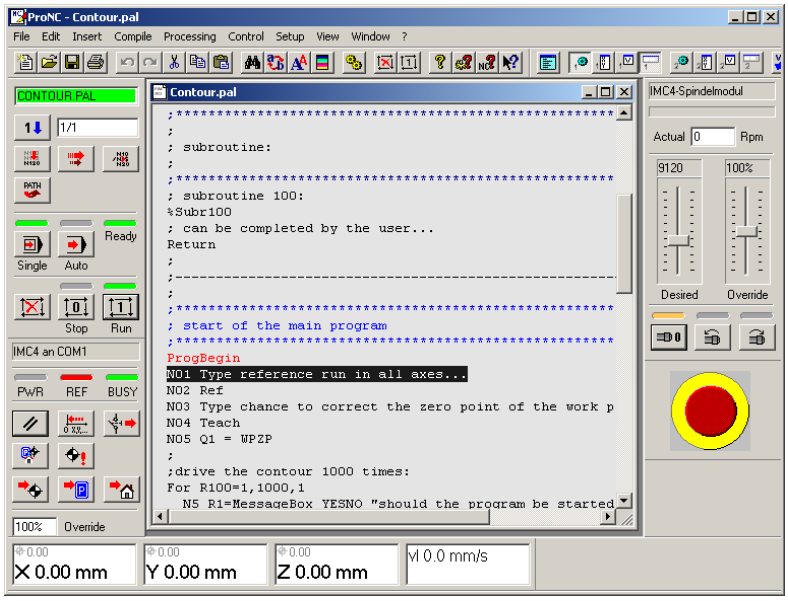


isel-ProNC

Operating Instruction

www.isel.com

isel®

To the Manual:

In this manual you find same symbols pointing out your attention to important information.

Caution:	**Example:**	**Hint:**	**Information:**
			

© ***isel***automation KG 2003
All rights reserved.

In spite of all care printing errors and mistakes cannot be ruled out completely.
Suggestions for improvement and notes on errors are always welcomed.

No part of this publication may be copied or distributed, transmitted, transcribed, stored in a retrieval system, without the express written permission of iselautomation KG.
All information are supplied without liability. Changes are possible at any time without prior notice.

Producer: *isel*automation KG
Bürgermeister-Ebert-Straße 40
D-36124 Eichenzell

Tel.: (06659) 981-0
Fax: (06659) 981-777
email: automation@isel.com
http://www.isel.com

Version 19/2003

Contents

1 INTRODUCTION 6

1.1 COPYRIGHT 6
1.2 SCOPE OF DELIVERY AND INSTALLATION 6
1.3 HARD- AND SOFTWARE REQUIREMENTS 9
1.4 PROGRAM CALL 9
1.5 DEDICATION OF THE PROGRAM-PACKAGE PRONC 10
1.5.1 DLL Concept 10
1.5.2 Functionalities 11
1.5.3 Use of ProNC 12
1.5.4 Programming support 12
1.5.5 Implementation 13
1.6 SCREEN LAYOUT 14
1.6.1 Menu bars, symbol bars, state bars 15
1.6.2 Standard windows 16
1.6.3 Additional windows 19

2 THE FILES 22

2.1 THE INITIALIZATION FILES 22
2.1.1 ProNc.ini 22
2.1.2 NcCtrl.ini 22
2.2 THE USER FILES 22
2.2.1 The source file 23
2.2.2 The geometry file 25
2.2.2.1 Definition 25
2.2.2.2 Structure of the geometry file 26
2.2.2.3 Construction of geometry file 29
2.3 FILES GENERATED FROM THE COMPILER 30
2.3.1 The CNC target file 30
2.3.2 The label file 31
2.3.3 The error file 31
2.4 FILES CREATED BY THE NC INTERPRETER 31
2.4.1 The report file 31
2.4.2 The variable file 32

3 SOFTWARE COMPONENTS OF PRONC 33

3.1 THE EDITOR 33
3.1.1 Positioning of the cursor in text 34
3.1.2 Editing functions 35
3.1.3 Block operation copy, insert 35
3.1.4 Functions Search, Replace 36
3.1.5 Other functions 37
3.1.6 File functions 37
3.2 THE COMPILER 37
3.3 THE CNC INTERPRETER 38
3.4 CONFIGURATION 39
3.4.1 File structure CNCworkbench 39
3.4.2 Principle of Module DLLs 41
3.4.2.1 Module types and DLLs 42
3.4.3 Assignment interface 44
3.4.4 Configuration dialog 45

4 COORDINATION OF ALL SYSTEM COMPONENTS 47

4.1 DATA FLOW AND INTERFACES 47

5 OPERATING 49

5.1 PHILOSOPHY 49
5.2 MENU FILE 49
5.2.1 Create a new file 51
5.2.2 Open file 51
5.2.3 Import file 51
5.2.4 Set up printer 51
5.2.5 Exit of ProNC 51
5.2.6 Close file 52
5.2.7 Save 52
5.2.8 Print 52
5.2.9 Side view 52
5.3 MENU EDIT 52
5.3.1 Delete actions, restore actions 53
5.3.2 Copy passage, insert passage 53
5.3.3 Searching and replacing functions 54
5.3.4 Format text 54
5.4 MENU INSERT 54
5.4.1 Working with program templates 55
5.4.2 Date and time 56
5.4.3 Insert file 56
5.4.4 Insert command 56
5.4.5 Insert subprogram 56
5.4.6 Linear movement 57
5.4.7 Linear movement in space 58
5.4.8 Circle or arc 59
5.4.9 Rectangle 61
5.4.10 Polygon 61
5.4.11 Drilling cycle 63
5.4.12 Signal input, signal output 63
5.4.13 User-defined message 66
5.4.14 Branching 67
5.4.15 Counting loop 68
5.5 MENU COMPILE 70
5.5.1 Error window 70
5.6 MENU PROCESSING 72
5.6.1 Operating mode Start, Stop, Terminate 72
5.6.2 Breakpoints 73
5.6.3 Program skip, Rapid traverse, Set skip 74
5.6.4 Online path calculation 75
5.7 MENU CONTROL 75
5.7.1 Reset 75
5.7.2 Reference run 76
5.7.3 Machine 76
5.7.3.1 Simulate reference run 76
5.7.3.2 Activate work piece zero point 77
5.7.3.3 Delete active work piece zero point 78
5.7.3.4 Approach to work piece zero point 78
5.7.3.5 Move to parking position 78
5.7.3.6 Move to home position 79
5.7.3.7 Override 79
5.7.3.8 Velocities 80
5.7.3.9 Manual movement 80
5.7.3.10 Setup machine positions 82
5.7.3.11 Move to machine positions 86
5.7.4 Spindle 86
5.7.5 Input/Output 88

5.7.6 Toolchange ... *88*
5.8 MENU SETUP ... 89
5.8.1 Workspace ... *90*
5.8.2 Directories ... *90*
5.8.3 Units ... *91*
5.8.4 Interpreter ... *92*
5.8.5 Compiler ... *93*
5.8.6 Processing ... *93*
5.8.7 Control ... *94*
5.8.7.1 Motion Control Module ... 96
5.8.7.2 Main spindles ... 97
5.8.7.3 Input /output module ... 97
5.8.7.4 Tool administration ... 99
5.9 MENU VIEW ... 101
5.10 MENU WINDOW ... 102

6 WORKING SESSION ... 103

6 WORKING SESSION ... 103
6.1 PROCESSING OF PROGRAMS CREATED WITH ISY ... 103
6.1.1 Start of ProNC ... *103*
6.1.2 Open user program ... *104*
6.1.3 Perform the ready state of the equipment ... *106*
6.1.4 Determine work piece zero point ... *106*
6.1.5 Settings for processing of the CNC file ... *110*
6.1.6 Setting tool change ... *111*
6.1.7 Program start ... *114*
6.2 TEXTUALLY PROGRAMMING AND PROCESSING ... 115
6.2.1 Start of the program ProNC ... *115*
6.2.2 User program open and compile ... *115*
6.2.3 Perform the ready state of the equipment ... *117*
6.2.4 Program start ... *117*
6.2.5. Statements to work piece zero point ... *118*
6.2.6 Program process ... *119*

GLOSSARY ... 123

INDEX ... 124

1 Introduction

1.1 Copyright

Copyright: All rights to the manual and programming package ProNC, especially the copyright are owned by

isel-automation KG
Bürgermeister-Ebert-Straße 40
D - 36124 Eichenzell

Copying and transfer the software: All unauthorised copying, distribution or transfer of this software is strictly forbidden and will be prosecuted criminally.

The legal use and distribution of ProNC: The purchase of the installation-CD or disks including the manuals allows the using comparable to that of a book. With a book, it is not possible for a book to be read at different places by several people at the same time. Similarly, the program-package ProNC may not be used on different controllers (PC-based) at different places by several people simultaneously.

Backup copies: Backup copies of the program-package Pro-NC are permitted. In no case it is allowed to make those copies available to third parties.

Compensation: At violations against the copyright the purchaser engages opposite the owner of the copyright, ***isel***automation KG, to damage compensation.

Liability for application programs: The program-package ProNC as well as the manual was produced with carefulness. All application programs printed or as example programs on CD, were tested on a corresponding hardware. iselautomation does not assume any liability or guarantee, that this manual, the program-package ProNC or application programs are perfectly or suitable for a particular purpose. For consequence detriments each legal responsibility or liability is ruled out.

Suggestions: Since mistakes and errors cannot be ruled out, we are always grateful for suggestions, written notes or opinions.
email: tech-support@isel.com

1.2 Scope of delivery and installation

Delivery: Included in delivery of the software product ProNC are:
The Manual 1 **Operating Instructions** and the Manual 2 **Programming Instructions** as PDF-file (on the installation CD) and one installation CD or three installation disks with the following directory structure and the named files:

Directories / sub directories	Files
\CNCworkbench\Bin	**Programs CNCworkbench** **Configuration files** for ProNC
\CNCworkbench\Control	**Configuration files** for - I/O module - Motion module - Spindle module - Tool change module
\CNCworkbench\Help	**Help files**
\CNCworkbench\NcProg	CNC user files in ISO or PAL syntax **Frame files** **User template**
\CNCworkbench\Temp	**Error files**

Hint:

The operating instruction and programming instruction can be ordered additional as manuals:
Part.-no: 970333 BE001 ProNC Operating instruction
Part.-no: 970333 BE002 ProNC Programming instruction

ProNC software components:

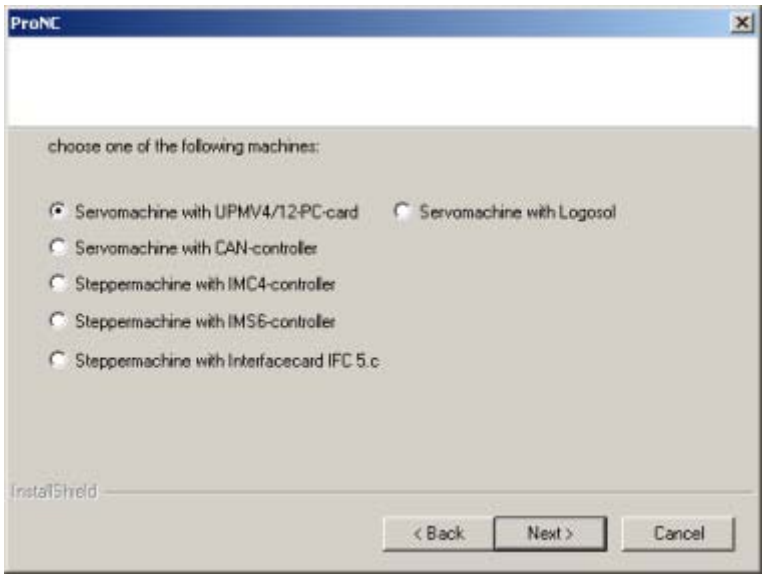
- Converter NCP -> PAL for import of NCP files created with ISY
- Editor (Rich Text Format RTF), ISO / PAL compiler, CNC interpreter (performance range described in the manual "Programming instruction")
- Code generator for input-dialog (ISO or PAL): Macro programming
- Handling of geometry files (frame files): Teach-In, correction, test
- Configuration dialog (motion control, 4 x spindle, 4 x I/O module, tool changer)
- Online-help (Operating- and Programming instruction) in Windows-hlp format
- Imbedding of user software with USER-DLL

Installation of ProNC:

Before starting the installation we recommend you, to **quit all programs** currently running.

Insert the installation CD into the appropriate drive of your computer. The installation will start **automatically**. Follow the directions on screen.

Please **select** within the installation the relevant type of your applied **isel-machine or controller**.

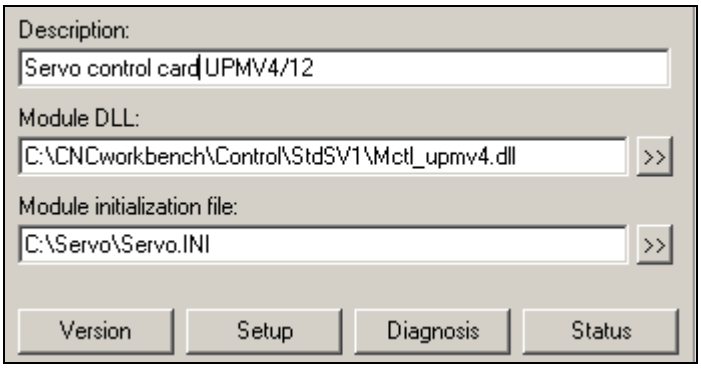


After a **successful setup** the program icon will appear on your desktop.

In a particular case (no CD-drive), it is also possible, to install the program ProNC from disk. It is the same proceeding, additional you have to change the disk after request.

Hint:

Would you like to install ProNC for an isel-machine with UPMV4/12-PC-Card, after deciding for a servo controller you can select such an already created servo.ini-file.
The file servo.ini can be created by two adjustment and service tools PAREIN.exe and PARKON.exe, which are included in delivery.
After creating the servo.ini-file must be assigned to ProNC.
Please select the function "Setup" from the menu bar and then enter "Control". Select "Motion control modules", "Axis system 1" and then move the cursor into the line below the text **"Module DLL",** press the button >>, select the directory containing the servo.ini file and choose it with double click.



More: 5.8.7 Menu Setup - Control

Tip: You can also start setup with help of the **system control** in Windows: Please select the following sequence:

Start - Einstellungen - Systemsteuerung - **Software** - Installieren

The automatic search for a setup program is carried out in the floppy disk drive or CD-ROM drive. After a successful search please press the button "Complete" and the installation will be started.

1.3 Hard- and software requirements

System requirement: The use of ProNC requires:

- **PC from Pentium II**, at least 333 MHz

- at least 64 **Mbyte memory**

- Windows 98, Windows NT 4.1, Windows 2000 or Windows XP

Hint:

For use of the operating system depending on the used control you will find more details in section 1.5.1 DLL-Concept.

1.4 Program call

Program call with **program icon**: Please start **ProNC** with double click on the **program icon** .

Program call with **start menu**: Please click on the Windows button **Start**.
Below the name **programs** you will find the program group **CNCworkbench**.

By clicking with the mouse on **ProNC** the program will be loaded and ProNC reports with the **start screen** on your desktop.

More:
1.6 Screen layout

1.5 Dedication of the program-package ProNC

The software product **ProNC** is the further development of the control software



- **Remote**
- **Pro-DIN** and
- **Pro-PAL**

till now offered by iselautomation KG exclusive for the operating system MS-DOS ® as one new, efficient operator and program interface for PC under the operating system Windows ® (Windows 98, Windows NT, Windows 2000, Windows XP). **All NC-programs** for Remote (isel-NCP-format), Pro-DIN (DIN- / ISO-format) or
Pro-PAL (isel-PAL-format) that have been applied by users **can be processed with ProNC** .

1.5.1 DLL Concept

Concept: DLLs

ProNC uses consequently the MS-Windows concept of **D**ynamic **L**ink **L**ibrary - **DLL** to implement the necessary modules resp. device interfaces for the controlling of control units, motor controller boards or intelligent power modules for movement axes / axis systems -> ***Motion control DLLs***

Frequency converter for main spindle drives -> ***Spindle DLLs***

Hardware of binary / analog input and output -> ***Input/Output DLLs***



Hardware for operator control and safety functions,
measurement equipment and CAN field bus interface (CAN) -> ***additional DLLs***

More: 4.1 Data flow and interface

The item **module is uniform used** in all following executions. This item could be replaced context-dependent **with control/device/controller boards/hardware**.

The Module DLLs make so-called real-time requests to the operating system (How quickly is the operating system able to finish or interrupt an actual task with low priority and to serve the request(s) with higher priority ?).

If there are **no** Module DLLs activated in ProNc, ProNC can be started under all Windows operating systems(98 / 2000 / NT / XP).

More: 3.4.2 Principle of Module DLLs

According to the ***motion control DLLs*** can be generalized:

Order-No. ProNC	PC-operating system	Target controller/ number of axes	Continuous path mode (look ahead)
Z11-333111	Win98 NT4.x Win2000 WinXP	IMC4 / 4 axes	-
Z11-333112	Win98 NT4.x Win2000 WinXP	C116/4 or C142/4 with interface card I5 / 3 axes	-
Z11-333121	Win98 NT4.x Win2000 WinXP	IMS6 / 6 axes IML4 / 4 axes	-
Z11-333122	Win98 NT4.x Win2000 WinXP	IMS6 / 6 axes IML4 / 4 axes	**x**
Z11-333211	Win98	UPMV4/12 / 4 axes	-
Z11-333212	Win98	UPMV4/12 / 4 axes	**x**
Z11-333312	NT4.x Win2000 WinXP	CVC496 / 4 axes (with CAN-Interface) 6 axes and 2 handling axes, if 2 CVC496 will be connected	**x**

1.5.2 Functionalities



ProNC involves an operating surface according to the SAA-standard and a programming platform to

- Edit / debug / process of **DIN**- / ISO-user programs (textually programming by **DIN** 66025), **PAL**- user programs (textually programming by isel-**PAL**-syntax)

- Processing / interpretation of user programs, produced by the isy-CAD/CAM system as result of the postprocessor run (**NC**-**P**ostprocessor format **NCP**)

like extensive dialog software

- to **configuration / parameter setting / diagnosis** of numerical axes / equipments inclusive of the necessary periphery:
  - Input and output modules
  - Converter controlled spindles
  - Tool changer

The operator interaction is efficiently supported by a mouse or with a trackball resp. function keys.

1.5.3 Use of ProNC



The application fields of the program package ProNC contain the following **isel-systems**:

- **CNC-machines as XYZA-portal machines** with maximum **1** infeed axis (A):
  - **with stepper motors** (with Controller IMC4):
    - CPM 2018
    - CPM 3020
    - CPM 4030
    - GFM 4433
  - **with DC Servomotors** (with Servo Controller Card UPMV4/12 or Controller CVC496 with CAN-Interface):
    - CPV 4030
    - GFV 4433
    - GFV 4473
    - GFV 68/52, GFV 102/62, GFV 142/102, GFV 142/232 without cover
    - GFV 48/52-SW, GFV 102/72-SW, GFV102/112-SW, GFV 142/112-SW, GFV 142/162-SW, GFV 142/252-SW with cover
- **Robot and Handling systems** with Cartesian cinematic (at least 2 / max. 4 axes)
  - Cross table 1 (X: Linear Guide LF 5; Y: Linear Guide LF 5)
  - Cross table 2 (X: Linear Guide LF 5; Y: Linear Guide LF 4)
  - 2-Axis-H-Construction consisting of Blue-Line-Axis with Drive Belt
  - 3-Axis-Lifting Arm with optional rotatory axis
  - Lifting Arm with optional rotatory axis
  - Handling system (max. 6 axes)
    - Use of 1 x CVC 496 with 4 DC amplifiers and 1 x CVC 496 with 2 DC amplifiers

1.5.4 Programming support



The software product ProNC is an operating- **and** programming system. Implementing ProNC a special attention was attached to an efficient support of textually programming. Being part of it:

- an efficient **editor**, supporting the Rich Text Format (RTF)

  So you can also work with an editor, which supports the format of the source code files (*.PAL or *.ISO).

This means reversed: Programs in RTF format created outside of ProNC can be continued with the editor in ProNC.

More: 3.1 The editor

- a **dialog-oriented generation of commands** (straight line, circle, rectangle, polygon and others), programming constructions (program branch, loops, subprograms and others) at any place in the current user program

  More: 5.4 Menu Insert

- a quick **2-Phase-Compiler**

  More: 3.2 The compiler

The efficiency of the programming system is determined by the available language constructions (grammar). This grammar is documented in the manual ProNC Programming Instruction.

Using the textually programming method the following typical language constructions are available:

- Subprogram technology
- Constructions to control the programming process (branches and loops)
- Arithmetical and Boolean expressions
- Real and trigonometrical functions
- Using of variables (real-, process- and frame-variables)
- Operator interaction during the run time of user program and others

This programming comfort requires a NC Compiler with the tasks of a syntactical analysis of the ISO/PAL source program and the generation of the CNC target program as input file for the CNC-interpreter.

1.5.5 Implementation



To be able to use ProNC efficient for implementation / test of user programs, the following debug features were implemented:

- Single step mode
- Set up / Teach-In as well as correction and test of arbitrary machine positions in two axes systems
- Configurable system monitor to display relevant real- and process-variables
- Display window for velocity and actual coordinates in two axis systems
- Display of the state of the motion control(s) for two axis systems (incl. hard- and software limit switches)
- Independent spindle operator panel(s) for maximum four main spindle(s) with revolution override
- Independent machine operator panel for one or two axis systems with motion override, manually set / delete of work piece zero point
- Definition of break points to arbitrary program lines in the user program
- Manipulation of real-variables at run time

Teach-In and manual axis motion at the run time of the user program

1.6 Screen layout

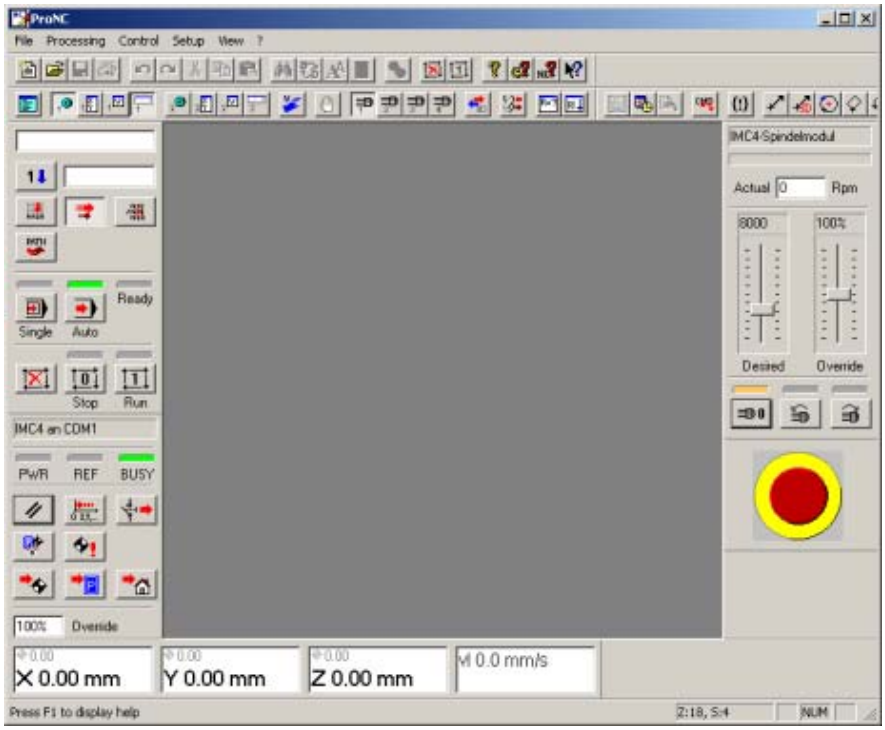
Start screen:

Starting the program **ProNC** an **operating surface** is offered to the user containing all major windows for the equipment setup and the processing of the user program.
According to the Windows philosophy all windows can be removed and placed on screen **arbitrary**.

More: 5.9 Menu View

Depending on the importance of the respective contents for the current processing the user can decide himself about the arrangement of the windows.
To arrange the windows please click with the mouse into such a window and move it with the mouse to the desired position on screen.

The arrangement of operating elements in groups of **thematic associated functions is user friendly**.
Only those menus and buttons are active usable in the respective program state.



START SCREEN **WITHOUT** AN ACTIVE USER PROGRAM

Meaning and contents:

- The symbol bars contain **buttons** to **process the source file** and **operating fields** to **set up the** working environment at processing.



- The windows have the state of an **operating panel.** Using as display windows they have **control functions** during the setting and processing of the user program.

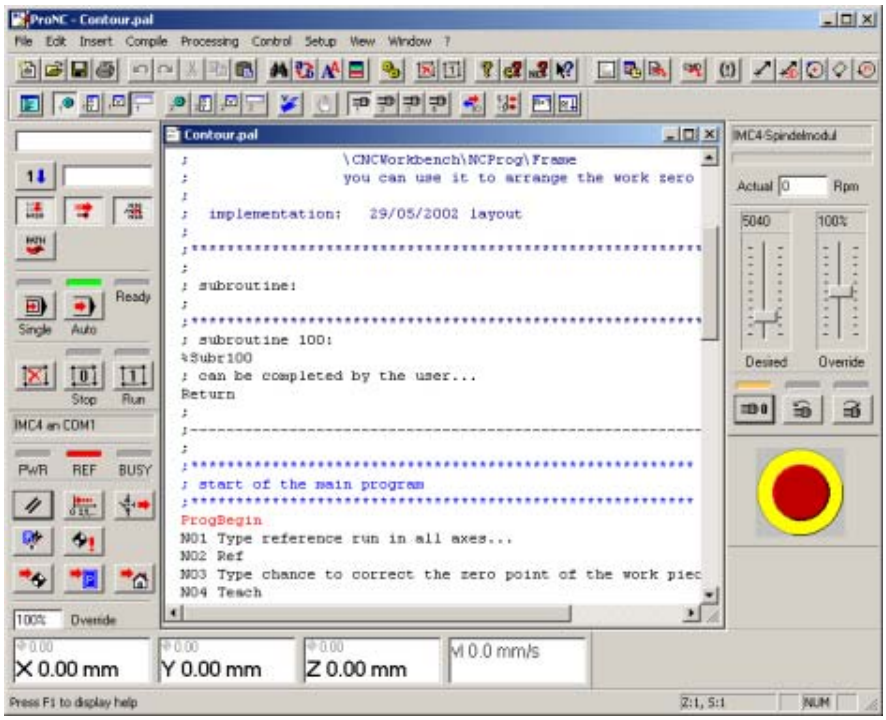
- Each function deposited on the symbol bar or within dialog windows

is also available **within** the **menu bar**.

- The **section** in the **middle of the operating surface** is reserved to edit the source file. Restarting the program ProNC this area is empty if the option to open the last used file automatically is deactivated.

- Finishing the program the last adjusted **window layout** is stored. At next program start the layout will be restored.

After opening a user program e. g. „contour.pal“ out of the directory \CNCworkbench\NCProg\Pal\Sample this program is shown in the editing area.



SCREEN WITH OPEND USER PROGRAM „**CONTOUR.PAL**“

1.6.1 Menu bars, symbol bars, state bars

The Main menu:



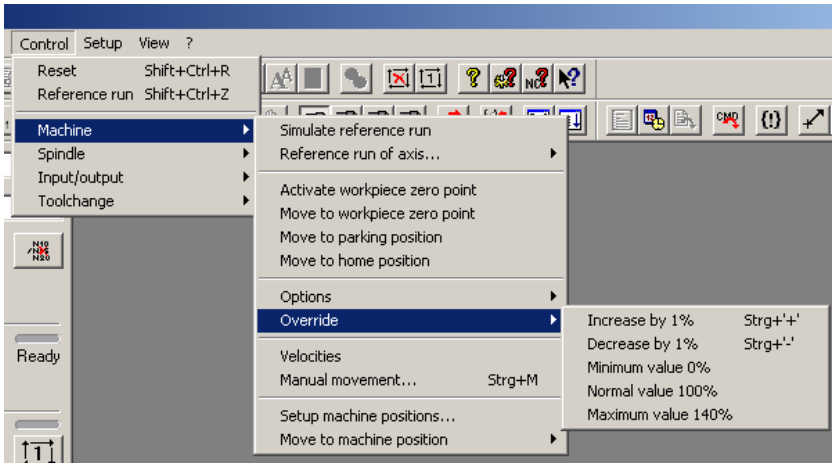


The **menu bar** is the basic to reach all functions of the program **ProNC**. **Submenus** are located behind of each menu. These submenus can contain activities again.

The menu bar offers a good possibility, to get an overview about all offered services of ProNC.

Example: Menu **Control - Machine - Override**

Modification of the axis velocity during the run time of the program



Edit file:



According to the Windows standards the **tool bar** "File" contain all essential functions to edit file like **New**, **Open**, **Edit**, **Save**, **Print** inclusive **Search, Copy, Insert** and **Format.**
Using the well-known help symbol **„?“** you can call for information of the program and also for an overview of the used control environment.

Insert of program parts:



The **tool bar „Insert“** contains functions, which offer some possibilities for the user, when editing a loaded source file, to use prefabricated program constructions and to complete them with user specific parameters. It is possible to assume **program loops**, **program branches** and **commands**, the **definition of a circle motion** or a **drilling cycle** in the source file. Thereby changes in the source file will be possible fast and simply.

More: 5.4 Menu Insert

Display of help information in the status bar:

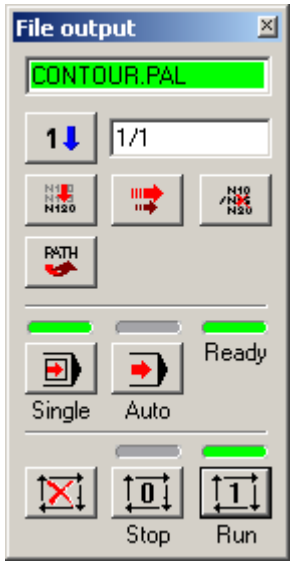


In this line a short description appears showing the current chosen function (select with mouse by menu or button).
Editing a source file the actual **position** of the cursor can be read after line and column.

1.6.2 Standard windows

The five standard windows in ProNC display all relevant parameters at processing of the user file inclusive of all machine states.
According to the configuration of the equipment there available e. g. two „Machine“-windows for 2 axis systems and/or four „Spindle“-windows for 4 main spindles …

Control panel Processing:



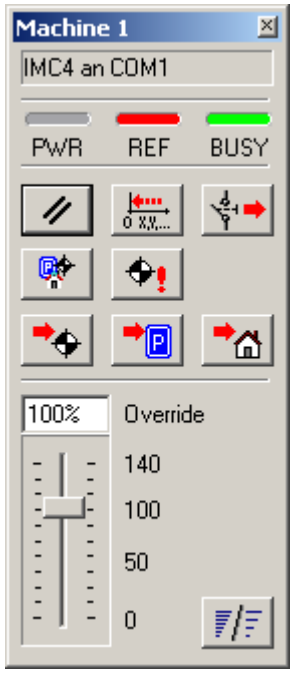
→ Program repetition

→ Set skip
→ Rapid traverse
→ Program skip
→ Online path calculation

→ Single mode
→ Automatic mode

→ Abort
→ Stop
→ Start

Control panel Machine:



→ **PWR**: green --> power on
→ **REF**: reed --> no reference run executed
→ **BUSY**: green --> machine is busy

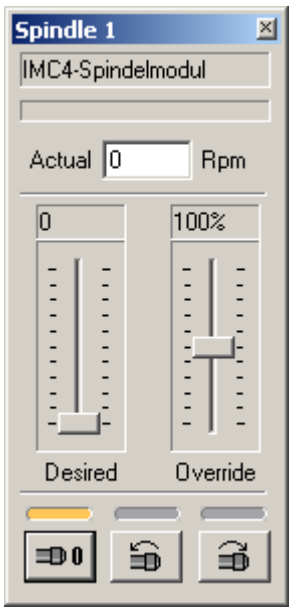
→ Reset
→ Reference run
→ Move axes out of limit switch

→ Set up machine positions
→ Activate/deactivate work piece zero point
→ Move to work piece zero point
→ Move to park position
→ Move to home position

→ Velocity override

Control panel Spindle:





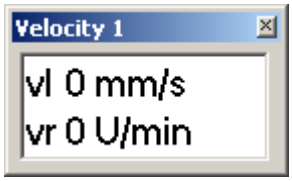
→ Current number of revolutions per minute

→ Slide control to correction of the spindle speed

→ Spindle off
→ Spindle start counterclockwise
→ Spindle start clockwise

Velocity:

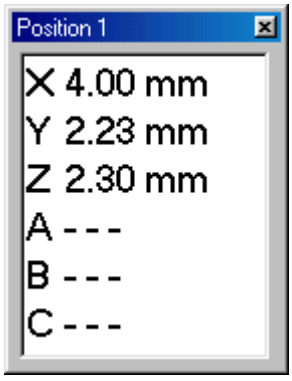




Display the **current feed rate** (current path velocity during a linear interpolation, a circle interpolation or an path processing)
This velocity is the velocity at the top of tool, it isn't the velocity of single axes.

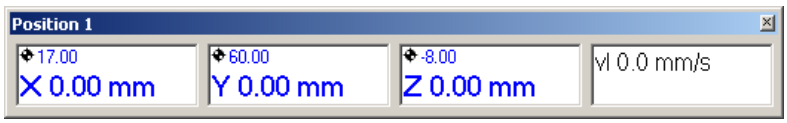
Current axis position:





Positioning indicator of the possible 6 axes e. g.
- Set up work piece zero point with manual movement of axes (teach)
- Output user file

Axis position/ velocity:



Combined display axis-position/velocity with display of work piece zero point (activated work piece zero point = blue display)

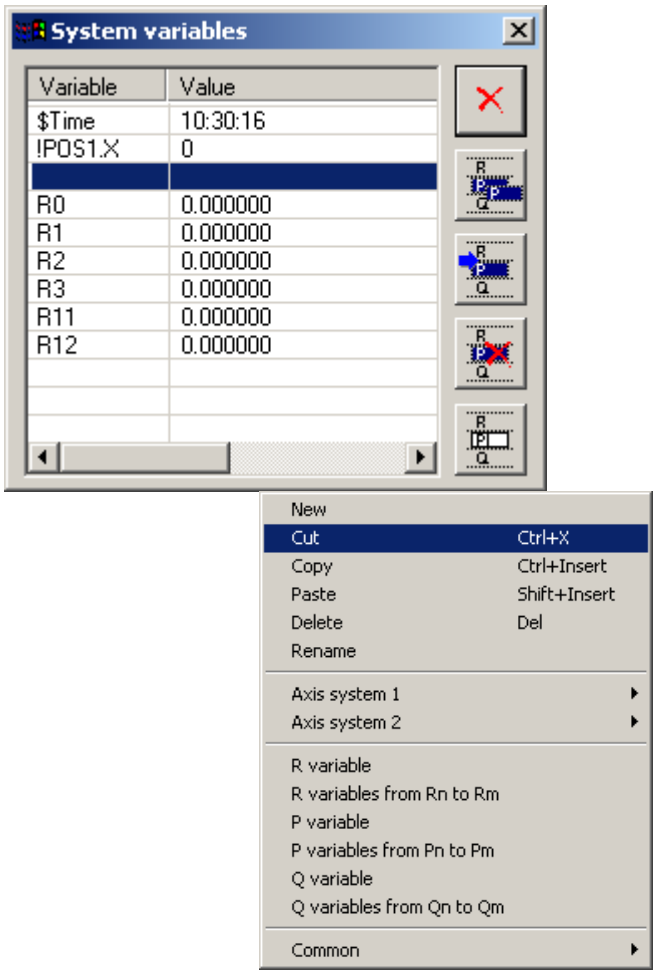
1.6.3 Additional windows

In addition to the „Standard windows“ the windows **system monitor**, **status** and **status security circuit** can be activated to get special information.

Monitoring of ProNC-system variable:

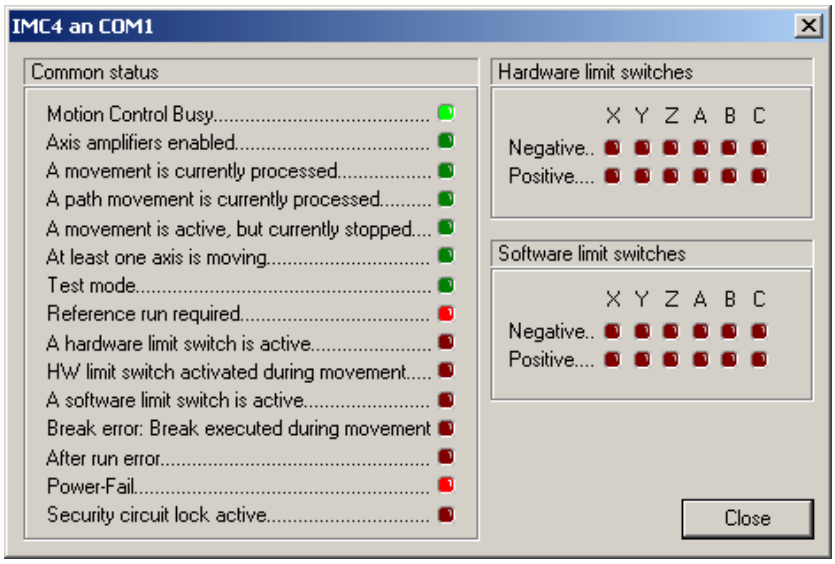
Intending to monitor the contents of variables during processing of a CNC program (prior in the phase of program tests), the operator can activate this window.

Please move the cursor in the window "System monitor" and press the right mouse key. A **context menu** will be offered and you can chose your own special **selection** of the variables which shall be displayed.



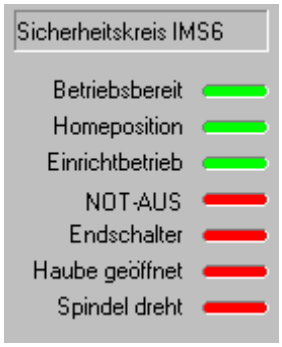
Machine status:

In certain program situations or if you get an error message it will be necessary to get an overview about the state of the plant.
Information about the motion control, reference run, „is a limit switch reached ?“ are given in this window at any time.
In error state or in the test phase you can display these windows on the screen permanently.



Security circuit-module (SC):

Using a security circuit-module (**SC module**) as attachment to the security emergency circuit of CNC machines, please use the following window to check the equipment state.



→ Security circuit module on
→ Prominent position (supervised by sensor)
→ The state of operator mode switch is "Set up"
→ At least one hardware limit switch is activated
→ Cover open
→ Spindle on

Hints:

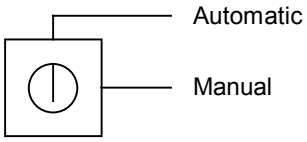


① isel-CNC machines and equipments are controlled in the mode

- Manual (hand operated)
  or
- Automatic (program controlled)

The change of the operating mode is possible:

- On the machine operator panel by turning the key switch



- By operating with PC keyboard respectively mouse/trackball with click on the corresponding buttons
  Manual drive  or **Machine positions manage** **.**

② In operating mode "Automatic" (program controlled) the security circuit module ensures that the cover (if existing) can only be opened in the prominent position (HOMEPOSITION), if the main spindle doesn't turn.

③ In operating mode "Manual" (hand operated) the numerical axes of a CNC machine / equipment can be moved with opened cover in tip mode. Tip mode means that an axis moves only as long as a key is pressed.

2 The files

2.1 The initialization files

2.1.1 ProNc.ini



The initialization file **pronc.ini** in the directory \CNCWorkbench\Bin contains parameters, that are predefined as standard parameters at first start of the plant.
Examples for such parameters are the order of movement of the axes when doing the reference run at start or end of the processing, the spindle speed, directory entries, screen layout.

If changes of the parameters happen during the processing, the **new parameters** are **stored**, when leaving the program. At ProNC restart the last status will be restored.

2.1.2 NcCtrl.ini



The initialization file **ncctrl.ini** contains all parameters for the devices, configured within the Menu **Setup - Control** e. g. motion control module, I/O module, spindle.

Within this **configuration dialog** the user must enter the information (description) of the module and select directory- and name character strings of the concrete Module DLL resp. its initialization file. These parameters will be written into the initialization file **ncctrl.ini** in the directory \CNCworkbench\Bin resp. they will be read out of this file at a new (re-) configuration.

2.2 The user files

ProNC-User files: Source file and geometry file



The ProNC user files surround the files:
- Source files with the extension .pal or .iso, created with a text editor
- NCP files, converted to source files with the extension .pal

The geometry files created within ProNC by Teach-In are also a part of the user files.

More:
5.7.3.10 Menu **Control - Machine - Setup machine positions**

There are two types of files:

- the source file with the extension **.iso** for DIN/ISO user programs resp. the extension **.pal** for PAL user programs

- the geometry file with the extension **.fra** stores the geometry information, assigned to a user program.

2.2.1 The source file

Textual programming: In the program system ProNC the user can create his programs (e.g. handling- or processing programs), after it was found an algorithm to realize this task according to the technological formulation. The found algorithm will be converted into the program text structure (**sequence, loop, branch**). This way of programming will be described as **textually programming** (well-known by common programming languages of EDP like e.g. FORTRAN, PASCAL or C).

Edit of existing files: Programs created with **Pro-DIN** or **Pro-PAL** can be implicitly loaded in ProNC and processed in the corresponding syntax (ISO or PAL).



Additionally ProNC offers a service, which primarily offers new possibilities for a program modification for user of **ISY.**
A program created with **ISY** with the extension ***.ncp** can be imported and converted into a **PAL source file**.
This file can be extended **functionally**, e. g. you can insert a program loop.

If no changes shall be carried out, the program can be immediately started after a translation run.

Grammar: Creating the program text certain rules must be observed. These rules are described generally as a grammar.

Grammar of ProNC: The **grammar** defines **syntax** (which connection of words produces a valid set ?) and **semantics** (what means this set ?) for the programming language. For ProNC the valid grammar is described in detail and explained with examples in the Programming instruction.

Compiler as 2-Phase-Compiler: The source file is the only one file which is processed by the compiler. All necessary program information must exist in this file.
Implementing of ProNC it didn't seem meaningful to produce one target file out of several source files by the compiler. This causes a very high organisation overhead. The available subprogram technology enables program sections, existing and already tested in other source programs to integrate in the actual source file with help of the Menu **Insert - File**, to declare it as subprogram and then to compile it within a uniform, complex source program.



Example for a syntactical errorless source file: **contour.pal**
after PAL-syntax

```
;*************************************************************************************************
;
;
; Project:    Test program for XYZ-Milling Machines ->
;             simple milling contour in the XY-plane
;
```

```
;  Editor:     Joachim Görner, isel-automation KG
;
;  Hint:       1. to test this program a Motion Control-Dll is required for the axis system 1
;              2. to test this program a Spindle Control-Dll is required for the milling spindle 1
;              3. the geometry file contour.fra in the directory CNCworkbench\NCProg\Frame
;                 can be used to setup the work piece zero point WSNULLPUNKT
;
; created:   11/01/2001 Layout
;
;*************************************************************************************************************
;
; Subprograms:
;
;*************************************************************************************************************
; Subprogram 100:
%Subr100
; the user can complete ...
Return
;
;-----------------------------------------------------------------
;
;*********************************************************
; Start of the main program
;*********************************************************
ProgBegin
N01 Type Reference run in all axes ...
N02 Ref
N03 Type Possibility to correct the work piece zero point ...
N04 Teach
N05 Q1 = WSNP
;
; drive the contour 1000 x :
For R100=1,1000,1
  N5 R1=MessageBox YESNO "Shall the program run be started ?"
  If R1==IDNO
    Type Cancel the program...
    Abort
  Else
    TYPE Processing is started ...
..Endif
  N10  Type Move to work piece zero point in X and Y
  N15  R1 = Q1:X R2 = Q1:Y R3 = Q1:Z
  N20  FastAbs XR1 YR2 FastVel 40
  N25  Type spindle (5000 revolutions per minute, clockwise) on ...
  N30  S5000 SCLW
  N35  Type Sink to the surface of the work piece ...
  N40  FastAbs ZR3
; define a new work piece zero point:
  N45  WpZero
  N50  Type Move fast to left corner with the coordinates (20,15,2) ...
  N55  FastAbs X20 Y15 Z2
; drill in the material with a depth 4 mm :
  N60  MoveAbs Z-4
  N65  Type Drill contour with a feed 25 mm/sec
  N70  MoveAbs Y60 F25.0
  N75  ClwAbs  X45 Y85 I25 J0
  N80  MoveAbs X140
  N85  MoveAbs X180 Y45
```

```
  N90  MoveAbs Y15
  N95  MoveAbs X106
  N100 Y40
  N105 CclwAbs X70 Y40 I-18 J0
  N110 MoveAbs Y15
  N115 MoveAbs X20
; clear work piece zero point:
  N120 WpClear
; raise the Z-axis:
  N125 FastAbs Z0
; approach the machine zero point:
  N130 FastAbs X0 Y0
  N130 Type spindle off...
  N135 Soff
  N140 Type processing ends... Please clamp off!
EndFor
;
N06 Type Program end is reached, please clamp off the work piece ...
;
ProgEnd
;
;*********************************************************
; End of main program
;*********************************************************
```

2.2.2 The geometry file

2.2.2.1 Definition

Importance of geometry file:

The geometry file contains geometry data stored as **frames**. These geometry data represent the current coordinate values at most of six axes (p*rogramming in joint coordinates*) in the simplest case.
For the user the geometry file offers the possibility to define coordinates of the work space within the work program independently of the technological instructions.



The user can manipulate the parameters in the window **Machine positions** (Menu **Control-Machine-Setup machine positions**).

The user does not have to stress himself with the following internal structure of the geometry file.
It is important that **all coordinate values** as **real numbers** have their fixed place **within the complex geometry data structure** and that the user can access both the complete structure and all their components within the user program during run time.

More: Programming Instruction:
Q-Variable
R-Variable

2.2.2.2 Structure of the geometry file

Declarations to the frame:

To the identification of the geometry file this file gets the extension ".fra" in its file name.

Each frame contains **3 „Head lines“** :



- **Name of frame**

Each frame within a geometry file has a name, e.g. PPOINT1 or PALPOS (**PALPOS** for **PAL**ette **POS**ition).
Every frame name may appear within a geometry file only once.
Names of frames within a geometry file consist of at least four capital letters, followed by numbers, the underscore "_" and further capital letters (all in all at most 20 signs).
The number of frames isn't limited in a geometry file.

Coordinate values::

The stored information within the frames are the coordinate values of the single axes of a numeric controlled device (e.g. Cartesian plant / portal or industry robot).

The unit of the coordinate statement of the X-, Y- and Z-axis is always **micrometer [µm]** in the geometry file.

The unit of the position of the rotatory axis as entrainment axes is always **arcsec ["]**.

The details in the (3.3) rotation matrix are relevant only from a 5-axis system and describe the correct position of the point in the room.
The elements of the rotation matrix R00[1], R11[1], R22[1] are initialized with the value 1.000000 (unit matrix).
The other positions of the matrix are only reported, if their value is non-zero.

The following table reports the structure of the geometry file in its maximal configuration:

General instructions		
[POINT 1]		frame name
MoveOrder		sequence of axis motion
Usage		special use
Axis system 1	Axis system 2	Meaning
R00[1]	R00[2]	element of the rotation matrix [0,0]
R01[1]	R01[2]	element of the rotation matrix [0,1]
R02[1]	R02[2]	element of the rotation matrix [0,2]
R10[1]	R10[2]	element of the rotation matrix [1,0]
R11[1]	R11[2]	element of the rotation matrix [1,1]
R12[1]	R12[2]	element of the rotation matrix [1,2]
R20[1]	R20[2]	element of the rotation matrix [2,0]
R21[1]	R21[2]	element of the rotation matrix [2,1]
R22[1]	R22[2]	element of the rotation matrix [2,2]
AxX[1]	AxX[2]	coordinate of the X-axis in [µm]
AxY[1]	AxY[2]	coordinate of the Y-axis in [µm]
AxZ[1]	AxZ[2]	coordinate of the Z-axis in [µm]
AxA[1]	AxA[2]	position of the 1. rotation axis as entrainment axis in ["]
AxB[1]	AxB[2]	position of the 2. rotation axis as entrainment axis in ["]
AxC[1]	AxC[2]	position of the 3. rotation axis as entrainment axis in ["]
ANG[1]	ANG[2]	gripper close angle
FRC[1]	FRC[2]	gripper force
STA[1]	STA[2]	gripper state

Coordinates at an *isel*-flatbed plant:

At simple kinematics, e.g. an ***isel*-**flatbed plant, the Cartesian coordinates X, Y and Z as well as the possible fourth axis A are stored into the complex geometry data structure (frame) as follows:

Example to store coordinate values in the frame structure:

An equipment contains four axes and one Cartesian kinematics. The actual indication appears the current values of the axes, e.g.:



X axis: **100** mm = 100000 µm
Y axis: **200** mm = 200000 µm
Z axis: **300** mm = 300000 µm
A axis: **45** grad = 162000 "

It is one axis system (axis system 1), i.e. the values for the axis system 2 are 0.
These values are stored within the frame data structure and are written common with a name **Point1** defined by user into the geometry file e.g.:

The description of this point has the following appearance in the geometry file:

```
[Point1]
MoveOrder=X,Y,Z,AUTO
Usage=0
R00[1]=1.000000
R11[1]=1.000000
R22[1]=1.000000
AxX[1]=100000
AxY[1]=200000
AxZ[1]=300000
AxA[1]=162000
AxB[1]=0
AxC[1]=0
ANG[1]=0.000000
FRC[1]=0.000000
STA[1]=0
R00[2]=1.000000
R11[2]=1.000000
R22[2]=1.000000
AxX[2]=0
AxY[2]=0
AxZ[2]=0
AxA[2]=0
AxB[2]=0
AxC[2]=0
ANG[2]=0.000000
FRC[2]=0.000000
STA[2]=0
```

Frame structure:

At first view this structure seems too expensive to store only four coordinate values. However, it is pointed out that for more complicated kinematics, e.g. 6-axis systems, the frame description has gained acceptance internationally to a clear description of the orientation.

2.2.2.3 Construction of geometry file

Define or update of frames:

The **definition** of frames in the geometry file carries out within ProNC and the Menu **Control** - **Machine**- **Setup machine positions** .

The coordinate values of axes are manipulated during **Teach-In** or by **keyboard input**. These values can be arbitrarily often updated.



Teach-In means, that the current coordinate values are new entered respectively updated into a geometry file within an exact described structure (the f*rame structure*). These information are available for an user program interpretation at all times and they can be used as target coordinates for motion commands.

A geometry file can be stored with the same name related to the user program and with the extension ***.fra**.

If you don't want a **fixed definition** of the frames, the **coordinate instruction** are available according to the values in the window „**MACHINE POSITIONS**“ (activated about the Menu **Control - Machine -** Setup machine positions). These data are stored in the standard geometry file **StdFrame.fra** in the directory /CNCworkbench/Bin.

Starting a CNC program, at first it will be looked for the existence of a user geometry file in the directory CNCWorkbench/NCProg/Frame. If it isn't found, the data of the standard geometry file StdFrame.fra are used.

Definition of frames with a text editor:



The geometry file is a text file. Therefore it can also be edited with a text editor by the user. Strictly you should take care that the described frame structure will be kept.
For the definition of the axis coordinates the dialog "Setup machine positions" has to be used preferably.

Advantage of the frame concept:



The advantage of the concept of the geometry file consists in the fact that there is a clear separation between coordinates of the workspace of one or two plants and the technological instructions within a user program.
If corrections of positions are necessary, only the geometry file is corrected, the user program remains unchanged.

More:
2.2.2.2 Structure of the geometry file

Relation between frame and geometry variable:

The access to the information in a frame structure carries out during the run time of the user program with help of geometry variables (Q-variables). To initialize Q-variables you have to enter the name of a frame in the current opened geometry file, e.g. Q1=PARKPOSITION.



The program of a movement to a target position described by the coordinates in a frame, carries out with help of the commands **G10** | **FASTFRAME** and **G11** | **MOVEFRAME.**

More:
Programming Instruction: Fast velocity with statement of frame
Programming Instruction: Processing velocity with statement of frame

2.3 Files generated from the compiler

Files after the compiler run:

The Compiler generates three files from the current source file, the



- CNC target file
- **label file (label table)**
- **error file**

Therefore all these three files get the same name supplemented with the following typical extensions:

Example: source file/user file: contour.pal

- the CNC target file: contour.cnc
- the label file: contour.ltb
- the error file: contour.err

2.3.1 The CNC target file

CNC target file:



The CNC target file is produced by the compiler out of the source file. Therefore the CNC target file can be described as one of three output files of the compiler while the source file represents the input file of the compiler.
The absolutely necessary program text from the source file is included in the CNC target file.
All comments are missing in the CNC target file.

Run time error:

!

The data in the CNC target file must not be changed by the user because otherwise unpredictable run time errors can happen during the program processing .

More:
4.1 Data flow and interface

2.3.2 The label file

Contents of the label file:

The label file (label table) isn't relevant for the user. It contains merely the absolute addresses within the CNC target file which are jumped according to the goto-instructions at program **branches .**

Purpose of label file:

The label file is an important "bridge" between compiler and CNC interpreter.
The CNC interpreter evaluates the complex program constructions during the interpretation of the CNC target program and it always finds with help of the data in the label table the right places in the CNC target program where the program interpretation can be continued.

Run time error:

!

The data in the label table may not be changed by the user since otherwise during the program processing dangerous run time errors can happen.

More:
4.1 Data flow and interface

2.3.3 The error file

Error file:



The compiler writes all errors recognized during a compiler run into the error file. Three types of errors are differed:

- syntactical errors
- errors at program constructions
- file error

For the user these errors are immediately listed in an error protocol with detail of the line number and possible error notes after the compiler run. The compiler informs at syntax errors which text construction is expected.

More:
5.5 Menu Compile

2.4 Files created by the NC interpreter

2.4.1 The report file

Report file:



The **report file** is not produced by the compiler, but by the **CNC interpreter**.
It contains information, if run times errors have appeared during the interpretation of the CNC target program.

Run time error:

Such run time errors happen then, if file read errors (e.g. in the CNC target file or in the label file) appear or if information has been manipulated in the mentioned files.

More:
4.1 Data flow and interface

2.4.2 The variable file

Variable file:



The **management** of the variable file is done by the **interpreter**.
It has the ability to write the current values of the R variables R0 to R299 into the special variable file "*.var" if the application program ProNC is complete or aborted.

At the next start these **data** are read out of the variable file, converted and stored **index-right** in the working memory**.**

***Hint*:**

Storing the values of R variables R0 to R299 to the *.var file is performed at normal **program end** and **defined program abort**.

In the Menu **Setup - Interpreter** you have the possibility to deactivate the loading and initialization of R variables R0 to R299 with values from the *.var file before starting the application program.

More:
1.6.3 Additional windows
4.1 Data flow and interface
5.8.4 Menu Setup - Interpreter

3 Software components of ProNC

3.1 The editor

Aim and objective of ProNC editor:

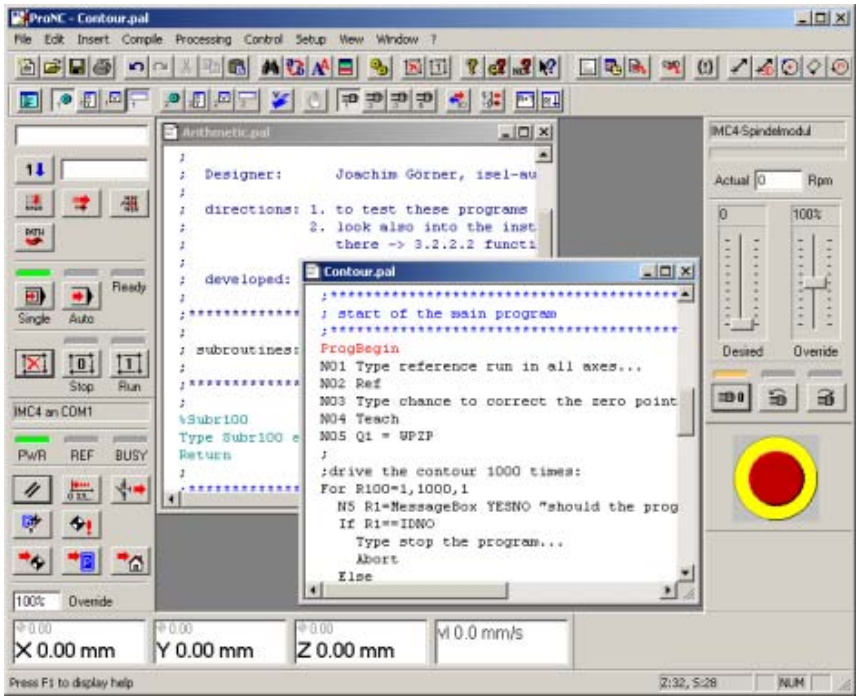


The editor was integrated in the software to give the operator the possibility to carry out corrections and completions without leaving the operating surface.
Obtainable about the symbol bars respectively the menus **File , Edit , Insert** all functions are implemented for a comfortable editing of the source program.
With the **formatting function** now it is also possible, to indicate visually selected program passes with determine colours or scripts.

Editing of source files:



The use of the editor within ProNC permits the processing of ***several source programs.***
In dependence of the active window, all processing actions will be effectively for this file. The same principle applies to a following compiling as a precondition for the processing of the program. The respectively **active file** is compiled. The activation is carried out by a simply clicking with the mouse in the desired program (window).
The setting, how the loaded programs can be shown you can determine with help of the Menu **Window** e.g. complete picture modus or side by side or overlapped …
Several simultaneously loaded user programs allow furthermore an easy overtake of program passes from one file into another.



SCREEN: TWO OPENED USER PROGRAMS „ARITHMETIC“ AND „CONTOUR“ WITH ACTIVATED FILE „**CONTOUR**“

Editing functions: The possible editing functions are important to the user:



1. Positioning of the cursor in text
2. Editing function, e.g. insert or delete character
3. Copy, Insert, Search, Replace

These editing functions are explained in the following sections.

3.1.1 Positioning of the cursor in text

Notation of key combination: Buttons which have to be pressed to perform an editing action are included in pointed brackets:



For example the notation **<←>** means, the corresponding key must be pressed one time.

Key combinations are linked together with a plus sign and included in pointed brackets too.

For example the notation **<Strg+Pos1>** means, the key **<Strg>** and the key **<Pos1> must** be pressed at the same time.

Key(s)	Function	Hint
<←> or **<→>**	cursor control (text line)	one column to left or one column to right
<↑> or **<↓>**	cursor control (text column)	one line to left or one line to right
<Strg+→>	cursor control (word in the text line)	cursor beginning of the next word
<Strg+←>	cursor control (word in the text line)	cursor to beginning of the preceding word
<Pos1>	cursor control (text line)	cursor to line start
<End>	cursor control (text line)	cursor to line end
<Page↑>	browse	page up
<Page↓>	browse	page down

<Strg+Pos1>	cursor control (text)	cursor to beginning of text
<Strg+End>	cursor control (text)	cursor to beginning of text

Key(combination) to realize the cursor motion and cursor positioning in text

3.1.2 Editing functions

Key(s)	Function	Hint
<⇧+→>	mark several characters with key function	move the cursor to beginning of the area, you will mark, press and hold the shift key **<⇧>,** move with key **<→>** to the end of the wished area
	mark arbitrary many characters/lines/blocks with the mouse	mark signs with the mouse by clicking with left mouse key at the beginning of the area you will mark, press and hold the left mouse key, drag until the end of the area, let off the mouse key
<BACKSTEP>	delete characters	the character, which stands on the left of the cursor
<del>	delete one or more characters/lines	the sign(s)/line(s), which is(are) marked by the cursor

Key/mouse (combinations) to realize editing functions

3.1.3 Block operation copy, insert

☞

A **block** is a connected area, perhaps extending over **several characters.** The first and the last character of a block can be at arbitrary place, within a line. A block consists of two characters standing side by side in the simplest case.
If a block is "pulled" over several lines, the complete line is always integrated in the block.

Possible block operations:

To carry out one of the possible block operations

- Move a block
- Copy a block

the block must be marked. This happens with the keys **<⇧+→>** or by drag with the mouse over the wished text passage.

More: 3.1.2 Editing functions

Key(s)	Button	Menu	Function	Hint
<Strg+A>		Edit	**Mark All**	the whole text (file) is marked
<Strg+C>		Edit	**Copy**	the whole block is copied from the current position
<Strg+V>		Edit	**Insert**	the first character of the marked block is scrolled to the current cursor position, all further characters of the marked block refill to right

Operating devices to realization block operations

3.1.4 Functions Search, Replace



Within the editor the comfortable functions Search and Replace can also be used for a fast and sure editing of your application program in ProNC.

Key(s)	Button	Menu	Function	Hint
<Strg+F>		Edit	**Search**	a dialog box appears, input of the character string, you look for, start of search with **<Enter ↵>**
<Strg+H>	-	Edit	**Search/ Replace**	a dialog box appears, input of the character(s) you look for and input of the new character(s); start of replacing with **<Enter ↵>**

Operating device to a fast searching and/or replacing

3.1.5 Other functions

Key(s)	Button	Menu	Function	Hint
<Strg+Z>		Edit	**Undo**	the last action(s) is(are) in the file is(are) cancelled
<Strg+Y>		Edit	**Restore**	cancelled actions are restored
-		Edit	**Font**	the marked character(s) is(are) formatted
-		Edit	**Font**	the marked character(s) get(s) the chosen color

Operating elements to cancel, restore and format

3.1.6 File functions

Key(s)	Button	Menu	Function	Hint
<Strg+N>		File	**N**ew	the screen area for file editing is deleted
<Strg+O>		File	**O**pen	a user program is loaded to processing or to editing
<Strg+S>		File	**S**ave	corrections/changes are saved
<Strg+P>		File	**P**rint	the file or parts of the file are printed

Operating elements to create, open, save, print of user programs

3.2 The compiler

Task of the compiler:



The **input file** for the compiler run is always the **source file**. Thereby it is without importance, if the user has decided to program in **ISO/DIN format** or **PAL format**. The adjustment of the current syntax (ISO or PAL) must be carried out in the Menu **5.8.5 Setup - Compiler** - **Common**.

The compiler creates the CNC target file, the error file and the label file out of the source file during a compiler run.



The following tasks are realized during the compiler run:

Task	**Meaning (among others):**
Lexical analysis	There is a false/forbidden character in the character string ?
Syntactical analysis	Was the language syntax (grammar) violated? Were subprograms called without they were declared?
Analysis of nesting of **program constructions**	Was the nesting depth exceeded for program constructions ?

CNC target file:

The CNC target file includes the relevant **program text** from the **source file.**
In addition to the text from the source program the compiler writes corresponding **goto-**instructions or marks (labels) in the CNC target file. Because of the use from complex instructions for control of the program flow it is necessary to support the realizing program branches at the interpretation of the CNC target program by the CNC interpreter.

Start the compiler:

The compiler is started with the Menu **Compile - Compiling** of **name.iso** *or* **name.pal**.

A **successful** compilation finishes without error message.
In the other case a window appears with an error message and you can locate and correct the errors.

3.3 The CNC interpreter

The CNC interpreter:

The CNC interpreter immediately determines the productivity of the operating and programming system ProNC.
While the complete available functionality of ProNC is **specified** on a user level as program text, the CNC interpreter must immediately execute all functionalities at the run time of the user program or delegate the actions to other modules (e.g. to the motion module).

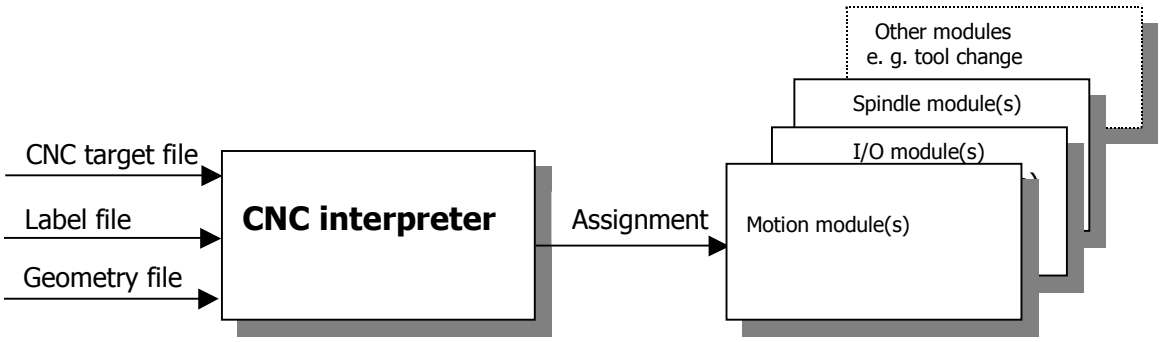
Main task of CNC interpreter:

The major task of the CNC interpreter consists in the interpretation of the CNC target program.
This interpretation of the CNC target program includes the essential tasks:

- Execution of all program branches by use of the information in the label tables
- Realization of the subprogram technology by use of a stack for push/pop the return addresses
- Handing over of motion tasks to the motion module and synchronization with this module, to ensure the time reference if a motion is carried out
- Calculation of all arithmetical expressions (parameter calculation)
- Realization of the access to geometry files and initialization of geometry variables
- Realization of Boolean operations
- Execution of all trigonometric and real functions

Input files of CNC interpreter:

The CNC interpreter processes as input files the CNC target file, the label file and the geometry file. It immediately engages the motion module with help of calls to DLL functions.



3.4 Configuration

3.4.1 File structure CNCworkbench

Structure of CNCworkbench:

CNCworkbench is the directory name of the ***isel***-control software under the operating system Windows. The most important subdirectories are:

- the ***Bin*** directory with the EXE-file ProNC.exe and the DLLs resp. initialization files



- the ***Control*** directory with the subdirectories
  - ***Io*** for input- / output DLLs
  - ***Mctl*** for motion control DLLs
  - ***Spindle*** for spindle DLLs
  - ***ToolChange*** for tool changer DLLs
- ***SecC*** for security circuit DLLs

- the ***Help*** directory with the help files and
- the Operating instruction PRONC_OPERATE_ENG.hlp as online help in Windows format

- the Programming instruction PRONC_PROGRAM_ENG.hlp as online help in Windows format

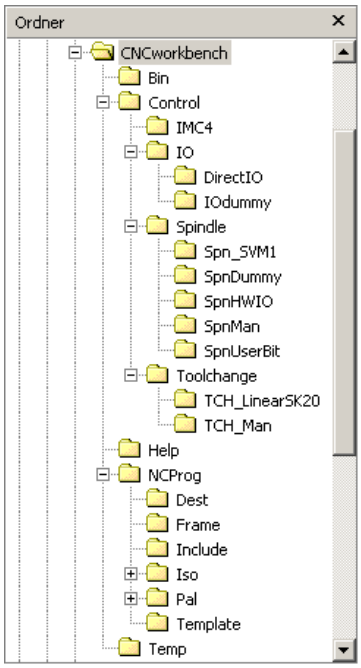
- the ***NcProg*** directory with the subdirectories
  - ***Dest*** for the files generated by the NC compiler
    - CNC target file
    - label file

    and the files generated by the CNC interpreter
    - report file
    - variable file
  - ***Frame*** for the geometry files created by the user
  - ***Iso*** for the ISO user program
  - ***Pal*** for the PAL user program
  - ***NCP*** for example program in ncp format (produced by ISY)

  - ***Template*** for provided program templates

- the ***Temp*** directory for
  - the error file created by the compiler during the compiling
  - the PAL or ISO user programs in text format produced from the RTF format before compiling

These uniform data structure below the main directory ***CNCworkbench*** shall be a simple orientation to the user and give support by access to certain files from own applications (e.g. Microsoft-Word or -Excel). ***CNCworkbench*** is comparable with a well led storage, everything has his place and is found fast.



3.4.2 Principle of Module DLLs



ProNC is the operating- and programming surface of the isel control software under Windows.
Within the hierarchy of the isel control software ProNC applies on the Module DLLs.
These **Module DLLs (also named software driver) open up the access to the complete control hardware**. To a better understanding a compare to a word processor (e.g. Microsoft-Word) can be enlisted:

MS-Word doesn't control a printer (device) itself, but it uses usually the implemented printer driver from the printer manufacturer, which also can be understood as a software driver to the special device printer . All printer producers realize for their devices a software interface (Server), which is uniform opposite to the superior program (Client, e.g. MS-Word).

These module concept is generally realized in ProNC :

All Module DLLs for ProNC dispose of a uniform assignation interface always containing the basic functionalities:

- Initialization / exit
- Version request as dialog
- Setup to adjustment of relevant parameters as dialog
- Diagnosis to test hardware functions as dialog
- State request to representation of the current module state

provide as dialog plus the module specific functions.

The Module DLLs are created by isel-automation KG resp. by OEM-suppliers and are a part of delivery of the respective module.

Each Module DLL serves to control a module. The most important modules are:

- **Controller** to motion control and to control mechanical systems with two to at most six numerical axes
- **Converter** to control a main spindle
- **Hardware** to realize a binary and/or analog input and/or output
- **Hardware** to realize supervision functions at a machine/equipment
- **Hardware** to realize operator functions (operator panel or operator bar of a CNC control)
- **Hardware** for the access to CAN bus in the controlling PC



The naming for this Module DLLs is uniform:

- the Module DLLs for motion control are called **Mctlxxx**.dll
- the Module DLLs to spindle control are called **Spnxxx**.dll
- the Module DLLs to input and output are called **Ioxxx**.dll

Hint: The motion control DLL for isel-CAN-controller is called **CanApi.dll.**

More: 3.4.3 Assignment interface

3.4.2.1 Module types and DLLs

Overview modules and hardware:

The following table shows an overview to module types, the necessary DLLs and the special hardware used by the respective DLL: (selection)

Module type	**Name of DLL**	**Controller / hardware**	**Comment to controller / hardware**
Motion control	Mctl_Upmv4.dll	Servomotor control card UPMV4/12	PC-Slot card for the ISA-Bus; requires interrupt request IRQ 10 or IRQ11
Motion control	Mctl_Imc4.dll	IMC4 Controller for at most 4 Stepper motors	For all machine of CPM and GFM family, (with 16 bit processor)
Motion control	Mctl_Ims6.dll	IMS6-Controller for at most 6 Stepper motors IML4-Controller for at most 4 Stepper motors	Controller with 32-bit-processor
Motion control	CanApi.dll	Intelligent amplifiers UVE 8112 for DC Servo motors from isel-automation KG MV-30, MV-120 and MV-320	Requires for the access to CAN bus a CAN dongle for the printer interface
Spindle control	SpnAnalog.dll	isel Multi I/O card with DA converter	For the ISA bus, generation of an analog signal with a resolution of 8 bits to set point target to converter with analog interface
Spindle control	SpnCan.dll	KEB converter F4 with CAN interface	CAN protocol by CANopen (CiA DS 402)
Input and output	MultiIo.dll	isel Multi I/O card without DA converter	ISA slot card with 16 binary inputs and 8 binary outputs
Input and output	IoCan.dll	isel CAN I/O module	Intelligent input- / output module with CANopen-Interface (CiA DS 401) with 16 inputs / 16 outputs

3.4.3 Assignment interface

Assignment interface:



All Module DLLs of a module type (selection)

- Motion control
- Spindle control
- Input / output

have a uniform assignment interface. This interface is a functional / procedural interface. In other words all module control DLLs of a module type provides a uniform function interface as library of functions with fixed defined function names and a fixed number / type of delivery parameters. The return parameter of all DLL-functions is always an unsigned 32 bit value (type: DWORD), which represent an error code by the isel error management.

Following the most important functions for the respective module types are composed:

Module type	**Functions**

Motion control	• Basic functions:
	  • Initialization, exit
	  • Version dialog, setup dialog, diagnosis dialog, state dialog
	
	• Module specific functions:
	  • Reference run
	  • Query the state
	  • Set/query processing velocity
	  • Set/query fast velocity
	  • Set/query feed override
	  • Set work piece zero point (absolute and relative)
	  • Query or delete current work piece zero point
	  • Linear interpolation (absolute- or incremental measure, processing or fast velocity)
	  • Circle- or helix interpolation (absolute or incremental measure)
	  • Query position (actual- and set value)
	  • Stop/continue/cancel the motion

Spindle control	• Basic functions:
	  • Initialization, exit
	  • Version dialog, setup dialog, diagnosis dialog, state dialog
	
	• Module specific functions:
	  • Query state
	  • Spindle on/off resp. get up to target speed / switch off
	  • Set/query spindle target speed
	  • Query spindle actual speed
	  • Set/query spindle direction of rotation
	  • Set/query spindle revolution override

Input and output	• Basic-functions:
	  • Initialization, exit
	  • Version dialog, setup dialog, diagnosis dialog, state dialog
	
	• Module specific functions:
	  • Query state
	  • Read/query input bit / byte / word / double word
	  • Set output bit / byte / word / double word to initialization value
	  • Read/ write out output bit / byte / word / double word
	  • Query current output bit / byte / word / double word

3.4.4 Configuration dialog

Configuration dialog: The configuration should be pursuable to the user with the mediated relations in the basics of file structure, module DLLs and assignment interface. Configuration means to inform the operating- and programming surface ProNC, which modules are used in the current control configuration for the concrete machine resp. equipment:



- as motion control(s)
- as spindle control(s)
- for input and output
- as tool changer
- as security circuit

This configuration is carried out dialog oriented and should be executed always in the following order:

- Version access (to check the functions of the Module DLL)
- Setup (to adjustment the relevant parameters)
- Diagnosis (to check the functionalities)
- State (to determine and check the current state of the module resp. of the hardware)

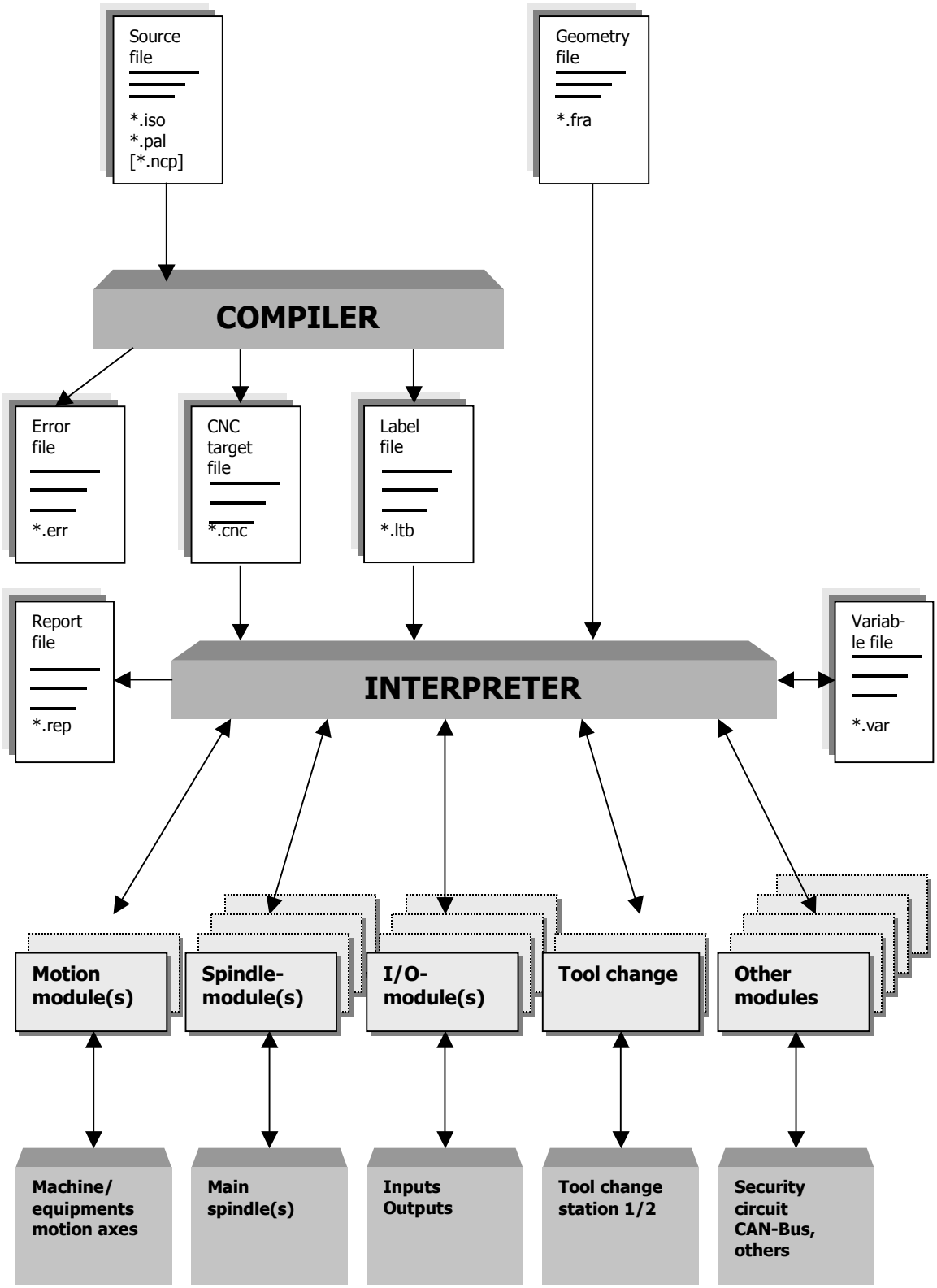
The information (description) entered by the user within the configuration dialog as well as the chosen directory- and name-character strings of the concrete module DLLs resp. their initialization files are written in the file ***NcCtrl.ini*** in the directory ***\CNCworkbench\Bin*** resp. read out from the file at a new (re-) configuration.

At start of ProNC the current/last configuration for the concrete user control is always carried out automatically. All with the symbol ☑ marked modules are assigned with the initialization by ProNC.

After a successful initialization of the respective Module DLL the functions (e.g. reference run of a motion control DLL, spindle switch on of a spindle DLL or query input / set outputs of an input- and output DLL) can be used by ProNC.

4 Coordination of all system components

4.1 Data flow and interfaces



5 Operating

5.1 Philosophy



The operating surface of ProNC offers both an optimal possibility for **creating and editing of source files** and all functions **to output these as a CNC file** and for the adjusting of all required machine and work piece parameters.

All functions can be activated about menu selection and their submenus.

To **trigger off the selected functions quickly** you can use **buttons** for selected functions. These buttons are located in the symbol bars or within separate windows. The symbol bars are classified in specific function groups in the overall concept.

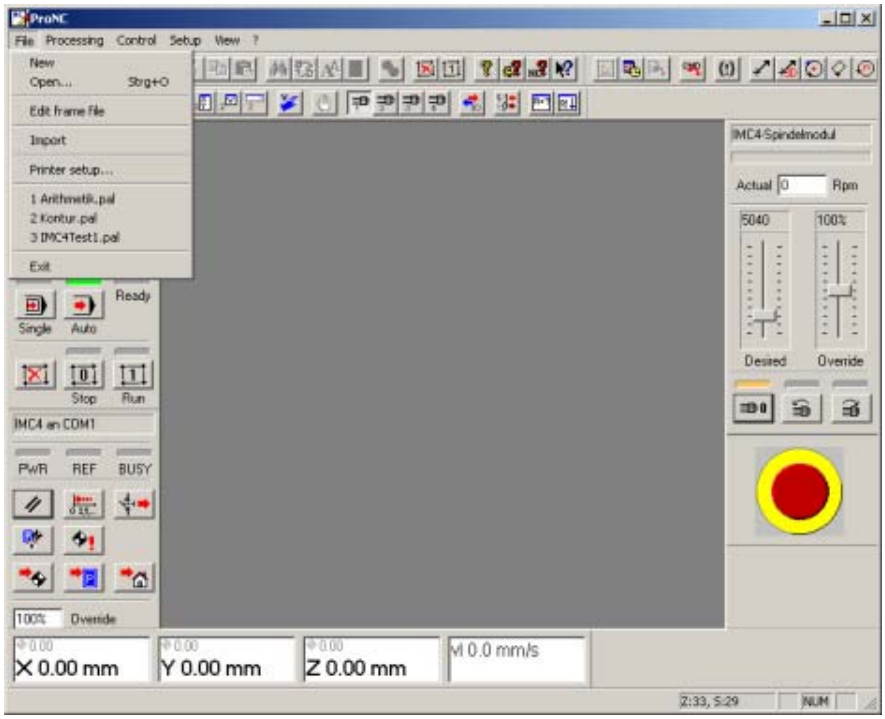
All windows using for inputs of the operator contain the buttons **OK** and **Cancel**.
Please select "**OK**", if you want to save the new parameter or your decision.
If you want to leave the dialog without modifications please select "**Cancel**".

5.2 Menu File

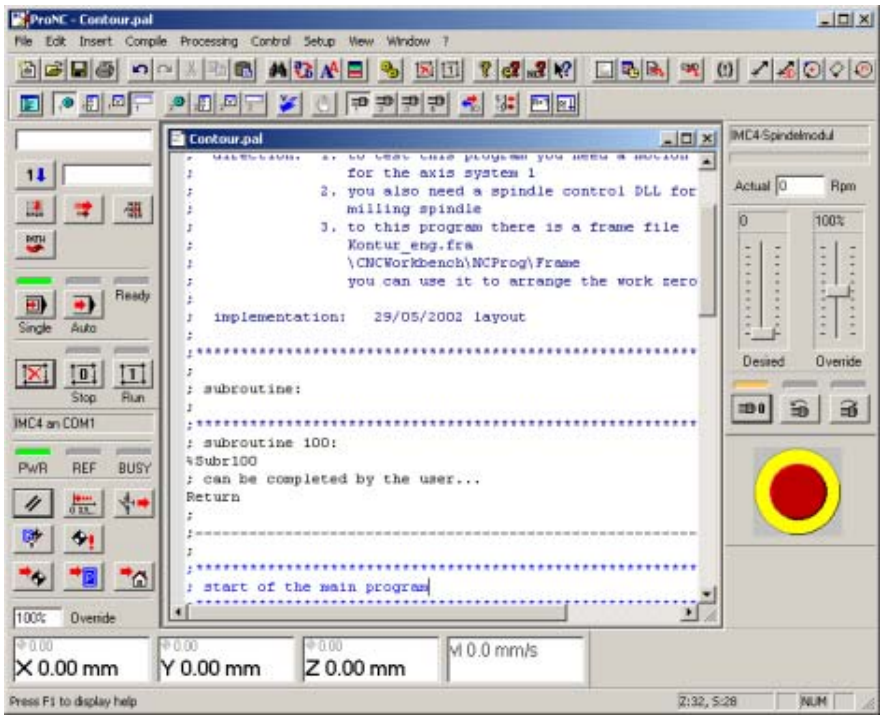
Menu File:



After start of the program ProNC the editing area is empty; no source program is loaded.
With the Menu File an available file can be opened, imported or a new source file can be created.



MENÜ „**FILE**“ WITHOUT OPENED SOURCE FILE



MENU „**FILE**“ WITH OPENED SOURCE FILE „CONTOUR.PAL“

5.2.1 Create a new file

Menu **File - New ...**



Display of an empty window for creating the new source program. In full screen mode a white area appears to edit the file after selection the Menu File - New.
Source files already loaded remain open furthermore.

5.2.2 Open file

Menu **File** - **Open...**



The directory \CNCworkbench\NcProg\PAL is presented in the dialog window to chose the wished source file. Programs with the extension ***.pal** are displayed.

To choose programs created with **ISO**, please select under **File type** the extension ***.iso**.
Please, mark the wished program with the mouse and click on the button „**Open**“ or load the program by **double click**.

If you choose *.fra as the file extension, you can load **your special user frame file** from the directory /CNCWorkbench/NcProg/Frame. It will be displayed in a window with the name.fra as headline and can be changed in this dialog.

Tip:

After opening the Menu "**File**" the last (four) opened in the so called MRU list are displayed.
With **double click** at one of these files you can immediately load them in the editing area to edit or to output.

In the Menu **Setup - Workspace** the number of files in the Most Recently Used (MRU) list is modifiable.
In this windows you can also define, that the last processed file will be **opened automatically** at start of ProNC.

More: 5.8.1 Menu Setup - Workspace

5.2.3 Import file

Menu **File** - **Import**

Please use this command to import a **NCP file** , created e. g. with ISY, in the editing area.
The **import** of the file contains the **converting into PAL format**.
Changes have to be carried out only in the PAL syntax.
Files in the PAL or ISO format are prerequisite for the compiler run in which a CNC file is produced that can be started.

5.2.4 Set up printer

Menu **File** - **Set up printer...**

Select the **printer type** and its characteristics, **paper size** and **print format** with help of a dialog box.

5.2.5 Exit of ProNC

Menu **File** - **Exit**

Finish the program ProNC.

5.2.6 Close file

Menu **File** - **Close**

The active file is **closed**. If you have carried out changes, you decide in the dialog whether you want to **store** them.

5.2.7 Save

Menu **File** - **Store**

Changes are assumed into the active file, the file is stored under the same name.



Menu **File** - **Store under...**

File is stored under a **new** name.
Please choose the desired **directory** and the **name** of the source file.

If you don't enter a new name, please decide in the dialog if the old file shall be overwritten.

Menu **File** - **Store as template ...**

File is stored as program template in the directory „**Template**“.
You can select each other folder.

5.2.8 Print

Menu **File**- **Print...**

The complete source or parts/pages of it can be printed.



5.2.9 Side view

Menu **File** - **Side view...**

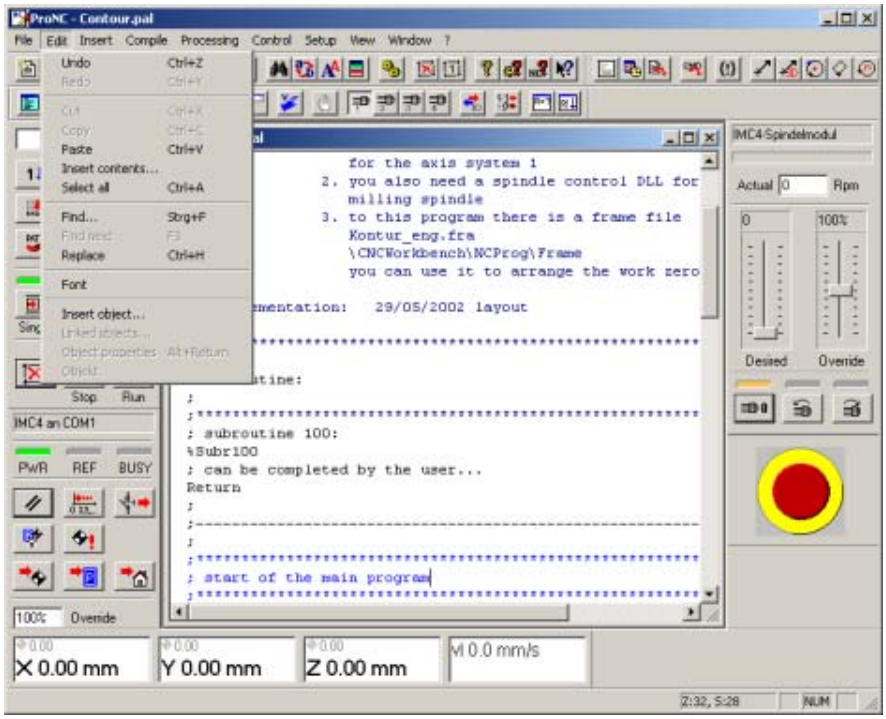
Display of the file, how it seems as printed document.

5.3 Menu Edit

Menu Edit:



The menu Processing contains all functions to edit a file and it corresponds essentially to the standard functions of editing programs.



MENÜ „**EDIT**“

5.3.1 Delete actions, restore actions

Menu **Edit** - **Undo** The **last action** at the file processing is **undone**.



Menu **Edit** - **Restore** An undone action is executed **once more**.



5.3.2 Copy passage, insert passage

Menu **Edit - Cut out** A highlighted word or a highlighted area are deleted (cut out) and stored in the clipboard.



Menu **Edit - Copy** A **highlighted word** or a **highlighted area** are stored in the **clipboard**. The **highlighted text** remains **unchanged** in the file.



Menu **Edit - Insert**



The stored text/graphic is inserted out of the clipboard to the place in the file residing the cursor.

Menu **Edit** - **Insert contents...**

Insert the **clipboard** into your program e. g. as graphic.

Menu **Edit** - **Mark all**

The **complete file** is highlighted.

Menu **Edit** - **Insert new object ...**

A file of your choice e. g. a **bitmap** is inserted in your program.

5.3.3 Searching and replacing functions

Menu **Edit** - **Search ...**



Please enter the text to be **looked for**. It will be looked for the next appearance of the entered text.
The parameter „Word“ and „Upper and lower case“ determine the **search criteria** nearly.

Menu **Edit** - **Continue to search**

After enter of the searching string and finish the command „**Search**“ you can look for the same word by the Menu **Continue to search** without a new **input**.

Menu **Edit** - **Replace**

Enter the text to be **looked for**.
Enter the text which shall **replace** the text to be looked for.
You can look for step by step and decide after found the text, if the substitution shall be carried out. (button "continue to search + replace)
By selection the button "**Replace all**" it will be replaced automatically after **each** agreement.

5.3.4 Format text



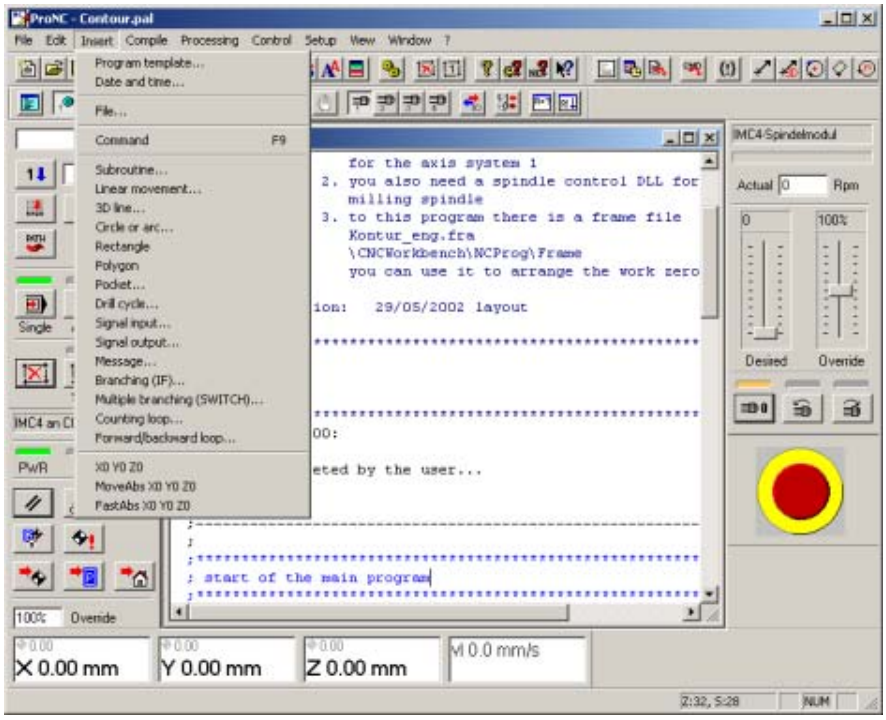

Select your favor **script font**, the **size** and **color** which the highlighted text area shall look like.

5.4 Menu Insert

Menu Insert:



This menu was implemented to be able to insert **often used program constructions** in the source file. Generally the inserting is always carried out after the current cursor position in the **active file**.
After choosing the insert function a window is opened, containing the command as well as demanded and possible input parameters.
A basic framework of the command is offered to the user, the required parameter inputs are sometimes supported graphically.



MENÜ „**INSERT**“

5.4.1 Working with program templates

Menu **Insert** -
Program template ...

A template to insert in the own user program you find in the directory \CNCworkbench\NcProg\Template. The template contains a complete program framework and is arbitrary expandable.



In dependence of your used syntax PAL or ISO please choose

- neutral.pal or
- neutral.iso

Do you want to use this insert function, please create first a new file by choosing the button  and then chose the function "Program template" in the main menu "Insert".
The user has the possibility to store programs or passages, which he would like to use often as template, into this directory.

More: 5.2.7 Menu **File** - Save

5.4.2 Date and time

Menu **Insert - Date and time ...**

This function can be used to a version identification of the program for a fast inserting of the **current time** and the **current date**.



5.4.3 Insert file

Menu **Insert - File ...**



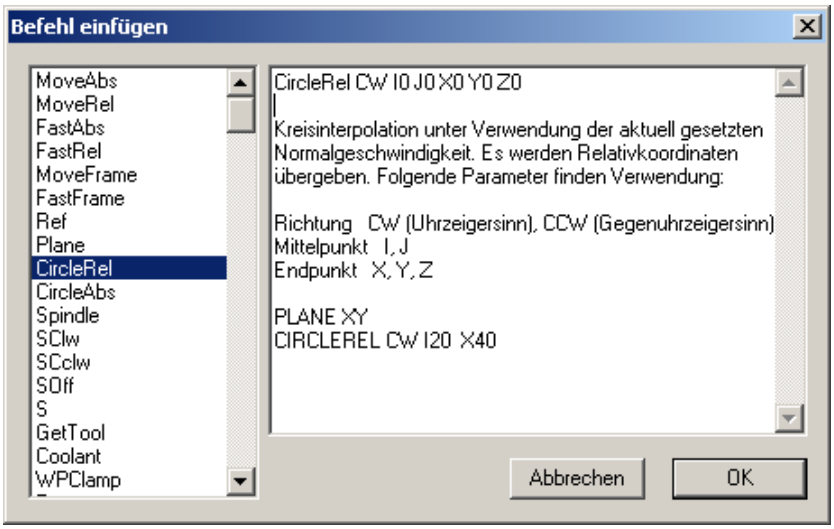
With help of the **file selection** you can insert each arbitrary file from each arbitrary folder.
The default folder is: **\CNCworkbench\NcProg\PAL**.

5.4.4 Insert command

Menu **Insert - Command**



You get a list of all commands being available in ProNC.
Every command is explained in a short description inclusive an example.
After choice of your desired command and giving a receipt with "OK" the command is taken on in your program in standard form . The input of the specific parameters must be carried out by the user.



Hint:

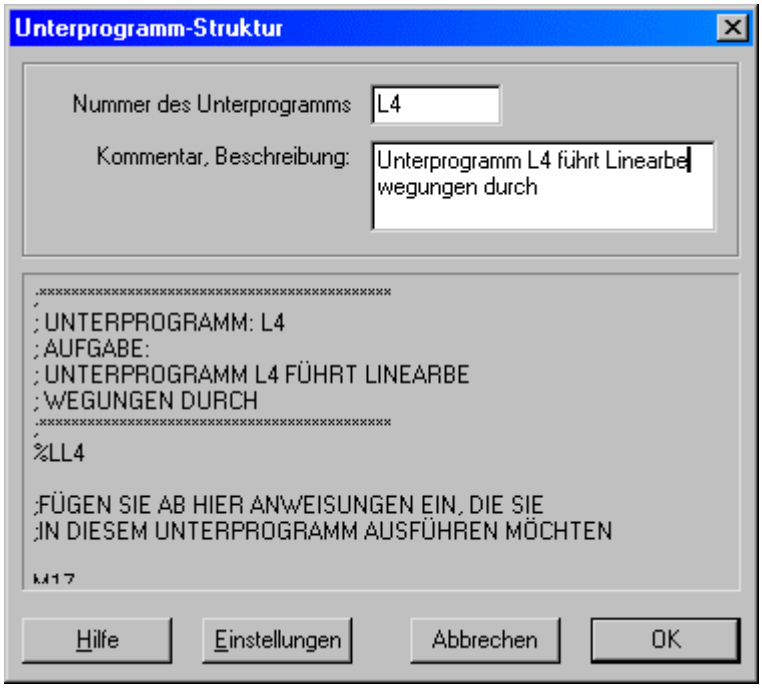
For the **program supported input** of **commands** the following described functions **5.4.5** to **5.4.14** are recommended.

5.4.5 Insert subprogram

Menu **Insert - Subprogram…**

Insert of a **subprogram structure** in the active source program.





Please enter the following parameters:

- **Number** of subprogram
- **Description** of subprogram

The number may occur in a source program only once. Plausibility control isn't carried out.
After the insert operation an "empty" subprogram is displayed in which you can write your commands and / or instructions.

Hint:

Please select with the button „Set up“ within the dialog window, if you use **PAL** or **ISO syntax** for your programming.

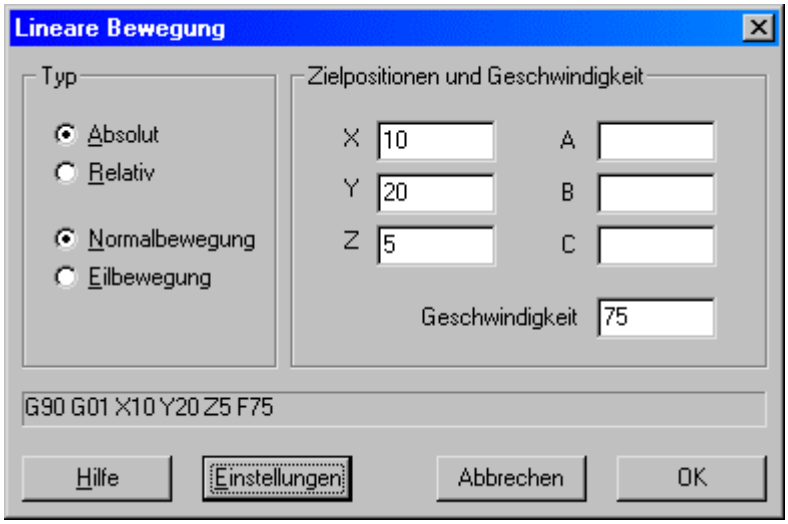
More:
Programming Instruction: Subprogram call

5.4.6 Linear movement

Menu **Insert** - **Linear movement ...**



Inserting the command for a linear motion with declaration of the target coordinates. Move to the input fields with the tabulator button or place the cursor with the mouse into the desired input field.



- **Type**
  declaration of a linear motion command with fast velocity or processing velocity with absolute or relative coordinate values

- **Target positions**
  linear axis position values in [mm] for at most 6 axes
  unit of measure adjustable under:
  5.8.3 Menu **Setup - Units**

- **Velocity**
  axis linear velocity in [mm/s]
  unit of measure adjustable under:
  5.8.3 Menu **Setup - Units**

Hint:

Please select with the button „Set up“ within the dialog window, if you use **PAL** or **ISO syntax** for your programming.

More:
Programming Instruction: Positioning with fast velocity
Programming Instruction: Linear interpolation

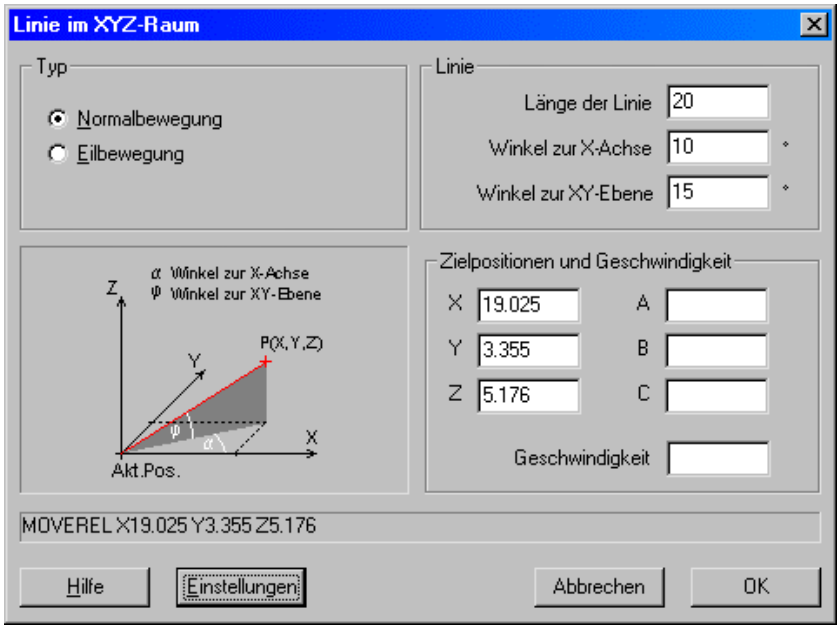
5.4.7 Linear movement in space

Menu **Insert** - **3D line ...**



This macro contains the inserting of the command for a **linear motion** with declaration of the **line length**, **angle to the X axis**, **angle to the XY plane**. Only commands with relative coordinate description with fast or processing velocity are created.

The **advantage** of this input help exists in the fact that with the input of the line length, the angle to the X axis and the angle to the XY plane the target coordinates X, Y, Z are found out automatically.



- **Type**
  declaration of a linear motion command with fast velocity or processing velocity with absolute or relative coordinate values

- **Line**
  - length between the current position and target position in [mm]
  - phase angle α to X axis in [°] after axis projection
  - angle φ to XY plane in [°] after axis projection

- **Target positions**
  linear axis position values in [mm] for the axes A, B, C
  unit of measure adjustable under:
  5.8.3 Menu **Setup - Units**

- **Velocity**
  axis linear velocity in [mm/s]
  unit of measure adjustable under:
  5.8.3 Menu **Setup - Units**

Hint:

Please select with the button „Setup“ within the dialog window, if you use **PAL** or **ISO syntax** for your programming.

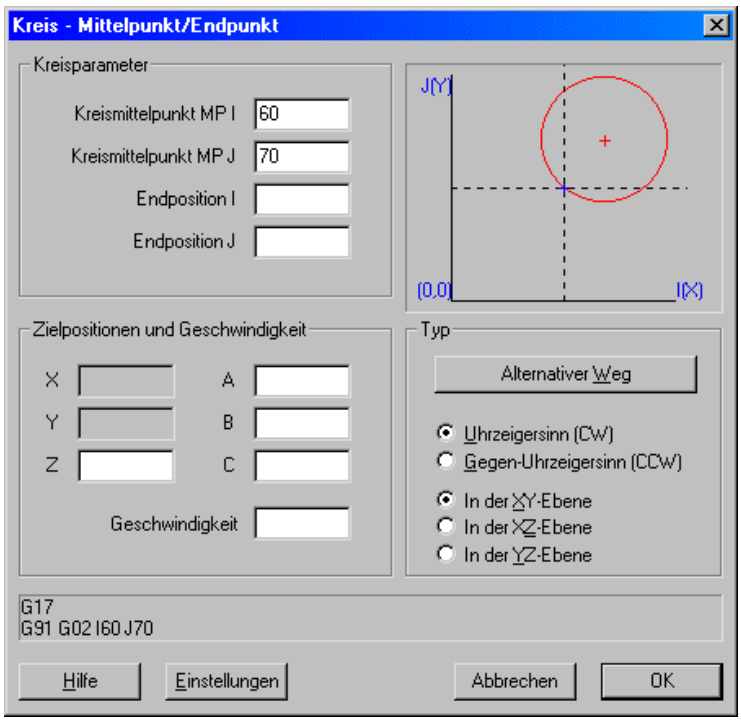
More:
Programming Instruction: Positioning with fast velocity
Programming Instruction: Linear interpolation

5.4.8 Circle or arc

Menu **Insert** - **Circle or arc ...**

Insert the command **Circle / Circle arc** in the active interpolation plane clockwise or counterclockwise.





- **Type**
  - select the direction of rotation
  - select the plane, on which the circle command relates
    the XY plane is always default

- **Circle parameter**
  - input of the center coordinates in [mm]
  - input of the corresponding center coordinates in [mm]

- **Target position**
  linear axis position values in [mm] for at most 4 axes
  unit of measure adjustable under:
  5.8.3 Menu **Setup - Units**
- **Velocity**
  axis processing velocity in [mm/s]
  unit of measure adjustable under:
5.8.3 Menu **Setup - Units**

Hint:

Please select with the button „Setup“ within the dialog window, if you use **PAL** or **ISO syntax** for your programming.

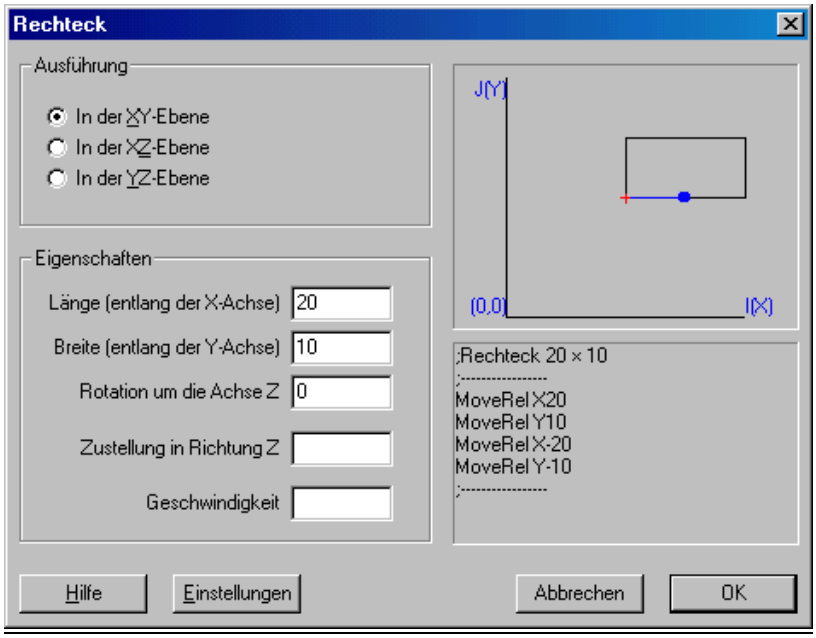
More:
Programming Instruction: Circular interpolation clockwise
Programming Instruction: Circular interpolation counter clockwise

5.4.9 Rectangle

Menu **Insert - Rectangle ...**

Insert the **command sequence** to **create a rectangle**



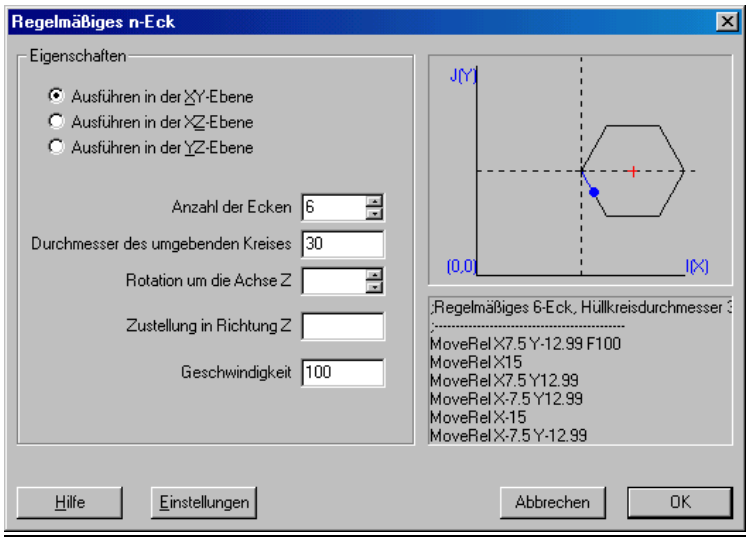


5.4.10 Polygon

Menu **Insert Polygon ...**

Insert the **command sequence** to **create a polygon** with choice of the number of corners



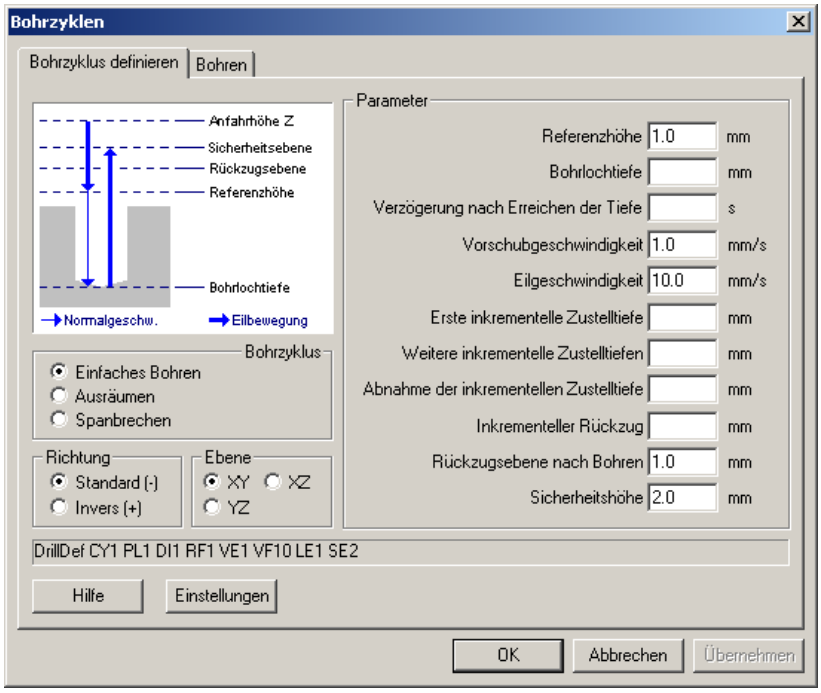


5.4.11 Drilling cycle

Menu Insert - Drilling cycle



Please define the **drilling parameter** for the drilling command alternatively for **single drilling, drilling mode counter sick, drilling with break chip**.
After choice of the index card „Drilling“ please enter the **coordinates** for the drilling process.

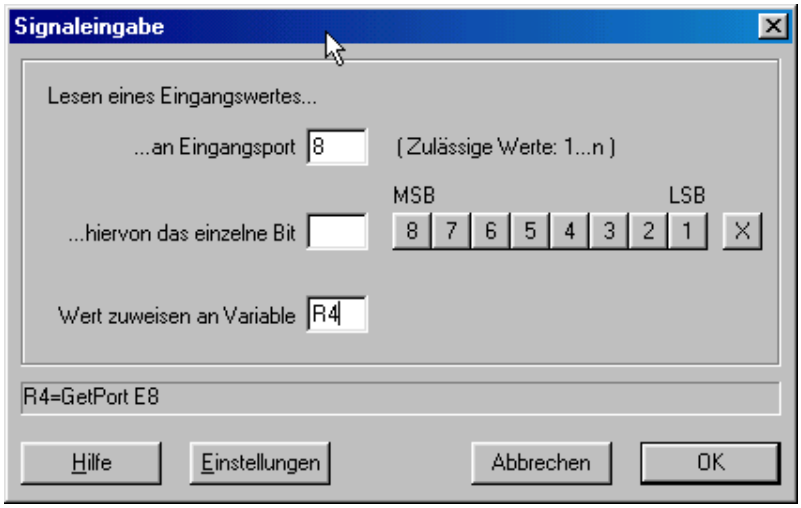


5.4.12 Signal input, signal output

Menu **Insert - Signal input …**



Version 1:
Insert of a command to set the chosen input port. You determine the wished input port and assign a value.



Menu **Insert - Signal input ...**



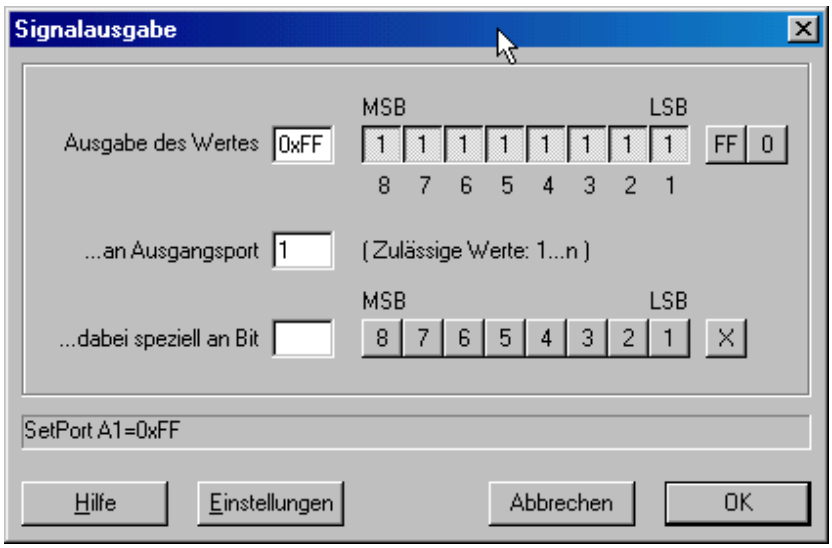
Version 2:
Insert of a command to evaluate a **definite bit** at input port and submit the value to a variable.



Menu **Insert - Signal output …**



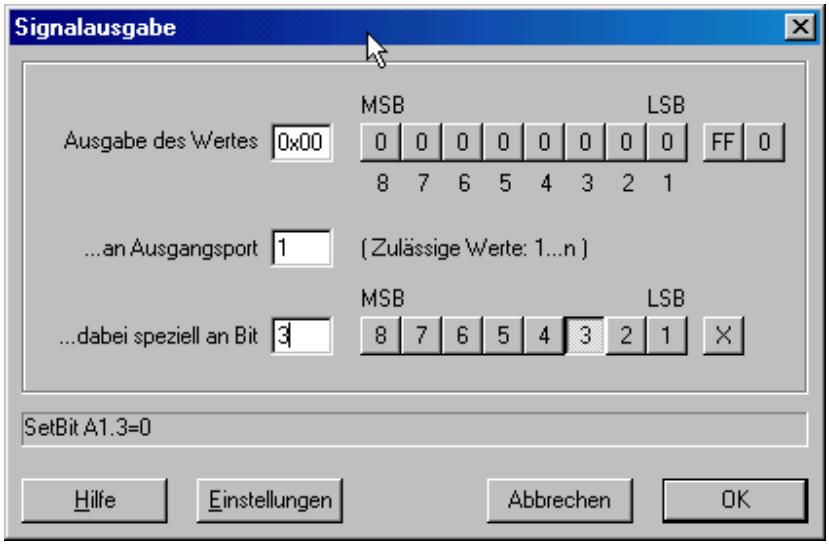
Version 1:
Insert of a command to set the **chosen input port** and assign the wished value



Menu **Insert - Signal output ...**



Version 2:
Insert of a command to set or reset of an **output bit 1 to 8** on the chosen output port

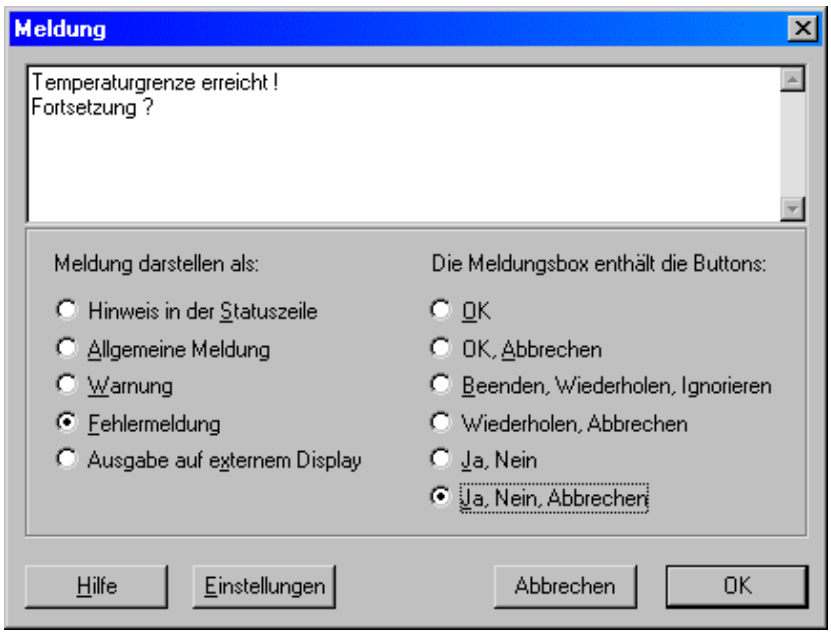


5.4.13 User-defined message

Menu **Insert - Message ...**



Selection of a message box; the text and the **wished button** for the dialog you can choose **yourself**.



- Please enter the wished text, that shall be published as a message in a certain program situation

- Please select the **format** of the **message box** and the including **buttons**

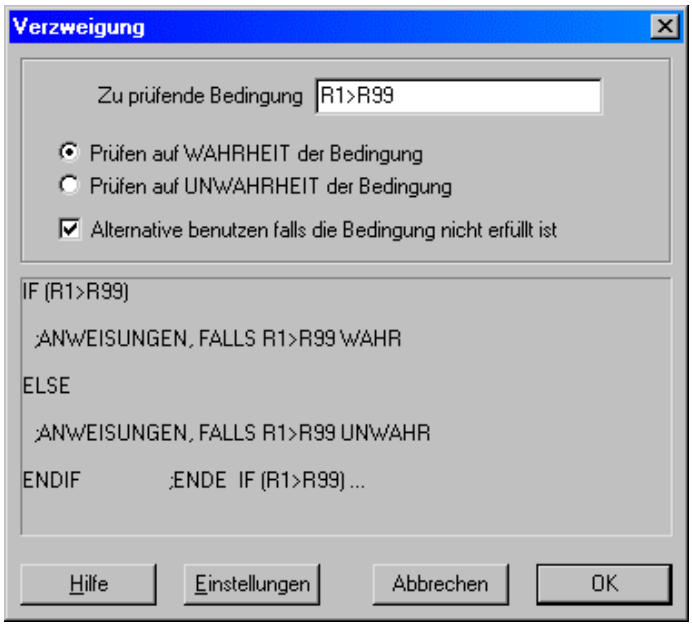
More:
Programming Instruction: Request for an operator dialog

5.4.14 Branching

Menu **Insert** - **Branching ...**

Inserting a **structure** for programming of **branches** in dependence of a **condition**, declaration of an **alternative** if the **condition isn't satisfied.**



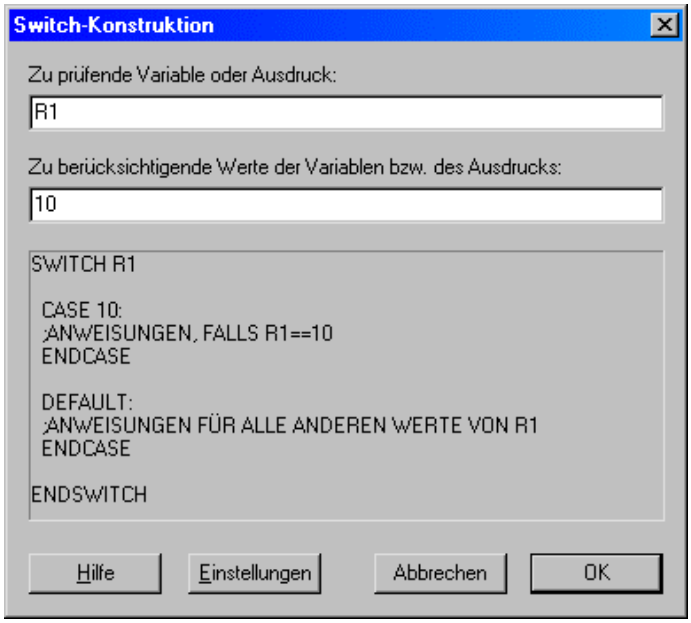


More:
Programming Instruction: Branch

Menu **Insert** -
Multiple branching ...



Selection instructions, realizing a **multiple branch** by extension of the IF-instruction in dependence of the **contents** of the **variables** *or* **expression** which shall be checked.



More:
Programming Instruction: Selection instruction

5.4.15 Counting loop

Menu **Insert** -
Counting loop ...



An **instruction** or a **sequence** of **instructions** is/are executed as long as a counting variable, whom a **start value** was assigned, reaches a value, which had reached by a **cyclic increment**, that is greater then the defined **end value**.

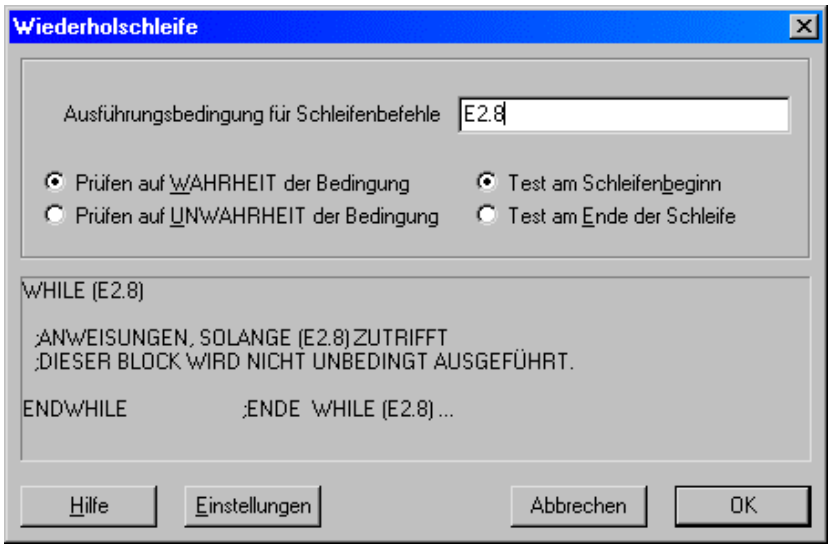


More:
Programming Instruction: Counting loop

Menu **Insert** - **Forward/backward loop ...**



An instruction or a sequence of instructions is/are executed as long as the defined condition is true.



More:
Programming Instruction: Loop with test at start
Programming Instruction: Loop with test at end

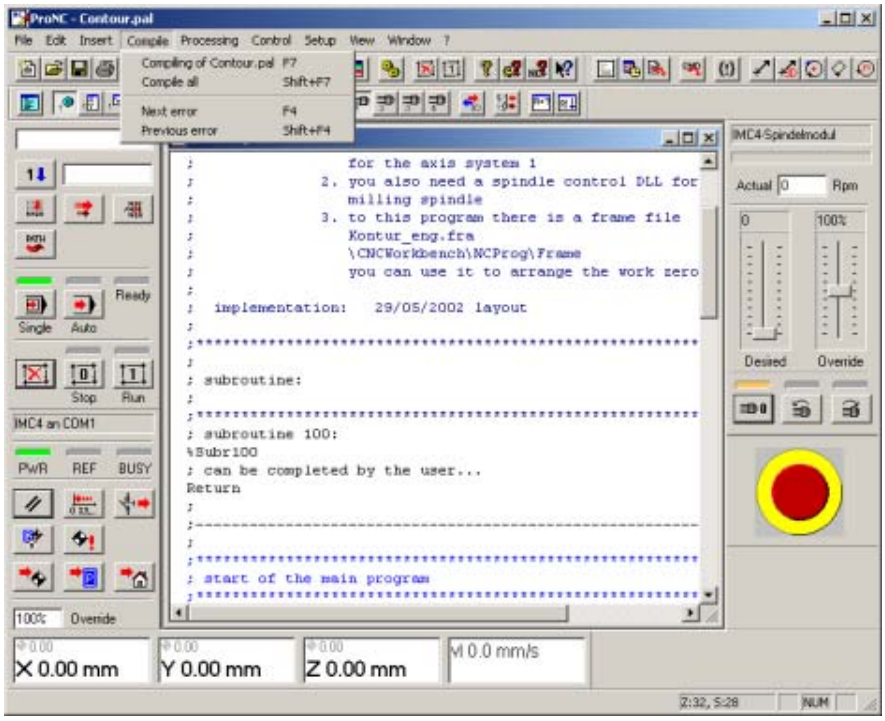
5.5 Menu Compile

Menu Compile:

The select of Menu **Compile - Compile file** starts the compiler run.

In case of syntactical faultless of the source file the CNC target file and the label file are created or updated, and the program processing can be started immediately out of the Menu **Processing - Start**

If the compiler run isn't faultless, a dialogue window is shown, which includes an error list, the syntactic errors in the source program have to be corrected.



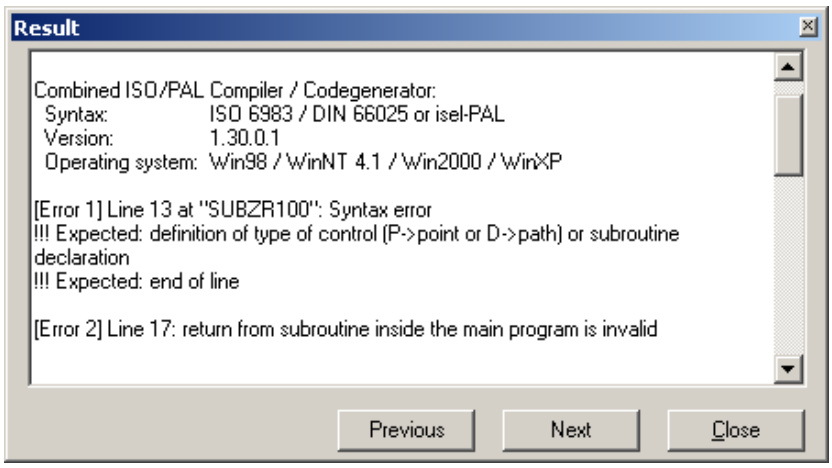
MENU **COMPILE**

5.5.1 Error window

Example for an error list on screen

A write error in a program with subprogram (extract) produces an error window with following contents:

```
;-----------------------------------------------------
; Subprogram carries out a full circle clockwise:
;
%SUBZR100
; REL   : the coordinates of X and Y are relative values
.
.
.
; call Subprogram:
  N70 SUBR
.
.
.
```



The line numbers are obviously, in which the errors were recognized. In the same windows hints are given to you for a correction.

In the example the command to declaration a subprogram **%SUBR** was recognized as syntactic wrong because of the incorrect notation, as an inherited error seems the non available subprogram declaration.

You can move to the set of the source program you have to correct by clicking the button „**Next**“ or/and „**Previous**“.

The positioning of the line pointer on the wrong line you can execute with the key **F4** resp. **Shift+F4**. F4 causes the jump to the next error, Shift+F4 moves back the line pointer in the error list.

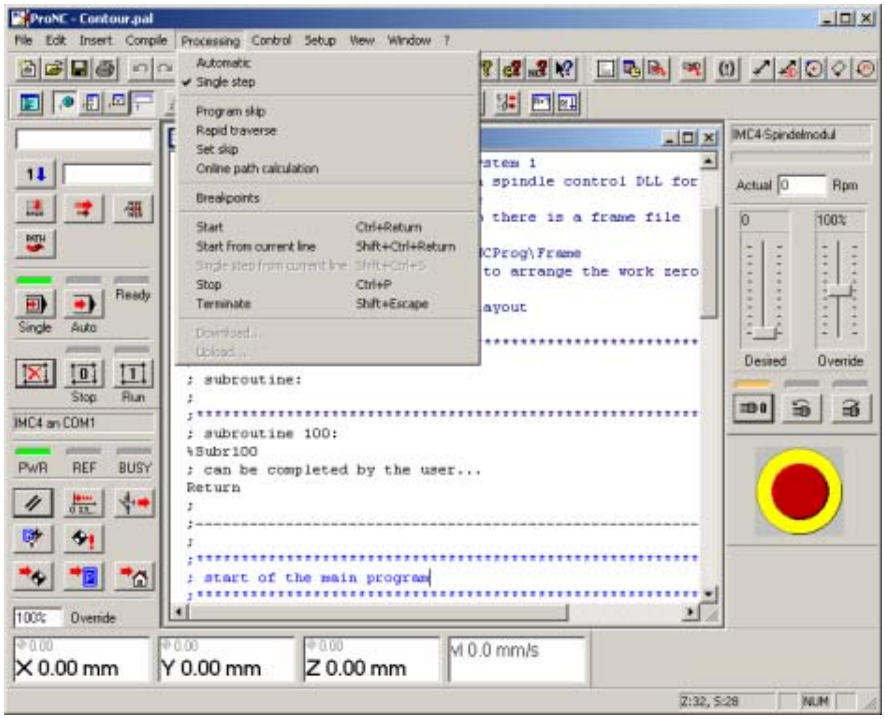
5.6 Menu Processing

Menu Processing:



After call of these menu and select of the submenu **Start** the Interpreter integrated in ProNC is started. The **interpreter** analyses the NC information saved in the CNC program and submits it **to the machine** as corresponding **commands** and **driver calls**.

In dependence of adjustment the processing of the user program carries out in Automatic- or Single mode.
Stops can be agreed, the set skip after defined program abort is adjustable, Online-path calculation is possible, if the realizing by the control is possible.



MENU **PROCESSING**

5.6.1 Operating mode Start, Stop, Terminate

Menu **Processing** - **Automatic mode**



You start the processing of the CNC target file in sequence of the program lines automatically.

Menu **Processing** - **Single step**



Select this function, to process a program **step by step**. The running command is marked by a beam.
The continue of the program in single step mode carries out by click with the mouse on the button start .
This **operating mode** is used primary in the test phase of a program, generally to find bugs.

Menu **Processing - Stop**



To make e. g. a quality control of the processing, you can define to interrupt the execution of the current motion by the driver.

To continue please click on the button Start .

Menu **Processing- Start**



The **user program** is started with the first command line of the main program. According to the defaults in the Menu - Setup - Processing e. g. a reference run can be executed **after** the start of the processing function and **before** the execution of the first line in the NC program.
To start the program beginning from an arbitrary place in the program please place the cursor on the wished line and select out of the **Menu - Processing:** Start from current line.

Menu **Processing - Terminate**

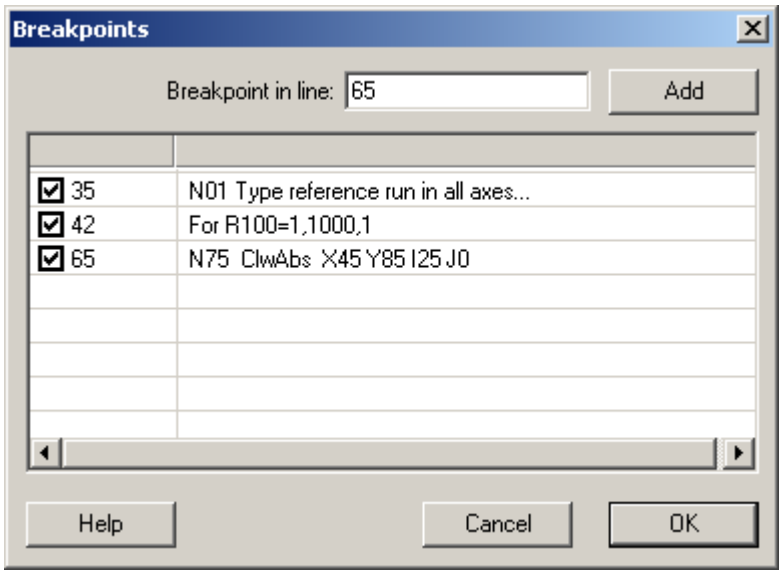


The user program is finished immediately. The spindle is switched off. The valid **settings** in program are stored in the **initialization files** by finishing the program.
Using the function Set skip the program can be continued at any time.

More: 5.6.3 Menu Processing - Program skip

5.6.2 Breakpoints

Menu **Processing - Breakpoints ...**



Definition of breakpoints:

Before starting the processing of the CNC file, you can define **breakpoints** in your program.

Please position the **cursor** in the **wished line**. Activate the function breakpoints in **Menu Processing**.

The line number is adjusted and after clicking the button **Attach** this line appears in the list. The s**ymbol** ☑ points up, that the break point is **active**. A **deactivation** is displayed by the **symbol** ☐.

These break points are highlighted in your program with the marker "▲" (active) and with "△" (inactive).

You have also the possibility, to enter the line number direct.

Processing the program at this **break points** will be switched to **single step**, after that you can switch to automatic mode until the next break point or you continue the process in mode single step.

5.6.3 Program skip, Rapid traverse, Set skip

Menu Processing - Program skip



With the function **Program skip** user programs can continue at the break point, breaking off during the automatic mode.

To activate the Program skip, click the button  before starting the program again.
A condition for a perfect continuation is, that the position of the workpiece did not change and the program abort occurred with the button  or with the menu **Processing - Abort**.

After you have activated the Program skip and started your program, a window is displayed by reaching the breakpoint, you should check it before starting.

Menu Processing - Rapid traverse



The function **Rapid traverse** causes, that all motions with programmed feed (ISO: G1, G2, G3, G11; PAL: MOVEABS, CLWABS, CCLWABS, MOVEFRAME) will run with fast velocity.
This function can be activated and deactivated during the runtime of the user program.
The function hasn't an effect, if the spindle isn't switch on.
To **test a large program** you can use this function, but please **pay attention to the following hint:**

Attention:

Please note, that a **meaningful processing is not possible** with an active **rapid traverse**. **Clamp** the tool **out** of the spindle, **correct** the work piece zero point in **+Z** or **do not clamp a work piece**.
At Laser- or Water jet equipments you should provide security for a deactivated Laser aggregate / High pressure pump.

Menu Processing - Set skip



If you want to use the function **Set skip**, please mark the sets in your ISO or PAL file with **the character " / "** as first character of the set. All sets with this marking will be skipped during processing.

More:
Programming Instruction Set number: N-word

5.6.4 Online path calculation

Menu **Processing - Online path calculation**



The controller IMS6/UPMV4 and CAN have the ability of a buffered continuous path calculation (Online).

This feature can be activated with this button.

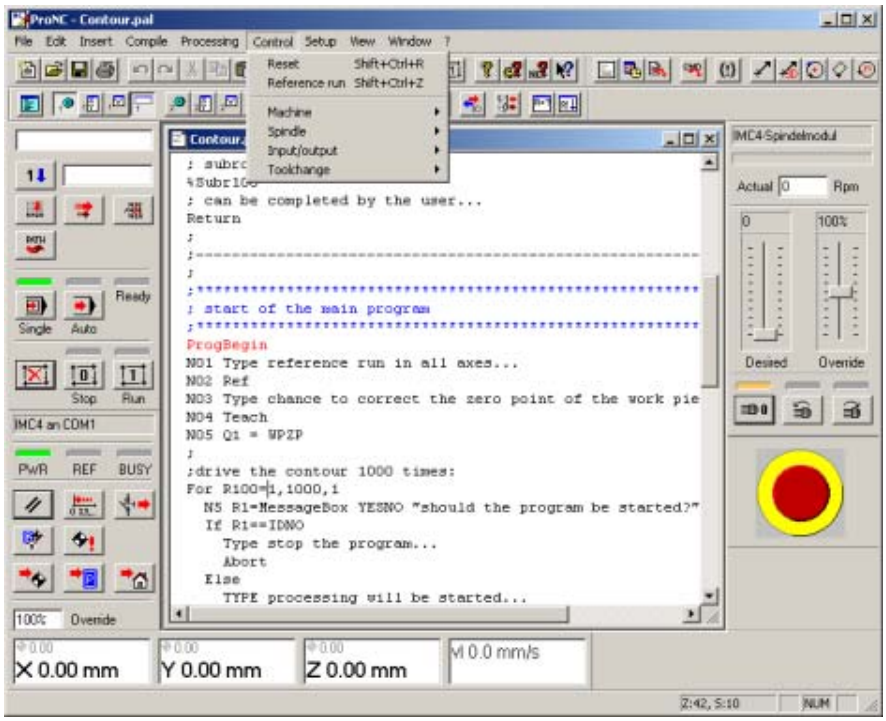
More: 5.8.4 Menu Setup - Interpreter

5.7 Menu Control

Menu Control:



This menu is **the main menu** for the **operator** and **service technician** for defining the machine parameters and to **execute an operator control action.**
It contains the machine set ups and the setting of processing parameters after a successful defining of the user program .



MENU „**CONTROL**“

5.7.1 Reset

Menu **Control - Reset**



Choose this program function to execute a **reset to the motion control module**.
This command must be carried out generally at the beginning of a processing so that an **initialization of the generated modules** can be executed.
During the processing a Reset by the driver should only be executed if a fault situation has happened.

5.7.2 Reference run

Menu **Control - Reference run**



The **Reference run** function causes the motion driver to execute a reference movement of all connected axes.
The axes in **order Z, Y, X** move one after the other until reaching the reference switches.
If the motion control recognizes a limit switch, this position will become **the machine zero point for all following movements of the relevant axis.**

The **Reference run** enables the motion control **to assign** the **position value** (0,0,0) to the **mechanical zero point** of all axes.

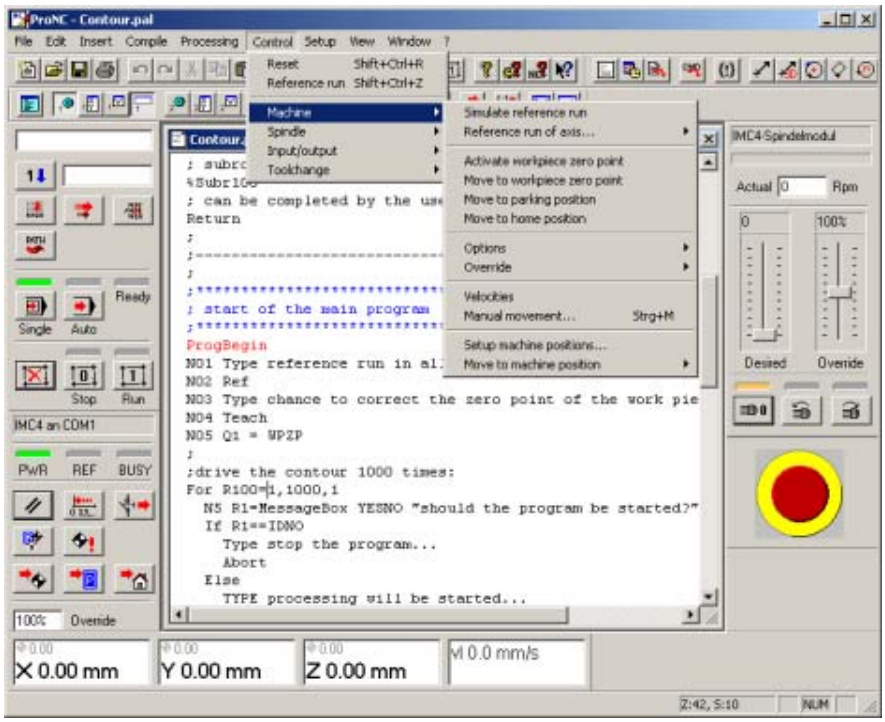
After a reference run the current activated work piece zero point is deleted . Before starting or continuing the program please set the desired the work piece zero point.

5.7.3 Machine

Menu Machine:



The **submenu Machine** contains in addition to the Menu Set up the most important functions to adjustment the plant and for the relevant settings in processing.



5.7.3.1 Simulate reference run

Menu **Control - Machine - Simulate reference run**

In the fault case (e. g., a reference switch isn't recognized at a reference run) this program function offers to you the possibility to move the plant **nevertheless into a restrictedly operational state** without executing the obligatory required reference run. This function indicates the driver to accept the current position of the plant as a new machine zero point without a real execution of the axes movement.

The access to some important program functions which would be disabled in the fault case becomes possible in this case.

Menu **Control - Machine - Reference run of axis**

After selecting this function you get a list of the six axes X, Y, Z, A, B, C . You have the possibility to activate the reference run **only for the each selected axis**. This can be important to drive e. g. an axis out of a **dangerous area**.

5.7.3.2 Activate work piece zero point

Menu **Control - Machine - Activate work piece zero point**



With this function the position of axes defined in the list of machine positions is activated as work piece zero point.
The activation resp. deactivation is visible by a marker in front of the text, a "pressed" button stands for activation of the work piece zero point and reversed.
This position is defined in dependence of the processing task by the operator and serves as starting point of the processing as long as a new work piece zero point is defined and activated in program or manually or as long as the zero point is deleted by a reset with following reference run.

The work piece zero point must be defined in the list of machine positions by the marking . This symbol marks the position as current work piece zero point.

More: 5.7.3.10 Menu Control **- Machine - Setup machine positions**

If no work piece zero is defined in the list of machine positions, the command is ignored.

The result of the activation of the zero point is visible by changing the color of the axes positions.

Hint:

The position next to the symbol  marks the defined work piece zero point, the values next to the axes (X, Y, Z ...) the current axis position relatively to the activated zero point.



For a visual support of activation of the zero point please use the combined display position and velocity.
The **execution of this function** doesn't yet contain the motion of the axes to this position.

5.7.3.3 Delete active work piece zero point

Deactivate work piece zero point

Before setting up the work piece zero point it makes sense to delete an already **activated** zero point.

By an additional pressing of the button  the work piece zero point can be deleted again. The current position of the axes related to the machine zero point is displayed in black color.

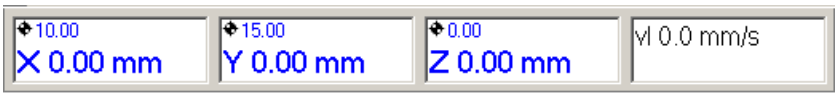


5.7.3.4 Approach to work piece zero point

Menu **Control - Machine - Move to work piece zero point**



The defined work piece zero point from the list of the machine positions is approached. When approaching is done, the current position of the axes X, Y, Z has the value zero, if the work piece zero point was activated.



All coordinate declarations of the following movement to positions refer to this current work piece zero point.

Hint:

Approaching a work piece zero point **does not enclose the activation** of the work piece zero point.

Tip:

Before you choose this function, please check in the window "Machine positions" or on the position display the defined work piece zero.
If the displayed position lies e. g. in the work piece raw material the proceeding motion could damage the work piece.

5.7.3.5 Move to parking position

Menu **Control - Machine - Move to parking position**



The park position is a position defined by the user, which is approached e. g. after finishing the processing function of the program, for loading with material, for unloading of the work piece.

The approach to the parking position is started by select the menu or by pressing the button.
The approach to the parking position can be executed also automatically after the end of a processing.

For this the option in Menu **Setup - Processing** „Move to parking position " must be activated.

More:
5.7.3.10 Menu **Control - Machine - Setup machine positions**

5.7.3.6 Move to home position

Menu **Control - Machine - Move to home position**



The home position is an exposed position of the machine axes, which is usually defined by the machine manufactures.
This position ensures a maximum distance to the tool because it is provided for a harmless opening of the cover e. g. during program runs.
The analysis of the home position is sensor monitored.

If you are using a security circuit module to extension the security function of the plant, the control signal of the Home position is needed, to open the cover locking.
The opening of the cover is possible if the Home position is reached **and** the spindle is switched off.

If the processing is exclusive program controlled, the Home position can be approached with motion commands or manually with buttons on the machine panel resp. with the described functions.

More:
1.6.3 Additional windows: SC module

5.7.3.7 Override

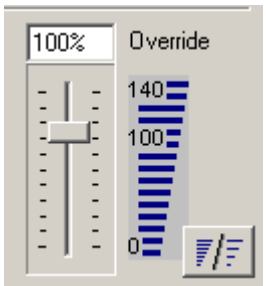
Menu Control - Machine - Override

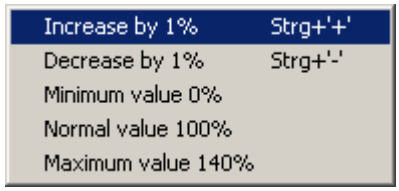
With this function the **current velocity** of the **axes** can be changed, that is defined in the initialization file of the motion module or in the application program.

The modification is possible during the run time of the CNC program. The current velocity is changed.
Therewith you can afterwards correct a value for the processing velocity, defined too high or too low, during the work processing.

Please take into account: Using a stepper motor drive the velocity change always gets effective just in the next set of the user program.

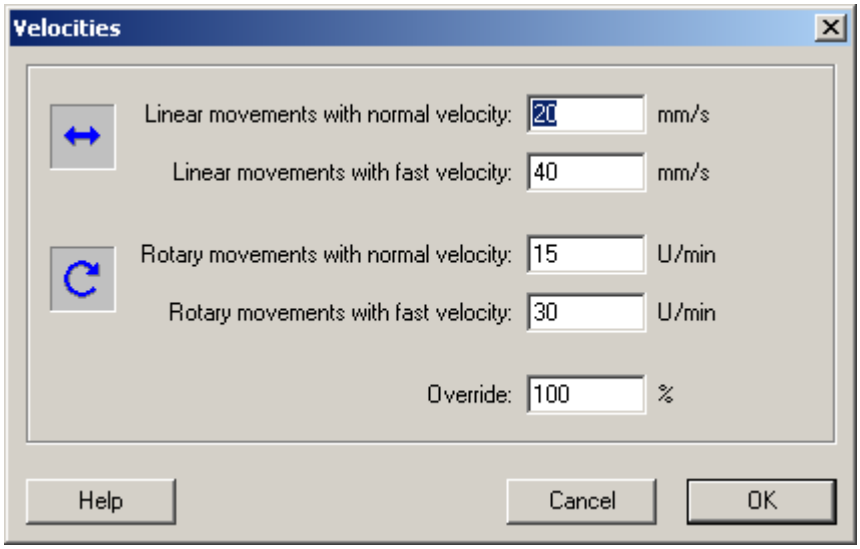
You can change the override about slide control or the modification can be an absolute value. The button  enables the die default with slide control in 1% or 10% steps.





5.7.3.8 Velocities

Menu **Control - Machine - Velocities**



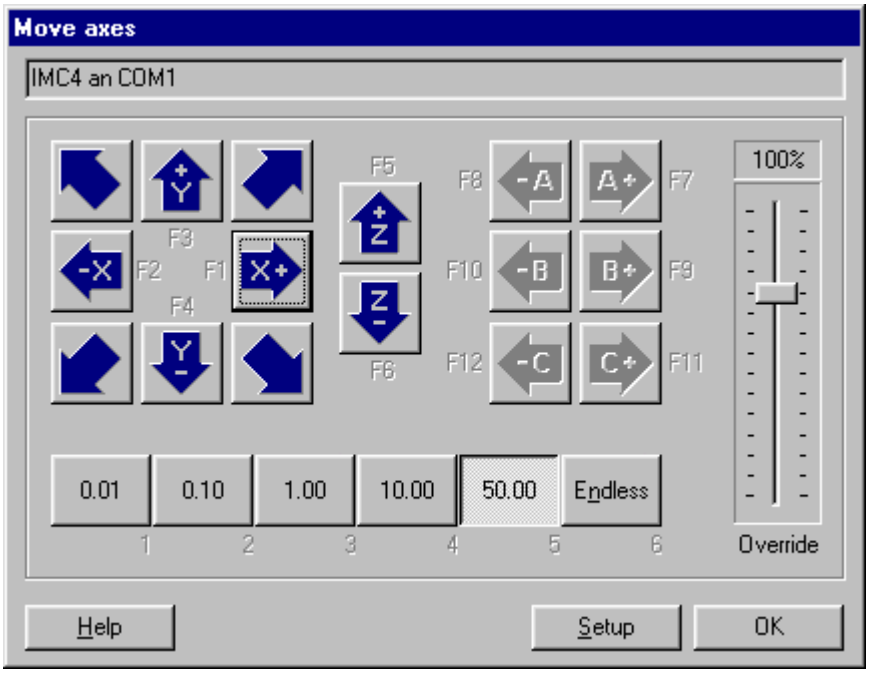
After the installation standard velocity values are initialized.
It is differed in fast velocity (for positioning) and normal velocity (for processing).
These speeds get effective if an application program is started which doesn't contain any details on the speed.

Velocities defined in the user program have always priority.

5.7.3.9 Manual movement

Menu **Control - Machine - Manual movement ...**





This function enables the manual movement of all activated numerical axes.

Please select step widths of the movement with the buttons (0,01, 0,10, 1,00 ...) or you can define it after choosing the button "Setup".
To move the Z-axis you should prefer small distances to touch the surface with the tool for defining the zero point.
For the movement of the axes please select one of these three possibilities:

1. **Mouse click on the dart** with pretended direction (e. g. -X, +Z);
   for a slantwise motion of X and Y axes you can also use the diagonal darts

2. **Move with help of the function keys** F1 ... F6 (for the axes X, Y, Z) respectively F8 ... F12 (for the axes A, B, C).

3. **Move** the axes with **joystick** (option).

Tip: The menu is immediately shown on screen with the key combination Ctrl+M.

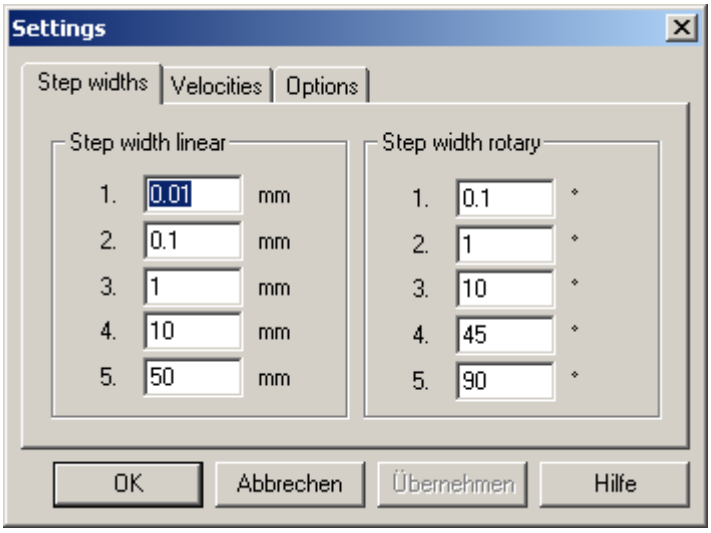
Hint: If two axis systems are configured, please select the machine for which the axis motion shall be carried out

Motion modes when driving axes: The possible motion modes are:
Step
or
Axes drive

Axes drive is reached, if the button „**Endless**“ was activated.
If you want to approximate to a certain tool position in **step mode** e. g. at fixing of the work piece zero point (Scribing of the work piece surface), please choose the button with the wished step width.
The distances 0.01 mm, 0.10 mm, 1.00 mm, 10.00 mm, 50.00 mm are offered to you.

Adjustment of step width and motion velocity

A **modification** of the **step width**, **velocities** of the linear- resp. rotatory axes, or the option "**Display of keyboard assignment**" you can **carry out with** „Setup“ within the window „Move axes“.



5.7.3.10 Setup machine positions

Menu **Control - Machine - Setup machine positions**



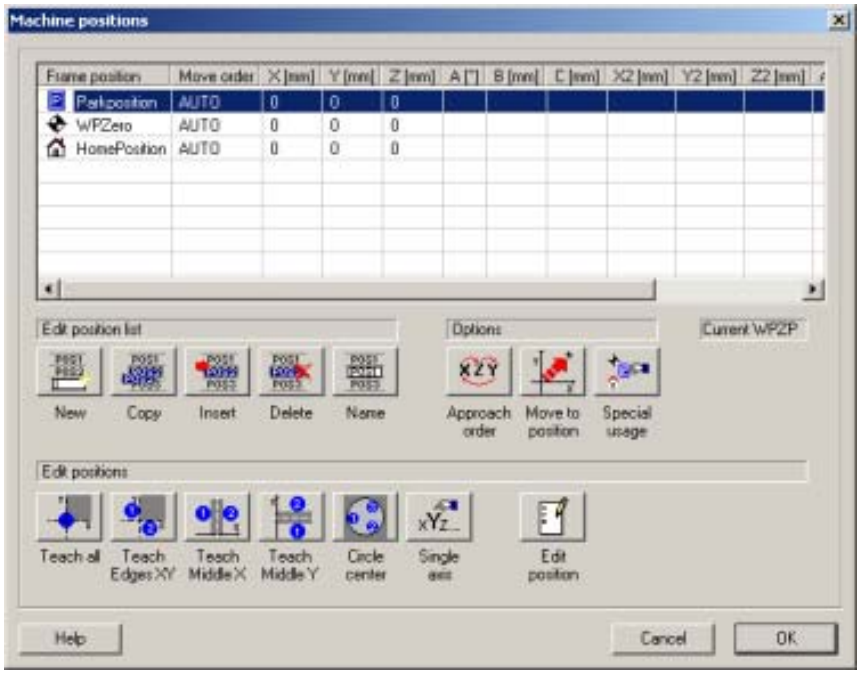
Machine positions (also named frames) define a certain **point in the coordinate system** e. g. work piece zero point, park position, start position for processing.
A **Special use** e. g. use position as work piece zero point, use position as tool change position can be assigned to these machine positions.
This marking can be used to approach such a position independently of program before start of processing.

More: 5.8.6 Menu **Setup - Processing**

You can define arbitrary many points in the coordinate system in the list of the machine positions either by teach or by editing the position.

So that it will be possibly **to define different zero points for different work pieces** one-time and to assign the special use "define as work piece zero point" according to the processing task before program start.
Additional to the version to define the work piece zero point manually, a **direct access** to the coordinates of the **machine positions** (fames) can be executed by **the application program**.

More:
Programming Instruction: 3.1.1.7 FastFrame
Programming Instruction: 3.1.1.8 MoveFrame



After selecting of this function the list of the machine positions is shown to you (standard fame file).
The display is line-based and contains:

- **Name of machine positions (frame)**
- **Approach order**
- special **usage** of frame e. g.

  for current work piece zero point

- **Positions** of **axes** X, Y, Z and A, B, C.

For editing a machine position please select the wished line. A beam points out the selection to you.
All further inputs refer to this machine position.

Edit list:

To correct, to construct or to delete machine positions you can use the buttons **New, Copy, Insert, Delete, Name.**

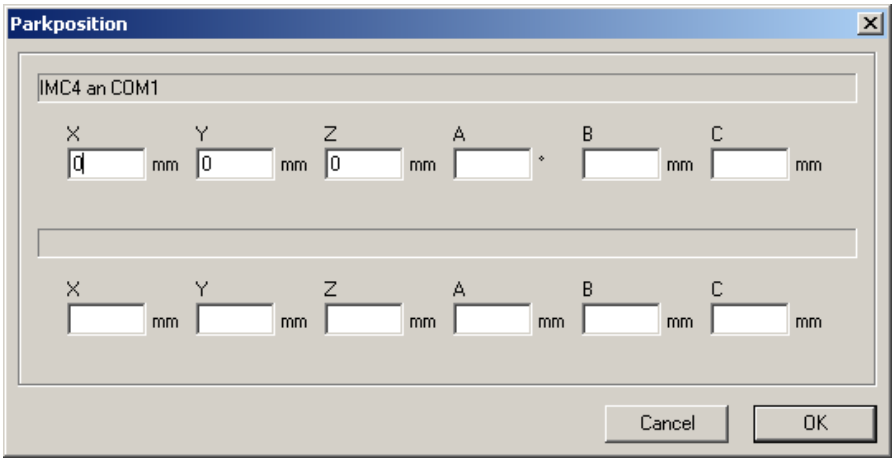
If you want to define a new frame, please follow the next steps:
Mark with a mouse click e. g. the line "WPZero".

Select from the "Edit position list" the button 

Insert.
Above the marked line a empty line appears.
Please enter a name for the position and enter the coordinates or teach them, how it is described in the following both sections.

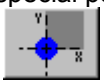
Edit of frame positions:

With the button  or with double click at the marked frame a **window** is opened. In this window it is possible to input the numerical values for the activated axes.
The second line is reserved for the extension of the equipment with two axis systems.



Teach positions:

To define a special position by moving the axes manually, please select

- the button  Teach all
- the window **Manual movement** will be opened
- move the axes with mouse click, keys on keyboard or on Operator panel to the desired position

After exiting with OK the actual position values are entered in the highlighted line.

The further five offered buttons contain also teach functions. After hints to convenient approaching orders the window "**Move axes**" is opened and you can execute the Teach-in.

Move to position:

A movement to the marked machine position in the list will be carried out. **Before starting** this function you should examine whether the movement **will be possible without danger**.

Options:

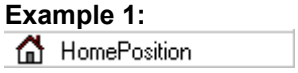
- **Approach order**

  Setting movement order

- **Special usage**

  Assigning a special usage to a machine position

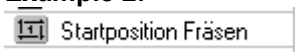
Example 1:



Defining this position as **Home position** please select the icon within the option "Special usage".

Activating e. g. the option „Move to Home position"  the axes will approach to this position.

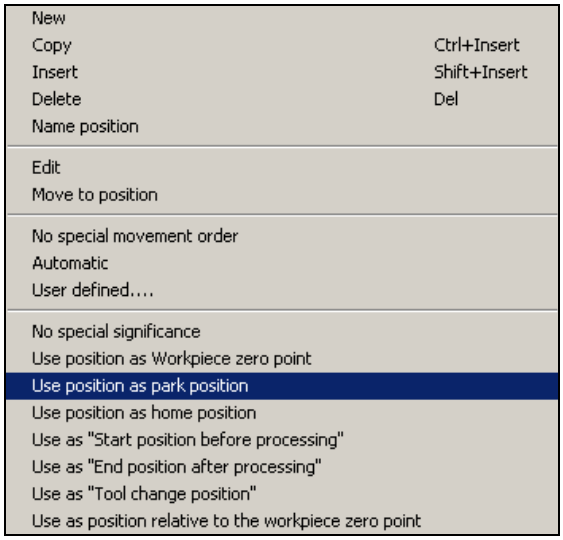
Example 2:



Activating with the Menu **Setup - Processing** the option *Before processing starts, move to position "Start of processing*"„ the defined position of the axes will be approached before executing the first instruction in the user program.

Tip:

You reach the explained functions faster if you mark the wished machine position with the mouse and then press the right mouse button.



Hint !

According to the explanations for creating a geometry file it will be referred here again that you can specify your geometry file suitable to the user program.

These frames to the user program are stored in the directory CNCworkbench\NCProg\Frame.

If you want to open this user geometry file and edit on screen like the standard geometry file, please execute the following steps:

- Menu **File - Open** or button 
- Select the file type frame file (*:FRA)
- Select in the directory CNCworkbench\NCProg\Frame your geometry file
- Edit your user geometry file just like a standard geometry file

Hint ! Please, use the standard frame StdFrame.fra for the new creation of your geometry file out of the directory CNCworkbench\Bin. Copy StdFrame.fra to the folder CNCWorkbench\NcProg\Frame, rename it and manipulate your frames according to your user program.

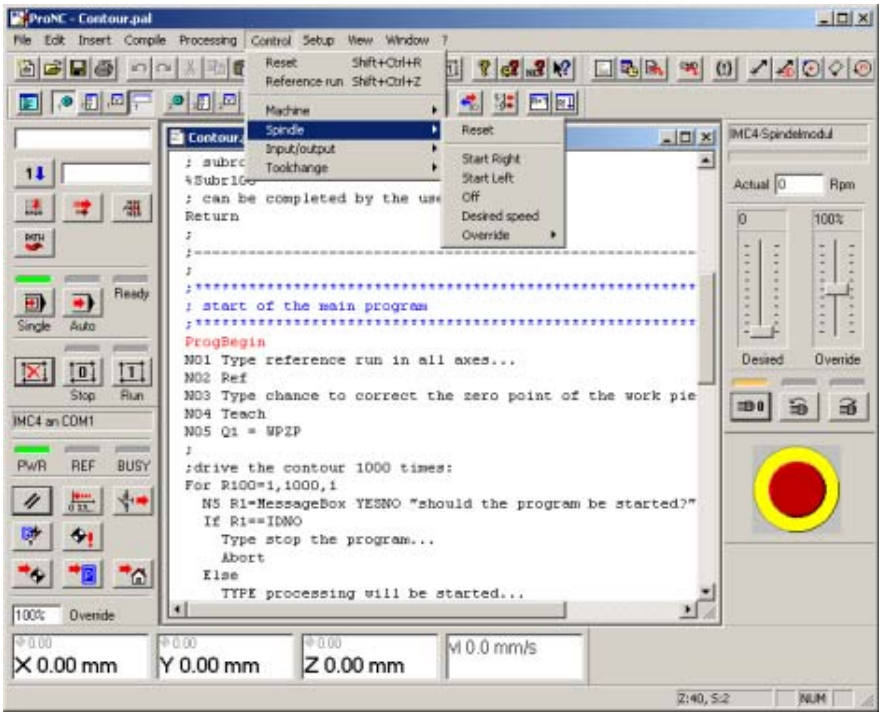
5.7.3.11 Move to machine positions

Menu **Control - Move to machine - positions** All symbolic machine positions stored in the standard geometry file StdFrame.fra which have gotten a special usage are displayed as a list by selecting of this menu function.
By your selection a movement is executed to the desired position.

Please notice, that the position is approached in the concerted order (approach order), and you shall always check, if a harmless movement of the tool is possible.

5.7.4 Spindle

Menu **Control - Spindle**



A continuous **setting** of the spindle speed with nominal value (absolute instruction) or with **override** (percent instruction) is possible, if the processing spindle is controlled with a frequency converter.

If the equipment doesn't have a spindle with a control by converter the spindle speed should be fixed at the spindle by hand.

Menu **Control - Spindle on counter clockwise**



With this function please switch on the spindle counter clockwise.

Menu **Control - Spindle on clockwise**



With this function or with the button on the spindle panel please switch on the spindle clockwise.

Menu **Control - Spindle - off**



With this function or with the button on the spindle panel please switch off the spindle.

Menu **Control - Spindle - target speed**

With the functions **Desired speed** or **Override** it is generally possible to define a value for the spindle revolutions per minute. This value is a default value; it gets effective only if the command is executed to start the spindle in the CNC user program without explicit declaration of the spindle speed.

Menu **Control - Spindle - Override**

Please define the spindle speed in relation to the fixed speed. The following scaling is possible:

- increase at 5%
- reduction at 5%
- minimum value 0%
- standard value 100%
- maximum value 140%

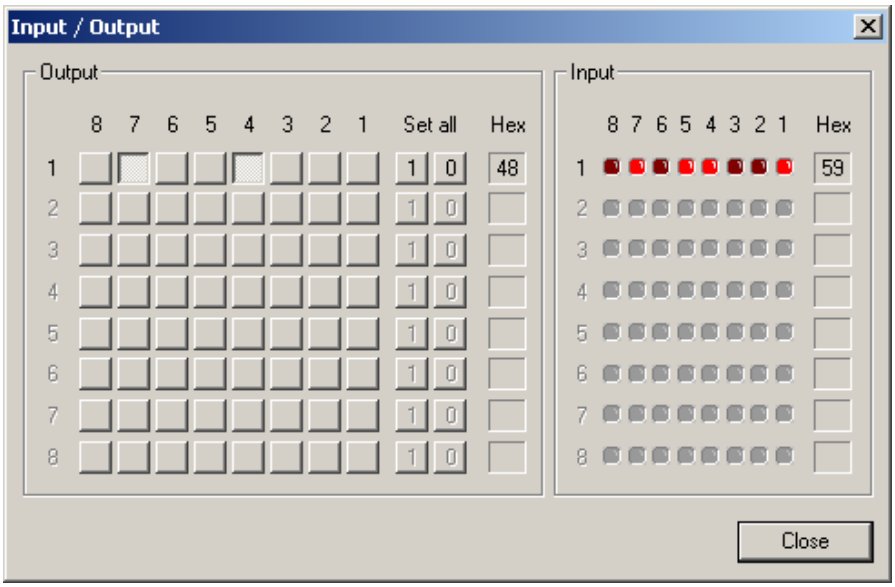
You can also choose the continuously setting with slide control on the spindle panel.

5.7.5 Input/Output

Menu **Control - Input/Output**



Selecting the function **direct access** a window opens:



You have the possibility to set single bits in a chosen output port.

At the same time a **display** of the current state of the **binary inputs** is shown.
In the program this function can be realized with the miscellaneous command „Set output port“.

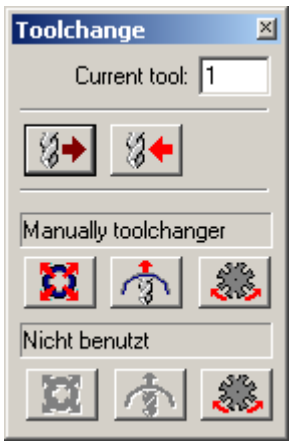
More: Programming Instruction: 3.1.2.11 Set outputs

5.7.6 Toolchange

Menu **Control - Toolchange**

Corresponding to the configuration of the module motion control, spindle module and the I/O module the set up of the tool changer is executed with the Menu **Setup - Control - Toolchange**.

A numbering of tools allows the manual control of operator actions at tool change obtained to the select tool.
The following functions are available:

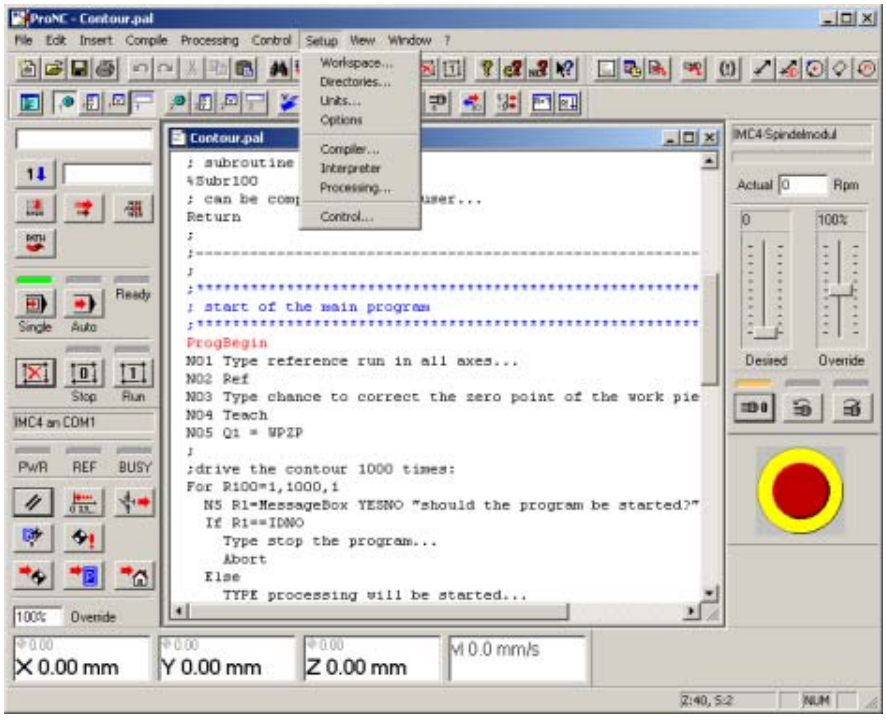


→ Tool put away
(categorizing in a tool magazine)
→ Get a new tool
(from a tool changer)

→ Tool clamp open / close
→ Tool hood of a tool changer open / close
→ Tool magazine rotate

5.8 Menu Setup

Menu Setup:

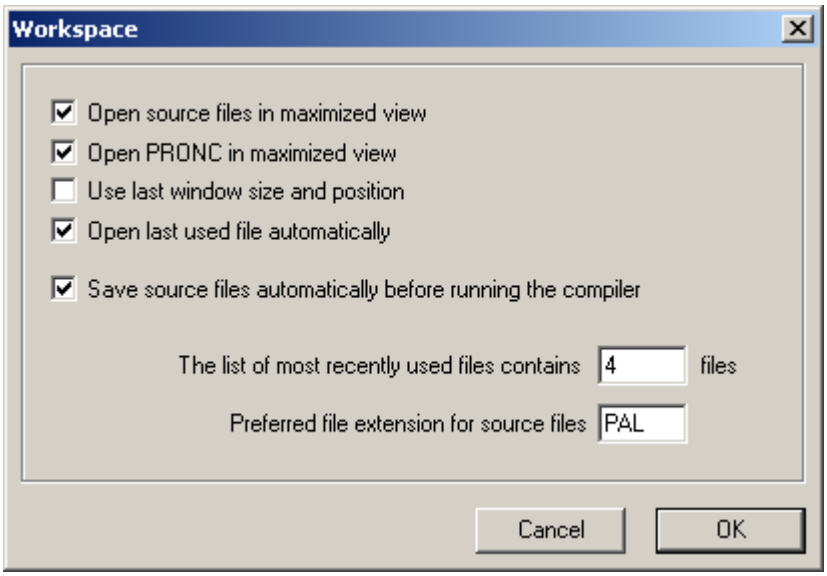


MENU „SETUP“

5.8.1 Workspace

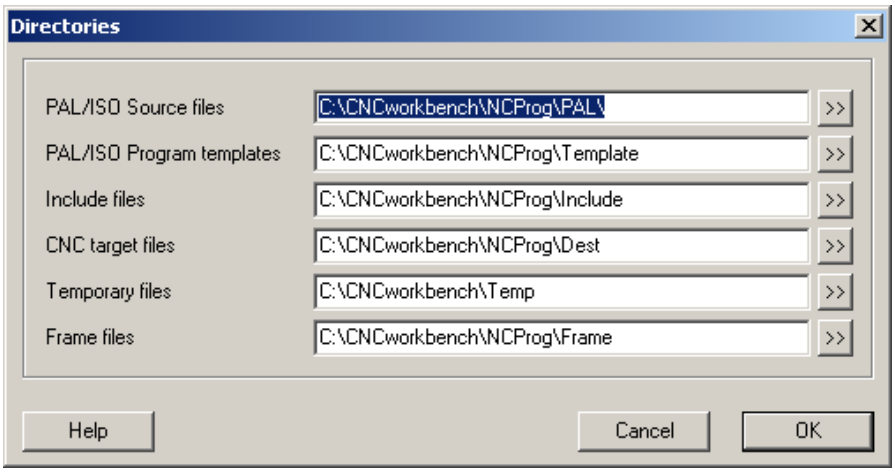
Menu **Setup - Workspace**

Define your special parameters according to the processing task for the **file editing**.



5.8.2 Directories

Menu **Setup - Directories ...**



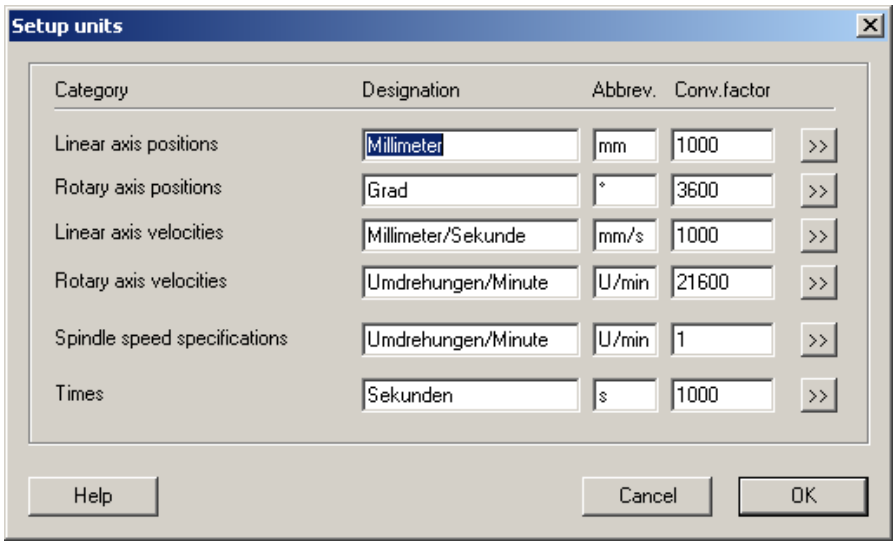
The directory structure of the components and user programs of ProNC is displayed after installation of software.

By double click on the button  you can orientate in the whole directory of your workstation.

You have also the possibility to store, e. g. your special frame files in a selected folder. Inform the system ProNC about a modification of directories over this window.

5.8.3 Units

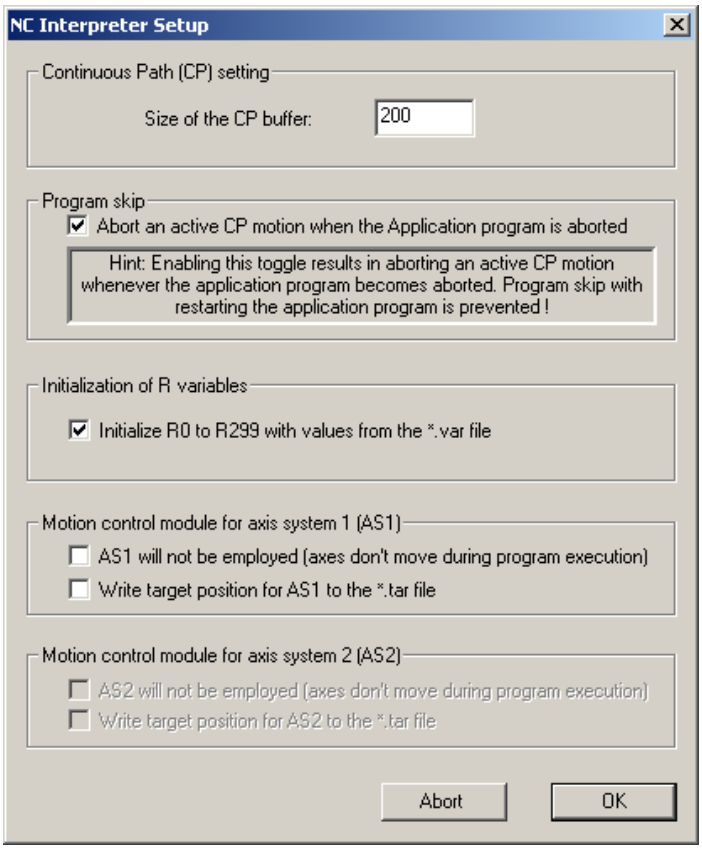
Menu **Setup - Units ...**



About this function a change of the measure units is possible for the axis position values, the linear and rotatory velocities of the axes, the revolutions per minute of main spindles and e.g. the dwell time. Choose your favored unit from the list.

5.8.4 Interpreter

Menu **Setup - Interpreter**



- **Path calculation**
  The size of the path buffer should be changed only by service staff.

- **Program skip**
  If this control box is activated, a path motion is immediately aborted, when the application program is aborted; but a program skip won't be possible then.

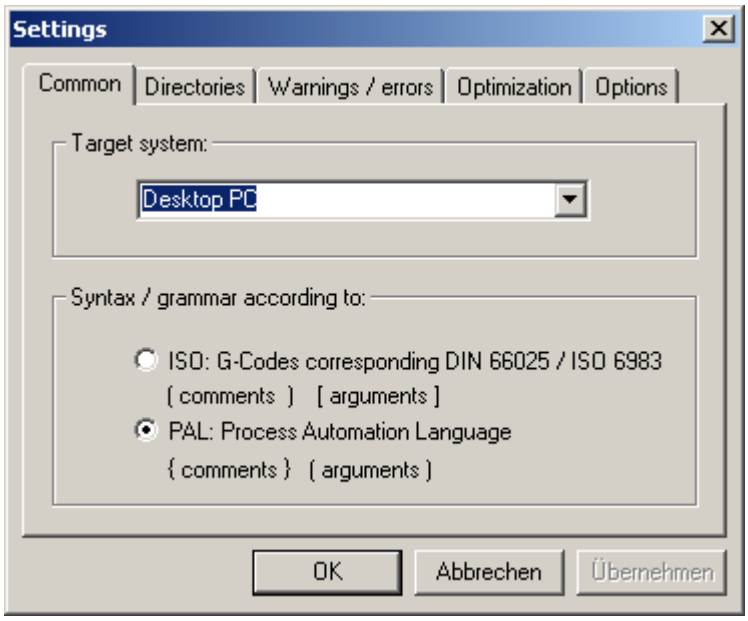
- **Initialization of R variables**
  This control box is deactivated only if you want to start your user program with R variables setting up with the value 0.0.

- **Motion Control Module axis system 1**
  The activation of this control box starts the program processing without movement of axes. Therewith a simulation of the programmed motion commands gets possible.
  The c**oordinates of the simulated movements** can be stored in the file *.tra by activating the option "Write target for AS1 to the *.tra file.

  The file *.tra gets the name of the user program automatically and is located in the directory CNCWorkbench/NCProg/Dest after completion of processing.
  The file *.tra can be opened with the **graphic program NCD** (NC display) and the motion path can be displayed. You find the program NCD in the directory CNCWorkbench/Bin.

5.8.5 Compiler

Menu **Setup - Compiler**



- **Please choose ISO for DIN/ISO user programs**
  Hint: Comments about several lines are written with round brackets

- **Please choose PAL for PAL user programs**
  Hint: Comments about several lines are written with square brackets

5.8.6 Processing

Menu **Setup - Processing**

About this menu can be agreed, which **functions** shall be executed **in addition to the user program** before begin and after the end of the processing.

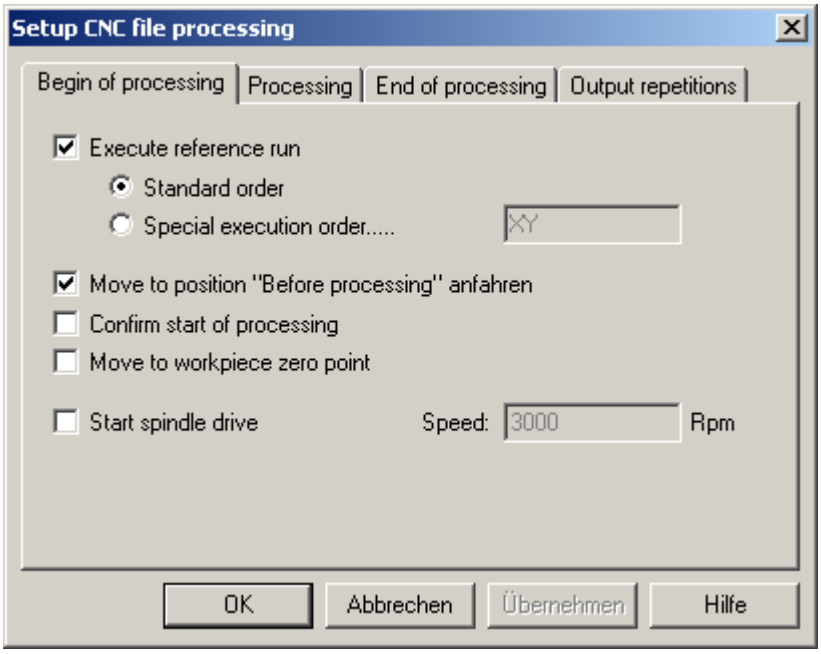
- Execute a reference run in standard order (Z,X,Y) or user defined sequence

- Approach to selected positions; the definition of these positions carries out in Menu **Control - Machine - Setup machine positions** by the button *Special usage* .

- Creation of a stop immediate before processing of the CNC file; you decide in the dialog whether the file shall be started or the process shall be canceled

- Approach to current work piece zero point

- Switch on / off spindle, definition spindle speed



Hint:

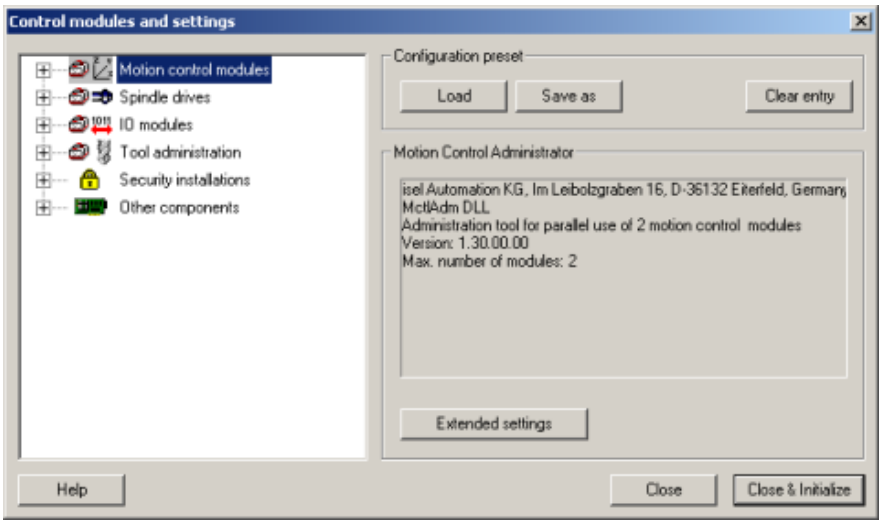
The setting options for machine positions are valid only when the positions were defined in the standard frame.
This means a default work piece zero point which shall be approached automatically must be agreed as **position** in the **standard frame** and must have the icon .

Usages (assigned in an user frame e. g. as a work piece zero, as a park position, as a home position) are ineffective.

More: 5.7.3.10 Menu Control - Machine - Setup machine positions

5.8.7 Control

Menu **- Setup - Control ...**



Configuration and installation of the modules:

This window offers the **dialog aided** possibility alike the system controlling in Windows, to configure the special Module-DLLs **including the required initialization files**.

In the rule the **modules and settings are factory-installed** according to the **customers target controller**.

Changing or extending the equipment **the user is able** to configure the Hardware/Modules with the corresponding Motion Control, Spindle control, IO control and Tool changer (DLLs) himself.

More: 1.2 Scope of delivery and installation

Module parameters:

A selected motion control is defined by the following information:

- Module naming:
  please enter a name of the equipment/module
- ***Dynamic Link Library for access to the module:***
  please select the DLL-file according to the control in your equipment with a click at the button .

You find the relevant files e. g. in the directories:
\CNCworkbench\Control\Imc4 for modules with IMC4
\CNCworkbench\Control\StdSV1 for modules with UPMV4

You find a further support for selection of the required file in dependence of the hardware in the section:

More: 3.4.2.1 Module types and DLLs

- ***Module initialization file:***
  select the corresponding initialization file for the configured module with a click on the button .

You find the relevant files e. g. in the directories:
\CNCworkbench\Control\Imc4 for the controller IMC4

For more support of the configuration you find the following functionalities for each module:

- Versions request with common information about the special module DLL
- Set up of relevant parameters
- Diagnostics for checking the functions of the Module-DLL and the equipment
- Status to determination and checking the current state of the module

After changing please use always the button 

; therewith all new parameters become active.

More: 3.4 Configuration

Hint !

Please note that at every start of ProNC only the modules are initialized, which are marked with ☑.

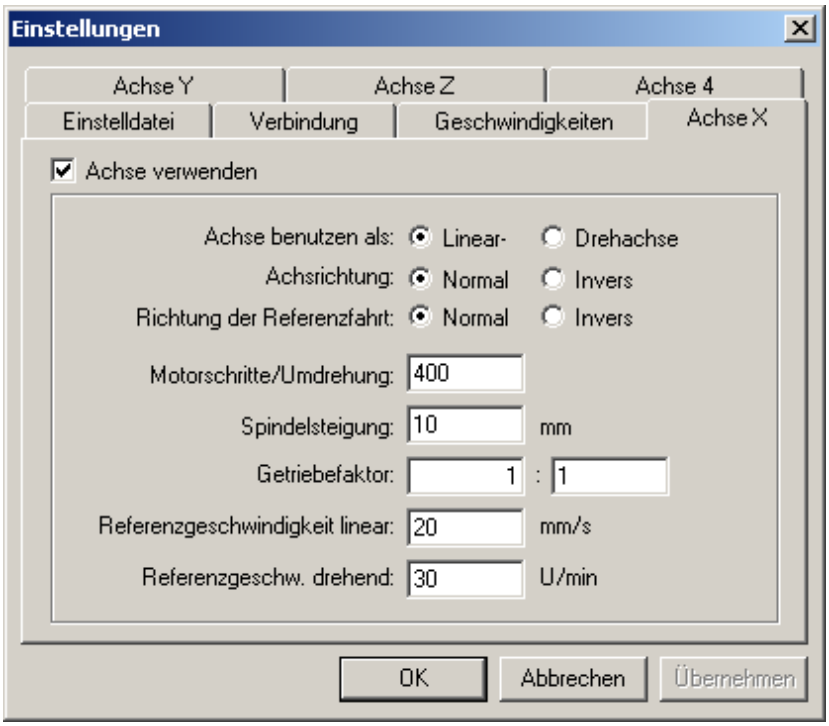
If you should want to blank modules at times, remove the marking and at the next restart of the program these modules are deactivated.

5.8.7.1 Motion Control Module

The module for the motion control is required to pursue the mechanical system.
The most important parameters of the connected axes are part of the setup dialog which is activated with the button Setup.

e. g:
→ pitch per axis
→ gearing
→ axis velocity

Parameters in Setup:



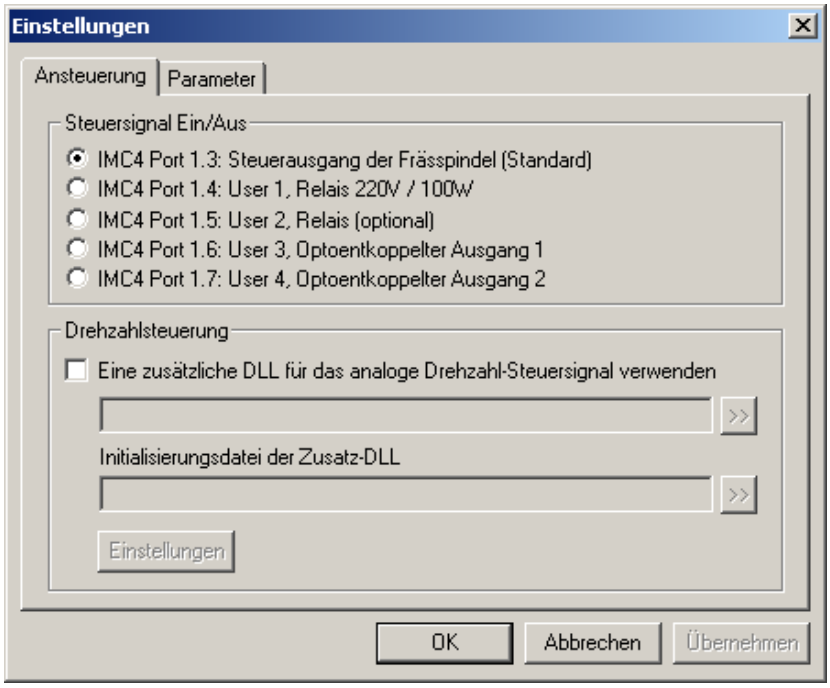
Hint:

After the installation a spindle pitch of 10 mm is defined.
If the spindle pitch of the axes **in your plant differs from the adjusted pitch**, please **enter** in this dialog the current spindle pitch (modification per initialized axis).

5.8.7.2 Main spindles

The module for the converter is competent to control the main spindle. At most four main spindles can be controlled in ProNC.

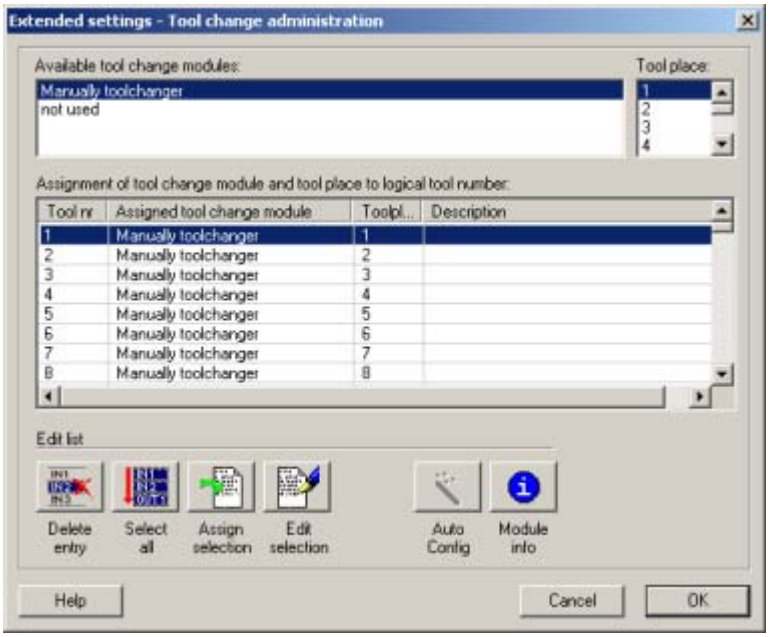
Parameters in Setup:



5.8.7.3 Input /output module

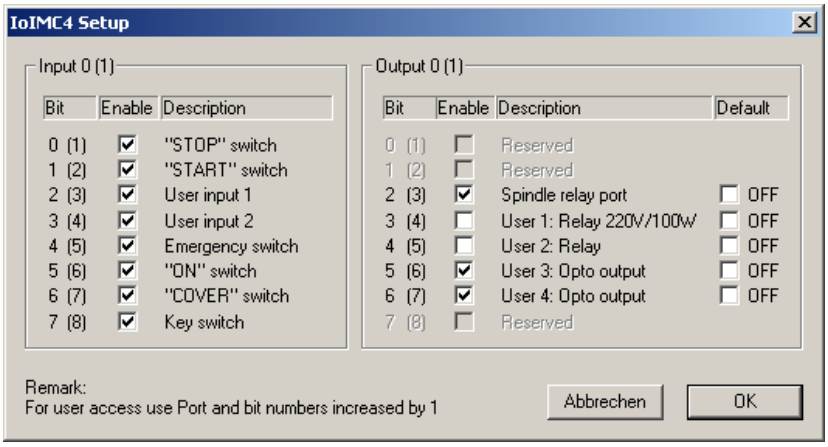
The input /output module executes the binary and/or analog input and/or output.

Before you select the input /output module, you reach by double click on the button Extended settings in a dialog which makes it possible to assign further IO modules and their local port number to the logical port number of the module.

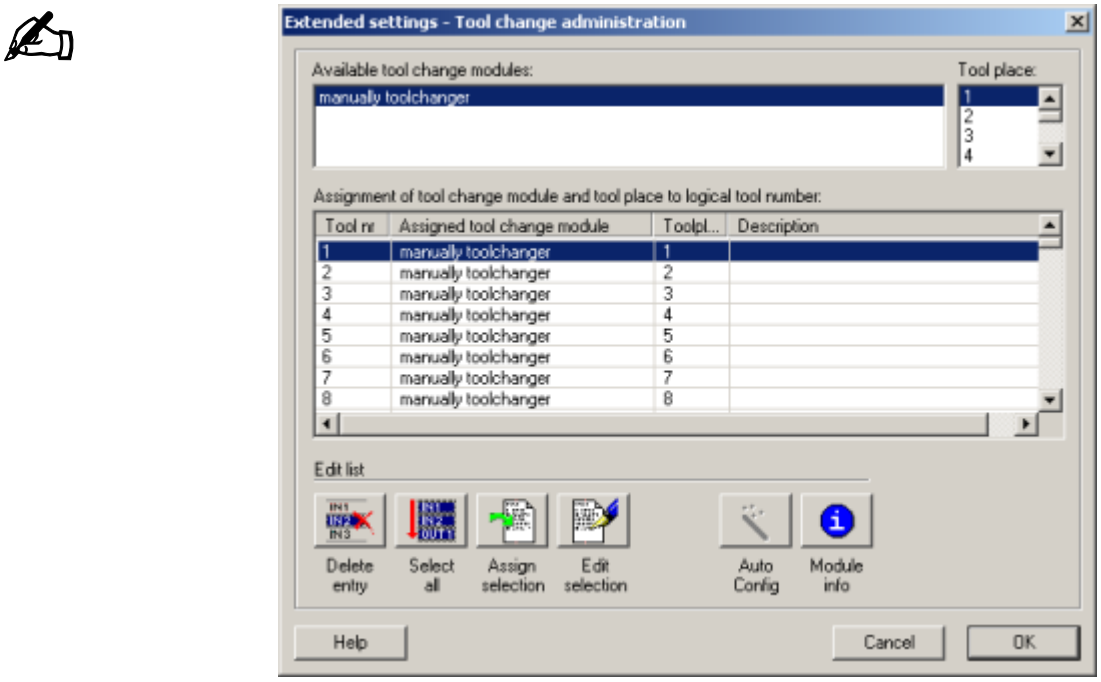


Parameters in Setup:

After selection of the button Setup the assignment of the inputs and outputs e. g. start/stop key, key switch …. for the chosen I/O module will be displayed.

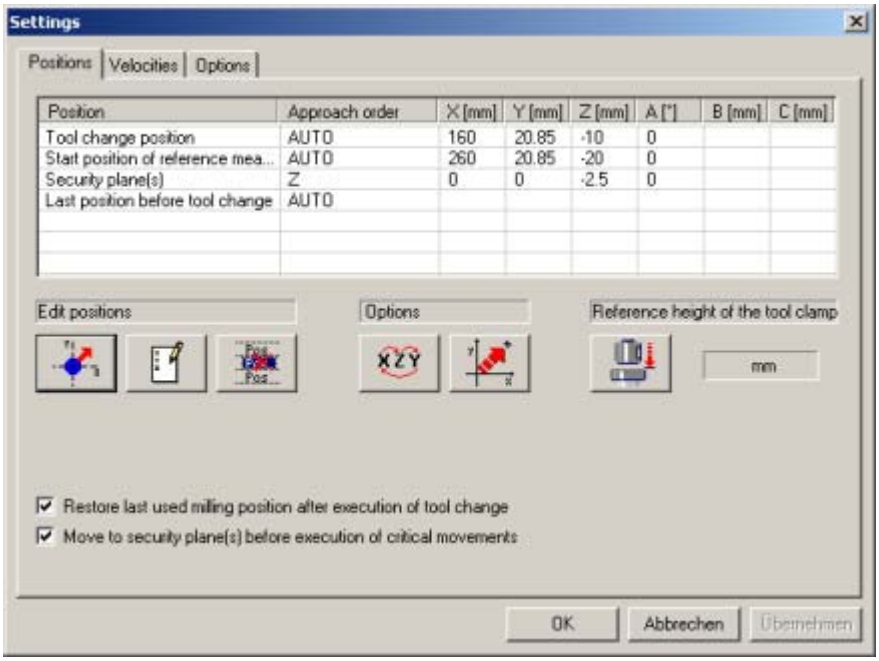


5.8.7.4 Tool administration



This dialog you can reach with the Menu **Setups** - **Control** - Tool administration - Extended Setup.
If necessary please execute an assignment of the tools to the according tool place.

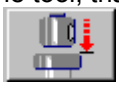
Installation of the positions for the tool change:



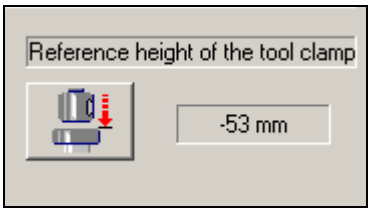
Setting up the tool changer the positions

- **Tool change position** (an advantageous position of the axes to change the tool at manual tool change)
- **Start position for referencing** after tool change (position about the length measuring calliper)
- **Last position before changing** (optional)

can be adapted to your equipment/machine by teaching or editing.

For the referencing of the tool, that means for the tool length determination please push the button .

After moving to this tool change position please clamp a reference tool to carry out the referencing.
The **measurement of the reference height** will be executed automatically; the detected value as basic value for each tool change is displayed.

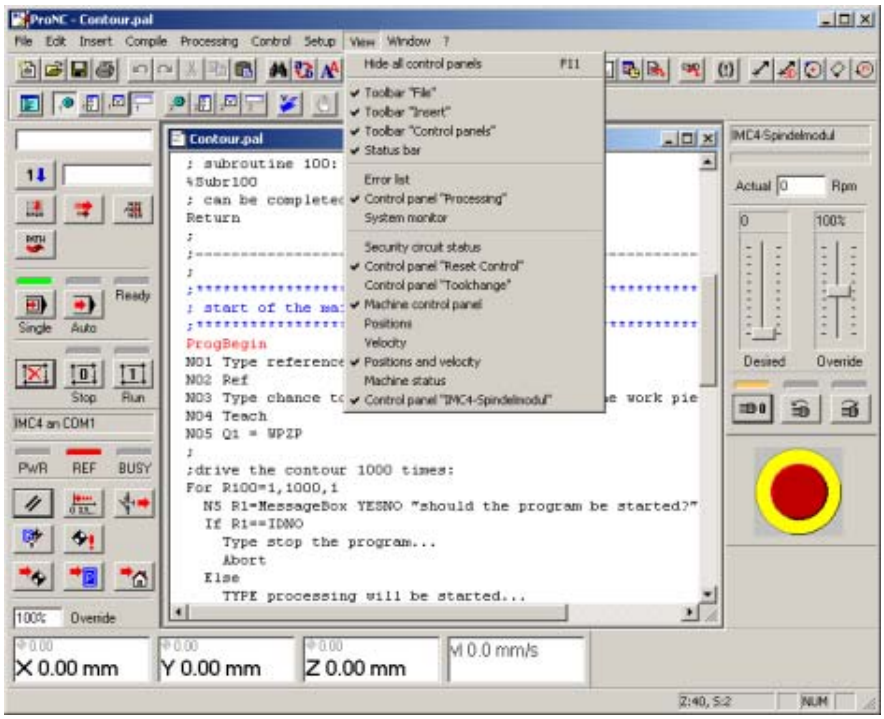


Hint: Detailed information to installation of the tool changer including measurement of the reference height with help of the measuring calliper you get in the working session Menu 6.1.6 Setting tool change.

Hint ! In this window "Setup" you can select the index card "Options" for the possibility to remove the measurement of the reference height, to turn off the spindle before tool change, for declaration of the security plane.

5.9 Menu View

Menu **View**



Select this menu, if you will fade in/out the symbol bars.

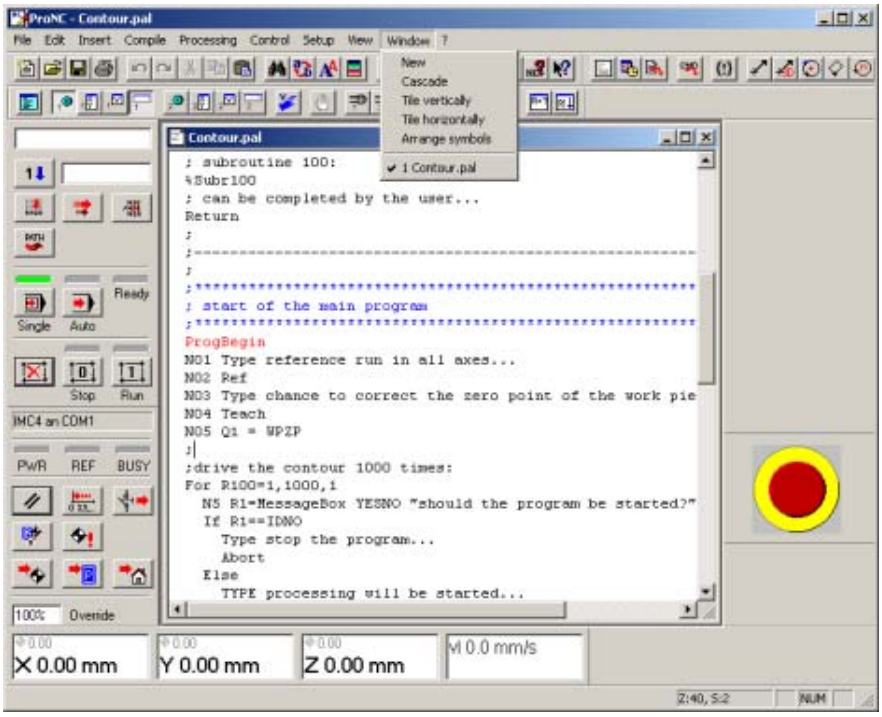
The visible symbol bars are marked with a token.
To remove one of the symbol bars or display window please click on it.

It is pointed out that always **standard and additional windows for two equipments** will be offered to the user, if the equipment **controls two plants**.

You can hide all operating panels with the button , this could be useful e.g. during the program editing.

5.10 Menu Window

Menu **Window**



With this function please **select among several opened user programs**, each one in a separate window.
In addition you can change the arrangement and the size of the programs in the working area.

6 Working session

6 Working session

In the following chapter we introduce **in two examples** the **application flow** of the required operations e.g. to mill a contour.
In dependence of the procedure to create this program two versions are described:

1. The user program was created with the CAD/CAM program ISY

A program in the NCP format generated by ISY is a simple program with all required commands for the processing operation without loops and branches.
Parameters to the machine setup are determined and fixed by the operator. These required inputs are explained detailed in the example.

2. The application program was created textually with the integrated editor

In addition to the commands for creation of the contour this program contains a loop for program repetition, an operator dialog, the determination of the zero point in the user frame file including a correction possibility by the operator during the run time of the program. Because positioning commands are included before and after of the actual processing, all options in the Menu **Setup - Processing** must be deactivated.

More: 5.8.6 Menu Setup - Processing

More example programs (created textually) to learn and understand possible program flows you find according to your chosen programming language (PAL or ISO) in the directories:

\CNCworkbench\NCProg\Pal\Sample
\CNCworkbench\NCProg\Iso\Sample

6.1 Processing of programs created with ISY

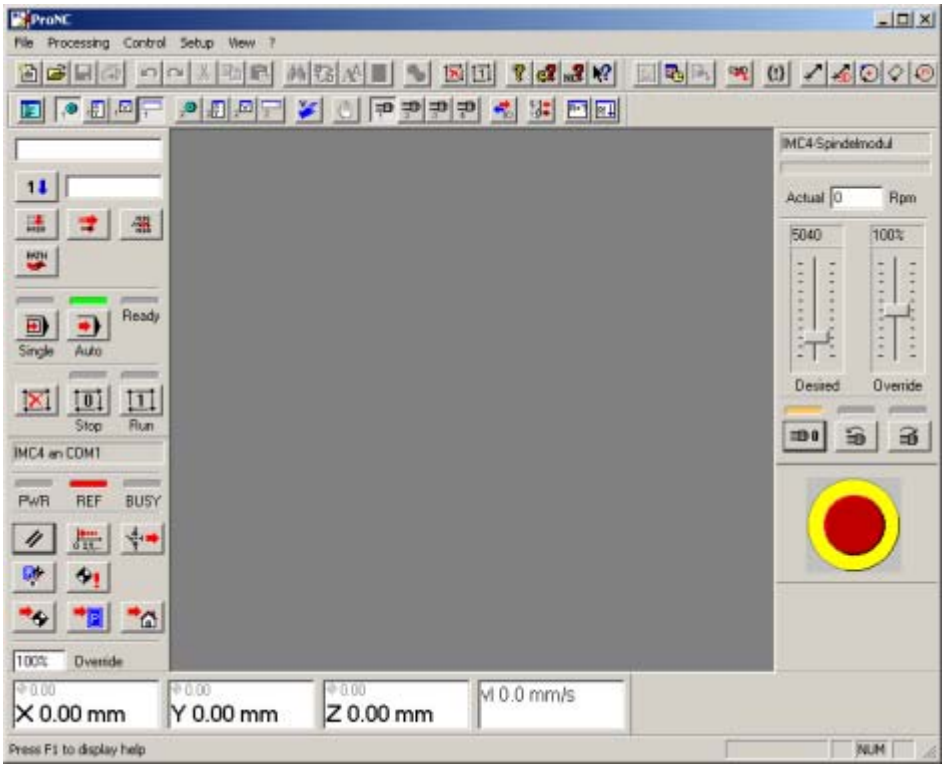
6.1.1 Start of ProNC

Please unlock the emergency switch at the controller, at the machine or at power supply.
Switch on the power in front of the equipment using the green key (POWER), close the cover.

Controller is powered up, PC boots up.

Please start **ProNC** with double click on the **program icon** .

ProNC will be loaded with the start screen.



More:
1.4 Program call
1.6 Screen layout

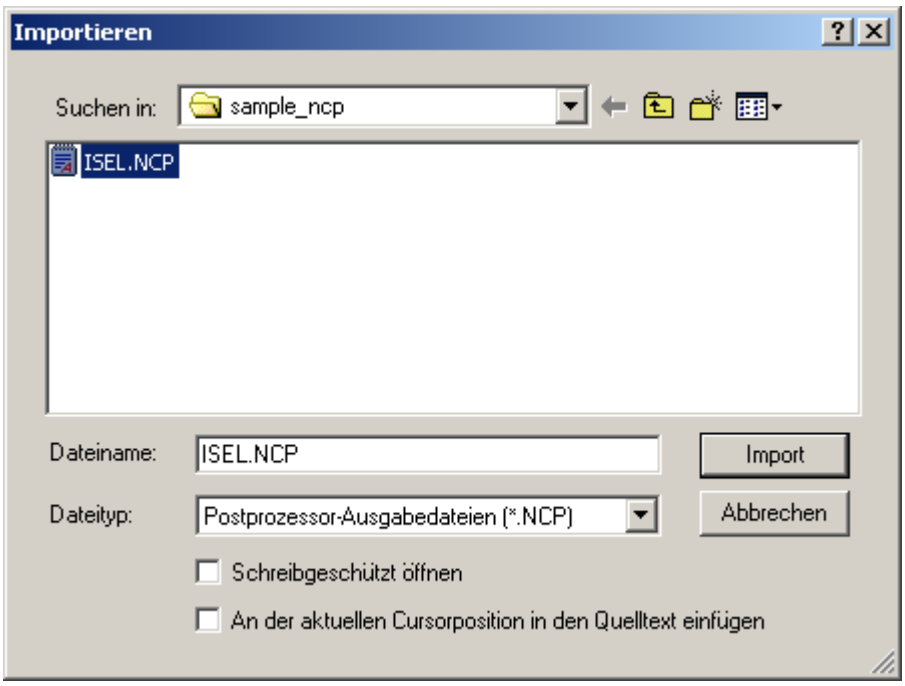
6.1.2 Open user program

With the program IsyCAM 3.0 the text "isel" was created with all necessary CAM-parameters and is located as example program **isel.ncp** in the directory \CNCworkbench\NCProg\Ncp. The work piece zero point is marked by darts ⇨X and ⇧Y.



Please open the file in NCP format always about the Menu **Import.**

Please activate the Menu **File - Import,** select under category file type the extension for postprocessor file (*.ncp) and load the specific ISY program with a double click.



The file is converted and henceforth it is available in PAL syntax to a further using.

Please compile now the converted file, so that you get a CNC program which is ready to start.

Please select the button  or the Menu **Compile - Compile of isel.ncp**.

! *In the fault case, i.e. syntactical errors are recognizing in the source program a error list is shown to you. Please correct these errors, if required also with the help of the programming instructions, and compile again.*
Only a syntactically faultless source file can be processed after the compiling.
Without message the program is syntactically correct and can be started.

More:
5.5 Menu Compile
5.5.1 Error window

6.1.3 Perform the ready state of the equipment

Unless not passed already please unlock the emergency switch at the controller, at the machine or at power supply and close the cover.

Please execute a

Reset:

button , Menu **Control - Reset** or **operator field Reset Control** 

with a resulting **reference run**.

Please **pay attention** to a **harmless possibility** to approach the axes.

! *If you will get a failure message e.g.*

„The module IMC4.DLL was not initialized“
or
the axes do not approach not at all -

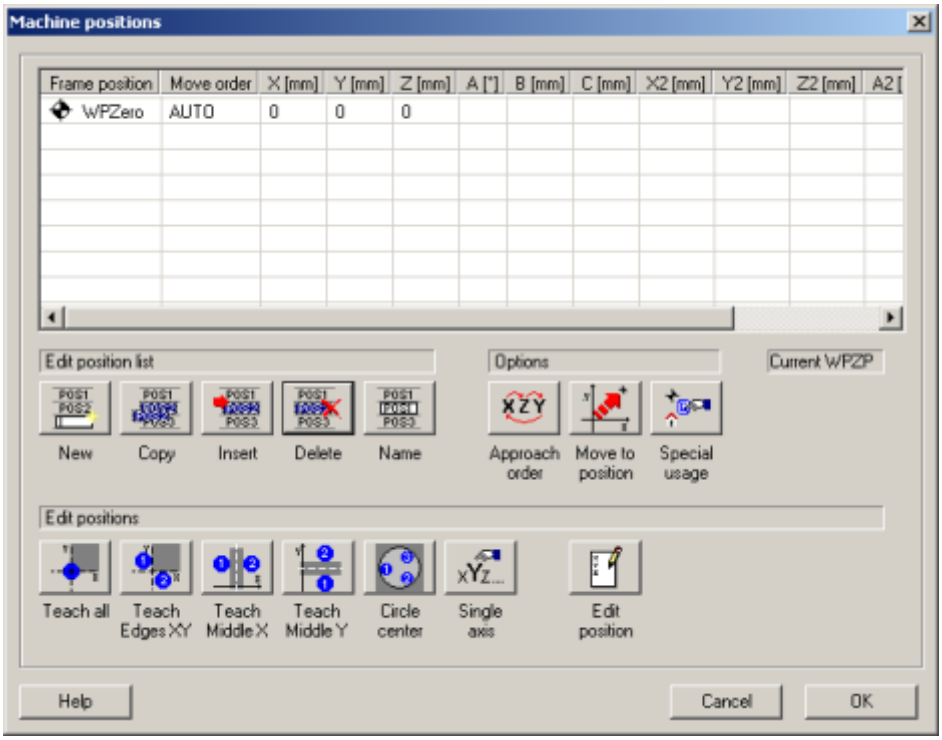
is this a hint for you, that the motion control and the settings are not correct.

*In the dialog box "**Control modules and settings**" you have the possibility, to get an overview about the configured modules and the defined settings.*
If necessary you can change settings.

6.1.4 Determine work piece zero point

As result of the created file with ISY you know the dimension of the work piece and the position of the zero point on your work piece.
In this example you need a specimen with the dimensions at least 100 x 60 mm.
The zero point was defined in the lowest point of the letter i.
After clamping the work piece, it is necessary to define this work piece zero point at the equipment.

Select the Menu **Control - Machine - Set up machine positions or the button** .



1.

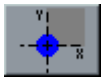
A frame position with the coordinate values zero is provided as standard. You can use the position and you can teach and edit the work piece zero point how described in section 2.
If you like to install a new machine position, please select next to the term "Edit position list"

the button  or the key combination **Alt + N** (**N**ew). With that a new position is installed.

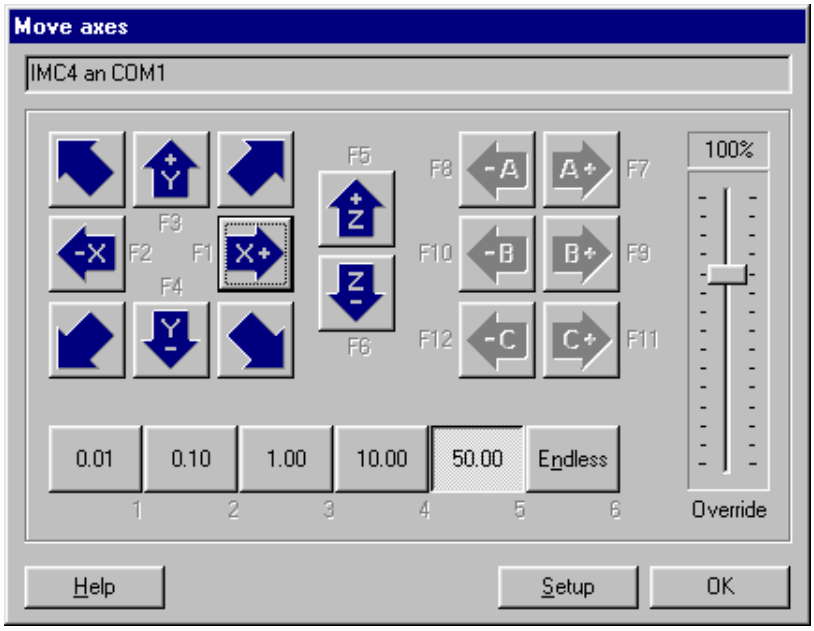
Please name the zero point e. g. WS1 and quit with OK.

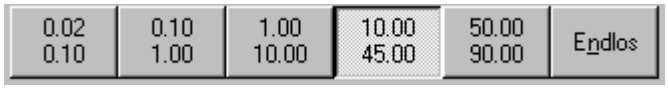
2.

If you like to teach/approach manual, please mark the wished frame with the mouse and choose then

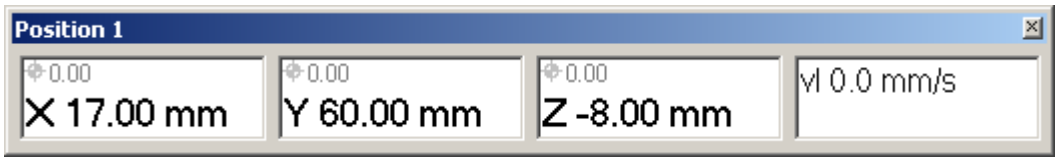
the button  or the key combination Alt + T (Teach all).

The window for the manual teach of the work piece zero point is displayed:



Please select with the buttons  the wished step width of the movement (unit = mm).

Approach the axes, that the tool peak touches the surface of the work piece directly. Approach the X- and the Y-axis to the desired start position and sink the Z-axis carefully as long, as the work piece is touched. Quit the dialog with "OK".
In the position window the approached coordinates are displayed e.g.:



Quit with "OK", the window machine position is displayed again and the position values are assumed.

To define this position as zero point for all movements with absolute measurement, please select:

3.

the button  or the keys **Alt + V** (special usage) and assign the usage "Work piece zero point" to the position. Now in front of the frame name the token  is displayed to mark this position.

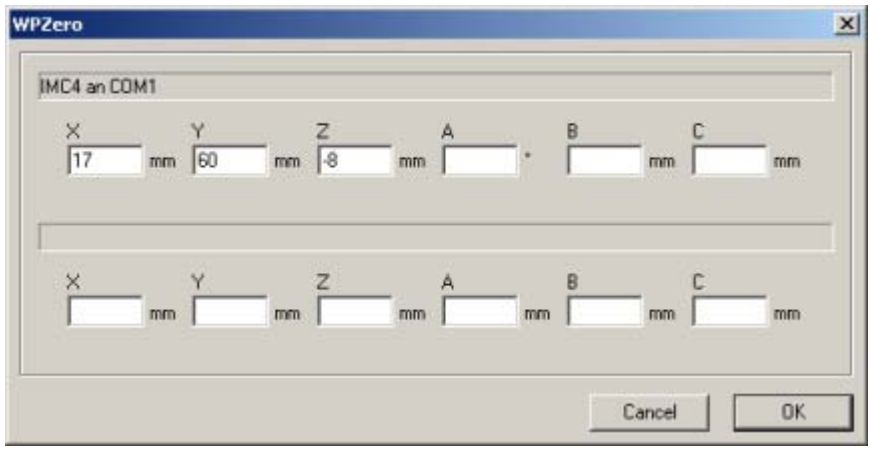
Caution: Only the marking **of your teached zero point as work piece zero point** ensures, that the options in the Menu **Setup - Processing** will be effectively.

4.

To define the approach order of the work piece zero point, please activate the button  or the keys Alt + S.
The selection "Move automatically" defines the option AUTO (default approach order X+Y (diagonal), Z).

Hint ! If you do not want to approach the work piece zero point with hand (teach), but you want to edit the coordinate values, please double click in the window "Machine positions" on the marking position *or* push the button  *or* use the keys **Alt + e** (Edit position).

In this window enter the position of the zero point with keyboard.

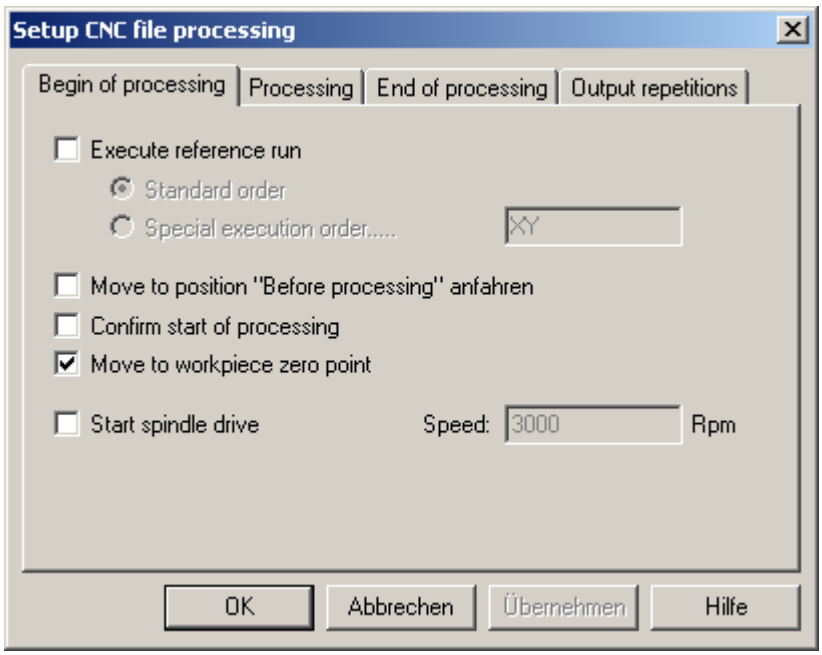


After quitting with "OK" the values for the zero point are assumed and the window **Machine position** is displayed again.

6.1.5 Settings for processing of the CNC file

Select the Menu **Setup - Processing** and define, which actions shall be done before/after the processing of the user program.

Select the index card "Begin of processing":



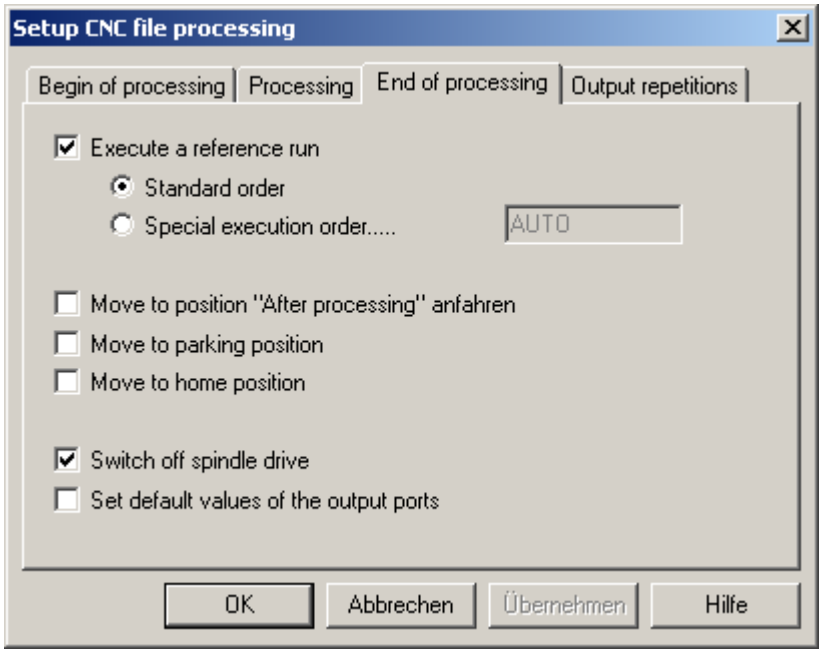
Activate the function:

- Move to work piece zero point.

This has the following effect:

Before executing the first command in your NCP file by the interpreter the work piece zero point, defined in "Machine positions", is approached.
If no work piece zero point was defined in the window "Machine positions", the processing starts running out from the actual position of the tool.

Select the index card "End of Processing":



Activate the functions:

- Execute a reference run
- Switch off spindle drive

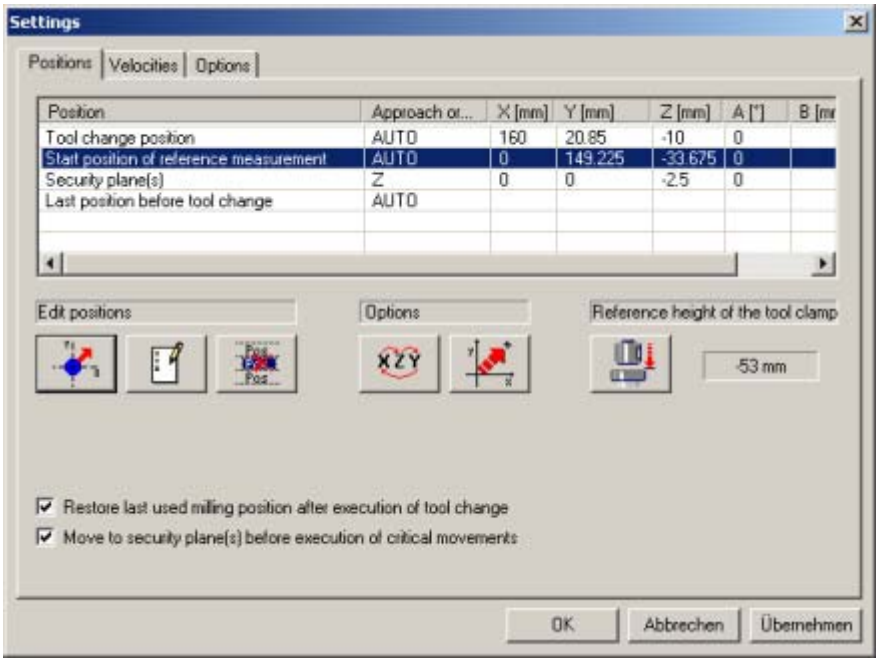
The defining of these functions is not connected to the respective user program.
If you start another user program you should check up, if the adjustments should be kept.

6.1.6 Setting tool change

Select the Menu **Setup - Control** and click with the mouse to the tool administration/changer station 1.

After activating the button 

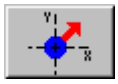
the input window for tool change and reference positions is open.



Please define in according to the processing task the coordinates for the best position of the manual tool change by moving the axes with hand or entering with keyboard.
Please mark the line

Tool change position	AUTO	160	20.85	-10	0

with the mouse.

Then please select the button . The menu for manual movement of the axes is displayed.
Please do the same similar to define the work piece zero point.

More: 5.7.3.9 Menu Control - Machine - Manual movement

After selecting the button  you can enter the coordinates with the keyboard.

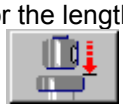
More: 5.7.3.10 Menu Control - Machine - Setup machine positions

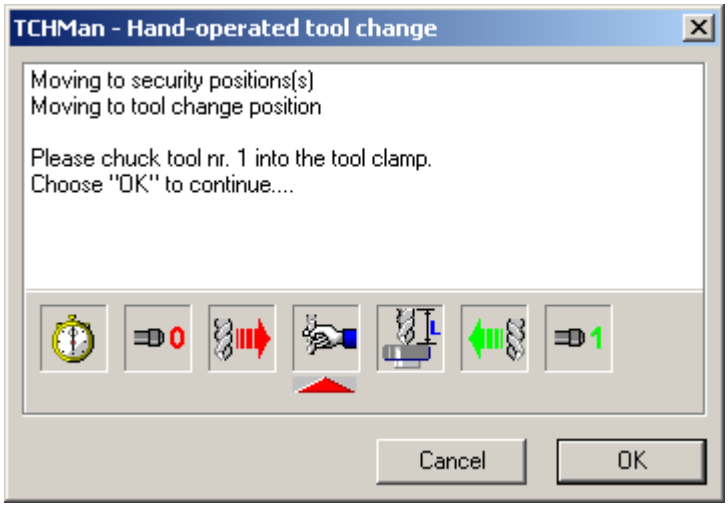
Please define the start point to reference the tool clamp by approaching the axes with hand. Choose such a position, that the **longest tool can be placed**. During tool change a length measurement of the new clamped tool, starting from this position, will be executed automatically.

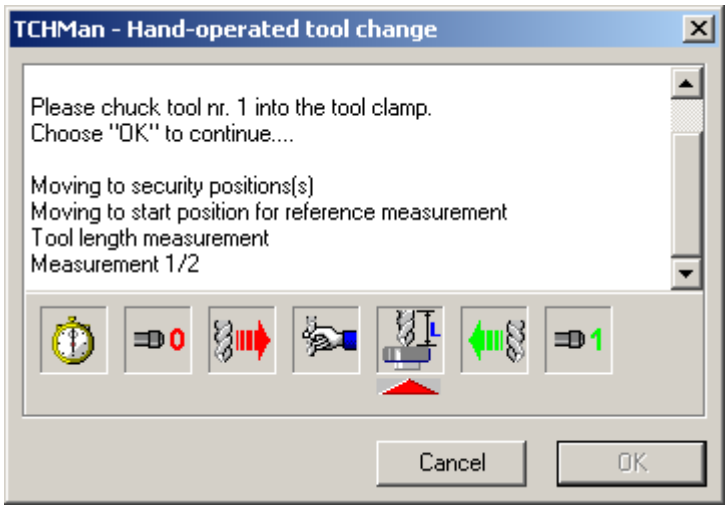
Select the line

Start position of reference measurement	AUTO	0	149.225	-33.675	0

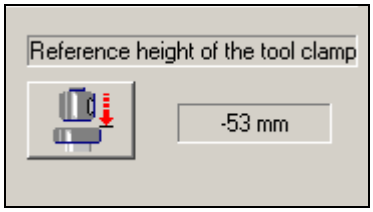
Teach the start position for referencing just as the teaching of the tool change position.

Please determine the height of the clamp with a reference tool as base for the length measurement after tool change . Clamp a reference tool, push the button  and the measurement of the reference height will be executed automatically.





After referencing the value will be enlisted and it is visible in this window.
Example:



6.1.7 Program start

We recommend to use the mode „**Single step mode**“ (step by step) during the testing phase.

Please adjust this mode with the button  **or** with the Menu **Processing** - **Single step**.

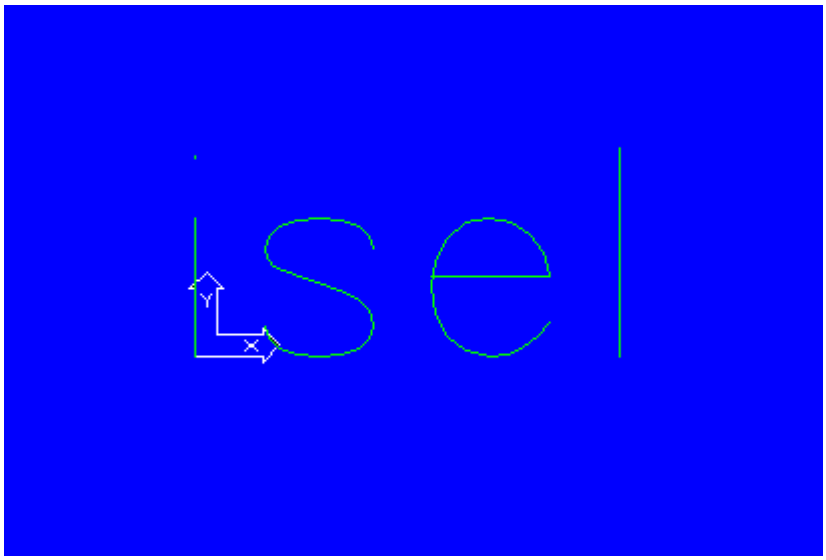
This mode enables you, to follow the process step by step. After each command you can change to the mode "Automatic" (button ).

Start the processing of the program with the button

 or with the Menu **Processing - Start**.

The first line of the program is highlighted with a beam. After every click on the button start  *the marked line will be executed.*
In this way please follow also the annotations displaying in the status line of the screen.

After finishing the milling process, a reference run will be executed and the spindle drive will be switched off according to the setting "End of the processing".
The result of your processing should correspond to this picture.



6.2 Textually programming and processing

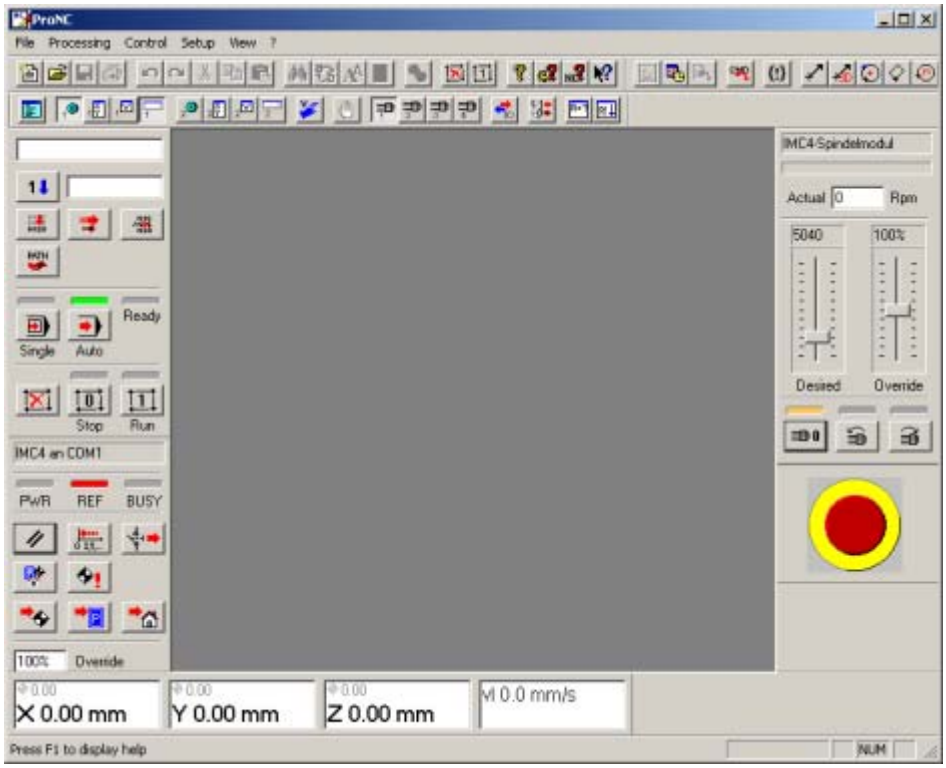
6.2.1 Start of the program ProNC

Please unlock the emergency switch at the controller, at the machine or at power supply.
Switch on the power in front of the equipment using the green key (POWER), close the cover.

Controller is powered up, PC boots up.

Please start **ProNC** with double click on the program icon  or about the menu bar Start - Programs - CNCworkbench - ProNC.

ProNC is loaded with the start screen.

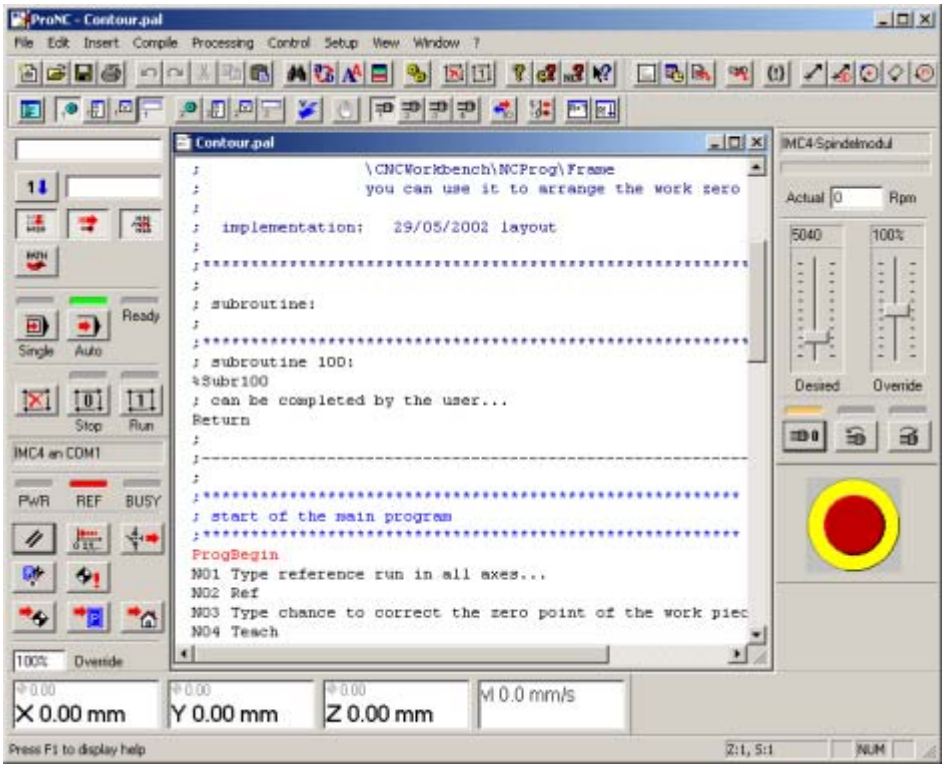


More:
1.4 Program call
1.6 Screen layout

6.2.2 User program open and compile

Activate the Menu **File - Open** or the button  and load the program **contour.pal**. out of the directory CNCworkbench\NCProg\Pal\Sample.

*The program **contour** is displayed in the editing screen area*.



More:
5.2.2 Open file
3.1 The editor

Please compile the file contour.pal, so that you get a CNC program which is ready to start.

Please select the button  or the Menu **Compile - Compile of contour.**

! *In the fault case, i.e. syntactical errors are recognizing in the source program a error list is shown to you. Please correct these errors, if required also with the help of the programming instructions, and compile again.*
Only a syntactically faultless source file can be processed after the compiling.
Without message the program is syntactically correct and can be started.

More:
5.5 Menu Compile
5.5.1 Error window

6.2.3 Perform the ready state of the equipment

Unless not passed already please unlock the emergency switch at the controller, at the machine or at power supply and close the cover.

Please execute a

Reset:

button , Menu **Control - Reset** or **operator field Reset Control** 

with a resulting **reference run**.

Please **pay attention** to a **harmless possibility** to approach the axes.

! *If you will get a failure message e.g.*

„The module IMC4.DLL was not initialized“
or
the axes do not approach not at all

is this a hint for you, that the motion control and the settings are not correct.

*In the dialog box "**Control modules and settings**" you have the possibility, to get an overview about the configured modules and the defined settings.*
If necessary you can change settings.

More:
5.7.1 Menu Control - Reset
5.7.2 Menu Control - Reference run
5.8.7 Menu Setup - Control

6.2.4 Program start

We recommend you to use a soft material e. g. polystyrene with the dimension of approx. 200 mm length and 120 mm width to visualize the contour resulting after processing of the file.
The programmed contour has a dimension of 180 x 85 mm.
The start point of the contour is X=20, Y=15 relative to the current **work piece zero point**. The current **work piece zero point is defined** in this example program in the **user frame file contour.fra**.
You don't have to fix the **work piece zero point** in the **standard frame** (window machine positions) and not before the start of the program.
A stop happens at **the beginning of the program;** the **work piece zero point** can be **defined** by teaching or editing.

We recommend to use the mode „Single step mode“ (step by step) during the testing phase.

Please put in this mode with the button  or with the Menu **Processing - Single step**.

More:
1.6.2 Screen layout - Standard windows
5.6 Menu Processing

Start the processing of the program with the button

 **or** with the Menu **Processing - Start.**



The first line of the program is highlighted with a beam. After every click on the button start  *the marked line will be executed.*
In this way please follow also the annotations displaying in the status line of the screen.

6.2.5. Statements to work piece zero point

After start a reference run is executed.

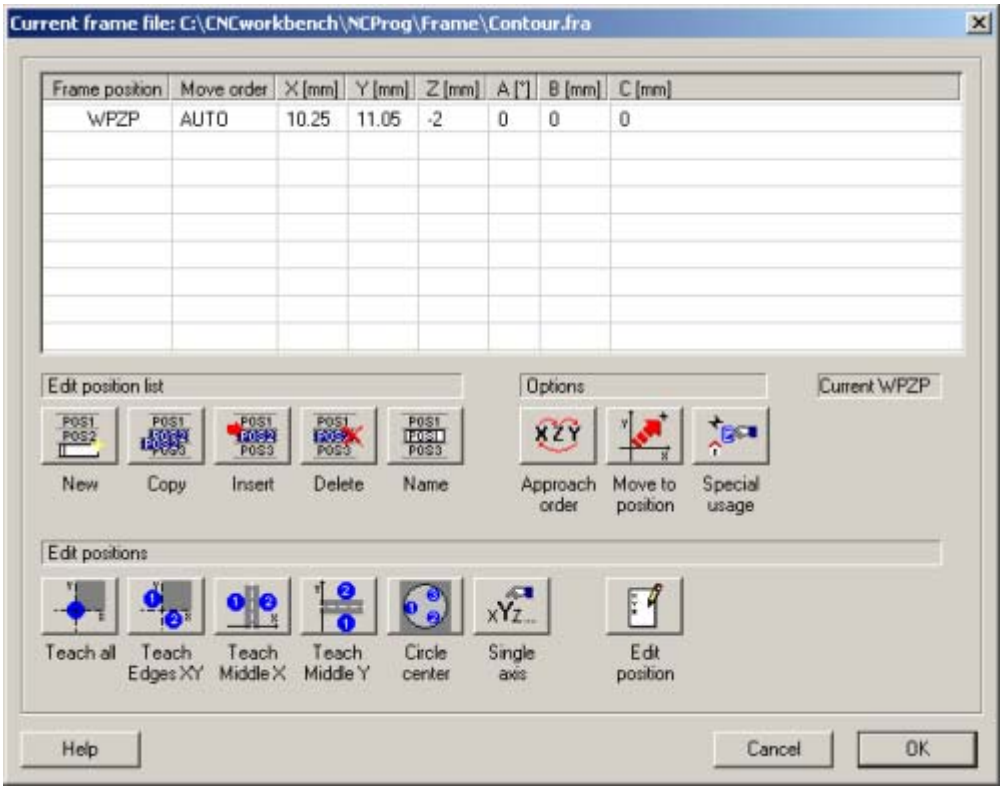
This program has an user frame file.
The command **TEACH** causes a stop.

The dialog window "current geometry file ..." is displayed.
The file contour.fra out of the directory /CNCworkbench/NCProg/Frame as current geometry file is displayed in the dialog Machine positions.

After mouse click on the geometry file contour.fra, put down in the task bar, (at automatic mode the window appears automatically) a window opens analog to the Menu **Control - Machine - Edit machine positions.**
The file contour.fra contains a frame with the name **WPZP**.
The inscribed values for the **X-, Y- and Z axis** are over taken in the user program as **work piece zero point**.

Please check if the inscribed position is meaningful and if you want to make changes please use the button Teach all.

Quit with **OK** and **continue** the program with the button .



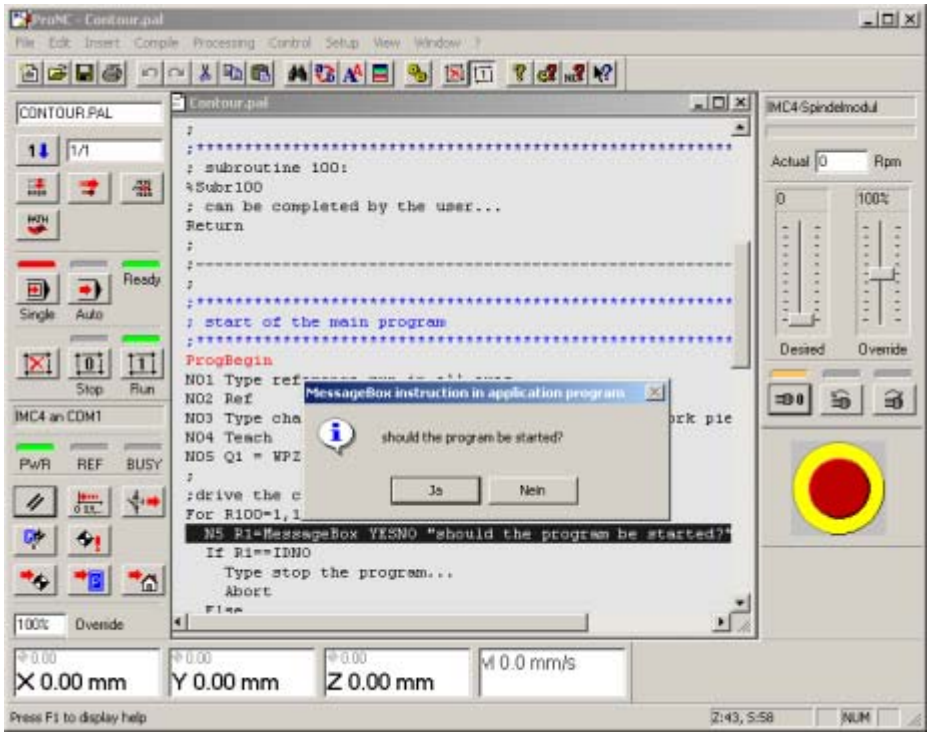
The functionalities of the dialog are also usable at editing of the user frame file.

! The indication of a position to **special usage** is not **reasonable**, because the execution **of these functions occurs only**, if they relates to the standard frame. That means, all setting options in Menu Setup **- Processing** relates exclusively to the **standard frame**.

More: 5.7.3.10 Menu Control - Machine - Setup machine positions
Programming Instruction: TEACH

6.2.6 Program process

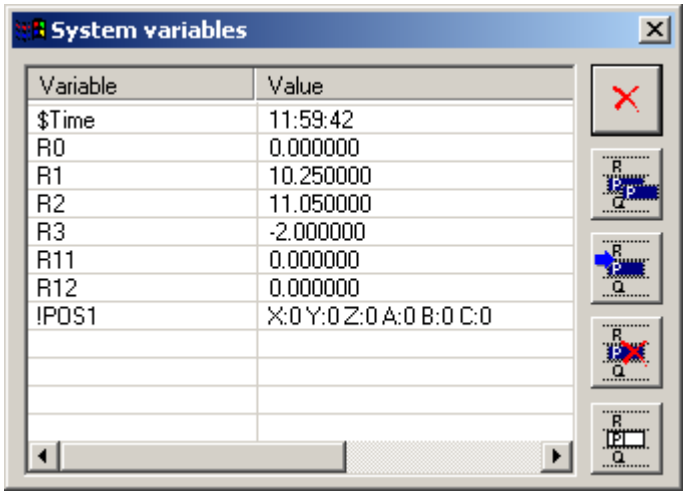
Before the execution of the first movement a dialog box appears on the screen.
Please decide at this stop, if you want to start or cancel the program. These query is offered to the operator after each program run.



More:
Programming Instruction: Request for an operator dialog

After transfer of the WPZP in the user program a fast movement to the coordinates of the frame in X and Y direction is executed.
The sink of the spindle in Z direction is executed after switch on the spindle.

☞ *The transfer of the frame **WPZP** into the variable R1, R2, R3 you can visualize by showing the system monitor (Menu **View - System monitor**).*

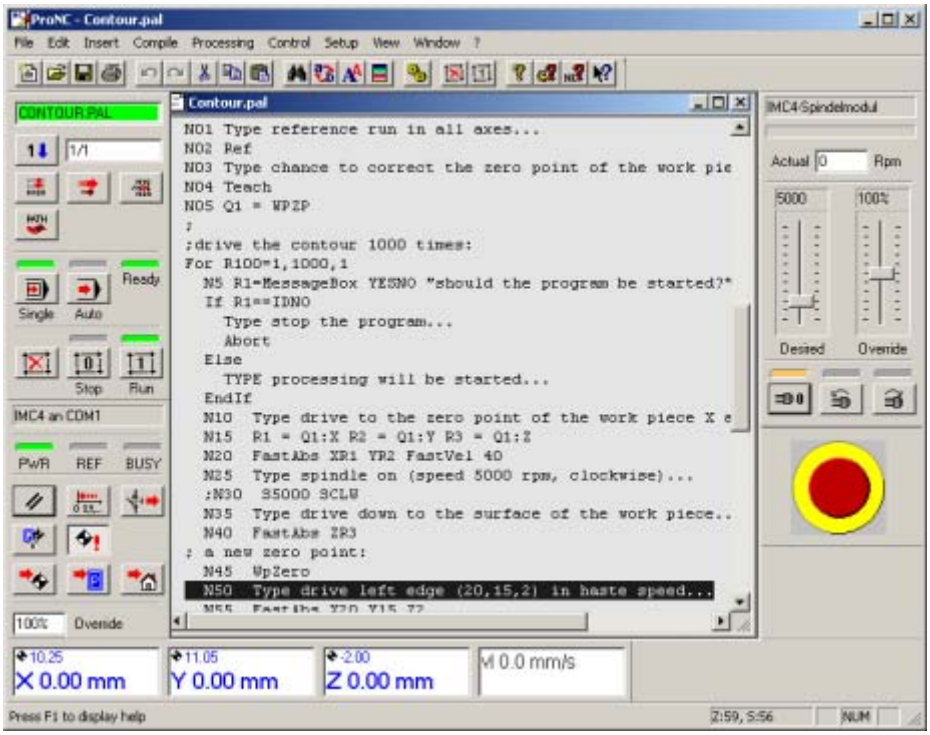


More:
1.6.3 Screen layout - Additional windows
Programming Instruction: R variable
Programming Instruction: P variable
Programming Instruction: Q variable

With the command WPZERO the approached point is set up as new zero point. All coordinate values with absolute measurement relates to this point.

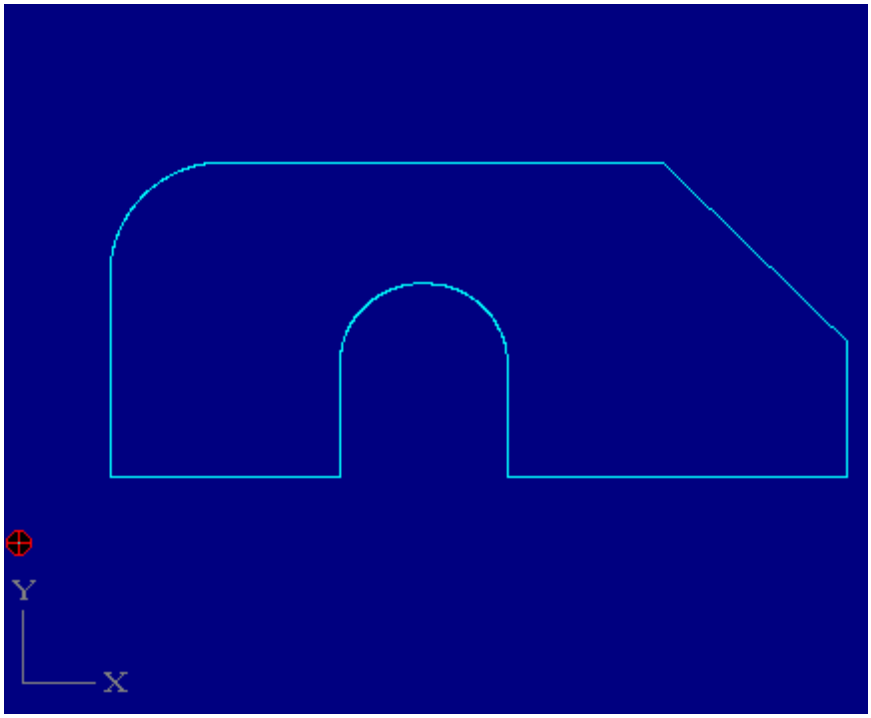
! *In the window "Position" the values of the axes X, Y, Z get the value zero and the display changes its color from black to blue.*
This is the confirmation, that the current zero point was set.

The coordinate values of the work piece zero point highlighted with the token  *are displayed In the combined position /velocity display.*



More:
1.6.2 Standard windows

The following linear path commands and circular interpolations produce this contour.



Glossary

DLL
Dynamic-Link-Libraries are software modules, containing one or more executable functions and they are stored as independent files. They are called from the main program during run time of the main program, linked to the application and executed.

EDP
Electronic data processing

goto-instructions
Added label, written from the compiler into the CNC target file during the compiler run

Module
Controller (e.g. a Stepper Motor Controller) or a device (e.g. a converter for a main spindle) or a Interface Card (e.g. the Servo Motor Control Card or Multi-I/O-Card) or a hardware (e.g. a CAN Field Bus Interface)

Rich Text Format
All formattings are stored. Formattings are converted to instructions, which can be read and interpreted from other programs inclusive of compatible Microsoft programs.

OEM
Orginal Equipment Manufacturer
Company which uses components of other manufacturers and sells under its own name

SC-Module

Security Circuit Module
Accessory device for Security Circuit of CNC Controls

Work piece zero point
Exposed point (e. g. the left lower corner of the work piece in the X-Y-plane) is taught or measured with the tool in the manual mode

Index

.

.cnc 30
.err 30
.fra 26
.ltb 30

A

Adjustment of step width and motion velocity . 78
applied isel-machine 8
Approach order 79, 80
automatic mode 68
automatic start 8

B

Bin directory 39
binary inputs 84
block operations 35
breakpoints 69
browse 34
button Cancel 47
button OK 47

C

CNC interpreter 38
CNC target file 30
Compensation 6
Compile file 66
configuration 44
Configuration and installation of the modules . 91
continuously setting 83
Control – 83
Control - Approach machine - positions 82
Control - Input/Output 84
Control - Machine - Setup machine positions. 78
Control - Machine – Activate work piece zero point 73
Control - Machine - Approach 74
Control - Machine - Approach to Home position 75
Control - Machine – Reference run of axis 73
Control - Machine – Reference run simulation 72
Control - Machine - Velocities 76
Control - Machine -Approach to park position . 74
Control – Machine-Manual movement 76
Control directory 39
Control panel Spindle 18
Control-Reference run 72
Control-Reset 71
Control-Spindle 82
copy 36
Copy 51
current axis position 18
cursor control 34
Cut out 51

D

Date and time 54
Deactivate work piece zero point 74
debug features 13
Definition of frames with a text editor 29
delete characters 35
delivery 6
diagnosis 44
dialog- oriented generation of commands 12
dialog software 11
directory structure 6
DLL concept 10
driver calls 68

E

Edit file 16
Edit of frame positions 80
Edit positions 80
Editing functions 34
Editing of source files 33
error file 31
error list 66
example program 99

F

File - Close 50
File - Exit 49
File - Print 50
File - store 50
file functions 37
file type 49
Font 37
frame name 79

G

gearing 92
geometry file 25
geometry information 25
Grammar of ProNC 23

H

Help directory 39

I

IMC4 Controller 42
IMS6 Controller 42
Input and output 44
Input files of CNC interpreter 39
Input/Output-DLLs 10

Insert 52
Insert - circle or pitch circle 57
Insert – Drilling cycle 60
Insert - Repeat loop 65
Insert branches 63
Insert command 54
Insert file 54
Insert Line in XYZ-space 56
Insert linear motion 55
Insert message 62
Insert multiple branch 64
Insert of program parts 16
Insert Polygon 59
Insert -Rectangle 59
Insert Signal input 60
Insert Signal output 61
Insert subprogram 54
installation 8
Installation of ProNC 8
ISY 7, 23

L

label file 31
lexical analysis 38
Liability 6

M

Machine status 20
Main menu 15
Main task of CNC interpreter 39
Menu View 97
Menu Window 98
Modification of step width 78
module 10
Module initialisation file 91
Module interface DLL 91
Module-DLL 91
modules and hardware 42
Motion control 43
Motion control DLLs 10
motion modes 77

N

ncctrl.ini 22
NcProg directory 39

O

Online path calculation 71
Open Datei 37
operating surface 14
Options - 86
Options-Units 87
output port 84
Override 83

P

pitch 92
possible applications 12
Print 37
Pro-DIN 10
program call 9
program icon 8, 9, 99, 111
Program skip 70
Program template 53
programming method 12
pronc.ini 22
Pro-PAL 10

R

Rapid traverse 70
referencing the tool clamp 96
Remote 10
Replace 36, 52
Report file 31
Restore 37, 51
Rich Text Format 12
Robot and Handling systems 12
RTF 12

S

Save 37
Search 36, 52
Security circuit- 20
Servo.ini 8
Servomotor control card 42
Set skip 70
set up printer 49
setup 44
Setup - Interpreter 88, 89
Setup-Controls 90
Side view 50
Single step 68
slide control 83
special usage 79
Special usage 80
Spindle control 44
Spindle on clockwise 83
Spindle-DLLs 10
stack 39
standard geometry file 29
Start 69
Start position for referencing 95
Start the compiler 38
state 44
status bar 16
StdFrame.fra 29
Stop 69
subdirectories 39
symbol bars 97
syntactic analysis 38
System requirement 9
system variable 19

T

Task of the compiler 37
Temp directory 39

Terminate 69
textual programming 23
tool change position 95

U

Undo 37, 51
UPMV4/12 42
user geometry file 29
UVE 8112 42

V

variable file 32
Velocity 18
version access 44