 EXERCISE 1M DIN	Blank material: EN AW-AlCu4PbMg	Machine configuration:
Drawing: 1(1)	Blank dimensions: Fl 100 x 100 x 50	Milling X Y Z A C axes

	Technological data							
Tool name in OFFSET	Spot_drill_12	Deburrer_8	Cutter_32	Cutter_16	Cutter_10	Cutter_4	Drill_8,5	Tap_10
Tool Ø (mm)	12 x 90°	8 x 90°	32	16	10	4	8,5	M10
Cutting speed (m/min)	80	105	150 / 175	150 / 175	150 / 175	150 / 175	60	10
Cutting depth a_p = max.	5	3	5	8	5	2		
Cutting material	HSS	SC	SC	SC	SC	SC	HSS	HSS
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,05 / 0,04	0,03 / 0,02	0,02 / 0,01	0,1	F = P
Number of teeth N		4	2	4	4	4		
								
				✓	✓	✓		

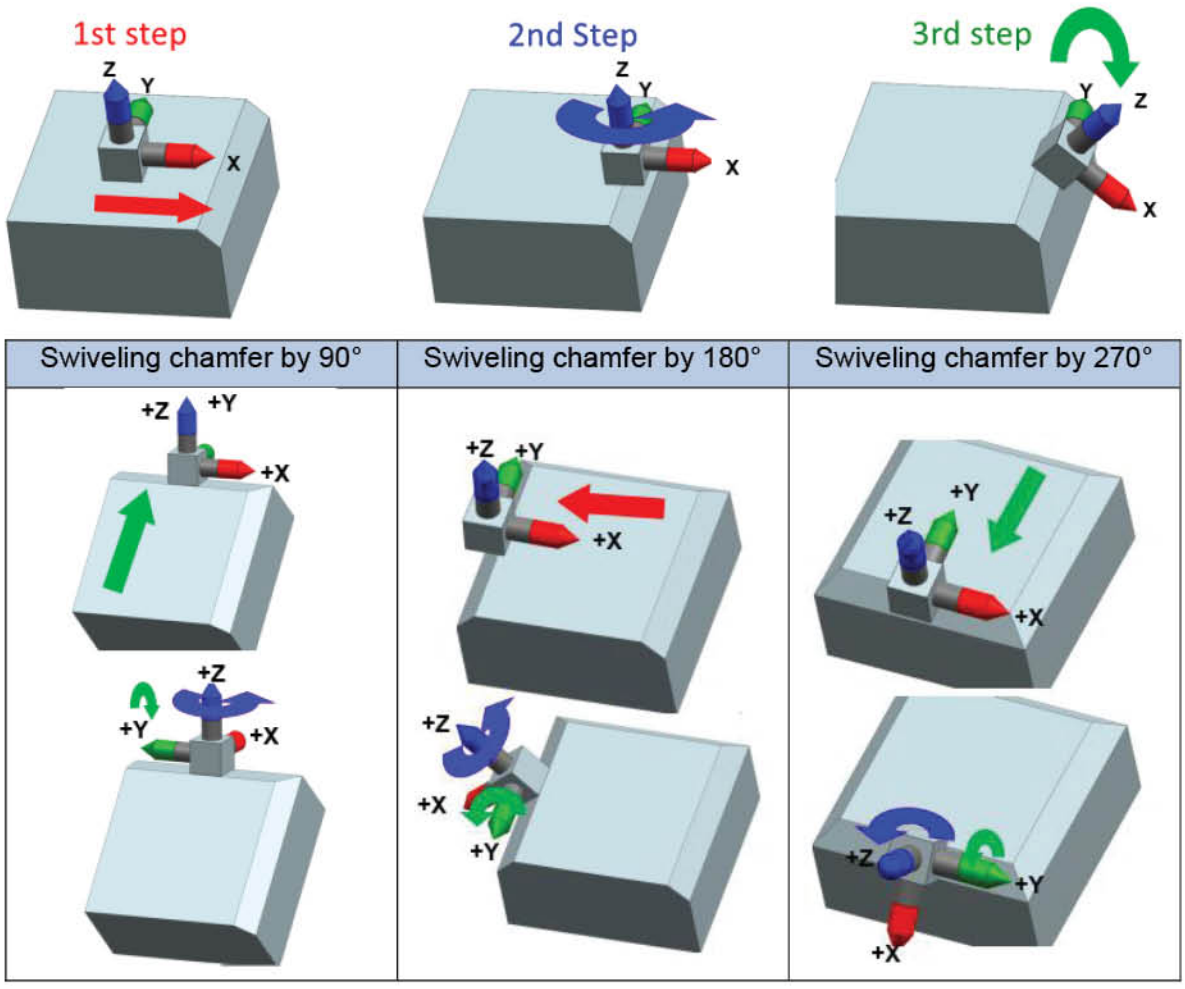
d. The workflow sequences are to programme for all three swivel modes:

- Programming the commands for the machining in G17
- Swivel basic values
- Program header
- Swivel 1
- Chamfer by 0°
- Swivel 2
- Chamfer by 90°
- Swivel 3
- Chamfer by 180°
- Swivel 4
- Chamfer by 270°
- Swivel basic values

- Deactivate swivel data set with “T0”
- Program end

e. The sequence for swiveling for all four chamfers is as follows:

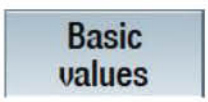
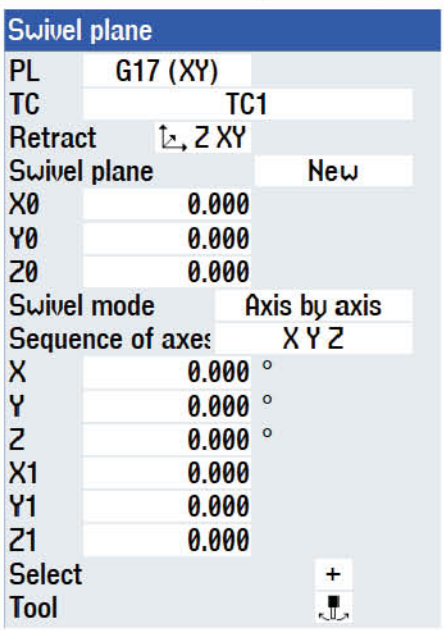
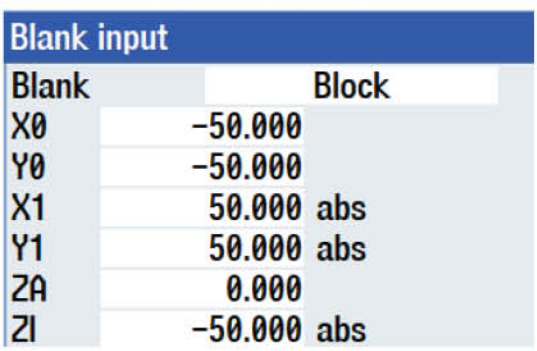
- The 1st step: Move the coordinate system from the center to the point where the edge should result.
- The 2nd step: Always rotate at the center point.
- The 3rd step: Swivel by 45°.

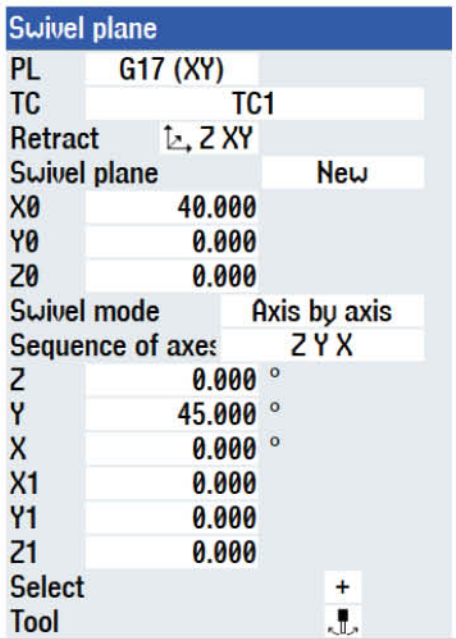


- **Swivelling axis-by-axis:**

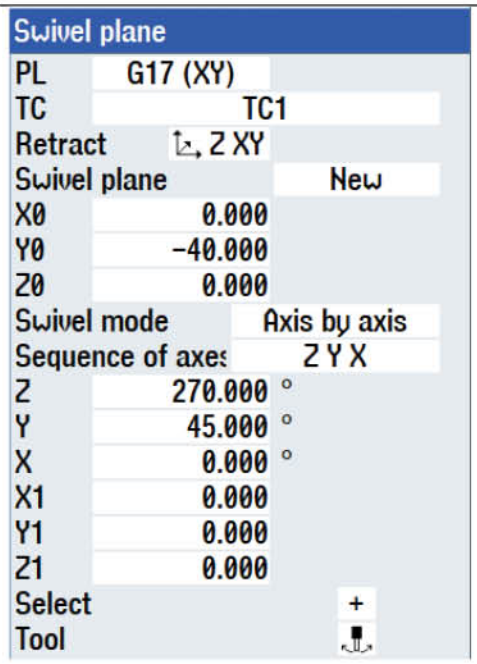
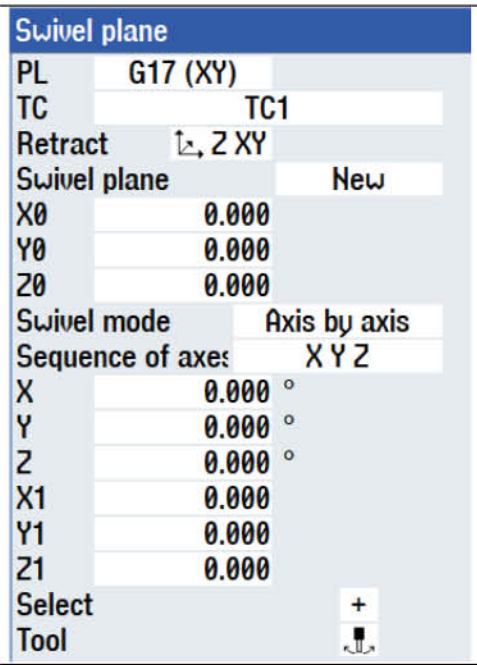
Note: Swiveling is normally performed "axis-by-axis". Consequently, this is used in most examples and the axis sequence is generally arbitrary.

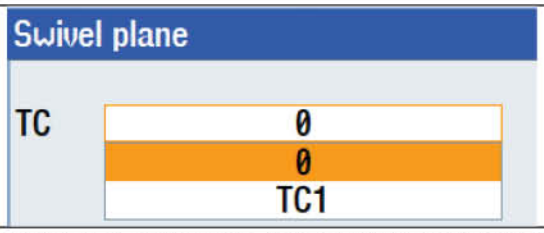
Create a new program, for example "3+2 EXERCISE_AXIS.MPF".

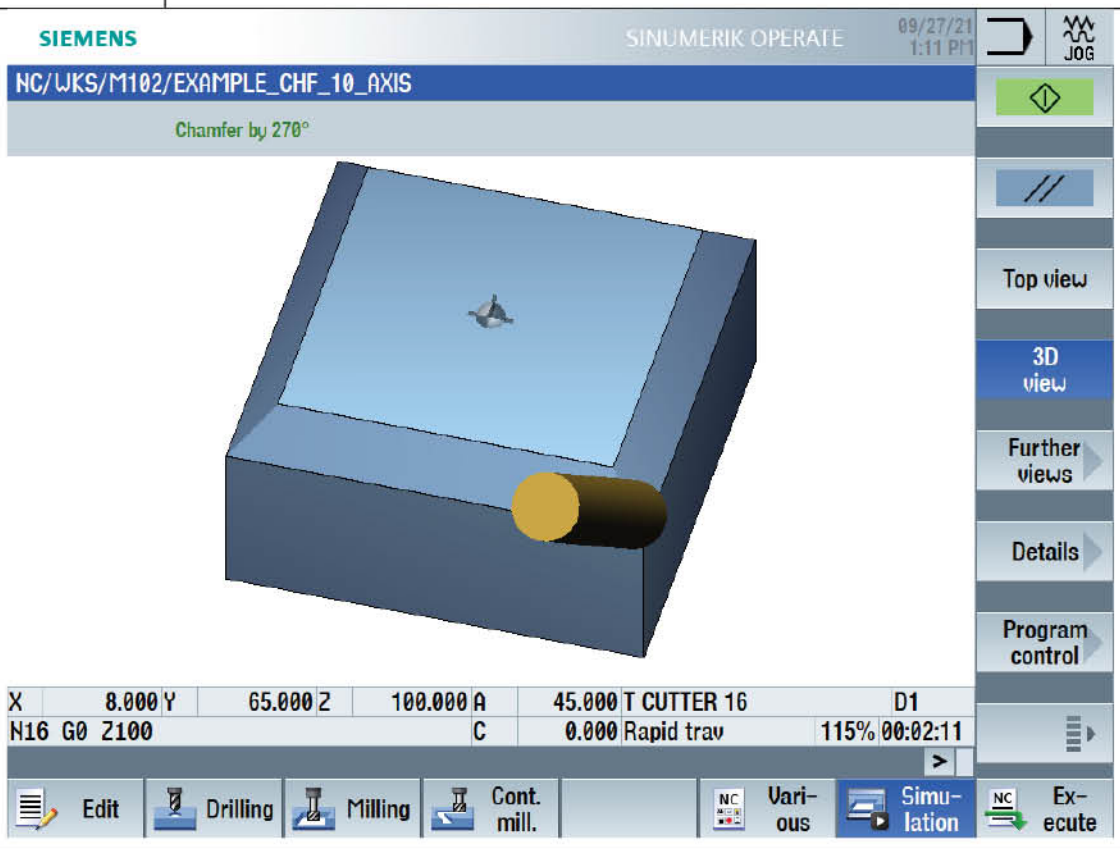
1.	Program the first lines of the program by inserting the following G code commands: **N1 G54 G17 G40 G71 G90**
2. 	Insert now the basic values for swivelling into the program. Press the HSK 1.6 “Various” to open the work area “Swivel plane”. Press the HSK 1.2 “Swivel plane”. The following input mask for “Swivel plane” opens.
3. 	Press HSK 1.3 “Basic values” and following values/parameters are displayed: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N3 CYCLE800(2,"TC1",200000,57,0,0,0,0,0,0,0,0,0,1,,1)**
4.	Enter following values/parameters for the blank: 

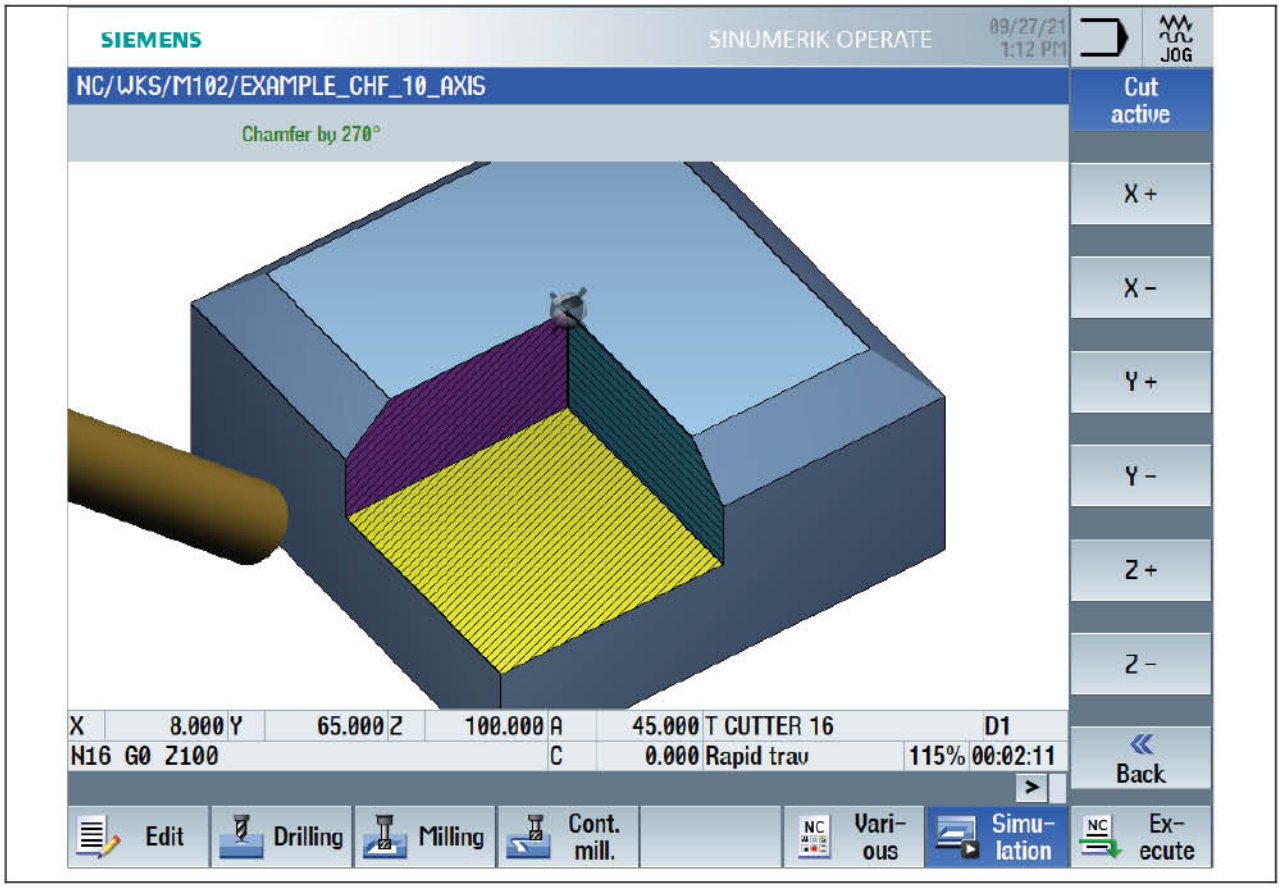
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N4 WORKPIECE(,"",,"BOX",112,0,-50,-80,-50,-50,50,50)**
5.	Insert the following technology commands into the program: **N5 T="CUTTER 16"** **N6 M6** **N7 G97 S2000 M3** **N8 G95 FZ=0.05** **N9 MSG("Chamfer by 0°")**
6. 	 Enter the following values into the input mask. Use the “Select” key where indicated.
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N10 CYCLE800(2,"TC1",200000,27,40,0,0,0,45,0,0,0,0,1,,1)**
7.	Program the following G code commands: **N11 CHR_ST:** **N12 G0 X8 Y-65 Z100** **N13 G0 Z5** **N14 G1 Z0** **N15 G1 Y65** **N16 G0 Z100** **N17 CHR_END:** **N18 MSG("Chamfer by 90°")**

8. Swivel plane	**Swivel plane** PL G17 (XY) TC TC1 Retract Z XY Swivel plane New X0 0.000 Y0 40.000 Z0 0.000 Swivel mode Axis by axis Sequence of axes Z Y X Z 90.000 ° Y 45.000 ° X 0.000 ° X1 0.000 Y1 0.000 Z1 0.000 Select + Tool	Enter the following values into the input mask. Use the "Select" key where indicated.
Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N19 CYCLE800(2,"TC1",200000,27,0,40,0,90,45,0,0,0,0,1,,1)**	
9.	Program the following commands to repeat the labels: **N20 REPEAT CHR_ST CHR_END** **N21 MSG("Chamfer by 180°")**	
10. Swivel plane	**Swivel plane** PL G17 (XY) TC TC1 Retract Z XY Swivel plane New X0 -40.000 Y0 0.000 Z0 0.000 Swivel mode Axis by axis Sequence of axes Z Y X Z 180.000 ° Y 45.000 ° X 0.000 ° X1 0.000 Y1 0.000 Z1 0.000 Select + Tool	Enter the following values into the input mask. Use the "Select" key where indicated.
Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N22 CYCLE800(2,"TC1",200000,27,-40,0,0,180,45,0,0,0,0,1,,1)**	

11.	Program the following commands to repeat the labels: **N23 REPEAT CHR_ST CHR_END** **N24 MSG("Chamfer by 270°")**
12. Swivel plane	 Enter the following values into the input mask. Use the "Select" key where indicated.
Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N25 CYCLE800(2,"TC1",200000,27,0,-40,0,270,45,0,0,0,0,1,,1)**
13.	Program the following commands to repeat the labels: **N26 REPEAT CHR_ST CHR_END** **N27 MSG("Basic values swivel")**
14. Swivel plane Basic values	 Press HSK 1.3 "Basic values" and following values/parameters are displayed:

Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N28 CYCLE800(2,"TC1",200000,57,0,0,0,0,0,0,0,0,1,,1)**
15. Swivel plane	 Choose “T0” to deactivate swivel data set with.
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N29 CYCLE800()**
16.	Program the following G code commands: **N30 T0 M6** **N31 M30**
17.	Finally check the machining with the simulation by using of different views.




SIEMENS
SINUMERIK OPERATE
09/27/21
1:12 PM
JOG
NC/WKS/M102/EXAMPLE_CHF_10_AXIS
Chamfer by 270°
Cut active
X +
X -
Y +
Y -
Z +
Z -
X 8.000 Y 65.000 Z 100.000 A 45.000 T CUTTER 16 D1
N16 G0 Z100 C 0.000 Rapid trav 115% 00:02:11
Back
Edit
Drilling
Milling
Cont. mill.
Vari- ous
Simu- lation
Ex- ecute

- **Swivelling projection angle:**

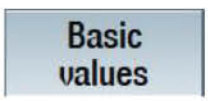
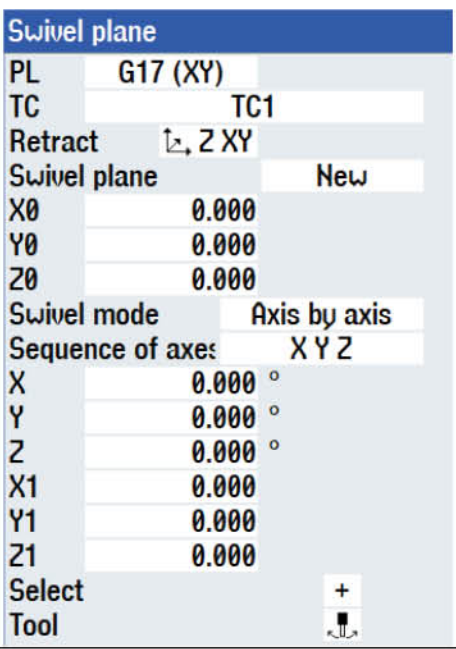
  Note: The projection angle is used for rectangular workpieces.

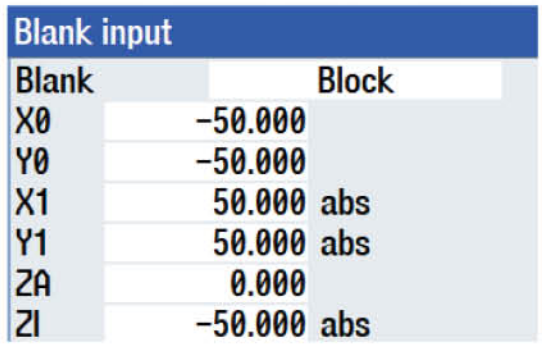
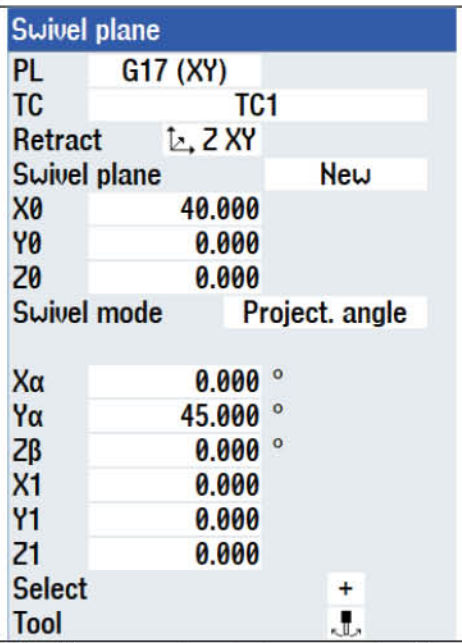
  The projection angle also consists of angles that build on each other.

  The angles “Xα”and “Yα” always refer to the non-rotated coordinate system.

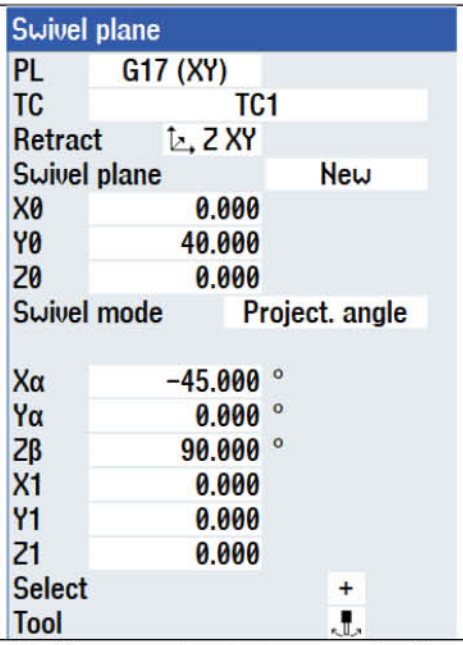
  The angle “Zβ” refers to the rotated coordinate system.

  Create a new program, for example "3+2 EXERCISE_PROJECTION.MPF".

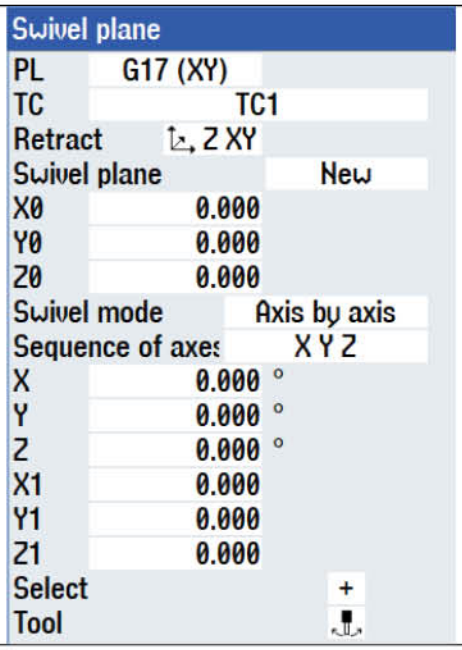
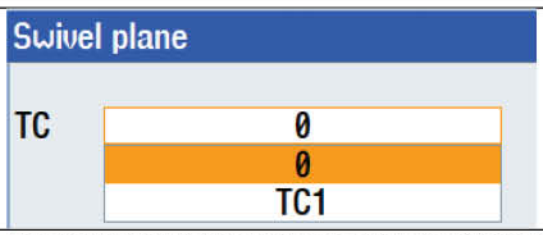
1.	Program the first lines of the program by inserting the following G code commands: **N1 G54 G17 G40 G71 G90**
2. 	Insert now the basic values for swivelling into the program. Press the HSK 1.6 “Various” to open the work area “Swivel plane”. Press the HSK 1.2 “Swivel plane”. The following input mask for “Swivel plane” opens.
3. 	Press HSK 1.3 “Basic values” and following values/parameters are displayed: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N3 CYCLE800(2,"TC1",200000,57,0,0,0,0,0,0,0,0,0,1,,1)**

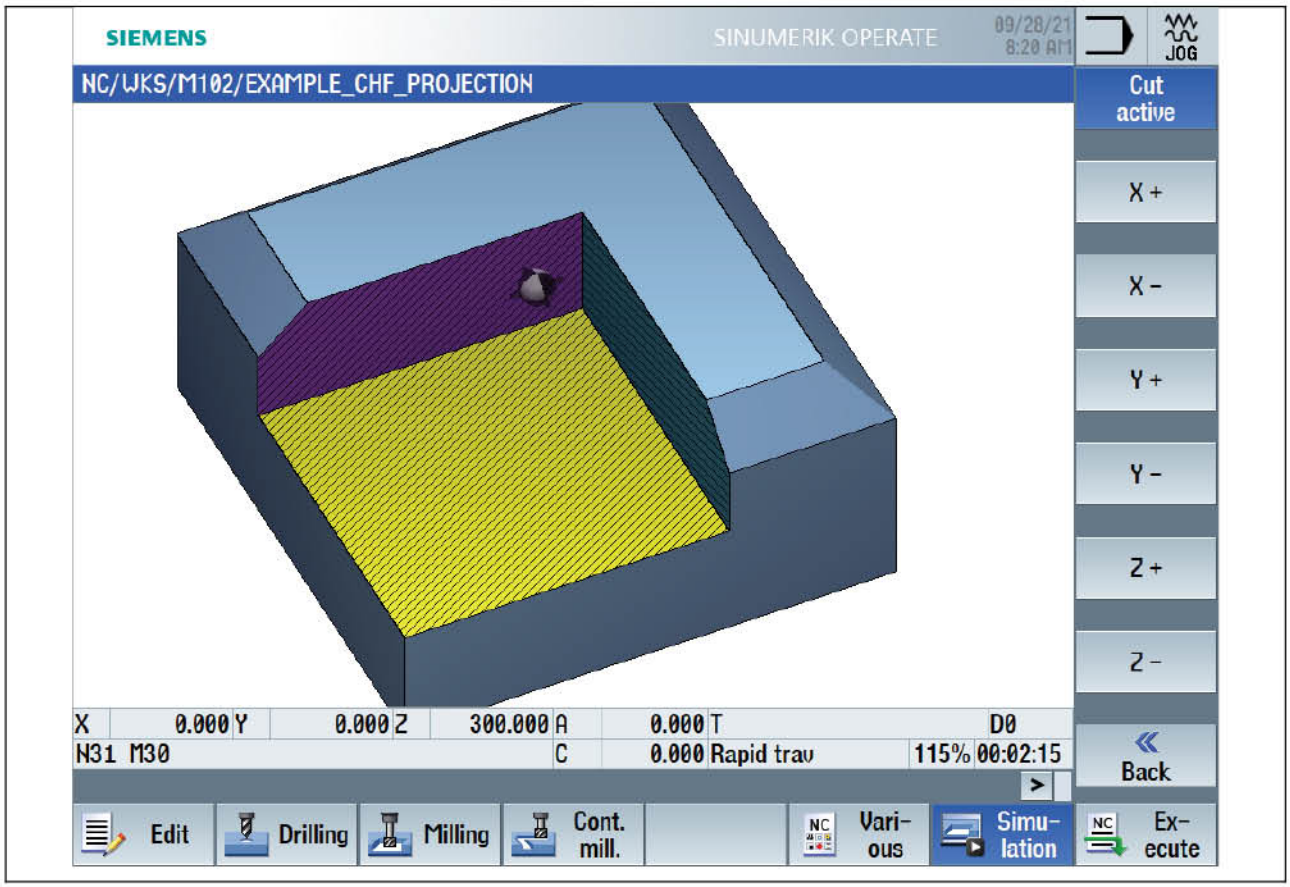
<table>
<tr><td>4.</td><td>Enter following values/parameters for the blank:
 
</td></tr>
<tr><td>
</td><td>Confirm your inputs by pressing the VSK 8 “Accept”.
The following line is inserted into the program:
N4 WORKPIECE(,"",,"BOX",112,0,-50,-80,-50,-50,50,50)</td></tr>
<tr><td>5.</td><td>Insert the following technology commands into the program:
N5 T="CUTTER 16"
N6 M6
N7 G97 S2000 M3
N8 G95 FZ=0.05
N9 MSG("Chamfer by 0°")</td></tr>
<tr><td>6.
</td><td> 

Enter the following values into the input mask. Use the “Select” key where indicated.</td></tr>
<tr><td>
</td><td>Confirm your inputs by pressing the VSK 8 “Accept”.
The following line is inserted into the program:
N10 CYCLE800(2,"TC1",200000,185,40,0,0,0,45,0,0,0,0,1,,1)</td></tr>
</table>

<table>
<tr><td>7.</td><td>Program the following G code commands:
N11 CHR_ST:
N12 G0 X8 Y-65 Z100
N13 G0 Z5
N14 G1 Z0
N15 G1 Y65
N16 G0 Z100
N17 CHR_END:
N18 MSG("Chamfer by 90°")</td></tr>
<tr><td>8.
 </td><td> 
Enter the following values into the input mask. Use the “Select” key where indicated.</td></tr>
<tr><td> </td><td>Confirm your inputs by pressing the VSK 8 “Accept”.
The following line is inserted into the program:
N19 CYCLE800(2,"TC1",200000,185,0,40,0,-45,0,90,0,0,0,1,,1)</td></tr>
<tr><td>9.</td><td>Program the following commands to repeat the labels:
N20 REPEAT CHR_ST CHR_END
N21 MSG("Chamfer by 180°")</td></tr>
</table>

10. Swivel plane	Swivel plane PL G17 (XY) TC TC1 Retract Z XY Swivel plane New X0 -40.000 Y0 0.000 Z0 0.000 Swivel mode Project. angle Xα 0.000 ° Yα -45.000 ° Zβ 180.000 ° X1 0.000 Y1 0.000 Z1 0.000 Select + Tool	Enter the following values into the input mask. Use the "Select" key where indicated.
Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N22 CYCLE800(2,"TC1",200000,185,-40,0,0,0,-45,180,0,0,0,1,,1)**	
11.	Program the following commands to repeat the labels: **N23 REPEAT CHR_ST CHR_END** **N24 MSG("Chamfer by 270°")**	
12. Swivel plane	Swivel plane PL G17 (XY) TC TC1 Retract Z XY Swivel plane New X0 0.000 Y0 -40.000 Z0 0.000 Swivel mode Project. angle Xα 45.000 ° Yα 0.000 ° Zβ 270.000 ° X1 0.000 Y1 0.000 Z1 0.000 Select + Tool	Enter the following values into the input mask. Use the "Select" key where indicated.
Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N25 CYCLE800(2,"TC1",200000,185,0,-40,0,45,0,270,0,0,0,1,,1)**	

13.	Program the following commands to repeat the labels: **N26 REPEAT CHR_ST CHR_END** **N27 MSG("Basic values swivel")**
14. Swivel plane Basic values	 Press HSK 1.3 “Basic values” and following values/parameters are displayed:
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N28 CYCLE800(2,"TC1",200000,57,0,0,0,0,0,0,0,0,0,1,,1)**
15. Swivel plane	 Choose “T0” to deactivate swivel data set with.
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N29 CYCLE800()**
16.	Program the following G code commands: **N30 T0 M6** **N31 M30**
17.	Finally check the machining with the simulation by using of different views.

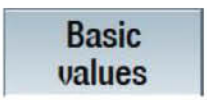
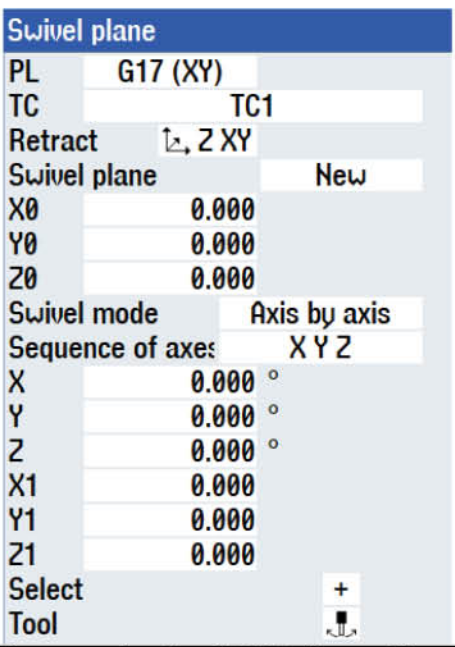

SIEMENS
SINUMERIK OPERATE
09/28/21
8:20 AM
JOG
NC/WKS/M102/EXAMPLE_CHF_PROJECTION
Cut active
X +
X -
Y +
Y -
Z +
Z -
X 0.000 Y 0.000 Z 300.000 A 0.000 T D0
N31 M30 C 0.000 Rapid trav 115% 00:02:15
Back
Edit
Drilling
Milling
Cont. mill.
Vari- ous
Simu- lation
Ex- ecute

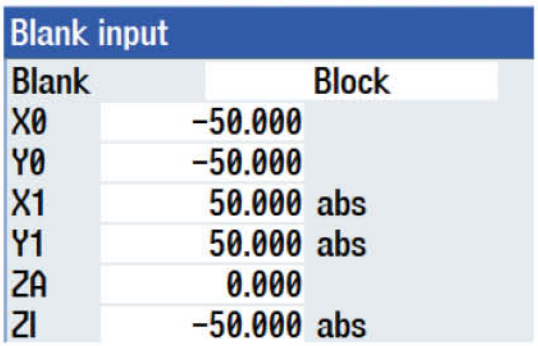
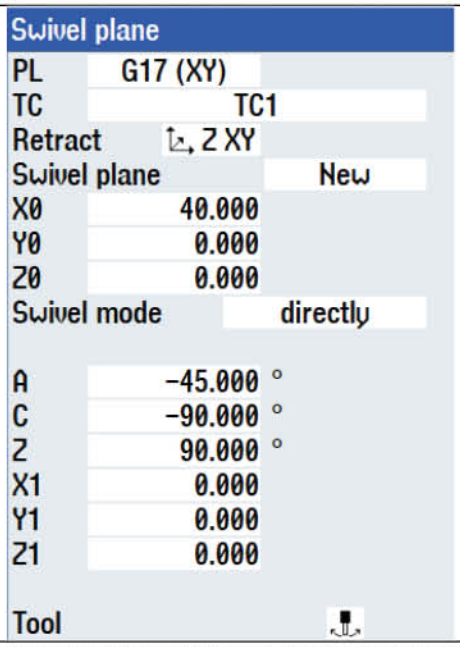
- **Swivelling by rotary axes directly:**

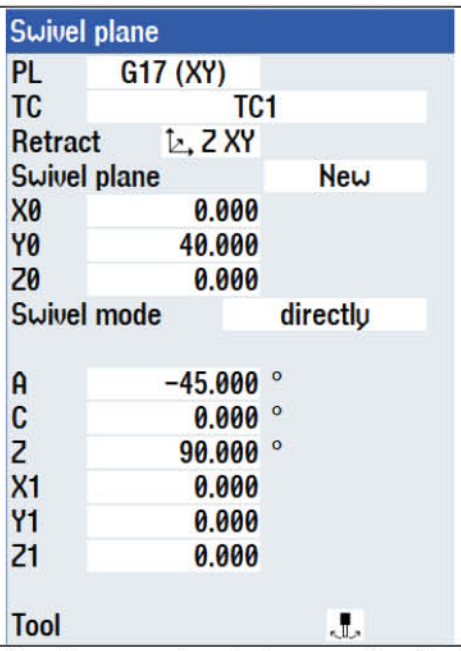
  Note: The machine kinematics are now programmed directly here.

  The program depends on the real machine kinematics, as explained in this unit for the "TC1" with axes A and C.

  Create a new program, for example "3+2 EXERCISE_DIRECTLY.MPF".

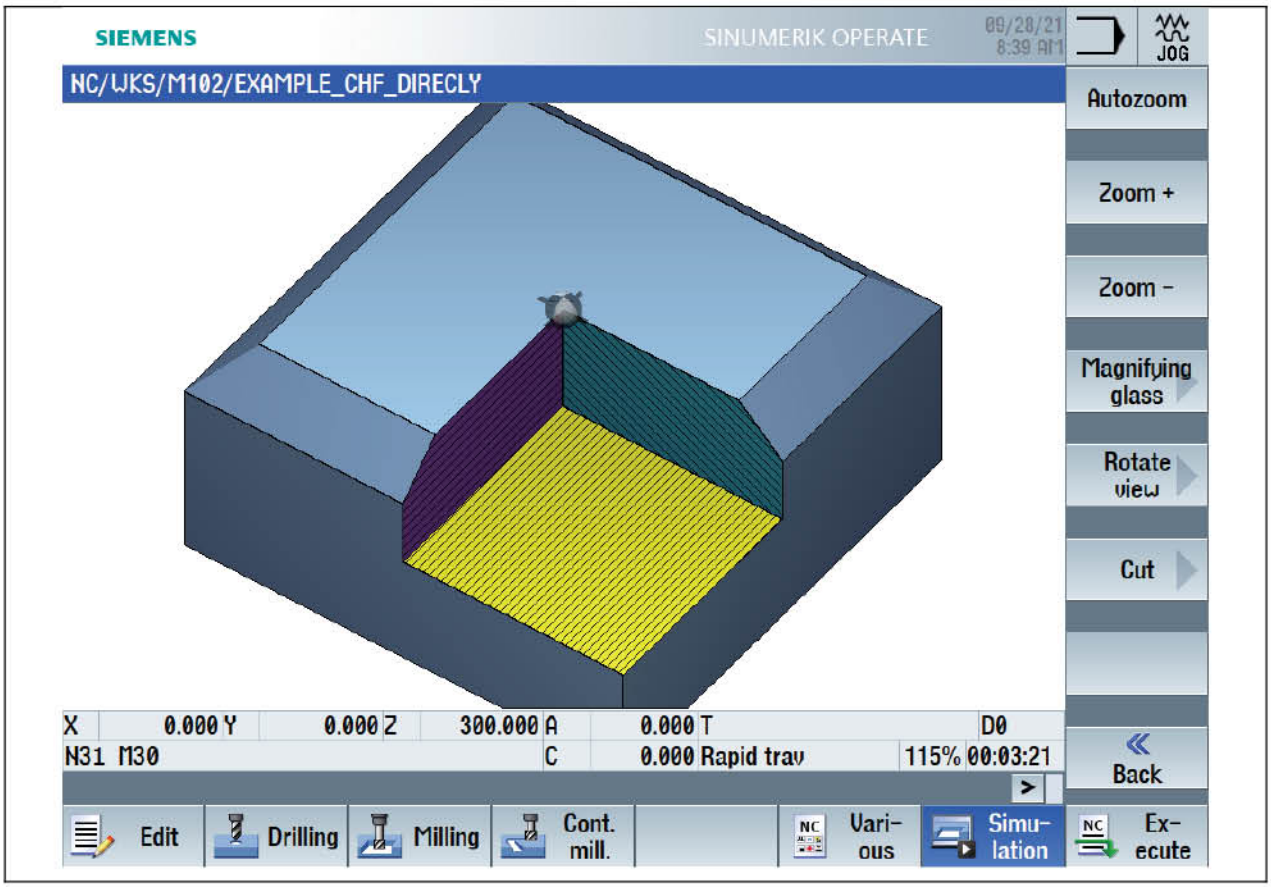
1.	Program the first lines of the program by inserting the following G code commands: **N1 G54 G17 G40 G71 G90**
2. 	Insert now the basic values for swivelling into the program. Press the HSK 1.6 “Various” to open the work area “Swivel plane”. Press the HSK 1.2 “Swivel plane”. The following input mask for “Swivel plane” opens.
3. 	Press HSK 1.3 “Basic values” and following values/parameters are displayed: 
	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N3 CYCLE800(2,"TC1",200000,57,0,0,0,0,0,0,0,0,0,1,,1)**

<table>
<tr><td>4.</td><td>Enter following values/parameters for the blank:
 
</td></tr>
<tr><td>
</td><td>Confirm your inputs by pressing the VSK 8 “Accept”.
The following line is inserted into the program:
N4 WORKPIECE(,"",,"BOX",112,0,-50,-80,-50,-50,50,50)</td></tr>
<tr><td>5.</td><td>Insert the following technology commands into the program:
N5 T="CUTTER 16"
N6 M6
N7 G97 S2000 M3
N8 G95 FZ=0.05
N9 MSG("Chamfer by 0°")</td></tr>
<tr><td>6.
</td><td> 
</td><td>Enter the following values into the input mask. Use the “Select” key where indicated.</td></tr>
<tr><td>
</td><td>Confirm your inputs by pressing the VSK 8 “Accept”.
The following line is inserted into the program:
N10 CYCLE800(2,"TC1",200000,192,40,0,0,-45,-90,90,0,0,0,1,,1)</td></tr>
</table>

<table>
<tr><td>7.</td><td>Program the following G code commands:
N11 CHR_ST:
N12 G0 X8 Y-65 Z100
N13 G0 Z5
N14 G1 Z0
N15 G1 Y65
N16 G0 Z100
N17 CHR_END:
N18 MSG("Chamfer by 90°")</td></tr>
<tr><td>8.
 </td><td> 
Enter the following values into the input mask. Use the “Select” key where indicated.</td></tr>
<tr><td> </td><td>Confirm your inputs by pressing the VSK 8 “Accept”.
The following line is inserted into the program:
N19 CYCLE800(2,"TC1",200000,192,0,40,0,-45,0,90,0,0,0,1,,1)</td></tr>
<tr><td>9.</td><td>Program the following commands to repeat the labels:
N20 REPEAT CHR_ST CHR_END
N21 MSG("Chamfer by 180°")</td></tr>
</table>

10. Swivel plane	Swivel plane PL G17 (XY) TC TC1 Retract Z XY Swivel plane New X0 -40.000 Y0 0.000 Z0 0.000 Swivel mode directly A -45.000 ° C 90.000 ° Z 90.000 ° X1 0.000 Y1 0.000 Z1 0.000 Tool	Enter the following values into the input mask. Use the "Select" key where indicated.
Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N22 CYCLE800(2,"TC1",200000,192,-40,0,0,-45,90,90,0,0,0,1,,1)**	
11.	Program the following commands to repeat the labels: **N23 REPEAT CHR_ST CHR_END** **N24 MSG("Chamfer by 270°")**	
12. Swivel plane	Swivel plane PL G17 (XY) TC TC1 Retract Z XY Swivel plane New X0 0.000 Y0 -40.000 Z0 0.000 Swivel mode directly A -45.000 ° C 180.000 ° Z 90.000 ° X1 0.000 Y1 0.000 Z1 0.000 Tool	Enter the following values into the input mask. Use the "Select" key where indicated.
Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N25 CYCLE800(2,"TC1",200000,192,0,-40,0,-45,180,90,0,0,0,1,,1)**	

13.	Program the following commands to repeat the labels: **N26 REPEAT CHR_ST CHR_END** **N27 MSG("Basic values swivel")**	
14. Swivel plane Basic values	Swivel plane PL G17 (XY) TC TC1 Retract Z XY Swivel plane New X0 0.000 Y0 0.000 Z0 0.000 Swivel mode Axis by axis Sequence of axes X Y Z X 0.000 ° Y 0.000 ° Z 0.000 ° X1 0.000 Y1 0.000 Z1 0.000 Select Tool +	Press HSK 1.3 "Basic values" and following values/parameters are displayed:
Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N28 CYCLE800(2,"TC1",200000,57,0,0,0,0,0,0,0,0,0,1,,1)**	
15. Swivel plane	Swivel plane TC 0 0 TC1	Choose "T0" to deactivate swivel data set with.
Accept	Confirm your inputs by pressing the VSK 8 "Accept". The following line is inserted into the program: **N29 CYCLE800()**	
16.	Program the following G code commands: **N30 T0 M6** **N31 M30**	
17.	Finally check the machining with the simulation by using of different views.	


SIEMENS
SINUMERIK OPERATE
09/28/21
8:39 AM
JOG
NC/WKS/M102/EXAMPLE_CHF_DIRECLY
Autozoom
Zoom +
Zoom –
Magnifying glass
Rotate view
Cut
X 0.000 Y 0.000 Z 300.000 A 0.000 T D0
N31 M30 C 0.000 Rapid trav 115% 00:03:21
Back
Edit
Drilling
Milling
Cont. mill.
Vari- ous
Simu- lation
Ex- ecute

3.7.2. Complex milling exercise with 3+2 axes with programGUIDE

a. Description of the work sequence

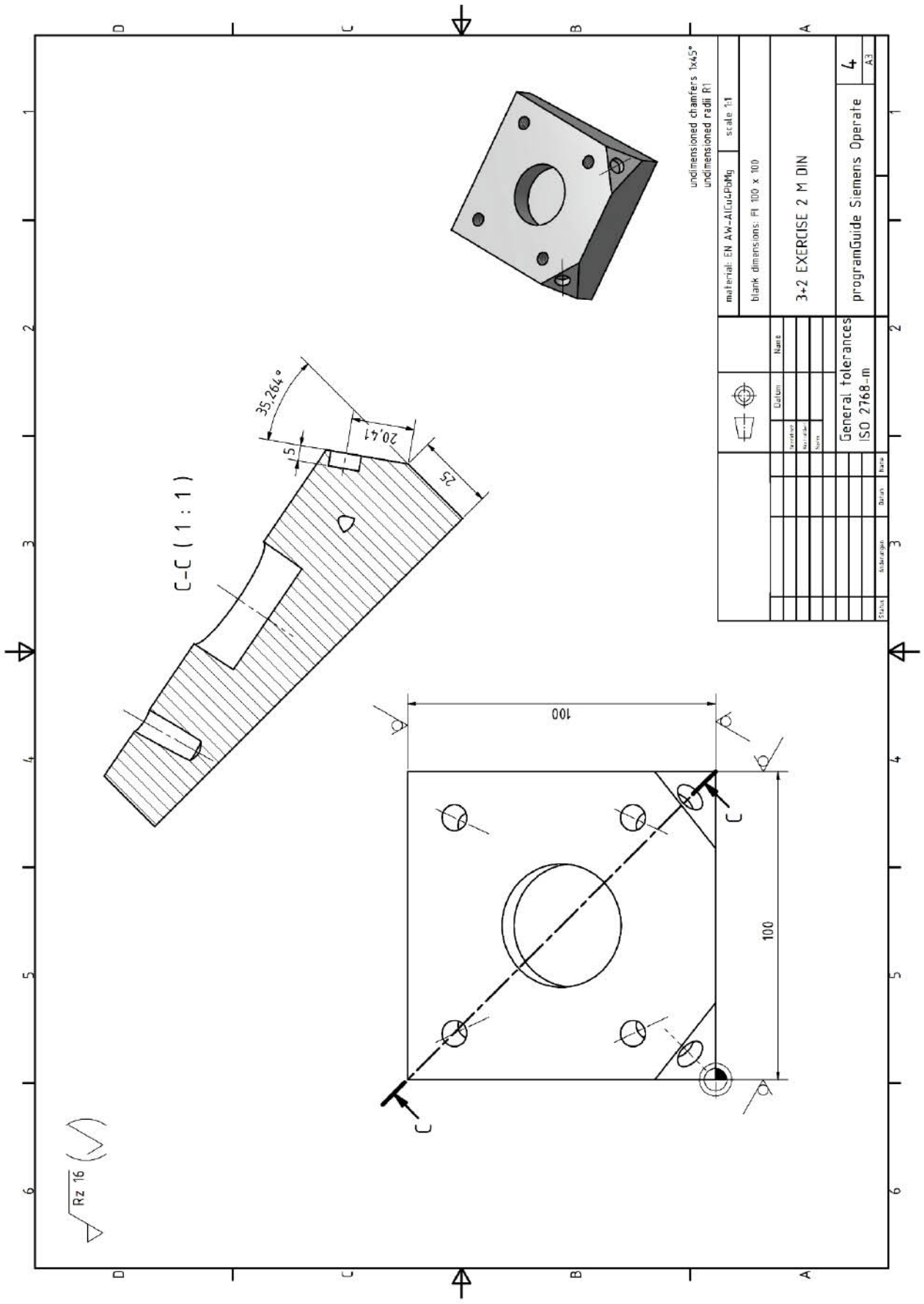
The following workpiece is to be programmed with G code on a workpiece with the blank dimensions of 100 x 100 x 52.

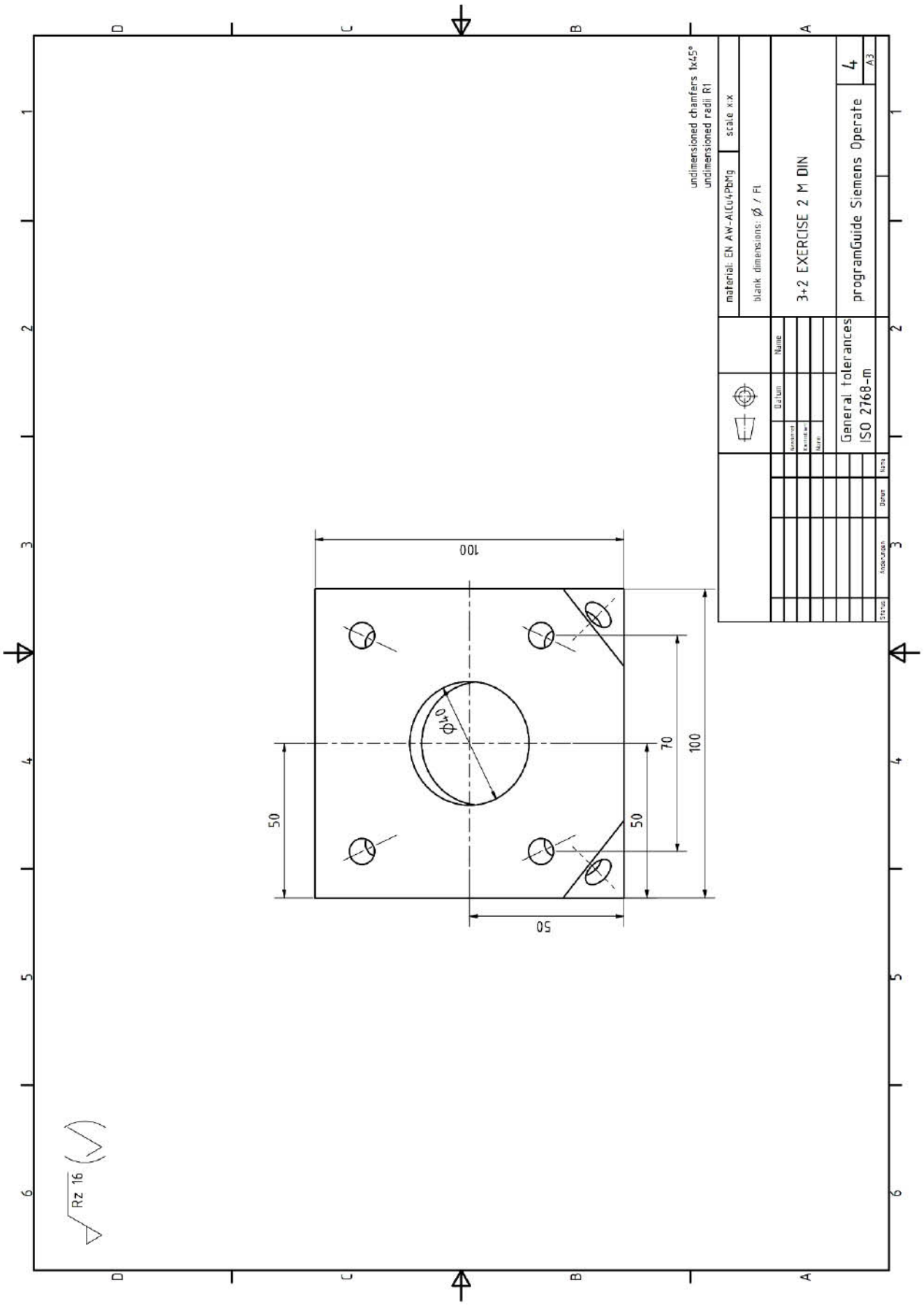
- Programming in programGUIDE
- Face milling in the initial setting
- Milling oblique surface 15°
- Milling centred circular pocket (roughing, finishing)
- Milling "1st corner" (roughing to the final dimension)
- Milling "2nd corner" (roughing to the final dimension)
- Drilling operations for the four taps M10 (-7° and +7°) with:
  + Spot drilling Ø10,8
  + Core hole drilling Ø8,5 mm
  + Tapping M10
- 1st circular pocket on oblique surface left on the initial work zero (roughing, finishing)
- 2nd circular pocket on oblique surface right (roughing, finishing)
- Optional: Chamfer milling 1,5 mm on five contour elements.

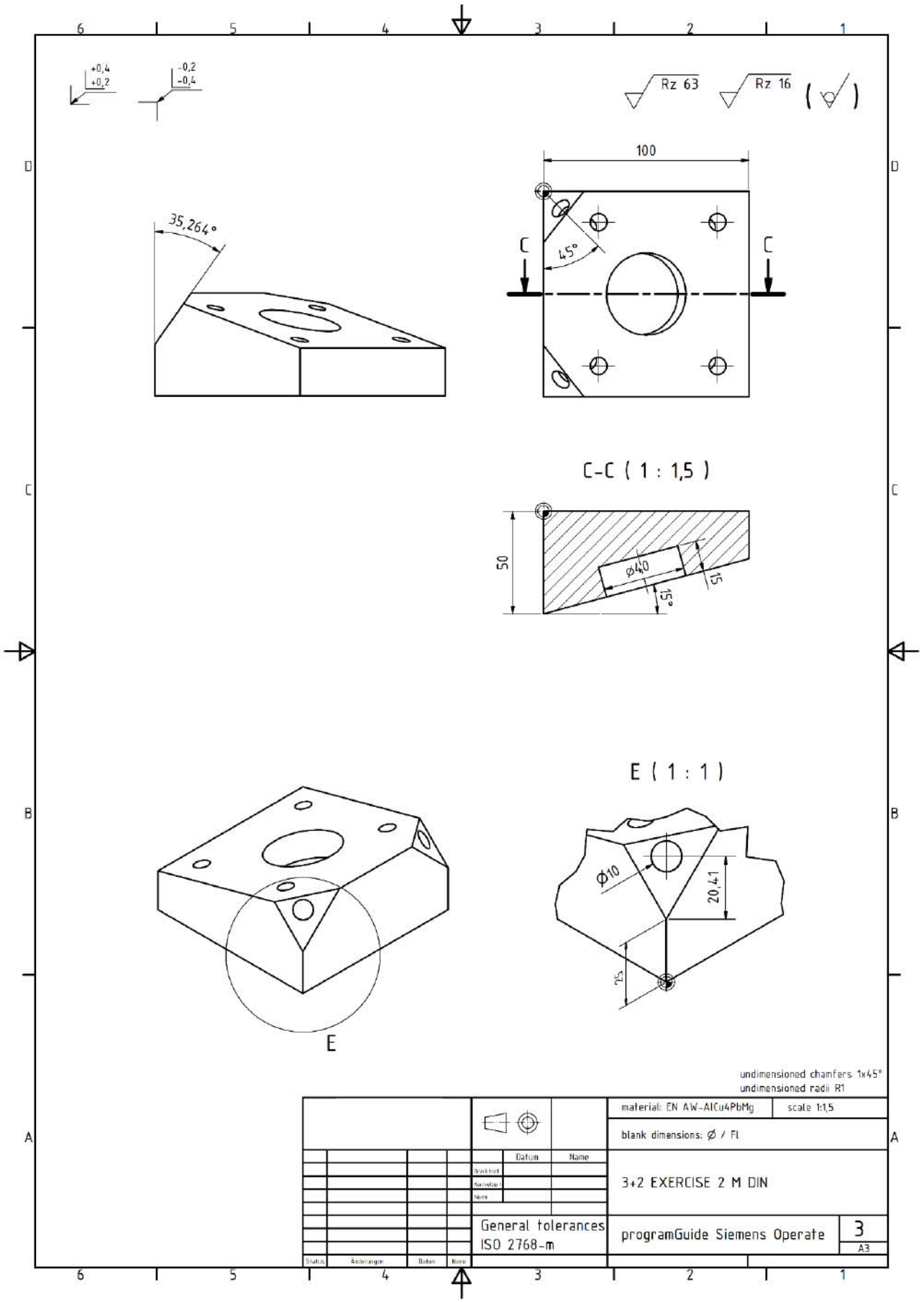
In order to see all details of the workpiece four technical drawings are shown on the pages 294 until 297.

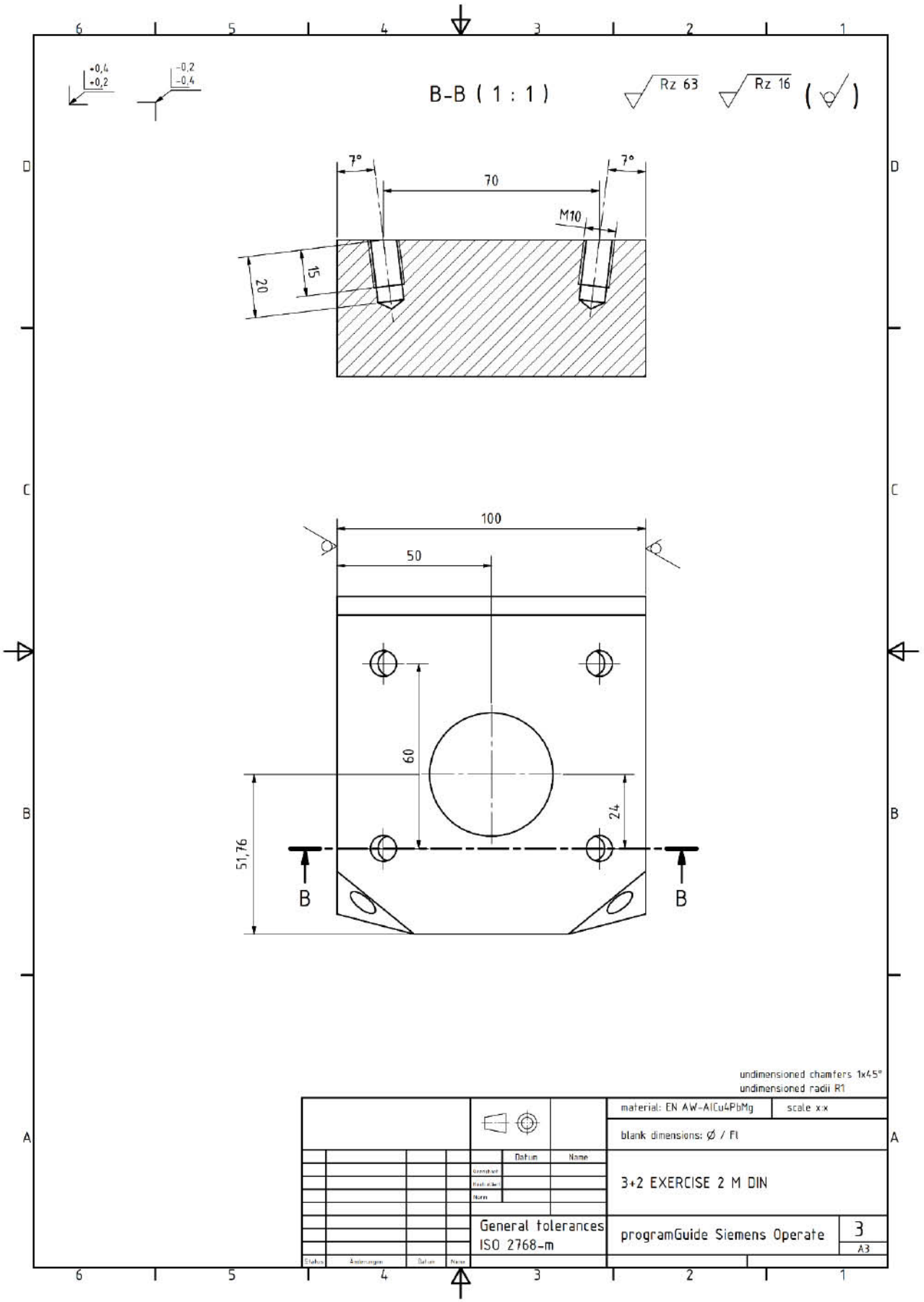
The available tool shape for the example is displayed at page number 298.

Some additional tasks are described at the end of this example to deepen the learned steps and in order to mill chamfers by swivelling on the workpiece.


undimensioned chamfers 1x45°
undimensioned radii R1
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Fl 100 x 100
3+2 EXERCISE 2 M DIN
4
A3
programGuide Siemens Operate
General tolerances
ISO 2768-m
Name
Datum
35,264°
20,41
15
25
C-C (1 : 1)
100
100
C
C
Rz 16

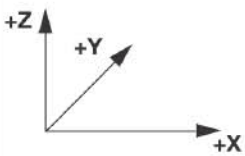

undimensioned chamfers 1x45°
undimensioned radii R1
material: EN AW-AlCu4PbMg
scale xx
blank dimensions: Ø / Ø1
3+2 EXERCISE 2 M DIN
programGuide Siemens Operate
4
A3
General tolerances
ISO 2768-m
Name
Datum
100
50
70
100
50
50
Rz 16


+0,4
+0,2
-0,2
-0,4
Rz 63
Rz 16
100
35,264°
45°
C
C
C-C (1 : 1,5)
50
ø40
15
15°
E (1 : 1)
ø10
20,41
25
E
undimensioned chamfers 1x45°
undimensioned radii R1
material: EN AW-AlCu4PbMg
scale 1:1,5
blank dimensions: Ø / FL
Datum
Name
3+2 EXERCISE 2 M DIN
General tolerances
ISO 2768-m
programGuide Siemens Operate
3
A3

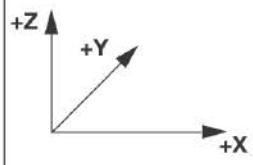

+0,4
+0,2
-0,2
-0,4
B-B (1 : 1)
Rz 63
Rz 16
( )
7°
7°
70
M10
20
15
100
50
60
24
51,76
B
B
undimensioned chamfers 1x45°
undimensioned radii R1
material: EN AW-AlCu4PbMg
scale x:x
blank dimensions: Ø / Fl
Datum
Name
3+2 EXERCISE 2 M DIN
General tolerances
ISO 2768-m
programGuide Siemens Operate
3
A3

b. The following tools and technology data are required for programming:

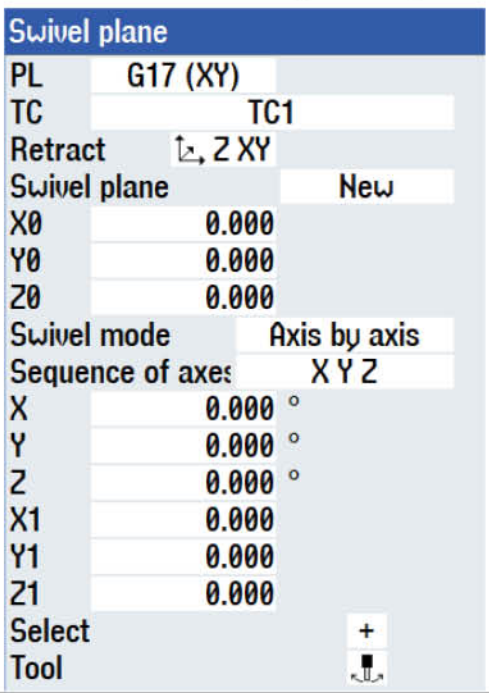
programGuide Siemens Operate VIETNAM 2021/2022		
Tool shape file	CNC Milling	
Work piece: 3+2 EXERCISE 2M DIN	Blank material: EN AW-AlCu4PbMg	Machine configuration:
Drawing: 1(1)	Blank dimensions: Fl 100 x 100 x 52	Milling 5 axes (X Y Z A C)

		Technological data						
Tool - Nr.	T1	T2	T3	T4	T5	T6	T7	T8
Tool Ø (mm)	12 x 90°	10 x 90°	20	16	16	8	8	
Cutting speed (m/min)	40	105	150 / 175	150	175	150	175	
Cutting depth a_p = max.	5	2	5	8	8	6	6	
Cutting material	HSS	SC	SC	SC	SC	SC	SC	
Feed tooth fz / rev. (mm)	0,1	0,03	0,1 / 0,08	0,06	0,05	0,03	0,02	
Number of teeth N		4	2	4	4	4	4	
								
				✓	✓	✓	✓	✓

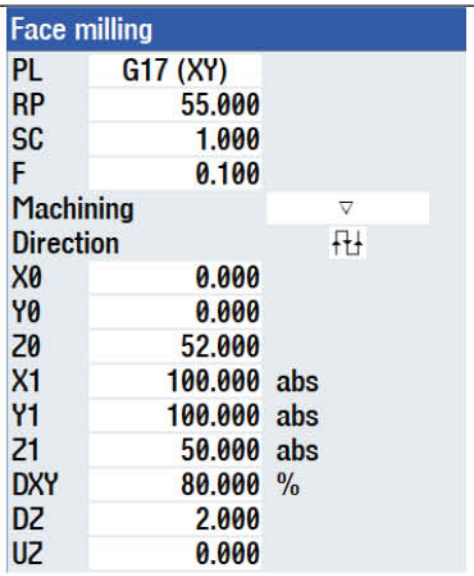
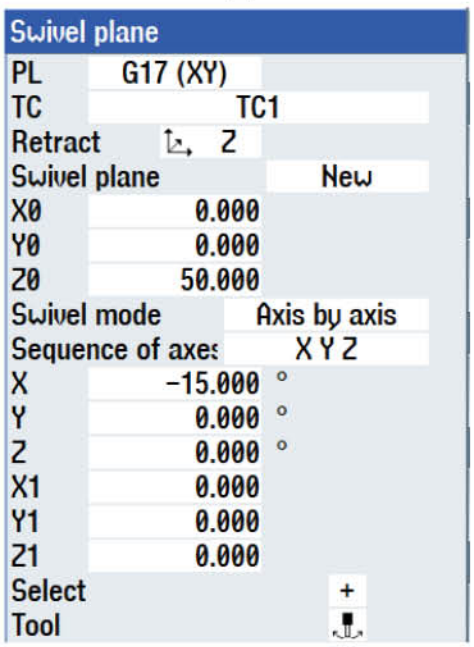
		Technological data						
Tool - Nr.	T9	T10	T11	T12	T13	T14	T15	T16
Tool Ø (mm)						8,5	M10	9,8
Cutting speed (m/min)						40	5	40
Cutting depth a_p =								

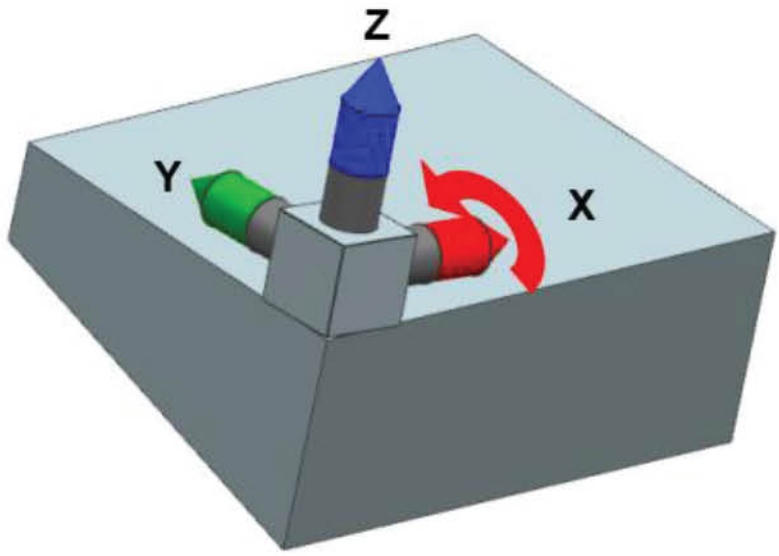
max.								
Cutting material						HSS	HSS	HSS
Feed tooth fz / rev. (mm)						0,1	F = P	0,1
Number of teeth N								
								

c. Create a new program, for example "3+2 EXERCISE_2M.MPF".

1.	Program the first lines of the program by inserting the following G code commands: **N1 G17 G40 G71 G90**
2. 	Insert now the basic values for swivelling into the program. Press the HSK 1.6 “Various” to open the work area “Swivel plane”. Press the HSK 1.2 “Swivel plane”. The following input mask for “Swivel plane” opens.
3. 	Press HSK 1.3 “Basic values” and following values/parameters are displayed: 

Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N2 CYCLE800(2,"TC1",200000,57,0,0,0,0,0,0,0,0,1,,1)**
4. Swivel plane	As next deactivate the name of the swivel data set with “T0”. **Swivel plane** TC 0
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N3 CYCLE800()**
5.	Enter following values/parameters for the blank: **Blank input** Blank / Block X0 0.000 Y0 0.000 X1 100.000 abs Y1 100.000 abs ZA 0.000 ZI 52.000 abs
Accept	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N4 WORKPIECE(,"",,"BOX",112,0,52,-80,0,0,100,100)**
6.	Insert the following technology commands into the program: **N5 G54** **N6 ;*** **N7 MSG("FACE MILLING INITIAL SETTING")** **N8 T="CUTTER 20"** **N9 M6** **N10 G97 S2387 M3** **N11 G95 FZ=0.1 M8**
7.	Face milling height 50 mm:

Accept	 **N13 CYCLE61(55,52,1,50,0,0,100,100,2,80,0,0.1,41,0,1,11010)**
8.	Insert the following information commands into the program: **N14 ;*** ** **N15 MSG("ROUGHING OBLIQUE SURFACE 15°")**
9.	Swivel the plane to 15° in the "SWIVEL_PLANE" block.  **N16 CYCLE800(1,"TC1",200000,57,0,0,50,-15,0,0,0,0,0,1,,1)**
Accept	
The zero point for the next operation:	



1 Adjacent side (AK)

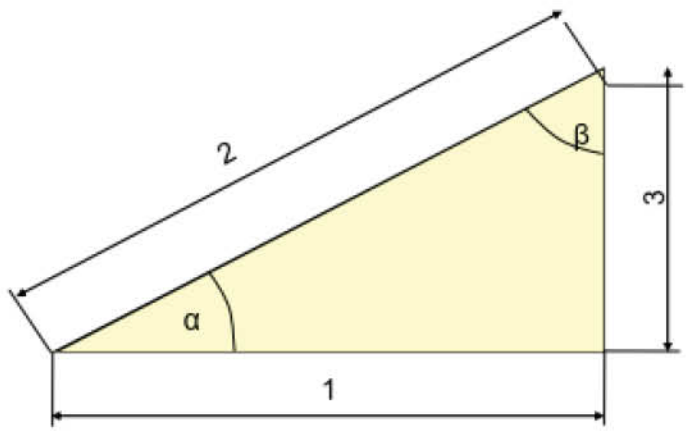
2 Hypotenuse (H)

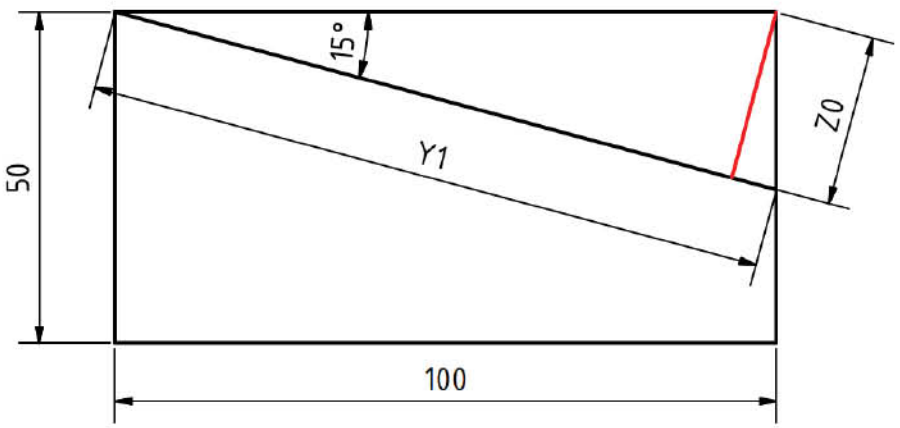
3 Opposite side (GK)

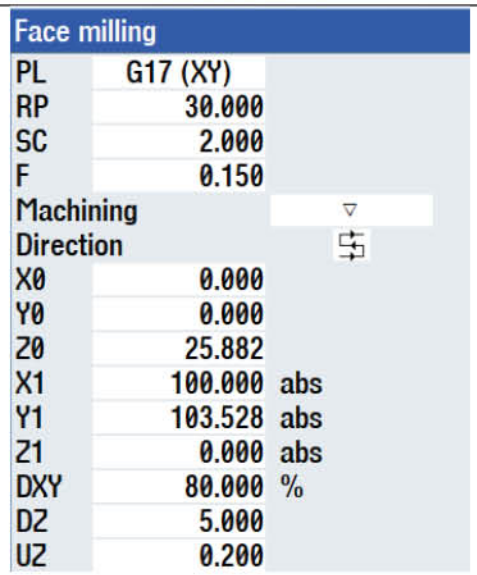
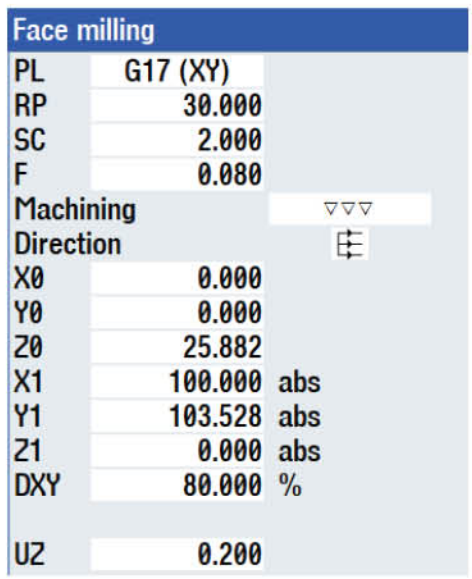
α angle

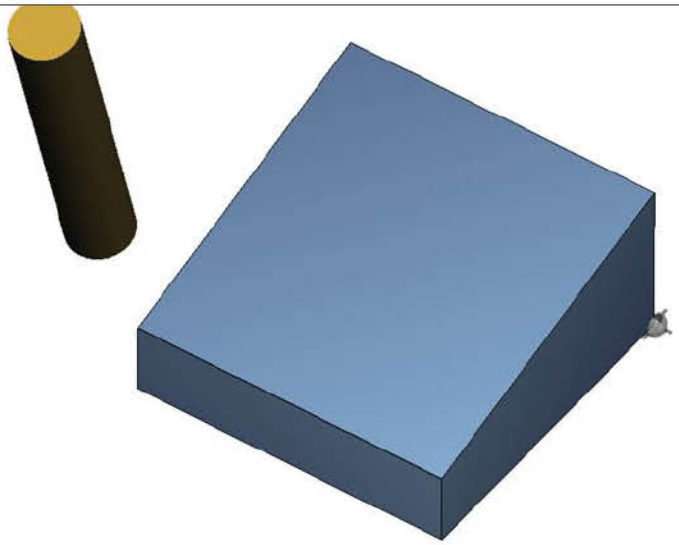
β angle

Preview for the next program sequence: The values for the traversing distances must now be calculated with the trigonometric functions.
In order to program the start position in "Z" coordinate it is often necessary to calculate the height of for traversing distances with rapid speed.

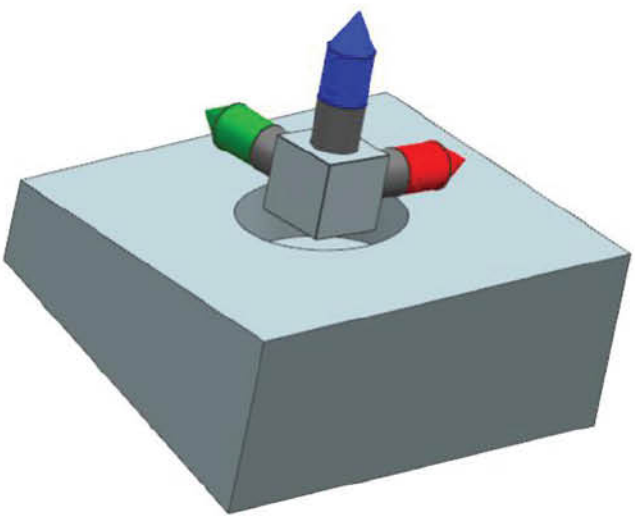


	$\sin\alpha = \frac{GK}{H}$ $\quad H = \frac{GK}{\sin\alpha}$ $\quad GK = \sin\alpha * H$ $\cos\alpha = \frac{AK}{H}$ $\quad H = \frac{AK}{\cos\alpha}$ $\quad AK = \cos\alpha * H$ $\tan\alpha = \frac{GK}{AK}$ $\quad AK = \frac{GK}{\tan\alpha}$ $\quad GK = \tan\alpha * AK$
	This results in the following calculation for the current example for the determination of "Z0" and "Y1" for the face cutting cycle:  sinα = GK/HP $\quad$ sin15° = GK / 100 GK = sin15° * 100 $\quad$ GK = 25,882 mm GK =Z0 cosα = AK / HP $\quad$ cos15° = 100 / HP HP = 100 / cos15° $\quad$ HP = 103,528 mm HP = Y1
10.	The face cutting for roughing in the swivel plane with 15° follows.

Accept	 **N17 CYCLE61(30,25.882,2,0,0,0,100,103.528,5,80,0.2,0.15,31,0,1,11010)**
11.	Insert the following technology commands into the program: **N18 MSG("FINISHING OBLIQUE SURFACE 15°")** **N19 G97 S2785 M3** **N20 G95 FZ=0.08**
12. Accept	The face cutting for finishing in the swivel plane with 15° follows.  **N21 CYCLE61(30,25.882,2,0,0,0,100,103.528,2,80,0.2,0.08,12,0,1,11010)**



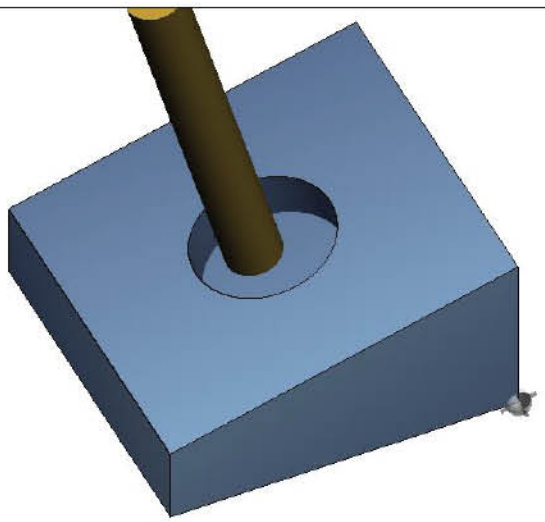
13.	Insert the following technology commands into the program: **N22 G0 Z150 M9** **N23 MSG("MILLING CENTRED CIRCULAR POCKET Ø40")** **N24 T="CUTTER 16"** **N25 M6** **N26 S2984 M3 FZ=0.06** **N27 G0 X50 Y51.76** **N28 Z5 M8**

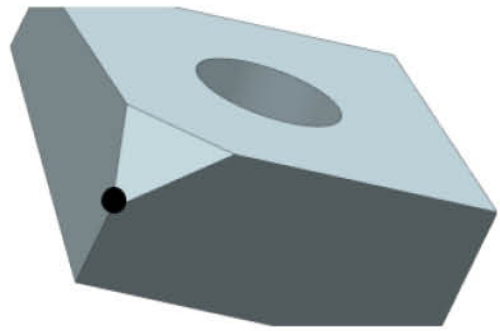
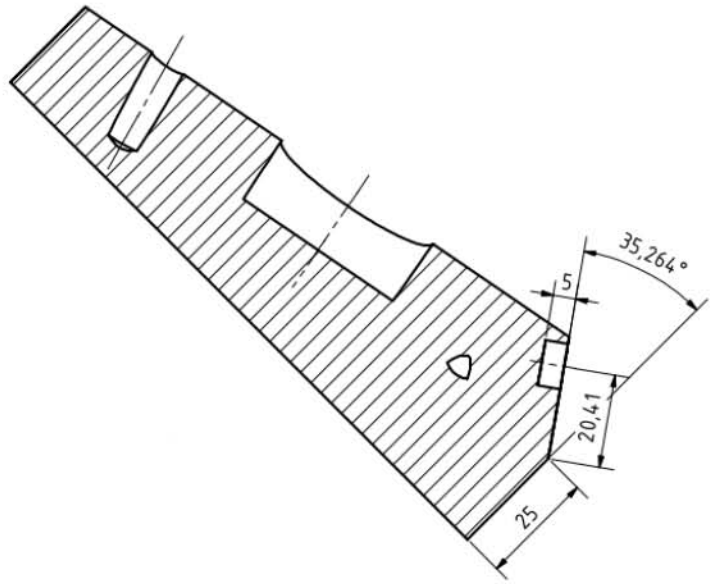
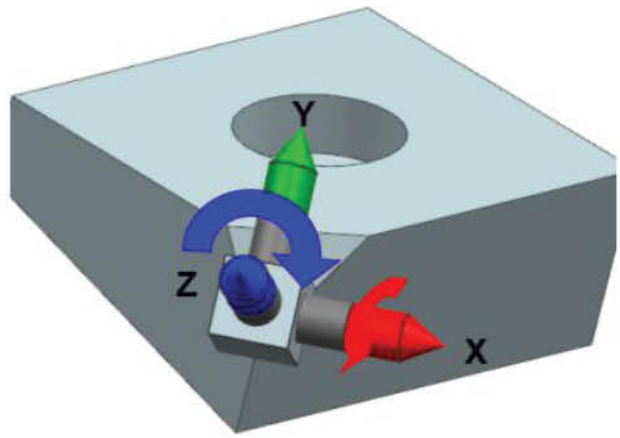


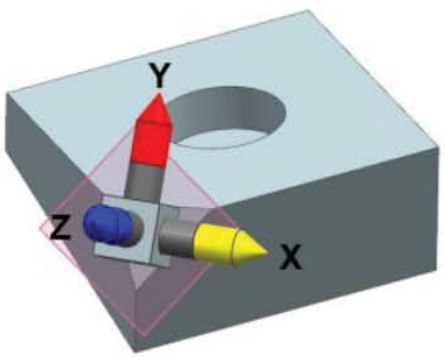
The tool preposition for the next operation:

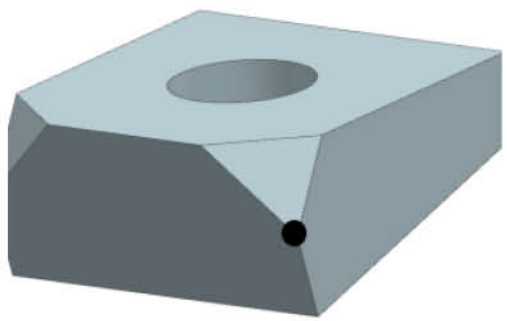
14.	The circular pocket for roughing in the swivel plane with 15° follows.

<table>
<tr>
<td>Accept</td>
<td>
<table>
<tr><td colspan="3">Circular pocket</td></tr>
<tr><td>Input</td><td></td><td>Complete</td></tr>
<tr><td>PL</td><td>G17 (XY)</td><td>Down-cut</td></tr>
<tr><td>RP</td><td>5.000</td><td></td></tr>
<tr><td>SC</td><td>2.000</td><td></td></tr>
<tr><td>F</td><td>0.060</td><td></td></tr>
<tr><td>Machining</td><td></td><td>▽</td></tr>
<tr><td></td><td>Centric</td><td></td></tr>
<tr><td></td><td>Single position</td><td></td></tr>
<tr><td>X0</td><td>50.000</td><td></td></tr>
<tr><td>Y0</td><td>51.760</td><td></td></tr>
<tr><td>Z0</td><td>0.000</td><td></td></tr>
<tr><td>⌀</td><td>40.000</td><td></td></tr>
<tr><td>Z1</td><td>-15.000</td><td>abs</td></tr>
<tr><td>DXY</td><td>50.000</td><td>%</td></tr>
<tr><td>DZ</td><td>5.000</td><td></td></tr>
<tr><td>UXY</td><td>0.200</td><td></td></tr>
<tr><td>UZ</td><td>0.100</td><td></td></tr>
<tr><td>Insertion</td><td></td><td>Helical</td></tr>
<tr><td>EP</td><td>2.500</td><td></td></tr>
<tr><td>ER</td><td>7.000</td><td></td></tr>
<tr><td>Removing</td><td></td><td>Comp. machining</td></tr>
</table>
N29 POCKET4(5,0,2,-15,40,50,51.76,5,0.2,0.1,0.06,2000,0,21,50,9,15,7,2.5,0,1,2,10100,111,110)
</td>
</tr>
<tr>
<td>15.</td>
<td>Insert the following technology commands into the program:
N30 G0 Z150 M9
N31 T="CUTTER 16 FIN"
N32 M6
N33 S3482 FZ=0.05
N34 G0 X50 Y51.76
N35 Z5 M8</td>
</tr>
<tr>
<td>16.</td>
<td>The circular pocket for finishing in the swivel plane with 15° follows.</td>
</tr>
</table>

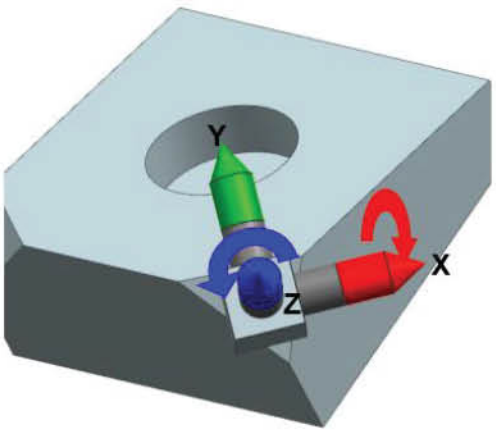
Accept	**Circular pocket** Input: Complete PL: G17 (XY) — Down-cut RP: 5.000 SC: 2.000 F: 0.050 Machining: ▽▽▽ Centric Single position X0: 50.000 Y0: 51.760 Z0: 0.000 ⌀: 40.000 Z1: -15.000 abs DXY: 75.000 % DZ: 15.000 UXY: 0.200 UZ: 0.100 Insertion: Vertical FZ: 0.050 **N36 POCKET4(5,0,2,-15,40,50,51.76,15,0.2,0.1,0.05,0.05,0,12,75,9,15,0,2,0,1,2,10100,111,110)**
	
17.	Insert the following technology commands into the program: **N37 G0 Z150 M9** **N38** **N39 MSG("MILL 1st CORNER RH+FIN")** **N40 T="CUTTER 16 RH"** **N41 M6** **N42 S2984 M3 FZ=0.06 M8**
18.	The oblique surface is programmed in the next step. To do this, a swivel is made to a new machining plane.

<table>
<tr><td></td><td> 
 
</td></tr>
<tr><td colspan="2">The zero point for the next operation:
 
</td></tr>
<tr><td>19.</td><td>Swivel the plane for the 1st corner in the "SWIVEL_PLANE" block.
The degree value for "X" must be calculated in this way: 90°- 35,264° = 54,736</td></tr>
</table>

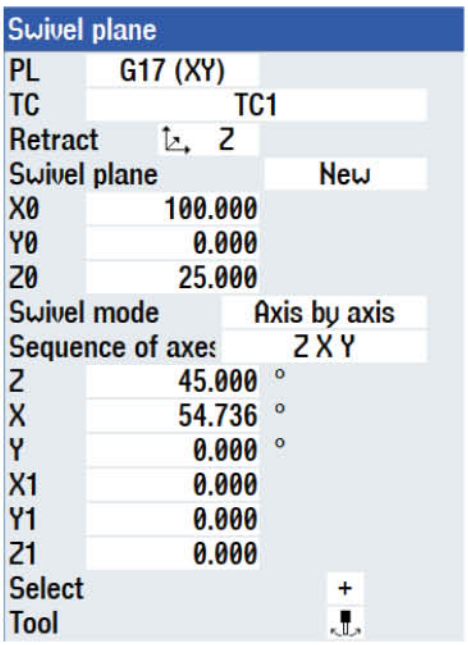
Accept	Swivel plane PL G17 (XY) TC TC1 Retract Z Swivel plane New X0 0.000 Y0 0.000 Z0 25.000 Swivel mode Axis by axis Sequence of axes Z X Y Z -45.000 ° X 54.736 ° Y 0.000 ° X1 0.000 Y1 0.000 Z1 0.000 Select + Tool **N43 CYCLE800(1,"TC1",200000,39,0,0,25,-45,54.736,0,0,0,0,1,,1)**
20.	The first corner is programmed with the face milling cycle in the "MILLING_FIRST_CORNER" block. The parameter "Z0" must be calculated. sin α = GK / HP GK = sin 35,264 * 25 GK = 14,4336 The values "X0", "X1", "Y0" and "Y1" specify the dimensions of the milling plane. 

Accept	Face milling PL G17 (XY) RP 17.000 SC 1.000 F 0.060 Machining Direction X0 -20.000 Y0 0.000 Z0 14.434 X1 40.000 inc Y1 32.000 abs Z1 0.000 abs DXY 80.000 % DZ 3.000 UZ 0.000 **N44 CYCLE61(17,14.434,1,0,-20,0,40,32,3,80,0,0.06,31,0,1,10010)**
21.	Insert the following technology commands into the program: **N45** **N46 MSG("MILL 2nd CORNER RH+FIN")**
22.	This is programmed in the "MILLING_SECOND_CORNER" block. The coordinate system is programmed accordingly starting at this point. The degree value for "X" must be calculated again. 90°- 35,264° = 54,736 

The zero point for the next operation:



23. Swivel the plane for the 2nd corner in the "SWIVEL_PLANE" block.



Accept

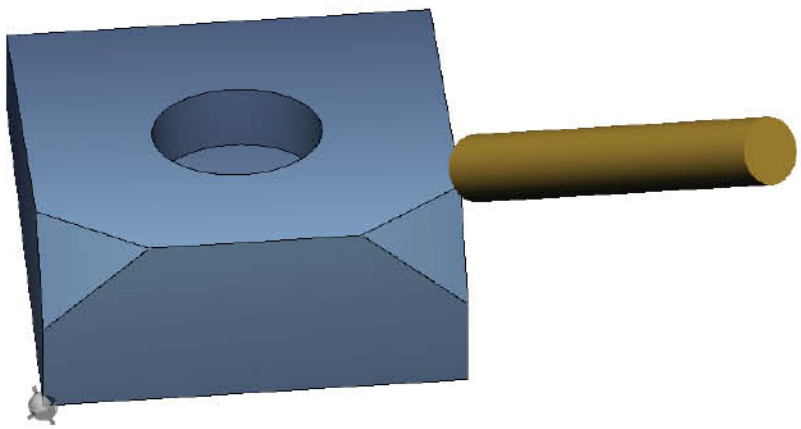
N47 CYCLE800(1,"TC1",200000,39,100,0,25,45,54.736,0,0,0,0,1,,1)

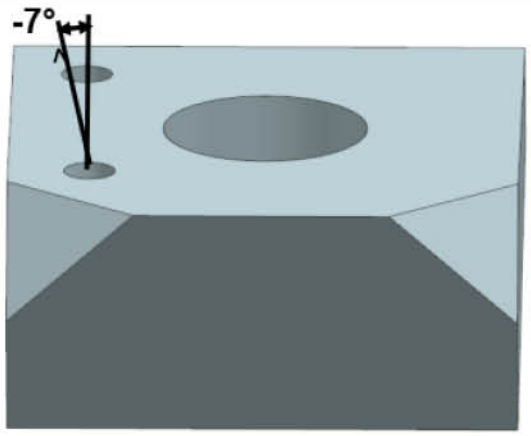
24. The inputs for the face milling are then made. Parameter "Z0" must be calculated like before with the 1st corner.

sin α = GK / HP

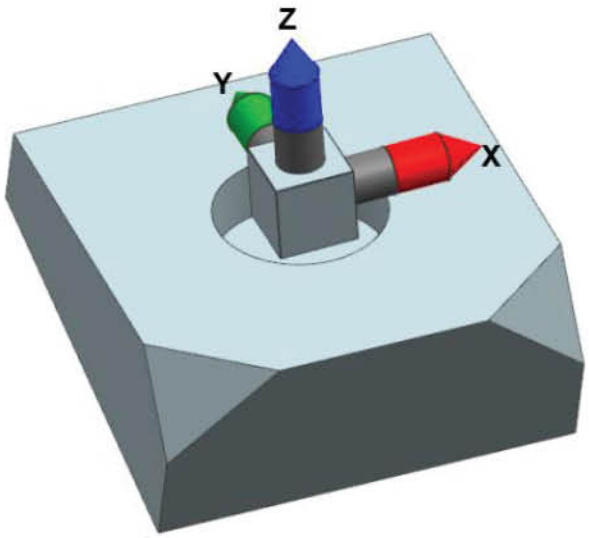
GK = sin 35,264 * 25

GK = 14,4336

	Face milling PL G17 (XY) RP 17.000 SC 1.000 F 0.060 Machining Direction X0 -20.000 Y0 0.000 Z0 14.434 X1 40.000 inc Y1 32.000 inc Z1 0.000 abs DXY 80.000 % DZ 3.000 UZ 0.000 **N48 CYCLE61(17,14.434,1,0,-20,0,40,32,3,80,0,0.06,31,0,1,10)**
	
25.	Insert the following technology commands into the program: **N50 MSG("SPOT DRILLING Ø10.8")** **N51 T="CENTERDRILL 12"** **N52 M6** **N53 S1061 M3 F0.12**
26.	The spot drill holes (Ø10.8) in the 15° oblique surface is programmed in the next step. To do this, an additional swivel is made to a new machining plane by 7° in minus and later in the program in plus direction.



The zero point for the next operation:



27.

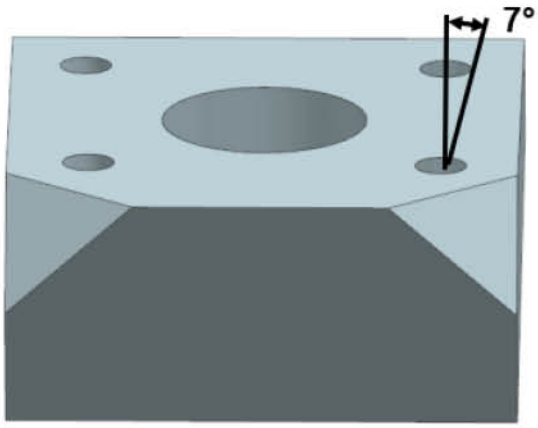
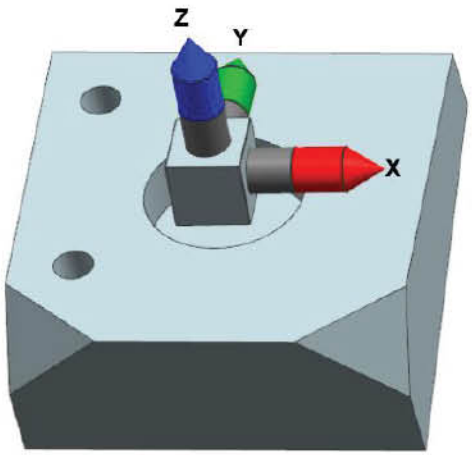
First the “New” swivel plane with 15° is programmed.

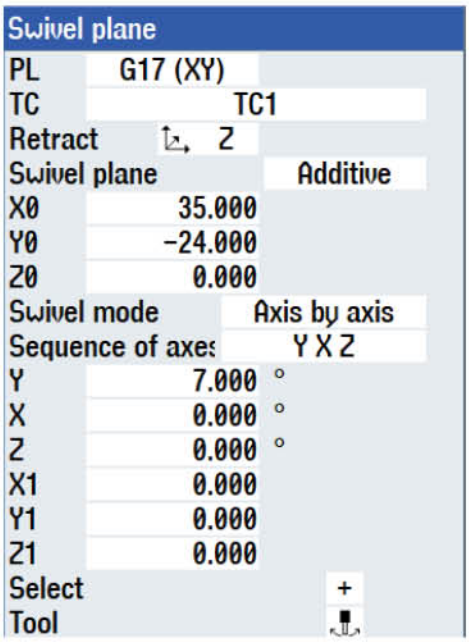
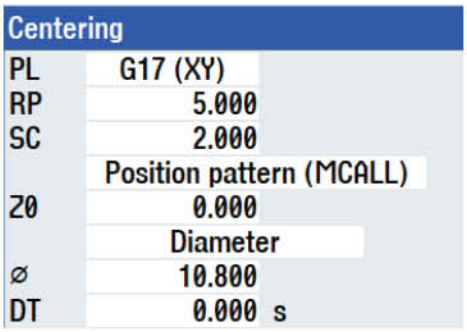
Swivel plane		
PL	G17 (XY)	
TC	TC1	
Retract	Z	
Swivel plane	New	
X0	0.000	
Y0	0.000	
Z0	50.000	
Swivel mode	Axis by axis	
Sequence of axes	X Y Z	
X	-15.000	°
Y	0.000	°
Z	0.000	°
X1	50.000	
Y1	51.760	
Z1	0.000	
Select		+
Tool		

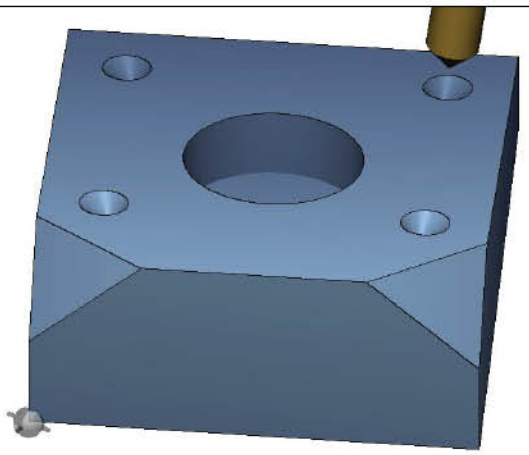
Accept

N54 CYCLE800(1,"TC1",200000,57,0,0,50,-15,0,0,50,51.76,0,1,,1)

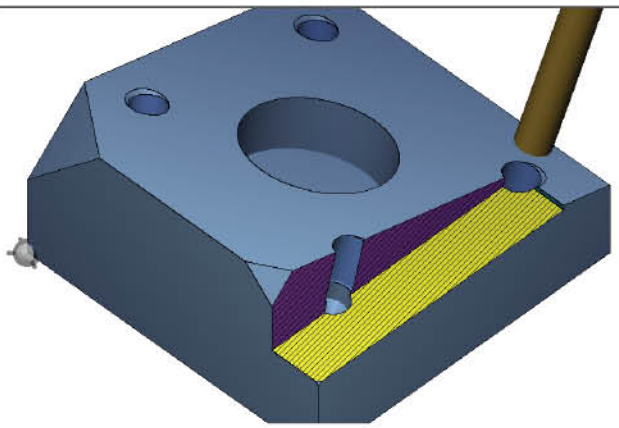
28. Accept	Second the “Additive” swivel plane for the -7° drills are programmed for the first spot drill hole. **Swivel plane** PL G17 (XY) TC TC1 Retract Z Swivel plane Additive X0 -35.000 Y0 -24.000 Z0 0.000 Swivel mode Axis by axis Sequence of axes Y X Z Y -7.000 ° X 0.000 ° Z 0.000 ° X1 0.000 Y1 0.000 Z1 0.000 Select + Tool **N55 CYCLE800(1,"TC1",200001,54,-35,-24,0,-7,0,0,0,0,0,1,,1)**
29. Accept	The inputs for the centering are then made. **Centering** PL G17 (XY) RP 5.000 SC 2.000 Position pattern (MCALL) Z0 0.000 Diameter ⌀ 10.800 DT 0.000 s **N56 MCALL CYCLE81(20,0,5,10.8,,0,10,1,12)**
30.	Insert the following G code commands into the program for the second centering position. **N57 G0 X0 Y0** **N58 G0 Y60** **N59 MCALL**

31.	Now the next centerings by +7° are programmened. 
The zero point for the next operation: 	
32. Accept	First the “New” swivel plane with 15° is programmed. **Swivel plane** PL G17 (XY) TC TC1 Retract Z Swivel plane New X0 0.000 Y0 0.000 Z0 50.000 Swivel mode Axis by axis Sequence of axes X Y Z X -15.000 ° Y 0.000 ° Z 0.000 ° X1 50.000 Y1 51.760 Z1 0.000 Select + Tool **N60 MSG("+7°")** **N61 CYCLE800(1,"TC1",200000,57,0,0,50,-15,0,0,50,51.76,0,1,,1)**

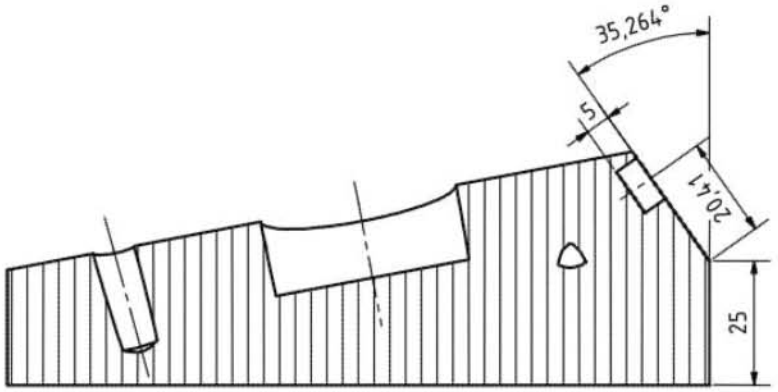
33. 	Second the “Additive” swivel plane for the +7° drills are programmed for the first spot drill hole.  **N62 CYCLE800(1,"TC1",200001,54,35,-24,0,7,0,0,0,0,0,1,,1)**
34. 	The inputs for the centering are then made.  **N56 MCALL CYCLE81(20,0,5,10.8,,0,10,1,12)**
35.	Insert the following G code commands into the program for the second centering position. **N64 G0 X0 Y0** **N65 G0 Y60** **N66 MCALL** **N67 G0 Z150**

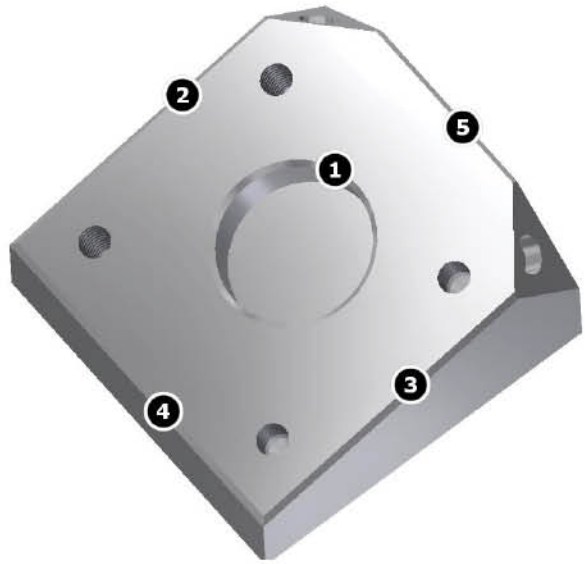


36.	Task: Now program the drilling of the core holes and the M10 threaded holes for all four positions yourself.



37.	Task: Now independently program the circular pockets for the two positions.



38.	Task: Now program 1,5 mm chamfers independently for the five contour elements displayed. 
39.	Task: Now program 0,4 mm chamfers independently for the two corner pockets displayed.
40.	Finally quit the program: **N... CYCLE800(1,"TC1",200000,57,0,0,0,0,0,0,0,0,0,1,,1)** **N... CYCLE800()** **N... T0 M6** **N... M30**

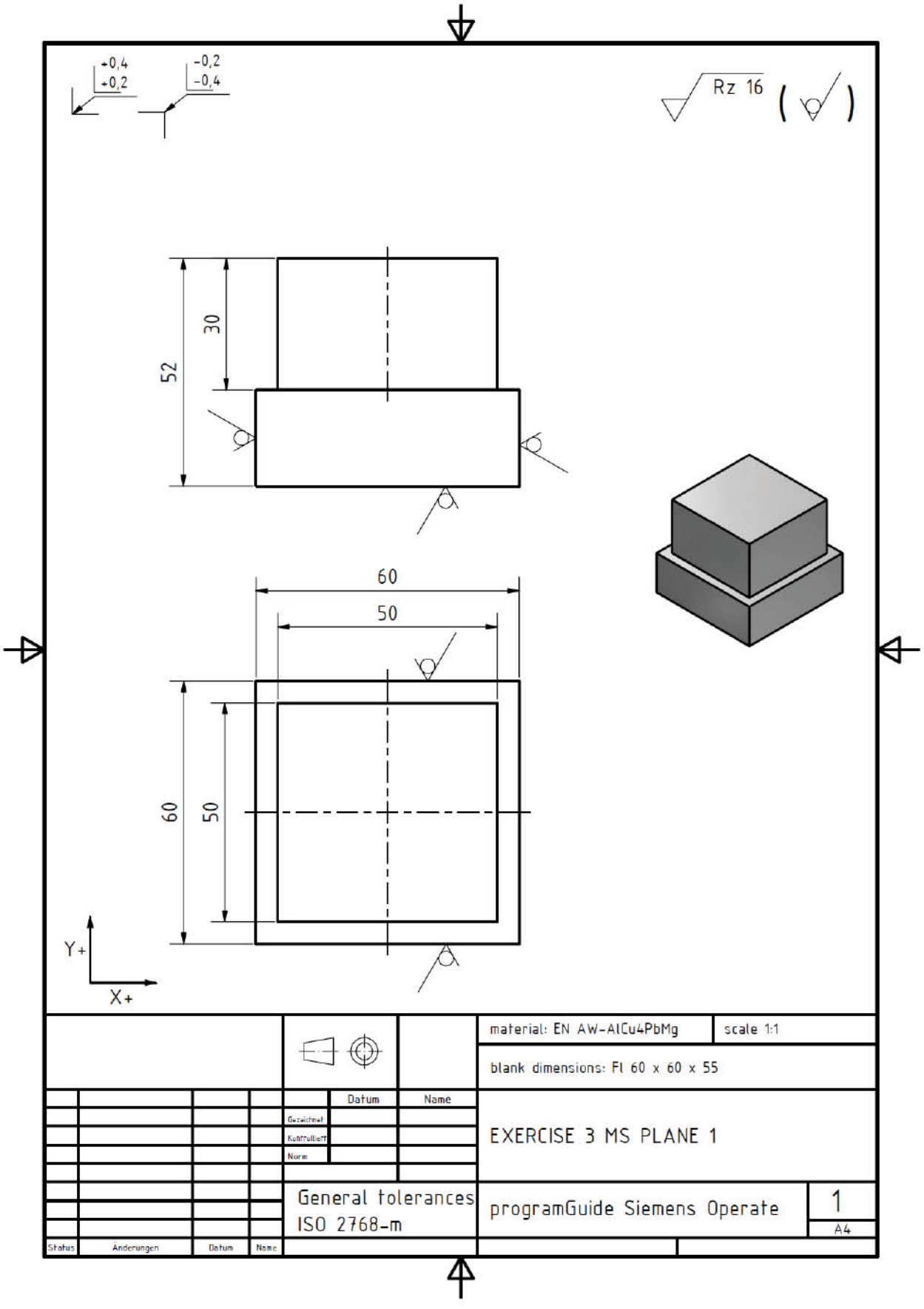
3.8. Plan and realise operational work order of the industry

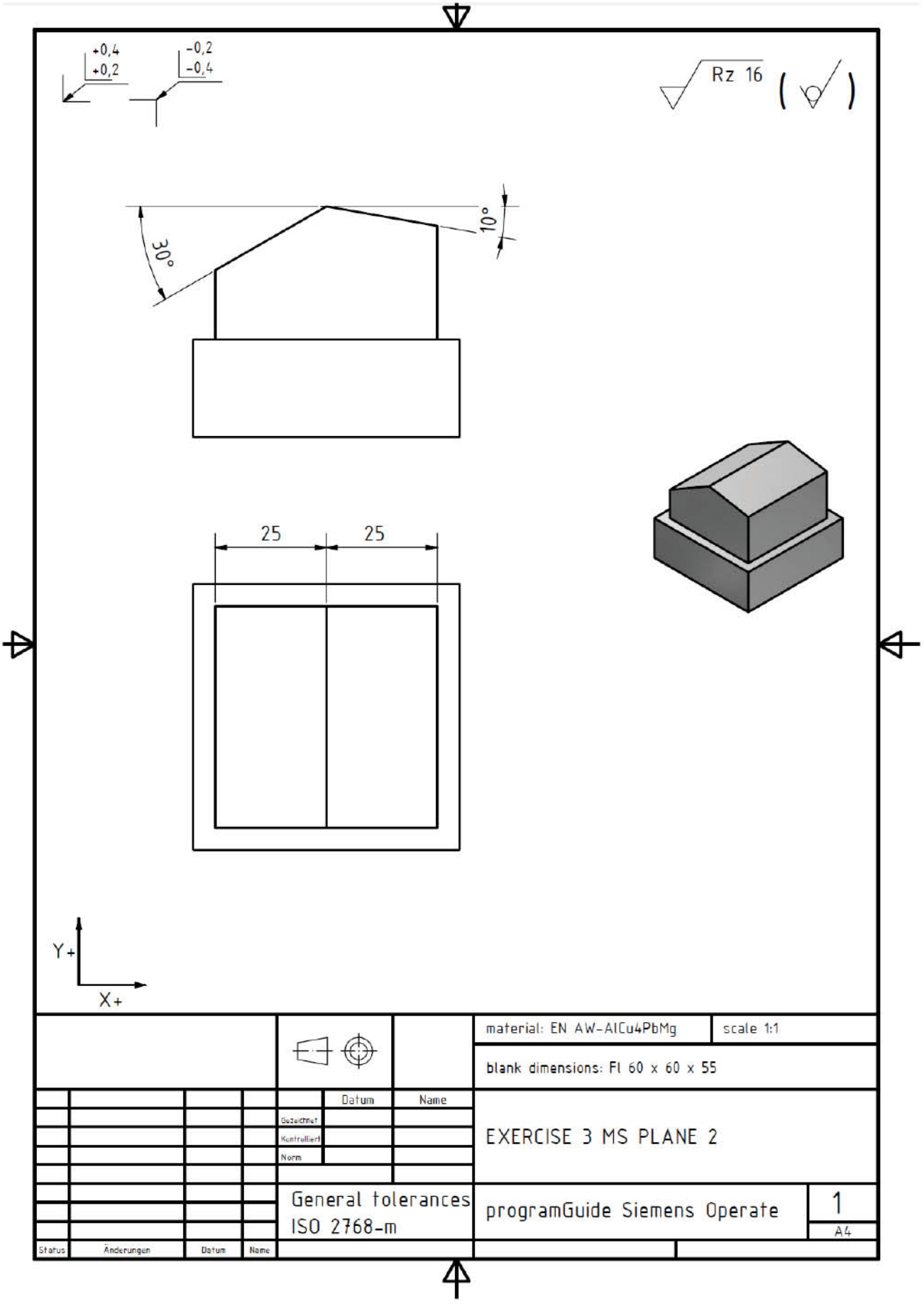
3.8.1. Using programGUIDE with swivelled planes, the cycles and G-codes for advanced milling application 1 with 3+2 axes

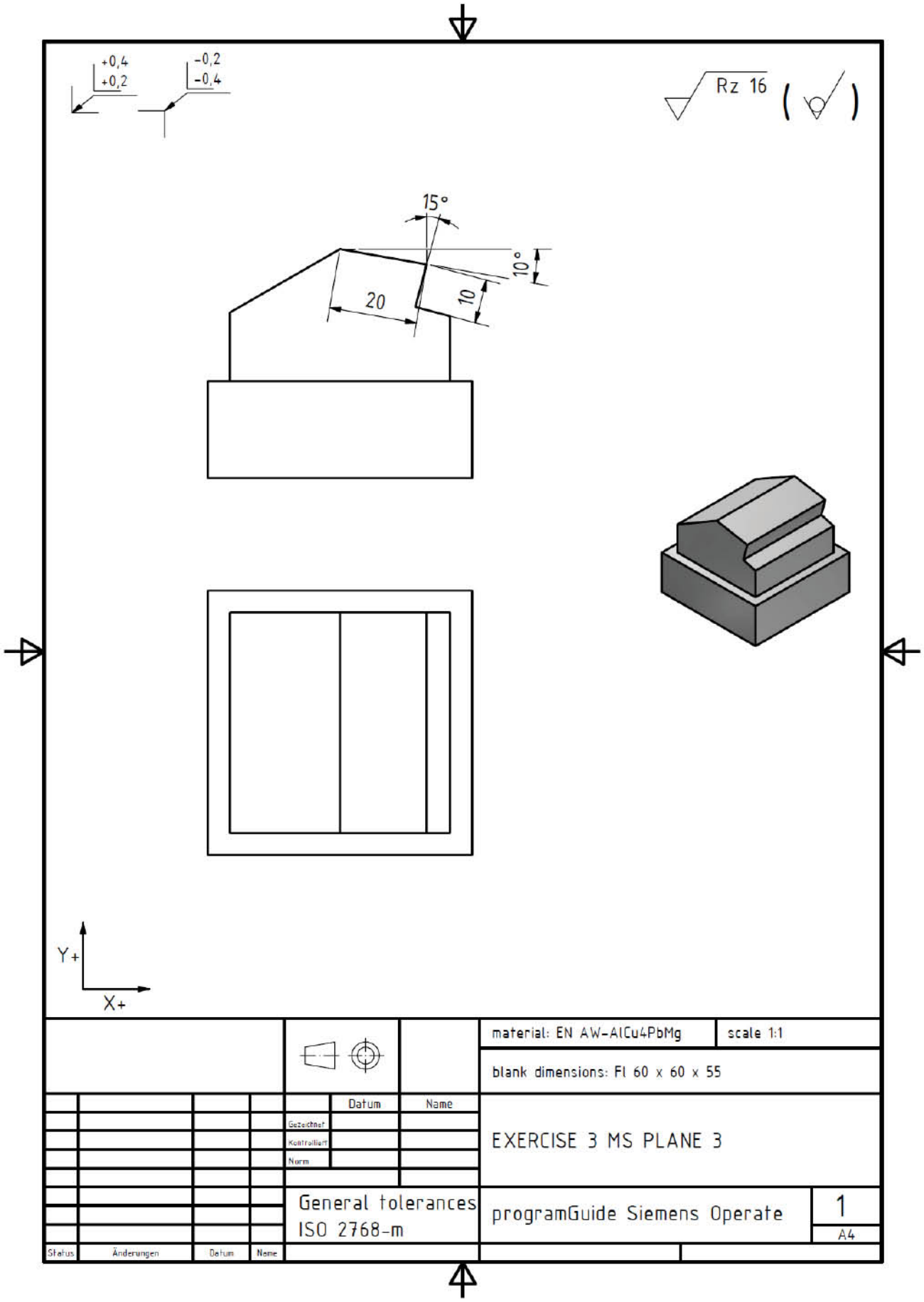
a. Objective:

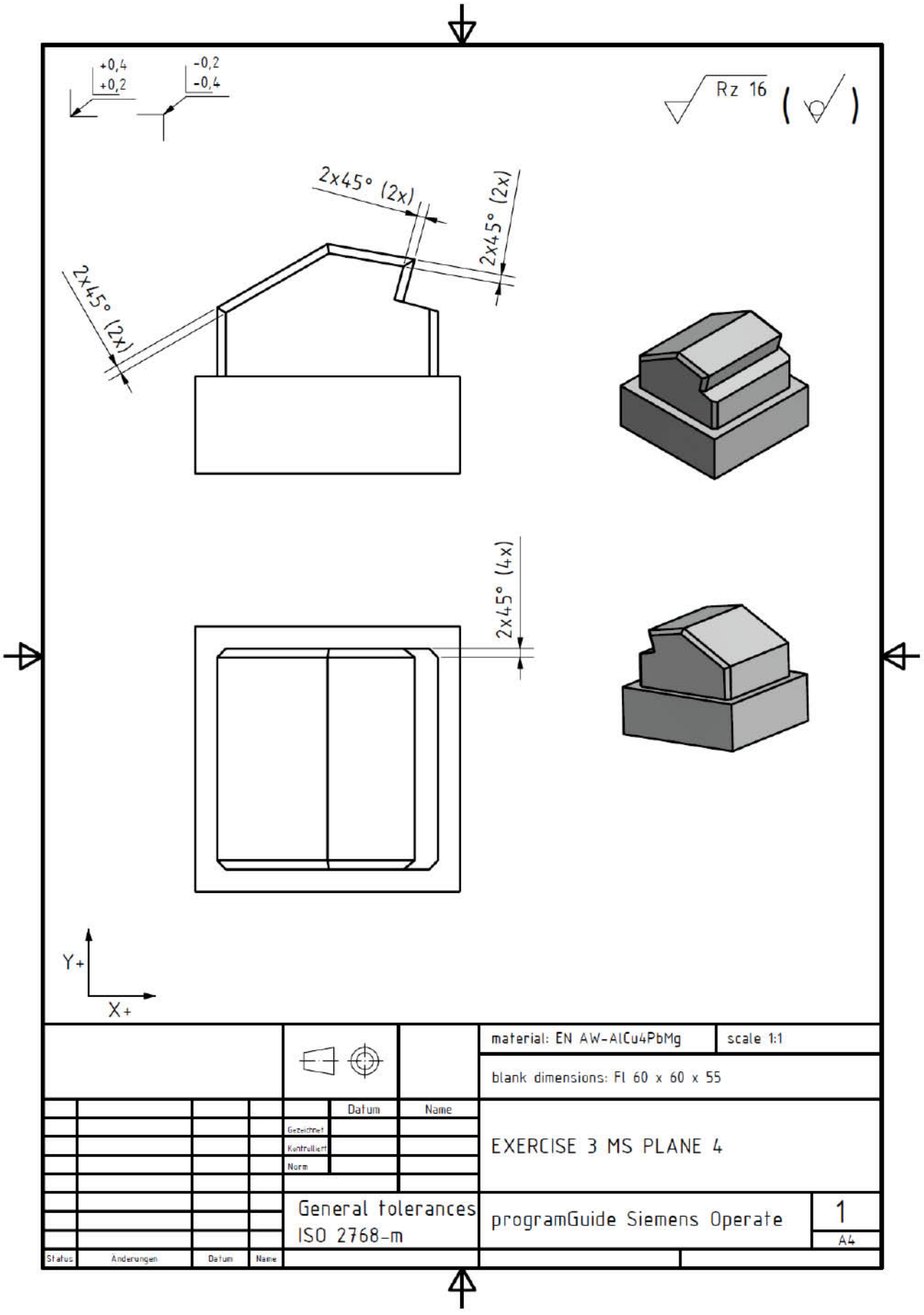
- Complete work plan ((refer template at page 252)
- Program to machine the relevant application using the appropriate cycles and G-codes

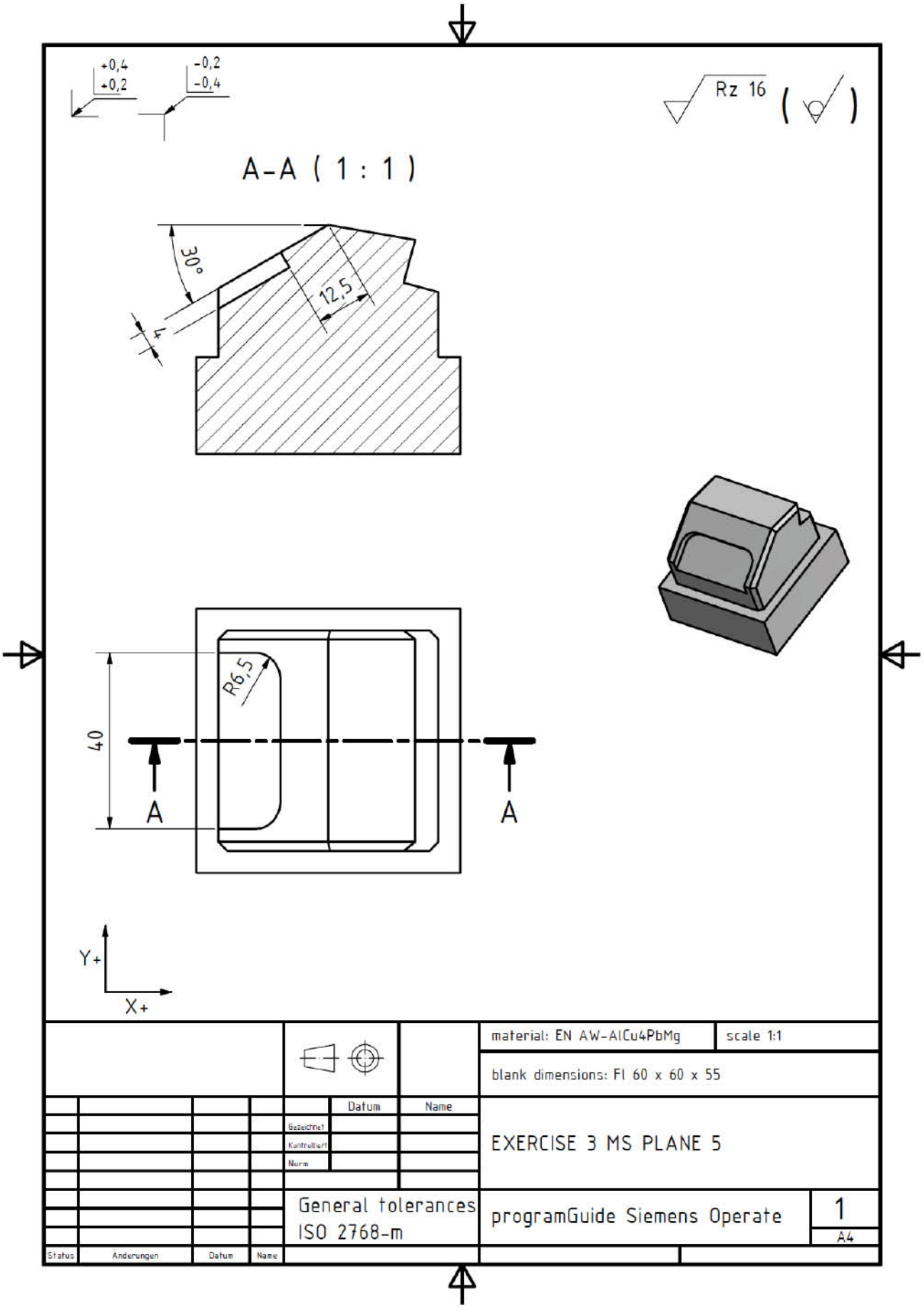
b. Technical drawings




+0,4
+0,2
-0,2
-0,4
Rz 16
30°
10°
25
25
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Fl 60 x 60 x 55
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE 3 MS PLANE 2
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name


+0,4
+0,2
-0,2
-0,4
Rz 16
15°
10°
20
10
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Fl 60 x 60 x 55
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE 3 MS PLANE 3
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name


+0,4
+0,2
-0,2
-0,4
Rz 16
2x45° (2x)
2x45° (2x)
2x45° (2x)
2x45° (4x)
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Fl 60 x 60 x 55
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE 3 MS PLANE 4
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name

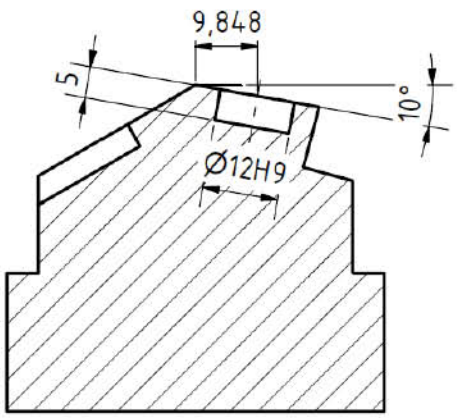

+0,4
+0,2
-0,2
-0,4
Rz 16
A-A (1 : 1)
30°
12,5
4
R6,5
40
A
A
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: FI 60 x 60 x 55
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE 3 MS PLANE 5
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name

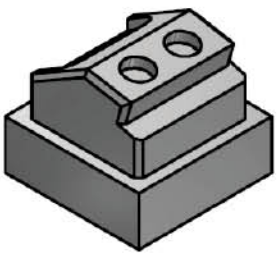
+0,4
+0,2

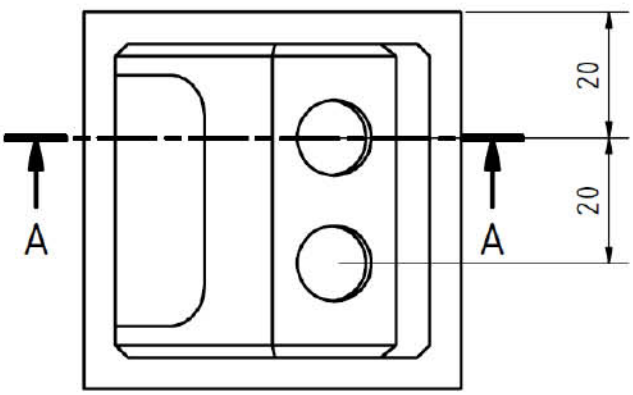
-0,2
-0,4

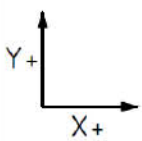
Rz 16

A-A (1 : 1)









material: EN AW-AlCu4PbMg		scale 1:1	
blank dimensions: Fl 60 x 60 x 55			
	Datum	Name	
Gezeichnet			EXERCISE 3 MS PLANE 6
Kontrolliert			
Norm			
General tolerances ISO 2768-m		programGuide Siemens Operate	1 A4

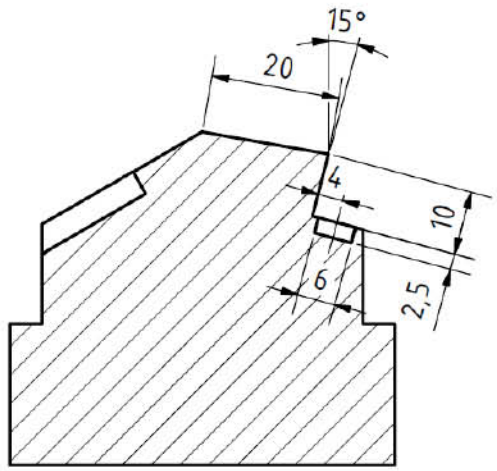
Status	Änderungen	Datum	Name

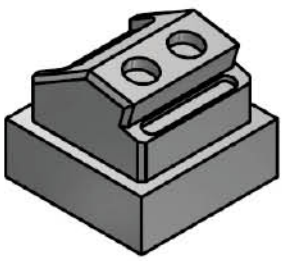
+0,4
+0,2

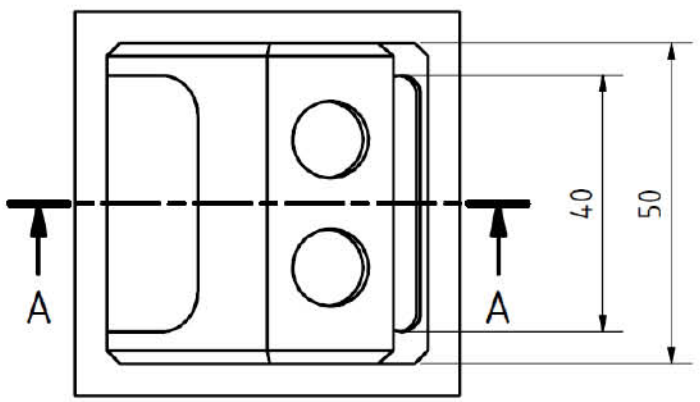
-0,2
-0,4

Rz 16

A-A (1 : 1)





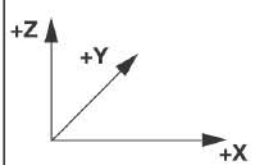


Y+
X+

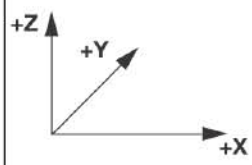
					Datum	Name	material: EN AW-AlCu4PbMg
							scale 1:1
							blank dimensions: Fl 60 x 60 x 55
				Gezeichnet			
				Kontrolliert			EXERCISE 3 MS PLANE 7
				Norm			
				General tolerances ISO 2768-m			programGuide Siemens Operate
							1
							A4
Status	Änderungen	Datum	Name				

c. The following tools and technology data are required for programming

programGuide Siemens Operate VIETNAM 2021/2022		
Tool shape file	CNC Milling	
Work piece: PLANE 1-8 DIN	Blank material: EN AW-AlCu4PbMg	Machine configuration:
Drawing: 1(1)	Blank dimensions: Fl 60 x 60 x 52	Milling 5 axes (X Y Z A C)

	Technological data							
Tool - Nr.	T1	T2	T3	T4	T5	T6	T7	T8
Tool Ø (mm)	12 x 90°	6 x 90°	20	16	16	10	10	
Cutting speed (m/min)	60	105	150 / 175	150	175	150	175	
Cutting depth a_p = max.	6	3	5	8	8	6	6	
Cutting material	HSS	SC	SC	SC	SC	SC	SC	
Feed tooth fz / rev. (mm)	0,1	0,03	0,1 / 0,08	0,06	0,05	0,05	0,03	
Number of teeth N		4	2	4	4	4	4	
								
				✓	✓	✓	✓	

	Technological data							
Tool - Nr.	T9	T10	T11	T12	T13	T14	T15	T16
Tool Ø (mm)	8	8	4			5	M6	4 x 90°
Cutting speed (m/min)	150	175	150 / 175			60	15	105

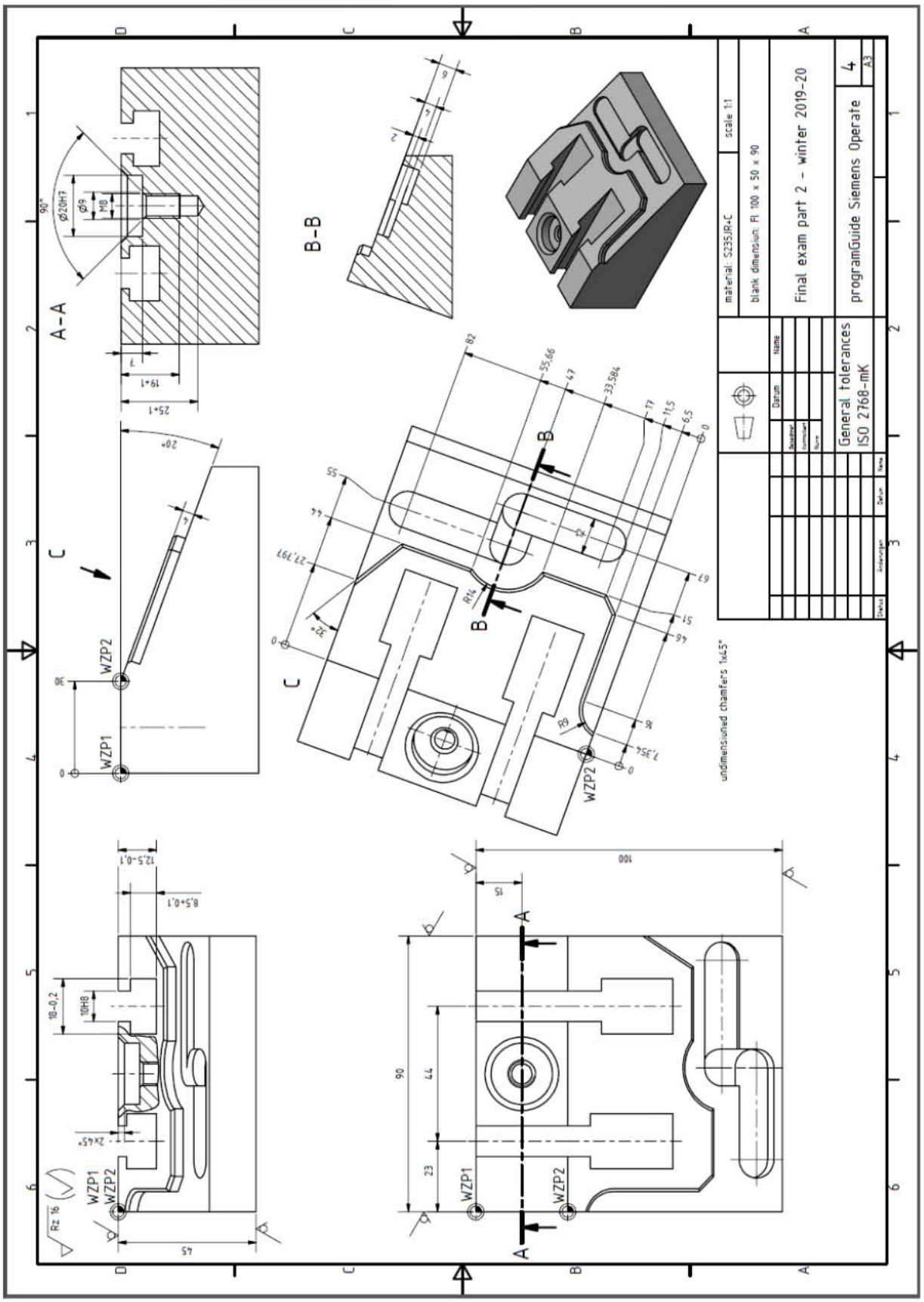
Cutting depth a_p = max.	5	5	6					2
Cutting material	SC	SC	SC			HSS	HSS	SC
Feed tooth fz / rev. (mm)	0,03	0,02	0,01			0,05	F = P	0,01
Number of teeth N	4	4	4					4
								
	✓	✓	✓					

3.8.2. Using programGUIDE with swivelled planes, the cycles and G-codes for advanced milling application 2 with 3+2 axes

a. Objective:

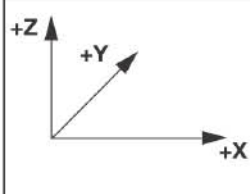
- Complete work plan (refer template at page 260)
- Program to machine the relevant application using the appropriate cycles and G-codes

b. Technical drawing

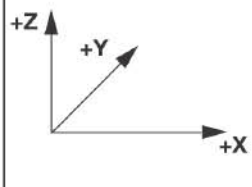


c. The following tools and technology data are required for programming

programGuide Siemens Operate VIETNAM 2021/2022		
Tool shape file	CNC Milling	
Workpiece: AP2 W2019-20 DIN	Blank material: S235JR+C	Machine configuration:
Drawing: 1(1)	Blank dimensions: Fl 100 x 50 x 90	Milling 5 axes (X Y Z A C)

	Technological data							
Tool - Nr.	T1	T2	T3	T4	T5	T6	T7	T8
Tool Ø (mm)	12 x 90°	8 x 90°	20	25	25	10	10	16
Cutting speed (m/min)	30	105	150 / 175	150	175	150	175	150
Cutting depth a_p = max.	6	4	5	12	12	6	6	5
Cutting material	HSS	SC	SC	SC	SC	SC	SC	SC
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,1	0,07	0,05	0,03	0,06
Number of teeth N		4	2	4	4	4	4	3
								
				□	□	□	□	□

	Technological data							
Tool - Nr.	T9	T10	T11	T12	T13	T14	T15	T16
Tool Ø (mm)	8	8	4	16 x 8	16	6,8	M8	12 x 90°
Cutting speed (m/min)	150	150 / 175	150 / 175	35	175	30	10	105

Cutting depth a_p = max.	5	5	6	6	5			6
Cutting material	SC	SC	SC	HSS	SC	HSS	HSS	SC
Feed tooth fz / rev. (mm)	0,03	0,02	0,01	0,03	0,05	0,07	F = P	0,025
Number of teeth N	4	4	4	6	3			4
								
	☐	☐	☐		☐			

UNIT 4: CNC PROGRAMMING OF HIGH-LEVEL LANGUAGE FOR EXTREMELY DEMANDING CNC TURNING AND MILLING APPLICATIONS OF THE INDUSTRY

1. **Introduction:** As the fourth unit in the program of the module, the trainees will be familiar with programming methods using programGUIDE, variables, arithmetic parameters, system variables, user variables and loops to solve applications such as advanced Turning, Milling that cannot be fully implemented using the instruction set offered by DIN 66025 or graphical CNC programming.
2. **Objective:** By the end of this training unit, the trainees will be able to

- Develop configurable CNC programmes for component families that differ only in a few geometric sizes.
- Apply the higher level language such as R parameter, arithmetic parameters, system variables, user variables and loops for programs that require their own technology cycle because not all special tasks can be handled using standard cycles.
- Programme a simple operator message to be generated from an ongoing CNC program that cannot be fully implemented using the instruction set offered by DIN 66025 or graphical CNC programming.
- Develop comprehensive programmes logging of machining parameters and measurement results.
- Know and observe the regulations on IT security and occupational safety as well as health and environmental protection.
- Optimize specified programming files.

3. **Content**:

3.1. Introduction and overview of the basics and functionality

3.1.1. Rules for naming of a NC program

Every NC-program has its own name (identifier), which can be chosen at will when creating the program keeping in mind the following rules:

The following guidelines apply to programming frames:

- The length of the name should not exceed 24 characters, since only the first 24 characters are displayed on the NC.
- Permissible characters are:
  + letters: A...Z, a...z
  + numbers: 0...9
  + underlines: _
- The first two characters should be
  + two letters

    or
  + an underline and a character

If this condition is fulfilled, an NC program can be called up as a subprogram from out of another program solely by stating the program name.

If the program name commences with numbers, the sub-program call-up is possible only by using the statement “CALL”.

Examples for right naming are followings:

_MPF100

Exercise_2

3.1.2. Elements of the NC-high level language

Since the set of commands available as per DIN 66025 is no longer adequate for the programming of modern machine tools, it was extended by the elements of the NC-high level language.

Included in this are amongst others:

- Commands of the NC-high level language
  - Contrary to the commands as per DIN 66025 the commands of the NC-high level language feature more than one address letter, e.g.:
    - OVR for speed correction (override)
    - SPOS for spindle positioning
    - TRANSMIT for milling and drilling in G17 on turning machines
- Identifiers (defined names) for:
  - System variables
  - User-defined variables
  - Subroutine
  - Keywords
  - Jump markers
  - Macros

For attention: An identifier must be unique and cannot be used for different objects.

- Relational operators
- Logic operators
- Arithmetic functions
- Control structures

Block length

- A block can consist of a maximum of 512 characters (including commentary and End-of-block-character "LF").

Sequence of statements

- In order to lay out the blocks as clearly as possible, the statements should be arranged within the block according to the following sequence:

  N… G… X… Y… Z… F… S… T… D… M… H…

Note: Some addresses can be used more than once within a block, for instance:

- G…
- M…
- H…

Summary:

N	Address of the block number
G	Departure command
X, Y, Z	Path information
F	Feed
S	Speed
T	Tool
D	Tool correction number
M	Additional functions
H	Auxiliary functions

Table 4.1: Table of some command addresses used in block

3.1.3. Value assignments:

Values can be assigned to the addresses.

The following rules apply in such a case:

- An "="-sign between the address and the value must be written if:
  + the address consists of more than one letter.
  + the value consists of more than one constant.
  + the "="-sign can be omitted if the address consists of a single letter
  + and the value consists of just one constant.
- Signs are permissible.
- Separation characters after the address letter are permissible.
- Examples for right value assignment are followings:
- X25 Value assignment (25) to the address X. An"=" sign is not required
- X1=15 Value assignment (15) to an address (X) with a numerical extension (1). An "=" sign is required
- X=17*(5+SIN(37.5)) Value assignment by means of a numerical expression. An "=" sign is required.

Note:

A special sign "=", "(", "[", ")", "]", "," or an operator must always follow a numerical address extension in order to differentiate between the address and the value.

3.1.4. Programme the basic program using programGUIDE.

a. Description:

This first ProgramGuide exercise is for programming with the help of drill and tap modes with different cycle commands.

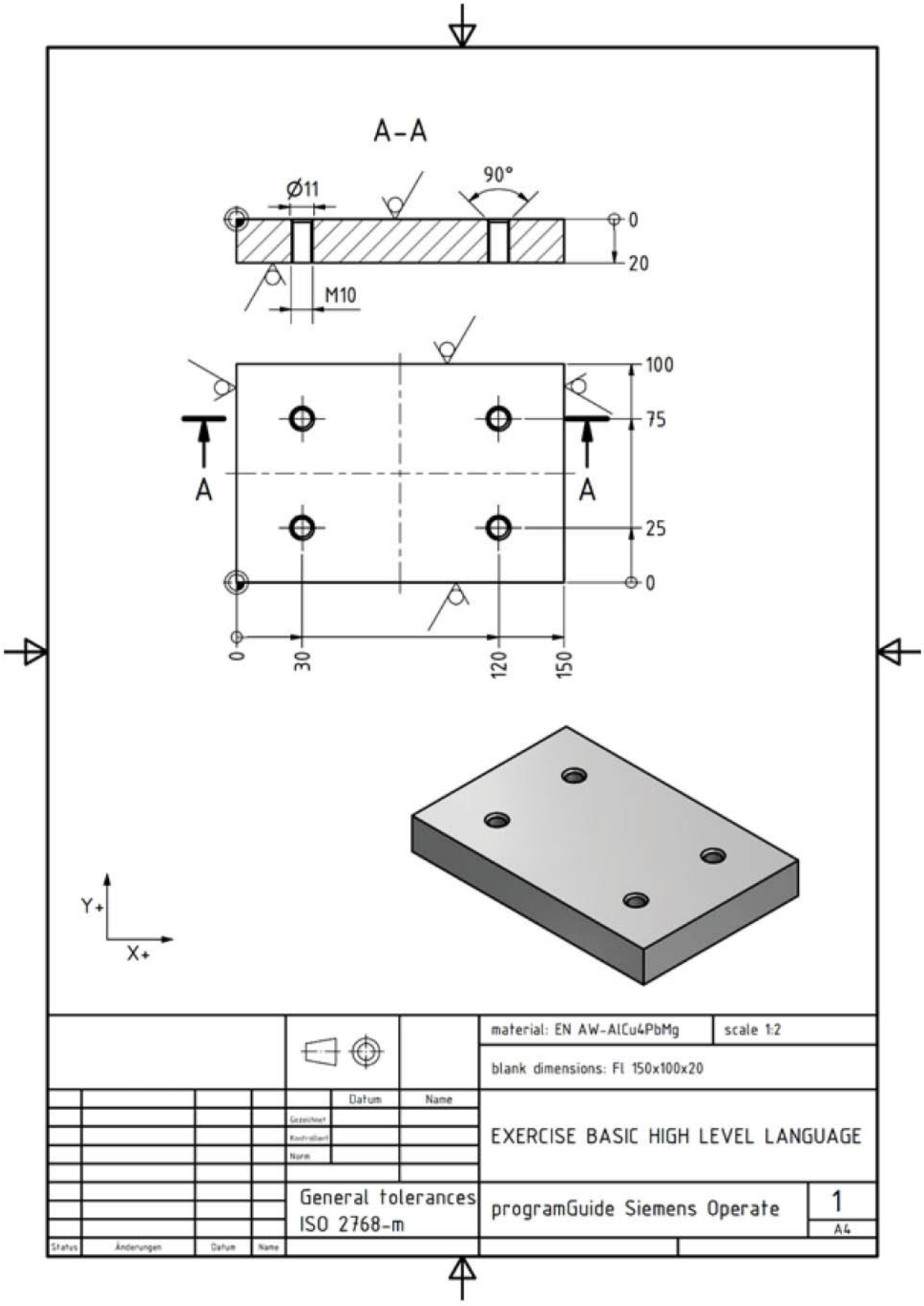
b. Objective:

A new M_200 program should be created and opened in the editor.

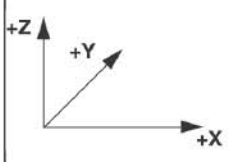
The program header, build groups, centerdrill, deep drilling, threading and finally the simulation programming.

This program was created to form a basis for optimizing the program with high-level language commands in the following articles.

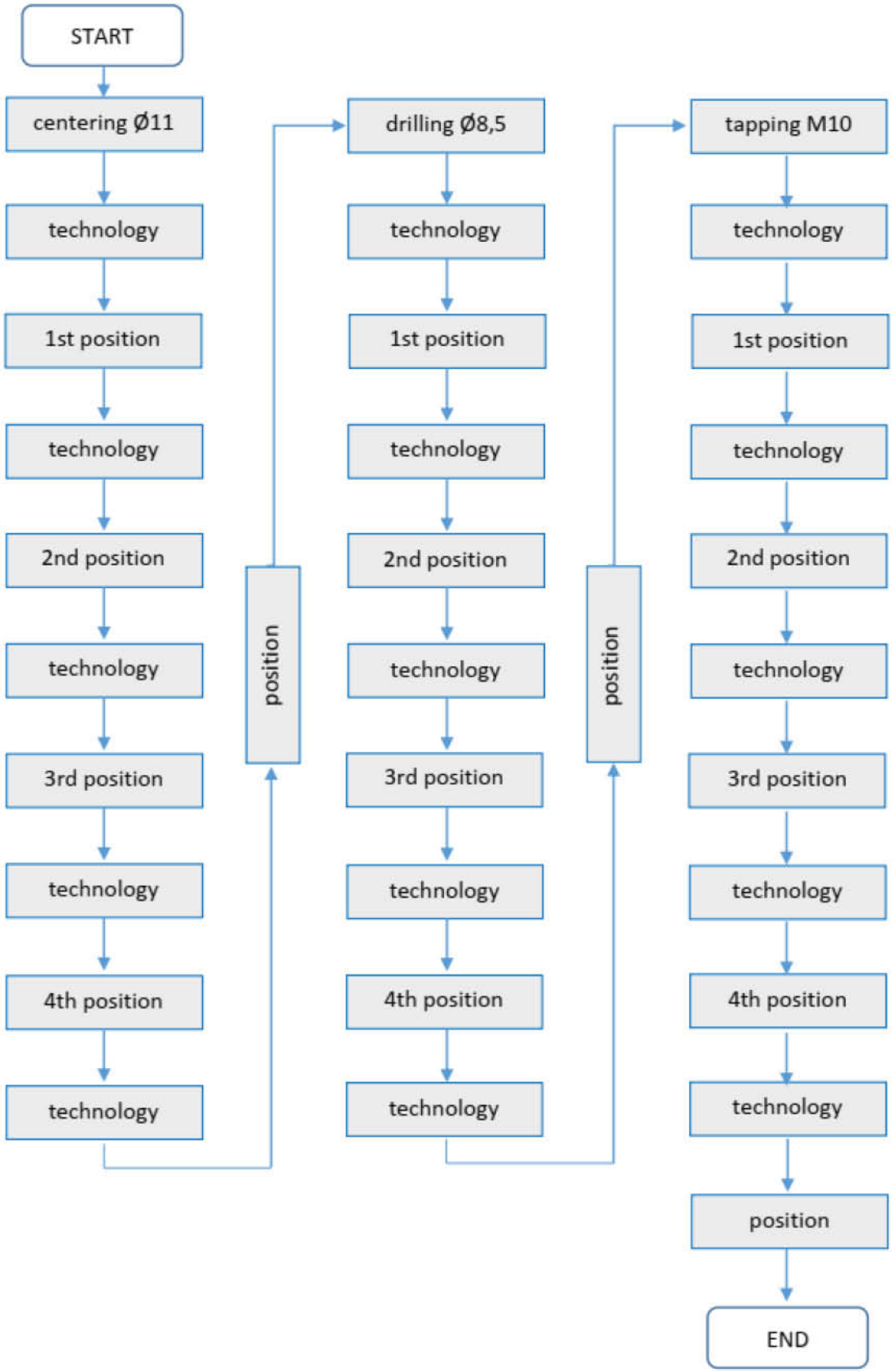
c. The technical drawing EXERCISE BASIC HIGH LEVEL LANGUAGE



d. The following tools and technology data are required for programming:

Workpiece: EXERCISE BASIC HIGH LEVEL LANGUAGE		Blank material: EN AW-AlCu4PbMg				Machine configuration:		
Drawing: 1(1)		Blank dimensions: Fl 150 x 100 x 20				Milling 3 axes		
	Technological data							
Tool name in OFFSET	Spot_drill_12	Deburrer_8	Cutter_32	Cutter_16	Cutter_10	Cutter_4	Drill_8,5	Tap_10
Tool Ø (mm)	12 x 90°	8 x 90°	32	16	10	4	8,5	M10
Cutting speed (m/min)	80	105	150 / 175	150 / 175	150 / 175	150 / 175	60	10
Cutting depth a_p = max.	5	3	5	8	5	2		
Cutting material	HSS	SC	SC	SC	SC	SC	HSS	HSS
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,05 / 0,04	0,03 / 0,02	0,02 / 0,01	0,1	F = P
Number of teeth N		4	2	4	4	4		
								
				✓	✓	✓		

e. Flowchart: A Flowchart is drawn up based on the task at hand.



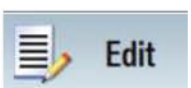
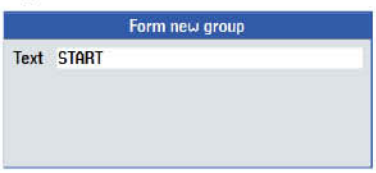
Flowchart 4.1: Technological steps for practical exercise M_200

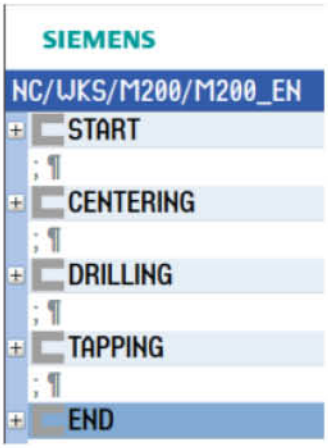
f. Commands used in the program:

SUPA	has the same effect as G153 and also suppresses all other offsets (hand wheel, overlaid movements, external zero offset, PRESET offset)
G17	Working plane X/Y
G54	settable zero offset
G710	Activating the metric system of units All geometrical and technological data in units of length is read and written using the metric system of units.
G40	deactivate tool radius compensation (TRC)
G90	absolute dimensions
D0 / D1	cutting edge number (D1 – D9)
G94	linear feedrate in mm/min
G95	revolutional feedrate in mm/revolution
G331	Tapping in the tapping direction up to the end of thread point
G332	Retraction motion up to the tapping block G331 with automatic spindle direction of rotation reversal

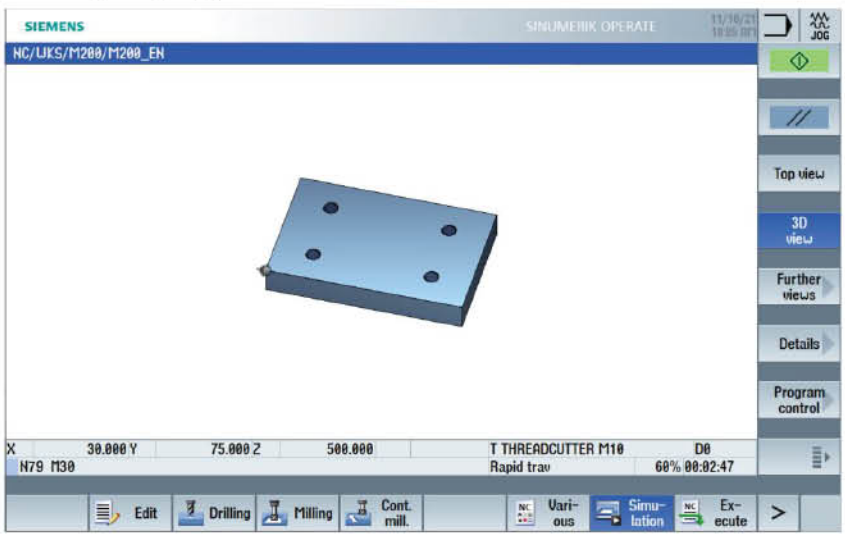
g. Create a new G code programGuide program, like described in this unit.
Create a new program directory and call the program for example "M_200.MPF".

The following blocks are generated in the program for the above example:

1. 	Create a new programGuide directory and main program called “M_200”.
2. 	The “Edit” operation mode opens automatically.
3. 	Press the VSK 2 “Build groups” to open the dialog menu.
4.	Input the first block name “START”. 
5. 	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program:

6.	Create now following blocks in relation the work sequence: 
7.	Enter the following NC blocks for the “START” group: **N2 G17** **N3 G54 D0** **N4 WORKPIECE(,"",,"BOX",0,0,-20,-80,0,0,150,100)** **N5 G0 SUPA Z500 D0 M9** **N6 G40** **N7 G710** **N8 G90** **;** **N9 End of group**
8.	Enter the following NC blocks for the “CENTERING” group: **N10 ;GROUP_BEGIN(0,"CENTERING",0,0)** **N11 G0 SUPA Z500 D0 M9** **N12 T="CENTERDRILL 12" D1** **N13 M6** **N14 G17** **N15 G54 G90** **N16 G95 S2122 F0.12 M3 M8** **N17 G0 Z100** **N18 G0 X30 Y25** **N19 G0 Z2** **N20 G1 Z-5.5** **N21 G0 Z2** **N22 G0 X120** **N23 G1 Z-5.5** **N24 G0 Z2** **N25 G0 Y75** **N26 G1 Z-5.5**

	N27 G0 Z2 **N28 G0 X30** **N29 G1 Z-5.5** **N30 G0 Z2** **N31 G0 SUPA Z500 D0 M9** **;** **N32 End of group**
9.	Enter the following NC blocks for the “DRILLING” group: **N33 ;GROUP_BEGIN(0,"DRILLING",0,0)** **N34 T="DRILL 8.5" D1** **N35 M6** **N36 G17** **N37 G54 G90** **N38 G95 S2247 F0.1 M3 M8** **N39 G0 Z100** **N40 G0 X30 Y25** **N41 G0 Z2** **N42 G1 Z-27** **N43 G0 Z2** **N44 G0 X120** **N45 G1 Z-27** **N46 G0 Z2** **N47 G0 Y75** **N48 G1 Z-27** **N49 G0 Z2** **N50 G0 X30** **N51 G1 Z-27** **N52 G0 Z2** **N53 G0 SUPA Z500 D0 M9** **;** **N54 End of group**
10.	Enter the following NC blocks for the “TAPPING” group: **N55 ;GROUP_BEGIN(0,"TAPPING",0,0)** **N56 T="THREADCUTTER M10" D1** **N57 M6** **N58 G17** **N59 G54 G90** **N60 G95 S318 M3 M8** **N61 G0 Z100**

	N62 G0 X30 Y25 **N63 G0 Z2** **N64 G331 Z-25 K1.5 S318** **N65 G332 Z2 K1.5 S636** **N66 G0 X120** **N67 G331 Z-25 K1.5 S318** **N68 G332 Z2 K1.5 S636** **N69 G0 Y75** **N70 G331 Z-25 K1.5 S318** **N71 G332 Z2 K1.5 S636** **N72 G0 X30** **N73 G331 Z-25 K1.5 S318** **N74 G332 Z2 K1.5 S636** **N75 G0 SUPA Z500 D0 M9** **N76 End of group**
11.	Enter the following NC blocks for the “END” group: N77 ;GROUP_BEGIN(0,"END",0,0) N78 **G0 SUPA Z500 D0 M9** **N79 M30** **N80 End of group**
12.	Check your program with the “Simulation” function. 
13.	The basic program has been completed. This program is now optimized with high-level language commands in the next module. It is recommended that you copy this program to SinuTrain so that the following modules can be reproduced.

	Create a new G code program. Copy the G code program from this module to the editor of the newly created G code program. You now have the basic program for the following modules and can therefore easily reproduce the next steps.

3.2. Subprogram

3.2.1. Working with subprogram

- Produces a clean program structure. The program is clearer and easier to read.
- Program sections can be used repeatedly, also as "block" and machining library.
- Since program sections do not have to be written again and again, this saves time when creating a program and memory space.

3.2.2. Stucture of subprogram

- In principle, the subprogram is structured in the same way as the main program.
- There are specific commands for the subprogram depending on the application.
- The call of subprograms is subject to a certain hierarchy:
  - + "Programname.MPF" is found when it is in the same directory (WPD).
  - + "Programname.SPF" is found in precisely the following search order:
    1. In the active director
    2. In the "Subprograms" folder
    3. In the user cycle directory
    4. In the manufacturer cycle directory
    5. In the standard cycle directory

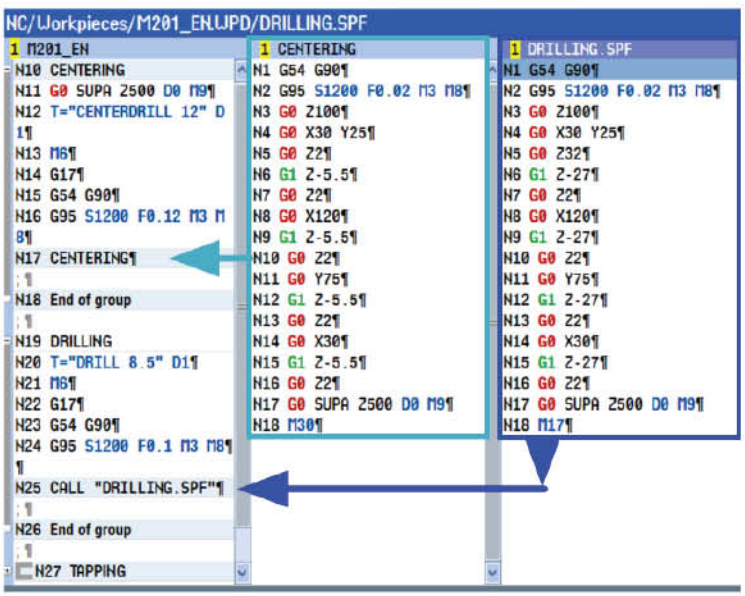


Figure 4.1: Stucture of subprogram

3.2.3. Commands M17, RET, M30

M17	Address of the block number
RET	Departure command
Name	Path information

- M17: Marks the end of the subprogram with instruction to return to the main program. Return is via PLC.
- RET: Is also for the end of the subprogram, but without interruption of the continuous-path mode and without function output on the PLC. (Quicker return than M17)
- Name: The same rules (DOS conventions) apply for the program names of subprograms as for the main program:
  + Program name max. 24 characters. They are displayed in the NC.
  + Letters “A...Z” and “a...z”
  + Digits “0...9”
  + Underscores “_ “
- Reserved keywords, e.g. "RET", cannot be selected for names.

3.2.4. Programme the basic program using programGUIDE.

a. Description:

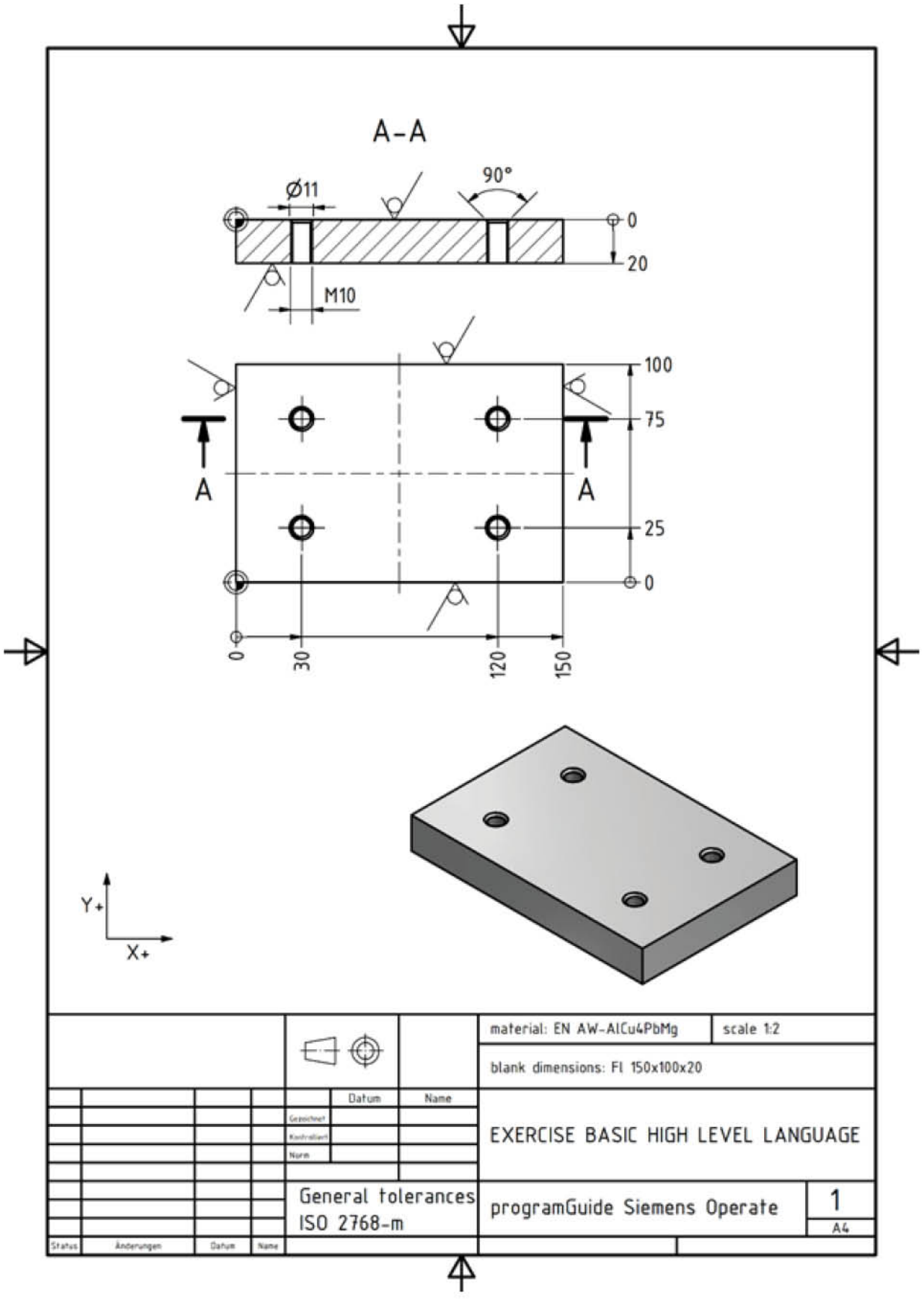
Perform basic program setup from folder M_200 which is divided into subprogram depending on the machining job. The main program and subprogram end in “.MPF” and must be in the same folder M_201.

b. Objective:

A new M_201 program should be created and opened in the editor.

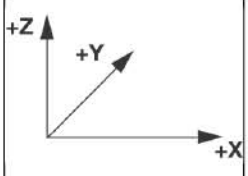
Using the subroutine name call, the subroutine call using the M17 instruction, the RET instruction, and finally the simulator programming.

c. The technical drawing EXERCISE BASIC HIGH LEVEL LANGUAGE

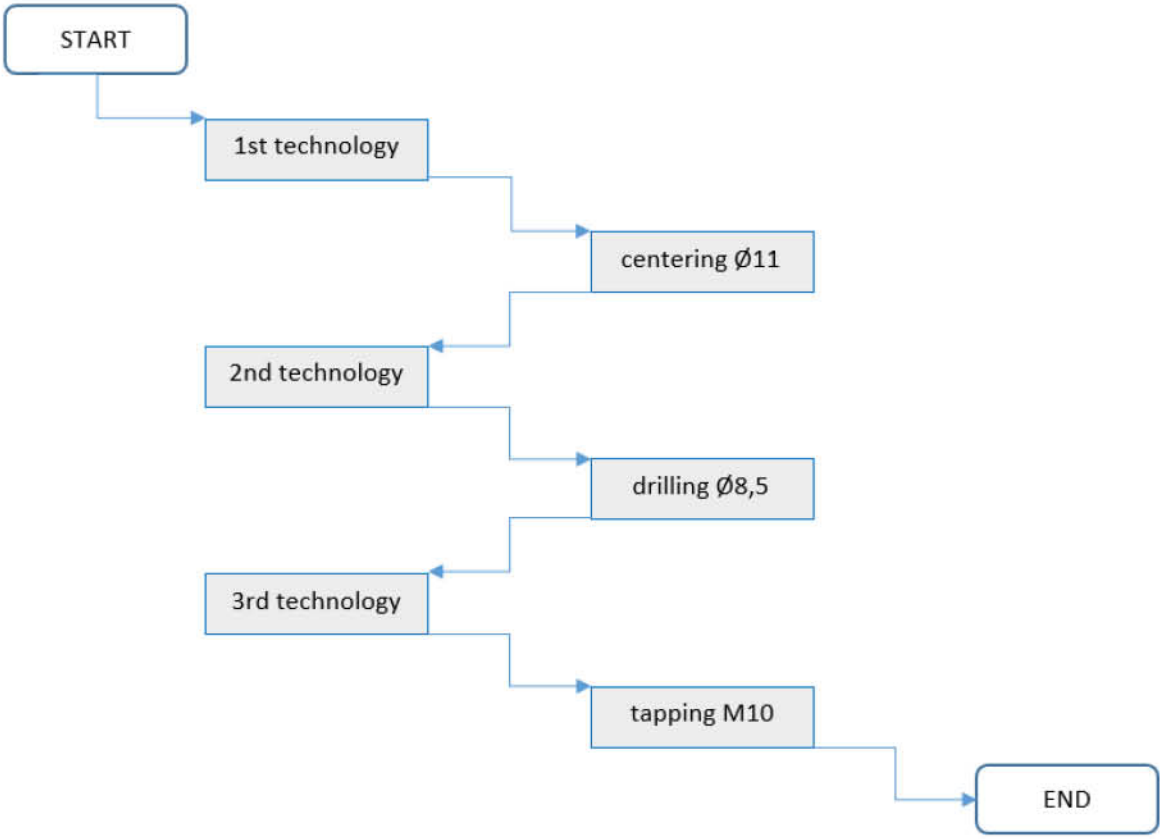


d. The following tools and technology data are required for programming:

Workpiece: EXERCISE BASIC HIGH LEVEL LANGUAGE	Blank material: EN AW-AlCu4PbMg	Machine configuration:
Drawing: 1(1)	Blank dimensions: FI 150 x 100 x 20	Milling 3 axes

	Technological data							
Tool name in OFFSET	Spot_drill_12	Deburrer_8	Cutter_32	Cutter_16	Cutter_10	Cutter_4	Drill_8,5	Tap_10
Tool Ø (mm)	12 x 90°	8 x 90°	32	16	10	4	8,5	M10
Cutting speed (m/min)	80	105	150 / 175	150 / 175	150 / 175	150 / 175	60	10
Cutting depth a_p = max.	5	3	5	8	5	2		
Cutting material	HSS	SC	SC	SC	SC	SC	HSS	HSS
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,05 / 0,04	0,03 / 0,02	0,02 / 0,01	0,1	F = P
Number of teeth N		4	2	4	4	4		
								
				✓	✓	✓		

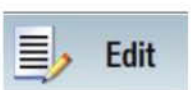
e. Flowchart: A Flowchart is drawn up based on the task at hand.



Flowchart 4.2: Technological steps for practical exercise M_201

f. Create a new G code programGuide program, like described in this unit.

Create a new program directory and call the program for example "M_201.MPF" by copying the folder M_200. The following blocks are created in the program for the above example:

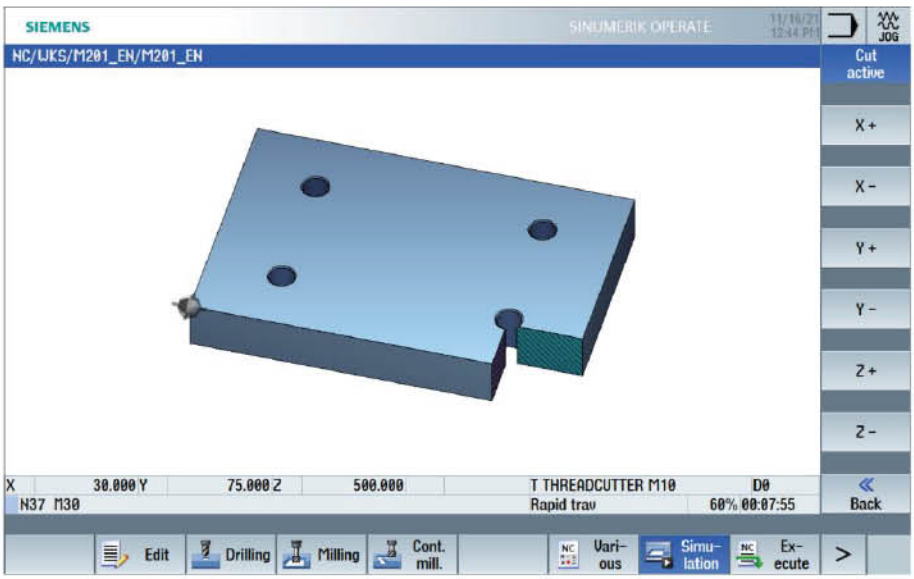
1.	Create a new programGuide directory and main program called “M_201”.
2. 	The “Edit” operation mode opens automatically.
3.	Press the VSK 2 “Build groups” to open the dialog menu.
4.	Input the first block structure as follow:

	NC/WKS/M201_EN/M201_EN N1 Start ;¶ N10 CENTERING ;¶ N19 DRILLING ;¶ N27 TAPPING ;¶ N35 Ende
5.	The commands for the program header are programmed in the "START" block. **N1 GROUP_BEGIN(0,"Start",0,0)** **N2 G17** **N3 G54 D0** **N4 WORKPIECE(,"",,"BOX",0,0,-20,-80,0,0,150,100)** **N5 G0 SUPA Z500 D0 M9** **N6 G40** **N7 G710** **N8 G90** **;** **N9 GROUP_END(0,0)** NC/WKS/M201_EN/M201_EN N1 Start N2 G17¶ N3 G54 D0¶ N4 WORKPIECE(,"",,"BOX",0,0,-20,-80,0,0,150,100)¶ ¶ N5 G0 SUPA Z500 D0 M9¶ N6 G40¶ N7 G710¶ N8 G90¶ ;¶ N9 End of group
6.	The technology, travel commands and subroutine calls are programmed in the "CENTERING" block: **N10 GROUP_BEGIN(0,"CENTERING",0,0)** **N11 G0 SUPA Z500 D0 M9** **N12 T="CENTERDRILL 12" D1** **N13 M6** **N14 G17** **N15 G54 G90** **N16 G95 S2122 F0.12 M3 M8**

	N17 CENTERING **;** **N18 GROUP_END(0,0)** NC/WKS/M201_EN/M201_EN N1 Start ;¶ N10 CENTERING N11 G0 SUPA Z500 D0 M9¶ N12 T="CENTERDRILL 12" D1¶ N13 M6¶ N14 G17¶ N15 G54 G90¶ N16 G95 S2122 F0.12 M3 M8¶ N17 CENTERING¶ ;¶ N18 End of group
7.	The technology, travel commands and subroutine calls are programmed in the "DRILLING" block using the M17 instruction: **N19 GROUP_BEGIN(0,"DRILLING",0,0)** **N20 T="DRILL 8.5" D1** **N21 M6** **N22 G17** **N23 G54 G90** **N24 G95 S2247 F0.1 M3 M8** **N25 CALL "DRILLING.SPF"** **;** **N26 GROUP_END(0,0)** N19 DRILLING N20 T="DRILL 8.5" D1¶ N21 M6¶ N22 G17¶ N23 G54 G90¶ N24 G95 S2247 F0.1 M3 M8¶ ¶ N25 CALL "DRILLING.SPF"¶ ;¶ N26 End of group
8.	The technology, travel commands and subroutine calls are programmed in the "TAPPING" block using the RET instruction: **N27 GROUP_BEGIN(0,"TAPPING",0,0)** **N28 T="THREADCUTTER M10" D1** **N29 M6** **N30 G17**

	N31 G54 G90 **N32 G95 S318 M3 M8** **N33 CALL "TAPPING.SPF"** **N34 GROUP_END(0,0)** N27 TAPPING N28 T="THREADCUTTER M10" D1¶ N29 M6¶ N30 G17¶ N31 G54 G90¶ N32 G95 S318 M3 M8¶ N33 CALL "TAPPING.SPF"¶ ¶ N34 End of group
9.	The commands for the end of the program are programmed in the "END" block: **N35 GROUP_BEGIN(0,"Ende",0,0)** **N36 G0 SUPA Z500 D0 M9** **N37 M30** **N38 GROUP_END(0,0)** N35 Ende N36 G0 SUPA Z500 D0 M9¶ N37 M30¶ N38 End of group
10.	A new G code program is created with the name "CENTERING.MPF" for the subprogram call in the "CENTERING" block.
11.	The positions of the centering bores are programmed. **N1 G54 G90** **N2 G95 S2122 F0.12 M3 M8** **N3 G0 Z100** **N4 G0 X30 Y25** **N5 G0 Z2** **N6 G1 Z-5.5** **N7 G0 Z2** **N8 G0 X120** **N9 G1 Z-5.5** **N10 G0 Z2** **N11 G0 Y75** **N12 G1 Z-5.5** **N13 G0 Z2** **N14 G0 X30**

	N15 G1 Z-5.5 **N16 G0 Z2** **N17 G0 SUPA Z500 D0 M9** **N18 M30**
12.	A new G code program is created with the name “DRILLING.SPF” for the subprogram call in the "DRILLING" block using the M17 instruction.
13.	The positions of the holes are programmed: **N1 G54 G90** **N2 G95 S2247 F0.02 M3 M8** **N3 G0 Z100** **N4 G0 X30 Y25** **N5 G0 Z32** **N6 G1 Z-27** **N7 G0 Z2** **N8 G0 X120** **N9 G1 Z-27** **N10 G0 Z2** **N11 G0 Y75** **N12 G1 Z-27** **N13 G0 Z2** **N14 G0 X30** **N15 G1 Z-27** **N16 G0 Z2** **N17 G0 SUPA Z500 D0 M9** **N18 M17**
14.	A new G code program is created with the name "_TAPPING:SPF" for the subprogram call in the "TAPPING" block using the RET instruction.
15.	The positions of the threaded holes are programmed: **N1 G0 Z100** **N2 G0 X30 Y25** **N3 G0 Z2** **N4 G331 Z-25 K1.5 S318** **N5 G332 Z2 K1.5 S636** **N6 G0 X120** **N7 G331 Z-25 K1.5 S318** **N8 G332 Z2 K1.5 S636** **N9 G0 Y75** **N10 G331 Z-25 K1.5 S318** **N11 G332 Z2 K1.5 S636**

	N12 G0 X30 **N13 G331 Z-25 K1.5 S318** **N14 G332 Z2 K1.5 S636** **N15 G0 SUPA Z500 D0 M9** **N16 RET**
17.	Check your program with the “Simulation” function. 

3.3. R Parameter

3.3.1. Predefined arithmetic parameters

3.3.1.1. Command:

R	Predefined arithmetic parameters for which the letter "R" is provided followed by a classification number.

- The arithmetic parameters are available for every SINUMERIK control and can be used by users/programmers.
- The number of available arithmetic parameters varies and can be defined by a machine data item. Normally, 99 variables are available.
- Please note the specifications of the machine manufacturer.

3.3.1.2. Function:

- Arithmetic parameters can be used when an NC program is to be configured flexibly because the values vary within a program depending on the workpiece.
- These parameters can be assigned arbitrary NC addresses within a program with a flexible value.
- The assigned values must be numeric.
- It is important that the variables are assigned the correct values because they are not checked for plausibility as standard.
- R variables are variables that are retained (retentive) after the machine is switched off.
- R variables are local, only visible in the current channel.

With R-parameter programming, one program can be used for different workpieces.

3.3.2. R variable requirements

- The value for the R variable must be assigned in a block
- Example: R12=100
  - + Maximum eight decimal places are permitted.
  - + The decimal point can be omitted with integers.
  - + The sign can be omitted for numbers with a positive sign. Example: R12=100; R12=(-100)
  - + Arithmetic functions can be performed with R variables - the mathematical notation applies.
  - + An assignment can be made to nearly all addresses with R variables, except (N, G and L).

3.3.3. Calculated parameters

- R parameters are calculated in the program using mathematical formulas and displayed on the control panel in the Sinumerik operating system.
- R variables in SINUMERIK

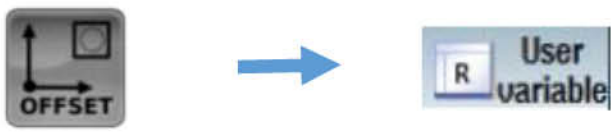


Figure 4.2: Opening user variables

- Display of the R variables on the control:

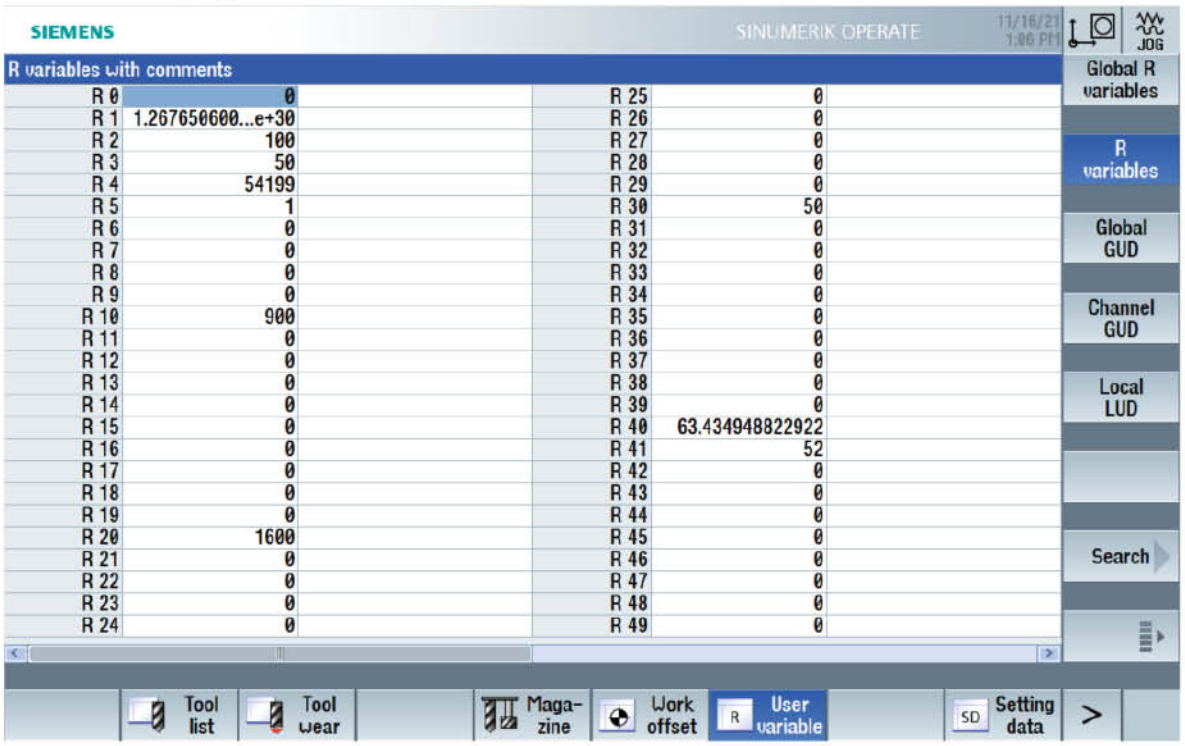
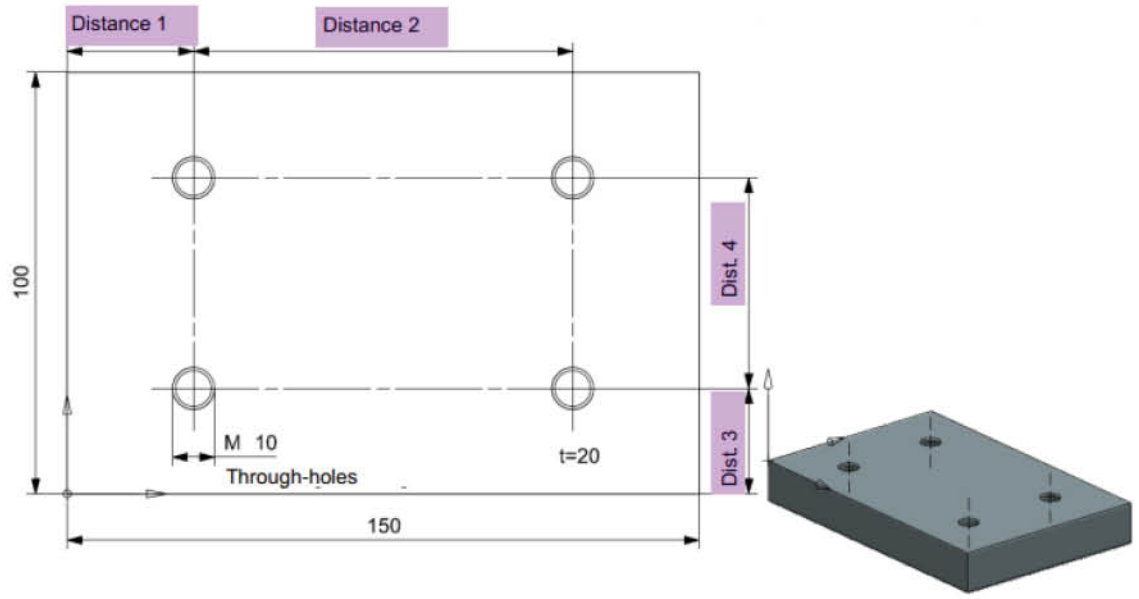


Figure 4.3: R parameters are displayed in the user variables section

Example: R variables are declared in the program according to the parameters in the following drawing:



- With: R1= distance 1 in the drawing
  R2= distance 2 in the drawing
  R3= distance 3 in the drawinng
  R4= distance 4 in the drawing
- The parameters of the above variables can change values, assuming:
  R1= 30
  R2=90
  R3=25
  R4=50
- With command line:
  G0 X=R1 Y=R3

Then the program understands will run to position

  G0 X30 Y25

- With command line:
  G0 X=R1+R2

Then the program understands will run to position

  G0 X120

- To reset the R variable to its default value, assign the value =0 after each variable, for example:
  R1=0
  R2=0
  R3=0
  R4=0

3.3.4. Calculated R parameter displayed by message text

3.3.4.1. Command:

MSG	Arbitrary texts, but also values of variables, can be output in the alarm line in the Automatic operating area.

3.3.4.2. Function:

- Messages can be programmed for the operator with the MSG command that are to be displayed during the program execution.
- Arbitrary texts and also values of variables can be displayed via the MSG command.
- Example of message text:

  MSG("This is a sample text")
  - + This text remains active and visible until the end of the program.

    MSG()
  - + This command deselects the currently displayed text at an arbitrary position in the program
- Example of message text with variable:

  MSG("Value of variable R1="<<R1)
  - + Text between the quotation marks can be arbitrary "Value of variable R1=", and is displayed in the Automatic operation area.
  - + Two "less than" symbols << show the value in this case of the R variable R1.

3.3.5. Programme the basic program according to technical drawings.

a. Description:

This ProgramGuide exercise is programmed with the help of drill and tap modes with different cycles using R variables and messages.

This new program can be used for machining different workpieces with R variable programming.

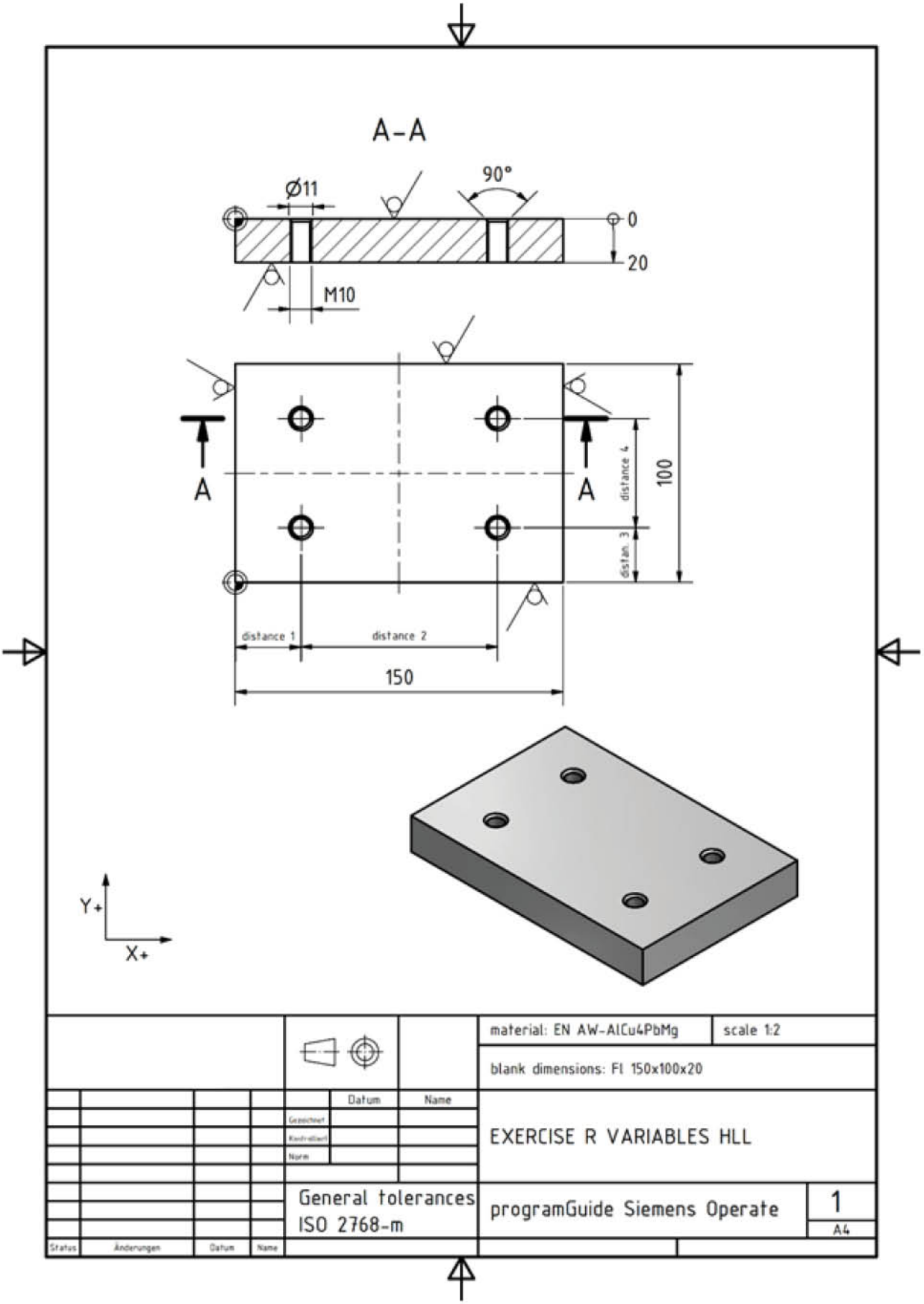
b. Objective:

Create a new program named M_202.

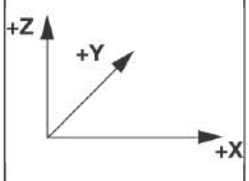
Perform program tasks:

- Task for R: The hole distances in relation to the workpiece zero are to be programmed variably in the program.
- Task for MSG:
  - + A message text is to show the operator whether the values in the R variables have been entered correctly.
  - + At the end, the current values of the R variables are to be displayed to the operator in the Automatic operating area.

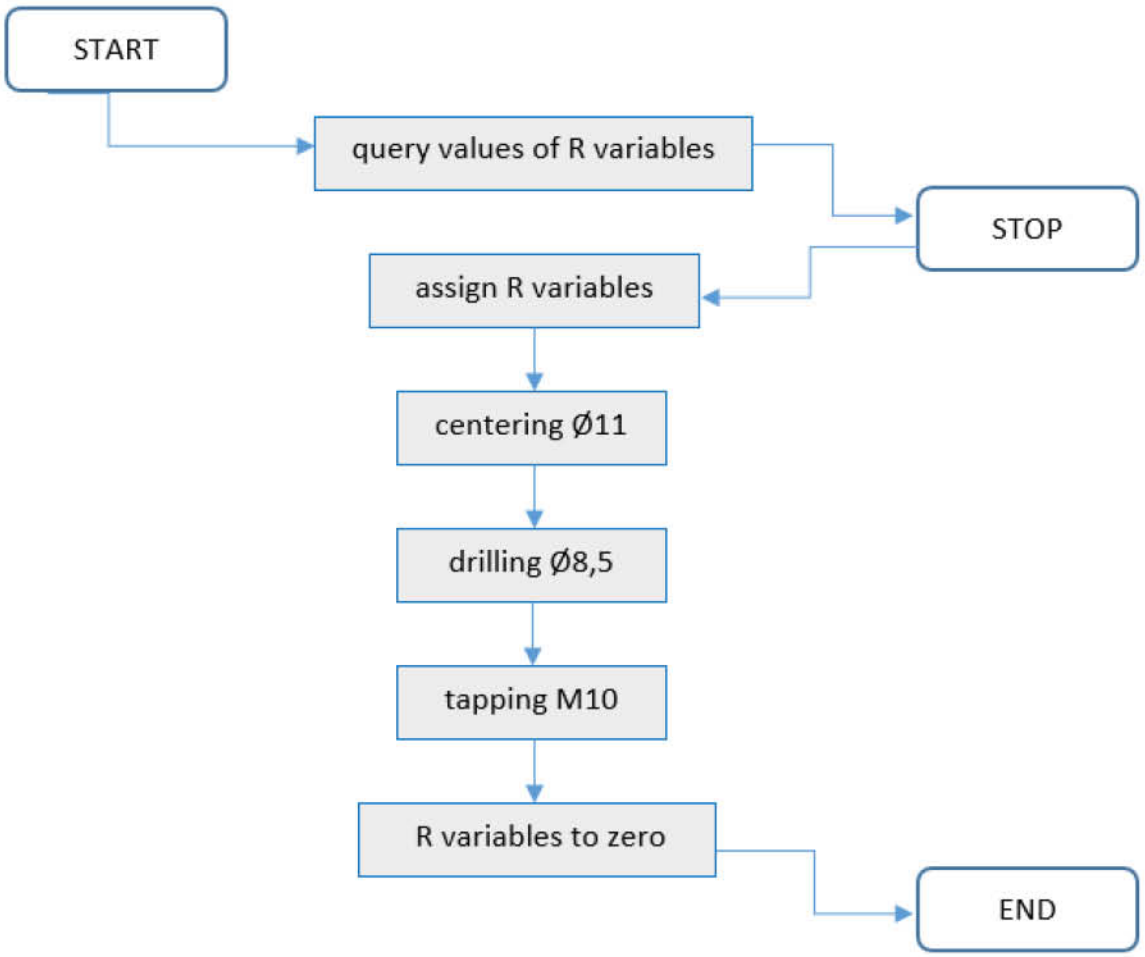
c. The technical drawing EXERCISE R VARIABLES HLL



d. The following tools and technology data are required for programming:

Workpiece: EXERCISE R VARIABLES HLL		Blank material: EN AW-AlCu4PbMg				Machine configuration:		
Drawing: 1(1)		Blank dimensions: Fl 150 x 100 x 20				Milling 3 axes		
	Technological data							
Tool name in OFFSET	Spot_drill_12	Deburrer_8	Cutter_32	Cutter_16	Cutter_10	Cutter_4	Drill_8,5	Tap_10
Tool Ø (mm)	12 x 90°	8 x 90°	32	16	10	4	8,5	M10
Cutting speed (m/min)	80	105	150 / 175	150 / 175	150 / 175	150 / 175	60	10
Cutting depth a_p = max.	5	3	5	8	5	2		
Cutting material	HSS	SC	SC	SC	SC	SC	HSS	HSS
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,05 / 0,04	0,03 / 0,02	0,02 / 0,01	0,1	F = P
Number of teeth N		4	2	4	4	4		
								
				✓	✓	✓		

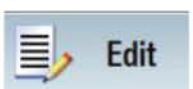
e. Flowchart: A Flowchart is drawn up based on the task at hand.

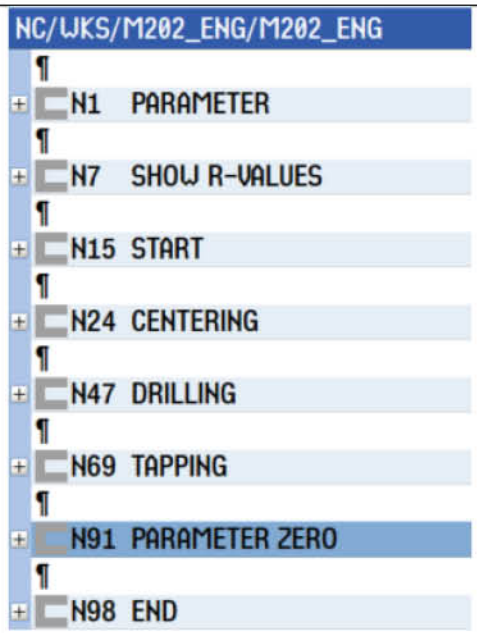
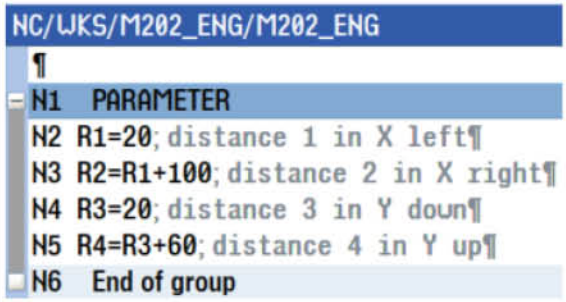


Flowchart 4.3: Technological steps for practical exercises using parameters

f. Create a new G code programGuide program, like described in this unit.
Create a new program directory and call the program for example "M_202.MPF".

The following blocks are created in the program for the above example:

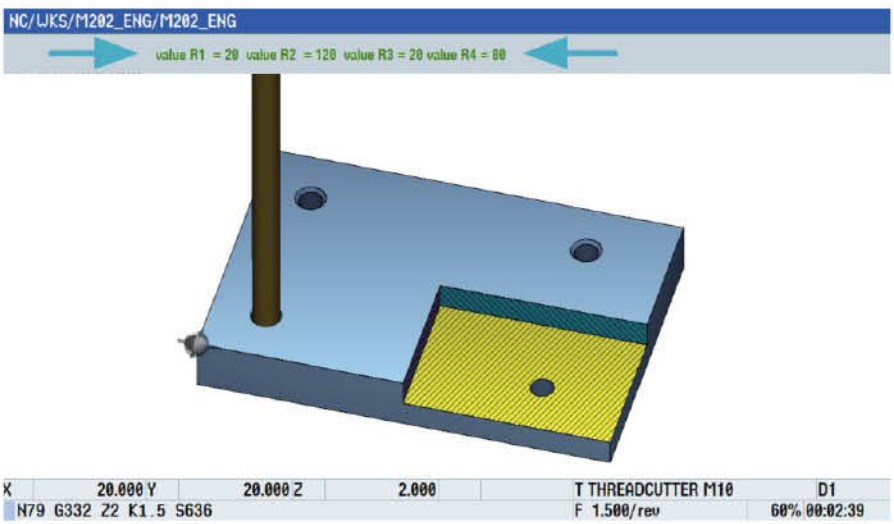
1.	Create a new programGuide directory and main program called “M_202”.
2. 	The “Edit” operation mode opens automatically.
3.	Press the VSK 2 “Build groups” to open the dialog menu.
4.	Input the first block structure as follow:

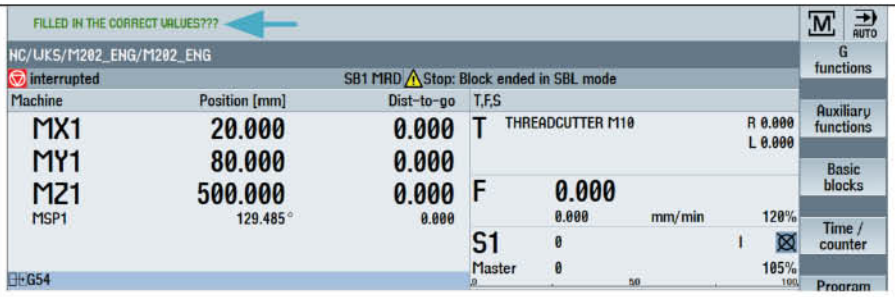
	
5.	***The commands for the parameter assign are programmed in the "PARAMETER" block.*** N1 GROUP_BEGIN(0,"PARAMETER",0,0) N2 R1=20 ;distance 1 in X left N3 R2=R1+100 ;distance 2 in X right N4 R3=20 ;distance 3 in Y down N5 R4=R3+60 ;distance 4 in Y up N6 GROUP_END(0,0) 
6.	***The commands to display the parameter values are programmed in the "SHOW R-VALUES" block.*** N7 GROUP_BEGIN(0,"SHOW R-VALUES",0,0) N8 MSG("FILLED IN THE CORRECT VALUES???") ;query values screen message N9 G4 F100 ;100 seconds dwell time N10 M0 ;program stop N11 MSG ;remove screen message N12 MSG(" value R1 = "<<R1<<" value R2 = "<<R2<<" value R3 = "<<R3<<" value R4 = "<<R4) ;parameter values are shown

	N13 M0 ;program stop N14 GROUP_END(0,0) N7 SHOW R-VALUES N8 MSG("MESSAGE TEXT") ;show screen message¶ N9 G4 F100 ;100 seconds duell time¶ N10 M0 ;program stop¶ N11 MSG ;remove screen message¶ N12 MSG(" value R1 = "<<R1<<" value R2 = "<<R2<<" value R3 = "<<R3<<" value R4 = "<<R4)¶ ;parameter values are shown¶ ¶ N13 M0 ;program stop¶ N14 End of group
7.	***The commands for the program header are programmed in the "START" block.*** N1 GROUP_BEGIN(0,"Start",0,0) N2 G17 N3 G54 D0 N4 WORKPIECE(,"",,"BOX",0,0,-20,-80,0,0,150,100) N5 G0 SUPA Z500 D0 M9 N6 G40 N7 G710 N8 G90 ; N9 GROUP_END(0,0) N15 START N16 G17¶ N17 G54 D0¶ N18 WORKPIECE(,"",,"BOX",0,0,-20,-80,0,0,150,100)¶ ¶ N19 G0 SUPA Z500 D0 M9¶ N20 G40¶ N21 G710¶ N22 G90¶ ;¶ N23 End of group
8.	***The technology, travel commands and subroutine calls are programmed in the "CENTERING" block:*** N24 GROUP_BEGIN(0,"CENTERING",0,0) N25 G0 SUPA Z500 D0 M9 N26 T="CENTERDRILL 12" D1 N27 M6 N28 G17

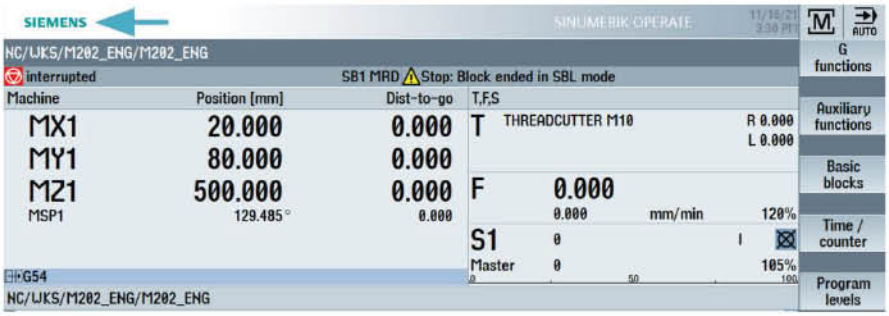
	N29 G54 G90 N30 G95 S2122 F0.12 M3 M8 N31 G0 Z100 N32 G0 X=R1 Y=R3 N33 G0 Z2 N34 G1 Z-5.5 N35 G0 Z2 N36 G0 X=R2 N37 G1 Z-5.5 N38 G0 Z2 N39 G0 Y=R4 N40 G1 Z-5.5 N41 G0 Z2 N42 G0 X=R1 N43 G1 Z-5.5 N44 G0 Z2 N45 G0 SUPA Z500 D0 M9 ; N46 GROUP_END(0,0)	N25 G0 SUPA Z500 D0 M9¶ N26 T="CENTERDRILL 12" D1¶ N27 M6¶ N28 G17¶ N29 G54 G90¶ N30 G95 S2122 F0.12 M3 M8¶ N31 G0 Z100¶ N32 G0 X=R1 Y=R3¶ N33 G0 Z2¶ N34 G1 Z-5.5¶ N35 G0 Z2¶ N36 G0 X=R2¶ N37 G1 Z-5.5¶ N38 G0 Z2¶ N39 G0 Y=R4¶ N40 G1 Z-5.5¶ N41 G0 Z2¶ N42 G0 X=R1¶ N43 G1 Z-5.5¶ N44 G0 Z2¶ N45 G0 SUPA Z500 D0 M9¶
9.	***The technology, travel commands and subroutine calls are programmed in the "DRILLING" block:*** N47 GROUP_BEGIN(0,"DRILLING",0,0) N48 T="DRILL 8.5" D1 N49 M6 N50 G17 N51 G54 G90 N52 G95 S2247 F0.12 M3 M8 N53 G0 Z100 N54 G0 X=R1 Y=R3	

	N55 G0 Z2 N56 G1 Z-27 N57 G0 Z2 N58 G0 X=R2 N59 G1 Z-27 N60 G0 Z2 N61 G0 Y=R4 N62 G1 Z-27 N63 G0 Z2 N64 G0 X=R1 N65 G1 Z-27 N66 G0 Z2 N67 G0 SUPA Z500 D0 M9 ; N68 GROUP_END(0,0)	N47 DRILLING N48 T="DRILL 8.5" D1¶ N49 M6¶ N50 G17¶ N51 G54 G90¶ N52 G95 S2247 F0.12 M3 M8¶ N53 G0 Z100¶ N54 G0 X=R1 Y=R3¶ N55 G0 Z2¶ N56 G1 Z-27¶ N57 G0 Z2¶ N58 G0 X=R2¶ N59 G1 Z-27¶ N60 G0 Z2¶ N61 G0 Y=R4¶ N62 G1 Z-27¶ N63 G0 Z2¶ N64 G0 X=R1¶ N65 G1 Z-27¶ N66 G0 Z2¶ N67 G0 SUPA Z500 D0 M9¶
10.	***The technology, travel commands and subroutine calls are programmed in the "TAPPING" block:*** N69 GROUP_BEGIN(0,"TAPPING",0,0) N70 T="THREADCUTTER M10" D1 N71 M6 N72 G17 N73 G54 G90 N74 G95 S318 M3 M8 N75 G0 Z100 N76 G0 X=R1 Y=R3 N77 G0 Z2 N78 G331 Z-25 K1.5 S318 N79 G332 Z2 K1.5 S636 N80 G0 X=R2 N81 G331 Z-25 K1.5 S318 N82 G332 Z2 K1.5 S636 N83 G0 Y=R4 N84 G331 Z-25 K1.5 S318 N85 G332 Z2 K1.5 S1636 N86 G0 X=R1	N25 G0 SUPA Z500 D0 M9¶ N26 T="CENTERDRILL 12" D1¶ N27 M6¶ N28 G17¶ N29 G54 G90¶ N30 G95 S2122 F0.12 M3 M8¶ N31 G0 Z100¶ N32 G0 X=R1 Y=R3¶ N33 G0 Z2¶ N34 G1 Z-5.5¶ N35 G0 Z2¶ N36 G0 X=R2¶ N37 G1 Z-5.5¶ N38 G0 Z2¶ N39 G0 Y=R4¶ N40 G1 Z-5.5¶ N41 G0 Z2¶ N42 G0 X=R1¶ N43 G1 Z-5.5¶ N44 G0 Z2¶ N45 G0 SUPA Z500 D0 M9¶

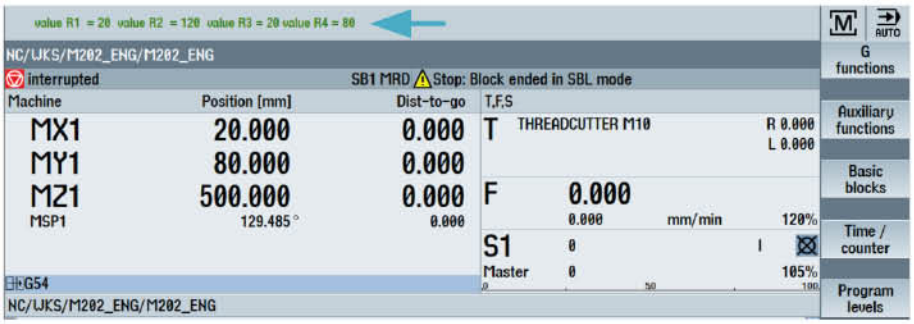
	N87 G331 Z-25 K1.5 S318 N88 G332 Z2 K1.5 S636 N89 G0 SUPA Z500 D0 M9 N90 GROUP_END(0,0)
11.	***The commands for the parameter set to zero are programmed in the "PARAMETER ZERO" block.*** N91 GROUP_BEGIN(0,"PARAMETER ZERO",0,0) N92 MSG N93 R1=0 N94 R2=0 N95 R3=0 N96 R4=0 N97 GROUP_END(0,0) N91 PARAMETER ZERO N92 MSG¶ ¶ N93 R1=0¶ N94 R2=0¶ N95 R3=0¶ N96 R4=0¶ N97 End of group
12.	***The commands for the end of the program are programmed in the "END" block:*** N98 GROUP_BEGIN(0,"END",0,0) N99 G0 SUPA Z500 D0 M9 N100 M30 N101 GROUP_END(0,0) N98 END N99 G0 SUPA Z500 D0 M9¶ N100 M30¶ N101 End of group
13.	***Check your program with the "Simulation" function.*** 
14.	***1st message text in "AUTOMATIC" operation mode:***



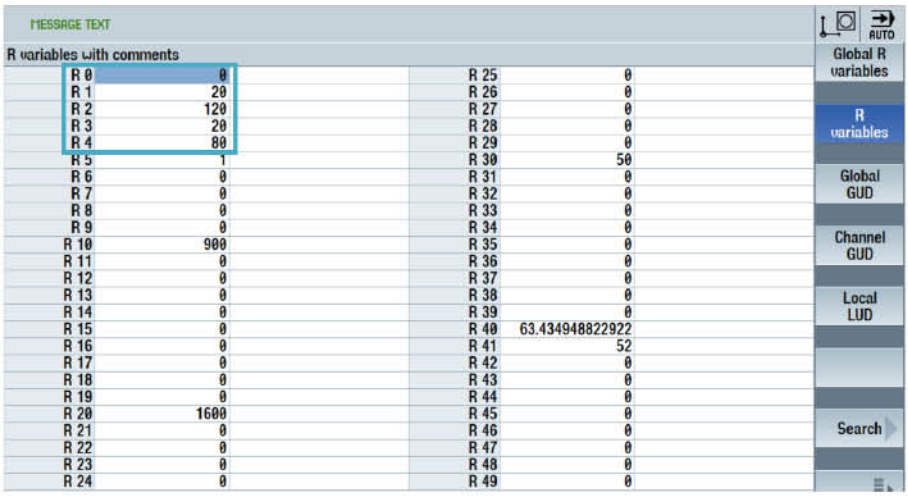
15. ***2nd the message is removed in "AUTOMATIC" operation mode:***

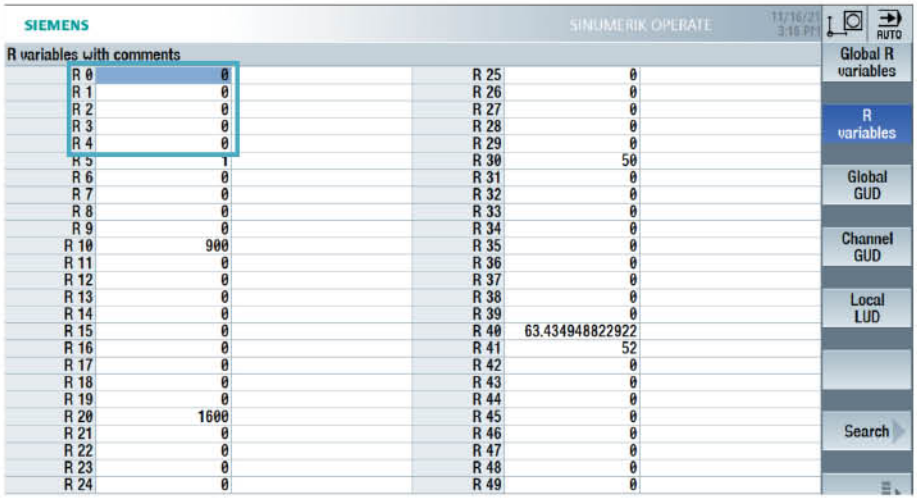


16. ***3rd the R parameter values are shown in "AUTOMATIC" operation mode:***



17. ***The entered values are assigned to the R variables in the R variable list and ...***



	… deselected at the end of the program. 

3.4.Jumps and user variables

Commands:

DEF INT	User-defined variables
IF	Conditions
LABEL	Jump labels
GOTO; GOTOF; GOTOB	Jump instructions
RO	Read Only = text is read-only (capitalization) with write protection

3.4.1. Explanation of variables

DEF INT: You can define your own variables and assign them values with this command. The so-called ***user variables***.

- User variables should always ***begin with an underscore*** in order to ensure that no preassigned name is used.
- A user variable should not have more than 15 characters.
- Notation always the same (all characters upper-case).
- INT always identifies integers. There are various variable types that are described in detail in the COMPLETE HIGH LEVEL LANGUAGE Manual of Siemens.
- The value is assigned to the variable using the equals sign (VALUE =).
- The query about the significance of a variable is performed using two equal signs (IF VALUE ==1).

IF: This command is a keyword to which a condition is always linked. In this way, the programmer can change the programmed sequence of NC programs in order to integrate a control structure.

3.4.2. Explanation of jumps

Label: Jump labels can be set with the so-called label, which is referenced from other positions in the same program.

- The label name can be selected arbitrarily.
- It is useful to use a "speaking" LABEL name, e.g. THREAD CUTTING.
- If a label is referenced as jump destination, there must be a colon behind the word.

  LABEL_1: This designates the label as such.

3.4.3. Jump commands GOTO/ GOTOF/ GOTOB

GOTO: are direct jump commands that are always used in conjunction with a jump label.

- The direction of the jump is defined with GOTOF or GOTOB. This shortens the search time.
- With the same name, the jump destination can be a label within the active program or a subprogram with the ending MPF or SPF.
- Unique names must be used when working with "speaking" identifiers.
- If the jump destination is not found, alarm 14080 "Jump destination not found" is output.

3.4.4. Programming of error queries and error loops.

a. Description:

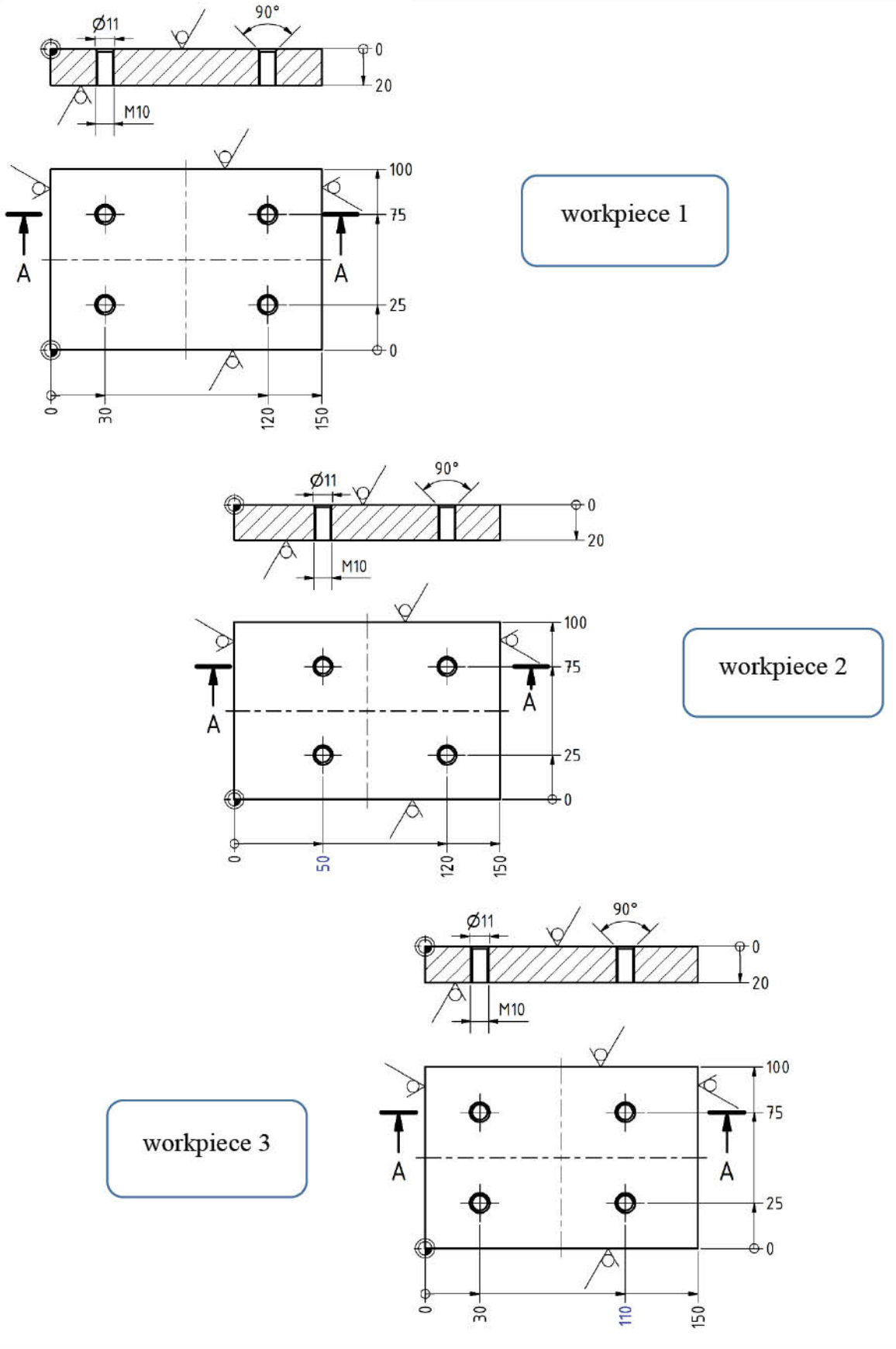
The basic program uses subprogram from the M_201 directory, to perform for three different workpieces with jump and user variable programming.

b. Objective:

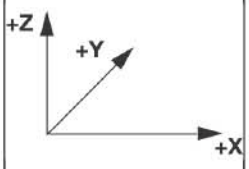
Create a new program named M_203 based on the M_201 folder, and open it in the editor.

- The operator is to be able to select between three programs with different hole distances.
- The corresponding subprogram with the appropriate hole distances is to be selected by entering a value between 1-3.
- If no value between 1-3 is selected, the program is aborted. An error message appears referring to an incorrect input.
- If no “RESET” follows the error message, the program continues with an error loop.

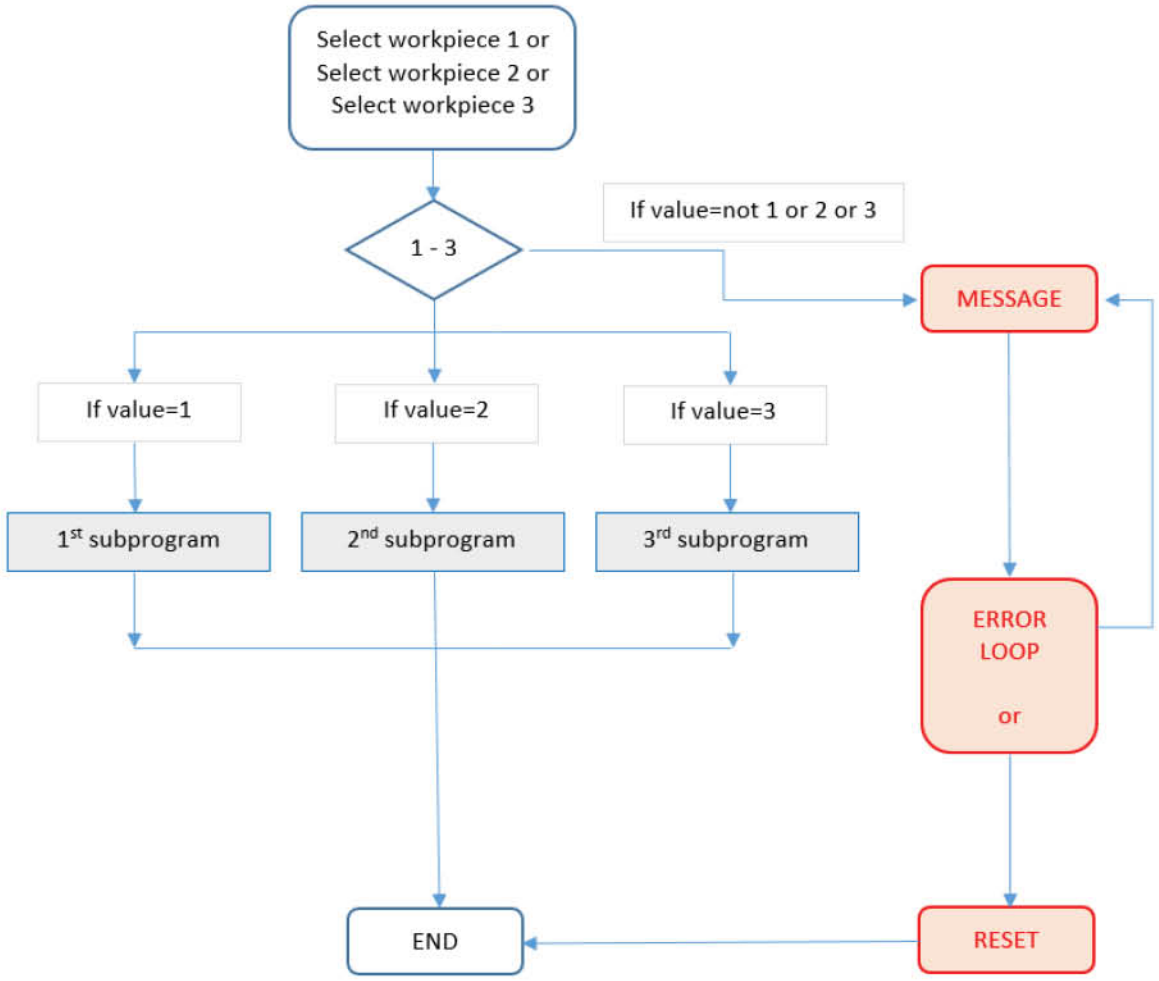
c. The technical drawing for Jumps and user variables



d. The following tools and technology data are required for programming:

Workpiece: Exercise for Jumps and User variables		Blank material: EN AW-AlCu4PbMg				Machine configuration:	
Drawing: 1(1)		Blank dimensions: Fl 150 x 100 x 20				Milling 3 axes	
	Technological data						
Tool name in OFFSET	Spot_drill_12	Deburrer_8	Cutter_32	Cutter_16	Cutter_10	Cutter_4	Drill_8,5	Tap_10
Tool Ø (mm)	12 x 90°	8 x 90°	32	16	10	4	8,5	M10
Cutting speed (m/min)	80	105	150 / 175	150 / 175	150 / 175	150 / 175	60	10
Cutting depth a_p = max.	5	3	5	8	5	2		
Cutting material	HSS	SC	SC	SC	SC	SC	HSS	HSS
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,05 / 0,04	0,03 / 0,02	0,02 / 0,01	0,1	F = P
Number of teeth N		4	2	4	4	4		
								
				✓	✓	✓		

e. Flowchart: A Flowchart is drawn up based on the task at hand.



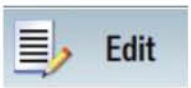
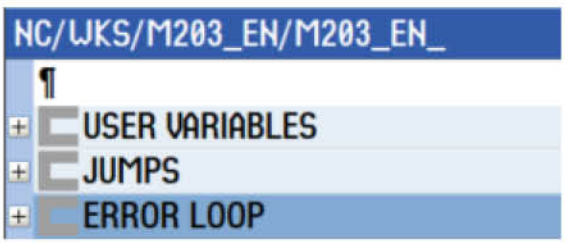
Flowchart 4.4: Technological steps for practical exercises using jump and user variables to error queries and error loops.

Note: Main program with user variables:

- A user variable must ***always be at the start of the program***. On the ***first line.***
- Therefore, it has to be defined ***outside of a block.***
- Otherwise the following alarm is displayed in the simulation:

f. Create a new G code programGuide program, like described in this unit.
Create a new program directory and call the program for example "M_203.MPF".

The following blocks are created in the program for the above example:

1.	Create a new programGuide directory and main program called “M_203”.
2. 	The “Edit” operation mode opens automatically.
3.	Press the VSK 2 “Build groups” to open the dialog menu.
4.	Input the first block structure as follow: 
5.	Enter following explanations and commands before the “USER VARIABLES” block: ;REAL: Real-numbers with decimal point ;example: DEF REAL _WORKPIECE ;INT: always identifies integers **N1 DEF INT _WORKPIECE ;*RO*** ;*RO* (Read Only) = text is read-only (capitalization) as write protection ;note: Remove write the protection with the cut out of * RO * ;ENTER NUMBER OF THE WORKPIECE IN THE NEXT BLOCK
6.	The implementation of the user variable is defined in the "USER VARIABLES" block: **N3 _WORKPIECE = 1** **N4 IF _WORKPIECE == 1 GOTO LABEL_1** **N5 IF _WORKPIECE == 2 GOTO LABEL_2** **N6 IF _WORKPIECE == 3 GOTO LABEL_3**

	N2 USER VARIABLES N3 _WORKPIECE = 1¶ ¶ N4 IF _WORKPIECE == 1 GOTO LABEL_1¶ N5 IF _WORKPIECE == 2 GOTO LABEL_2¶ N6 IF _WORKPIECE == 3 GOTO LABEL_3¶ N7 End of group
7.	The commands to display the error message or to continue the operating of one of the three workpieces is to programmed: **N8 GOTO MISTAKE_1** **N9 KEEP_GOING_1:** **N10 MSG("WORKPIECE = "<<_WORKPIECE)** **N11 M0 ;READ THE MSG!!!** **N12 M30** N8 GOTO MISTAKE_1¶ ¶ N9 KEEP_GOING_1:¶ ¶ N10 MSG("WORKPIECE = "<<_WORKPIECE)¶ ¶ N11 M0 ;READ THE MSG!!!¶ N12 M30¶
8.	The commands for the execution of “LABEL_1”, “LABEL_2” or “LABEL_3” are programmed in the "JUMPS" block: **N14 LABEL_1:** **N15 MSG("LABEL_1")** **;M0** **N16 CALL "M202_1.MPF"** **N17 GOTO KEEP_GOING_1 ;jump destination N9** **N18 ENDLABEL:** **N19 LABEL_2:** **N20 MSG("LABEL_2")** **;M0** **N21 CALL "M202_2.MPF"** **N22 GOTO KEEP_GOING_1 ; jump destination N9** **N23 ENDLABEL:** **N24 LABEL_3:** **N25 MSG("LABEL_3")** **;M0** **N26 CALL "M202_3.MPF"** **N27 GOTO KEEP_GOING_1 ;jump destination N9** **N28 ENDLABEL**:

	NC/WKS/M203_EN/M203_EN N13 JUMPS N14 LABEL_1:¶ N15 MSG("LABEL_1")¶ ;M0¶ N16 CALL "M202_1.MPF"¶ N17 GOTO KEEP_GOING_1 ;see N9¶ N18 ENDLABEL:¶ ¶ N19 LABEL_2:¶ N20 MSG("LABEL_2")¶ ;M0¶ N21 CALL "M202_2.MPF"¶ N22 GOTO KEEP_GOING_1 ;see N9¶ N23 ENDLABEL:¶ ¶ N24 LABEL_3:¶ N25 MSG("LABEL_3")¶ ;M0¶ N26 CALL "M202_3.MPF"¶ N27 GOTO KEEP_GOING_1 ;see N9¶ N28 ENDLABEL:¶ N29 End of group
9.	The commands for a wrong value in the implementation of the user variable is programmed in the "ERROR LOOP" block: **N31 MISTAKE_1:** **N32 MSG("Attention: wrong value in WORKPIECE1! Please only enter 1/2/3")** **N33 M0** **N34 MSG("cancel the ERROR with RESET ")** **N35 M0** **N36 GOTO MISTAKE_1 ;endless loop because jump destination is N31** **N37 M30** N30 ERROR LOOP N31 MISTAKE_1:¶ N32 MSG("Attention: wrong value in WORKPIECE1! Please only enter 1/2/3")¶ N33 M0¶ N34 MSG("cancel the ERROR with RESET ")¶ N35 M0¶ N36 GOTO MISTAKE_1 ;endless loop because jump destination is N31¶ N37 M30¶ N38 End of group

10.	The next step is to program the three subprogram, called in the "LABEL_1", "LABEL_2" and "LABEL_3" in the main program "M_203". Input the following block structure for "M203_1", "M203_2" and "M203_3" as next: N1 PARAMETER SET N12 START ¶ N21 CENTERING ¶ N44 DRILLING ¶ N66 TAPPING N88 PARAMETER ZERO ¶ N95 END
11.	They are only differ in the "PARAMETER SET" block when entering the R parameters.

"PARAMETER SET" block M203_1	"PARAMETER SET" block M203_2	"PARAMETER SET" block M203_3
N2 R1=30 ;DISTANCE 1 in X left **N3 R2=R1+90** ;DISTANCE 2 in X right **N4 R3=25** ;DISTANCE 3 in Y below **N5 R4=R3+50** ;DISTANCE 4 in Y up **N6 MSG("MESSAGE TEXT")** ;diplay screen messages **N7 G4 F100** ;100 dwell time ;with time speeder for the simulation **N8 MSG** ;remove screen messages **N9 MSG(" VALUE R1 = "<<R1<<" VALUE R2 = "<<R2<<" VALUE R3 =**	**N2 R1=50** ;DISTANCE 1 in X left **N3 R2=R1+70** ;DISTANCE 2 in X right **N4 R3=25** ;DISTANCE 3 in Y below **N5 R4=R3+50** ;DISTANCE 4 in Y up **N6 MSG("MESSAGE TEXT")** ;diplay screen messages **N7 G4 F100** ;100 dwell time ;with time speeder for the simulation **N8 MSG** ;remove screen messages **N9 MSG(" VALUE R1 = "<<R1<<" VALUE R2 =**	**N2 R1=30** ;DISTANCE 1 in X left **N3 R2=R1+80** ;DISTANCE 2 in X right **N4 R3=25** ;DISTANCE 3 in Y below **N5 R4=R3+50** ;DISTANCE 4 in Y up **N6 MSG("MESSAGE TEXT")** ;diplay screen messages **N7 G4 F100** ;100 dwell time ;with time speeder for the simulation **N8 MSG** ;remove screen messages **N9 MSG(" VALUE R1 = "<<R1<<" VALUE R2 = "<<R2<<" VALUE R3 =**

"<<R3<<" VALUE R4 = "<<R4) ;display calculated parameter values **N10 M0** ;programmed stop	**"<<R2<<" VALUE R3 = "<<R3<<" VALUE R4 = "<<R4)** ;display calculated parameter values **N10 M0** ;programmed stop	**"<<R3<<" VALUE R4 = "<<R4)** ;display calculated parameter values **N10 M0** ;programmed stop

12.	Input the program for the blocks “START”, “CENTERING”, “DRILLING”, “TAPPING”, “PARAMETER ZERO” and “END”. All three subprograms content exactly the same NC blocks after the input of the “PARAMETER SET” block: N12 GROUP_BEGIN(0,"START",0,0) N13 G17 N14 G54 D0 N15 WORKPIECE(,"",,"BOX",0,0,-20,-80,0,0,150,100) N16 G0 SUPA Z500 D0 M9 N17 G40 N18 G710 N19 G90 N20 GROUP_END(0,0) N21 GROUP_BEGIN(0,"CENTERING",0,0) N22 G0 SUPA Z500 D0 M9 N23 T="CENTERDRILL 12" D1 N24 M6 N25 G17 N26 G54 G90 N27 G95 S2122 F0.12 M3 M8 N28 G0 Z100 N29 G0 X=R1 Y=R3 N30 G0 Z3 N31 G1 Z-5.5 N32 G0 Z3 N33 G0 X=R2 N34 G1 Z-5.5 N35 G0 Z3 N36 G0 Y=R4 N37 G1 Z-5.5

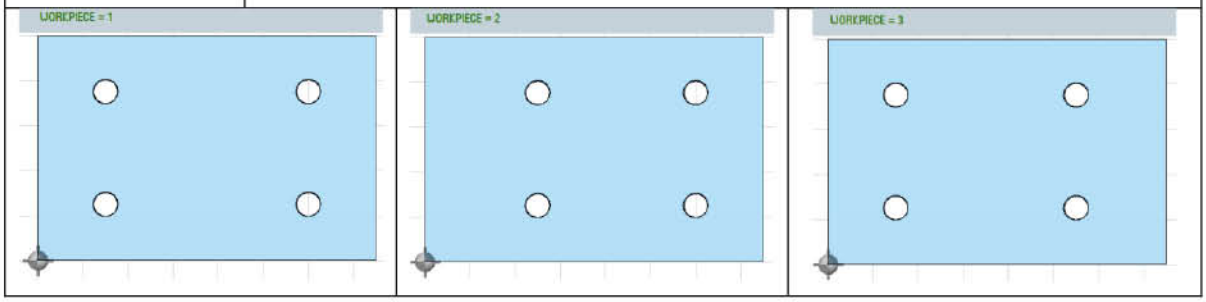
```
N38 G0 Z3
N39 G0 X=R1
N40 G1 Z-5.5
N41 G0 Z3
N42 G0 SUPA Z500 D0 M9
N43 GROUP_END(0,0)

N44 GROUP_BEGIN(0,"DRILLING",0,0)
N45 T="DRILL 8.5" D1
N46 M6
N47 G17
N48 G54 G90
N49 G95 S2247 F0.1 M3 M8
N50 G0 Z100
N51 G0 X=R1 Y=R3
N52 G0 Z3
N53 G1 Z-27
N54 G0 Z3
N55 G0 X=R2
N56 G1 Z-27
N57 G0 Z3
N58 G0 Y=R4
N59 G1 Z-27
N60 G0 Z3
N61 G0 X=R1
N62 G1 Z-27
N63 G0 Z3
N64 G0 SUPA Z500 D0 M9
N65 GROUP_END(0,0)

N66 GROUP_BEGIN(0,"TAPPING",0,0)
N67 T="THREADCUTTER M10" D1
N68 M6
N69 G17
N70 G54 G90
N71 G95 S318 M3 M8
N72 G0 Z100
N73 G0 X=R1 Y=R3
N74 G0 Z3
```

	N75 G331 Z-25 K1.5 S318 N76 G332 Z3 K1.5 S636 N77 G0 X=R2 N78 G331 Z-25 K1.5 S318 N79 G332 Z3 K1.5 S636 N80 G0 Y=R4 N81 G331 Z-25 K1.5 S318 N82 G332 Z3 K1.5 S636 N83 G0 X=R1 N84 G331 Z-25 K1.5 S318 N85 G332 Z3 K1.5 S636 N86 G0 SUPA Z500 D0 M9 N87 GROUP_END(0,0) N88 GROUP_BEGIN(0,"PARAMETER ZERO",0,0) N89 MSG N90 R1=0 N91 R2=0 N92 R3=0 N93 R4=0 N94 GROUP_END(0,0) N95 GROUP_BEGIN(0,"END",0,0) N96 G0 SUPA Z500 D0 M9 N97 M30 N98 GROUP_END(0,0)

13.	Finally check the machining with the simulation by the implementation of the user variable 1, 2 and 3.



3.5. System variables and subprogram

3.5.1. Function of system variables

3.5.1.1. Commands:

$AA_IW[Z1]	System variable
CALL	Call
RET	Return from subprogram

3.5.1.2. Function:

System variable:

- In addition to the two other variable types, the user variable (DEF) and the arithmetic variable (R variable), there are also the ***system variables.***
- These variables are provided by the ***machine manufacturer*** and have a specific notation.
- ***Machine and control states*** can be called via these system variables and integrated in an NC program.
- For example, work offsets, axis positions and tool offsets.
- These variables can be called and, partly, also written.
- A list of the system variables and their meaning can be found on the Internet using the search term "Siemens system variables", as well as in the “System variables.pdf” document. This document is free for download.

Call:

- Subprograms can be called indirectly with this command. Indirectly means that they are not in the same folder as the main program. The “Call” must be written together with the name of the subprogram and the file type.
- The subprogram must be without CALL are not in the same directory either, the fixed search hierarchy for the subroutine in the different folders searched.
- The entire path for the subprogram can also be entered. In this way, the jump to the subprogram is quicker. Example: Call "_Name.SPF”.

RET:

- This command can be used to return from a subprogram, bypassing the PLC and without interrupting the active continuous path mode.

3.5.2. Specific notation by the machine manufacturer

Some symbols of the machine manufacturer commonly used in the programming process with parameters are as follows:

- $AA_IW[Z] : Current position of Z axis
- $TC_DP1[t,d]: Tool type
- $TC_DP2[t,d]: Cutting edge position
- $TC_DP24[t,d] : Clearance angle

t	T number 1 - 32000
d	Cutting edge number / D number 1 - 32000

- $P_TOOLR: Active tool radius
- $P_TOOLNG: Number of defined tool groups
- $P_TOOLNO: Active tool number, according to the following table:

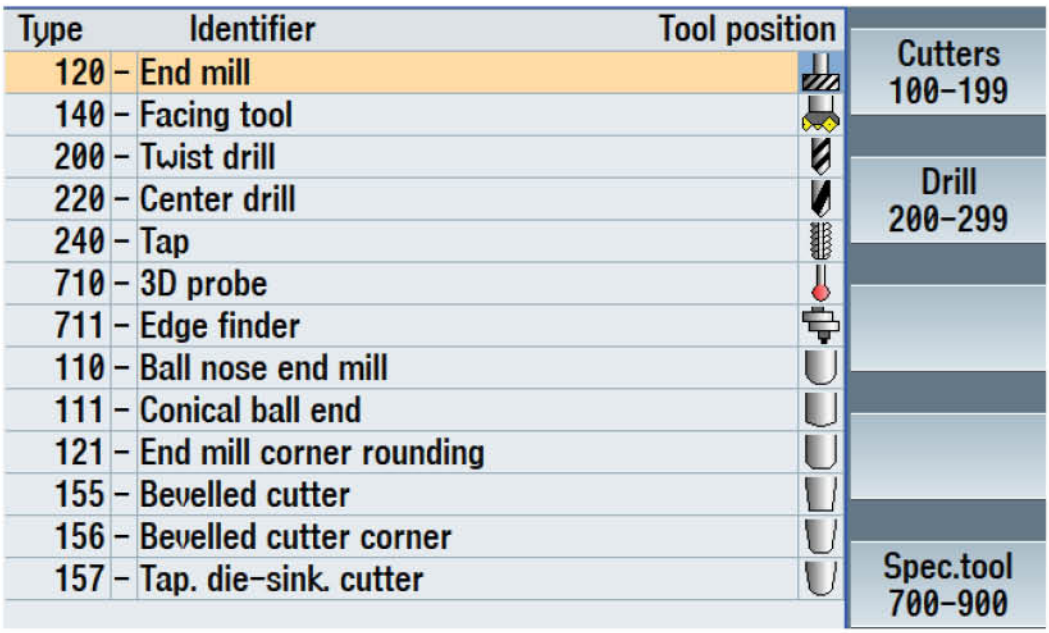


Figure 4.4: Table of tool numbers in the offset of the Milling machine

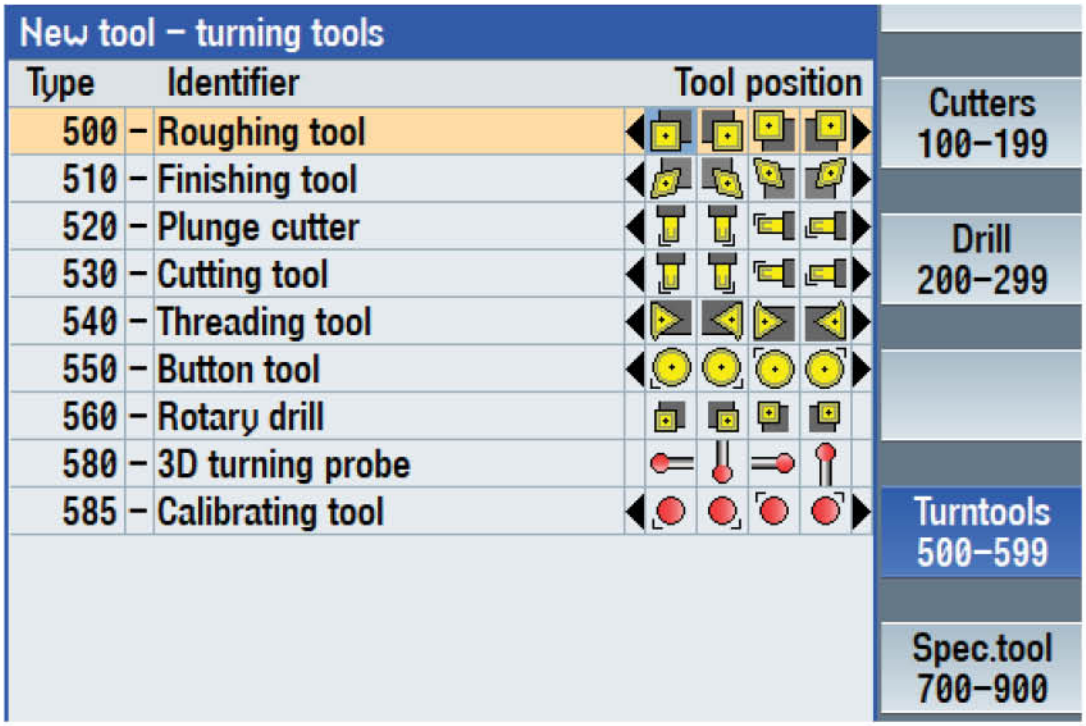


Figure 4.5: Table of tool numbers in the offset of the lathe

- $P_ATD [36,n] : Cutting edge parameters of the stated cutting edge, active tool
- $P_ATDT [36,n] : Active tool offsets of the stated cutting edge (active tool)

With: n: Parameter number 1 to 36

n = 1-25	$ TC_DP1 to $ TC_DP25
n = 26	$ TC_DPCE Number of the cutting edge (function: Unique D number)
n = 27	$ TC_DPH H number of the cutting edge (function: ISO mode)
n = 28	$ TC_DPV Tool orientation (function: Tool orientation)
n = 29	$ TC_DPV3 Component 1 of the tool orientation (function: Tool orientation)
n = 30	$ TC_DPV4 Component 2 of the tool orientation (function: Tool orientation)
n = 31	$ TC_DPV5 Component 3 of the tool orientation (function: Tool orientation)
n = 32	$ TC_DPVN3 Normal vector component 1 (function: Tool orientation)
n = 33	$ TC_DPVN4 Normal vector component 2 (function: Tool orientation)
n = 34	$ TC_DPVN5 Normal vector component 3 (function: Tool orientation)
n = 35	$ TC_DPNT Number of teeth on the cutting edge
n = 36	$ TC_DPROT Base angle of rotation of the cutting edge

- $P_GG[n]: Active G function
  - + Read active G function of G function group
  - + n: The index of the G function is supplied as described in the Programming Guide Fundamentals, Section "List of G

functions/preparatory functions". Refer to the n-indexes in the table B-1: G code table on page 153 of the “Programming Manual ISO Milling” document.

+ Example:

```
;Check for G55
IF $P_GG[8] == 3 GOTOF LABEL_G55
```

- $AC_MEAS_FRAME_SELECT: Frame selection for workpiece measurement
  + Variable for workpiece and tool measurement.
  + Variable $AC_MEAS_FRAME_SELECT is used to select the frame in which the calculated frame is entered.
  + The following values are possible:

0	$P_SETFRAME
1	$P_PARTFRAME
2	$P_EXTFRAME
10..25	$P_CHBFRAME[0..15]
50..65	$P_NCBFRAME[0..15]
100..199	$P_IFRAME
500	$P_TOOLFRAME
500: 501	$P_WPFRAME
502	$P_TRAFRAME
503	$P_PFRAME
504	$P_CYCFRAME
505	$P_RELFRAME (PCS)
506	$P_RELFRAME (ACS)
1010..1025	$P_CHBFRAME[0..15], when G500 is active
1050..1065	$P_NCBFRAME[0..15], when G500 is active
2000	$P_SETFR
2001	$P_PARTFR
2002	$P_EXTFR
2010..2025	$P_CHBFR[0..15]
2050..2065	$P_NCBFR[0..15]
2100..2199	$P_UIFR[0..99]
2500	$P_TOOLFR
2501	$P_WPFR
2502	$P_TRAFR
2504	$P_CYCFR
2505	$P_RELFR (PCS)

2506	$P_RELFR (ACS)
3010..3025	$P_CHBFR[0..15], when G500 is active
3050..3065	$P_NCBFR[0..15], when G500 is active

- $P_UIFR [n]: Settable data management frames
  - Array variable $P_UIFR[n] is used to program settable data management frames.
  - G500, G54 .. G599 can be used to activate the corresponding data management frame.
  - The data management frames are stored in SRAM and can be read in and out using the data backup feature.
  - The number of settable frames in configured with $MC_MM_NUM_USER_FRAMES.

0	G500
1	G54
2	G55
3	G56
4	G57
5	G505
6	G506
…	…
99	G599

Example:

$P_UIFR [1,Z]: activate Z axis in coordinate system according to G54

$P_UIFR [2,Z]: activate Z axis in coordinate system according to G55

3.5.3. Programme the program using system variables and subprogram.

a. Description:

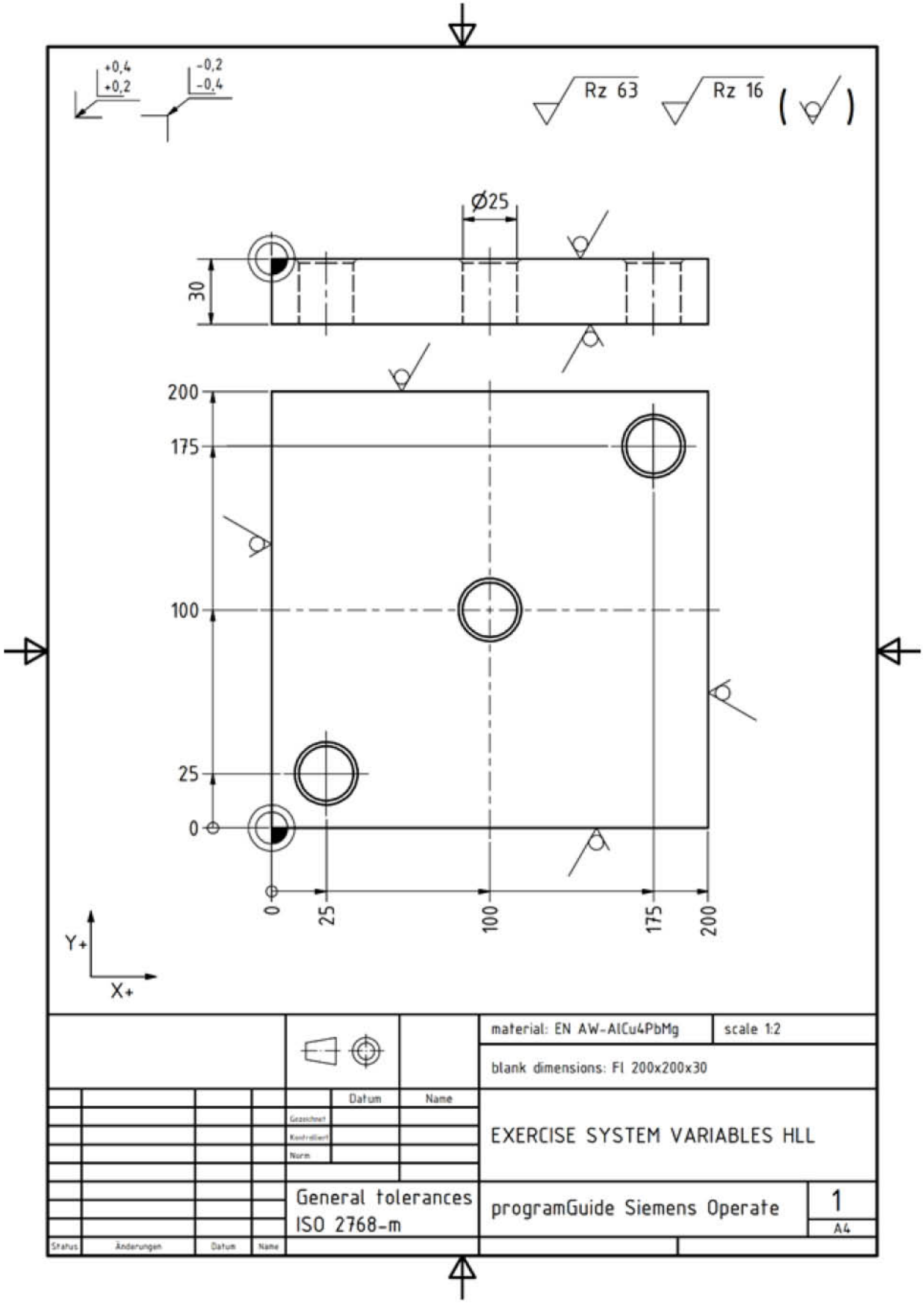
Programs are programmed using cycles with system variables that can be found in the manual for subprogram.

b. Objective:

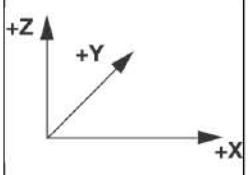
Create a new program named M_205 and open it in the editor.

- The retraction position after the machining operation is to be defined via a system variable and is to be the same for all three holes.
- The drilling positions are approached at the height of the retraction positions.
- The cycle with the system variable can be found in the manual for the system variables.
- The cycle with the system variable is to be written to a subprogram.
- This subroutine is to be defined as a subroutine and created in the “SUBPROGRAMS” folder or in the current directory “WORKPIECES”.
- The continuous-path mode is to remain active after the end of the subprogram call.

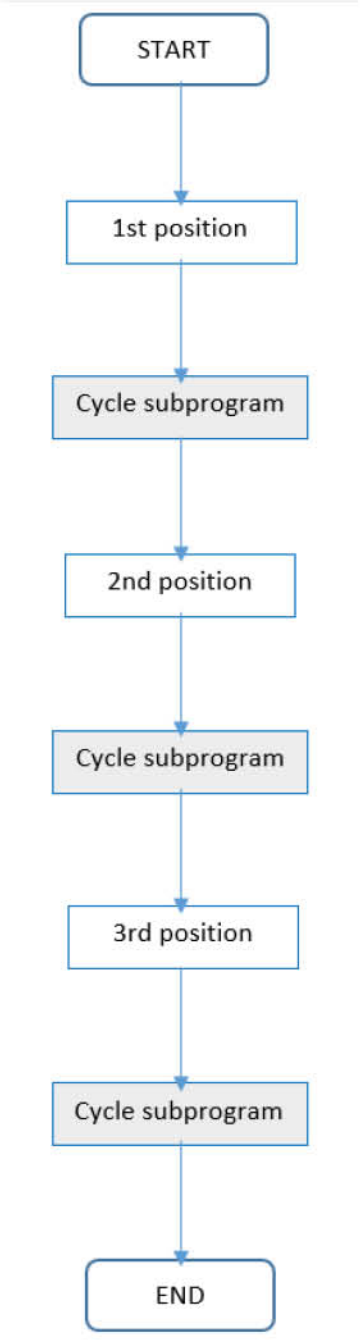
c. The technical drawing for Jumps and user variables



d. The following tools and technology data are required for programming:

Workpiece: EXERCISE SYSTEM VARIABLES HLL	Blank material: EN AW-AlCu4PbMg					Machine configuration:	
Drawing: 1(1)	Blank dimensions: Fl 200 x 200 x 30					Milling 3 axes	
	Technological data						
Tool name in OFFSET	Spot_drill_12	Deburrer_8	Cutter_32	Cutter_16	Cutter_10	Cutter_4	Solid_Drill_25
Tool Ø (mm)	12 x 90°	8 x 90°	32	16	10	4	25
Cutting speed (m/min)	80	105	150 / 175	150 / 175	150 / 175	150 / 175	120
Cutting depth a_p = max.	5	3	5	8	5	2	35
Cutting material	HSS	SC	SC	SC	SC	SC	SC
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,05 / 0,04	0,03 / 0,02	0,02 / 0,01	0,15
Number of teeth N		4	2	4	4	4	
								
				✓	✓	✓		

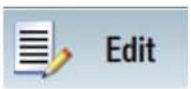
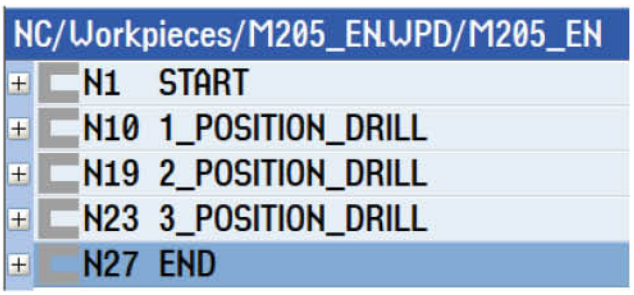
e. Flowchart: A Flowchart is drawn up based on the task at hand.



Flowchart 4.5: Technological steps for practical exercises using system variables and subprogram

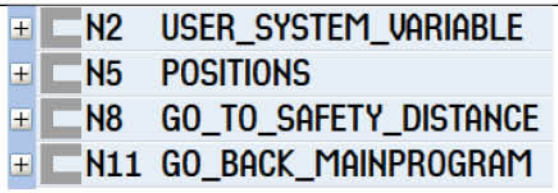
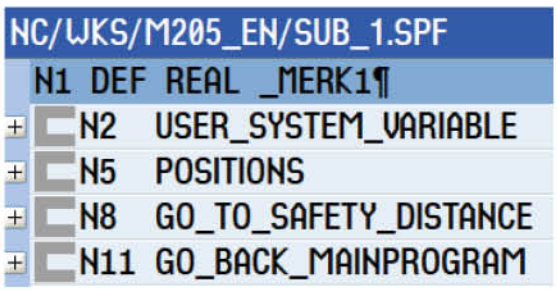
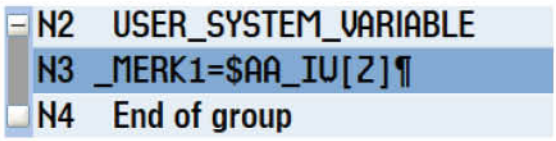
f. Create a new G code programGuide program, like described in this unit.
Create a new program directory and call the program for example "M_205.MPF".

The following blocks are created in the program for the above example:

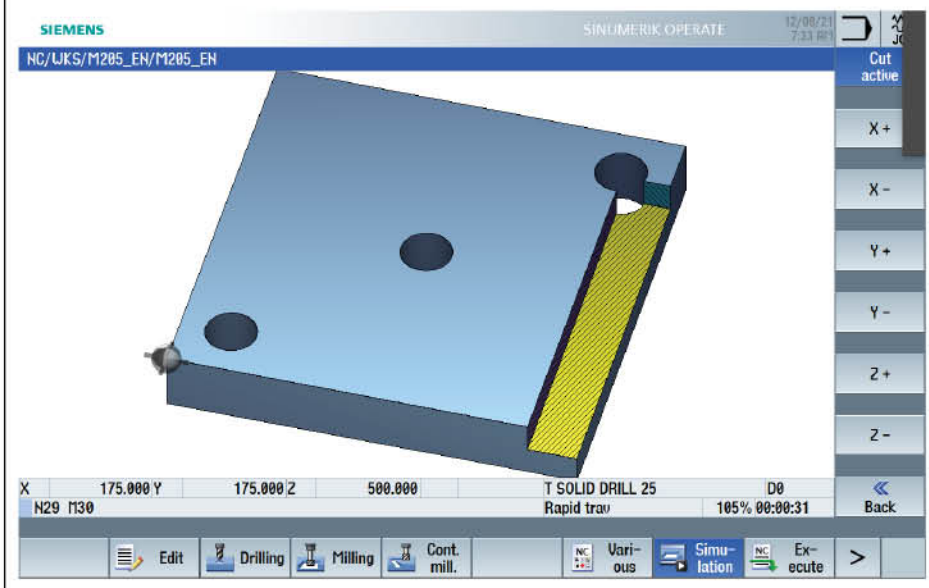
1.	Create a new programGuide directory and main program called "M_205".
2. 	The "Edit" operation mode opens automatically.
3.	Press the VSK 2 "Build groups" to open the dialog menu.
4.	Input the first block structure as follow: 
5.	The commands for the program header are programmed in the "START" block: N1 GROUP_BEGIN(0,"START",0,0) N2 G17 N3 G54 D0 N4 WORKPIECE(,"",,"BOX",48,0,-30,-80,0,0,200,200) N5 G0 SUPA Z500 D0 M9 N6 G40 N7 G710 N8 G90 N9 GROUP_END(0,0)

	N1 START N2 G17¶ N3 G54 D0¶ ¶ N4 WORKPIECE(,"",,"BOX",48,0,-30,-80,0,0,200,200)¶ N5 G0 SUPA Z500 D0 M9¶ N6 G40¶ N7 G710¶ N8 G90¶ N9 End of group
6.	The commands for the first drill hole is programmed in the "1_POSITION_DRILL " block with the subprogram “SUB_1”: N10 GROUP_BEGIN(0,"1_POSITION_DRILL",0,0) N11 T="SOLID DRILL 25" N12 M6 D1 N13 G95 S1910 F0.15 M3 M8 N14 G0 Z100 N15 G0 X25 Y25 N16 G0 Z2 N17 SUB_1 N18 GROUP_END(0,0) N10 1_POSITION_DRILL N11 T="SOLID DRILL 25"¶ N12 M6 D1¶ N13 G95 S1910 F0.15 M3 M8¶ N14 G0 Z100¶ N15 G0 X25 Y25¶ N16 G0 Z2¶ N17 SUB_1¶ N18 End of group
7.	The commands for the second drill hole is programmed in the "2_POSITION_DRILL " block with the subprogram “SUB_1”: N19 GROUP_BEGIN(0,"2_POSITION_DRILL",0,0) N20 G0 X100 Y100 N21 SUB_1 N22 GROUP_END(0,0)

	N19 2_POSITION_DRILL N20 G0 X100 Y100¶ N21 SUB_1¶ N22 End of group
8.	The commands for the third drill hole is programmed in the "3_POSITION_DRILL " block with the subprogram “SUB_1”: N23 GROUP_BEGIN(0,"3_POSITION_DRILL",0,0) N24 G0 X175 Y175 N25 SUB_1 N26 GROUP_END(0,0) N23 3_POSITION_DRILL N24 G0 X175 Y175¶ N25 SUB_1¶ N26 End of group
9.	The commands for the end of the program are programmed in the "END" block: N27 GROUP_BEGIN(0,"END",0,0) N28 G0 SUPA Z500 D0 M9 N29 M30 N30 GROUP_END(0,0) N27 END N28 G0 SUPA Z500 D0 M9¶ N29 M30¶ N30 End of group
10.	This subroutine is to be defined as a subroutine and created in the “SUBPROGRAMS” folder or in the current directory “WORKPIECES”. New G code program Type Subprogram SPF Name SUB_1
11.	After creating a G code program in the folder “Subprograms” with the correct name from the main program “SUB_1” a block structure is defined in the first step:

	
12.	The system variable is programmed in front of the program blocks: N1 DEF REAL _MERK1 
13.	In the block “USER_SYSTEM_VARIABLE” the link programmed between user variable and system variable: N2 GROUP_BEGIN(0,"USER_SYSTEM_VARIABLE",0,0) N3 _MERK1=$AA_IW[Z] N4 GROUP_END(0,0) 
14.	The positions for the depth of the holes are programmed in the "POSITIONS" block: N5 GROUP_BEGIN(0,"POSITIONS",0,0) N6 G1 Z-33 N7 GROUP_END(0,0)
15.	The retraction to the safety distance is programmed in the "GO_TO_SAFETY_DISTANCE" block: N8 GROUP_BEGIN(0,"GO_TO_SAFETY_DISTANCE",0,0) N9 G0 Z=_MERK1 N10 GROUP_END(0,0)

16.	Finally, the quick jump to main program is programmed in the "GO_BACK_MAINPROGRAM" block: N11 GROUP_BEGIN(0,"GO_BACK_MAINPROGRAM",0,0) N12 RET N13 GROUP_END(0,0) N11 GO_BACK_MAINPROGRAM N12 RET¶ N13 End of group
17.	Finally check the machining with the simulation:



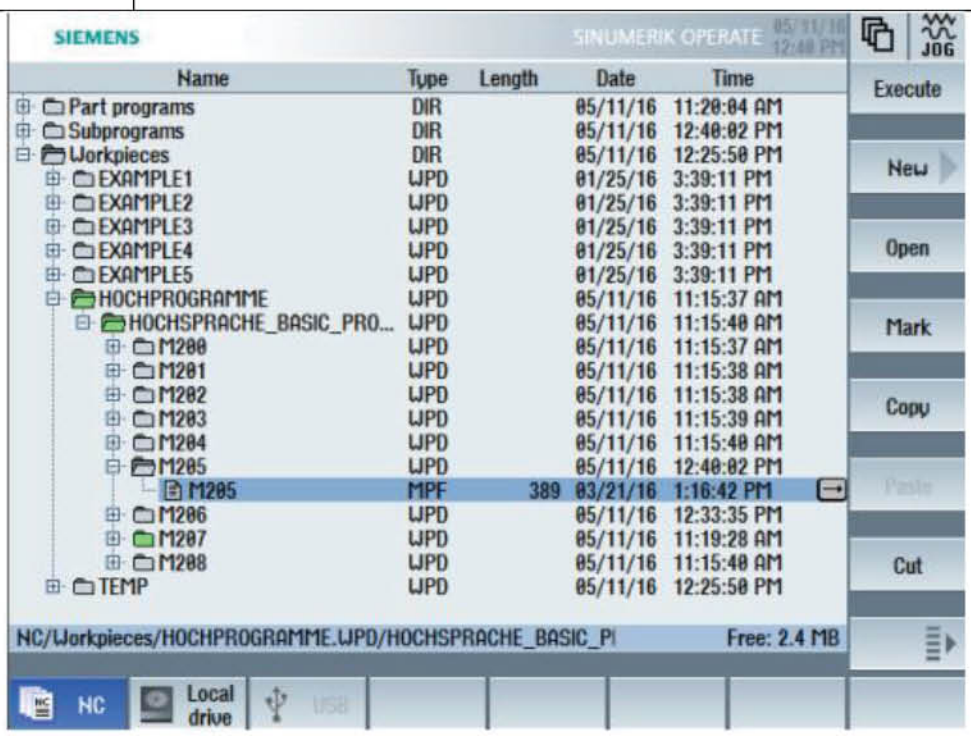


Figure 4.6: Main program memory is stored in the Workpieces folder with the extension MPF

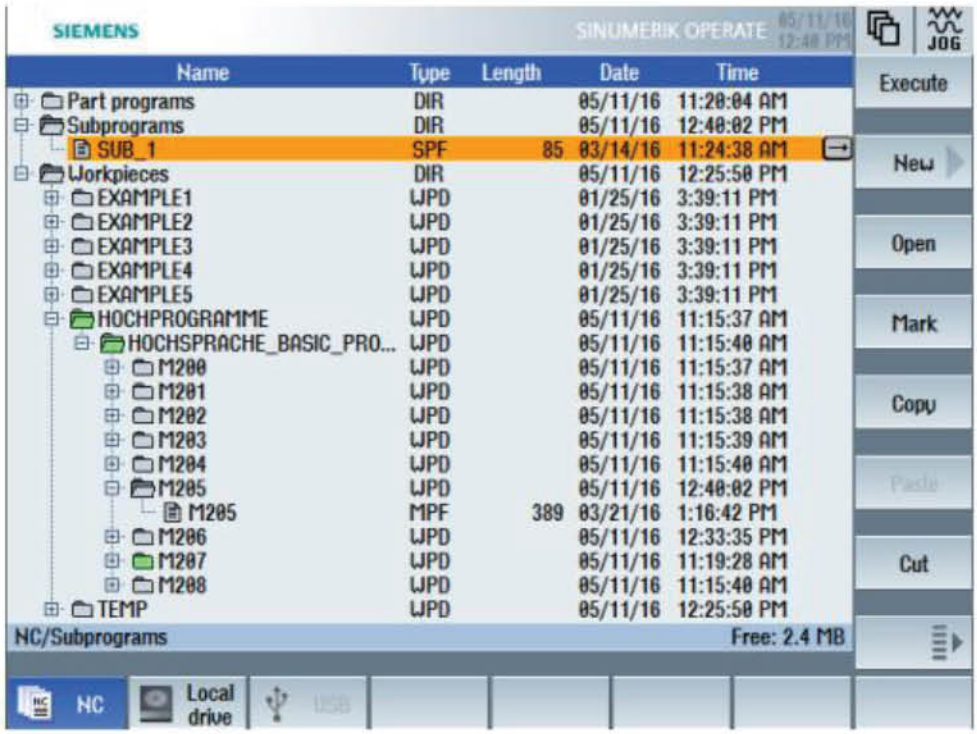


Figure 4.7: Subprogram memory of cycle is stored in the Subprograms folder with the extension SPF

3.6. System variables and safety query

3.6.1. Commands

$TC_DP1[T,D]	Tool type
$P_TOOLNO	Active tool number
$P_TOOL	Active tool edge

3.6.2. Function of system variables and safety query

a. **$TC_DP1[T,D]:** The tool type is defined via this ***system variable.***

- In SINUMERIK, tools are divided into ***tool types.***
- Each tool type is assigned a 3-digit number. The first digit assigns the tool type to a tool group with the appropriate technology.

Tool type	Tool group
1	Cutters
2	Drills
3	Reserved
4	Grinding tools
5	Turning tools
6	Reserved
7	Special tools such as the slotting saw

Milling tools - Group 1 "Cutters" includes the following tool types:

100 - Milling tool according to CLDATA (Cutter Location Data)
110 - Ballhead cutter (cylindrical die-sinking cutter)
111 - Ballhead cutter (tapered die-sinking cutter)
120 - End mill (without corner rounding)
121 - End mill (with corner rounding)
130 - Angle head cutter (without corner rounding)
131 - Angle head cutter (with corner rounding)
140 - Facing tool
145 - Thread milling cutter
150 - Side mill
151 - Saw
155 - Bevel cutter (without corner rounding)
156 - Bevel cutter (with corner rounding)
157 - Tapered die-sinking cutter
160 - Drill and thread milling cutter

Drills - Group 2 "Drills" includes the following tool types:

200 - Twist drill
205 - Drill
210 - Boring bar
220 - Center drill
230 - Countersink
231 - Counterbore
240 - Regular thread tap
241 - Fine thread tap
242 - Whitworth thread tap
250 - Reamer

Turning tools - Group 5 "Turning tools" includes the following tool types:

500 - Roughing tool
510 - Finishing tool
520 - Plunge cutter
530 - Parting tool
540 - Threading tool
550 - Button tool / shaping tool (TOOLMAN)
560 - Rotary drill (ECOCUT)
580 - Probe with cutting edge position parameter

Note: Only the groups that are most important for drilling, turning and milling are listed in this this document.

The tool number is visible in SINUMERIK Operate when the corresponding tool symbol "TYPE" is selected in the tool management. It opens a list with the type numbers of the possible tools.

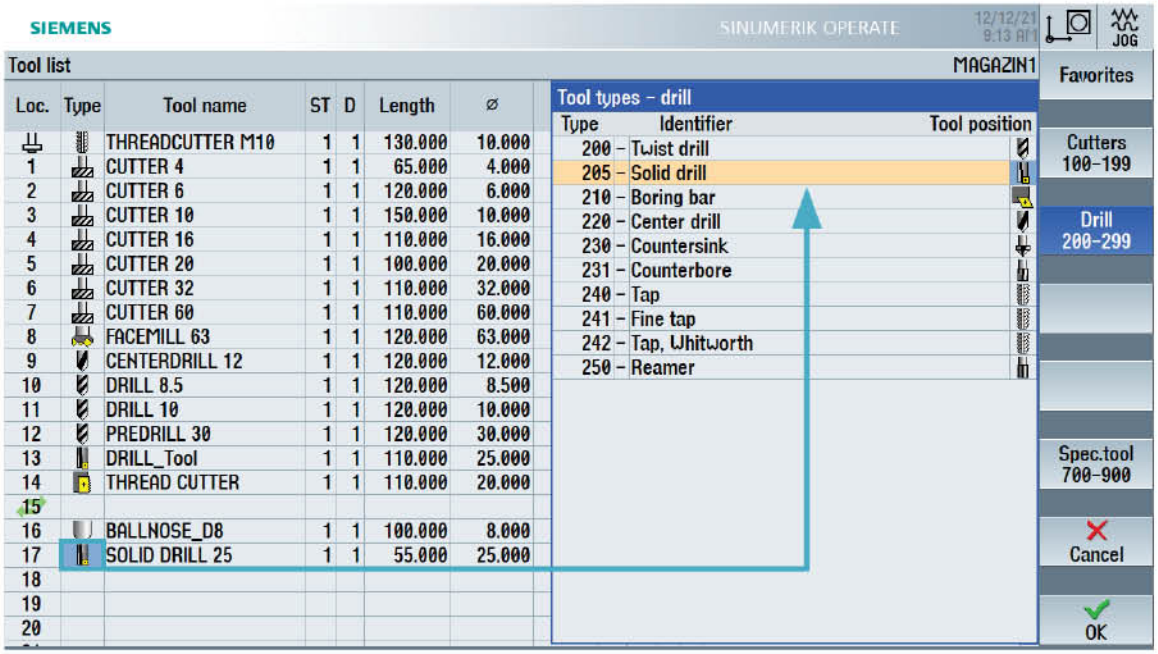


Figure 4.8: The tool type is shown when the tool symbol in the corresponding "TYPE" section is selected

b. $P_TOOLNO: This ***system variable*** describes the active tool number, i.e. the active tool in the NC:

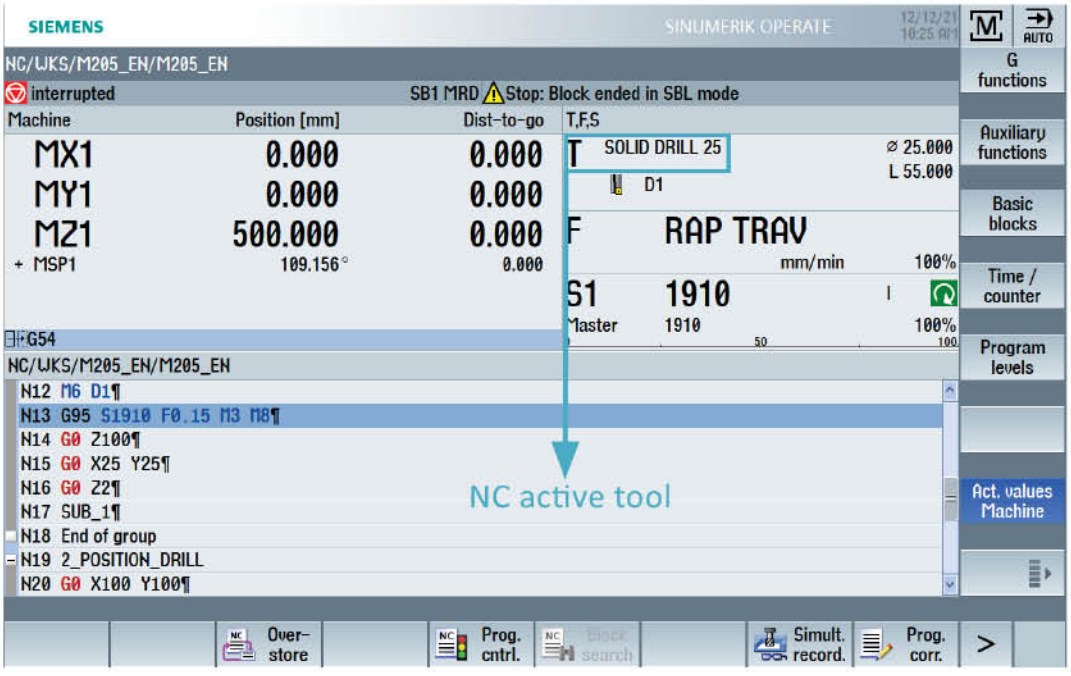


Figure 4.9: Showing the number of active tools on the working area

c. $P_Tool: This ***system variable*** describes the active tool edge and its active offsets.

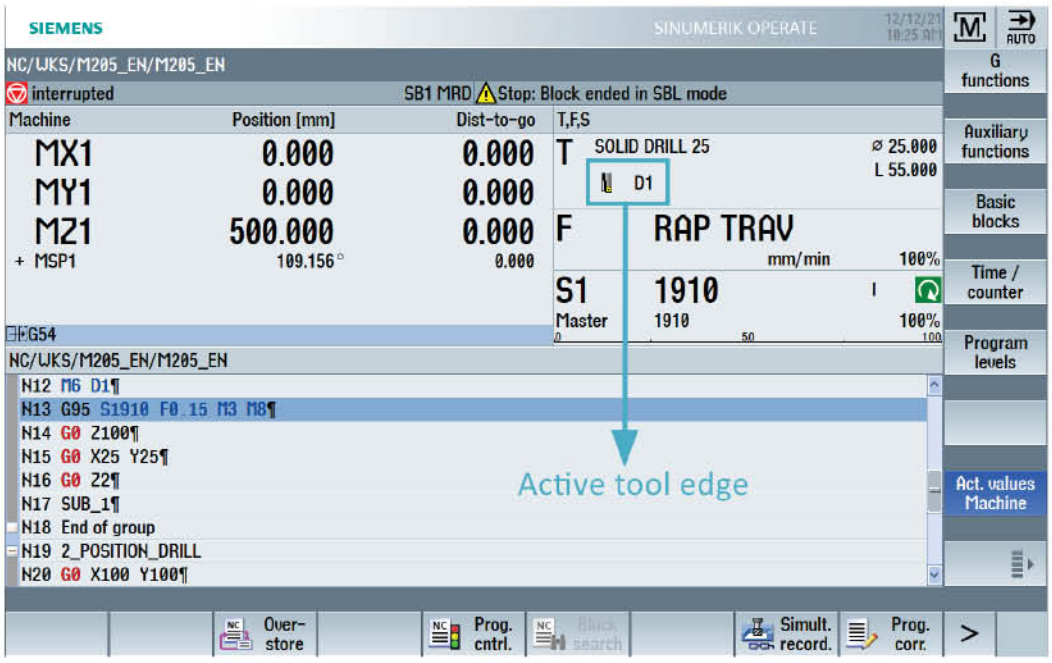


Figure 4.10: Shows the number of active tool edge on the work area

3.6.3. Programming with system variables, tool queries and error loops.

a. Description:

The program is programmed using the programs available from the folder M_205, to perform tool query and error loop through system variables.

b. Objective:

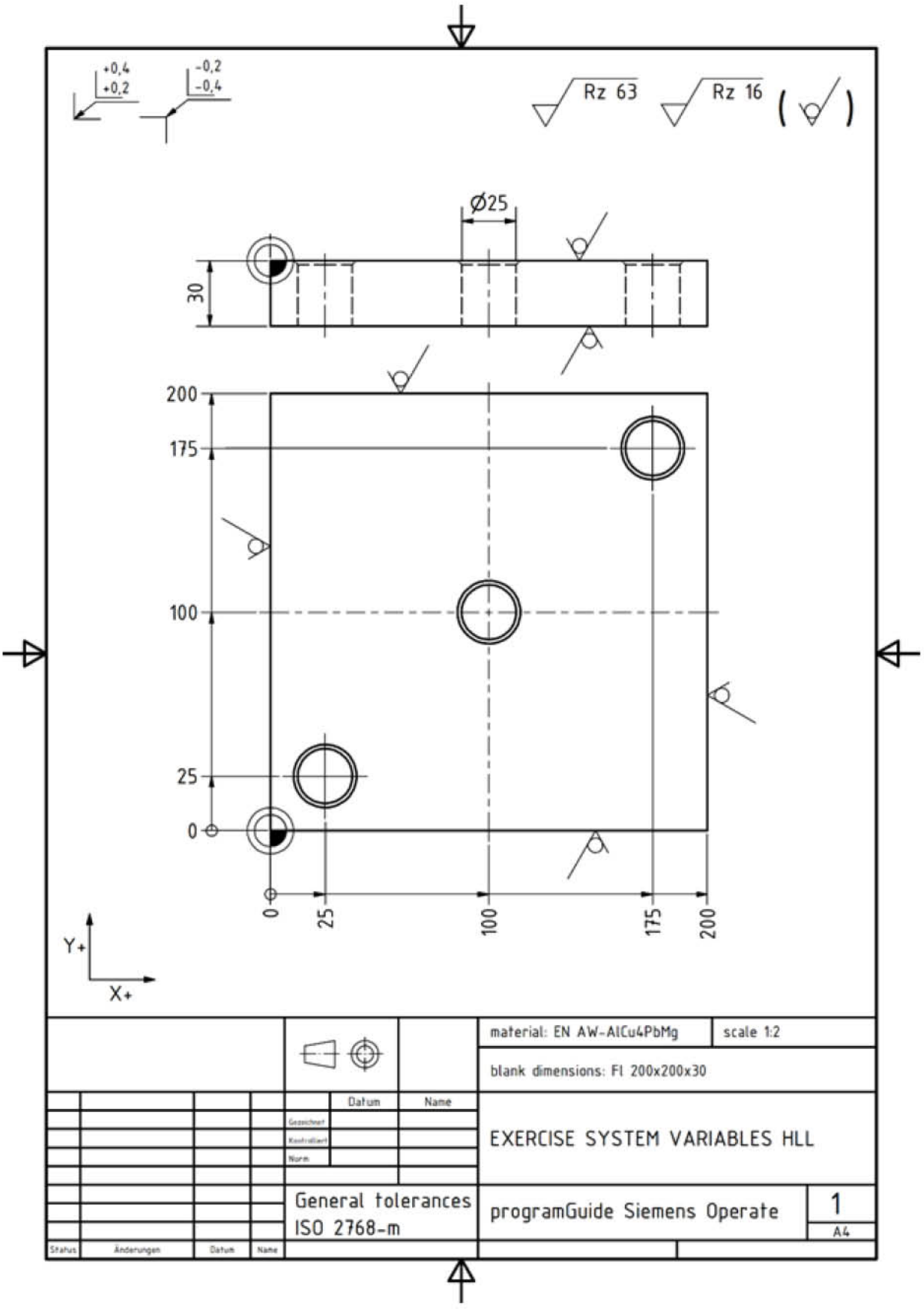
Create a new program named M_206 based on the M_205 folder, and open it in the editor.

Query the tool type, the current tool in the NC and the current tool edge via the system variables.

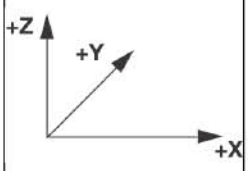
Create a loop if the tool type for machining is incorrect, the program is to jump to a loop which the operator can only leave by pressing program “Reset".

Displays a message text informing the operator that an incorrect tool is active.

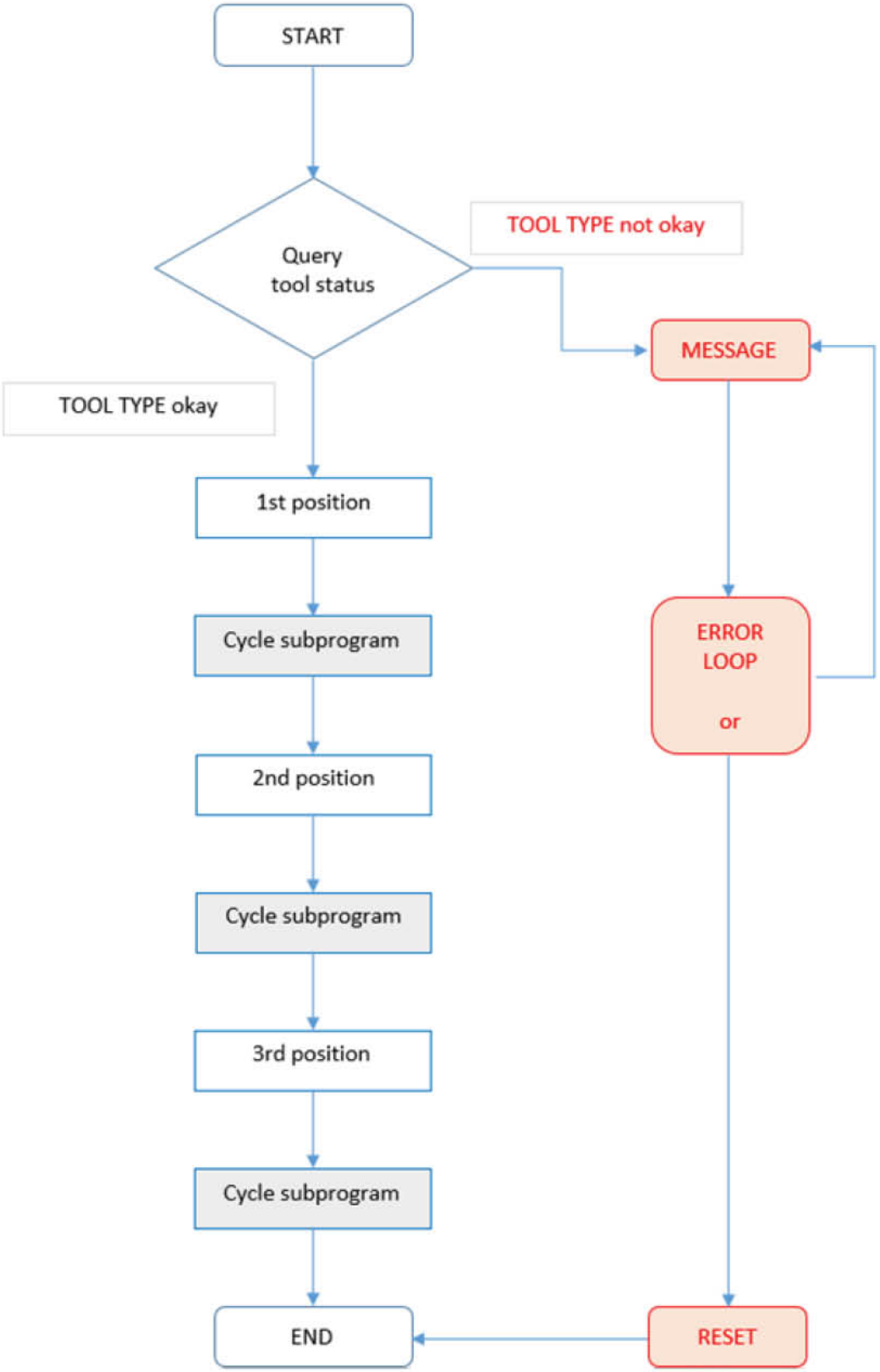
c. The technical drawing for tool queries and error loops, EXERCISE SYSTEM VARIABLES HLL



d. The following tools and technology data are required for programming:

Workpiece: EXERCISE SYSTEM VARIABLES HLL		Blank material: EN AW-AlCu4PbMg				Machine configuration:		
Drawing: 1(1)		Blank dimensions: FI 200 x 200 x 30				Milling 3 axes		
	Technological data							
Tool name in OFFSET	Spot_drill_12	Deburrer_8	Cutter_32	Cutter_16	Cutter_10	Cutter_4	Solid_Drill_25	
Tool Ø (mm)	12 x 90°	8 x 90°	32	16	10	4	25	
Cutting speed (m/min)	80	105	150 / 175	150 / 175	150 / 175	150 / 175	120	
Cutting depth a_p = max.	5	3	5	8	5	2	35	
Cutting material	HSS	SC	SC	SC	SC	SC	SC	
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,05 / 0,04	0,03 / 0,02	0,02 / 0,01	0,15	
Number of teeth N		4	2	4	4	4		
								
				✓	✓	✓		

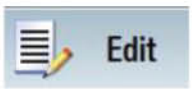
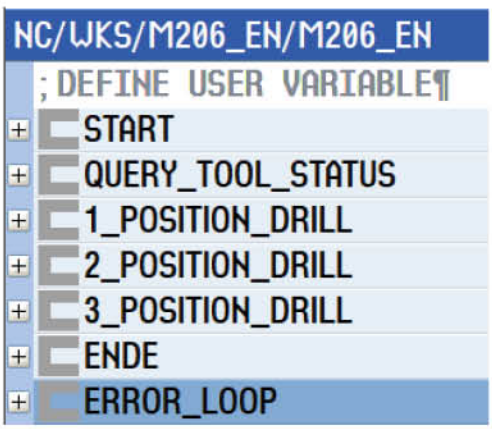
e. Flowchart: A Flowchart is drawn up based on the task at hand.



Flowchart 4.6: Technological steps for practical exercises using system variables to tool queries and error loops

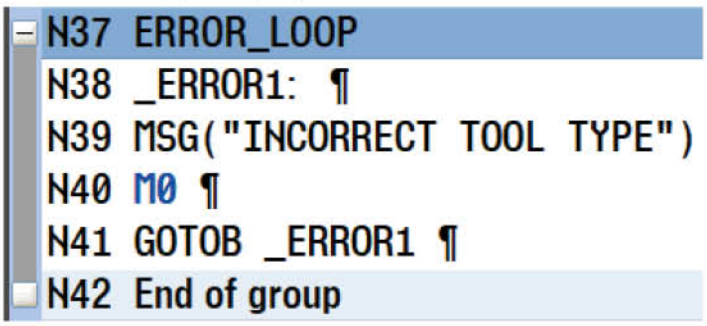
f. Create a new G code programGuide program, like described in this unit.
Create a new program directory and call the program for example "M_206.MPF".

The following blocks are created in the program for the above example:

1.	Create a new programGuide directory and main program called “M_206”.
2. 	The “Edit” operation mode opens automatically.
3.	Press the VSK 2 “Build groups” to open the dialog menu.
4.	Input the first block structure as follow: 
5.	The user variable is defined in the first lines before the “START” block: ;DEFINE USER VARIABLE: ;_TOOLTYPE= VARIABLE FOR TOOL TYPE WITH CURRENT ;TOOL NUMBER AND CUTTING EDGE NUMBER N1 DEF INT _TOOLTYPE
6.	The commands for the program header are programmed in the "START" block: N2 GROUP_BEGIN(0,"START",0,0) N3 G17 N4 G54 D0 N5 WORKPIECE(,"",,"BOX",0,0,-30,-80,0,0,200,200) N6 G0 SUPA Z500 D0 M9 N7 G40 N8 G710 N9 G90 N10 GROUP_END(0,0)

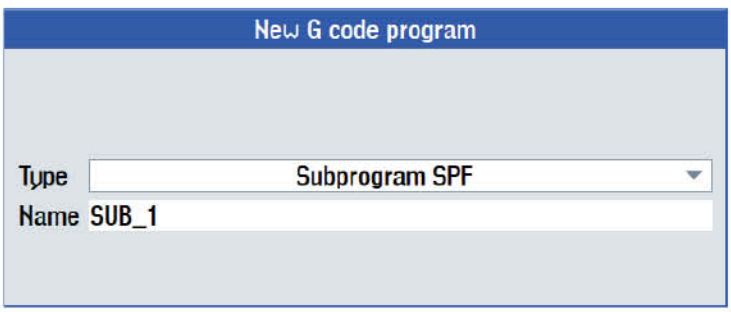
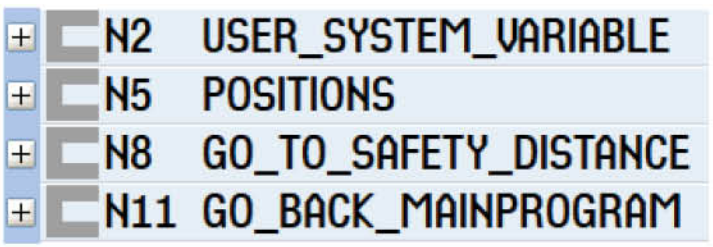
	N2 START N3 G17 ¶ N4 G54 D0 ¶ N5 WORKPIECE(,"",,"BOX",0,0,-30,-80,0,0,200,200)' N6 G0 SUPA Z500 D0 M9 ¶ N7 G40 ¶ N8 G710 ¶ N9 G90 ¶ N10 End of group
7.	The commands for the query of the current tool is programmed in the "QUERY_TOOL_STATUS" block with the system variable for the tool type, the current tool in the NC and the current tool edge: N11 GROUP_BEGIN(0,"QUERY_TOOL_STATUS",0,0) N12 T="SOLID DRILL 25" N13 M6 D1 N14 G95 S1910 F0.15 M3 M8 N15 _TOOLTYPE = $TC_DP1[$P_TOOLNO,$P_TOOL] N16 IF (_TOOLTYPE <200) OR (_TOOLTYPE >299) GOTOF _ERROR1 N17 GROUP_END(0,0) N11 QUERY_TOOL_STATUS N12 T="SOLID DRILL 25"¶ N13 M6 D1 ¶ N14 G95 S1910 F0.15 M3 M8¶ N15 _TOOLTYPE = $TC_DP1[$P_TOOLNO,$P_TOOL] ¶ N16 IF (_TOOLTYPE <200) OR (_TOOLTYPE >299) GOTOF _ERROR1 N17 End of group
8.	The commands for the first drill hole is programmed in the "1_POSITION_DRILL " block with the subprogram “SUB_1”: N18 GROUP_BEGIN(0,"1_POSITION_DRILL",0,0) N19 G0 Z100 N20 G0 X25 Y25 N21 G0 Z2 N22 SUB_1 N23 GROUP_END(0,0)

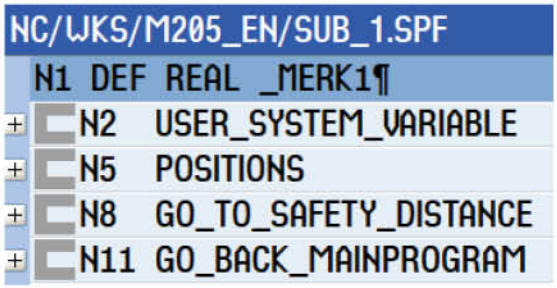
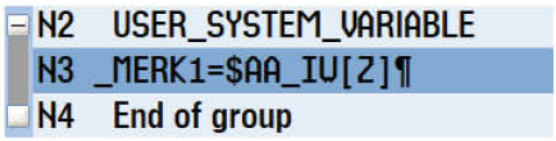
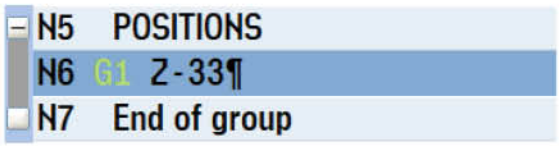
	N18 1_POSITION_DRILL N19 G0 Z100¶ N20 G0 X25 Y25¶ N21 G0 Z2¶ N22 SUB_1¶ N23 End of group
9.	The commands for the second drill hole is programmed in the "2_POSITION_DRILL " block with the subprogram “SUB_1”: N24 GROUP_BEGIN(0,"2_POSITION_DRILL",0,0) N25 G0 X100 Y100 N26 SUB_1 N27 GROUP_END(0,0) N24 2_POSITION_DRILL N25 G0 X100 Y100¶ N26 SUB_1¶ N27 End of group
10.	The commands for the third drill hole is programmed in the "3_POSITION_DRILL " block with the subprogram “SUB_1”: N28 GROUP_BEGIN(0,"3_POSITION_DRILL",0,0) N29 G0 X175 Y175 N30 SUB_1 N31 GROUP_END(0,0) N28 3_POSITION_DRILL N29 G0 X175 Y175¶ N30 SUB_1¶ N31 End of group
11.	The commands for the end of the program are programmed in the "END" block: N32 GROUP_BEGIN(0,"ENDE",0,0) N33 G0 SUPA Z500 D0 M9 N34 T0 M6 N35 M30 N36 GROUP_END(0,0) N32 ENDE N33 G0 SUPA Z500 D0 M9¶ N34 T0 M6¶ N35 M30¶ N36 End of group

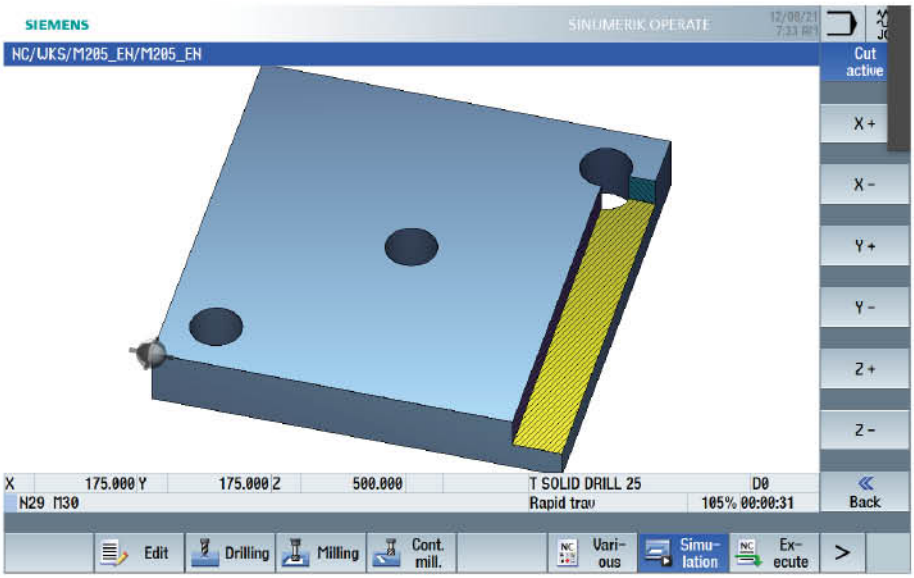
12.	The program loop commands are programmed in the “ERROR_LOOP” block. N37 GROUP_BEGIN(0,"ERROR_LOOP",0,0) N38 _ERROR1: N39 MSG("INCORRECT TOOL TYPE") N40 M0 N41 GOTOB _ERROR1 N42 GROUP_END(0,0) 

The subprogram “SUB_1” from “M205” is already saved in the “Subprogram” directory.

If not it has to be programmed, the following blocks are created in the program:

13.	This subroutine is to be defined as a subroutine and created in the “SUBPROGRAMS” folder or in the current directory “WORKPIECES”. 
14.	After creating a G code program in the folder “Subprograms” with the correct name from the main program “SUB_1” a block structure is defined in the first step: 

15.	The system variable is programmed in front of the program blocks: N1 DEF REAL _MERK1 
16.	In the block “USER_SYSTEM_VARIABLE” the link programmed between user variable and system variable: N2 GROUP_BEGIN(0,"USER_SYSTEM_VARIABLE",0,0) N3 _MERK1=$AA_IW[Z] N4 GROUP_END(0,0) 
17.	The positions for the depth of the holes are programmed in the "POSITIONS" block: N5 GROUP_BEGIN(0,"POSITIONS",0,0) N6 G1 Z-33 N7 GROUP_END(0,0) 
18.	The retraction to the safety distance is programmed in the "GO_TO_SAFETY_DISTANCE" block: N8 GROUP_BEGIN(0,"GO_TO_SAFETY_DISTANCE",0,0) N9 G0 Z=_MERK1 N10 GROUP_END(0,0)
19.	Finally, the quick jump to main program is programmed in the “GO_BACK_MAINPROGRAM” block: N11 GROUP_BEGIN(0,"GO_BACK_MAINPROGRAM",0,0) N12 RET N13 GROUP_END(0,0)

	N11 GO_BACK_MAINPROGRAM N12 RET¶ N13 End of group
20.	Finally check the machining with the simulation with a right tool for drilling: 
21.	Finally check the machining with the simulation with a wrong tool for milling, for this risen exchange in NC block “N12” the drilling tool with a milling tool: N11 QUERY_TOOL_STATUS N12 T="CUTTER 10"¶ N13 M6 D1 ¶ N14 G95 S1910 F0.15 M3 M8¶ N15 _TOOLTYPE = $TC_DP1[$P_TOOLNO,$P_TOOL] ¶ N16 IF (_TOOLTYPE <200) OR (_TOOLTYPE >299) GOTOF _ERROR1 ¶ N17 End of group Check the error message with using the simulation:

3.7. Frame concept

3.7.1. Function of frames

Frame is the common term for a geometric expression describing an arithmetic operation, for example, translation or rotation.

For SINUMERIK CNCs, the frame in the CNC program transfers from one Cartesian coordinate system to another, and represents the spatial description of the workpiece coordinate system.

The following are possible:

- Basic frames

  Coordinate transformation from basic coordinate system BCS into basic zero system BZS
- Adjustable frames

  Work offsets using G54 to G57/G505 to G599
- Programmable frames

  Definition of workpiece coordinate system WCS

The following instructions are used to program these options:

- TRANS programmable work offset
- ROT rotation in space or in a plane
- SCALE scaling (scale factor)
- MIRROR mirroring

The instructions can also be used several times within one program. Existing offsets can either be overwritten or new ones can be added.

Additive frame instructions:

- ATRANS additive programmable work offset
- AROT additive rotation in space or in a plane
- ASCALE scale factor (multiplication)
- AMIRROR repeated mirroring

Benefits:

The frame concept makes it possible to transform Cartesian coordinate systems very simply by offsetting, rotating, scaling and mirroring.

3.7.2. Explanation of the commands and variables used

$P_GG[n]	Read active G function of the G function group
$P_UIFR	Active settable frame
CTRANS	Coarse offset
CFINE	Fine offset

Function

a. $P_UIFR:

- This system variable allows direct access to the values of ***settable work offsets.***

b. b. $P_UIFR[N]:

- The number in the square brackets indicates which work offset is to be addressed.
- The command that describes what is to be done with the values of the work offset is after the system variable.

c. CTRANS:

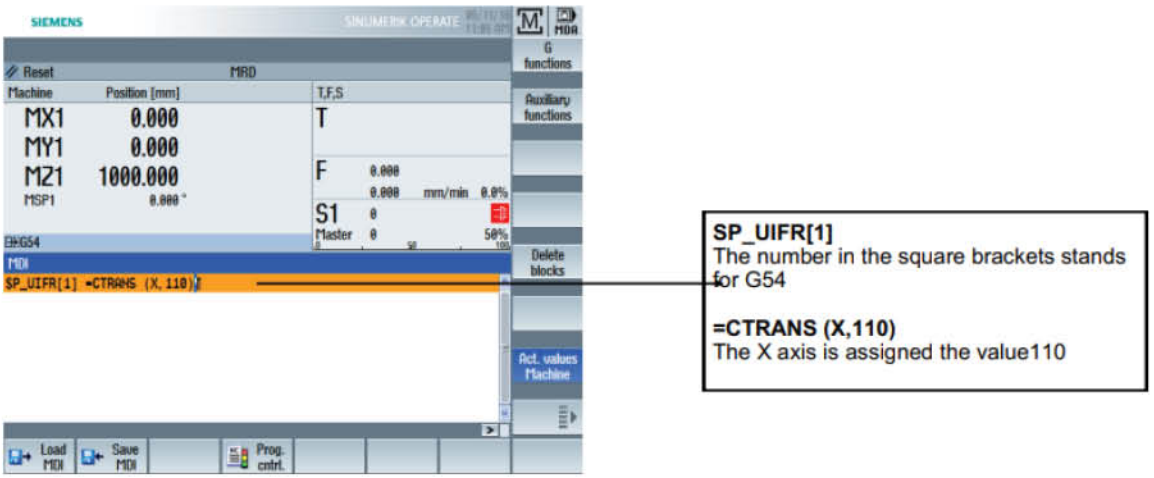
- A work offset (***coarse offset)*** can be written with this frame function.
- First the axis is defined, and then the value in the next parameter.
- The CTRANS command cannot be used on its own, ***only in conjunction with a system variable.***
- Attention: If only one axis is assigned a value, ***all others are set to zero.***

Example: Active values in work offset G54

Work offset – active [°]

	X	Y	Z	SP1
Act.value Machine	0.000	0.000	1000.000	0.000
G54	1.000	2.000	3.000	0.000
Total WO	1.000	2.000	3.000	0.000

Input in MDI: **$P_UIFR[1] =CTRANS (X,110)**



By pressing

, the value for "X" was taken into work offset G54.

The values for all other axes were set to zero.

Work offset – active [°]				
	X	Y	Z	SP1
Act.value Machine	0.000	0.000	1000.000	0.000
G54	110.000	0.000	0.000	0.000
Total WO	110.000	0.000	0.000	0.000

d. CFINE:

- The behavior with CFINE is similar as for CTRANS.
- All active values for the fine offset are set to zero if only one axis was assigned a value. See example above.
- CFINE is used when you do not want to change the values in the work offsets (coarse offsets).
- You can use access levels to set, for example, that an operator can only access the values of the fine offset.
- Example: An operator can only set the workpiece zero in a value range of 1mm using the fine offset to change.
- The maximum size in the fine offset can also be specified.

Attention:

- The values in CFINE are always active for the work offset and are included in the calculation for the respective axis.
- It is essential that the fine offset is also ***taken into account for the work offset.***
- If not all values in a frame are to be deleted, but rather only those for one axis, this can be achieved with the ***TR*** command in the frame.

Example: **$P_UIFR[1,X,TR]=25**

In the G54 frame, only the value for the X axis is changed to 25. ***All other values remain unchanged.***

3.7.3. Classification of the G-functions into so-called function groups

a. $P_GG[n]

- G functions are divided into so-called ***function groups.***
- This prevents commands from a group being called twice in a block.
- Example: The G commands G0, G1, G2, G3 are in function group 1.
- Logically, they cannot be written in one block because they belong to the same function group. (SINUMERIK Fundamentals Manual, Chapter 12 Tables, List of instructions).
- The active G code groups and their contents can be seen at all times in SINUMERIK Operate.

In the “Machine” operating area,

there are two options:

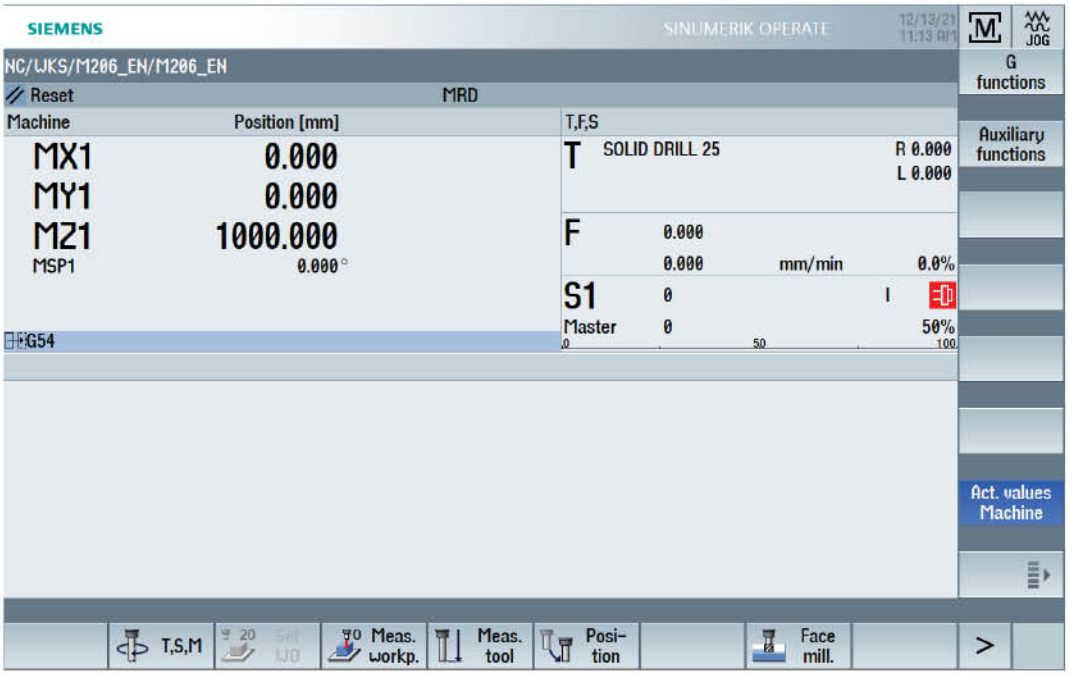


Figure 4.11: Machine operating area

Option 1: G functions

By pressing the “G functions” softkey, the most important G function groups with the active command are displayed.

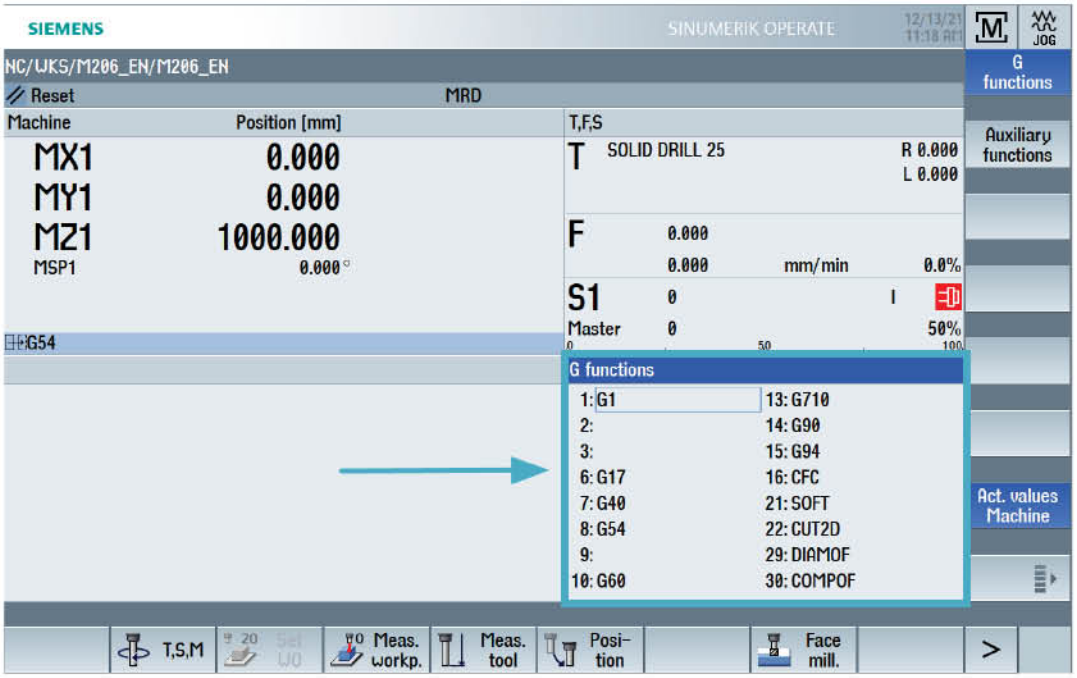


Figure 4.12: The important G function groups

Option 2:

By extending the softkey bar and pressing the softkey,

All active G function groups with the active command are displayed.

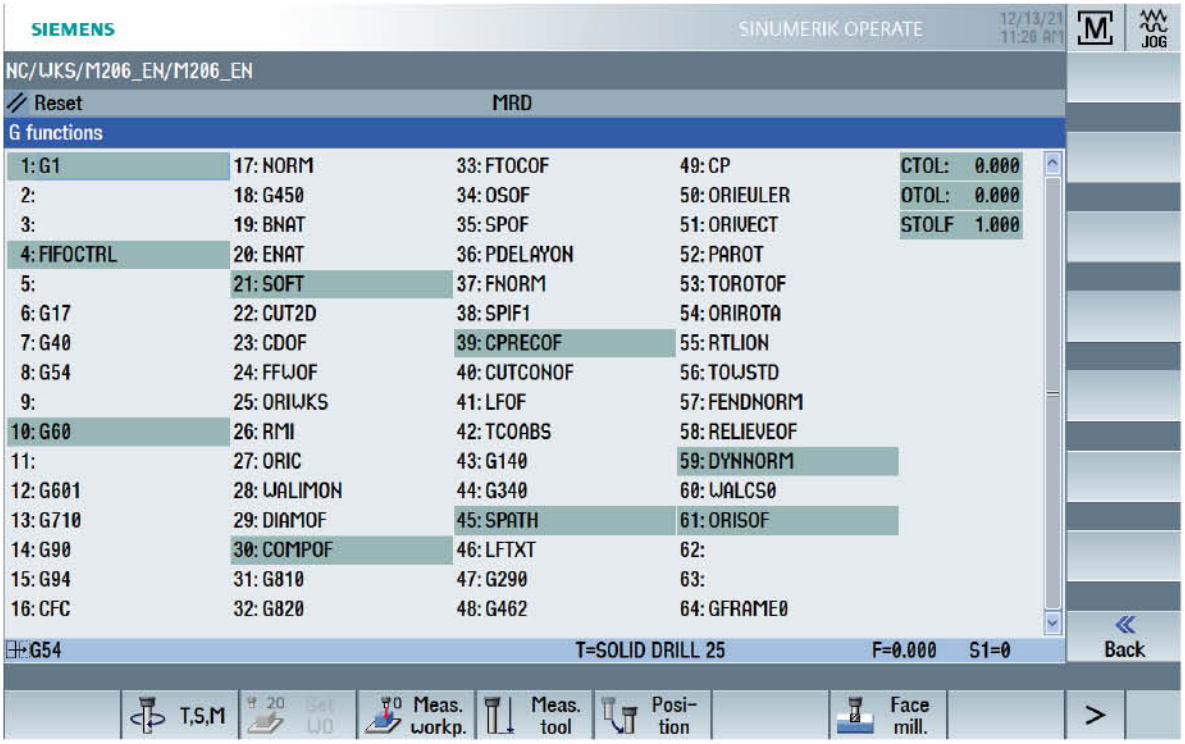


Figure 4.13: The G function groups with the active command

This view can also be active during machining so that you can follow which command is currently active in which group during the program execution.

3.7.4. Programming with R parameters, user and system variables, frames

a. Description:

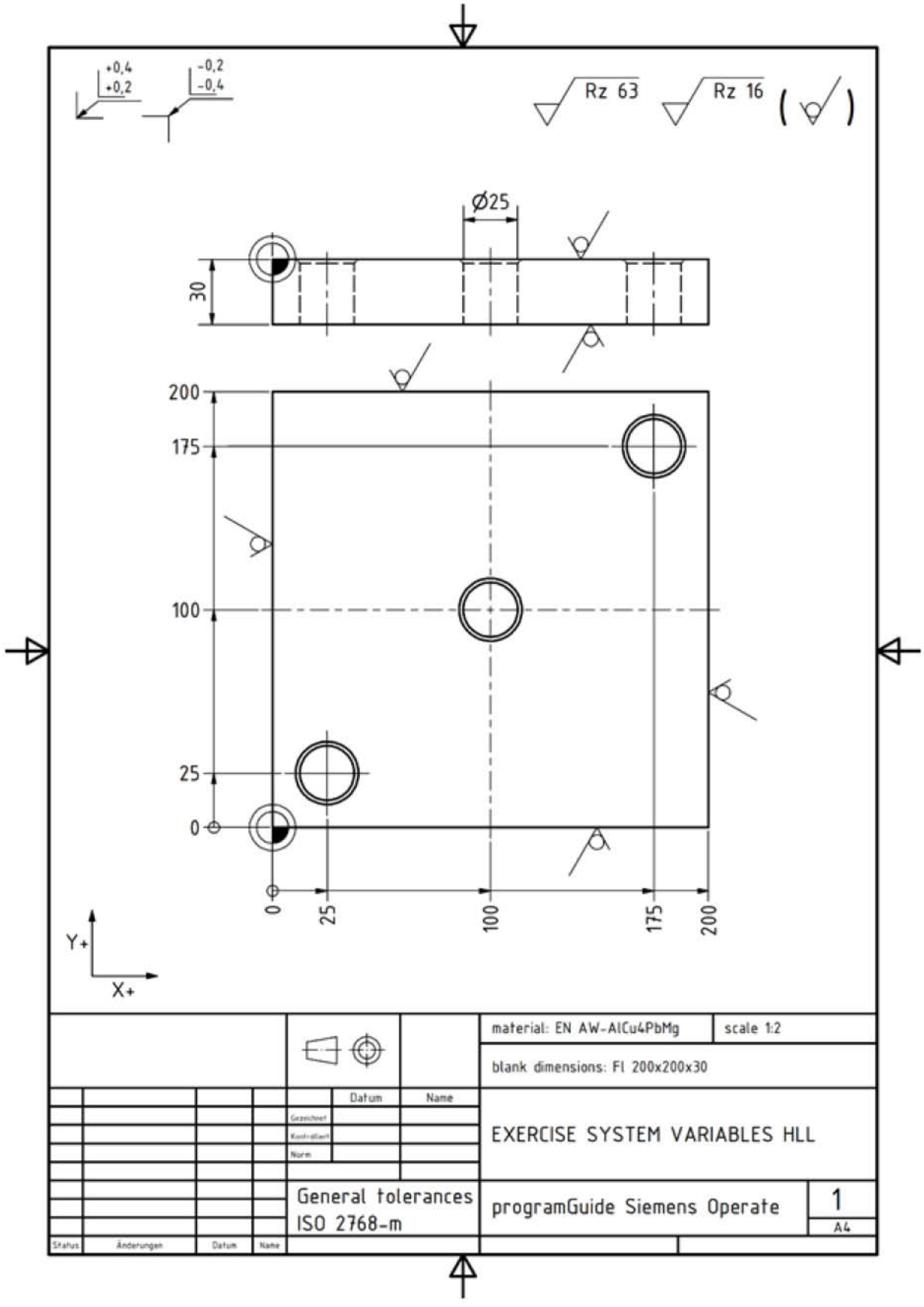
The programming program uses the programs available from the M_206 directory, to perform the query and offset the machining value in the G function instructions through parameters, user variables, system variables and Frame.

b. Objective:

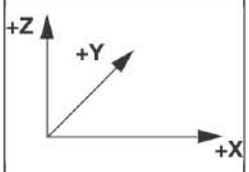
Create a program named M_207 based on the M_206 folder, and open it in the editor.

- Assumption: The workpiece is clamped via a ***permanently defined zero point clamping system***. The coordinates for this are predefined and written to a zero point.
- Configure the program so that the G-code group is queried and work offset G54 must be active. ***The operator should not change the work offsets.***
- If the active work offset is not G54, the program runs in a loop and can only be interrupted by pressing "RESET".
- Define the height
- Use the operator to specify the height of the workpiece in the program.
- Query of the system variables for the system frame and the description must take the fine offset into account in addition to the coarse offset.

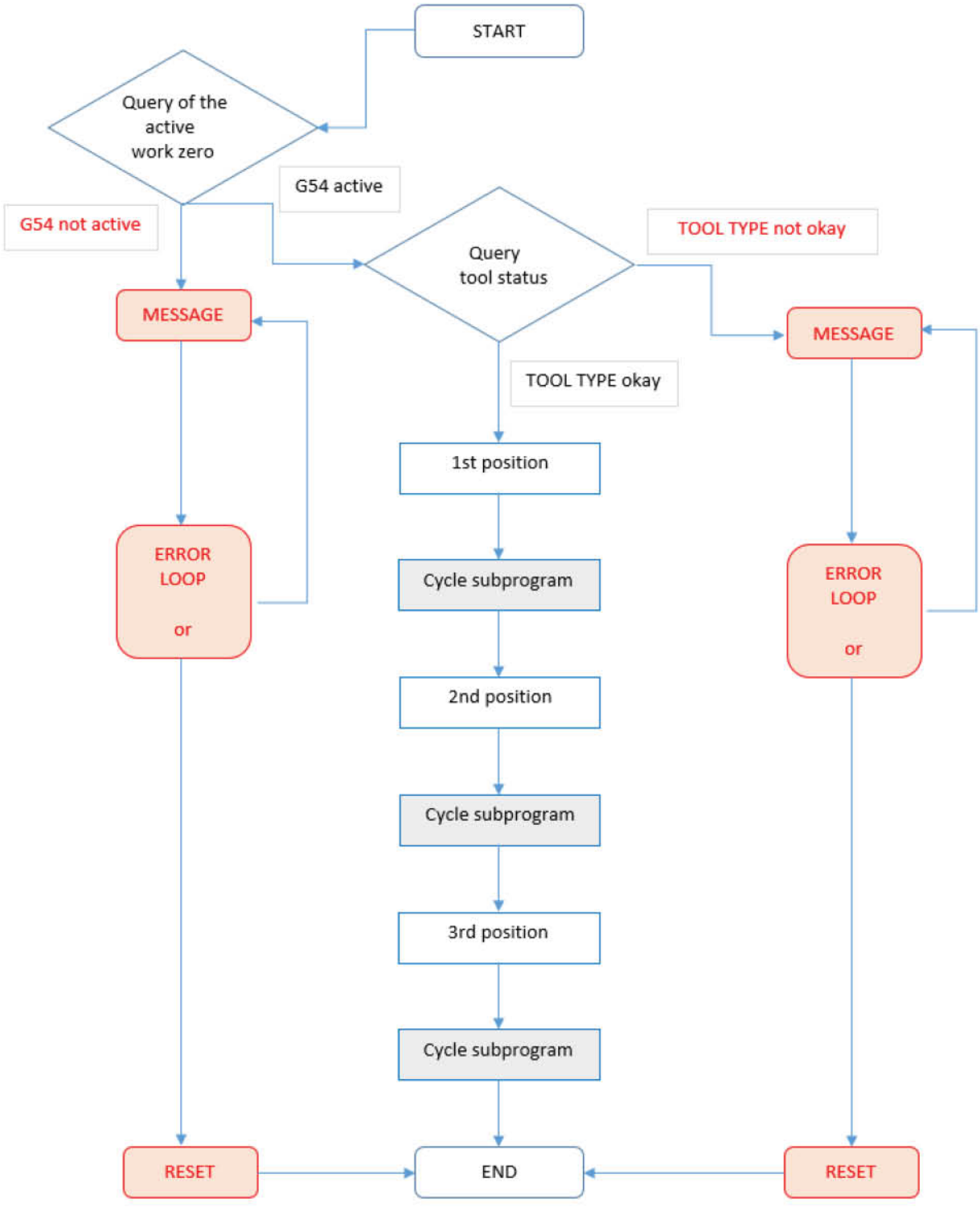
c. The technical drawing for query and work offset using R parameters, user and system variables, frames. EXERCISE SYSTEM VARIABLES HLL



d. The following tools and technology data are required for programming:

Workpiece: EXERCISE SYSTEM VARIABLES HLL		Blank material: EN AW-AlCu4PbMg				Machine configuration:		
Drawing: 1(1)		Blank dimensions: FI 200 x 200 x 30				Milling 3 axes		
	Technological data							
Tool name in OFFSET	Spot_drill_12	Deburrer_8	Cutter_32	Cutter_16	Cutter_10	Cutter_4	Solid_Drill_25	
Tool Ø (mm)	12 x 90°	8 x 90°	32	16	10	4	25	
Cutting speed (m/min)	80	105	150 / 175	150 / 175	150 / 175	150 / 175	120	
Cutting depth a_p = max.	5	3	5	8	5	2	35	
Cutting material	HSS	SC	SC	SC	SC	SC	SC	
Feed tooth fz / rev. (mm)	0,12	0,025	0,1 / 0,08	0,05 / 0,04	0,03 / 0,02	0,02 / 0,01	0,15	
Number of teeth N		4	2	4	4	4		
								
				✓	✓	✓		

e. Flowchart: A Flowchart is drawn up based on the task at hand.



Flowchart 4.7: Technological steps for practical exercises using R parameters, user and system variables, frames to query and work offset.

f. Create a new G code programGuide program, like described in this unit.

Create a new program directory and call the program for example "M_207.MPF".

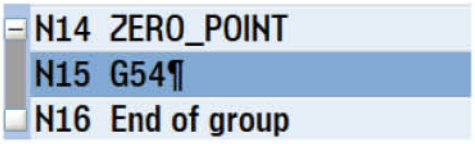
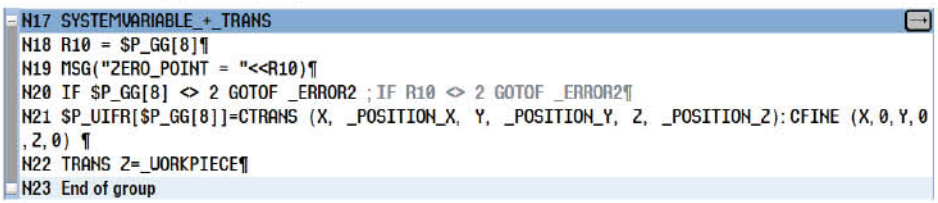
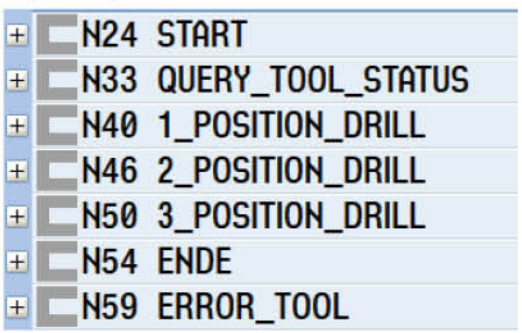
The following blocks are created in the program for the above example:

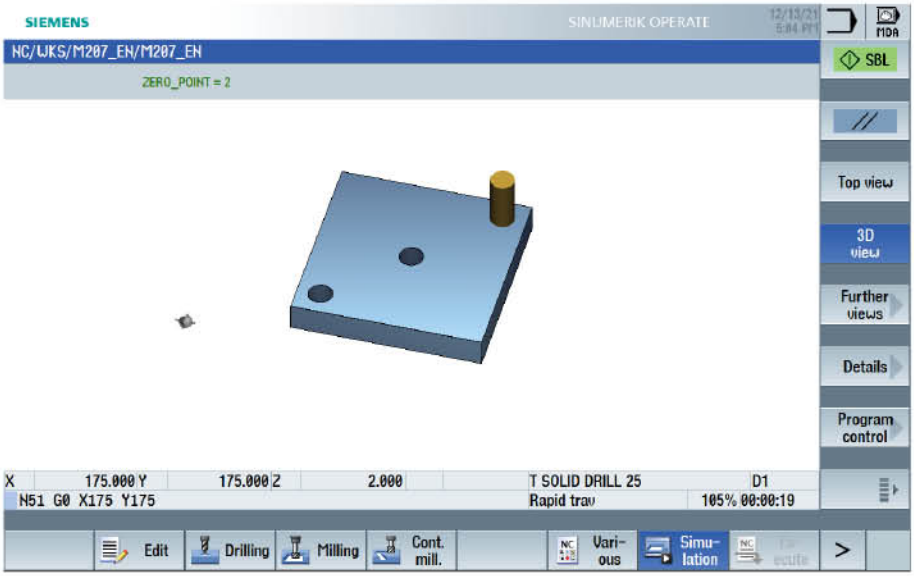
1.	Create a new programGuide directory and main program called “M_207”.
2.	The “Edit” operation mode opens automatically.
3.	Press the VSK 2 “Build groups” to open the dialog menu.
4.	Input the first block structure in the program as follow. Note: The blocks marked in bold italics are exactly the program “M_206” and must be copied into “M_207”.

Function	Name of the block
Programming (define) of the user variables	
Position of the zero point system	CLAMPING_SYSTEM
Height of the workpiece in Z coordinate	WORKPIECE_HEIGHT
Call the G code of current zero point G54	ZERO_POINT
Query and system variable for workpiece zero	SYSTEMVARIABLE_+_TRANS
Start commands	***START***
Tool query	***QUERY_TOOL_STATUS***
Drilling the 1st position	***1_POSITION_DRILL***
Drilling the 2nd position	***2_POSITION_DRILL***
Drilling the 3rd position	***3_POSITION_DRILL***
End commands	***END***
Error by wrong tool	***ERROR_TOOL***
Error by wrong zero point	ERROR_ZERO_POINT

5.	In the first step, the user variables are programmed outside of the blocks. ;DEFINE USER VARIABLE: ;_TOOLTYPE= VARIABLE FOR TOOL WITH CURRENT TOOL CUTTING EDGE NUMBER ;POSITIONS FOR CLAMPING SYSTEM N1 DEF INT _TOOLTYPE N2 DEF REAL _POSITION_X N3 DEF REAL _POSITION_Y N4 DEF REAL _POSITION_Z N5 DEF REAL _WORKPIECE

	NC/WKS/M207_EN/M207_EN ¶ ;DEFINE USER VARIABLE:¶ ;_TOOLTYPE= VARIABLE FOR TOOL WITH CURRENT TOOL CUTTING EDGE NUMBER ;POSITIONS FOR CLAMPING SYSTEM¶¶ N1 DEF INT _TOOLTYPE ¶ N2 DEF REAL _POSITION_X ¶ N3 DEF REAL _POSITION_Y ¶ N4 DEF REAL _POSITION_Z ¶ N5 DEF REAL _WORKPIECE¶
6.	The positions for the zero point clamping system are programmed in the "CLAMPING_SYSTEM" block: N6 GROUP_BEGIN(0,"CLAMPING_SYSTEM",0,0) ;POSITION OF ZERO POINT CLAMPING SYSTEM N7 _POSITION_X=100 N8 _POSITION_Y=100 N9 _POSITION_Z=0 N10 GROUP_END(0,0) N6 CLAMPING_SYSTEM ;POSITION OF ZERO POINT CLAMPING SYSTEM N7 _POSITION_X=100 ¶ N8 _POSITION_Y=100 ¶ N9 _POSITION_Z=0 ¶ N10 End of group
7.	The height for the current workpiece is entered in the "WORKPIECE_HEIGHT" block. N11 GROUP_BEGIN(0,"WORKPIECE_HEIGHT",0,0) ;INPUT HEIGHT OF WORKPEACE N12 _WORKPIECE=50 N13 GROUP_END(0,0) N11 WORKPIECE_HEIGHT ;INPUT HEIGHT OF WORKPEACE N12 _WORKPIECE=50 ¶ N13 End of group

8.	The workpiece zero point is programmed in the “ZERO POINT” block N14 GROUP_BEGIN(0,"ZERO_POINT",0,0) N15 G54 N16 GROUP_END(0,0) 
9.	The system variables for the zero point and the workpiece height are programmed in the "SYSTEMVARIABLE_+_TRANS" block: N17 GROUP_BEGIN(0,"SYSTEMVARIABLE_+_TRANS",0,0) N18 R10 = $P_GG[8] N19 MSG("ZERO_POINT = "<<R10) N20 IF $P_GG[8] <> 2 GOTOF _ERROR2 ;IF R10 <> 2 GOTOF _ERROR2 N21 $P_UIFR[$P_GG[8]]=CTRANS (X, _POSITION_X, Y, _POSITION_Y, Z, _POSITION_Z):CFINE (X,0,Y,0,Z,0) N22 TRANS Z=_WORKPIECE N23 GROUP_END(0,0) 
10.	At this point copy the whole program “M_206” in this current program “M_207”, here with these blocks: 
11.	Finally, in the "ERROR_WORKPIECE_ZERO" block, the details of the program loop are programmed in the event of an incorrect zero point call: N65 GROUP_BEGIN(0,"ERROR_WORKPIECE_ZERO",0,0) N66 _ERROR2: N67 MSG("INCORRECT ZERO POINT!! G54 MUST BE SELECTED") N68 M0 N69 GOTOB _ERROR2

	N70 GROUP_END(0,0) N65 ERROR_WORKPIECE_ZERO N66 _ERROR2: ¶ N67 MSG("INCORRECT ZERO POINT!! G54 MUST BE SELECTED") N68 M0 ¶ N69 GOTOB _ERROR2¶ N70 End of group
12.	Finally check the machining with the simulation with a right tool for drilling and the right call for the zero point with “G54”: 
13.	Finally check the machining with the simulation. Use a wrong call for the zero point with “G55”: N14 ZERO_POINT N15 G55¶ N16 End of group Check the error message with using the simulation: NC/WKS/M207_EN/M207_EN INCORRECT ZERO POINT!! G54 MUST BE SELECTED

3.8. Programming of arithmetic functions

3.8.1. Types of variables

Command	Meaning
PROC	First instruction of a program with variable declaration to take the values from the main program
DISPLOF	Current block display is suppressed
SAVE	Saving of the modal G functions
ABS	Absolute dimension
IF/ENDIF	Program loops to check the transferred variable contents for plau sibility
SBLOF	No single block execution
SBLON	Single block execution possible
LOOP	Start endless loop
ENDLOOP	End line of an endless loop
ENDIF/ELSE	Program loops
SETAL	Set alarms

3.8.2. Function of variables

a. PROC:

This command is used for subprograms whose parameter values are to be taken over by the calling program.

Example:

The DRILL.SPF cycle is a self-written user cycle. The cycle is to be active in the main program, i.e. the programmed values and parameters are to be taken into the main program.

If a user cycle is to accept this condition, the first command is always the "PROC" command.

All cycles from Siemens begin with this command.

b. DISPLOF:

The current block display can be suppressed with this command.

Example: If cycles are called in a main program, the individual NC blocks of the current DRILL.SPF cycle are not visible. They are suppressed. Cycles are usually programmed in this way.

c. SAVE:

With the SAVE command, the modal G functions and the current frame active before the subprogram call are saved by the main program and are active again after return.

d. ABS:

The value always refers to an absolute dimension.

e. IF/ ENDIF and ENDIF/ELSE:

These commands are used if the program loop is to contain an alternative program block.

If the IF condition is satisfied then the IF block is executed.

If the IF condition is not satisfied then the program block after the ELSE command is executed.

The ENDIF command marks the end of an IF loop. The program returns to the start of the loop.

f. SBLOF:

In programs in which SBLOF is programmed, the following blocks are executed completely, even when single block execution is active.

Example:

A coherent machining operation must not be performed in single blocks, because otherwise the tool or the workpiece will be dam aged.

g. SBLON:

After this command, the following NC blocks are executed in single blocks.

h. LOOP/ENDLOOP:

An endless loop is usually programmed with these commands.

The start of the loop is programmed with the "LOOP" command.

ENDLOOP marks the end of the loop and results in a return to "LOOP", the start of the loop.

Example: This command is used when a certain machining operation is repeated very often. A counter can be integrated in this program loop via R variables. When a certain number of machining operations has been reached, a jump is made to another program section.

i. SETAL:

An alarm can be programmed with this command.

SETAL must be programmed in a separate NC block together with the alarm number.

The alarm numbers and the alarms are structured as follows:

60000 to 64999, alarm numbers reserved for Siemens cycles.

65000 to 69999, alarm numbers available for users. These can also have texts added.

Example:

N100 SETAL (65000) ; **Set alarm no. 65000.**

3.8.3. Programming of arithmetic functions

a. Description:

Programs are programmed using variable types to perform calculation functions, correcting accuracy, absolute size, indirect programming, using program jumps to create program loops.

b. Objective:

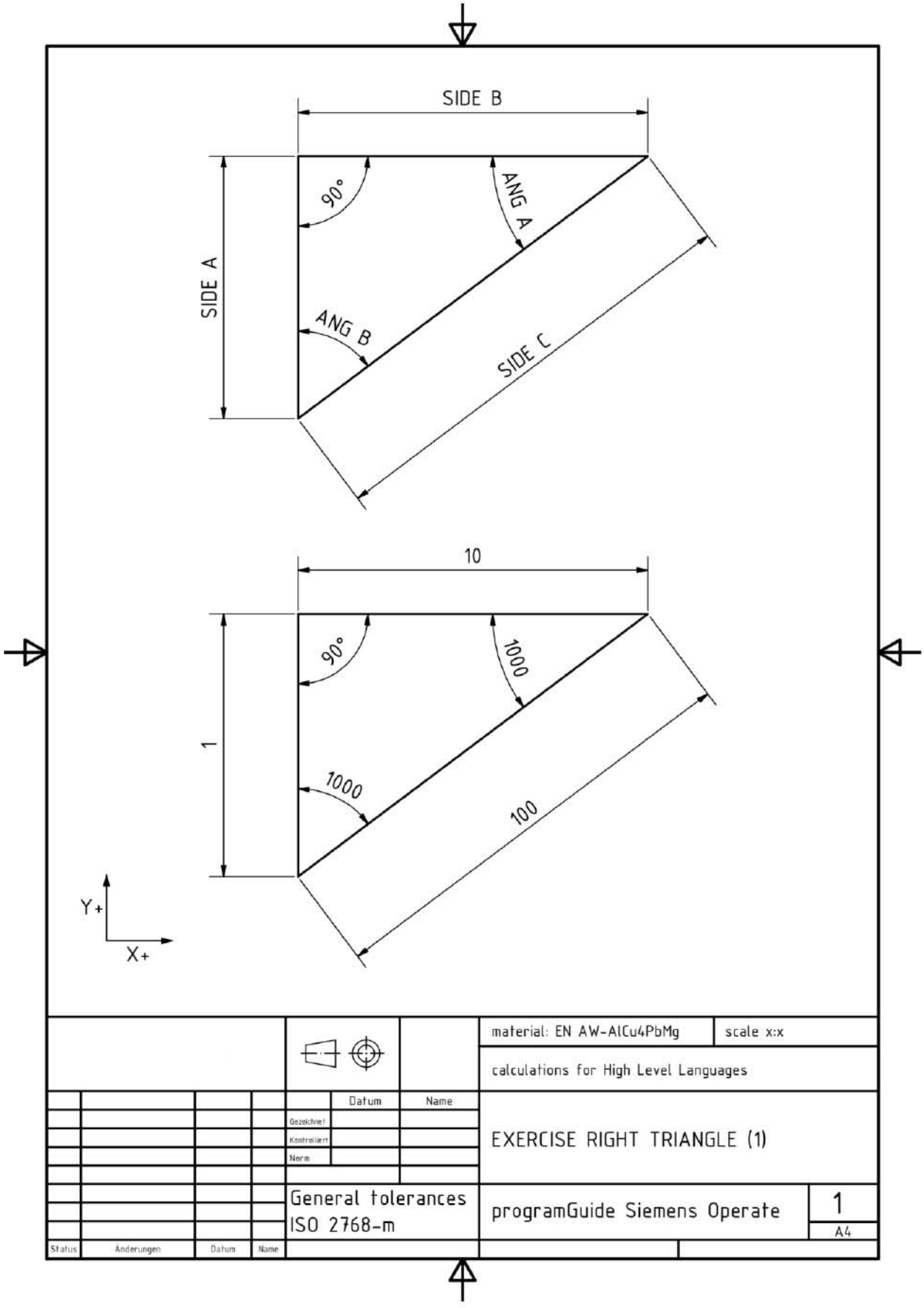
Use variable types to perform caculating functions for arithmetic functions of right triangles, calculate the lengths of the sides when the values of the other two sides are known, and the angle.

Correct the function's precision and determine the absolute size value.

Rounded calculation results, jump to program segments, results

Create an error loop, displaying an error message text letting the operator know that the data needs to be corrected.

c. The technical drawing for EXERCISE RIGHT TRIANGLE



d. Create a new G code programGuide program, like described in this unit.

Create a new program directory and call the program for example " EXERCISE RIGHT TRIANGLE ". The following blocks are created in the program for the above example:

PROBLEM 1: CALCULATION FOR THE FIRST RIGHT TRIANGLE

```
; User variable definition
DEF REAL _SIDEA, _SIDEB, _SIDEC, _ANGLEA, _ANGLEB
DEF INT _CHECK
;*****************************
; Declare variables, values of edges and corners can be edited
_SIDEA = 50
_SIDEB = 0
_SIDEC =  0
_ANGLEA = 80
_ANGLEB = 0
;*****************************
; Declare variable with absolute size
_SIDEA = ABS(_SIDEA)
_SIDEB = ABS(_SIDEB)
_SIDEC = ABS(_SIDEC)
_ANGLEA = ABS(_ANGLEA)
_ANGLEB = ABS(_ANGLEB)
;*****************************
; Check the first condition
IF _SIDEA > 0
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Check the second condition
IF _SIDEB > 0
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Check the third condition
```

```
IF _SIDEC > 0
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Check the fourth condition
IF _ANGLEA > 0
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Check the fifth condition
IF _ANGLEB > 0
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Condition jump to error position
IF (_CHECK > 2) OR (_CHECK < 2) GOTO ERROR_1
IF (_ANGLEA > 0) AND (_ANGLEB > 0) GOTO ERROR_2
IF (_SIDEC > 0)
 IF (_SIDEA >= _SIDEC) OR (_SIDEB >= _SIDEC) GOTO ERROR_3
ENDIF
IF (_ANGLEA >= 90) OR (_ANGLEB >= 90) GOTO ERROR_4
 IF _ANGLEB > 0
 _ANGLEA = 90-_ANGLEB
 ENDIF
;*********************************
; Programmatically jump to the calculated label according to the condition
MSG("PROGAM BRANCHING")
IF (_SIDEA > 0) AND (_SIDEB > 0) GOTO CALC_AB
IF (_SIDEB > 0) AND (_SIDEC > 0) GOTO CALC_BC
IF (_SIDEA > 0) AND (_SIDEC > 0) GOTO CALC_AC
IF (_SIDEA > 0) AND (_ANGLEA > 0) GOTO CALC_A_ANGA
```

```
IF (_SIDEB > 0) AND (_ANGLEA > 0) GOTO CALC_B_ANGA
IF (_SIDEC > 0) AND (_ANGLEA > 0) GOTO CALC_C_ANGA
;***********************************************
GOTO ERROR_5          ; LOOP BY PROGRAM ERROR
;***********************************************
; Calculate side C, angle A and angle B
CALC_AB:
_SIDEC=SQRT(POT(_SIDEA)+POT(_SIDEB))
_ANGLEA=ACOS(_SIDEB/_SIDEC)
_ANGLEB=ACOS(_SIDEA/_SIDEC)
;*****************************
GOTO RESULT
;*****************************
; Calculate side A, angle A and angle B
CALC_BC:
_SIDEA=SQRT(POT(_SIDEC)-POT(_SIDEB))
_ANGLEA=ACOS(_SIDEB/_SIDEC)
_ANGLEB=ACOS(_SIDEA/_SIDEC)
;*****************************
GOTO RESULT
;*****************************
; Calculate side B, angle A and angle B
CALC_AC:
_SIDEB=SQRT(POT(_SIDEC)-POT(_SIDEA))
_ANGLEA=ACOS(_SIDEB/_SIDEC)
_ANGLEB=ACOS(_SIDEA/_SIDEC)
;*****************************
GOTO RESULT
;*****************************
; Calculate side B, side C and angle B
CALC_A_ANGA:
_SIDEC=_SIDEA/SIN(_ANGLEA)
_SIDEB=SQRT(POT(_SIDEC)-POT(_SIDEA))
_ANGLEB=90-_ANGLEA
;*****************************
GOTO RESULT
;*****************************
```

```
; Calculate side A, side C and angle B
CALC_B_ANGA:
_SIDEC=_SIDEB/COS(_ANGLEA)
_SIDEA=SQRT(POT(_SIDEC)-POT(_SIDEB))
_ANGLEB=90-_ANGLEA
;*****************************
GOTO RESULT
;*****************************
; Calculate side A, side B and angle B
CALC_C_ANGA:
_SIDEB=_SIDEC*COS(_ANGLEA)
_SIDEA=SQRT(POT(_SIDEC)-POT(_SIDEB))
_ANGLEB=90-_ANGLEA
;*****************************
GOTO RESULT
;*****************************
; Showing results
RESULT:
; Round the result to 3 decimals
MSG("SIDEA = "<<_SIDEA<< " SIDEB = "<<_SIDEB<<" SIDEC = "<<_SIDEC<<" ANGLEA
= "<<ROUND(_ANGLEA*1000)/1000<<" ANGLEB = "<<ROUND(_ANGLEB*1000)/1000)
M0
M30
;*****************************
; Create an error loop, displaying an error message letting the operator know that the
data needs to be corrected.
ERROR_1:
MSG("PLEASE CHECK THE NUMBER OF ENTRY - ONLY 2 ARE ALLOWED")
M0
GOTO ERROR_1
;*****************************
ERROR_2:
MSG("PLEASE ENTER ONLY 1 ANGLE")
M0
GOTO ERROR_2
;*****************************
ERROR_3:
```

```
MSG("PLEASE CHECK THAT SIDEC IS LONGER THAN SIDEA OR SIDEB")
M0
GOTO ERROR_3
;*****************************
ERROR_4:
MSG("ATTENTION: ANGLE A OR B IST TOO LARGE - MAX 89,999°")
M0
GOTO ERROR_4
;*****************************
ERROR_5:
MSG("ATTENTION A PROGRAM ERROR HAS OCCURRED - PLEASE
NOTIFY THE PROGRAMMER") M0
GOTO ERROR_5
;*****************************
```

PROBLEM 2: CALCULATORS FOR THE SECOND RIGHT TRIANGLE WITH ADD VALUE

; User variable definition add value definition

```
DEF REAL _SIDEA, _SIDEB, _SIDEC, _ANGLEA, _ANGLEB
DEF INT _CHECK, _VALUE
;*****************************
```

; Declare variables, values of edges and corners can be edited

```
_SIDEA = 8.452
_SIDEB = 0
_SIDEC = 20
_ANGLEA = 0
_ANGLEB = 0
;*****************************
```

; Declare variable with absolute size

```
_SIDEA = ABS(_SIDEA)          ;1
_SIDEB = ABS(_SIDEB)          ;10
_SIDEC = ABS(_SIDEC)          ;100
_ANGLEA = ABS(_ANGLEA)    ;1000
_ANGLEB = ABS(_ANGLEB)    ;1000
;*****************************
```

; Check the first condition

```
IF _SIDEA > 0
```

```
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Check the second condition
IF _SIDEB > 0
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Check the third condition
IF _SIDEC > 0
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Check the fourth condition
IF _ANGLEA > 0
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Check the fifth condition
IF _ANGLEB > 0
_CHECK = _CHECK+1
MSG("VALUE OF CHECK = "<<_CHECK)
M0
ENDIF
;*****************************
; Condition jump to error position
IF (_CHECK > 2) OR (_CHECK < 2) GOTO ERROR_1
IF (_ANGLEA > 0) AND (_ANGLEB > 0) GOTO ERROR_2
IF (_SIDEC > 0)
```

```
 IF (_SIDEA >= _SIDEC) OR (_SIDEB >= _SIDEC) GOTO ERROR_3
ENDIF
IF (_ANGLEA >= 90) OR (_ANGLEB >= 90) GOTO ERROR_4
;*****************************
; Add value conversion condition, create program loop
IF _SIDEA > 0
 _VALUE = _VALUE+1
 ENDIF
 IF _SIDEB > 0
 _VALUE = _VALUE+10
 ENDIF
 IF _SIDEC > 0
 _VALUE = _VALUE+100
 ENDIF
 IF _ANGLEA > 0
 _VALUE = _VALUE+1000
 ENDIF
 IF _ANGLEB > 0
 _VALUE = _VALUE+1000
 ENDIF
 MSG(<<_VALUE)
 M0
 STOPRE                          ; INTERPRETER STOP
;***********************************
; Programmatically jump to the calculated label according to the condition
MSG("PROGAM BRANCHING")
IF _VALUE == 11 GOTO CALC_AB
IF _VALUE == 110 GOTO CALC_BC
IF _VALUE == 101 GOTO CALC_AC
IF _VALUE == 1001 GOTO CALC_A_ANGA
IF _VALUE == 1010 GOTO CALC_B_ANGA
IF _VALUE == 1100 GOTO CALC_C_ANGA
;***********************************************
GOTO ERROR_5     ; LOOP BY PROGRAM ERROR
;***********************************************
; Calculate side C, angle A and angle B
CALC_AB:
```

```
_SIDEC=SQRT(POT(_SIDEA)+POT(_SIDEB))
_ANGLEA=ACOS(_SIDEB/_SIDEC)
_ANGLEB=ACOS(_SIDEA/_SIDEC)
;*****************************
GOTO RESULT
;*****************************
; Calculate side A, angle A and angle B
CALC_BC:
_SIDEA=SQRT(POT(_SIDEC)-POT(_SIDEB))
_ANGLEA=ACOS(_SIDEB/_SIDEC)
_ANGLEB=ACOS(_SIDEA/_SIDEC)
;*****************************
GOTO RESULT
;*****************************
; Calculate side B, angle A and angle B
CALC_AC:
_SIDEB=SQRT(POT(_SIDEC)-POT(_SIDEA))
_ANGLEA=ACOS(_SIDEB/_SIDEC)
_ANGLEB=ACOS(_SIDEA/_SIDEC)
;*****************************
GOTO RESULT
;*****************************
; Calculate side B, side C and angle B
CALC_A_ANGA:
IF _ANGLEB > 0                    ; Assign conditions to angle B
 _ANGLEA = 90-_ANGLEB
 ENDIF
_SIDEC=_SIDEA/SIN(_ANGLEA)
_SIDEB=SQRT(POT(_SIDEC)-POT(_SIDEA))
_ANGLEB=90-_ANGLEA
;*****************************
GOTO RESULT
;*****************************
; Calculate side A, side C and angle B
CALC_B_ANGA:
IF _ANGLEB > 0                    ; Assign conditions to angle B
 _ANGLEA = 90-_ANGLEB
```

```
 ENDIF
_SIDEC=_SIDEB/COS(_ANGLEA)
_SIDEA=SQRT(POT(_SIDEC)-POT(_SIDEB))
_ANGLEB=90-_ANGLEA
;*****************************
GOTO RESULT
;*****************************
; Calculate side A, side B and angle B
CALC_C_ANGA:
IF _ANGLEB > 0                          ; Assign conditions to angle B
 _ANGLEA = 90-_ANGLEB
 ENDIF
_SIDEB=_SIDEC*COS(_ANGLEA)
_SIDEA=SQRT(POT(_SIDEC)-POT(_SIDEB))
_ANGLEB=90-_ANGLEA
;*****************************
GOTO RESULT
;*****************************
; Showing results
RESULT:
; Round the result to 3 decimals
MSG("SIDEA = "<<_SIDEA<< " SIDEB = "<<_SIDEB<<" SIDEC = "<<_SIDEC<<" ANGLEA
= "<<ROUND(_ANGL EA*1000)/1000<<" ANGLEB = "<<ROUND(_ANGLEB*1000)/1000)
M0
M30
;*****************************
; Create an error loop, displaying an error message letting the operator know that the
data needs to be corrected.
ERROR_1:
MSG("PLEASE CHECK THE NUMBER OF ENTRY - ONLY 2 ARE ALLOWED")
M0
GOTO ERROR_1
;*****************************
ERROR_2:
MSG("PLEASE ENTER ONLY 1 ANGLE")
M0
GOTO ERROR_2
```

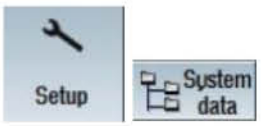
```
;*****************************
ERROR_3:
MSG("PLEASE CHECK THAT SIDEC IS LONGER THAN SIDEA OR SIDEB")
M0
GOTO ERROR_3
;*****************************
ERROR_4:
MSG("ATTENTION: ANGLE A OR B IST TOO LARGE - MAX 89,999°")
M0
GOTO ERROR_4
;*****************************
ERROR_5:
MSG("ATTENTION A PROGRAM ERROR HAS OCCURRED - PLEASE
NOTIFY THE PROGRAMMER") M0
GOTO ERROR_5
;*****************************
```

3.9. USER CYCLES

3.9.1. Configuration and management of a cycle on SINUMERIK

For Siemens, cycles are always subprograms with the ending ".SPF".

The philosophy is the following:

In the operating area at 

there are three other fold- ers in the Cycles folder.

a. Standard cycles

All cycles from SINUMERIK Operate are in this folder. These cycles are subprograms and can be opened and written.

However, when the control ramps up again, the cycles are returned to their original state.

b. Manufacturer cycles

The cycles from the manufacturer are in this folder.

Modified Cycles are active and stored. They are only be changed by machine manufacturer.

c. User cycles

The cycles written by the operator are in this folder.

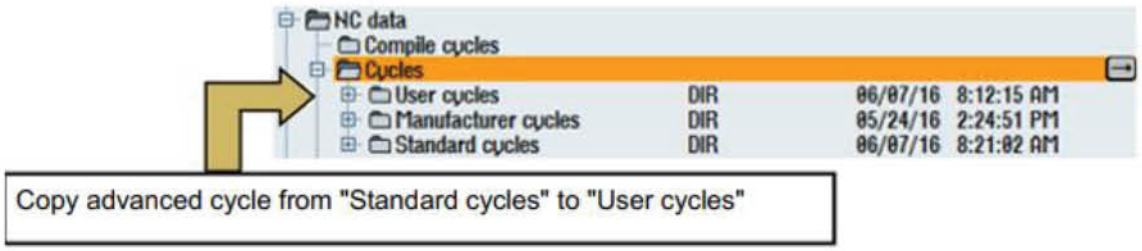
Modified Cycles are active and stored.

d. Procedure:

When an advanced user cycle is created, it must not necessarily have to be completely written. It is possible, for example, to modify standard cycles.

If you want to modify an existing advanced cycle, proceed as follows:

The advanced standard cycle from the "Standard cycles" folder is programmed in the "User cycles" folder.



The name of the advanced cycle is retained, the changes are made and then the control is restarted. After ramp-up, the modified advanced cycle is available.

Notice: **It is not recommended to change standard cycles, but if necessary, be very careful.**

The following routine applies when calling an advanced cycle.

The system checks whether the called advanced cycle is in the "User cycles" folder. If so, it is executed.

If the called advanced cycle is not in the "User cycles" folder but in the "Manufacturer cycles" folder, this is executed.

If the called advanced cycle is only in the "Standard cycles" folder, this is used.

This procedure ensures that the latest advanced cycle in the program is always used and therefore the changes are activated.

If you write a new advanced cycle, it is stored directly in the "User cycles" folder.

If an own advanced cycle is called, the operator has no direct influence on this because it is usually not visible and the operator does not know where to find it.

When the control ramps up again, the cycle is active in the "User cycles" folder.

3.9.2. Project work: Writing of an own drilling cycle step by step

a. Description:

In a company, pipes are drilled on a milling machine. The pipes have different diameters. Depending on the material and wall thickness, special long solid carbide drills with different angles and diameters are used.

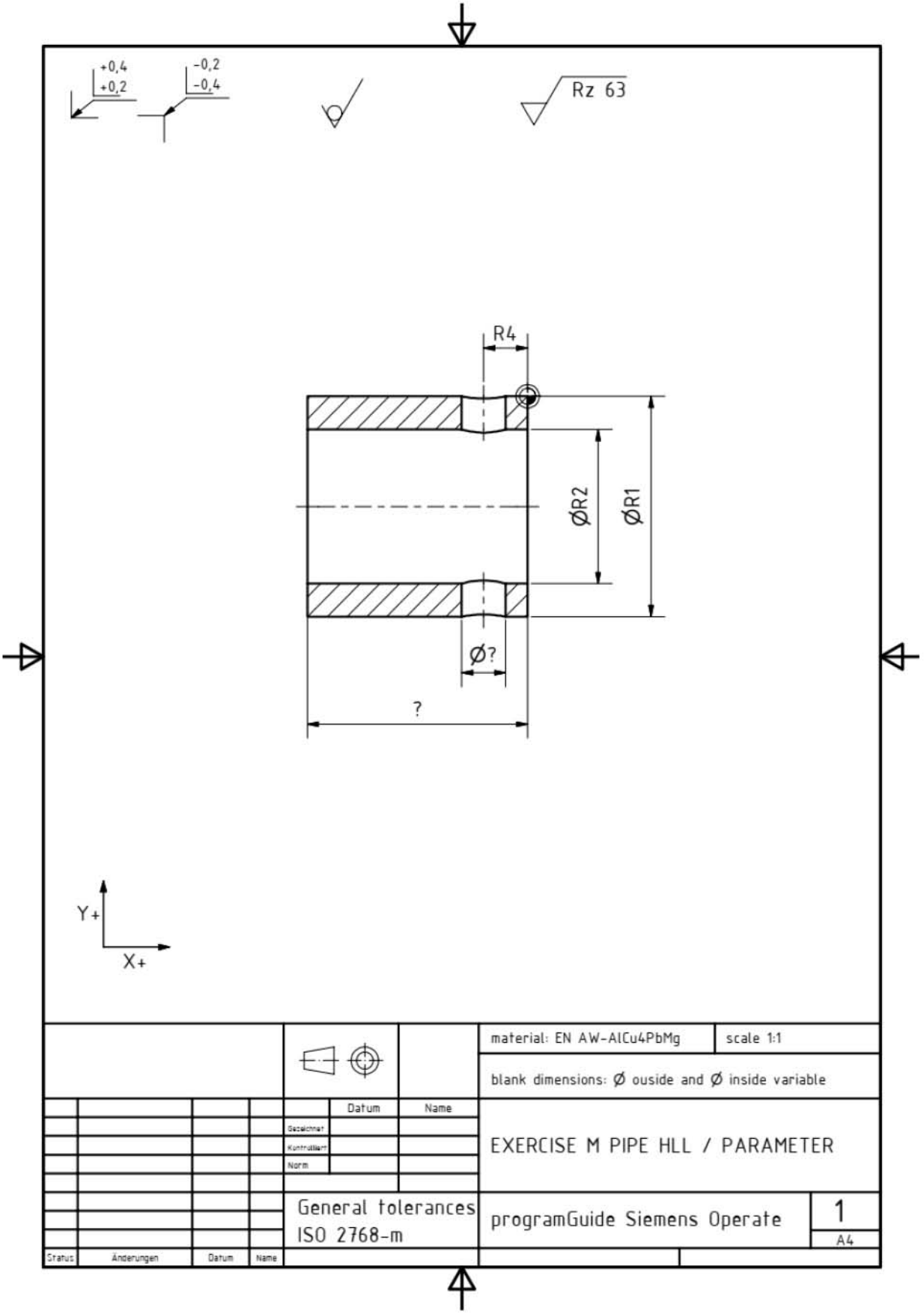
b. Objective:

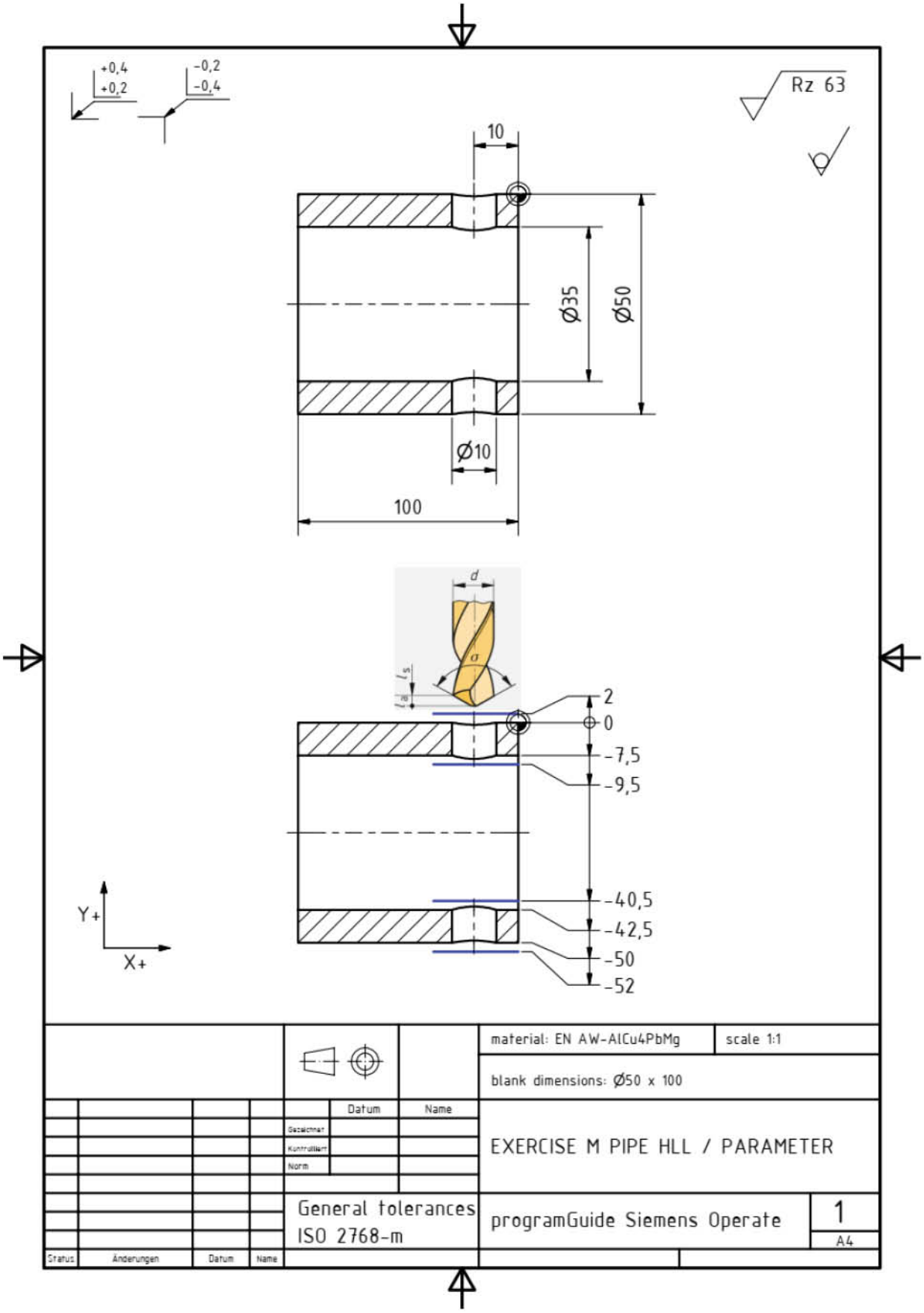
I. Create a “EXERCISE_PIPE.WPD” and develop a linear program for the drawing shown and save it as “DRILL_1_DIN.MPF” and fulfill the following requirements:

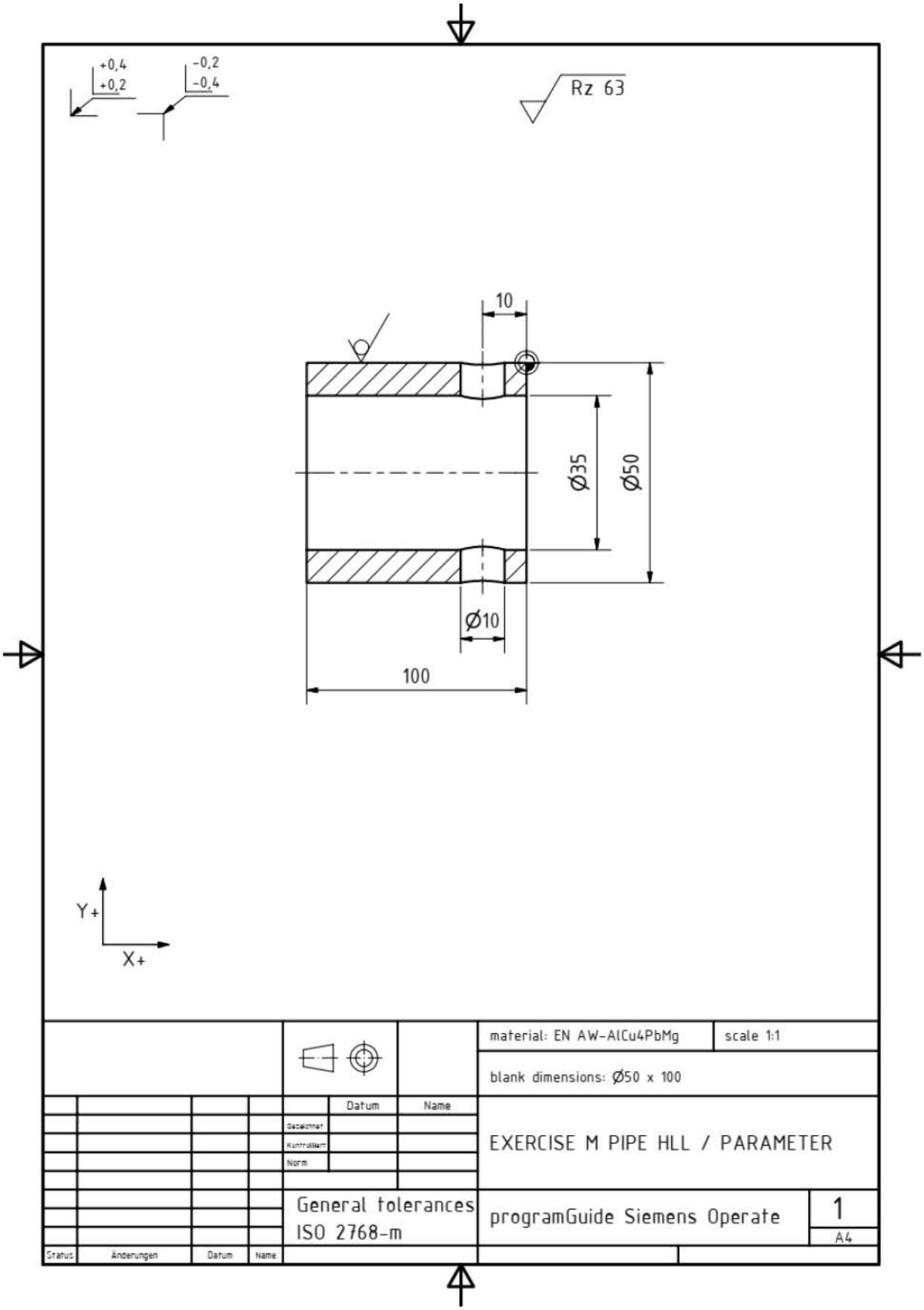
Specification of the main program:

1. Only one drilling tool should be used
2. A safety distance of 2 mm must be observed
3. Do not use cycles for programming
4. The cutting speed is predetermined with v_c = 80m/min
5. The feed is predetermined with F = 0.15 mm/rev in the material
6. The empty feed in the pipe is predetermined with F = 4000 mm/min
7. Only use the following G commands G00/ G01/ G94 for the program

c. The technical drawing for DRILL

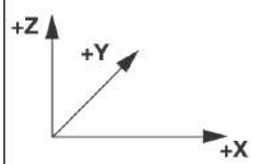



+0,4
+0,2
-0,2
-0,4
Rz 63
10
Ø35
Ø50
Ø10
100
d
σ
2
0
-7,5
-9,5
-40,5
-42,5
-50
-52
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Ø50 x 100
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE M PIPE HLL / PARAMETER
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name


+0,4
+0,2
-0,2
-0,4
Rz 63
10
Ø35
Ø50
Ø10
100
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Ø50 x 100
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE M PIPE HLL / PARAMETER
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name

d. The following tools and technology data are required for programming:

Workpiece: EXERCISE M PIPE HLL/PARAMETRER	Blank material: EN AW-AlCu4PbMg	Machine configuration:
Drawing: 1(1)	Blank dimensions: ϕ50 x 100	Milling 3 axes

	Technological data		
Tool name in OFFSET	Spot_drill_12	Deburrer_8	SC_Drill_10
Tool Ø (mm)	12 x 90°	8 x 90°	10
Cutting speed (m/min)	80	105	80
Cutting depth a_p = max.	5	3	
Cutting material	HSS	SC	SC
Feed tooth fz / rev. (mm)	0,12	0,025	0,15
Number of teeth N		4	
			

e. Create a new G code programGuide program, like described in this unit.

Create a new program directory and call the program for example "DRILL_1_DIN.MPF ". The following code is generated in the program for the above example:

```
CYCLE800()¶
                TC              0

WORKPIECE(,"",,"PIPE",320,0,-100,-80,50,35)¶
                Blank           Pipe
                ØA          50.000
                ØI          35.000 abs


                HA           0.000
                HI        -100.000 abs
T="SC_DRILL_10"
D1 M6
;*******************************************
```

```
CYCLE800(0, "TC1", 100000, 57, -25, 0, 0, 0, -90, 0, 0, 0, 0, -1, 100, 1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	No	
Swivel plane	New	
X0	-25.000	
Y0	0.000	
Z0	0.000	
Swivel mode	Axis by axis	
Sequence of axes	X Y Z	
X	0.000	°
Y	-90.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select	-	
Tool		

```
G94 S=((80*1000)/(3.1415*10)) F=((80*1000)/(3.1415*10))*0.15
G0 X-10 Y0
G0 Z2
;*********************************************
G1 Z-9.5 F=((80*1000)/(3.1415*10))*0.15
G1 Z-40.5 F4000
G1 Z-52 F=((80*1000)/(3.1415*10))*0.15
G0 Z2
;*********************************************
CYCLE800(2, "TC1", 200000, 57, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 100, 1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	Z XY	
Swivel plane	New	
X0	0.000	
Y0	0.000	
Z0	0.000	
Swivel mode	Axis by axis	
Sequence of axes	X Y Z	
X	0.000	°
Y	0.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select	+	
Tool		

M30

;***

II. Create a linear program for the displayed drawing and save it as "DRILL_2_DIN.MPF" and execute the following requirements:

Parameterize the program and use the following R parameters:

R1 = 50 ; outside diameter

R2 = 35 ; inside diameter

R3 = 2 ; safety distance

R4 = -10 ; drilling position in X

R5 = 0 ; drilling position in Y

R6 = 0 ; coordinate surface in Z

R7 = 80 ; cutting speed v_c in m/min

R8 = 0.15 ; feed mm/rev for machining

R9 = 4000 ; feed mm/min in the pipe

Use the following calculation parameters:

R20 = ; Wall thickness

R21 = ; Start position in Z

R22 = ; End in Z upper hole

R23 = ; Start drilling below

R24 = ; End hole below

Other R parameters e.g.: for feed and speed calculation are freely selectable.

R25= ; revolutions per minute

R26= ; feed mm/min

Test the program extensively with different parameters!

The following code is generated in the program for the above example:

```
CYCLE800()¶
          TC          0
;****************************************
R1 = 50 ; ****************** outside diameter
R2 = 35 ; ****************** inside diameter
R3 = 2 ; ******************* safety distance
R4 = -10 ; ***************** drilling position in X
R5 = 0 ; ******************* drilling position in Y
R6 = 0 ; ******************* coordinate surface in Z
R7 = 80 ; ****************** cutting speed vc in m/min
R8 = 0.15 ; **************** feed mm/rev for machining
```

```
R9 = 4000 ; ********** feed mm/min in the pipe
;*****************************************
```

```
WORKPIECE(,"",,"PIPE",320,0,-100,-80,50,35)¶
```

Blank		Pipe
ØA	50.000	
ØI	35.000	abs
HA	0.000	
HI	-100.000	abs

```
T="SC_DRILL_10"
D1 M6
;*********************************************
```

```
CYCLE800(0,"TC1",100000,57,-25,0,0,0,-90,0,0,0,0,-1,100,1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	No	
Swivel plane		New
X0	-25.000	
Y0	0.000	
Z0	0.000	
Swivel mode		Axis by axis
Sequence of axes		X Y Z
X	0.000	°
Y	-90.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select		-
Tool		

```
;*********************************************
R20=(R1-R2)/2                    ;********** Wall thickness
R21=R6+R3                        ;********** Start position in Z
R22=(R6-R20-R3)                  ;********** End in Z upper hole
R23=(R1+(R6-R20-R3))*(-1)        ;********** Start drilling below
R24=R6-R1-R3                     ;********** End hole below
R25=((R7*1000)/(3.1415*($P_TOOLR*2)))   ;********** revolutions per minute
R26=R25*R8                       ;********** feed mm/min
;*********************************************
G94 S=R25 F=R26 M3
```

```
G0 X=R4 Y=R5
G0 Z=R21
;*******************************************
G1 Z=R22 F=R26
G1 Z=R23 F=R9
G1 Z=R24 F=R26
G0 Z=R21
;*******************************************
```

```
CYCLE800(2, "TC1", 200000, 57, 0, 0, 0, 0, 0, 0, 0, 0, 1, 100, 1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	Z XY	
Swivel plane	New	
X0	0.000	
Y0	0.000	
Z0	0.000	
Swivel mode	Axis by axis	
Sequence of axes	X Y Z	
X	0.000	°
Y	0.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select	+	
Tool		

```
M30
;*************************************************************
```

III. Create a linear program for the displayed drawing and save it as "DRILL_3_DIN.MPF" and execute the following requirements:

Extend your program and catch the following possible errors when entering parameters:

1. Inside diameter larger than outside diameter
2. Feed bigger than 0.3 mm/rev
3. Cutting speed less than 5 m/min or bigger than 250 m/min
4. Feed in the pipe smaller than drilling feed
5. Wrong tool type only drills are allowed

Output a MSG with meaningful text for each error in the program (for example "ERROR_1"…).

The following code is generated in the program for the above example:

```
CYCLE800()¶
```

TC	0

```
;*****************************************
R1 = 50 ; ****************** Outside diameter
R2 = 35 ; ****************** Inside diameter
R3 = 2 ; ******************* Safety distance
R4 = -10 ; ****************** Drilling position in X
R5 = 0 ; ******************* Drilling position in Y
R6 = 0 ; ******************* Coordinate surface in Z
R7 = 80 ; ****************** Cutting speed vc in m/min
R8 = 0.15 ; ***************** Feed mm/rev for machining
R9 = 4000 ; ***************** Feed mm/min in the pipe
;*****************************************
```

```
WORKPIECE(,"",,"PIPE",320,0,-100,-80,50,35)¶
```

Blank		Pipe
⌀A	50.000	
⌀I	35.000	abs
HA	0.000	
HI	-100.000	abs

```
T="SC_DRILL_10"
D1 M6
;*****************************************
```

```
CYCLE800(0,"TC1",100000,57,-25,0,0,0,-90,0,0,0,0,-1,100,1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	No	
Swivel plane	New	
X0	-25.000	
Y0	0.000	
Z0	0.000	
Swivel mode	Axis by axis	
Sequence of axes	X Y Z	
X	0.000	°
Y	-90.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select	-	
Tool		

```
;*********************************************
R20=(R1-R2)/2                    ;********** Wall thickness
R21=R6+R3                        ;********** Start position in Z
R22=(R6-R20-R3)                  ;********** End in Z upper hole
R23=(R1+(R6-R20-R3))*(-1)  ;********** Start drilling below
R24=R6-R1-R3                      ;********** End hole below
R25=((R7*1000)/(3.1415*($P_TOOLR*2)))    ;********** Revolutions per minute
R26=R25*R8                       ;********** Feed mm/min
R27=$TC_DP1[$P_TOOLNO,$P_TOOL]; ********** Write active tooltype in R27
;QUERIES OF ERRORS
 IF R2 > R1 GOTO ERROR_1 ; ********** R1 and R2 are illogical
 IF R8 > 0.3 GOTO ERROR_2 ; ********** Input of feed rate to big
 IF (R7 < 5) OR (R7 > 250) GOTO ERROR_3 ; ********** Queries: Cutting speed too low or too
high
 IF R26 > R9 GOTO ERROR_4 ; ******* Calculation of feed for machining too high
IF (R27 < 200) OR (R27 > 205) GOTO ERROR_5; ******* Checking the right tool type drilling
;*********************************************
G94 S=R25 F=R26 M3
G0 X=R4 Y=R5
G0 Z=R21
;*********************************************
G1 Z=R22 F=R26
G1 Z=R23 F=R9
G1 Z=R24 F=R26
G0 Z=R21
;*********************************************
 GOTO FURTHER
 ERROR_1:
MSG("ATTENTION: OUTSIDE DIAMETER = "<<R1<<" INSIDE DIAMETER = "<<R2<<"
PLEASE CHECK")
M0
ERROR_2:
MSG("ATTENTION: FEED RATE FOR DRILLING TOO HIGH A VALUE OF "<<R8<<" HAS
BEEN ENTERED")
M0
;************************
ERROR_3:
MSG("ATTENTION: WRONG CUTTING SPEED HAS BEEN ENTERED WITH "<<R7<<"
M/MIN")
```

```
M0
;*************************
ERROR_4:
MSG("ATTENTION: FEED RATE BY EMPTY CUTS ARE ONLY "<<R9<<" MM/MIN")
M0
;*************************
ERROR_5:
MSG("WRONG TOOL TYPE FOR DRILLING")
M0
FURTHER:
```

```
CYCLE800(2, "TC1", 200000, 57, 0, 0, 0, 0, 0, 0, 0, 0, 1, 100, 1)¶
```

PL	G17 (XY)
TC	TC1
Retract	Z XY
Swivel plane	New
X0	0.000
Y0	0.000
Z0	0.000
Swivel mode	Axis by axis
Sequence of axes	X Y Z
X	0.000 °
Y	0.000 °
Z	0.000 °
X1	0.000
Y1	0.000
Z1	0.000
Select	+
Tool	

```
M30
```

IV. Create a linear program for the displayed drawing and save it as "DRILL_4_DIN.MPF" and execute the following requirements:

1. Expand your program and consider the point angle of the drill when drilling through.

   R28= ; current tool angle

   R29= ; calculation lead drill

2. The point angle of the active tool should be read from the tool table via high-level language in the program.

The following code is generated in the program for the above example:

```
CYCLE800()¶
```

TC	0

```
;*****************************************
R1 = 50 ; ****************** outside diameter
R2 = 35 ; ****************** inside diameter
R3 = 2 ; ******************* safety distance
R4 = -10 ; ****************** drilling position in X
R5 = 0 ; ******************* drilling position in Y
R6 = 0 ; ******************* coordinate surface in Z
R7 = 80 ; ****************** cutting speed vc in m/min
R8 = 0.15 ; ***************** feed mm/rev for machining
R9 = 4000 ; ***************** feed mm/min in the pipe
;*****************************************

WORKPIECE(,"",,"PIPE",320,0,-100,-80,50,35)¶
```

Blank		Pipe
ØA	50.000	
ØI	35.000	abs
HA	0.000	
HI	-100.000	abs

```
T="SC_DRILL_10"
D1 M6
;*****************************************
CYCLE800(0,"TC1",100000,57,-25,0,0,0,-90,0,0,0,0,-1,100,1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	No	
Swivel plane		New
X0	-25.000	
Y0	0.000	
Z0	0.000	
Swivel mode		Axis by axis
Sequence of axes		X Y Z
X	0.000	°
Y	-90.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select		-
Tool		

```
;*********************************************
;CALCULATION PARAMETERS
 R28=$TC_DP24[$P_TOOLNO,$P_TOOL] ;******* Current tool angle
 R29=$P_TOOLR/TAN(R28/2)                ; ******* Calculation lead drill
 MSG(<<R29)
M0
MSG()
;*********************************************
R20=(R1-R2)/2              ;********** Wall thickness
R21=R6+R3                  ;********** Start point in z coordinate
R22=(R6-R20-R3)-R29  ;********** First drilling depth and lead current tool
R23=(R1+(R6-R20-R3))*(-1);********* Start drillhole down
R24=R6-R1-R3-R29         ;********* End second drilling depth and lead current tool
R25=((R7*1000)/(3.1415*($P_TOOLR*2)));********** Revolutions per minute
R26=R25*R8 ; ********** Feed mm/min
R27=$TC_DP1[$P_TOOLNO,$P_TOOL]; ********** Write active tooltype in R27
;QUERIES OF ERRORS
IF R2 > R1 GOTO ERROR_1 ; ********** R1 and R2 are illogical
IF R8 > 0.3 GOTO ERROR_2 ; ********** Input of feed rate to big
 IF (R7 < 5) OR (R7 > 250) GOTO ERROR_3 ; ********** Queries: Cutting speed too low or too
high
 IF R26 > R9 GOTO ERROR_4 ; ******* Calculation of feed for machining too high
IF (R27 < 200) OR (R27 > 205) GOTO ERROR_5; ******* Checking the right tool type drilling
;*********************************************
G94 S=R25 F=R26 M3
G0 X=R4 Y=R5
G0 Z=R21
;*********************************************
G1 Z=R22 F=R26
G1 Z=R23 F=R9
G1 Z=R24 F=R26
G0 Z=R21
;*********************************************
GOTO FURTHER
ERROR_1:
MSG("ATTENTION: OUTSIDE DIAMETER = "<<R1<<" INSIDE DIAMETER = "<<R2<<"
PLEASE CHECK")
```

```
M0

ERROR_2:
MSG("ATTENTION: FEED RATE FOR DRILLING TOO HIGH A VALUE OF "<<R8<<" HAS
BEEN ENTERED")
M0

ERROR_3:
MSG("ATTENTION: WRONG CUTTING SPEED HAS BEEN ENTERED WITH "<<R7<<"
M/MIN")
M0
ERROR_4:
MSG("ATTENTION: FEED RATE BY EMPTY CUTS ARE ONLY "<<R9<<" MM/MIN")
M0

ERROR_5:
MSG("WRONG TOOL TYPE FOR DRILLING")
M0
FURTHER:
CYCLE800(2, "TC1", 200000, 57, 0, 0, 0, 0, 0, 0, 0, 0, 1, 100, 1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	Z XY	
Swivel plane	New	
X0	0.000	
Y0	0.000	
Z0	0.000	
Swivel mode	Axis by axis	
Sequence of axes	X Y Z	
X	0.000	°
Y	0.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select	+	
Tool		

```
M30
;*********************************************************************
```

V. Create a linear program for the displayed drawing and save it as "DRILL_5_DIN.MPF" and execute the following requirements:

1. Add a parameter R10 as a reduced feed rate value for spot drilling without centering and before the drill bit cuts out of the material.
2. Add a parameter R11 as depth for reduced feed value for spot drilling without centering and before the drill cuts out of the material.
3. R10 = 0.08 ; feed rate mm/rev. spotting and through drilling
4. R11 = 1 ; incremental distance for spot and through drilling
5. Catch an error when the spot/through drilling path is bigger than the wall thickness.

The following code is generated in the program for the above example:

```
CYCLE800()¶
          TC              0
;*****************************************
R1 = 50 ; ****************** outside diameter
R2 = 35 ; ****************** inside diameter
R3 = 2 ; ******************* safety distance
R4 = -10 ; ***************** drilling position in X
R5 = 0 ; ******************* drilling position in Y
R6 = 0 ; ******************* coordinate surface in Z
R7 = 80 ; ****************** cutting speed vc in m/min
R8 = 0.15 ; **************** feed mm/rev for machining
R9 = 4000 ; **************** feed mm/min in the pipe
;ADDITIONAL PARAMETERS FOR TASK NO 5
R10=0.08 ;************ Reduced feed in mm/rev for spot and through drilling
R11=1    ;************ Incremental spot and through drilling depth
;*****************************************
UORKPIECE(,"",,"PIPE",320,-10,-100,-80,R1,R2)¶
          Blank           Pipe
          ØA        R1
          ØI        R2  abs

          HA      -10.000
          HI     -100.000  abs
T="SC_DRILL_10"
D1 M6
;*****************************************
```

```
CYCLE800(0, "TC1", 100000, 57, -25, 0, 0, 0, -90, 0, 0, 0, 0, -1, 100, 1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	No	
Swivel plane	New	
X0	-25.000	
Y0	0.000	
Z0	0.000	
Swivel mode	Axis by axis	
Sequence of axes	X Y Z	
X	0.000	°
Y	-90.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select	-	
Tool		

```
;*****************************************
;CALCULATION PARAMETERS
 R28=$TC_DP24[$P_TOOLNO,$P_TOOL] ;******* Current tool angle
 R29=$P_TOOLR/TAN(R28/2)                ; ******* Calculation lead drill
 MSG(<<R29)
M0
MSG()
;*********************************************
R20=(R1-R2)/2              ;************ Wall thickness
R21=R6+R3                  ;************ Start point in z coordinate
R22=(R6-R20-R3)-R29  ;************ First drilling depth and lead current tool
R23=(R1+(R6-R20-R3))*(-1);********* Start drillhole down
R24=R6-R1-R3-R29         ;********* End second drilling depth and lead current tool
R25=((R7*1000)/(3.1415*($P_TOOLR*2)));********** Revolutions per minute
R26=R25*R8             ; ********** Feed mm/min
R27=$TC_DP1[$P_TOOLNO,$P_TOOL]; ********** Write active tooltype in R27
R30=R25*R10            ; ******* Feed in mm/min for spot and through drilling
;QUERIES OF ERRORS
IF R2 > R1 GOTO ERROR_1 ; ********** R1 and R2 are illogical
IF R8 > 0.3 GOTO ERROR_2 ; ********** Input of feed rate to big
 IF (R7 < 5) OR (R7 > 250) GOTO ERROR_3 ; ********** Queries: Cutting speed too low or too
high
 IF R26 > R9 GOTO ERROR_4 ; ******* Calculation of feed for machining too high
```

```
IF (R27 < 200) OR (R27 > 205) GOTO ERROR_5; ******* Checking the right tool type drilling
;CHECKING THAT THIKNESS PIPE IS BIGGER THAN SPOT AND THROUGH DRILLING
VALUE
IF R20 < R11 GOTO ERROR_6
;*********************************************
G94 S=R25 F=R26 M3
G0 X=R4 Y=R5
G0 Z=R21
;*********************************************
G1 Z=R6-R11-R29 F=R30     ; ********** Spot drilling
G1 Z=R6-R20+R11 F=R26     ; ********** Drilling with normal feed
G1 Z=R22 F=R30            ; ********** Through drilling
G1 Z=R23 F=R9  ; ********** Second start position for spot drilling inside the pipe
G1 Z=(R1+(R6-R20)+R11+R29)*(-1) F=R30 ; **********Second spot drilling
G1 Z=R6-R1+R11 F=R26      ; ********** Second drilling with normal feed
G1 Z=R24 F=R30            ; ********** Second through drilling
G0 Z=R21
;*********************************************
GOTO FURTHER
ERROR_1:
MSG("ATTENTION: OUTSIDE DIAMETER = "<<R1<<" INSIDE DIAMETER = "<<R2<<"
PLEASE CHECK")
M0

ERROR_2:
MSG("ATTENTION: FEED RATE FOR DRILLING TOO HIGH A VALUE OF "<<R8<<" HAS
BEEN ENTERED")
M0

ERROR_3:
MSG("ATTENTION: WRONG CUTTING SPEED HAS BEEN ENTERED WITH "<<R7<<"
M/MIN")
M0

ERROR_4:
MSG("ATTENTION: FEED RATE BY EMPTY CUTS ARE ONLY "<<R9<<" MM/MIN")
```

M0

ERROR_5:
MSG("WRONG TOOL TYPE FOR DRILLING")
M0

ERROR_6:
MSG("ATTENTION: WALL THICKNESS "<<R20<<" IS SMALLER THAN SPOT AND THROUGH DRILLING VALUE "<< R11)
M0
FURTHER:

```
CYCLE800(2, "TC1", 200000, 57, 0, 0, 0, 0, 0, 0, 0, 0, 1, 100, 1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	Z XY	
Swivel plane	New	
X0	0.000	
Y0	0.000	
Z0	0.000	
Swivel mode	Axis by axis	
Sequence of axes	X Y Z	
X	0.000	°
Y	0.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select	+	
Tool		

M30
;$$**$$

VI. Create a linear program for the displayed drawing and save it as "DRILL_6_DIN.MPF" and execute the following requirements:

1. Reduce the speed when spotting and drilling through by the value entered in R12 in %, like for example 10 percent lesser revolutions.
   R12 = 10 ; speed reduction percentage
2. Replace the diameters for "WORKPIECE" in the blank description with R parameters at the beginning of the program.
3. Replace the X value of the zero offset with R parameters in CYCLE800 at the beginning of the program.

The following code is generated in the program for the above example:

```
CYCLE800()¶
            TC              0

;*****************************************
R1 = 50 ; ***************** outside diameter
R2 = 35 ; ***************** inside diameter
R3 = 2 ; ****************** safety distance
R4 = -10 ; **************** drilling position in X
R5 = 0 ; ****************** drilling position in Y
R6 = 0 ; ****************** coordinate surface in Z
R7 = 80 ; ***************** cutting speed vc in m/min
R8 = 0.15 ; *************** feed mm/rev for machining
R9 = 4000 ; *************** feed mm/min in the pipe
;ADDITIONAL PARAMETERS FOR TASK NO 5
R10=0.08 ;************ Reduced feed in mm/rev for spot and through drilling
R11=1    ;************ Incremental spot and through drilling depth
;ADDITIONAL PARAMETERS FOR TASK NO 6
R12=10   ; ************ Percentage speed reduction for spot and through drilling
;*****************************************
WORKPIECE(,"",,"PIPE",320,-10,-100,-80,R1,R2)¶
            Blank           Pipe
            ⌀A              R1
            ⌀I              R2  abs


            HA          -10.000
            HI          -100.000 abs
T="SC_DRILL_10"
D1 M6
;*****************************************
```

```
CYCLE800(0, "TC1", 100000, 57, -R1/2, 0, 0, 0, -90, 0, 0, 0, 0, -1, 100, 1)¶
```

PL	G17 (XY)
TC	TC1
Retract	No
Swivel plane	New
X0	-R1/2
Y0	0.000
Z0	0.000
Swivel mode	Axis by axis
Sequence of axes	X Y Z
X	0.000 °
Y	-90.000 °
Z	0.000 °
X1	0.000
Y1	0.000
Z1	0.000
Select	-
Tool	

```
;*******************************************
;CALCULATION PARAMETERS
 R28=$TC_DP24[$P_TOOLNO,$P_TOOL] ;******* Current tool angle
 R29=$P_TOOLR/TAN(R28/2)                ; ******* Calculation lead drill
 MSG(<<R29)
M0
MSG()
;*********************************************
R20=(R1-R2)/2              ;********** Wall thickness
R21=R6+R3                  ;********** Start point in z coordinate
R22=(R6-R20-R3)-R29  ;********** First drilling depth and lead current tool
R23=(R1+(R6-R20-R3))*(-1);********* Start drillhole down
R24=R6-R1-R3-R29         ;********* End second drilling depth and lead current tool
R25=((R7*1000)/(3.1415*($P_TOOLR*2)));********** Revolutions per minute
R26=R25*R8               ; ********** Feed mm/min
R27=$TC_DP1[$P_TOOLNO,$P_TOOL]; ********** Write active tooltype in R27
R30=R25*R10              ; ******* Feed in mm/min for spot and through drilling
;QUERIES OF ERRORS
IF R2 > R1 GOTO ERROR_1 ; ********** R1 and R2 are illogical
IF R8 > 0.3 GOTO ERROR_2 ; ********** Input of feed rate to big
IF (R7 < 5) OR (R7 > 250) GOTO ERROR_3 ; ********** Queries: Cutting speed too low or too
high
IF R26 > R9 GOTO ERROR_4 ; ******* Calculation of feed for machining too high
```

```
IF (R27 < 200) OR (R27 > 205) GOTO ERROR_5; ******* Checking the right tool type drilling
;Checking that thikness pipe is bigger than spot and through drilling value
IF R20 < R11 GOTO ERROR_6
;*********************************************
G94 S=R25 F=R26 M3
G0 X=R4 Y=R5
G0 Z=R21
;*********************************************
G1 Z=R6-R11-R29 F=R30 S=(R25-(R25/100*R12)) ; ****** Spot drilling with reduced
revolutions
G1 Z=R6-R20+R11 F=R26 S=R25 ; ****** Drilling with normal feed
G1 Z=R22 F=R30 S=(R25-(R25/100*R12)) ; ****** Through drilling with reduced revolutions
G1 Z=R23 F=R9 ; ****** Second start position for spot drilling inside the pipe
G1 Z=(R1+(R6-R20)+R11+R29)*(-1) F=R30 S=(R25-(R25/100*R12)); ****** Second spot
drilling with reduced revolution
G1 Z=R6-R1+R11 F=R26 S=R25 ; ****** Second drilling with normal feed
G1 Z=R24 F=R30 S=(R25-(R25/100*R12)); ****** Second through drilling with reduced
revolution
G0 Z=R21
;*********************************************
GOTO FURTHER
ERROR_1:
MSG("ATTENTION: OUTSIDE DIAMETER = "<<R1<<" INSIDE DIAMETER = "<<R2<<"
PLEASE CHECK")
M0
ERROR_2:
MSG("ATTENTION: FEED RATE FOR DRILLING TOO HIGH A VALUE OF "<<R8<<" HAS
BEEN ENTERED")
M0

ERROR_3:
MSG("ATTENTION: WRONG CUTTING SPEED HAS BEEN ENTERED WITH "<<R7<<"
M/MIN")
M0

ERROR_4:
MSG("ATTENTION: FEED RATE BY EMPTY CUTS ARE ONLY "<<R9<<" MM/MIN")
M0
```

```
ERROR_5:
MSG("WRONG TOOL TYPE FOR DRILLING")
M0

ERROR_6:
MSG("ATTENTION: WALL THICKNESS "<<R20<<" IS SMALLER THAN SPOT AND
THROUGH DRILLING VALUE "<< R11)
M0
FURTHER:
CYCLE800(2, "TC1", 200000, 57, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 100, 1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	Z XY	
Swivel plane		New
X0	0.000	
Y0	0.000	
Z0	0.000	
Swivel mode		Axis by axis
Sequence of axes		X Y Z
X	0.000	°
Y	0.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select		+
Tool		

```
M30
;*****************************************************************
```

VII. Program the program and perform the following requirements:

1. Create a main program for the displayed drawing and save it under the name “DRILL_7_MAIN.MPF” based on the program of the exercise done above “DRILL_6_DIN.MPF”
2. Create a subroutine named “DRILL_7_SUB.SPF” and call it from the main program

The following code is generated in the program for the above example:

```
CYCLE800()¶
                    TC              0

CYCLE800(0, "TC1", 200000, 57, 0, 0, 0, 0, 0, 0, 0, 0, 1, 100, 1)¶
                    PL          G17 (XY)
                    TC                TC1
                    Retract         No
                    Swivel plane          New
                    X0              0.000
                    Y0              0.000
                    Z0              0.000
                    Swivel mode         Axis by axis
                    Sequence of axes      X Y Z
                    X               0.000 °
                    Y               0.000 °
                    Z               0.000 °
                    X1              0.000
                    Y1              0.000
                    Z1              0.000
                    Select                +
                    Tool
;*****************************************
R1 = 50 ; ****************** outside diameter
R2 = 35 ; ****************** inside diameter
R3 = 2 ; ******************* safety distance
R4 = -10 ; ***************** drilling position in X
R5 = 0 ; ******************* drilling position in Y
R6 = 0 ; ******************* coordinate surface in Z
R7 = 80 ; ****************** cutting speed v_c in m/min
R8 = 0.15 ; **************** feed mm/rev for machining
R9 = 4000 ; **************** feed mm/min in the pipe
;ADDITIONAL PARAMETERS FOR TASK NO 5
R10=0.08 ;************* Reduced feed in mm/rev for spot and through drilling
R11=1    ;************* Incremental spot and through drilling depth
;ADDITIONAL PARAMETERS FOR TASK NO 6
R12=10   ; ************* Percentage speed reduction for spot and through drilling
;*****************************************
```

```
WORKPIECE(,"",,"PIPE",320,-10,-100,-80,R1,R2)¶
```

Blank	Pipe	
ØA	R1	
ØI	R2	abs
HA	-10.000	
HI	-100.000	abs

```
T="SC_DRILL_10"
D1 M6
;*******************************************
CYCLE800(0,"TC1",100000,57,-R1/2,0,0,0,-90,0,0,0,0,-1,100,1)¶
```

PL	G17 (XY)	
TC	TC1	
Retract	No	
Swivel plane	New	
X0	-R1/2	
Y0	0.000	
Z0	0.000	
Swivel mode	Axis by axis	
Sequence of axes	X Y Z	
X	0.000	°
Y	-90.000	°
Z	0.000	°
X1	0.000	
Y1	0.000	
Z1	0.000	
Select	-	
Tool		

```
;*******************************************
;First drill hole
CALL "DRILL_7_SUB.SPF"
 ;Second drill hole
R4=-30
CALL "DRILL_7_SUB.SPF"
 ;Third drill hole
R4=-45
CALL "DRILL_7_SUB.SPF"
```

```
CYCLE800(2, "TC1", 200000, 57, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 100, 1)¶
```

PL	G17 (XY)
TC	TC1
Retract	Z XY
Swivel plane	New
X0	0.000
Y0	0.000
Z0	0.000
Swivel mode	Axis by axis
Sequence of axes	X Y Z
X	0.000 °
Y	0.000 °
Z	0.000 °
X1	0.000
Y1	0.000
Z1	0.000
Select	+
Tool	

```
T0 M6
M30
;***********************************************
```

2. Subprogram "DRILL_7_SUB.SPF"

```
;CALCULATION PARAMETERS
 R28=$TC_DP24[$P_TOOLNO,$P_TOOL] ;******* Current tool angle
 R29=$P_TOOLR/TAN(R28/2)                ; ******* Calculation lead drill
 MSG(<<R29)
M0
MSG()
;***********************************************
R20=(R1-R2)/2              ;********** Wall thickness
R21=R6+R3                  ;********** Start point in z coordinate
R22=(R6-R20-R3)-R29  ;********** First drilling depth and lead current tool
R23=(R1+(R6-R20-R3))*(-1);********* Start drillhole down
R24=R6-R1-R3-R29         ;********* End second drilling depth and lead current tool
R25=((R7*1000)/(3.1415*($P_TOOLR*2)));********** Revolutions per minute
R26=R25*R8               ; ********** Feed mm/min
R27=$TC_DP1[$P_TOOLNO,$P_TOOL]; ********** Write active tooltype in R27
R30=R25*R10              ; ******* Feed in mm/min for spot and through drilling
;QUERIES OF ERRORS
IF R2 > R1 GOTO ERROR_1 ; ********** R1 and R2 are illogical
IF R8 > 0.3 GOTO ERROR_2 ; ********** Input of feed rate to big
```

```
IF (R7 < 5) OR (R7 > 250) GOTO ERROR_3 ; ********** Queries: Cutting speed too low or too high
IF R26 > R9 GOTO ERROR_4 ; ******* Calculation of feed for machining too high
IF (R27 < 200) OR (R27 > 205) GOTO ERROR_5; ******* Checking the right tool type drilling
;Checking that thikness pipe is bigger than spot and through drilling value
IF R20 < R11 GOTO ERROR_6
;*********************************************
G94 S=R25 F=R26 M3
G0 X=R4 Y=R5
G0 Z=R21
;*********************************************
G1 Z=R6-R11-R29 F=R30 S=(R25-(R25/100*R12)) ; ****** Spot drilling with reduced revolutions
G1 Z=R6-R20+R11 F=R26 S=R25 ; ****** Drilling with normal feed
G1 Z=R22 F=R30 S=(R25-(R25/100*R12)) ; ****** Through drilling with reduced revolutions
G1 Z=R23 F=R9 ; ****** Second start position for spot drilling inside the pipe
G1 Z=(R1+(R6-R20)+R11+R29)*(-1) F=R30 S=(R25-(R25/100*R12)) ; ****** Second spot drilling with reduced revolution
G1 Z=R6-R1+R11 F=R26 S=R25 ; ****** Second drilling with normal feed
G1 Z=R24 F=R30 S=(R25-(R25/100*R12)) ; ****** Second through drilling with reduced revolution
G0 Z=R21
;*********************************************
GOTO FURTHER
ERROR_1:
MSG("ATTENTION: OUTSIDE DIAMETER = "<<R1<<" INSIDE DIAMETER = "<<R2<<" PLEASE CHECK")
M0

ERROR_2:
MSG("ATTENTION: FEED RATE FOR DRILLING TOO HIGH A VALUE OF "<<R8<<" HAS BEEN ENTERED")
M0

ERROR_3:
MSG("ATTENTION: WRONG CUTTING SPEED HAS BEEN ENTERED WITH "<<R7<<" M/MIN")
M0
```

```
ERROR_4:
MSG("ATTENTION: FEED RATE BY EMPTY CUTS ARE ONLY "<<R9<<" MM/MIN")
M0

ERROR_5:
MSG("WRONG TOOL TYPE FOR DRILLING")
M0

ERROR_6:
MSG("ATTENTION: WALL THICKNESS "<<R20<<" IS SMALLER THAN SPOT AND
THROUGH DRILLING VALUE "<< R11)

FURTHER:
M17
;*****************************************************************
```

VIII. Use the program and execute the request to create files in the following "User Cycles" folder:

Copy "DRILL_7_PARA_SUB.SPF" from subroutine into "User cycles"

To make the above request, do the following steps:

1. Copy "DRILL_7_SUB.SPF" from "Program Manager"
2. Select "MENU SELECT"
3. Select "Setup"
4. Select "System data"
5. Open "NC data"
6. Open "Cycles"
7. Open "User cycles" and paste it here
8. Rename the program to "CYCLE_DRILL".

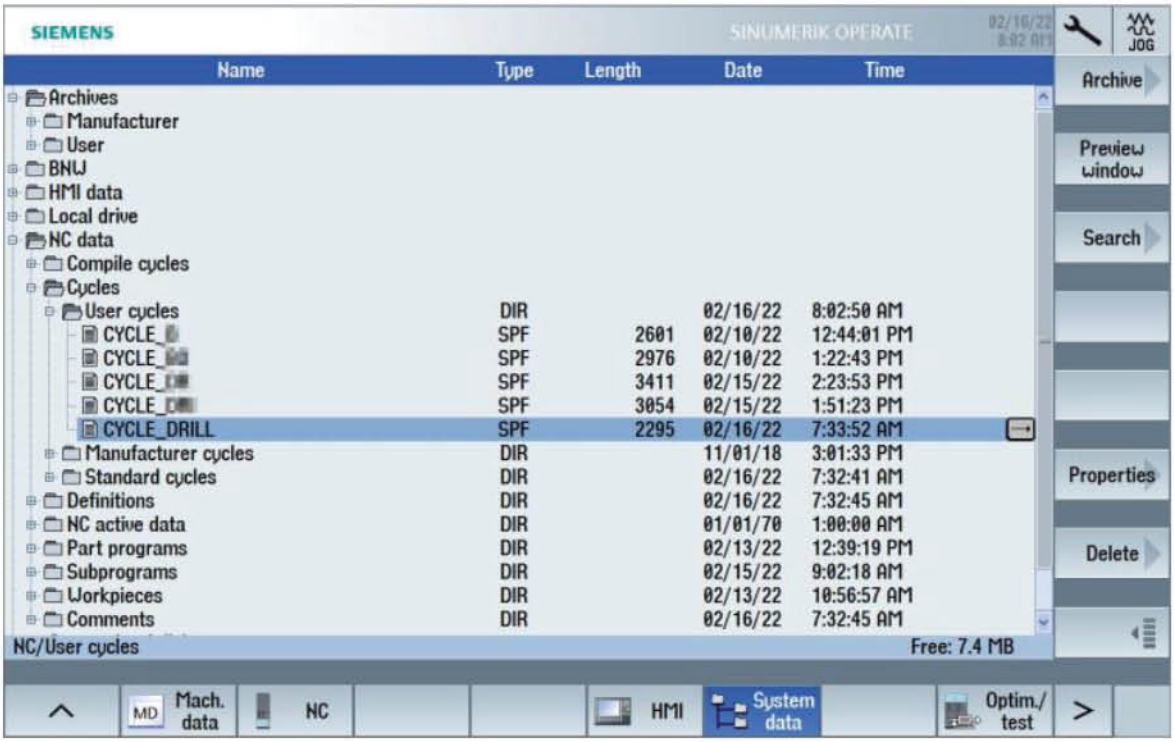


Figure 4.14: Copy the program to User Cycles

IX. Use the program "CYCLE_DRILL.SPF" just created in the folder "User Cycles" and execute the following request:

1. Input the parameter R1 until R12 at the beginning, copy from: "DRILL_7_PARA_MAIN.MPF" and ignore in this new "CYCLE_DRILL" the parameter R4 and R5 by deactivate both parameters with ";".

To make the above request, do the following steps:

1. Open "CYCLE_DRILL.SPF" in "User Cycles" folder
2. Copy parameters R1 to R12 from file "DRILL_7_MAIN.MPF"
3. Paste before the first command line in "CYCLE_DRILL.SPF"
4. Ignore parameters R4 and R5 by deactivating both parameters with a ";" in front of them.
5. Deactivate the quick tool call to positions X = R4 and Y = R5.

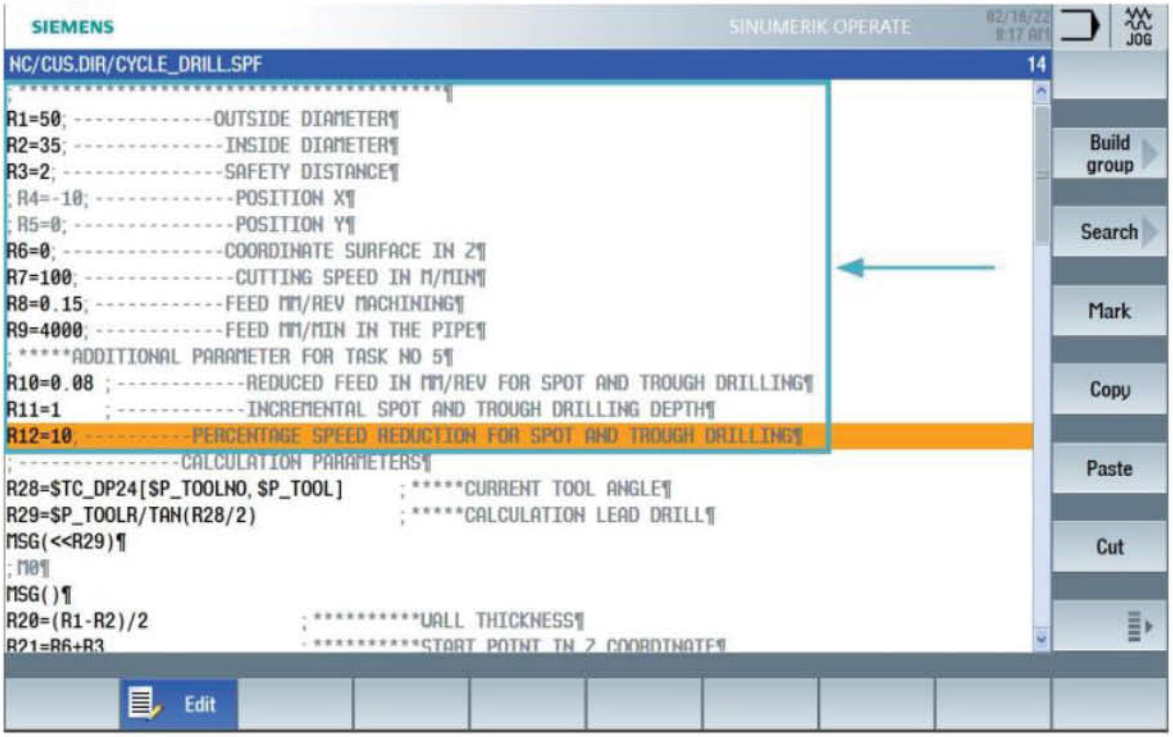


Figure 4.15: Copy parameters R1 to R12 and deactivate R4, R5

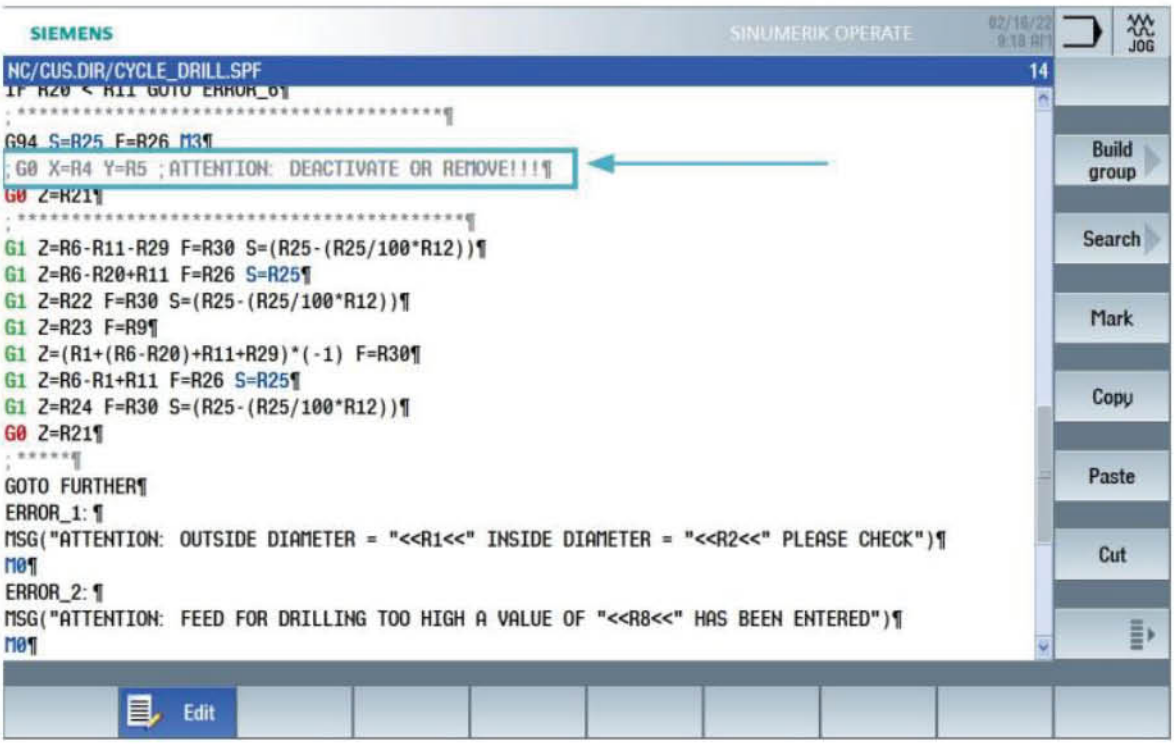


Figure 4.16: Deactivate the command line that calls the quick tool to positions X=R4 and Y=R5

X. Use the program "CYCLE_DRILL.SPF" just created in the folder "User Cycles" and execute the following request:

1. Copy the orange head bar "PROC CYCLE81(REAL…)" from the "Standard cycle" of the "CYCLE81" to "CYCLE_DRILL" to create the first instruction of the program with variable declaration that takes values from the main program.

To make the above request, do the following steps:

1. Open "CYCLE81" in "Standard cycle"
2. Copy the orange header "PROC CYCLE81(REAL…)"

Figure 4.17: Copy the orange header in "CYCLE81"

3. 3. Paste before the first command line in "CYCLE_DRILL.SPF"

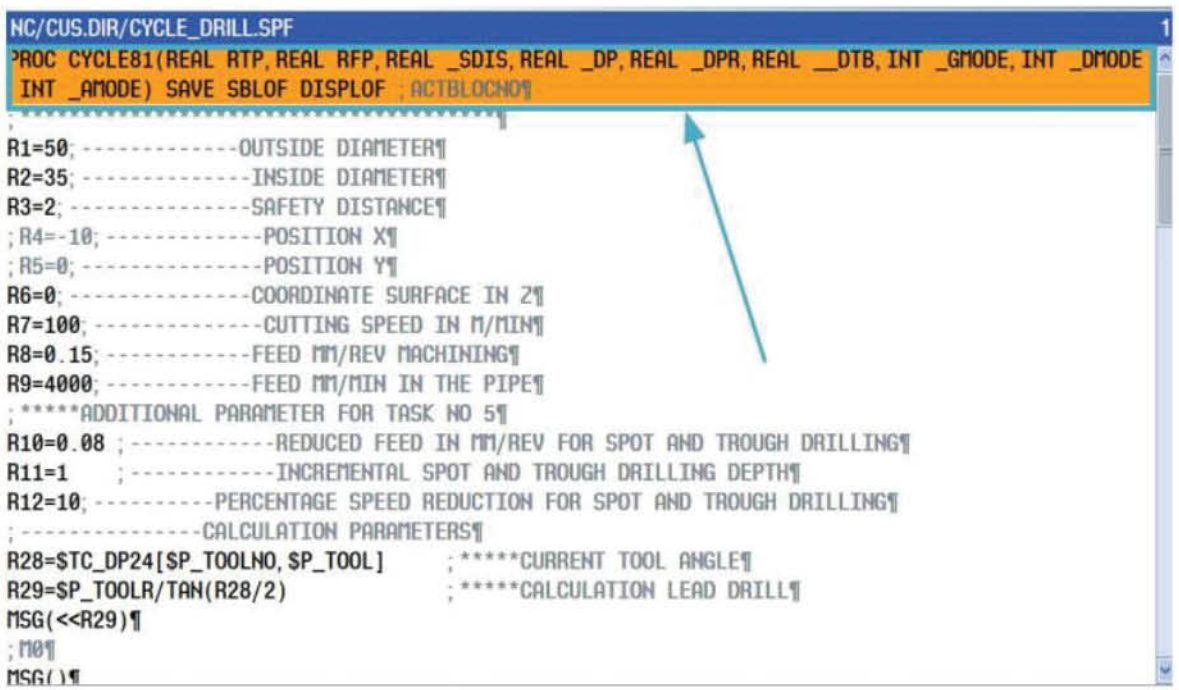


Figure 4.18: Paste the first instruction of the program with the variable declaration to get the values from the main program

XI. Use the program "CYCLE_DRILL.SPF" just created in the folder "User Cycles" and execute the following request:

1. Create a cycle from this program. In order to do that define for every "R" parameter R1 until R12 (without R4 and R5) a "user variable" and insert the user variables as well in the head line "PROC CYCLE_DRILL(REAL…)":

To make the above request, do the following steps:

1. Open "CYCLE_DRILL.SPF" in "User Cycles" folder
2. In the orange subject line just copied the word "CYCLE81" open the "Edit" tab to edit.
3. Insert the following user variables in the above head line:

PROC CYCLE_DRILL(REAL _OUD, REAL _IND, REAL _SDIS, REAL _COZ, REAL _VC1, REAL _FEM, REAL _FEP, REAL _FES, REAL _ISD, REAL _PRD) SAVE SBLOF DISPLOF ;ACTBLOCNO

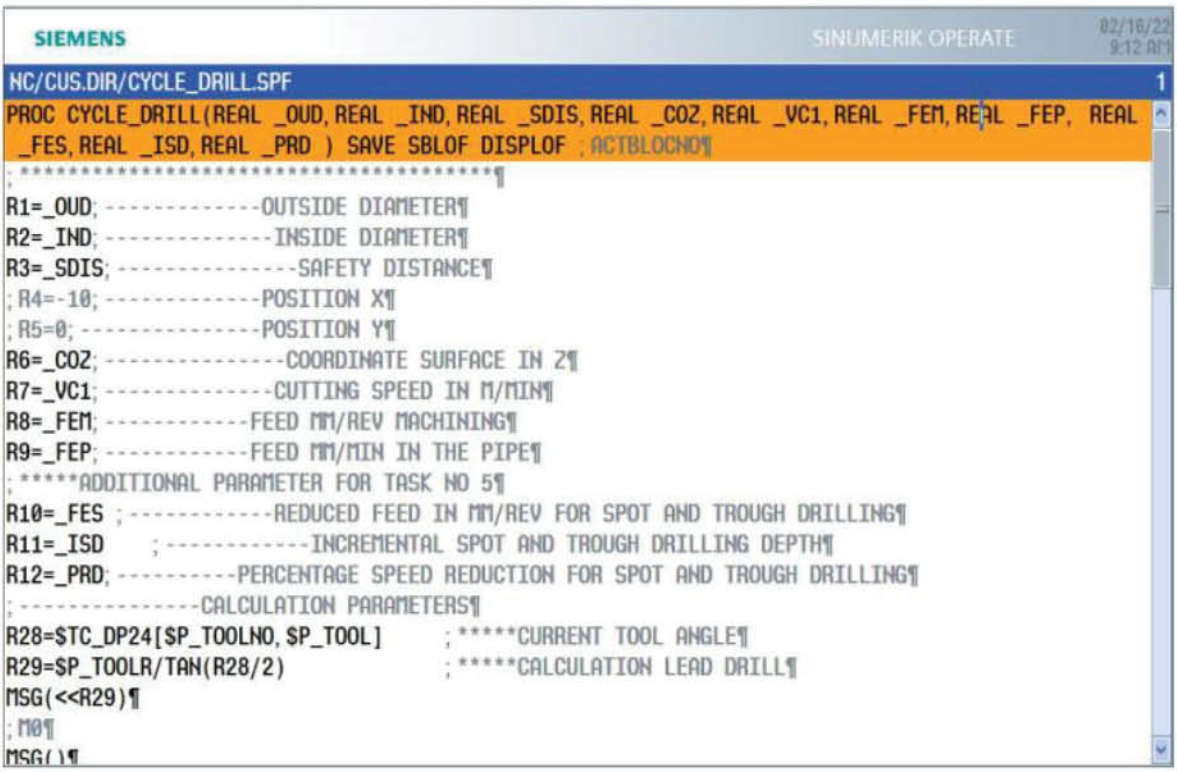


Figure 4.19: Inserts user variables in the instruction header line to get values from the main program.

4. Declare the user variable in the "CYCLE_DRILL.SPF" subroutine as follows:

PROC CYCLE_DRILL(REAL _OUD, REAL _IND, REAL _SDIS, REAL _COZ, REAL _VC1, REAL _FEM, REAL _FEP, REAL _FES, REAL _ISD, REAL _PRD) SAVE SBLOF DISPLOF ;ACTBLOCNO

;**********

R1 = _OUD ; ********** outside diameter

```
R2 = _IND  ; *****************  inside diameter
R3 = _SDIS ; ***************** safety distance
;R4 = -10    ; ***************** drilling position in X
;R5 = 0      ; *****************  drilling position in Y
R6 = _COZ ; *****************  coordinate surface in Z
R7 = _VC1  ; ***************** cutting speed vc in m/min
R8 = _FEM ; ***************** feed mm/rev for machining
R9 = _FEP  ; *****************  feed mm/min in the pipe
;ADDITIONAL PARAMETERS FOR TASK NO 5
R10 = _FES; ************ Reduced feed in mm/rev for spot and through drilling  R11 = _ISD  ;
************ Incremental spot and through drilling depth
; ADDITIONAL PARAMETERS FOR TASK NO 6
R12= _PRD  ; ************ Percentage speed reduction for spot and through drilling
;****************************************
;CALCULATION PARAMETERS
 R28=$TC_DP24[$P_TOOLNO,$P_TOOL] ;******* Current tool angle
 R29=$P_TOOLR/TAN(R28/2)                ; ******* Calculation lead drill
 MSG(<<R29)
M0
MSG()
;***********************************************
R20=(R1-R2)/2              ;************ Wall thickness
R21=R6+R3                  ;************ Start point in z coordinate
R22=(R6-R20-R3)-R29  ;************ First drilling depth and lead current tool
R23=(R1+(R6-R20-R3))*(-1);********* Start drillhole down
R24=R6-R1-R3-R29         ;********* End second drilling depth and lead current tool
R25=((R7*1000)/(3.1415*($P_TOOLR*2)));********** Revolutions per minute
R26=R25*R8             ; ********** Feed mm/min
R27=$TC_DP1[$P_TOOLNO,$P_TOOL]; ********** Write active tooltype in R27
R30=R25*R10            ; ******* Feed in mm/min for spot and through drilling
;QUERIES OF ERRORS
IF R2 > R1 GOTO ERROR_1 ; ********** R1 and R2 are illogical
IF R8 > 0.3 GOTO ERROR_2 ; ********** Input of feed rate to big
IF (R7 < 5) OR (R7 > 250) GOTO ERROR_3 ; ********** Queries: Cutting speed too low or too
high
IF R26 > R9 GOTO ERROR_4 ; ******* Calculation of feed for machining too high
IF (R27 < 200) OR (R27 > 205) GOTO ERROR_5; ******* Checking the right tool type drilling
;Checking that thikness pipe is bigger than spot and through drilling value
IF R20 < R11 GOTO ERROR_6
```

```
;*********************************************
G94 S=R25 F=R26 M3
;G0 X=R4 Y=R5 ;ATTENTION DEACTIVATE OR REMOVE
G0 Z=R21
;*********************************************

G1 Z=R6-R11-R29 F=R30 S=(R25-(R25/100*R12)) ; ****** Spot drilling with reduced revolutions
G1 Z=R6-R20+R11 F=R26 S=R25 ; ****** Drilling with normal feed
G1 Z=R22 F=R30 S=(R25-(R25/100*R12)) ; ****** Through drilling with reduced revolutions
G1 Z=R23 F=R9 ; ****** Second start position for spot drilling inside the pipe
G1 Z=(R1+(R6-R20)+R11+R29)*(-1) F=R30 S=(R25-(R25/100*R12)); ****** Second spot drilling with reduced revolution
G1 Z=R6-R1+R11 F=R26 S=R25 ; ****** Second drilling with normal feed
G1 Z=R24 F=R30 S=(R25-(R25/100*R12)); ****** Second through drilling with reduced revolution
G0 Z=R21
;*********************************************
GOTO FURTHER
ERROR_1:
MSG("ATTENTION: OUTSIDE DIAMETER = "<<R1<<" INSIDE DIAMETER = "<<R2<<" PLEASE CHECK")
M0
ERROR_2:
MSG("ATTENTION: FEED RATE FOR DRILLING TOO HIGH A VALUE OF "<<R8<<" HAS BEEN ENTERED")
M0

ERROR_3:
MSG("ATTENTION: WRONG CUTTING SPEED HAS BEEN ENTERED WITH "<<R7<<" M/MIN")
M0

ERROR_4:
MSG("ATTENTION: FEED RATE BY EMPTY CUTS ARE ONLY "<<R9<<" MM/MIN")
M0

ERROR_5:
MSG("WRONG TOOL TYPE FOR DRILLING")
M0
```

```
ERROR_6:
MSG("ATTENTION: WALL THICKNESS "<<R20<<" IS SMALLER THAN SPOT AND
THROUGH DRILLING VALUE "<< R11)
M0
FURTHER:
M17
;*****************************************************************
```

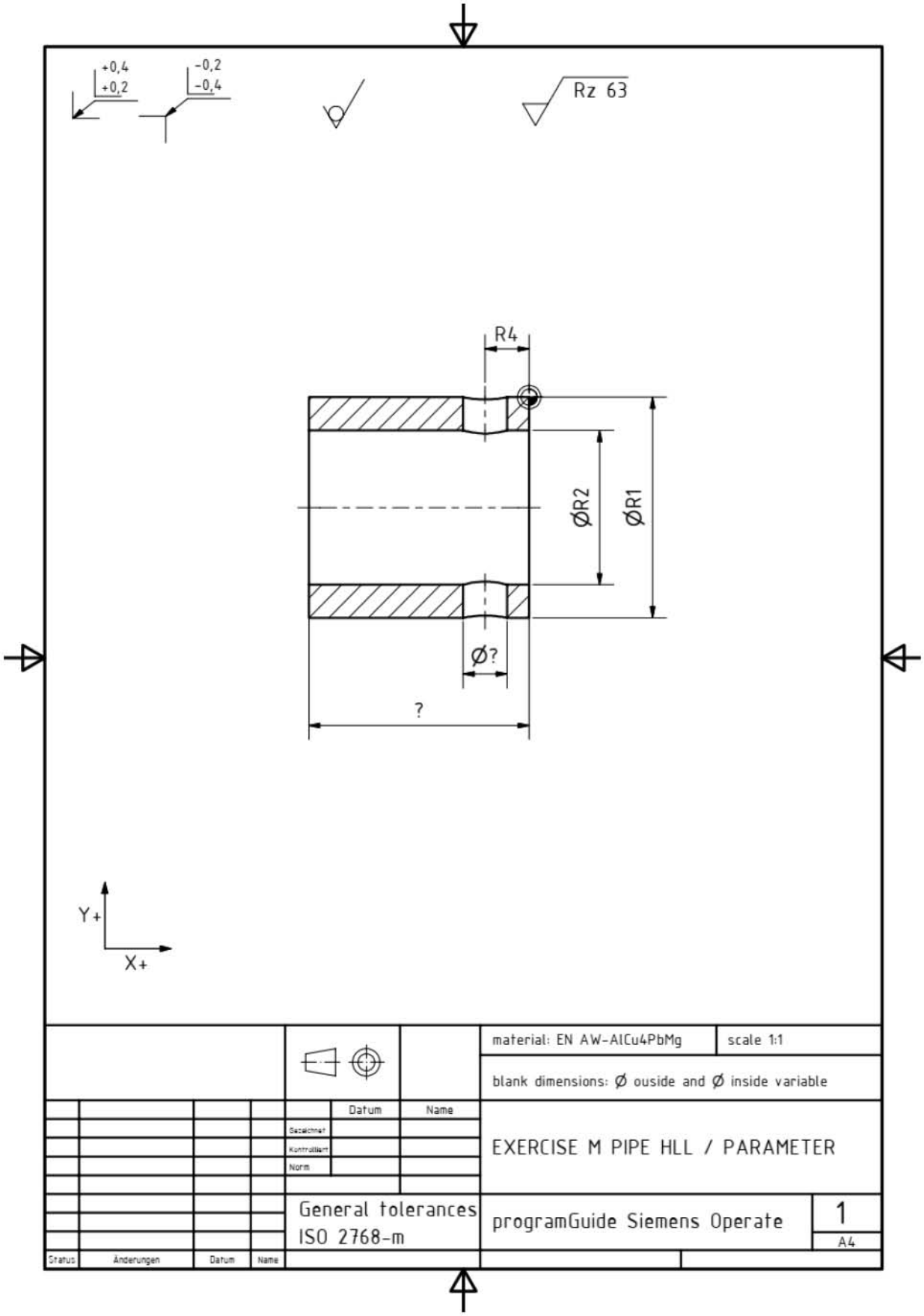
3.10. Plan and realise operational work order of the industry

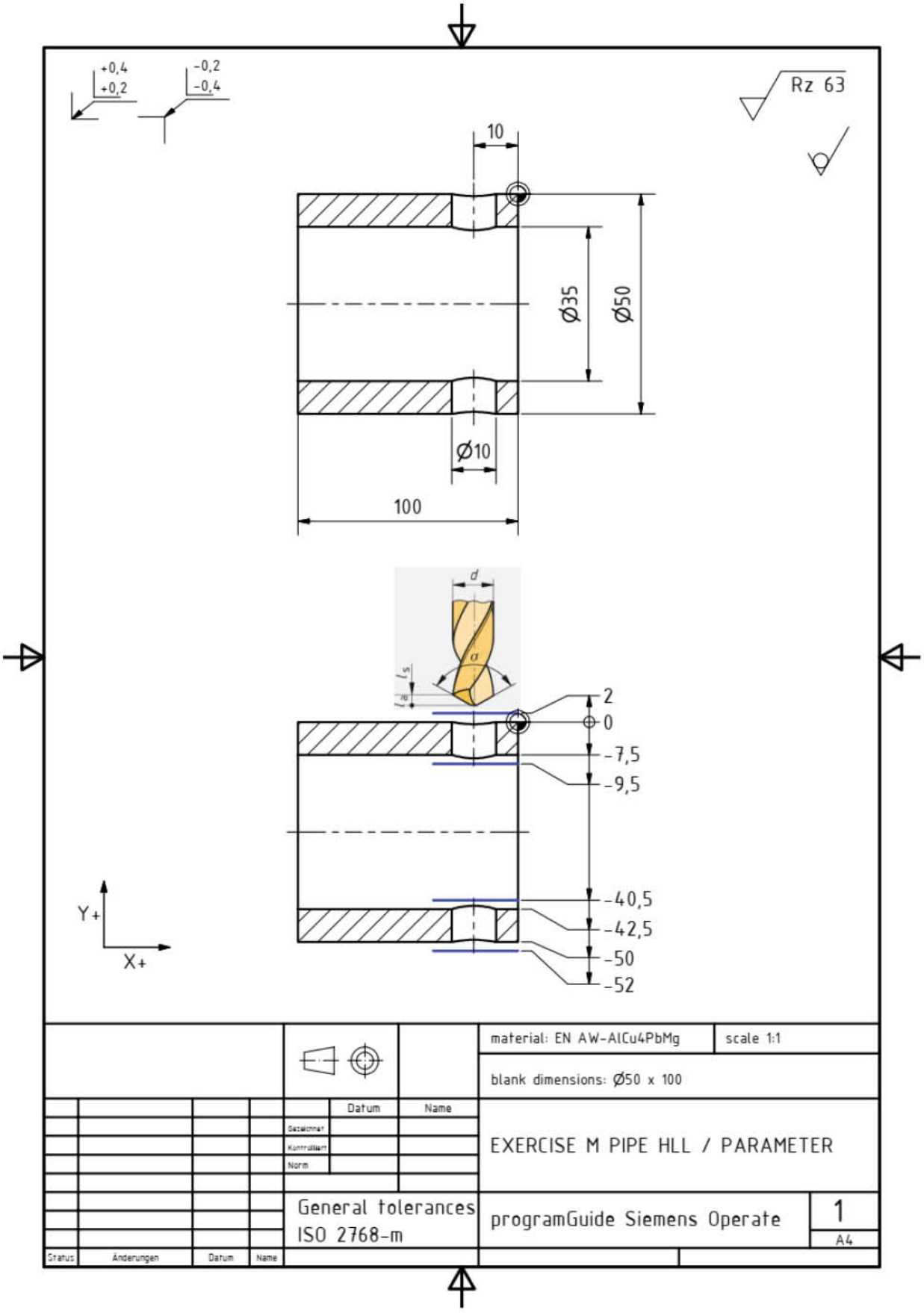
3.10.1. Use programGUIDE with cycles and G-codes to program the machining of a simple drill piece using R parameters, user variables, system variables, loops, calculations, etc.

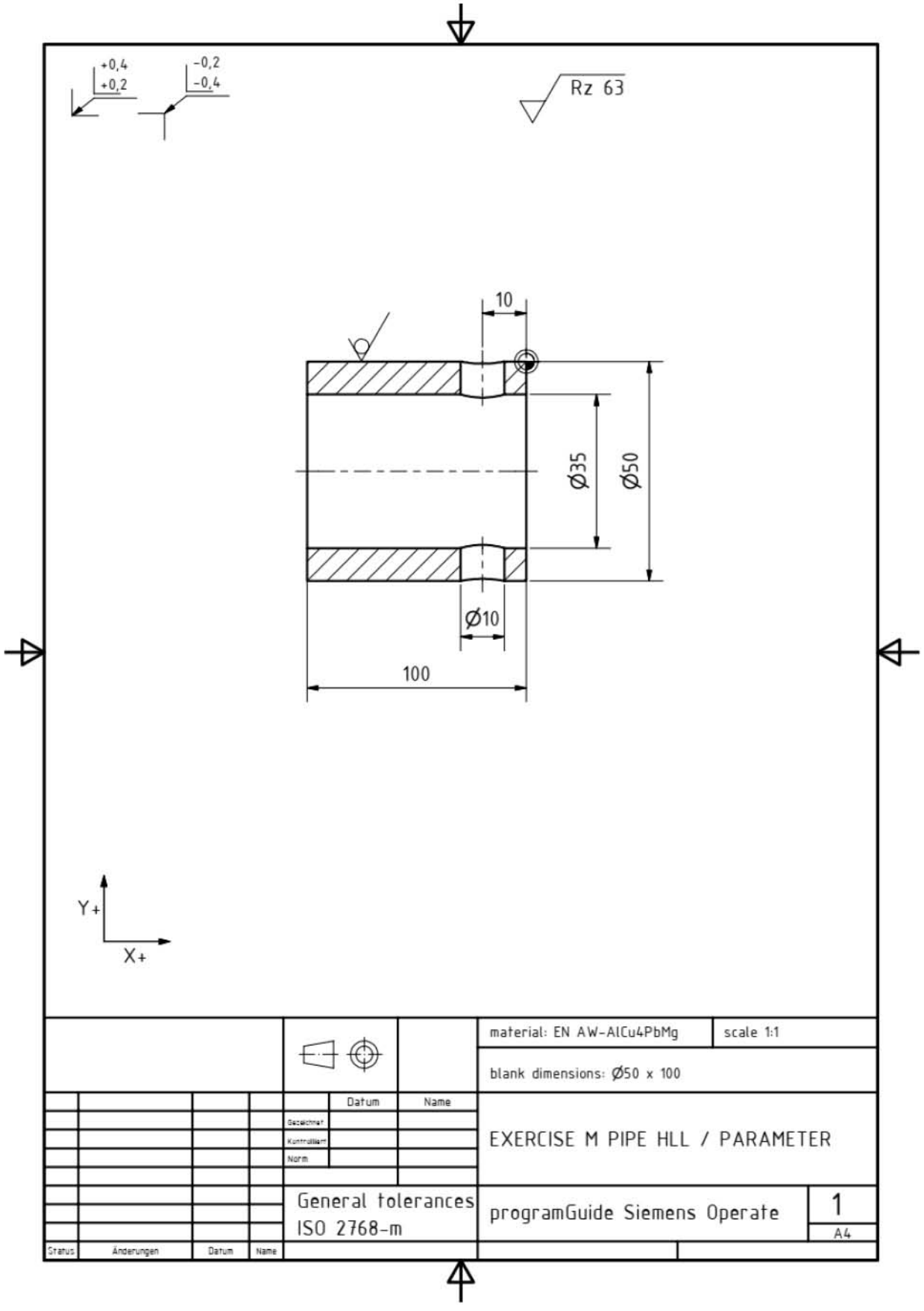
a. Objective:

- Programming application with cycles, G code, using R parameters, user variables, system variables, loops, calculations, ... are suitable.
- Using the "CYCLE_DRILL" cycle in section 3.9 for reference, create a new cycle that can be named "CYCLE_DR".
- Use newly created “CYCLE_DR” to make request in method mode in main program “.MPF”
- Convert all R parameters to a user variable to empty R variables from R1 and R99 in “User variable” for other applications or program steps.
- Replace parameters from R1 to R12 with user-supplied parameters.
- Replace parameter R20 to R30 with user-supplied parameters.
- Write new user parameters at the beginning of “CYCLE_DR” with user variable definition such as: DEF REAL _WAT, _SP1, ...

b. Technical drawings

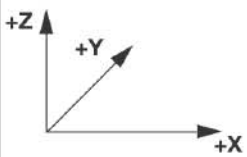

+0,4
+0,2
-0,2
-0,4
Rz 63
R4
ØR2
ØR1
Ø?
?
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Ø ouside and Ø inside variable
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE M PIPE HLL / PARAMETER
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name


+0,4
+0,2
-0,2
-0,4
Rz 63
10
Ø35
Ø50
Ø10
100
d
2
0
-7,5
-9,5
-40,5
-42,5
-50
-52
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Ø50 x 100
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE M PIPE HLL / PARAMETER
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name


+0,4
+0,2
-0,2
-0,4
Rz 63
10
Ø35
Ø50
Ø10
100
Y+
X+
material: EN AW-AlCu4PbMg
scale 1:1
blank dimensions: Ø50 x 100
Datum
Name
Gezeichnet
Kontrolliert
Norm
EXERCISE M PIPE HLL / PARAMETER
General tolerances
ISO 2768-m
programGuide Siemens Operate
1
A4
Status
Änderungen
Datum
Name

c. The following tools and technology data are required for programming:

Workpiece: EXERCISE M PIPE HLL/PARAMETRER	Blank material: EN AW-AlCu4PbMg	Machine configuration:
Drawing: 1(1)	Blank dimensions: ϕ50 x 100	Milling 3 axes

	Technological data		
Tool name in OFFSET	Spot_drill_12	Deburrer_8	SC_Drill_10
Tool Ø (mm)	12 x 90°	8 x 90°	10
Cutting speed (m/min)	80	105	80
Cutting depth a_p = max.	5	3	
Cutting material	HSS	SC	SC
Feed tooth fz / rev. (mm)	0,12	0,025	0,15
Number of teeth N		4	
			

UNIT 5: MACHINE INTEGRATED MEASURING OF WORKPIECES AND TOOLS WITH 3D TOUCH PROBES

1. **Introduction:** As the last unit of the module, the trainees will be familiar with measuring workpieces and tools with 3D touch probes integrated in CNC machines.
2. **Objective:** By the end of this training unit, the trainees will be able to

- Understand the structure and operating principle of measuring devices used when measuring workpieces and products on CNC milling machines;
- Repair the instruments for high accuracy before measuring;
- Measure the dimensions of various types of part shapes in manual and automatic modes on CNC milling machines;
- Measure the position of common workpiece standards before machining;
- Practice well in the work of redefining the standard and accuracy of CNC milling machines;
- Export the measurement result report as text;
- Use Mechanical Trade and Metal Handbook.

3. **Content:**

3.1. Measuring instruments for measuring workpieces

3.1.1. Introduction to workpiece measurement with SINUMERIK

Measuring the workpiece is used to define the position and alignment of the workpiece in the machine coordinate system.

Workpiece measurement can take place in two modes:

1. JOG – Setup mode	2. AUTOMATIC – Process measuring

Figure 5.1: Measure dimensions of product on CNC milling machines

Measuring cycles are used in various measuring tasks. The measuring cycle for the measuring task is configured by setting various parameters in the input screen.

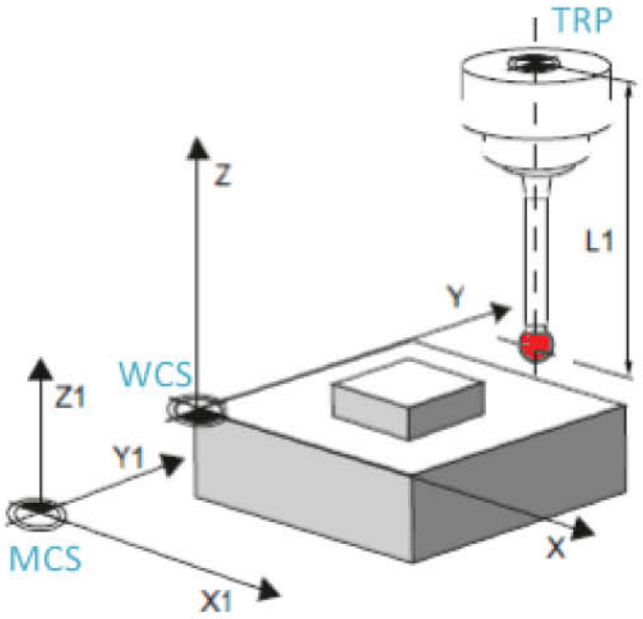
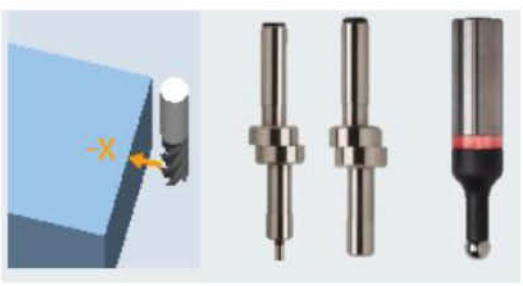


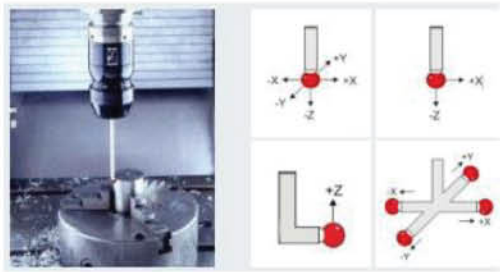
Figure 5.2: Workpiece coordinate system and machine coordinate system when measuring

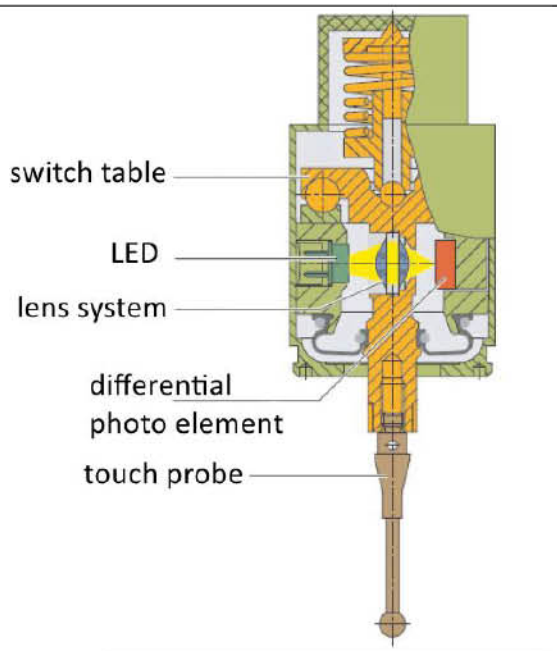
Automated and repeated execution of workpiece programs requires the definition of reference points:

Machine zero point	Workpiece zero point
• Located in the MCS machine coordinate system and defined by the machine manufacturer • Cannot be changed from end-user • Reference point for internal measuring systems in the machine • Limits the machine's working area	• The basis of the WCS workpiece coordinate system Zero point of the workpiece • Can be freely selected by the end-user • Is often positioned on a corner or in the centre of the • Workpiece, a hole, a pin (to suit workpiece machining)

Measuring instruments for measuring workpieces

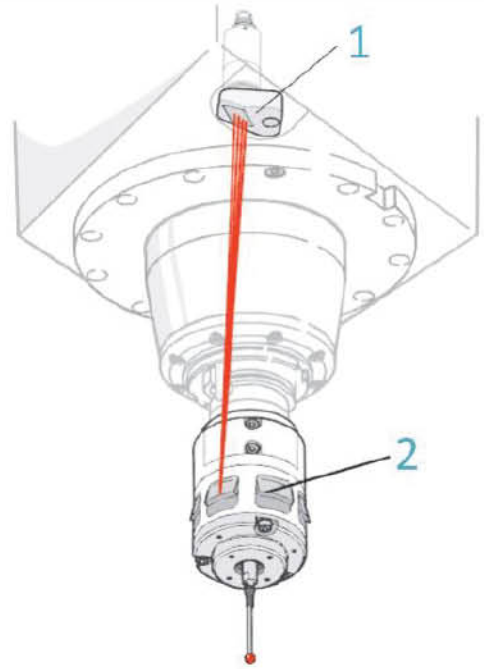
Manual contacting / scratching	
	• The operator manually moves an edge finder or a rotating tool to the workpiece • The operator determines the current position on the basis of visual signals • The operator confirms the touching via a softkey • The workpiece zero point is automatically saved by the CNC

Manual probe	
	• The measuring instrument is located in the spindle, which has its probe length and diameter stored in the tool offset • The operator determines the current position of the spindle centre above an edge using a scale • The operator confirms the touching • Via a softkey
Switching probes	
	• The operator positions an previously • Calibrated probe in the vicinity of the desired measuring object • The cycle is selected and the • Measurement is started via NC- Start • The desired position is automatically determined and saved by the CNC
	• The machine must be prepared on both, the hardware and software side for use of switch probes



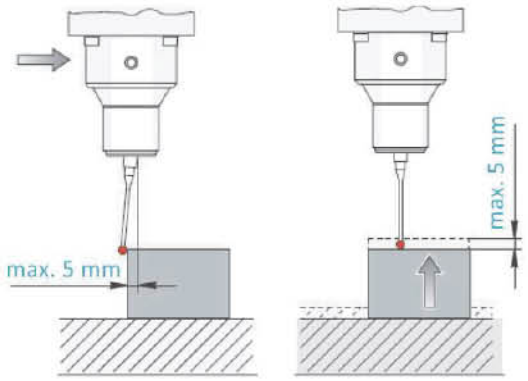
Functional principle of the sensor:

- The luminous flux emitted by an LED is bundled by a lens system and falls as a point of light on a differential photo element.
- When the stylus is deflected the differential photo element generates a switching signal.
- Due to the non-contact optical switch, the sensor works without wear.



Functional principle of the sensor:

- The probe is on the perimeter even with transmitter diodes (1) fitted.
- The transmitter diodes guarantee one in every position seamless signal transmission to the infrared receiver (2), which is in the spindle head lies.



Deflection of the contact element:

- The maximum permissible deflection of the stylus is 5 mm in each direction.
- The machine movement must be stopped within this path to avoid damage to the touch probe.
- Probing speed:
- Signal runtimes of the CNC influence the probing reproducibility of the touch probe.
- In addition to the signal propagation time, the permissible deflection must be taken into account for the maximum probing speed.
- The mechanically permissible probing speed is specified in the technical parameters.

3.2. Calibration of measuring instruments for measuring accuracy

Measuring accuracy - calibration, the key factor in accurate measurements

The measuring probe must be calibrated:

- When commissioning (first setup at the machine)
- When changing the stylus
- In the event of major temperature fluctuations
- Before 3D quick set is used

Notice:

- It should be recalibrated regularly
- When calibrating and measuring, it is absolutely necessary to use always the same feed rate
- This relates in particular to the feed override

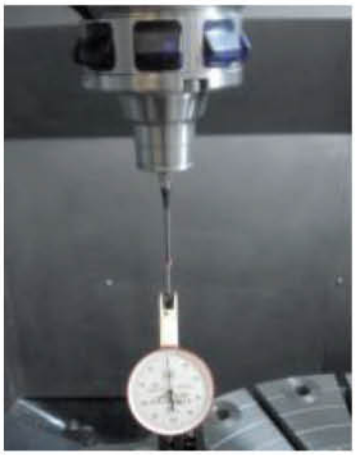
Calibration = calibration of the probe before use:

- Set the radial eccentricity of the probe, see manufacturer's specifications
- Radius alignment with calibration
- Length alignment with internal or external length measuring device
- Determination of individual switching points (trigger points) in reference to the spindle centre of each direction of the axis (e.g. +X, -X, +Y, -Y, -Z)

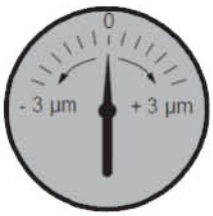
Required measuring equipment

- Calibration ring
- Calibration mandrel, if present
- Dial gauge max. 2µm
- Gauge block

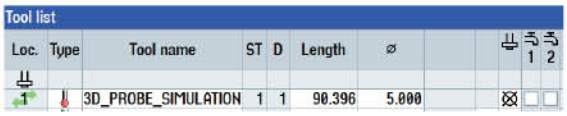
Method of calibration



- Check the concentricity of the stylus tip with a dial indicator
- Deviation if possible: < 0,003 mm



Define the measuring probe in the tool list



- Define the probe as type 710 or 712 and an approximate length (+/- 5mm)
- Specify the diameter
- Lock the spindle with the “Select” key to avoid the start of the spindle rotation (M3 / M4)

Adjust the length and radius of the probe

- With the “Calibrate probe” measuring function, a probe can be attached to a surface or be calibrated in a bore.

Define the reference point (ZO):

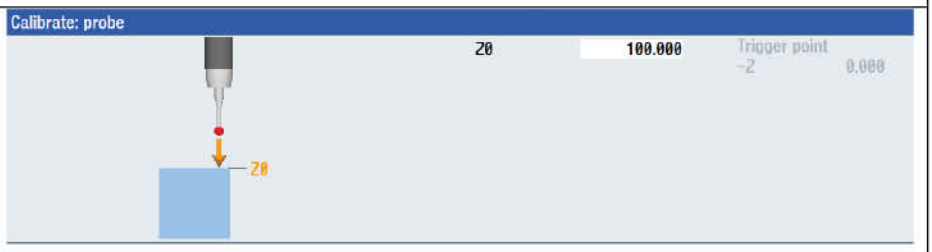
- Set the reference point in the Z-axis to adjust the length of the probe, e.g. with

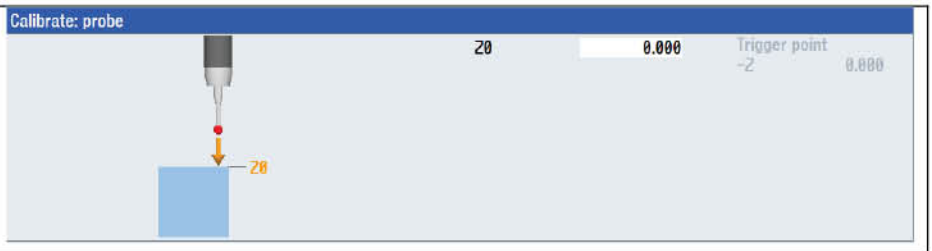
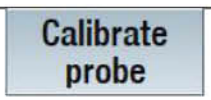
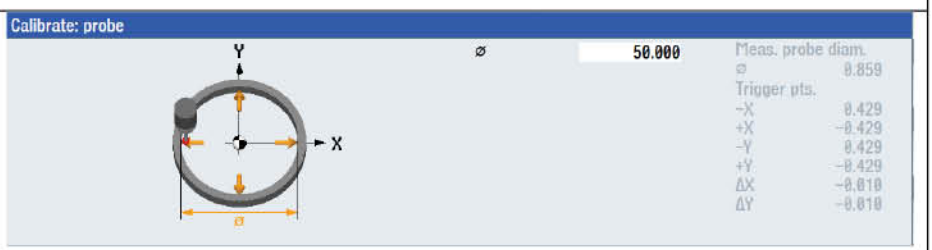
the calibration mandrel and gauge block on the machine table surface.

- With HSK spindles, it is also possible to open directly (without a mandrel) the spindle nose can be measured.

Adjust the length of the probe:

- Replace the measuring probe in the spindle
- Probe in "JOG" mode in Z about 10mm over the pre-position the gauge block or the machine table surface

Meas. workp.	Select the softkeys in the order shown:
Calibrate probe	
Length	
Gauge block 100	

Directly on machine table	
	Enter the reference point and start the cycle: The length adjustment is now started. The switching point of the workpiece probe is calculated in the measuring direction of the active working plane and entered in the tool list.
	Adjust the radius/diameter of the probe: • Replace the measuring probe in the spindle. • Probe in “JOG” mode approximately in the middle position of the calibration ring.
	Select the softkeys in the order shown:
	
	
	Enter the diameter of the calibration ring and start the cycle: The radius adjustment is now started. First the exact center of the calibration ring determined. Then the 4 switching points are on the inner wall of the calibration ring. This sequence is repeated a second time with a 180 ° reversal carried out around the positional deviation from the tool reference axis to be determined in X and Y.

Measurements conditions with touch probe

Measurements must be made under the same conditions as for the calibration:

- Calibrations and measurements should be done in the same planes (G17, G18, G19).
- The measuring feed must correspond to the calibration feed.
- At the start of the measurement, the probe should be located approximately in the centre of the calibration ring (by eye).
- In the best case, the probe will always be calibrated in the spindle (clamping tolerance).

The following lead to measuring inaccuracies:

- Geometric errors in the machine axes
- Radial eccentricity of the spindle
- A faulty calibration process (e.g. feedrate must be 100%)
- A faulty or dirty probe sphere
- A dirty surface of the measuring object (e.g. due to chips)
- Temperature fluctuations during the measurement

The obtainable measuring precision that can be obtained is dependent on the following factors:

- Accuracy of the machine
- Measuring precision of the calibrated probe

3.3. Measuring in the Setup mode

Modes information

When measuring workpieces, a measuring instrument similar to a tool is moved to the clamped workpiece.

These measuring tasks can take place in two modes:

JOG – setup mode	AUTOMATIC (in-process measurement)
• To prepare the machine for processing • The determination of the position and alignment of a workpiece • Definition of the zero point position of the workpiece in the machine space • Partially automatic measurement, as interactive operation in the set-up mode of the machine	• Quality assurance in the machining process • For the correction of work offsets • Correction of thermally influenced changes to the machine • The creation and provision of measurement reports • Fully automatic measurement by calling measuring cycles in the CNC machining program

Figure 5.3: Setting of manual measurement mode and automatic measurement mode

Setup in the JOG mode

General information - semi-automatic measuring:

- Manual measurement is used to prepare the machine for processing.
- Manual measurement is used to determine unknown workpiece or tool geometries.
- The measurement takes place as interactive operation in the manual operating area of the machine.
- The measuring functions support measuring with a switching measuring probe (e.g. electric infrared 3D measuring probe) or non-switching measuring probes (e.g. manual 3D probe).

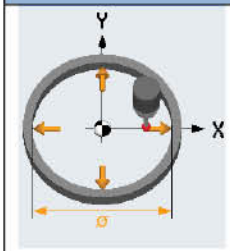
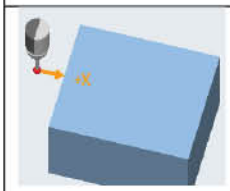
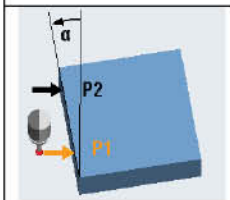
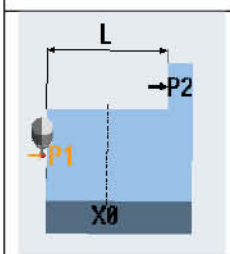
Workpiece measurement:

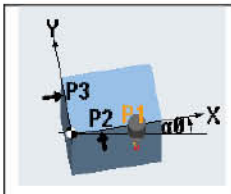
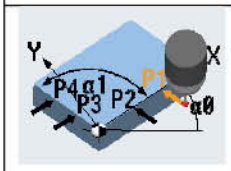
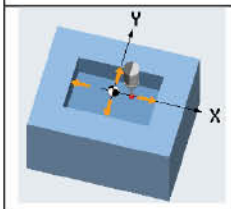
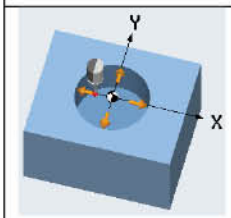
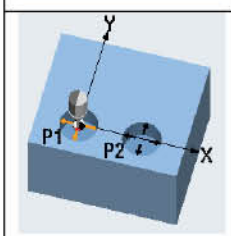
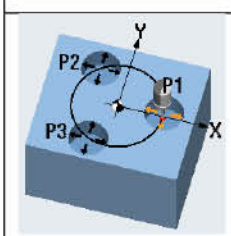
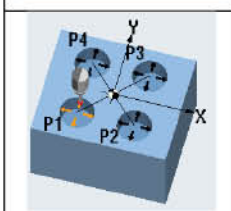
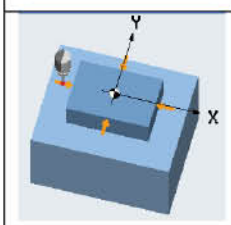
- When measuring workpieces with switching probes, you can:
  + Calibrate (adjust) the workpiece probe.
  + The contour elements (edge, corner, hole, spigot, rectangle, plane) and then measure the workpiece by defining and setting it align a zero offset correction.
- When measuring workpieces with non-switching measuring probes (manual 3D probe) you can:
  + The contour elements (edge, corner, hole, spigot, rectangle, plane) and then measure the workpiece by defining and setting it align a zero offset correction.

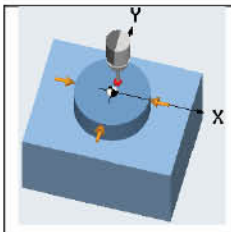
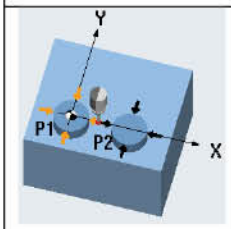
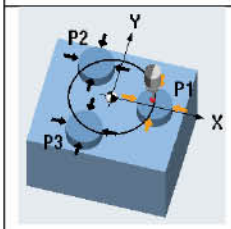
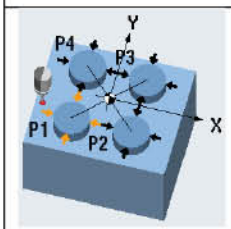
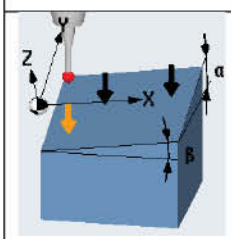
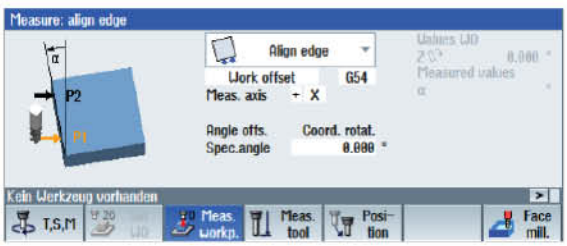
+ Exception! 2,3,4 holes measuring is not possible.

Non-switching 3D touch probe	Switching infrared 3D touch probe

Setup in the JOG mode - overview of the measurement functions

T,S,M	Set WO	Meas. workp.	Meas. tool	Posi-tion	Face mill.	Swi vel
	Calibrate probe length and diameter	for switching probes
	Measure edge	for manual measuring instruments for switching probes
	Align edge	for manual measuring instruments for switching probes
	Spacing 2 edges	for manual measuring instruments for switching probes

	Rectangular corner	for manual measuring instruments for switching probes
	Any corner	for manual measuring instruments for switching probes
	Rectangular pocket	for manual measuring instruments for switching probes
	1 hole	for manual measuring instruments for switching probes
	2 holes	for switching probes
	3 holes	for switching probes
	4 holes	for switching probes
	Rectangular spigot	for manual measuring instruments for switching probes

	1 circular spigot	for manual measuring instruments for switching probes
	2 circular spigots	for switching probes
	3 circular spigots	for switching probes
	4 circular spigots	for switching probes
	Align plane	for switching probes
Alignment on the workpiece edge 	• Saving the axis position in the rotary part of the zero point FRAME. • The twisting of the workpiece is compensated by rotating the coordinates of the workpiece coordinate system.	

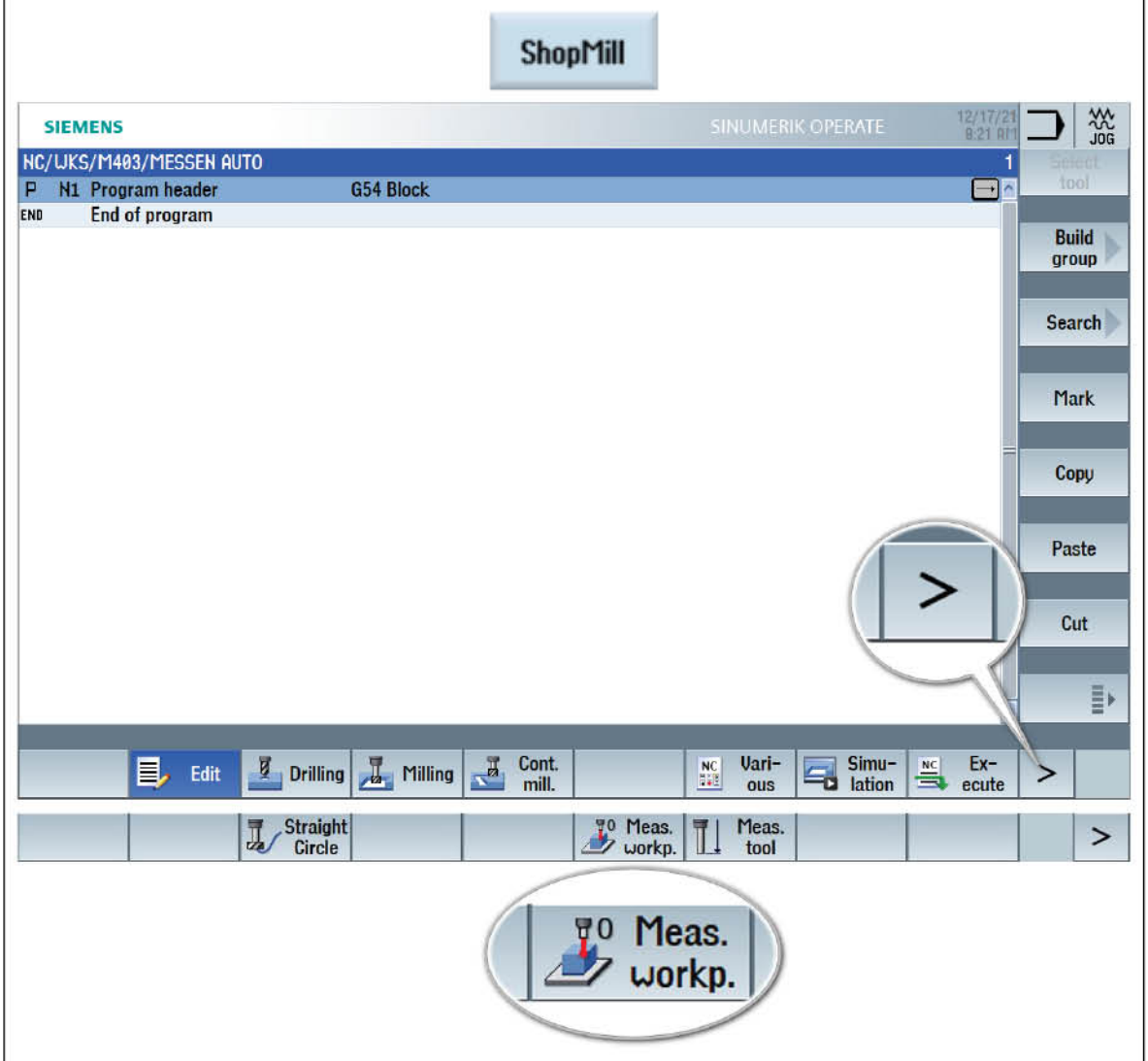
3.4. Measuring in Automatic mode

Workpiece measurement in AUTOMATIC mode

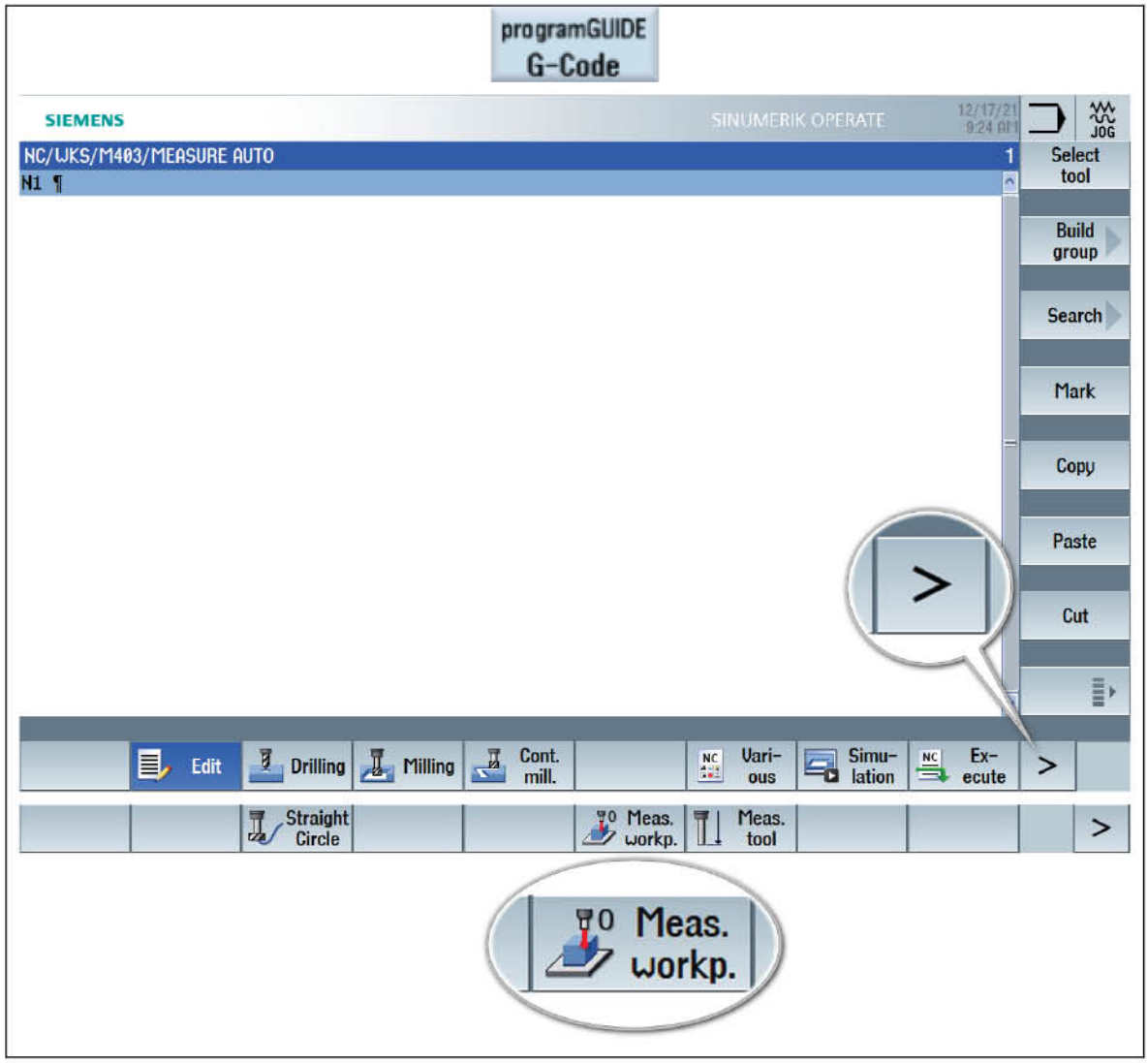
Mode information of measuring:

- Automatic measurement is used to check the conformity of dimensional details of certain workpieces are used.
- Automatic measurement is used to correct known workpiece and tool geometries used.
- The measurement is carried out by calling up a measurement cycle in the machining program.

Select machining program in ShopMill



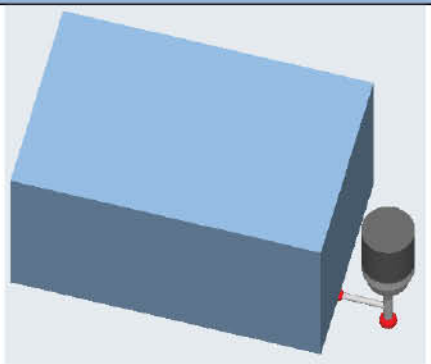
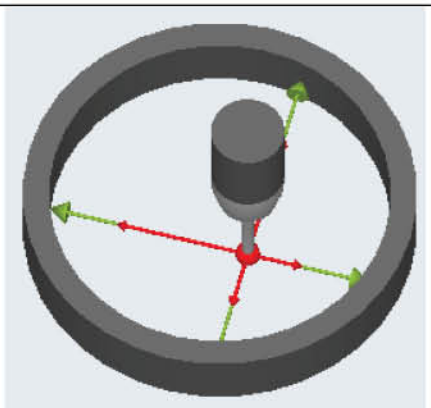
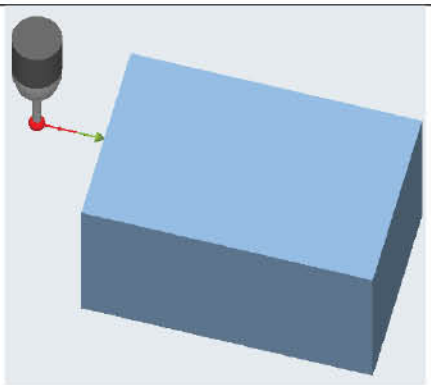
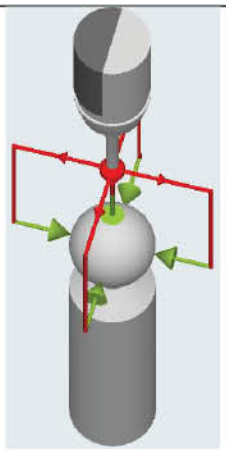
Select machining program in programGuide

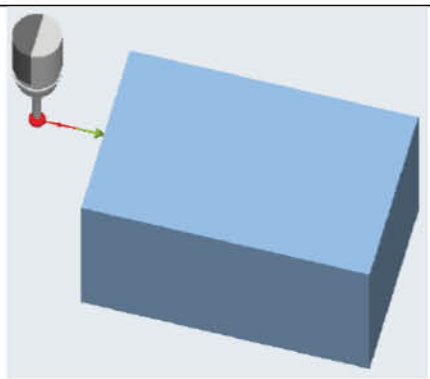
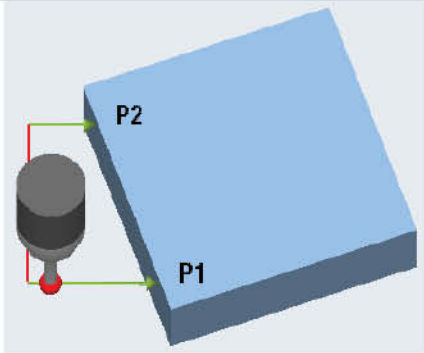
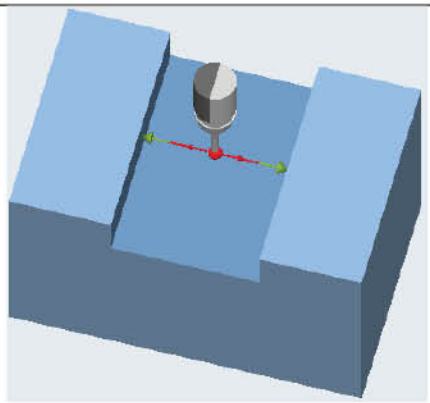
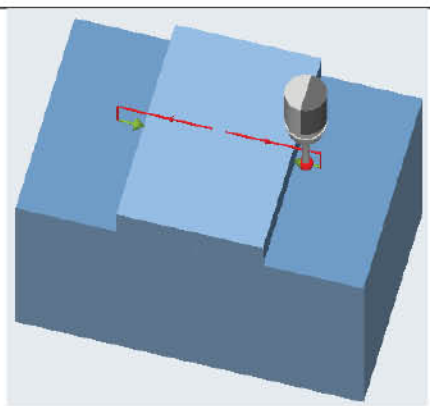


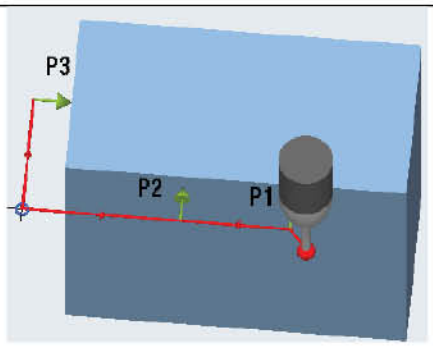
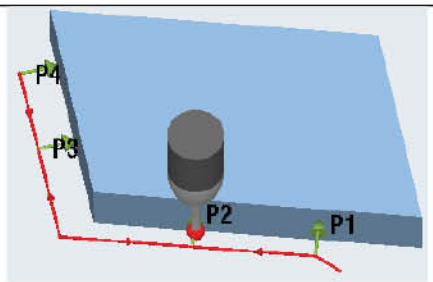
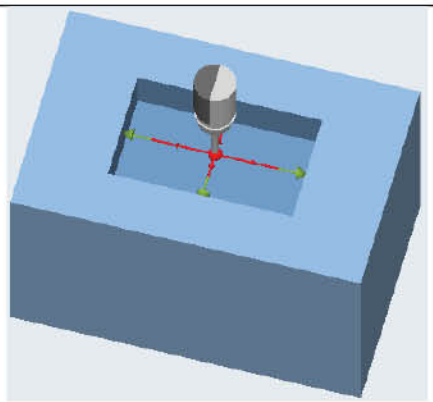
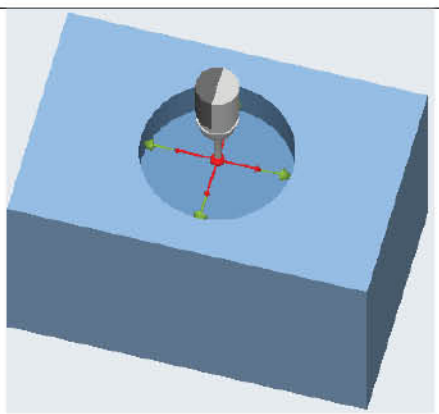
AUTOMATIC mode - overview of the measurement functions

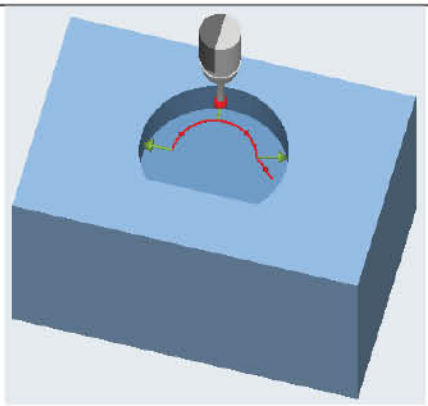
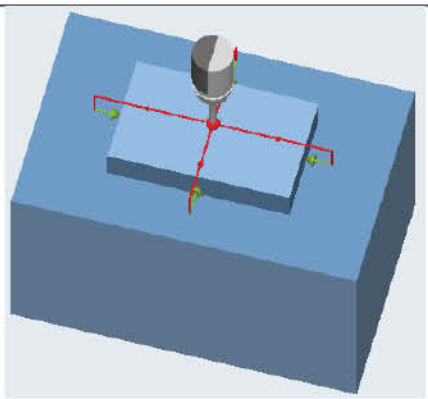
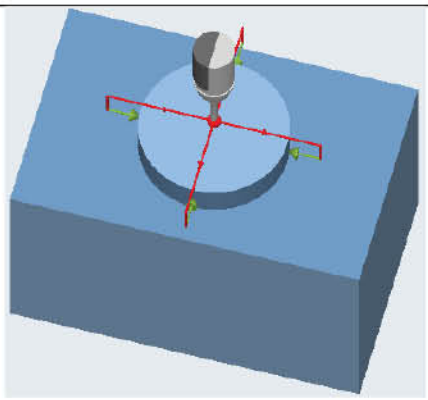
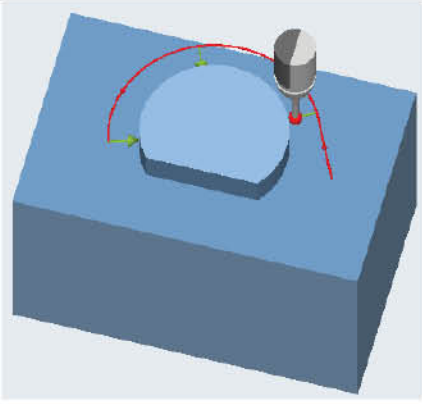
The measuring cycle support offers the following functionality:

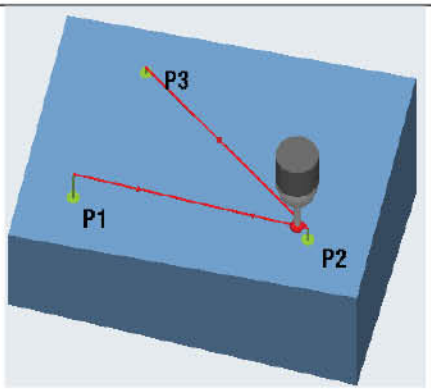
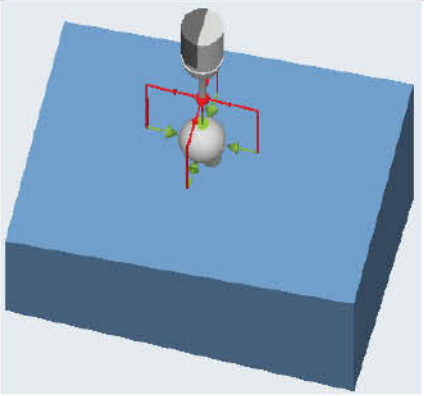
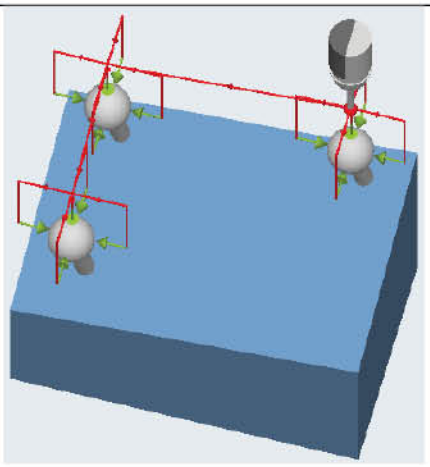
- Selection of measuring cycles via softkeys.
- Input masks for assigning parameters with animated help images.
- Recompilation of programmed cycles into the input mask.

Overview of the measurement functions in AUTOMATIC		
Calibrate probe		length on edge CYCLE976
Calibrate probe		radius on ring CYCLE976
Calibrate probe		radius on edge CYCLE976
Calibrate probe		calibrate probe on sphere CYCLE976

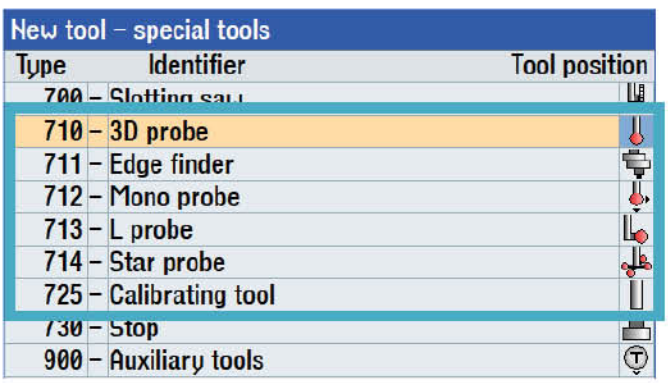
Edge distance		edge CYCLE978
Edge distance		align edge CYCLE998
Edge distance		slot (groove) CYCLE977
Edge distance		web CYCLE977

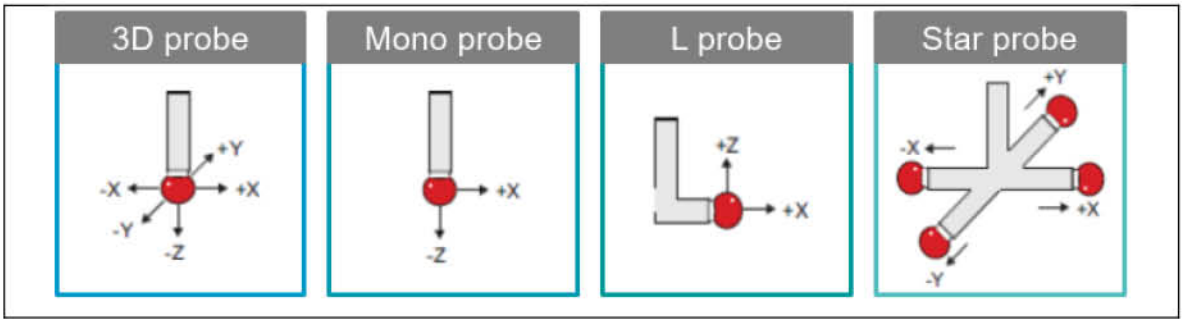
Corner		rectangular corner CYCLE961
Corner		any corner CYCLE961
Hole		Rectangular pocket CYCLE977
Hole		1 hole CYCLE977

Hole		inside circle segment CYCLE979
Spigot		rectangular spigot CYCLE977
Spigot		circular spigot CYCLE977
Spigot		outside circle spigot CYCLE979

3D		align plane CYCLE998
3D		ball CYCLE997
3D		3 balls CYCLE997

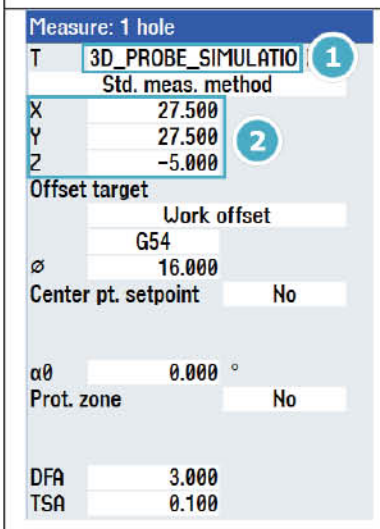
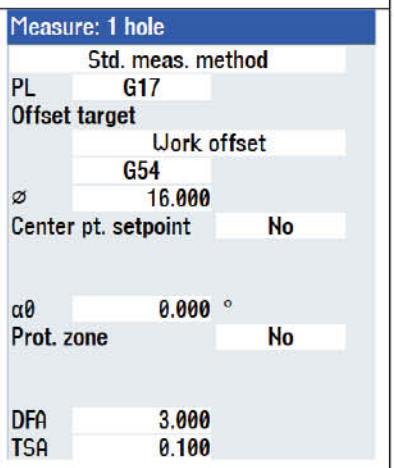
Note: All measuring variants are only available for switching touch probes.





Measurement variants - control-internal correction of measured values

Workpiece measurement **With the "SELECT" key it is possible to choose between different correction targets:**		
1 - with correction of the work offset	2 - with correction of the tool geometry	3 - measuring only
• Checking the current clam-ping situation • Correction of the workpiece zero point offset resulting from the manual or auto-matic feeding of workpieces	• Ensuring the adherence to • Workpiece tolerances • Continuous correction of the tool wear or thermal expan-sions that occur during machining	• Measurement of workpiece geometries for individual reuse • Display of the workpiece geometry
Summary: The measurement result is used to correct the zero offset.	Summary: The measurement result is used to correct the tool data in the tool offset.	Summary: Only measure - no correction of the zero offset.
Measure: 1 hole T D 1 Std. meas. method X 0.000 Y 0.000 Z 0.000 Offset target Work offset G54 ⌀ 10.000 Center pt. setpoint No α0 0.000 ° Prot. zone No DFA 1.000 TSA 1.000	Measure: 1 hole T D 1 Std. meas. method X 0.000 Y 0.000 Z 0.000 Offset target Tool offset TR CUTTER 6 D 1 ⌀ 16.000 α0 0.000 ° Prot. zone No DFA 3.000 TSA 1.000 Dim. toler. Yes TUL 0.100 TLL -0.100	Measure: 1 hole T D 1 Std. meas. method X 0.000 Y 0.000 Z 0.000 Offset target Meas. only ⌀ 16.000 α0 0.000 ° Prot. zone No DFA 3.000 TSA 1.000

Input differences between ShopMill and programGUIDE		
ShopMill measure 1 hole		programGUIDE measure 1 hole
	1. Call up the touch probe. 2. The one to be approached measuring position in X, Y and Z coordinate.	

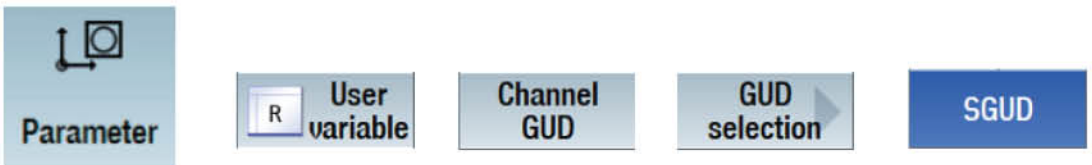
Workpiece measurement in AUTOMATIC mode

Result parameter

- Result parameters are parameters in which the measurement results of the measurement cycles be filed.

Parameter	Typ	Meanings
_OVR[ ]	REAL	Result parameter - real number
_OVI[ ]	INTEGER	Result parameter - Integer

- The result parameters of the measuring cycles are saved in the channel-specific user variables, which can be displayed in the "Parameters" operating area.



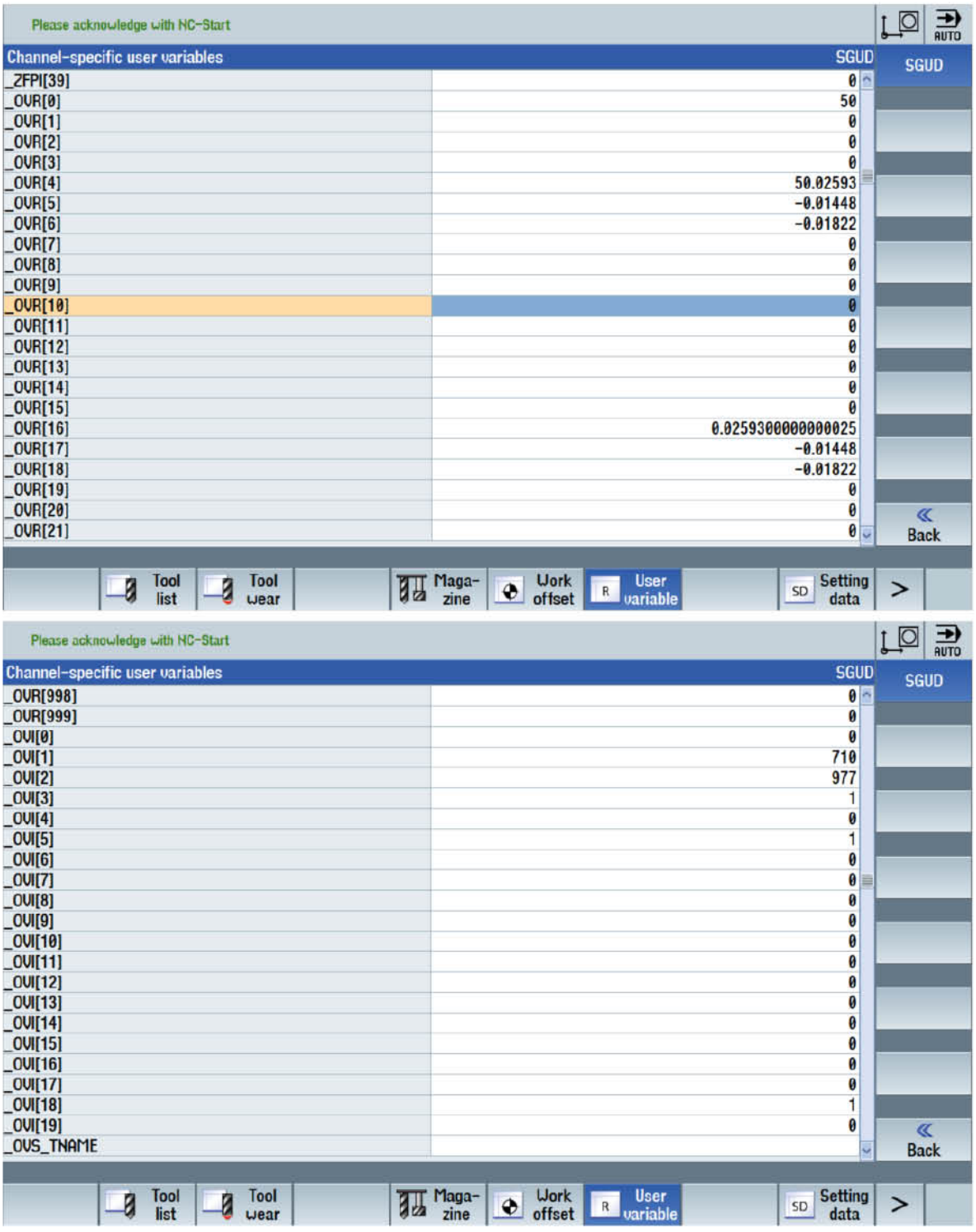
Select in the order

Example for list of the result parameters

The measuring variant "Hole" provides the following result parameters:

Parameter	description	unit
_OVR [0]	Hole diameter setpoint	mm
_OVR [1]	Hole center point setpoint in the 1st axis of the plane	
_OVR [2]	Hole center point setpoint in the 2nd axis of the plane	
_OVR [4]	Hole diameter actual value	
_OVR [5]	Hole center point actual value in the 1st axis of the plane	
_OVR [6]	Hole center point actual value in the 2nd axis of the plane	
_OVR [16]	Hole diameter difference	
_OVR [17]	Hole center point difference in the 1st axis of the plane	
_OVR [18]	Hole center point difference in the 2nd axis of the plane	
_OVI[0]	D number or WO number	
_OVI[2]	Measuring cycle number	
_OVI[3]	Measuring variant	
_OVS_TNAME	Tool name	

Example for result parameter “Hole” with Ø50 and measuring cycle “1 Hole”:



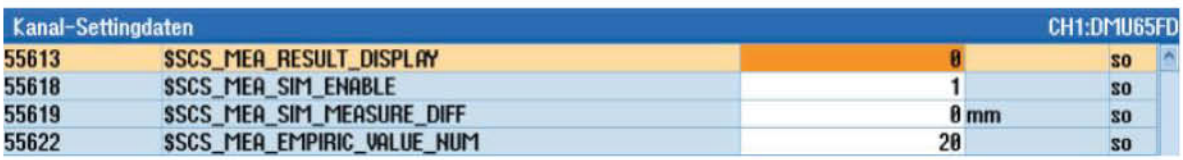
3.5. Measuring results through logging

Display of measurement result images

During the course of a measuring cycle, measurement result images can be displayed automatically. In the channel specific „SD 55613 $ SCS_MEA_RESULT_DISPLAY“ you can choose between the following image displays of the measurement result:

- “= 0” no display of the measurement result images (standard setting)
- “= 1” display of the measurement result image for 8 seconds
- “= 3” The measuring cycle stops at "M0", the display of the measurement result image is static. The measurement cycle is continued with cycle start, the measurement result screen is deselected
- “= 4” The measurement result image is only displayed for cycle alarms 61303, 61304, 61305, 61306. The measurement cycle is continued with cycle start, the measurement result image is deselected
- At the beginning of a program the display is written as follows:

  $SCS_MEA_RESULT_DISPLAY = 0 ; 0,1,3,4
- The setting can also be made manually via channel setting data:



Kanal-Settingdaten				CH1:DMU65FD
55613	$SCS_MEA_RESULT_DISPLAY	0		so
55618	$SCS_MEA_SIM_ENABLE	1		so
55619	$SCS_MEA_SIM_MEASURE_DIFF	0	mm	so
55622	$SCS_MEA_EMPIRIC_VALUE_NUM	20		so

Measuring example with program Guide

Description:

A first program Guide exercise is to be programmed with the help of spigot and pocket milling cycle calls. This module describes measuring in “AUTOMATIC” mode with the SINUMERIK measuring cycles. Measuring of the workpiece is shown with an automatic touch probe and following tool correction.

Objective:

You learn the fundamentals about programming and measuring in "AUTOMATIC" mode using SINUMERIK cycles.

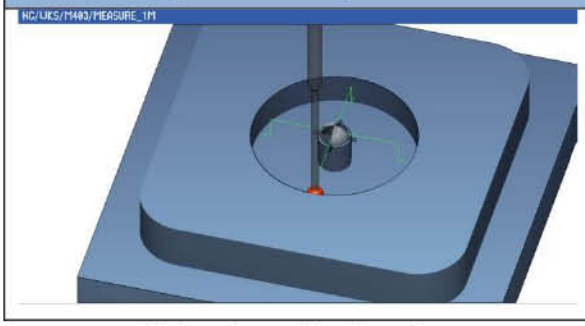
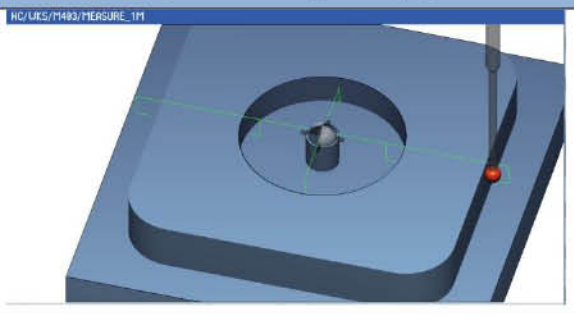
You can then measure your workpieces simply and precisely on a CNC milling machine with the

SINUMERIK measuring cycles.

The workflow sequences are to programmed:

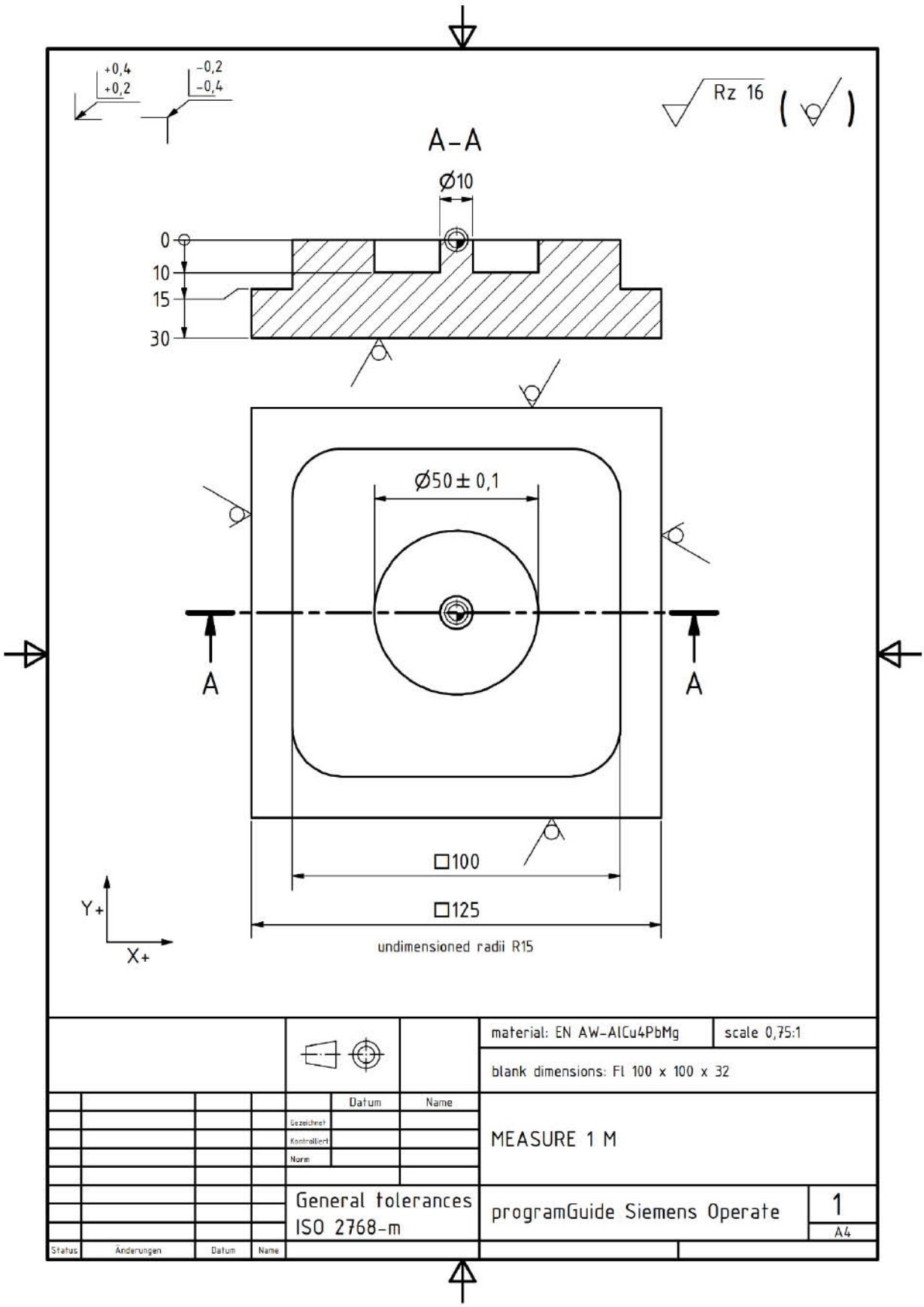
- Program the image displays of the measurement result
- Program header
- Programming the commands for the face machining in G17
- Facing the workpiece height with 30 mm with “Cutter 32”
- Rectangular spigot milling 100 mm x 100 mm with radii 15 mm with “Cutter 10”
- Program stop with “M0” for workpiece cleaning

- Circular pocket Ø50 mm with a pin Ø10 mm as spigot in the pocket
- Measuring the circular pocket Ø50 mm with program cycle and automatically tool correction for “Cutter 10”
- Display measure result CYCLE977 for “Cutter 10”
- Measuring the rectangular spigot 100 mm with program cycle and automatically tool correction for “Cutter 32”
- Display measure result CYCLE977 for “Cutter 32”

Measuring the circular pocket Ø50 mm	Measuring the rectangular spigot 100 mm
	

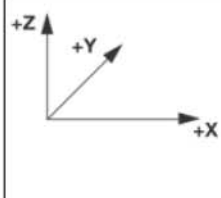
- Deburring with chamfers:
  + Circular pocket Ø50 mm
  + Circular pin Ø10 mm
  + Rectangular spigot 100 mm x 100 mm
  + Blank rectangle 125 mm x 125 mm
- Program end

Technical drawing measure program:



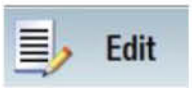
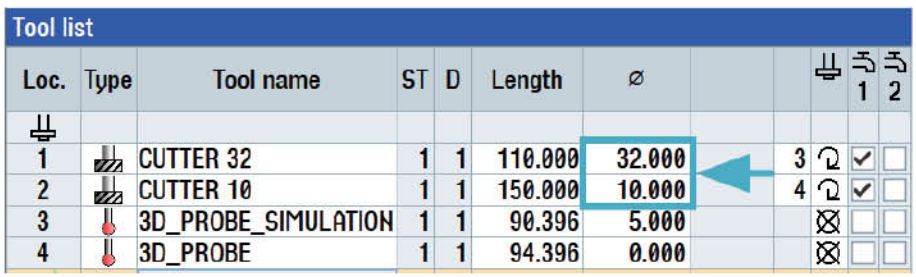
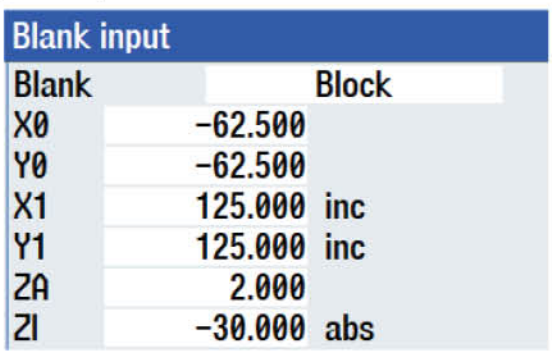
programGuide Siemens Operate VIETNAM 2021/2022	
Tool shape file	CNC Milling

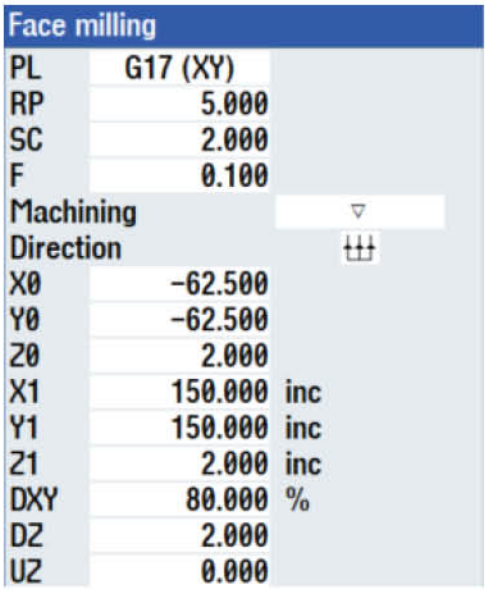
work piece: MEASURE 1 M DIN	blank material: EN AW-AlCu4PbMg	machine configuration:
drawing: 1(1)	blank dimensions: FI 125 x 125 x 32	Milling 3 axes 7 Milling 5 axes (X Y Z A C)

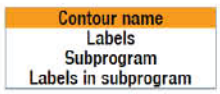
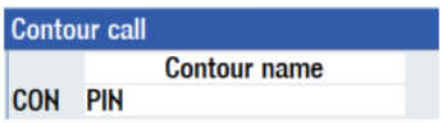
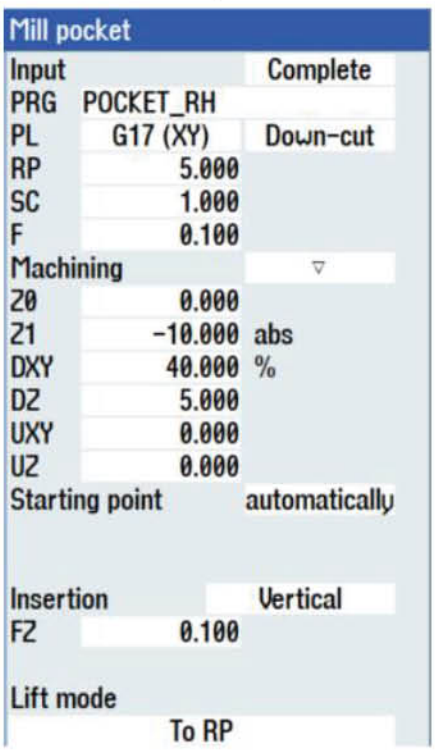
		Technological data						
Tool - Nr.	Cutter 32	Cutter 10	Deburrer 8	3D Touch Probe Simulation	3D Touch Probe			
Tool Ø (mm)	32	10	4	5	0			
Cutting speed (m/min)	150 / 175	150 / 175	105	-	-			
Cutting depth a_p = max.	8	5	4	-	-			
Cutting material	SC	SC	SC	-	-			
Feed tooth fz / rev. (mm)	0,1	0,05	0,025	300 mm / min	300 mm / min			
Number of teeth N	4	4	4					
								
	✓	✓						

A new workpiece directory “MEASURE_DIN” should be created and opened in the editor. Create a new G code program in programGuide, like described in the previous modules.

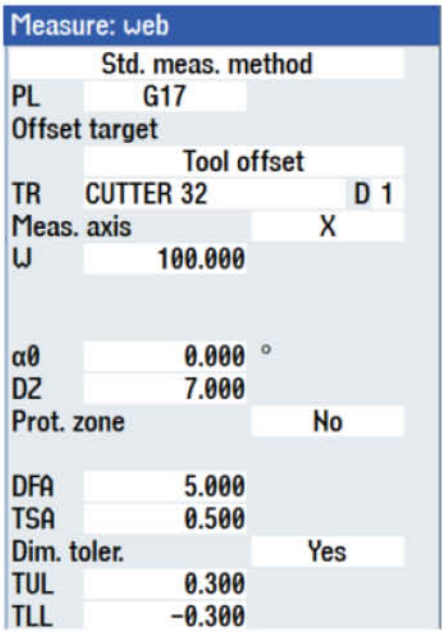
Give the program a name, for example “MEASURE_1_M_DIN”:

1.	Create a new programGuide directory and main program called “MEASURE_1_M_DIN”.
2. 	The “Edit” operation mode opens automatically. Note: To shorten the program and to focus the programming on the new measuring cycle programming, all milling cycles are programmed in roughing mode without allowances.
3.	Before started to program, press the function key “OFFSET” to open the “Tool list” for checking the diameter of the milling tools “Cutter 10” and “CUTTER 32” and as well the “3D Touch Probe” and “3D Touch Probe Simulation”: 
4.	Start the programming with the channel setting data for the image to display the measurement results: **N1 $SCS_MEA_RESULT_DISPLAY=3 ;0,1,3,4**
5.	Insert now a blank for the simulation into the program. Enter following values/parameters for the blank: 
6.	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N2 WORKPIECE(,"",,"BOX",64,2,-30,-80,-62.5,-62.5,125,125)**
7.	Continue the program by inserting the following G code commands: **N3 G17 G64 G71 G90**
8.	Insert the following technology commands into the program:

	N4 T="CUTTER 32" **N5 M6** **N6 G95 S2387 FZ=0.1 M3 M8** **N7 G0 Z10**
9. 	Enter the following values into the input mask for “FACE MILLING”:  Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N8 CYCLE61(5,2,2,2,-62.5,-62.5,150,150,2,80,0,0.1,21,0,1,11)**
10.	Insert the following technology commands into the program: **N11 T="CUTTER 10"** **N12 M6** **N13 G17 G40 G60 G71 G90** **N14 G95 S4775 FZ=0.05 M3 M8** **N15 G0 Z10**
11. 	Program now the call up of the contour “CYCLE62”. Note: The contour itself is programmed at the end of the program. For this, press the HSK 1.4 “Cont. mill.” to open the technology “Contour milling”. Press now the VSK 1 “Contour”. The vertical softkey bar with functions for creating or calling up a contour opens. Press the VSK 2 “Contour call” to open the input mask for the contour call up.

12.  	Insert the following “Contour name” (graphic editor) name into the input mask for the circular pocket: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N16 CYCLE62("CIRCLE",1,,)** Insert the following “Contour name” (graphic editor) name into the input mask for the circular pin in the pocket:  Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N17 CYCLE62("PIN",1,,)**
13.	Enter the following values into the input mask for “MILL POCKET”: 
14.	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N18 CYCLE63("POCKET_RH",1,5,0,1,-**

Accept	**10,0.1,0.1,40,5,0,0,0,0,0,2,2,15,1,2,"",1,,0,101,110)** **N19 G0 Z100**
15.	Enter the following auxiliary function before called the “3D Touch Probe”: **N20 M5** **N21 M0** Note: The program stop with “M0” is programmed to clean the workpiece before calling the measuring cycles.
16.	Insert the following technology commands into the program: **N22 T="3D_PROBE" ;only use for AUTOMATIC** N23 ;T="3D_PROBE_SIMULATION" **;only use for SIMULATION** **N24 M6** **N25 MSG(“TOOL CORRECTION BY MEASURING”)** **N26 G0 Z10** **N27 G0 X0 Y0** **N28 G0 Z2** **N29 G1 G94 F300**
17. Tool offset Meas. only Work offset Tool offset Accept	Enter the following values into the input mask for “MEASURE 1 HOLE”: Measure: 1 hole Std. meas. method PL G17 Offset target Tool offset TR CUTTER 10 D 1 ⌀ 50.000 α0 0.000 ° Prot. zone Yes ⌀S 10.000 DZ 7.000 DFA 3.000 TSA 0.500 Dim. toler. Yes TUL 0.100 TLL -0.100 Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N30 CYCLE977(1201,,4000001,1,50,,,3,0.5,0,7,10,,,1,"CUTTER 10",,0,1.01,0.1,-0.1,0.34,1,0,,1,1)**

18.	Insert the following technology commands into the program: **N31 G0 Z2** **N32 G0 X0 Y0**
19.	Enter the following values into the input mask for “MEASURE WEB”:  Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N33 CYCLE977(204,,4000001,1,100,,,5,0.5,0,7,1,,1,1,"CUTTER 32",,0,1.01,0.3,-0.3,0.34,1,0,,1,1)** N34 ;CYCLE978(0,,,1,-10,5,5,3,2,1,"",,0,1.01,1.01,-1.01,,,,,1,1) **N35 G0 Z100**

Explanations to some measuring cycle parameters on the example cycle parameter” MEASURE 1 HOLE”:

_TSA - Trust area

If this limit is reached, it can be concluded that:

- an incorrect target position specification or
- an impermissible deviation from the target position, e.g. dirt
- or the tool is worn out

Notice: If the _TSA is exceeded, no tool correction takes place.

_TUL - Upper tolerance limit

If _TUL is exceeded, a message “Undersize” is displayed.

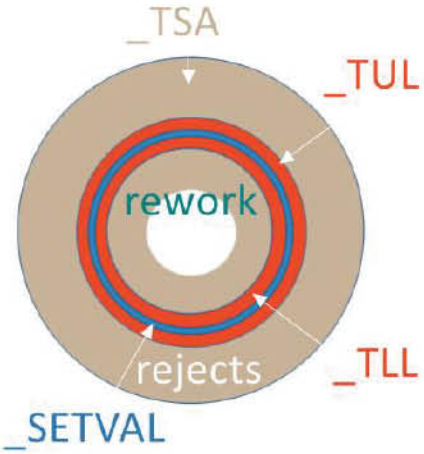
If the value is below _TSA, a tool offset takes place.

_TLL - Lower tolerance limit

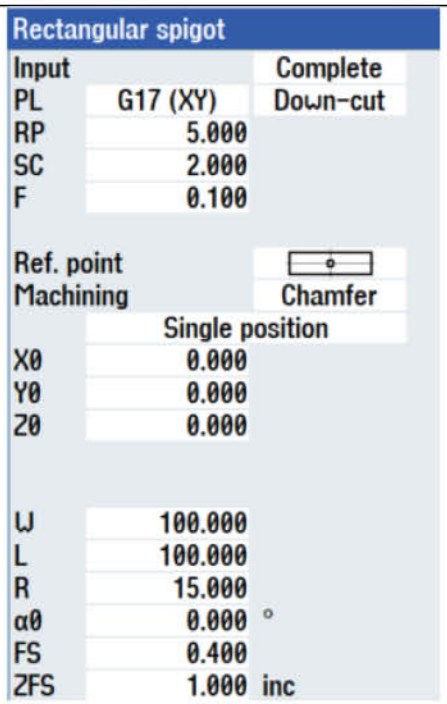
If _TLL is exceeded, a “Allowance” message is displayed.

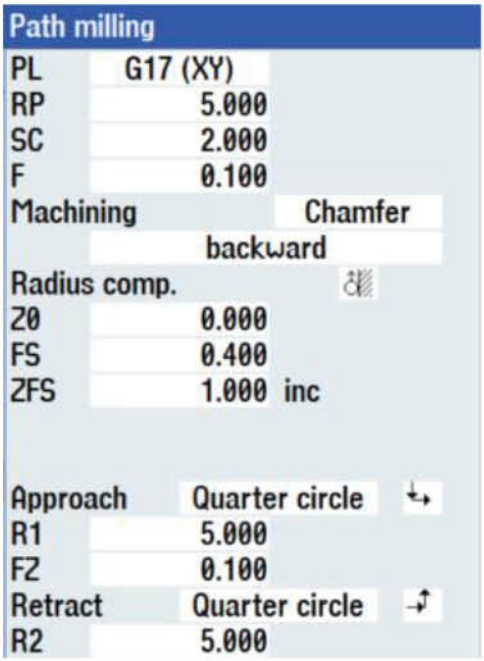
If the value is below _TSA, a tool offset takes place.

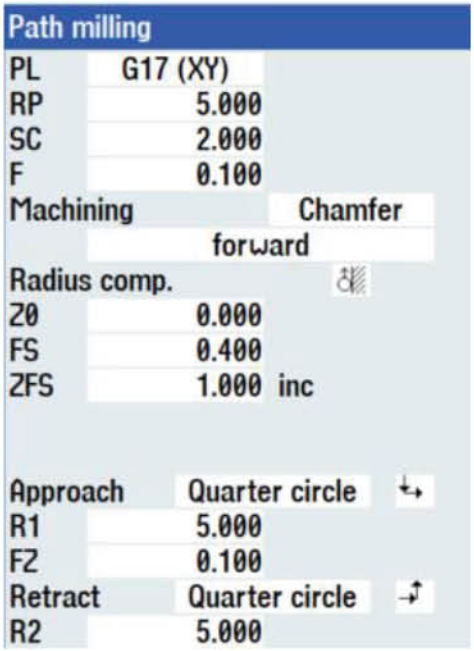
_SETVAL – Set point hole diameter

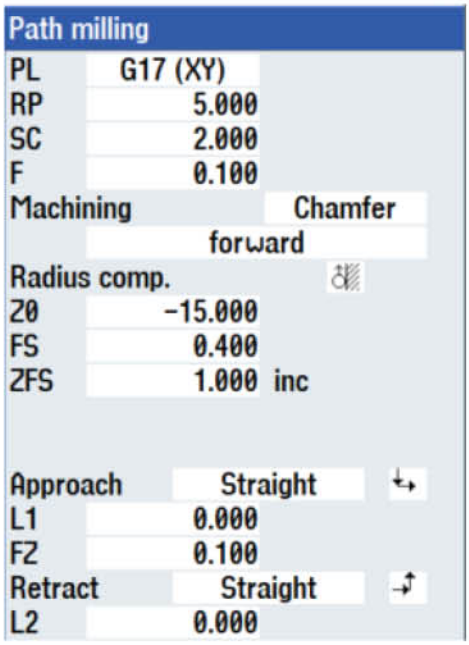


20.	Insert the following technology commands into the program: **N36 T="DEBURRER 8"** **N37 M6** **N38 S4178 G95 F0.1 M3 M8**
21.	Enter the following values into the input mask for “RECTANGULAR SPIGOT” to mill the deburring chamfer 0,4x45° on the spigot 100 mm x 100mm:

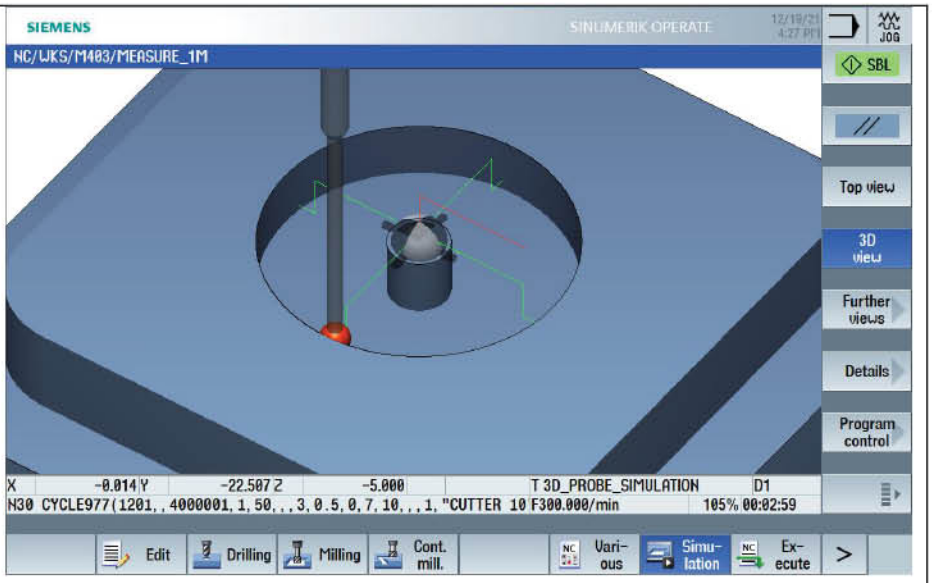
Accept	 Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N39** **N40 CYCLE76(5,0,2,-15,,100,100,15,0,0,0,15,0,0,0.1,0.1,0,5,125,125,0.4,1,1100,1,102)** **N41**
22. Accept	Insert the following “Contour name” into the input mask for the circular pin Ø10 mm in order to mill a chamfer 0,4x45°: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N42 CYCLE62("PIN",1,,)**

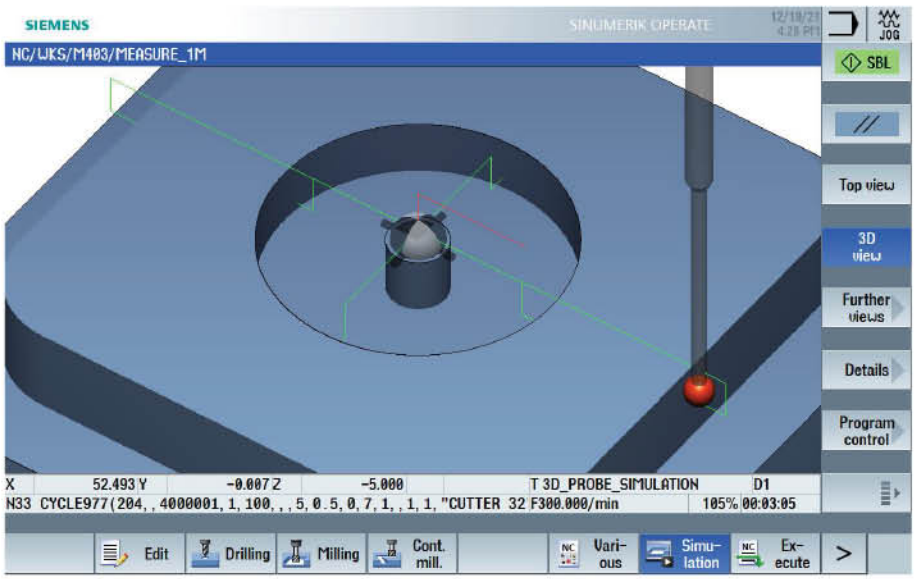
23.	Enter the following values into the input mask for “PATH MILLING”: 
24. 	Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N43** **CYCLE72("",5,0,2,20,2,0.5,0.5,0.1,0.1,10305,41,2,5,0.1,2,5,0,0.4,1,101, 1011,101)**
25. 	Insert the following “Contour name” into the input mask for the circular pocket Ø50 mm in order to mill a chamfer 0,4x45°: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N44** **N45 CYCLE62("CIRCLE",1,,)** **N46**

26.	Enter the following values into the input mask for “PATH MILLING”:  Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N47** **CYCLE72("",5,0,2,20,2,0.5,0.5,0.1,0.1,305,41,2,5,0.1,2,5,0,0.4,1,101,1011,101)** **N48**
27.	Insert the following “Contour name” into the input mask for the blank rectangle 125 mm in order to mill the deburring chamfer 0,4x45°: Confirm your input by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N48** **N49 CYCLE62("BLANK",1,,)**

28.	Enter the following values into the input mask for “PATH MILLING”:  Confirm your inputs by pressing the VSK 8 “Accept”. The following line is inserted into the program: **N50 CYCLE72("",5,-15,2,20,2,0.5,0.5,0.1,0.1,305,41,1,0,0.1,1,0,0,0.4,1,101,1011,101)**
29.	Finally, insert the following technology commands to finish the main program: **N51 T0 M6** **N52 M30**
30. 	Insert the contour for “CIRCLE” behind the main program with the graphic editor: N53 N54 E_LAB_A_CIRCLE: ;#SM Z:2 ;#7__DlgK contour definition begin - Don't change!;*GP*;*RO*;*HD* G17 G90 DIAMOF;*GP* G0 X25 Y0 ;*GP* G3 I=AC(0) J=AC(0) ;*GP* ;CON,0,0.0000,1,1,MST:0,0,AX:X,Y,I,J,TRANS:1;*GP*;*RO*;*HD* ;S,EX:25,EY:0;*GP*;*RO*;*HD* ;ACCW,EX:25,EY:0,CX:0,RAD:25;*GP*;*RO*;*HD* ;#End contour definition end - Don't change!;*GP*;*RO*;*HD* E_LAB_E_CIRCLE:

31. 	Insert the contour for “PIN” behind the main program with the graphic editor: N55 N56 E_LAB_A_PIN: ;#SM Z:2 ;#7__DlgK contour definition begin - Don't change!;*GP*;*RO*;*HD* G17 G90 DIAMOF;*GP* G0 X5 Y0 ;*GP* G3 I=AC(0) J=AC(0) ;*GP* ;CON,0,0.0000,1,1,MST:0,0,AX:X,Y,I,J,TRANS:1;*GP*;*RO*;*HD* ;S,EX:5,EY:0;*GP*;*RO*;*HD* ;ACCW,EX:5,EY:0,CX:0,RAD:5;*GP*;*RO*;*HD* ;#End contour definition end - Don't change!;*GP*;*RO*;*HD* E_LAB_E_PIN:
32. 	Insert the contour for “BLANK” behind the main program with the graphic editor: N57 N58 E_LAB_A_BLANK: ;#SM Z:2 ;#7__DlgK contour definition begin - Don't change!;*GP*;*RO*;*HD* G17 G90 DIAMOF;*GP* G0 X-62.5 Y-62.5 ;*GP* G1 Y62.5 ;*GP* X62.5 ;*GP* Y-62.5 ;*GP* X-62.5 ;*GP* ;CON,0,0.0000,4,4,MST:0,0,AX:X,Y,I,J,TRANS:1;*GP*;*RO*;*HD* ;S,EX:-62.5,EY:-62.5;*GP*;*RO*;*HD* ;LU,DEY:125;*GP*;*RO*;*HD* ;LR,DEX:125;*GP*;*RO*;*HD* ;LD,DEY:-125;*GP*;*RO*;*HD* ;LA,EX:-62.5,EY:-62.5;*GP*;*RO*;*HD* ;#End contour definition end - Don't change!;*GP*;*RO*;*HD* E_LAB_E_BLANK:
33.	Finally, at the end of programming check the machining with the simulation. Therefore, use “T="3D_PROBE_SIMULATION" in the program.



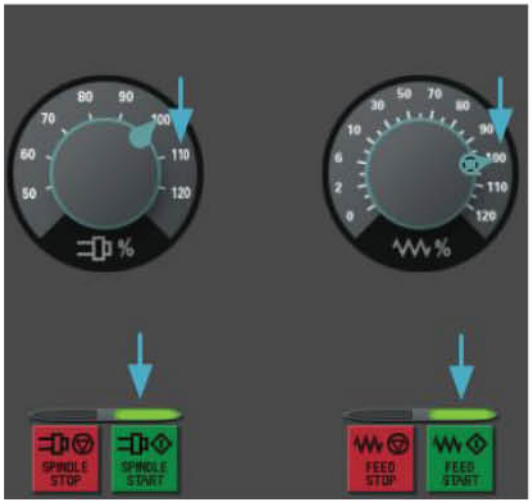
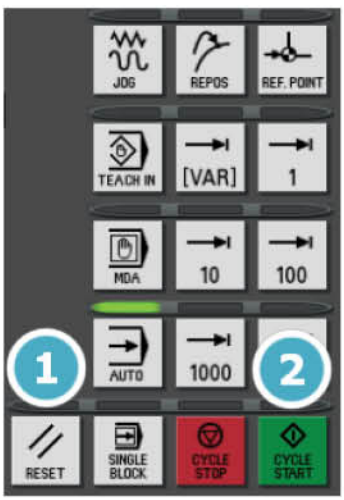


The tool list shows still the initial values in “OFFSET”:

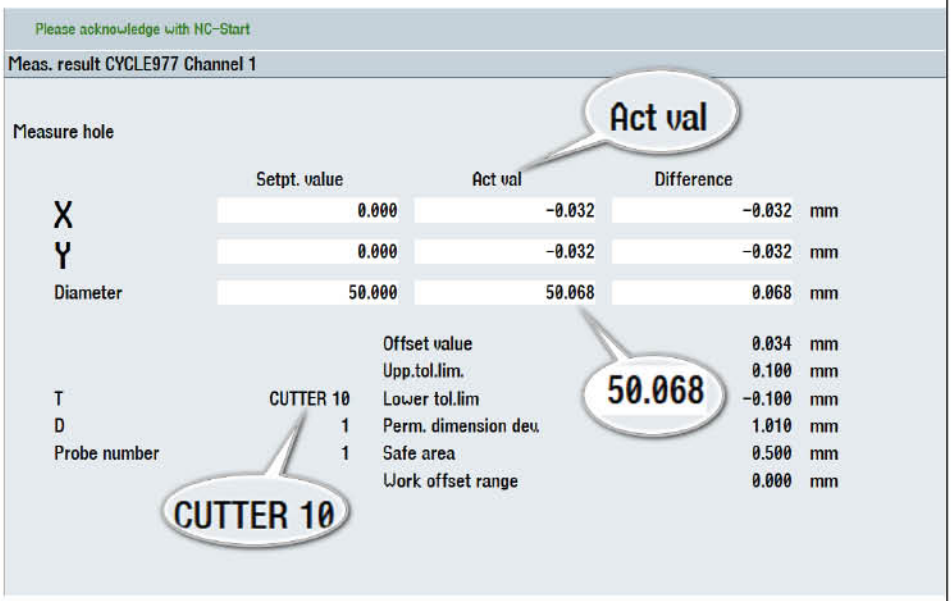
Tool list

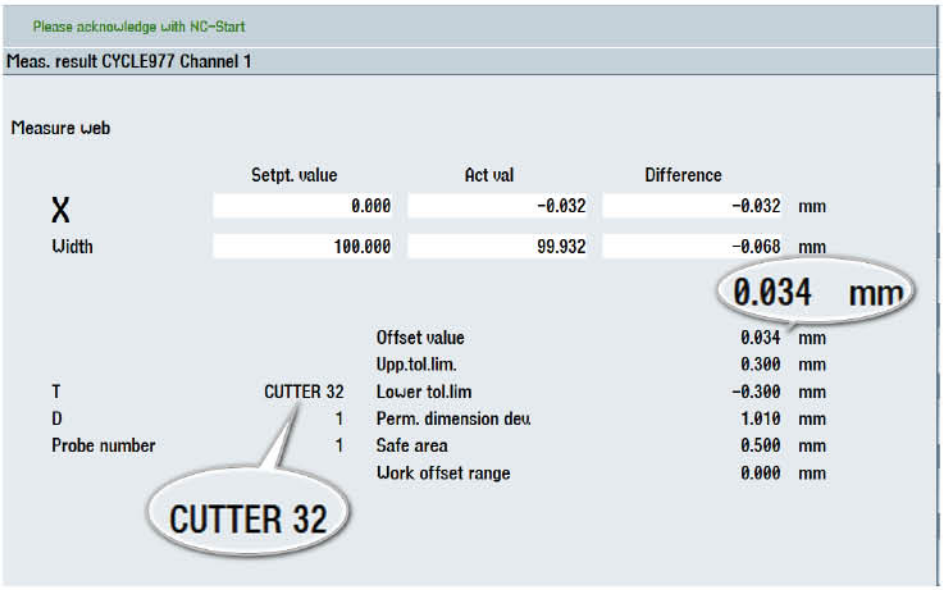
Loc.	Type	Tool name	ST	D	Length	ø			1	2
1		CUTTER 32	1	1	110.000	32.000		3	↺	☑	☐
2		CUTTER 10	1	1	150.000	10.000		4	↺	☑	☐
3		3D_PROBE_SIMULATION	1	1	90.396	5.000			⊠	☐	☐
4		3D_PROBE	1	1	94.396	0.000			⊠	☐	☐

34.	In order to see the measurement result images (reports) and the correction of the tool diameter values the program has to run in “AUTOMATIC” mode. Therefore, use “T="3D_PROBE" in the program.

35.	To start running the program in automatic mode switch on following functions on the operating panel: 
36. 	Choose the program for machining in “EDIT” mode with “EXECUTE”. Double check again the right tool call: N20 M5¶ N21 M0¶ N22 T="3D_PROBE" ;only use for AUTOMATIC¶ N23 ;T="3D_PROBE_SIMULATION" ;only use for SIMULATION¶ N24 M6 ¶
37.	Start the program machining with ”RESET” and “CYCLE START” now: 

38.	Display of the measurement result for the Ø50 by “CYCLE START”:
39.	Display of the measurement result for the length 100 mm by “CYCLE START”:
40.	After running the program, press the function key “OFFSET” to open the “Tool list” for checking the diameter of the milling tools “Cutter 10” and “CUTTER 32”. Here you can find now the correction of the tool diameter as result of the measuring cycle programming:





Tool list

Loc.	Type	Tool name	ST	D	Length	ø			1	2
1		CUTTER 32	1	1	110.000	32.068		3	✓	
2		CUTTER 10	1	1	150.000	10.068		4	✓	
3		3D_PROBE_SIMULATION	1	1	90.396	5.000				
4		3D_PROBE	1	1	94.396	0.000				

3.6. Log measurement results with using the high-level-language

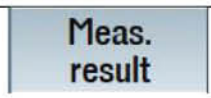
Measurement result - logging from software version 4.7

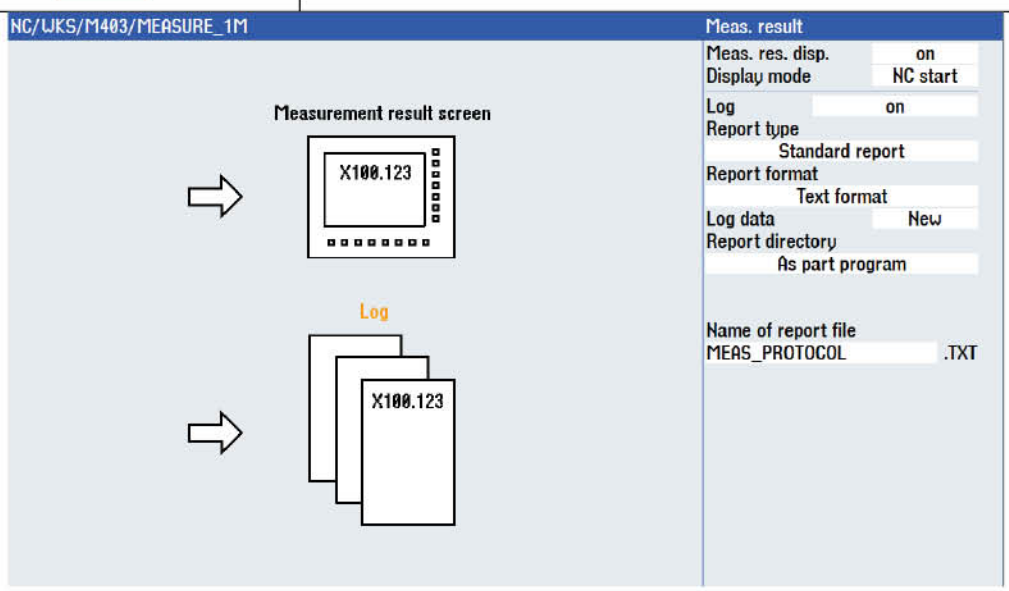
Function

- The activation of the logging function is performed by simply programming the CYCLE150 call.
- The result display and logging can be selected separately and can be controlled independently.
- Parameters in CYCLE150 have a modal effect until the end of the program or reset, or until the cycle is called again.

Procedure

- The logging is activated and deactivated via the program through the appropriate parameterization of CYCLE150.
- This must be programmed once at the start of the measuring program.
- Only when the input parameters are changed must the programming be performed again before the respective measuring cycle call.
- At the end of the program (channel RESET), the functions are deactivated automatically, explicit programming is not required

The part program or ShopMill program to be processed has been created and you are in the editor.	
	1 - Press the "Meas. workpiece" or "Meas. tool" softkey.
	2 - Press the "Meas. result" softkey. The "Measurement Result" input window opens.
	3 - Make the required selection in the select (toggle) fields - see following table.



- The screen form contains a selection field for activating and deactivating the measurement result display with the toggle states "Off"/"On".
- It also contains a selection field for activating and deactivating the logging with the toggle states "Off"/"On"/"Last measurement".
- If both selections fields have the toggle state "Off", all the following input fields for the respective function are hidden.

Parameter	Description
Measurement result display	Off / On
Display mode	Automatic for 8 s / NC start / for alarm
Log	Off/ On / Last measurement
Log type	Standard log / User log
Log format for standard	Text format / Tabular format file extension TXT/ file extension CSV
Log data	New/Append
Log storage	Directory / As part program / Variable
Name of log file	File type set according to selection of log format

Log type:

- The existing infrastructure for the logging of the measurement results should also be made accessible to users for their own purposes - output of user logs.
- A distinction is therefore made between the standard log and the user log.

Log format:

- Standard logs can be output in two log formats, in text format and in tabular format.
- The text format is based on the display of the measurement results on the screen.
- The tabular format is a format that can be imported by Excel (or other spreadsheet programs). This allows the measurement result logs to be statistically processed.
- The Text format / Tabular format selection is only available for standard logs.
- With user logs, the user is responsible for the formatting, i.e. the selection field is hidden for the user log.

Log data:

- A new log file can be created or an existing file appended.
- This is selected via Log data "New" or "Append".
- With "New", the existing file with the same name is deleted and a new file created during logging.

Log storage:

- The path can be specified explicitly or implicitly for the log storage, i.e. "Directory", "As part program" or "Variable" can be selected.
- With "As part program", the path of the higher-level NC program is automatically determined by the logging cycles and the log file is stored there. The following input field for the path is hidden.
- With "Directory", there is an additional input field in which the path is entered. The path does not have to be entered, it can be selected via a dialog which is opened via the additional VSK1 "Select directory".
- The name of the variable is entered for "Variable" in the following field. The name of the log file is then programmed via this variable. It is possible, but not necessary, to specify a path. The log file is then located in the same path as the higher-level NC program.

Name of log file:

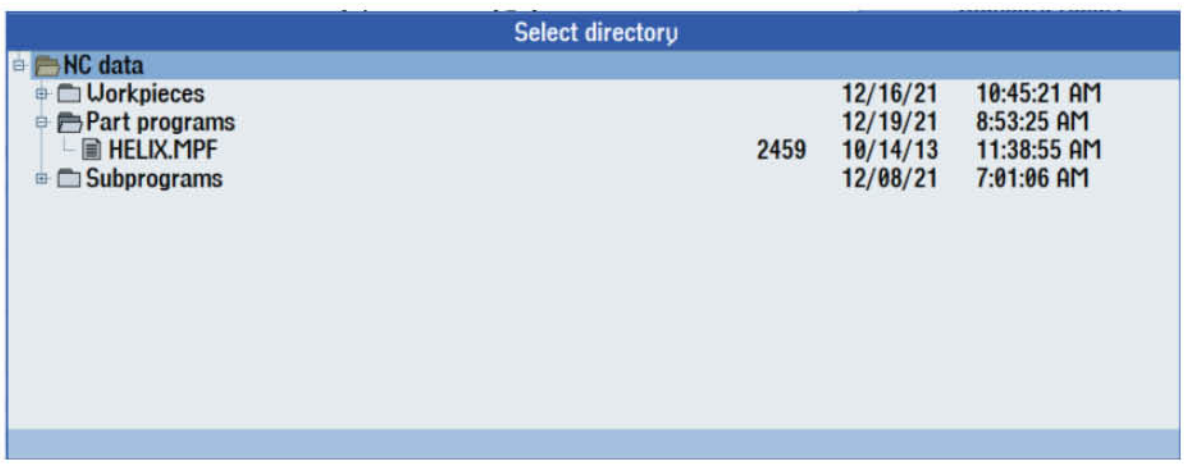
- The log file name can be freely selected. It must comply with the rules for program names in the NC, or for program names when writing to external drives.

File type:

- The following file types are supported:
- Text file - type TXT
- Tabular format – type CSV
- These file types depend on the selected log format.
- The file type cannot be changed; it is only displayed.

Entry level in the selection dialog for log storage:

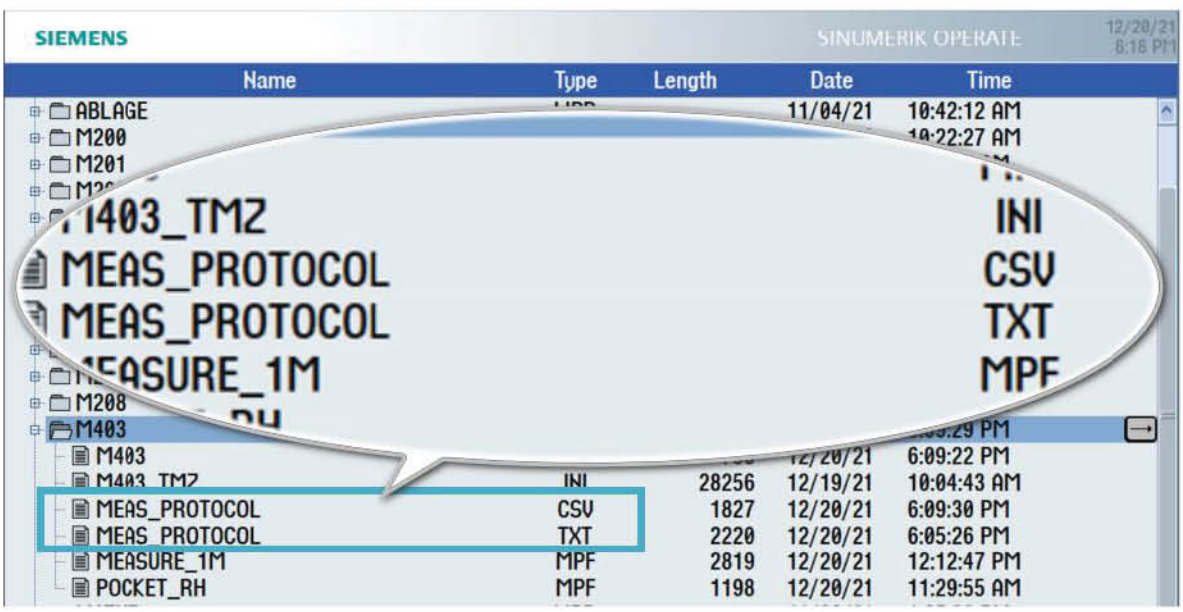
- This softkey "Select directory" only appears when "Directory" is set in the Log storage selection field.



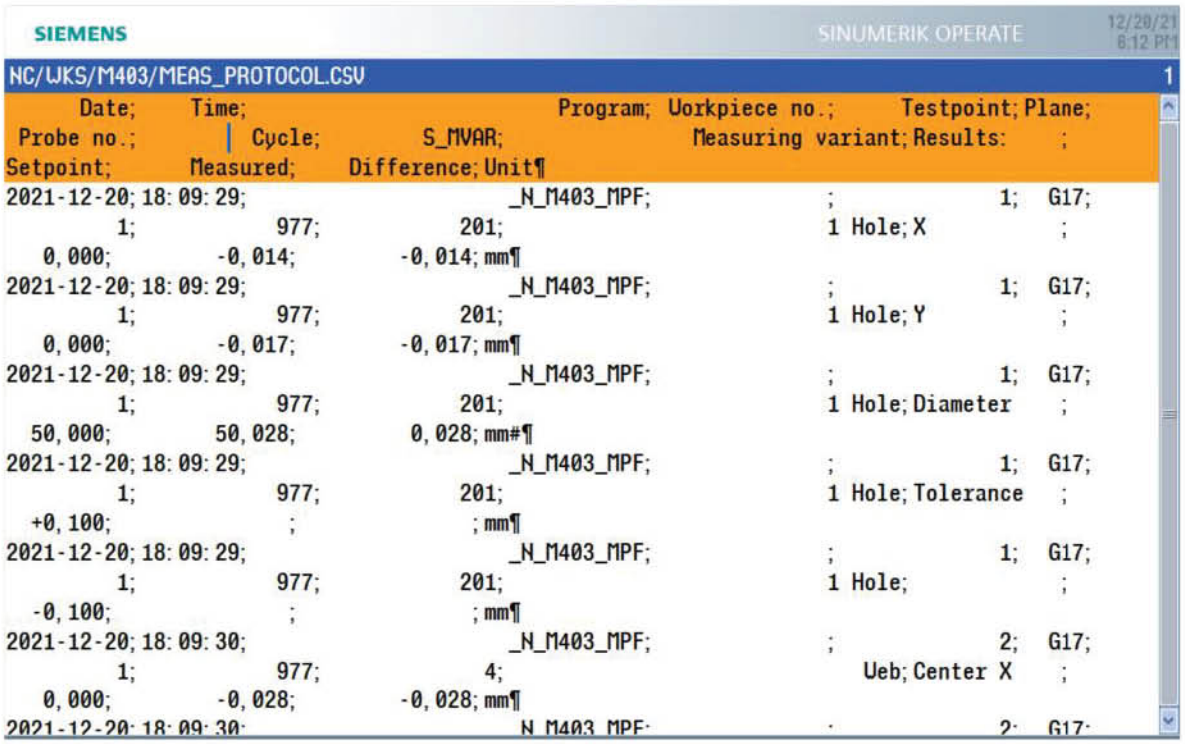
Selection dialog for log storage:

- All drives and paths available in the program manager can be selected.
  - + Local drive
  - + NC data (part program memory)
  - + Network drive(s), if connected
  - + USB (if available)
- Only a path or an available file can be selected in the dialog.
- If only a path is selected, the selection with the complete path is transferred to the screen form with the “Accept” softkey, but can be changed. The name of the log file is entered. If a file is selected, the full path and file name are transferred to the screen form, but can be changed.
- Alternatively, the storage location can be entered. Examples of the selected storage location:
  1. NC data -> Workpiece -> Workpiece "Logs"
     //NC:/WKS.DIR/PROTOKOLLE.WPD
  2. Network drive logs
     //d:/Logs
  3. USB -> measuring_cycle_logs
     //USB:/01/measuring_cycle_logs

Log formats



Format.CVS - Comma-Separated Values


SIEMENS SINUMERIK OPERATE 12/20/21 6:12 PM

NC/WKS/M403/MEAS_PROTOCOL.CSV 1

```
      Date;      Time;                                Program;  Workpiece no.;      Testpoint; Plane;
 Probe no.;         | Cycle;         S_MVAR;                  Measuring variant; Results:     ;
Setpoint;        Measured;     Difference; Unit¶
2021-12-20; 18: 09: 29;                          _N_M403_MPF;              ;              1;   G17;
         1;              977;           201;                                1 Hole; X            ;
   0,000;          -0,014;        -0,014; mm¶
2021-12-20; 18: 09: 29;                          _N_M403_MPF;              ;              1;   G17;
         1;              977;           201;                                1 Hole; Y            ;
   0,000;          -0,017;        -0,017; mm¶
2021-12-20; 18: 09: 29;                          _N_M403_MPF;              ;              1;   G17;
         1;              977;           201;                                1 Hole; Diameter     ;
  50,000;          50,028;         0,028; mm#¶
2021-12-20; 18: 09: 29;                          _N_M403_MPF;              ;              1;   G17;
         1;              977;           201;                                1 Hole; Tolerance    ;
  +0,100;                ;              ; mm¶
2021-12-20; 18: 09: 29;                          _N_M403_MPF;              ;              1;   G17;
         1;              977;           201;                                1 Hole;              ;
  -0,100;                ;              ; mm¶
2021-12-20; 18: 09: 30;                          _N_M403_MPF;              ;              2;   G17;
         1;              977;             4;                                  Ueb; Center X      ;
   0,000;          -0,028;        -0,028; mm¶
2021-12-20; 18: 09: 30;                          _N_M403_MPF;              ;              2;   G17;
```

Format.TXT:

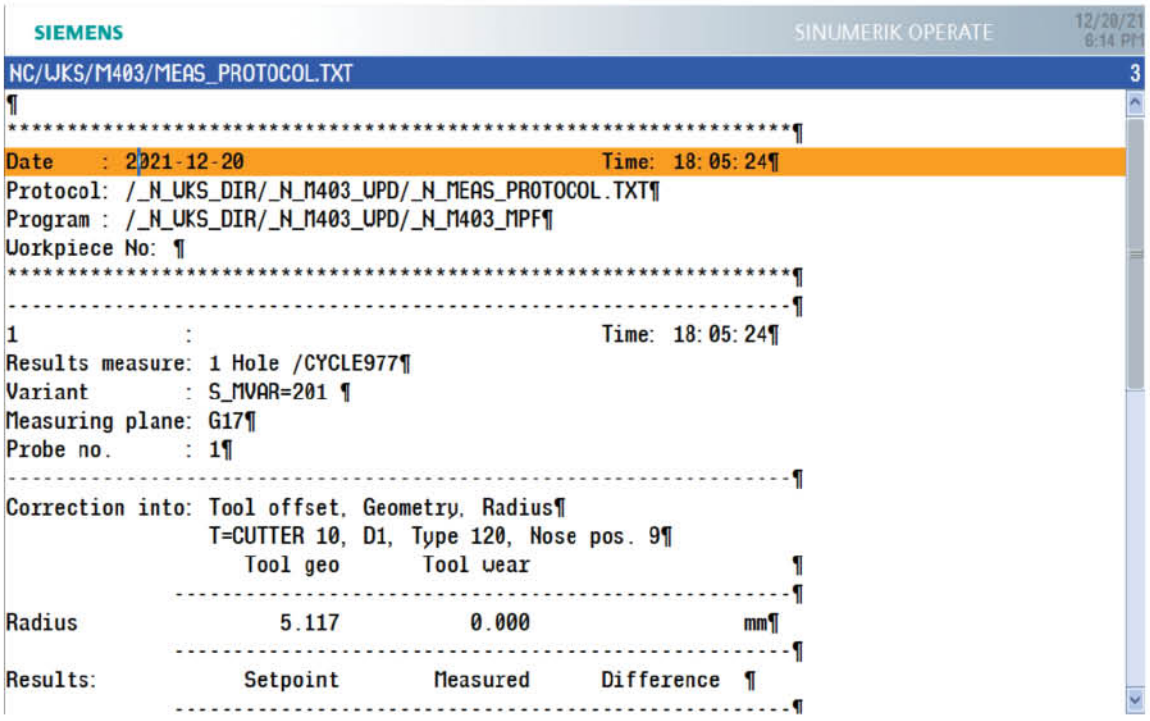


Figure 5.4: Report measurement results

Independent exercise with program Guide and ShopMill

Description:

- A first independent exercise is to be programmed with the help of spigot and pocket milling cycle calls in programGuide and ShopMill.
- This exercise repeats the measuring in “AUTOMATIC” mode with the SINUMERIK measuring cycles.
- The Measuring of the workpiece is to practise with an automatic touch probe and independent selection of the three different modes:
  + 1 - with correction of the work offset
  + 2 - with correction of the tool geometry
  + 3 - measuring only.
- The workpiece is to program and machine in the workshop directly on the CNC milling machine.
- The program includes the measuring cycles to create a measuring protocol in Format “TXT” for all shown elements 1 until 11.

Objective:

- You apply the fundamentals about programming and measuring in "AUTOMATIC" mode using SINUMERIK cycles.
- You can then measure your workpieces simply and precisely on a CNC milling machine with the SINUMERIK measuring cycles.
- The task for selecting the measuring cycles is shown in the graphic below.

- Set up the machine, measure the tools and determine the workpiece zero as shown in the technical drawing.
- All workpiece edges must be chamfered according to the specification from the technical drawing.
- Alternative to the milling tools in the “Tool shape” you can use the tools available at the tool magazine at the CNC.
- All work processes are to be programmed in roughing and finishing mode.
- All measurements are made by calling up the corresponding cycles in the machining program.

Notes:

The measurements are programmed using the graphics and archived in a measurement report:

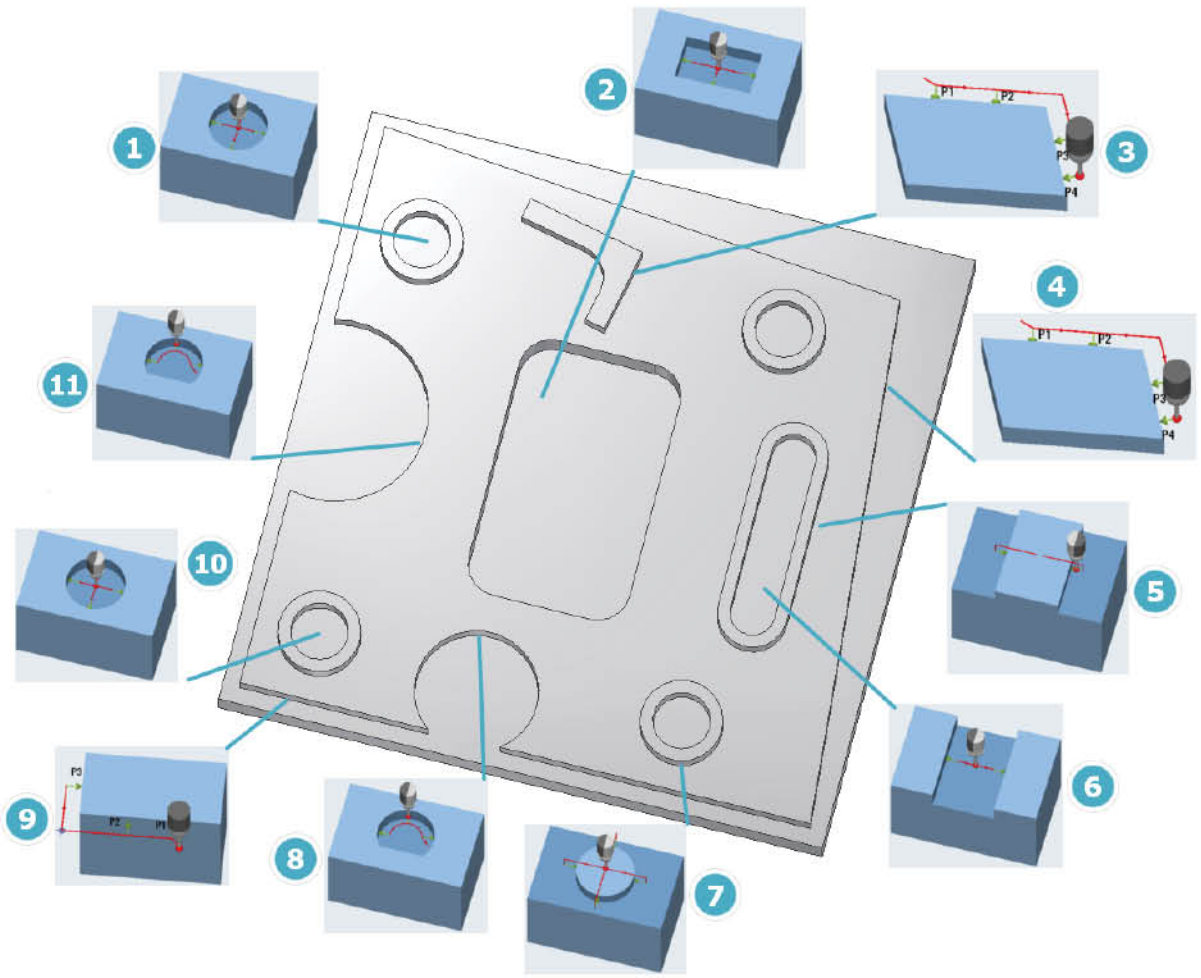
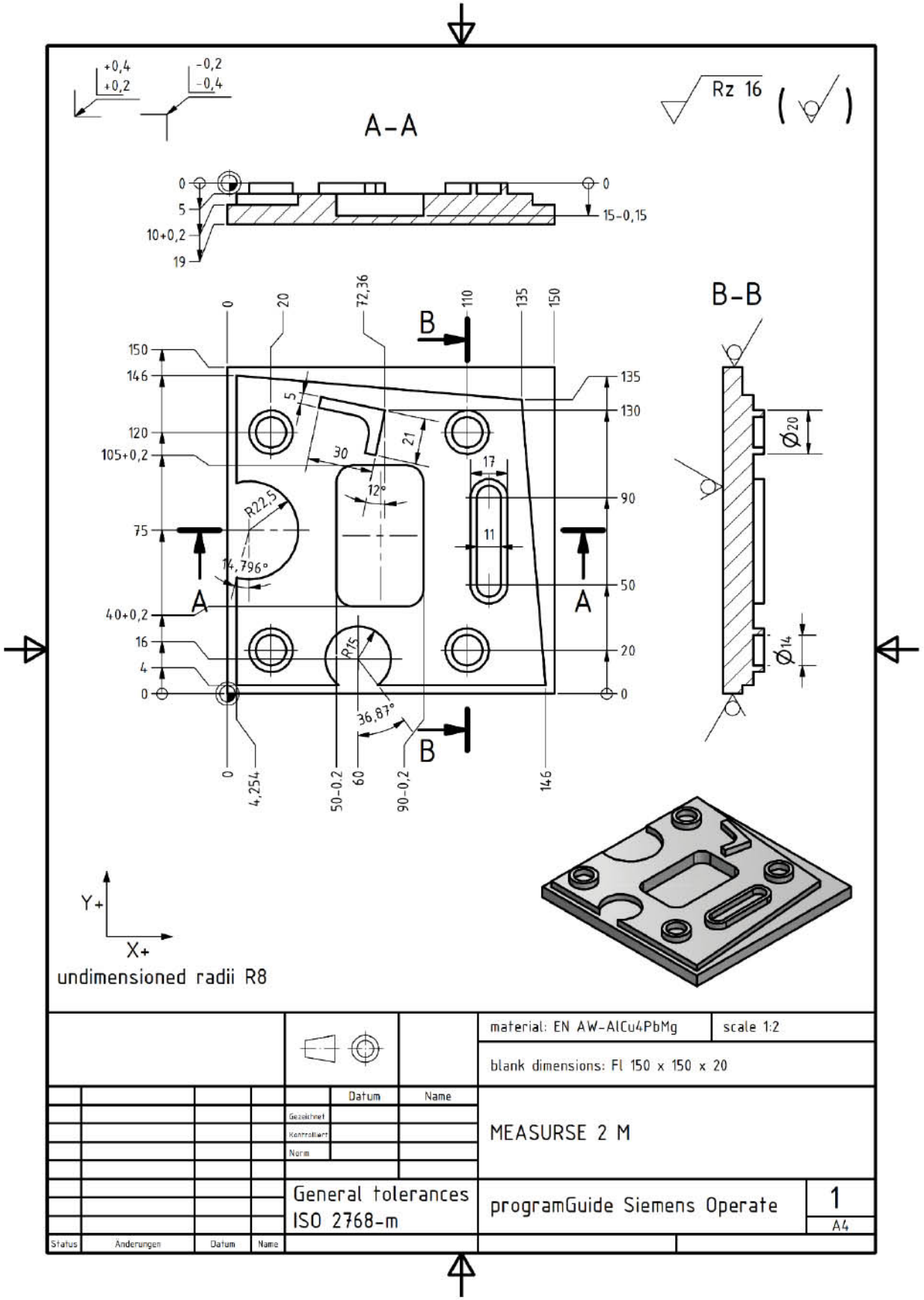
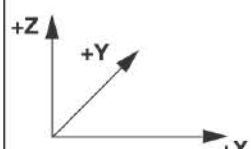


Figure 5.5: Measurements

Technical drawing measure exercise program:



programGuide Siemens Operate VIETNAM 2021/2022		
Tool shape file	CNC Milling	
Workpiece: MEASURE 2 M DIN	Blank material: EN AW-AlCu4PbMg	Machine configuration:
Drawing: 1(1)	Blank dimensions: Fl 150 x 150 x 20	Milling 3 axes 7 Milling 5 axes (X Y Z A C)

	Technological data							
Tool - Nr.	Cutter 32	Cutter 14	Deburrer 6	3D Touch Probe Simulation	3D Touch Probe	Cutter 12	Cutter 8	Cutter 6
Tool Ø (mm)	16	14	6	5	0	12	8	6
Cutting speed (m/min)	150 / 175	150 / 175	105	-	-	150 / 175	150 / 175	150 / 175
Cutting depth a_p = max.	8	7	4	-	-	6	4	3
Cutting material	SC	SC	SC	-	-	SC	SC	SC
Feed tooth fz / rev. (mm)	0,08	0,07	0,025	300 mm / min	300 mm / min	0,06	0,04	0,025
Number of teeth N	4	4	4			4	4	3
								
	✓	✓				✓	✓	✓

Function of the measuring cycles

- The following table describes all the measuring cycle functions for the turning and milling technologies.
- The table shows in different colours the availability of the measuring cycles for:

Only for milling technology

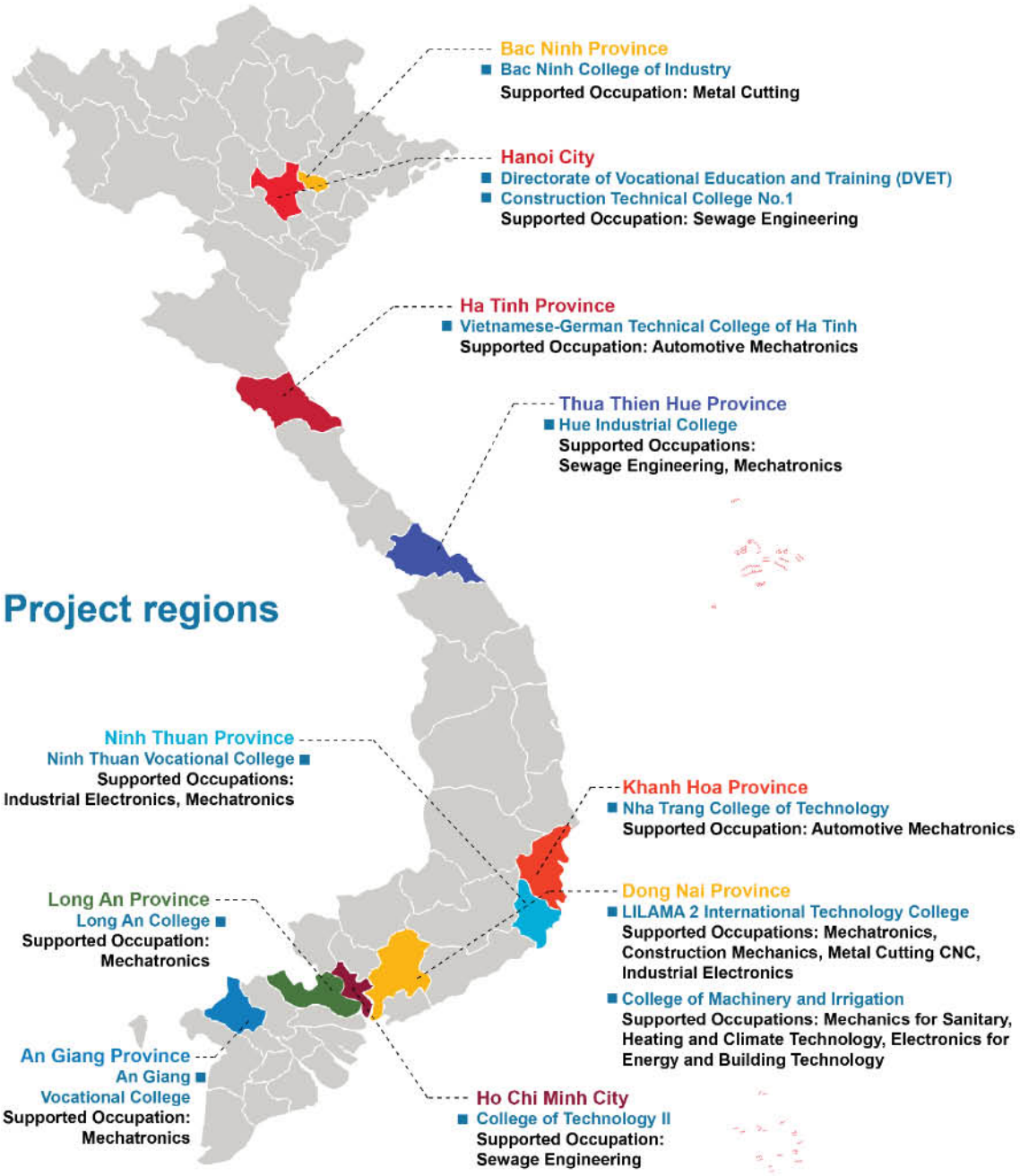
Only for turning technology *

Measuring cycle	Description	Measuring versions
CYCLE973*	This measuring cycle can be used to calibrate a workpiece probe on a surface on the workpiece or in a groove.	• Calibrate probe - length • Calibrate probe - radius on surface • Calibrate probe - probe in groove
CYCLE974*	This measuring cycle can be used to determine the workpiece zero in the selected measuring axis or a tool offset with 1-point measurement.	• Turning measurement - front edge • Turning measurement - inside diameter • Turning measurement - outside diameter
CYCLE994*	This measuring cycle can be used to determine the workpiece zero in the selected measuring axis with 2-point measurement. To do this, two opposite measuring points on the diameter are approached automatically in succession.	• Turning measurement - inside diameter • Turning measurement - outside diameter
CYCLE976	Using this measuring cycle, a workpiece probe can be calibrated in a calibration ring or on a calibration ball completely in the working plane or at an edge for a particular axis and direction.	• Calibrate probe - length on surface • Calibrate probe - radius in ring • Calibrate probe - radius on edge • Calibrate probe - calibration on sphere
CYCLE961	This measuring cycle can be used to deter-mine the position of a workpiece corner (inner or outer) and use this as work offset.	• Corner - right-angled corner • Corner - any corner
CYCLE977	This measuring cycle can be used to deter-mine the centre point in the plane as well as the width or the diameter.	• Edge distance - groove • Edge distance - rib • Hole - rectangular pocket • Hole - 1 hole • Spigot - rectangular spigot • Spigot - 1 circular spigot

CYCLE978	This measuring cycle can be used to measure the position of an edge in the workpiece coordinate system.	• Edge distance - set edge
CYCLE979	This measuring cycle can be used to measure the centre point in the plane and the radius of circle segments.	• Hole - inner circle segment • Spigot - outer circle segment
CYCLE995	With this measuring cycle the angularity of the spindle on a machine tool can be measured.	• 3D - angular deviation spindle
CYCLE996	This measuring cycle can be used to determine transformation-relevant data for kinematic transformations with contained rotary axes.	• 3D - kinematics
CYCLE9960	Transformation-relevant data for kinematic transformations that contain rotary axes can be determined with this measuring cycle.	• 3D - kinematics
CYCLE997	This measuring cycle can be used to determine the centre point and diameter of a ball. Furthermore, the centre points of three distributed balls can be measured. The plane formed through the three ball centre points, regarding its angular position, is determined referred to the working plane in the workpiece coordinate system.	• 3D - sphere • 3D - 3 spheres
CYCLE998	This measuring cycle can be used to determine the angular position of a surface (plane) referred to the working plane and the angle of edges in the workpiece coordinate system.	• Edge distance - align edge • 3D - align plane
CYCLE971	This measuring cycle can be used to calibrate a tool probe and measure the tool length and/or tool radius for milling tools.	• Calibrate probe • Measure too
CYCLE982*	This measuring cycle can be used to calibrate a tool probe and measure turning, drilling and milling tools on turning machines.	• Calibrate probe • Turning tool • Milling tool • Drill

APPENDIX

1. PROJECT REGIONS AND PARTNERING TVET INSTITUTES



APPENDIX

2. LIST OF CHARTS

Chart 1.1: Structure of main program 29
Chart 2.1: Product manufacturing process 212
Flowchart 4.1: Technological steps for practical exercise M_200 315
Flowchart 4.2: Technological steps for practical exercise M_201 324
Flowchart 4.3: Technological steps for practical exercises using parameters 336
Flowchart 4.4: Technological steps for practical exercises using jump and user variables to error queries and error loops 347
Flowchart 4.5: Technological steps for practical exercises using system variables and subprogram 363
Flowchart 4.6: Technological steps for practical exercises using system variables to tool queries and error loops 376
Flowchart 4.7: Technological steps for practical exercises using R parameters, user and system variables, frames to query and work offset 391

3. LIST OF TABLES

Table 1.1: Parameters of stock removal 1 cycle 45
Table 1.2: Parameters of stock removal for contour cycle 46
Table 1.3: Parameters of cut residual stock cycle 47
Table 1.4: Parameters of groove 2 cycle 48
Table 1.5: Parameters of undercut thread (DIN76) cycle 49
Table 1.6: Parameters of thread longitudinal cycle 50
Table 1.7: Parameters of cutoff cycle 51
Table 1.8: Parameters of path milling cycle 52
Table 1.9: Parameters of rectangular pocket cycle 53
Table 1.10: Parameters of circular pocket cycle 54
Table 1.11: Parameters of rectangular spigot cycle 55
Table 1.12: Parameters of circular spigot cycle 56
Table 1.13: Parameters of multi-edge spigot cycle 57
Table 1.14: Parameters of longitudinal slot cycle 58
Table 1.15: Parameters of circumferential slot cycle 59
Table 1.16: Parameters of open slot cycle 60
Table 1.17: Parameters of mill pocket cycle 61
Table 1.18: Parameters of pocket residual material cycle 62
Table 1.19: Parameters of milling spigot cycle 63
Table 1.20: Parameters of spigot residual material cycle 64
Table 1.21: Parameters of centering cycle 65
Table 1.22: Parameters of drilling cycle 66
Table 1.23: Parameters of deep hole drilling 1 cycle 67
Table 1.24: Parameters of reaming cycle 68
Table 1.25: Parameters of tapping cycle 69
Table 4.1: Table of some command addresses used in block 311

4. LIST OF FIGURES

Figure 1.1: Main screen of Sinumerik Operate 25
Figure 1.2: General structure of a G code program 30
Figure 1.3: Types of offsets 36
Figure 1.4: Detail contour created by graphic editor 43
Figure 1.5: The programmed contour is limited by the start label: BORDER_ST and the end label: BORDER_ST 43
Figure 1.6: Creating subprogram with programGUIDE G code 44
Figure 1.7: Subprogram with G/M code for tool change 44
Figure 1.8: Four types to call contour in the program 44
Figure 2.1: The 3 Plane selection G-Codes are G17, G18, G19 119
Figure 2.2: G18 - Turning plane selection 120
Figure 2.3: G17- Front face with polar coordinates 121
Figure 2.4: G17- Front face with virtual Y axis 122
Figure 2.5: G17- Front face machining with real Y axis 123
Figure 2.6: G19 - Cylinder surface with polar coordinates 124
Figure 2.7: G19 - Cylinder surface with virtual Y axis 124
Figure 2.8: G19 - Chord plane 125
Figure 2.9: G19 - Inclined chord plane 125
Figure 2.10: Administration of the work offset 129
Figure 2.11: Mirror, offset, rotate and scale 130
Figure 2.12: Shifting with “TRANS” and “ATRANS” 131
Figure 2.13: Rotating with “ROT” and “AROT” 134
Figure 2.14: Direction of rotation 136
Figure 2.15: Mirroring with “MIRROR” and “AMIRROR” 137
Figure 2.16: Scaling with “SCALE” and “ASCALE” 139
Figure 2.17: Diagram command ASCALE 140
Figure 2.18: Drilling on front surface - G17 143
Figure 2.19: Drilling on cylindrical surface - G19 154
Figure 2.20: Milling on front face - G17 167
Figure 2.21: Milling on front face - G17 170
Figure 2.22: Calculations of the work piece zero points (WPZ) 205
Figure 2.23: The workpiece is clamped on the self-centering 3-prong access plate 206
Figure 2.24: Main spindle chuck and opposite chuck (reverse chuck) 207
Figure 2.25: Opposite chuck position (reverse chuck) 207
Figure 2.26: Spindle chuck size 208
Figure 2.27: Dimension of chuck opposite chuck (reverse chuck) 208

Figure 2.28: Dimension diagram of main chuck and opposite chuck (reverse chuck) 209
Figure 2.29: Collet and chuck 211
Figure 2.30: Dimension diagram of spindle chuck and opposite chuck (reverse chuck) 215
Figure 2.31: Example dimension of spindle chuck and opposite chuck (reverse chuck) 216
Figure 2.32: Example spindle chuck size and opposite chuck (reverse chuck) 223
Figure 3.1: Rotary/swivel table 241
Figure 3.2: Articulated rotary/swivel table 241
Figure 3.3: Fork head 241
Figure 3.4: Articulated head kinematics 242
Figure 3.5: Mixed kinematics 242
Figure 3.6: Swivel plane parameters 243
Figure 3.7: Programming with swivel mode Axis by axis 245
Figure 3.8: Programming with swivel mode Projection angle 246
Figure 3.9: Programming with swivel mode Directly 246
Figure 4.1: Stucture of subprogram 321
Figure 4.2: Opening user variables 331
Figure 4.3: R parameters are displayed in the user variables section 331
Figure 4.4: Table of tool numbers in the offset of the Milling machine 356
Figure 4.5: Table of tool numbers in the offset of the lathe 357
Figure 4.6: Main program memory is stored in the Workpieces folder with the extension MPF 368
Figure 4.7: Subprogram memory of cycle is stored in the Subprograms folder with the extension SPF 369
Figure 4.8: The tool type is shown when the tool symbol in the corresponding "TYPE" section is selected 371
Figure 4.9: Showing the number of active tools on the working area 372
Figure 4.10: Shows the number of active tool edge on the work area 372
Figure 4.11: Machine operating area 386
Figure 4.12: The important G function groups 386
Figure 4.13: The G function groups with the active command 387
Figure 4.14: Copy the program to User Cycles 439
Figure 4.15: Copy parameters R1 to R12 and deactivate R4, R5 440
Figure 4.16: Deactivate the command line that calls the quick tool to positions X=R4 and Y=R5 440
Figure 4.17: Copy the orange header in “CYCLE81” 441
Figure 4.18: Paste the first instruction of the program with the variable declaration to get the

values from the main program 441
Figure 4.19: Inserts user variables in the instruction header line to get values from the main program. 442
Figure 5.1: Measure dimensions of product on CNC milling machines 451
Figure 5.2: Workpiece coordinate system and machine coordinate system when measuring .. 452
Figure 5.3: Setting of manual measurement mode and automatic measurement mode 460
Figure 5.4: Report measurement results..... 498
Figure 5.5: Measurements..... 499

REFERENCES

- Metal Engineering, Young Publishing House, Vietnam
- Mechanical and Metal Trades Handbook (Tabellen Buch Metall), 3rd English Edition, EUROPA –LEHRMITTEL, Germany.
- Chau Manh Luc, Textbook of machining technology on CNC machines, Danang University of Science and Technology.
- Tran The San, Nguyen Ngoc Phuong, CNC Programming Manual, Da Nang Publishing House.
- Tran Van Dich, Technology on CNC machines, Hanoi Science and Technology Publishing House.
- Ta Duy Liem, Bui Duc Anh, Phan Van, Le Duc Bao, Fundamental of CNC machines, Hanoi Bach Khoa Publishing House.
- Training materials related to Sinumerik Operate 840Dsl, Siemens Corp., 2019

The training module was developed in the frame of the Development Partnership with the Private Sector (DPP) "Vocational Training for Smart Manufacturing in Machine Tools". The DPP is supported by the develoPPP.de programme of the German Federal Ministry for Economic Cooperation and Development (BMZ) and jointly implemented by the cooperation partners Siemens Vietnam, LILAMA 2 International Technology College and Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH, represented by the Programme "Reform of Technical and Vocational Education and Training in Viet Nam".

Viet Nam, 12/2023