

**SIEMENS**

**SINUMERIK 802D solution line  
Machine Controller Handbook T/M**

Training Manual

Edition 2007.7

Training Material for none Metric Systems

# **SIEMENS**

## **SINUMERIK 802D sl**

**Operating, Programming  
and Service  
Turning and Milling**

Valid for

Control	Software
SINUMERIK 802D sl	1.4

## Module content for end users.

<u>Operating and Programming</u>		<u>Service</u>	
CNC Basic Principles Turning	C101		
CNC Basic Principles Milling	C102	Pushbutton test (MCP)	C8
Power On and Referencing	C69	LED Diagnosis Drive	C9
Control Structure / Navigation	C70	LED Diagnosis HMI	C10
Basic Program Structure	C98	MCPA Board	C11
Modality (preparatory functions)	C89	Save Data Backup	C17
Commonly used G functions - T	C91	Restore Data Backup	C18
Commonly used G functions - M	C92	External data backup and restore	C19
Basic Miscellaneous Codes	C97	Diagnose PLC program	C28
Additional G functions - T	C93	Diagnose PLC alarm	C31
Additional G functions - M	C94	Alarm structure	C37
Create part program - Milling	C77	Editing of NC Machine data	C41
Create part program - Turning	C78	Drive Diagnostic parameters	C53
Cycles - M	C99		
Cycles - T	C100		
Free Contour Programming T	C105		
Free Contour Programming M	C106		
File Management - CF card	C59		
Axis Control - Jog	C82		
MDA - Turning	C75		
MDA - Milling	C76		
Tools Turning - work planes	C71		
Tools Milling - work planes	C72		
Work offsets - Turning	C73		
Work offsets - Milling	C74		
Automatic	C81		
RCS 802 Data transfer tool	C1		
Mold & Die Programming	C104		
ISO Dialect programming - M	C95		
ISO Dialect programming - T A-B-C	C96		
Tools Turning—Milling functions	C83		

## 1 Brief description

**Module Objective:**

Upon completion of this module you can understand the basic functionality and requirements of a CNC controller

**Module description:**

CNC machines replicate the functions of a manual machine through the use of a sequential program, this program replicates the movements of a manual machine through the so called “CNC program”. Geometric relationships of the component to the machine are achieved with (Tool and component) numeric compensations. These compensations mathematically compensate the difference between component, Tool and machine.

**Module Content:**

Fundamentals of CNC Machines

Fundamentals of CNC  
Machines



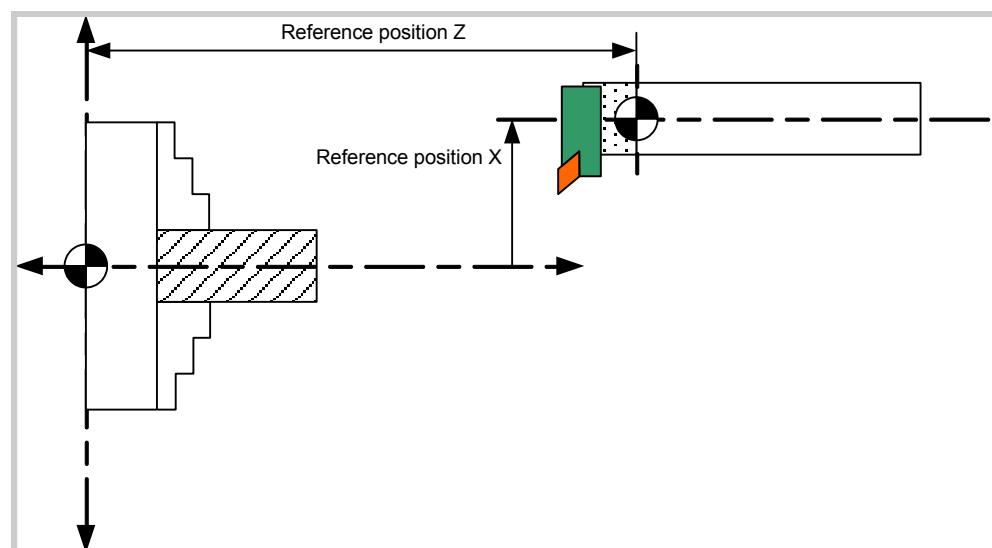
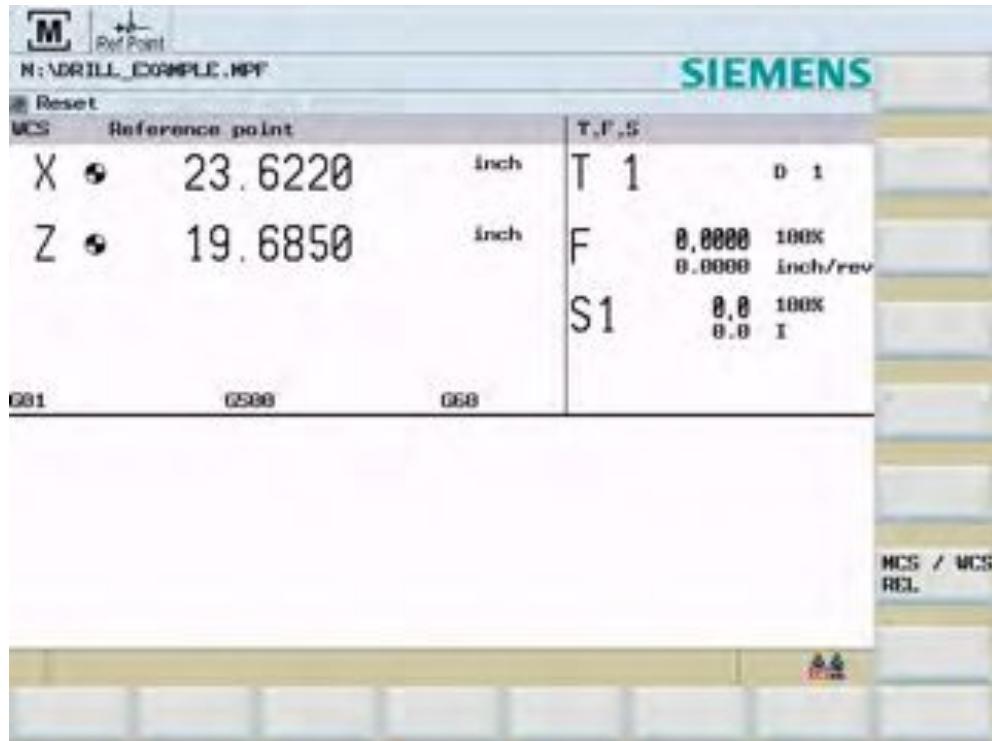
Section 2

## Section 2

### Fundamentals of CNC Machines

Notes

The difference between a Manual machine and a CNC machine is a logical difference based upon numerical values. This overview module and the following CNC modules should help explain the myth of CNC.  
ITS ONLY NUMBERS. No magic !!



**A referenced machine has a defined position.**

**If theoretically zero for all three axis is commanded, the reference point of the tool will drive to the zero point of the machine.**

**“CRASH”**

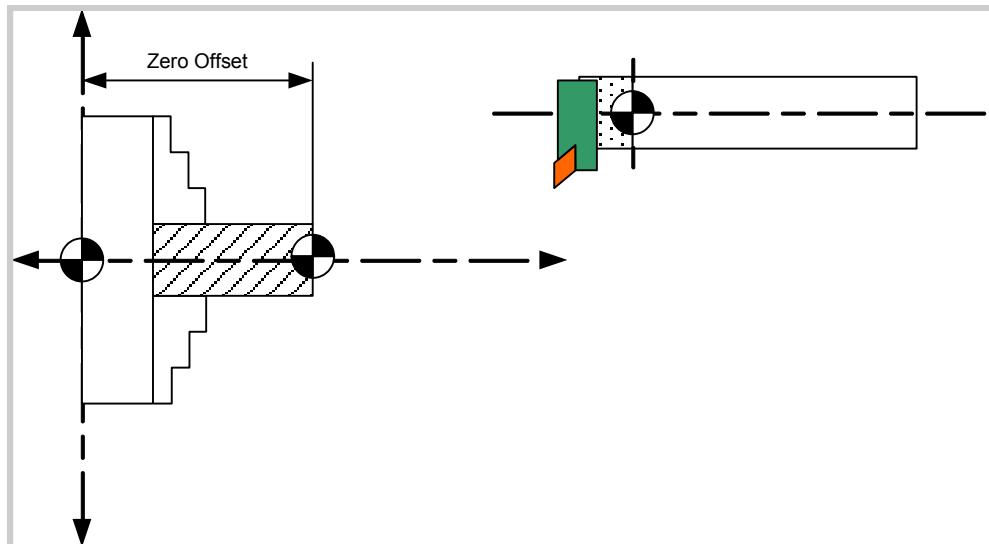
## Section 2

### Fundamentals of CNC Machines

Notes

The screenshot shows the 'Work offset' configuration screen. It displays two coordinate systems: 'MCS' and 'X1'. In 'MCS', X is 23.6220 inch, Z is 19.6850 inch, and C is 0.0000. In 'X1', X is 11.811 inch, Z is 19.685 inch, and C is 0.000. A table below lists various parameters like 'Base', 'G54', 'Program', etc., with values mostly at 0.000. A vertical toolbar on the right includes buttons for 'Further axes', 'Measure workpiece', 'Activate change', and others.

	X	inch	Z	inch	C	*	X	?	Z	?	C	?
Base	0.000		0.000		0.000		0.000		0.000		0.000	
G54	0.000		0.000		0.000		0.000		0.000		0.000	
G55	0.000		0.000		0.000		0.000		0.000		0.000	
G56	0.000		0.000		0.000		0.000		0.000		0.000	
G57	0.000		0.000		0.000		0.000		0.000		0.000	
G58	0.000		0.000		0.000		0.000		0.000		0.000	
G59	0.000		0.000		0.000		0.000		0.000		0.000	
Program	0.000		0.000		0.000		0.000		0.000		0.000	
Scale	1.000		1.000		1.000							
Mirror	0		0		0							
Total	0.000		0.000		0.000		0.000		0.000		0.000	



When a work piece offset is activated.

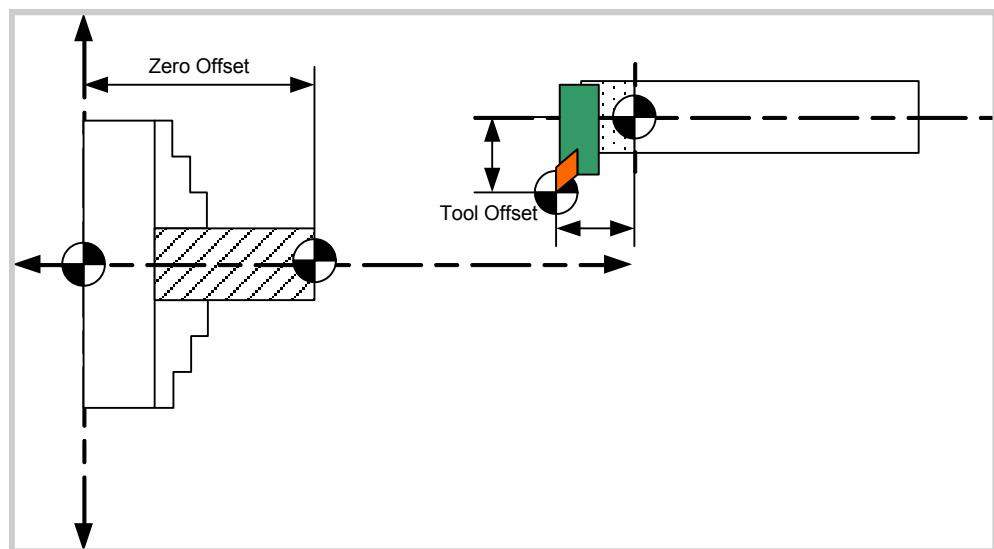
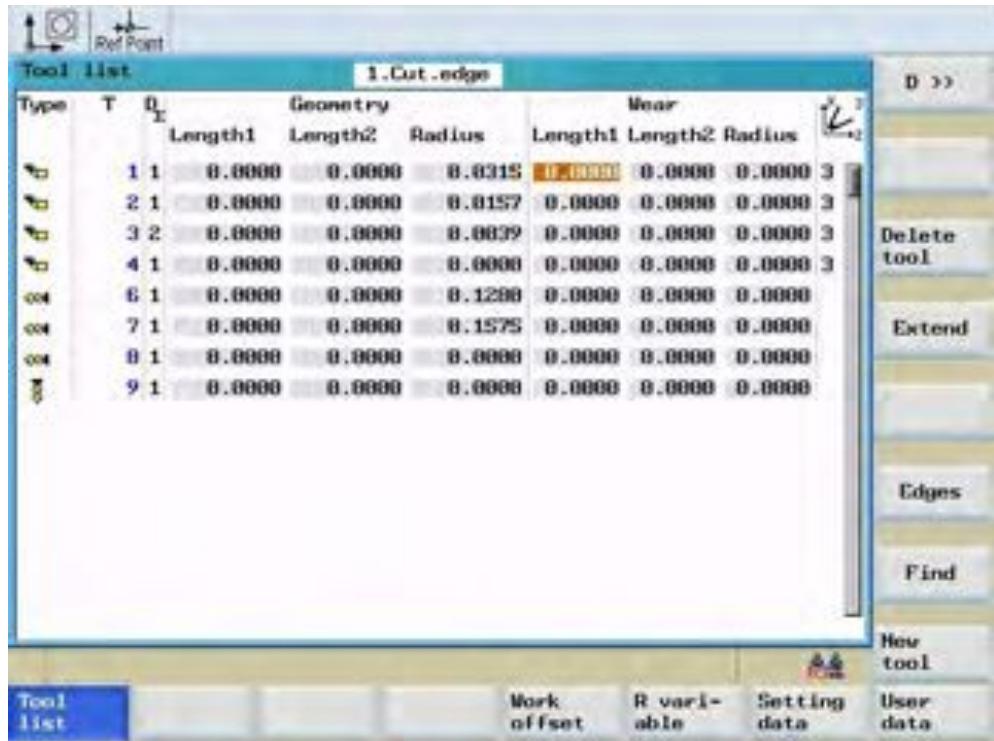
If theoretically zero for all three axis is commanded, the reference point of the tool will drive to the zero point of the workpiece.

"CRASH"

## Section 2

### Fundamentals of CNC Machines

Notes



When a tool offset is activated.

If theoretically zero for all three axis is commanded, the offset point of the tool will drive to the zero point of the workpiece.

"No Crash"

## 1 Brief description

**Module Objective:**

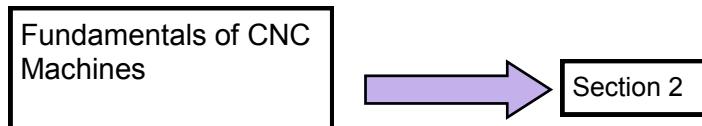
Upon completion of this module you can understand the basic functionality and requirements of a CNC controller

**Module description:**

CNC machines replicate the functions of a manual machine through the use of a sequential program, this program replicates the movements of a manual machine through the so called “CNC program”. Geometric relationships of the component to the machine are achieved with (Tool and component) numeric compensations. These compensations mathematically compensate the difference between component, Tool and machine.

**Module Content:**

Fundamentals of CNC Machines

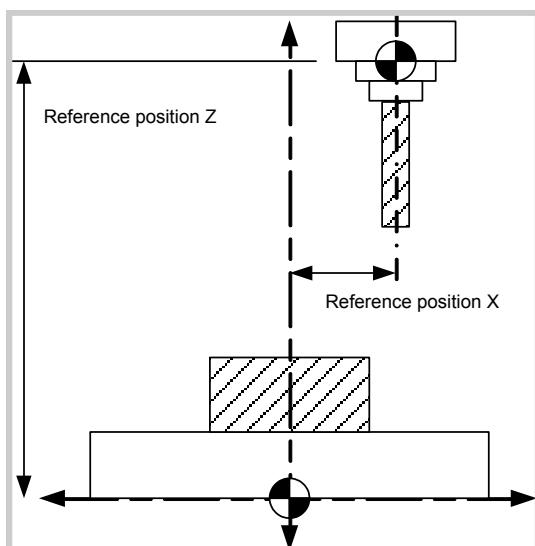
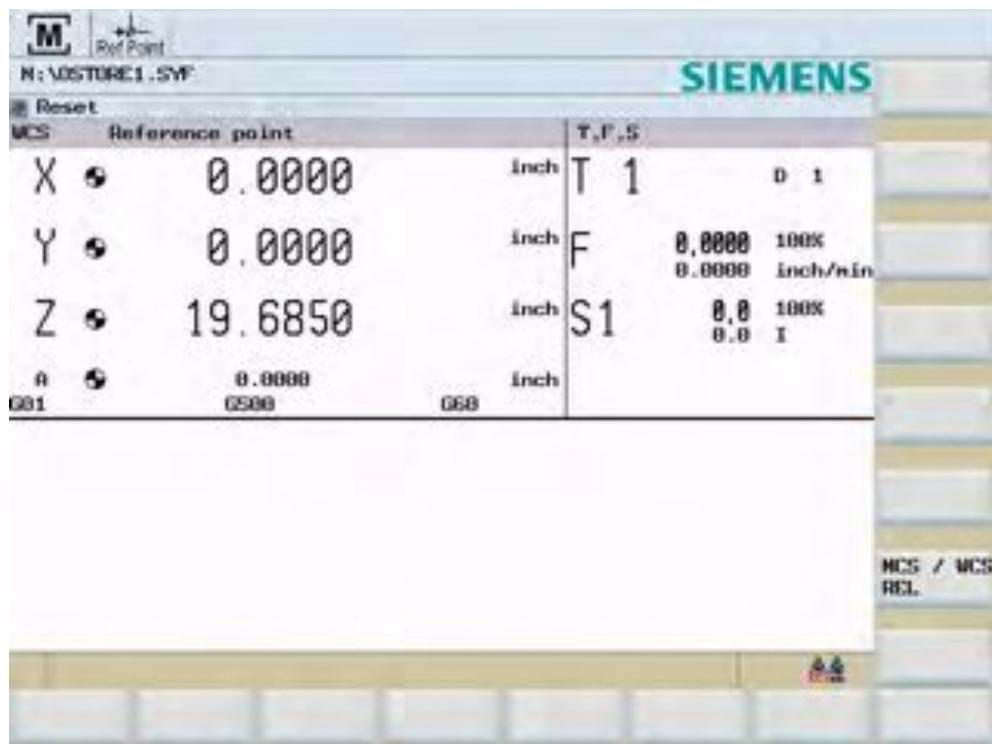


## Section 2

### Fundamentals of CNC Machines

Notes

The difference between a Manual machine and a CNC machine is a logical difference based upon numerical values. This overview module and the following CNC modules should help explain the myth of CNC.  
ITS ONLY NUMBERS. No magic !!



**A referenced machine has a defined position.**

**If theoretically zero for all three axis is commanded, the reference point of the tool will drive to the zero point of the machine.**

**"CRASH"**

## Section 2

### Fundamentals of CNC Machines

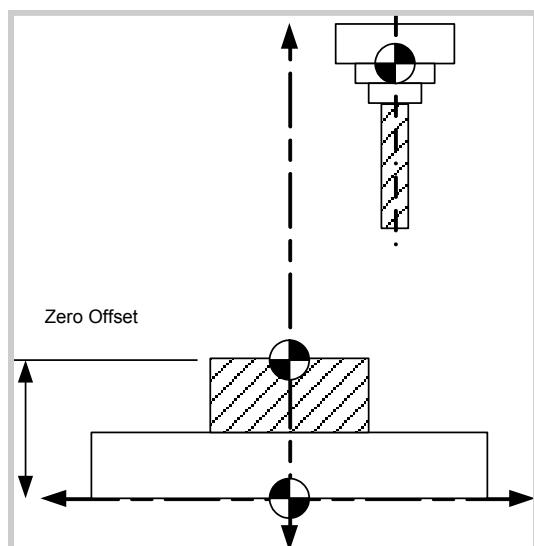
Notes



The screenshot shows the SINUMERIK 802D SI software interface. On the left, there's a toolbar with icons for 'Ref Point' and 'Work offset'. The main area displays 'Work offset' parameters in two coordinate systems (MCS and WCS) and a detailed table of offsets for various axes (X, Y, Z) across different base points (G54-G59, Program, Scale, Mirror) and a total row.

	X inch	Y inch	Z inch	X inch	Y inch	Z inch
Base	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G54	0.0175	-0.0254	-21.7535	0.0000	0.0000	0.0000
G55	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G56	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G57	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G58	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G59	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Program	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Scale	1.0000	1.0000	1.0000			
Mirror	0	0	0			
Total	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Below the table, there are tabs for 'Tool list', 'Work offset' (which is selected), 'R variable', 'Setting data', and 'User data'.



When a work piece offset is activated.

If theoretically zero for all three axis is commanded, the reference point of the tool will drive to the zero point of the workpiece.

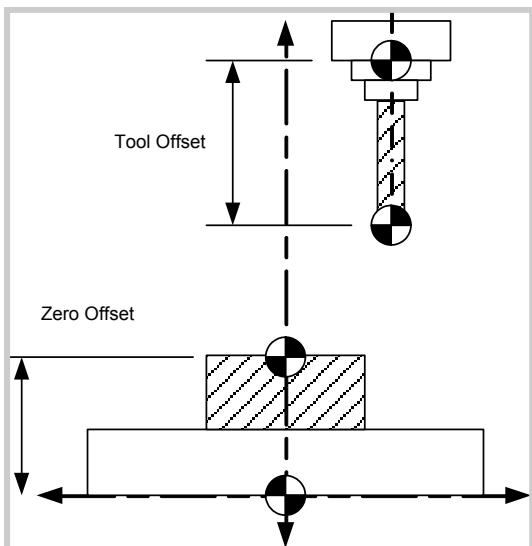
"CRASH"

## Section 2

### Fundamentals of CNC Machines

Notes

The screenshot shows a software interface for managing tools in a CNC machine. On the left, there is a 'Tool list' window titled '1.Cut.edges'. It contains a table with columns: Type, T, D\_z, Geometry, and Wear. The 'Geometry' column has sub-columns Length1 and Radius. The 'Wear' column also has sub-columns Length1 and Radius. There are six rows of data, each corresponding to a different tool type (1-6) and tool number (1). The first row (T=1, D\_z=1) has its 'Length1' value highlighted in orange. On the right side of the interface, a context menu is open with the following options: 'Delete tool', 'Extend', 'Edges', 'Find', 'New tool', and 'User data'. At the bottom of the interface, there are tabs for 'Tool list', 'Work offset', 'R variable', 'Setting data', and 'User data'.



When a tool offset is activated.

If theoretically zero for all three axis is commanded, the offset point of the tool will drive to the zero point of the workpiece.

"No Crash"

## 1 Description

**Module objective:**

Upon completion of this module you can switch on the machine and reference the axes.

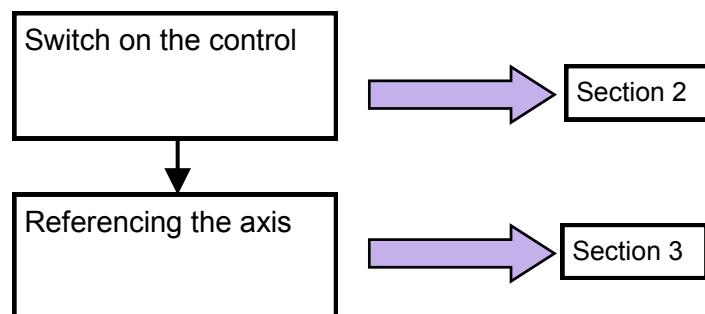
**Module description:**

Before you can work with the machine, the control has to be switched on. Axis with incremental measuring systems have to be referenced, so that the control knows where the axis are in the machine coordinate system.

It is not necessary to reference axis which have an absolute measuring system.

**Module content:**

Switch on the control  
Referencing the axis



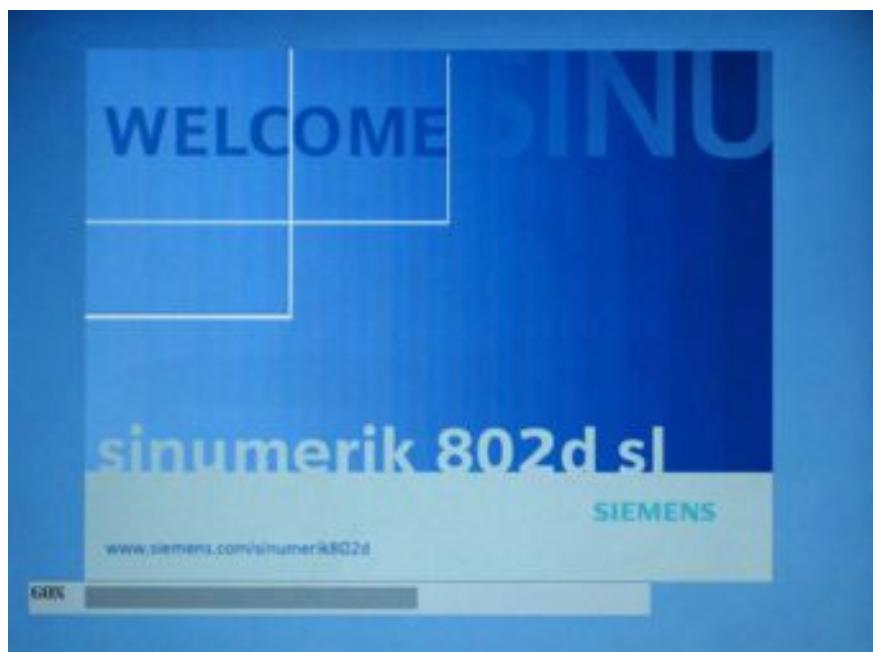
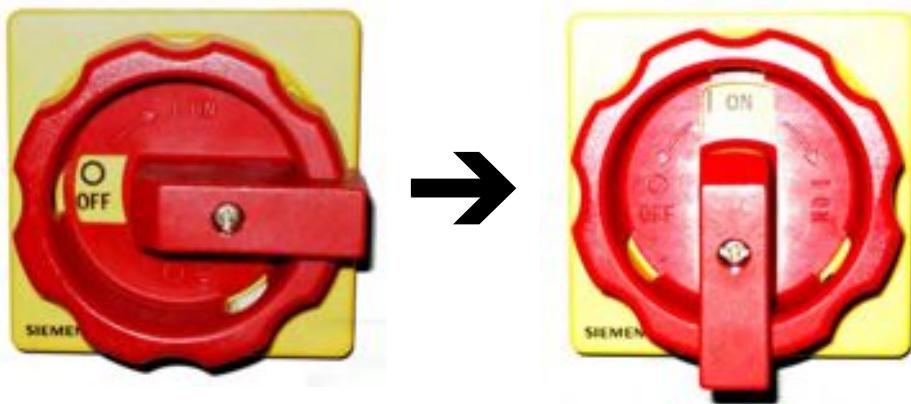
## Section 2

### Switch on the control

Notes

The controller starts up normally by switching the machine on.

**NOTE: Please take care to follow the switching on guidelines of the machine manufacturer**



Once that the controller is running, all E-Stop switches should normally be released. Please follow the guidelines of the machine manufacturer.



## Section 3

### Referencing the axis

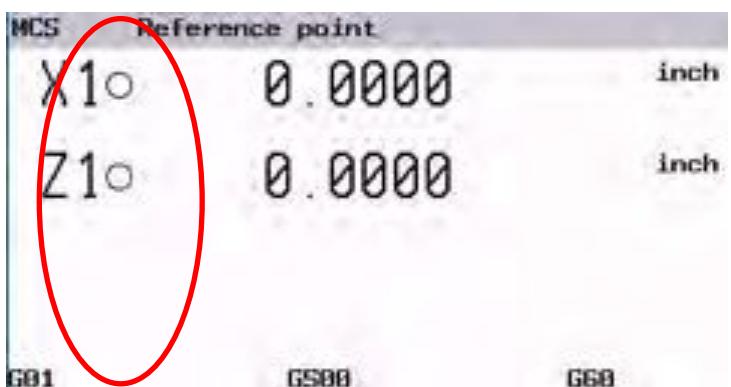
Notes

Machines with incremental measuring systems must first be referenced, in order to synchronise the measuring system with the machine coordinate system.

**The machine should be in a ready condition before trying to travel to the reference position !!**

The controller is automatically in Jog—reference mode after successfully powering up (JOG REF-mode).

Axis which require reference point approach are those shown below with an empty circle behind the axis name.



Dependant upon the commissioning of the machine, it is possible to use automatic referencing which is started simply by using " Cycle Start".



Or by using the following sequence:

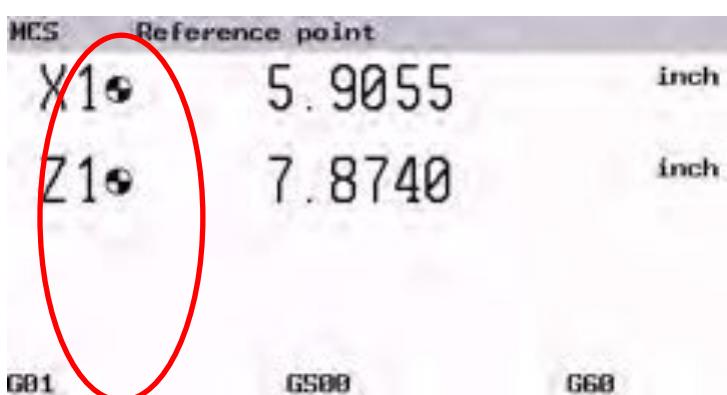
+Z

+X

Care to be taken  
with override  
switch.

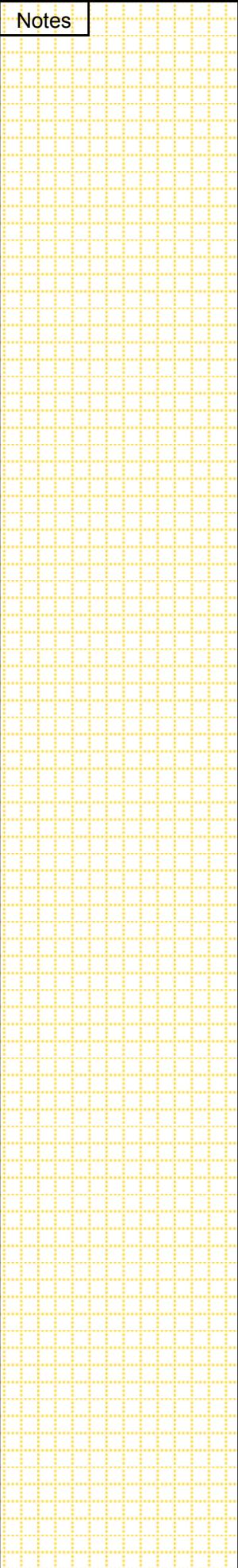


When the axis are successfully referenced, the referenced symbol will appear as shown below.



---

Notes



## 1 Description

**Module objective:**

Upon completion of this module you can switch on the machine and reference the axis.

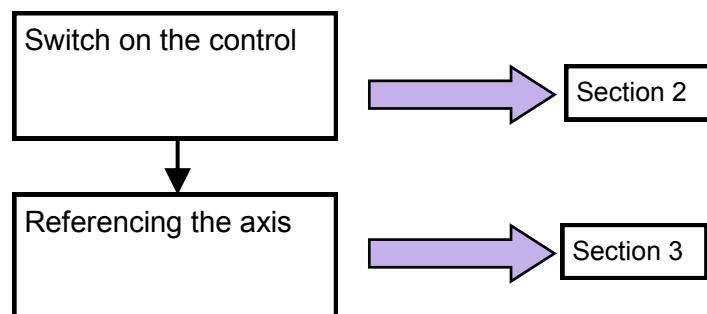
**Module description:**

Before you can work with the machine, the control has to be switched on. Axes with incremental measuring systems have to be referenced, so that the control knows where the axes are in the machine coordinate system.

It is not necessary to reference axes which have an absolute measuring system.

**Module content:**

Switch on the control  
Referencing the axis



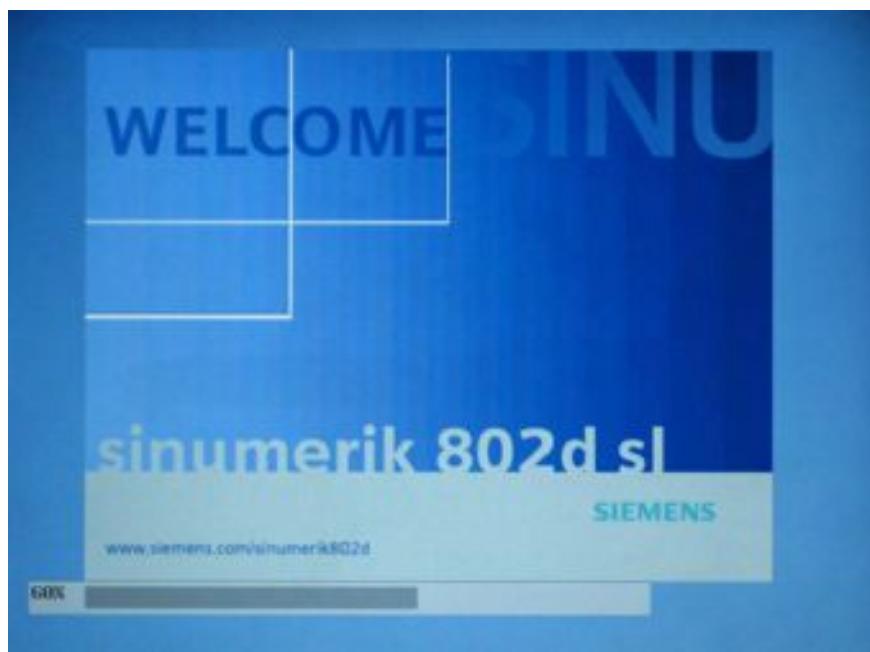
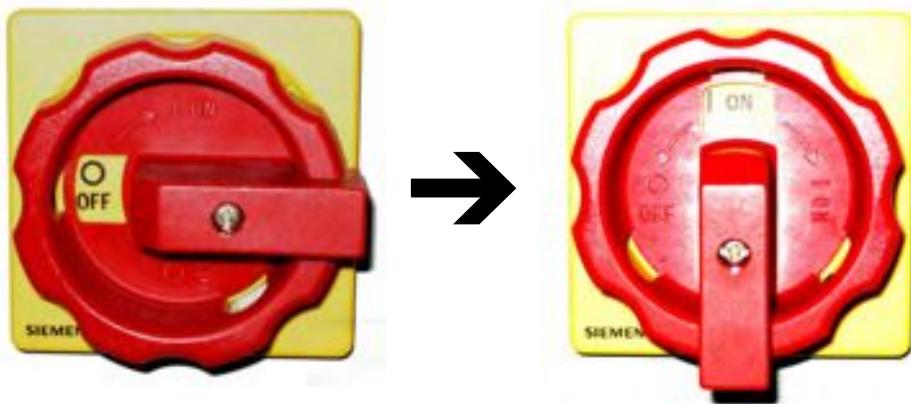
## Section 2

### Switch on the control

Notes

The controller starts up normally by switching the machine on.

**NOTE: Please take care to follow the switching on guidelines of the machine manufacturer**



Once that the controller is running, all E-Stop switches should normally be released. Please follow the guidelines of the machine manufacturer.



## Section 3

### Referencing the axis

Notes

Machines with incremental measuring systems must first be referenced, in order to synchronise the measuring system with the machine coordinate system.

**The machine should be in a ready condition before trying to travel to the reference position !!**

The controller is automatically in Jog—reference mode after successfully powering up (JOG REF-mode).

Axis which require reference point approach are those shown below with an empty circle behind the axis name.

X1○	0.0000	inch
Y1○	0.0000	inch
Z1○	0.0000	inch
A1 ○	0.0000	inch
G01	G500	G60

Dependant upon the commissioning of the machine, it is possible to use automatic referencing which is started simply by using “ Cycle Start“.



Or by using the following sequence:

+Z

+X

+Y

Care to be taken  
with override  
switch.

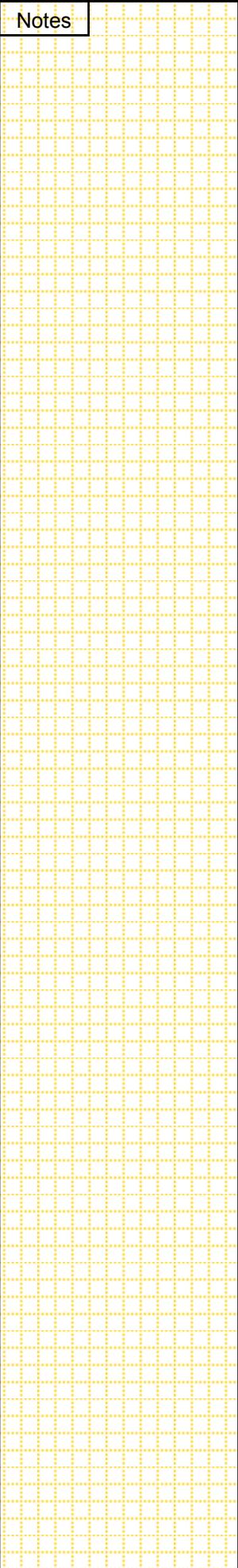


When the axis are successfully referenced, the referenced symbol will appear as shown below.

X1●	0.0000	inch
Y1●	0.0000	inch
Z1●	19.6850	inch
A1 ●	0.0000	inch
G01	G500	G60

---

Notes



# 1 Brief Description

**Module objective:**

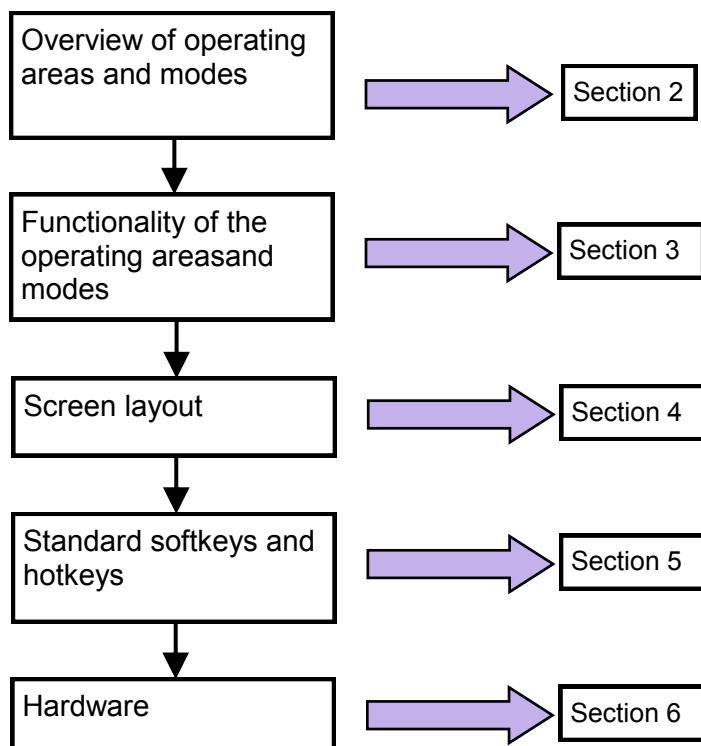
Upon completion of this module you can navigate the control.

**Module description:**

The 802D SL controller works on the principle of menus with softkeys, and associated to these are pictures. This module describes navigating these menus.

**Module content:**

Overview of operating areas and modes  
Functionality of the operating areas and modes  
Screen layout  
Standard softkeys and hotkeys  
Hardware



## Section 2

### Overview of operating areas and modes

Notes

The different operating areas of the control are reachable using the operator panel keyboard.

The following operating areas are available on the control:

Operating area	Softkey	Function
Operating area Machine		Setting functions, Program control, Measuring tools, Face milling, Face turning
Operating area Program		Program editing, Simulation, Cycle support, Block numbering
Operating area Offset / Parameter		Zero offsets, Tool parameters, R-Parameter, Setting data, User data
Operating area Program-Manager		Managing the NC program memory and the CF-Card, Program selection, Edit functions
Operating area System / Alarm		List of active alarms
Operating area customer		Machine builder specific

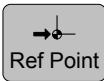
## Section 2

### Overview of operating areas and modes

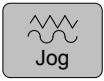
Notes

The different operating modes of the control are reachable using the screen softkeys..

The following operating modes are available on the control:



Use the **REF** key on the machine control panel to activate “**REFERENCE POINT APPROACH**”.



Use the **JOG** key on the machine control panel to select the **JOG** mode.



Use the **MDA** key on the machine control panel to select the **MDA** mode.



Use the **AUTO** key on the machine control panel to select the **AUTOMATIC** mode.



Use the **SINGLE BLOCK** key on the machine control panel to select **SINGLE BLOCK** mode.

## Section 3

### Functionality of the operating areas and modes

Overview of the softkeys in JOG



Notes

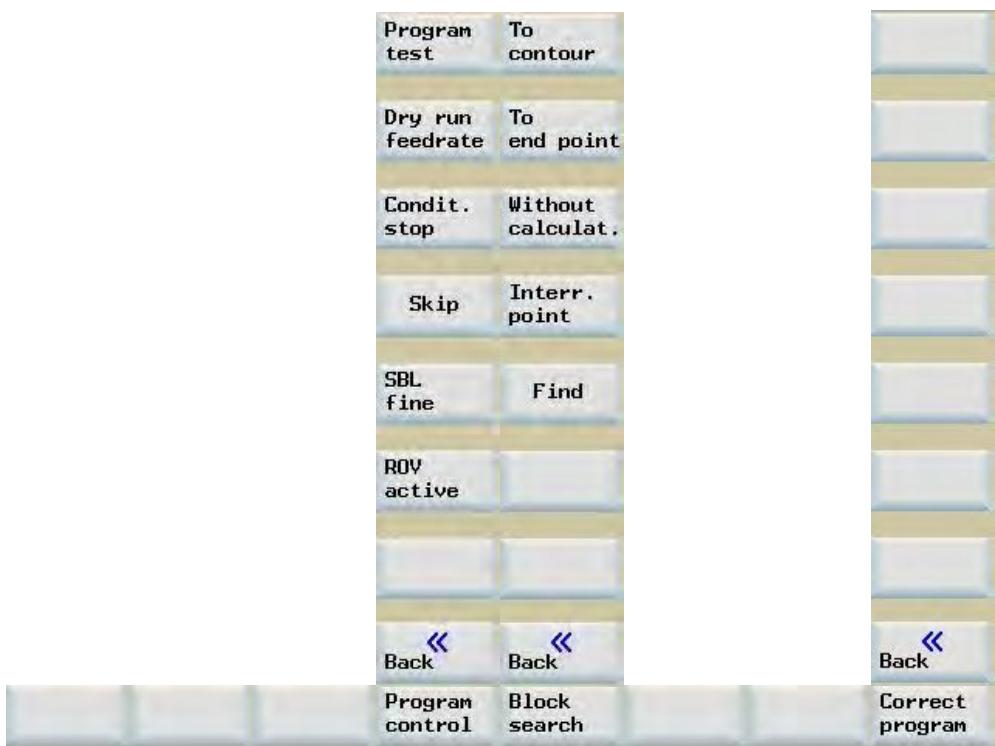
Overview of the softkeys in MDA



## Section 3

### Functionality of the operating areas and modes

Overview of the Softkeys in AUTOMATIC



Notes

Overview of the Softkeys in PROGRAM

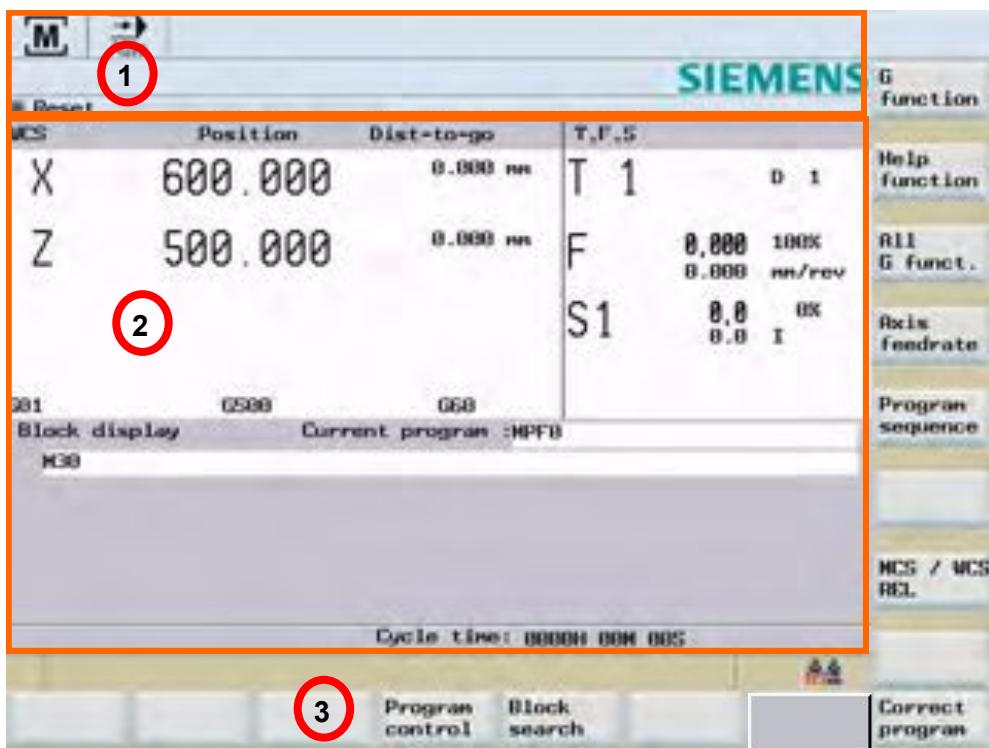


## Section 4

### Screen layout

#### Screen layout

Notes



**1** Status Area

**2** Application Area

**3** Tip and softkey Area

Notes

#### Status Area



#### 1 Active operating area, active mode

Position:

JOG: 1INC, 10INC, 100INC, 1000INC, VAR INC  
MDA  
AUTOMATIC  
OFFSET  
PROGRAM  
PROGRAM MANAGER  
SYSTEM

#### 2 Alarm and message line

1. Alarm number with alarm text or
2. Message text

#### 3 Program status

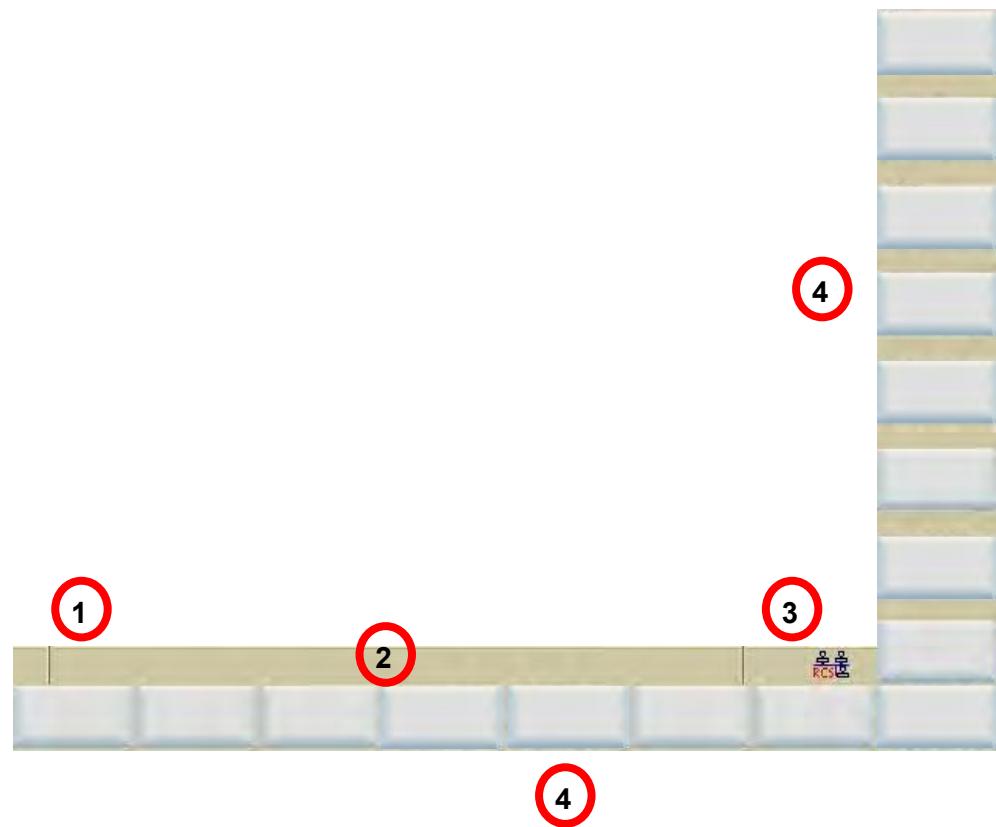
RESET	Program cancelled/default status
RUN	Program running
STOP	Program stopped

#### 4 Program control in Automatic mode

#### 5 NC messages

#### 6 Selected part program (main program)

#### Tip and softkey area



##### 1 Recall key



Pressing the recall key lets you return to the next higher menu level

##### 2 Tip line

Displays tips for the operator

##### 3 MMC status information



Data transfer running



Connection to PLC programming tool active

##### 4 Vertical and horizontal softkey bar

Notes

Notes

#### Standard softkey



Selecting this softkey to delete the content of the calculator.



Selecting this softkey to quit the screen form.



Selecting this softkey to cancel the input; the window is closed.



Selecting this softkey will complete your input and start the calculation.



Selecting this softkey will complete your input and accept the values you have entered.



The calculator function can be activated from any operating area using "SHIFT" and "=".

If the input field is already occupied by a value, the function will accept this value into the input line of the calculator.

When you press the **Input** key, the result is calculated and displayed in the calculator.

Selecting the **Accept** softkey enters the result in the input field at the current cursor position of the part program editor and closes the calculator automatically.



#### Hotkeys

This operator control can be used to select, copy, cut and deleted text using special key commands. These functions are available for the part program editor and for input fields.

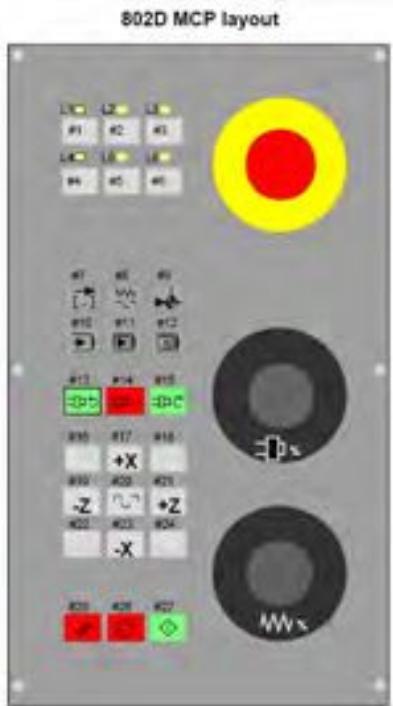
CTRL	+	C	Copy
CTRL	+	B	Select
CTRL	+	X	Cut
CTRL	+	V	Paste
ALT	+	L	Switch between upper/lower case
ALT	+	H	Help system
Or Info key			

**Key definition of CNC keyboard (Vertical Format)**

- "Recall" key
- ETC key
- "Acknowledge alarm" key
- Without function
- Enter key
- Shift key
- Control key
- Alt key
- SPACE
- Backspace

	Clear key
	Insert key
	Tabulator
	ENTER/Input key
	"Position" operating area key
	"Program" operating area key
	"Parameter" operating area
	"Program Manager" operating area
	"Alarm/System" operating area
	Not assigned
	PageUp/PageDown keys
	Cursor keys
	Selection key/toggle key
	Alphanumeric keys Double assignment on the Shift level
	Numeric keys Double assignment on the Shift level

Notes

**Key definition of machine control panel**

RESET



NC STOP



NC START



EMERGENCY STOP



Spindle override



User-defined key with LED



User-defined key without LED



INCREMENT

Incremental dimension



JOG



REFERENCE POINT



AUTOMATIC



SINGLE BLOCK



MANUAL DATA

Manual input



SPINDEL START LEFT

Spindle CCW rotation



SPINDLE STOP



SPINDEL START RIGHT

Spindle CW rotation



RAPID TRAVERSE OVERLAY

Rapid traverse override



X axis



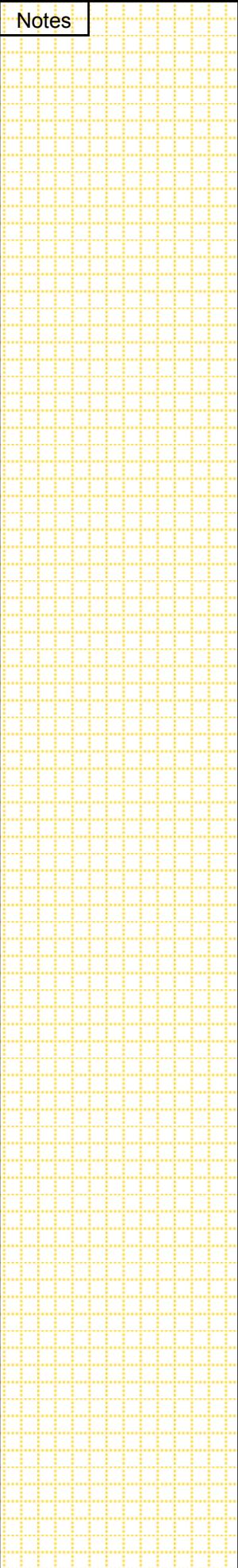
Z axis



Feed Rate Override

---

Notes



# 1 Brief Description

**Module objective:**

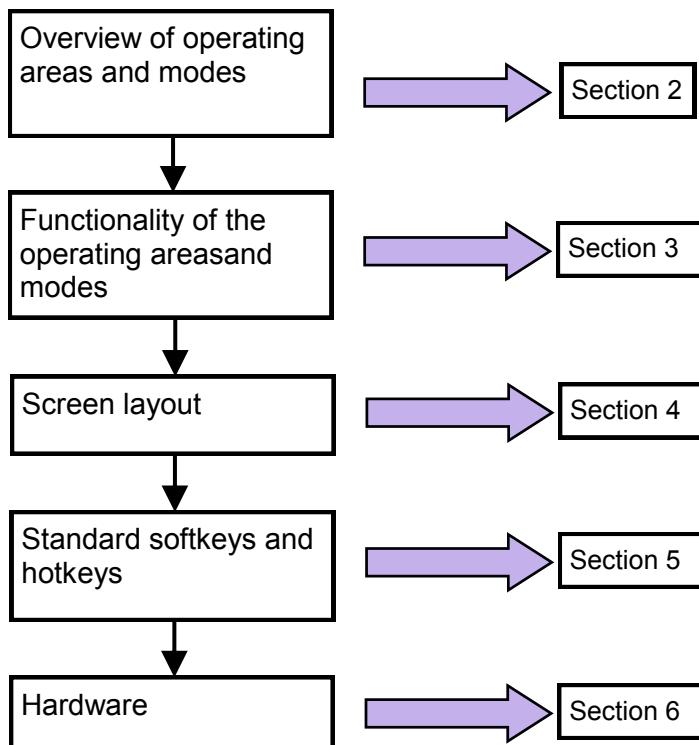
Upon completion of this module you can navigate the control.

**Module description:**

The 802D SL controller works on the principle of menus with softkeys, and associated to these are pictures. This module describes navigating these menus.

**Module content:**

Overview of operating areas and modes  
Functionality of the operating areas and modes  
Screen layout  
Standard softkeys and hotkeys  
Hardware



Notes

The different operating areas of the control are reachable using the operator panel keyboard.

The following operating areas are available on the control:

Operating area	Softkey	Function
Operating area Machine	 M POSITION	Setting functions, Program control, Measuring tools, Face milling, Face turning
Operating area Program	 PROGRAM	Program editing, Simulation, Cycle support, Block numbering
Operating area Offset / Parameter	 OFFSET PARAM	Zero offsets, Tool parameters, R-Parameter, Setting data, User data
Operating area Program-Manager	 PROGRAM MANAGER	Managing the NC program memory and the CF-Card, Program selection, Edit functions
Operating area System / Alarm	 SYSTEM ALARM	List of active alarms
Operating area customer	 CUSTOM	Machine builder specific

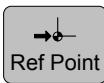
## Section 2

### Overview of operating areas and modes

Notes

The different operating modes of the control are reachable using the screen softkeys..

The following operating modes are available on the control:



Use the **REF** key on the machine control panel to activate “**REFERENCE POINT APPROACH**”.



Use the **JOG** key on the machine control panel to select the **JOG** mode.



Use the **MDA** key on the machine control panel to select the **MDA** mode.



Use the **AUTO** key on the machine control panel to select the **AUTOMATIC** mode.



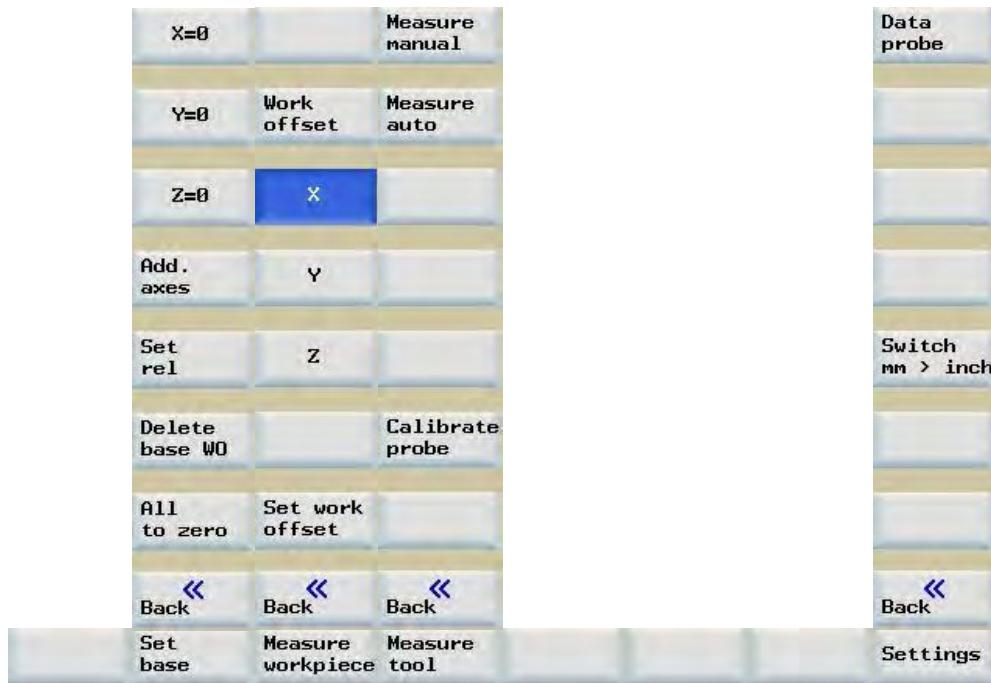
Use the **SINGLE BLOCK** key on the machine control panel to select **SINGLE BLOCK** mode.

## Section 3

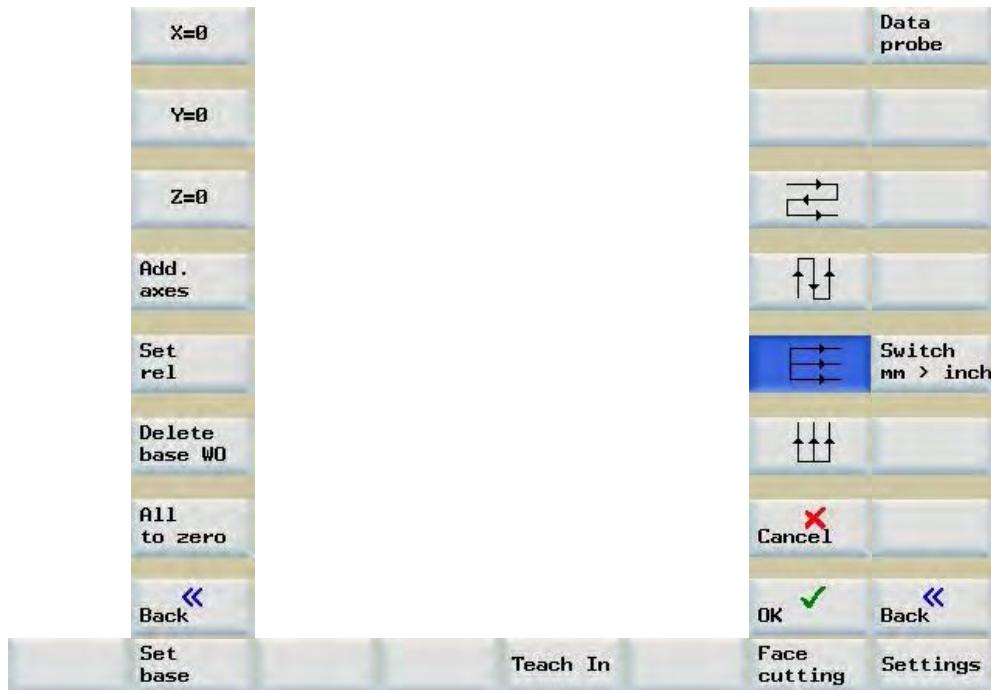
### Functionality of the operating areas and modes

Notes

Overview of the softkeys in JOG



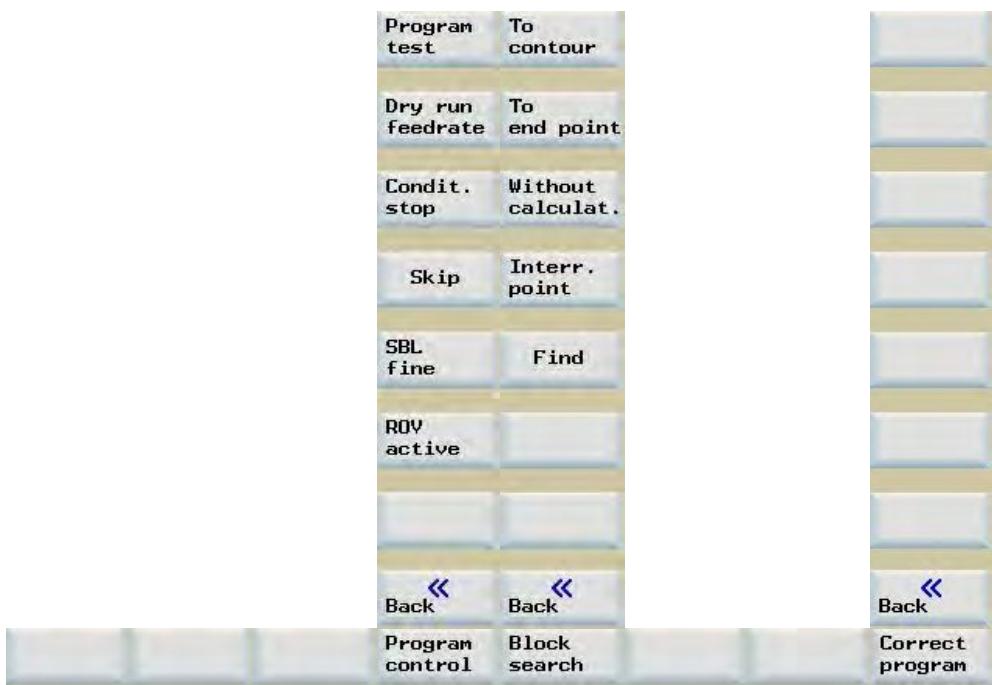
Overview of the softkeys in MDA



## Section 3

### Functionality of the operating areas and modes

Overview of the Softkeys in AUTOMATIC



Notes

Overview of the Softkeys in PROGRAM

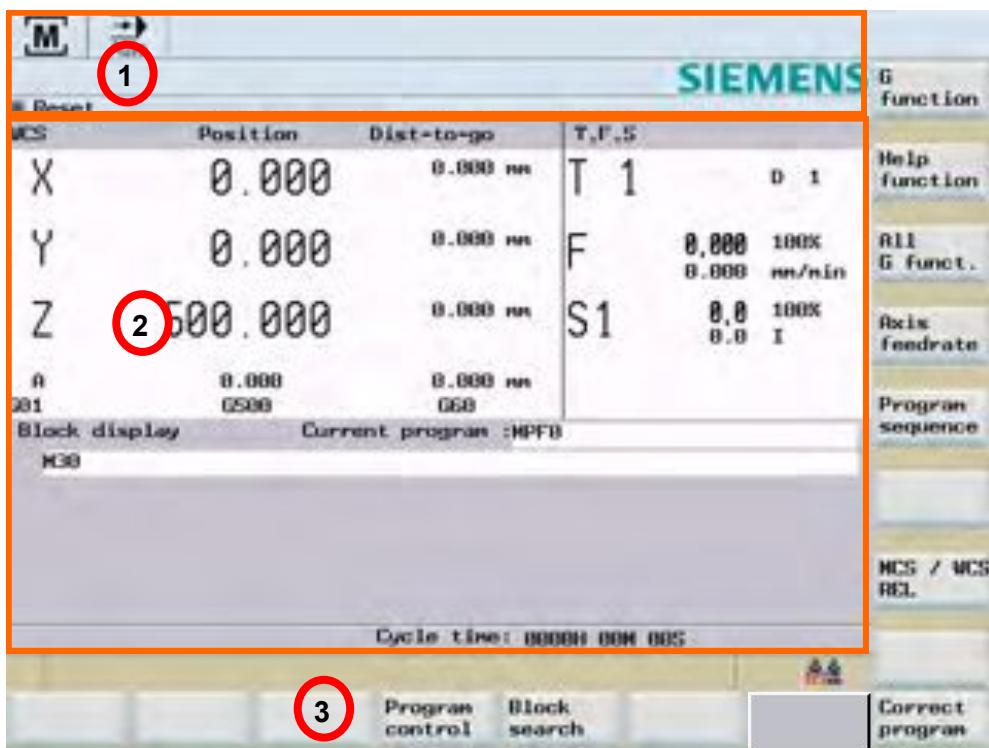


## Section 4

### Screen layout

#### Screen layout

Notes



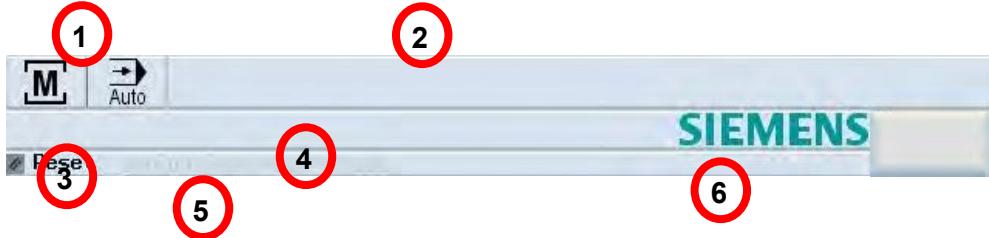
1 Status Area

2 Application Area

3 Tip and softkey Area

Notes

#### Status Area



#### 1 Active operating area, active mode

Position:

JOG: 1INC, 10INC, 100INC, 1000INC, VAR INC  
MDA  
AUTOMATIC  
OFFSET  
PROGRAM  
PROGRAM MANAGER  
SYSTEM

#### 2 Alarm and message line

1. Alarm number with alarm text or
2. Message text

#### 3 Program status

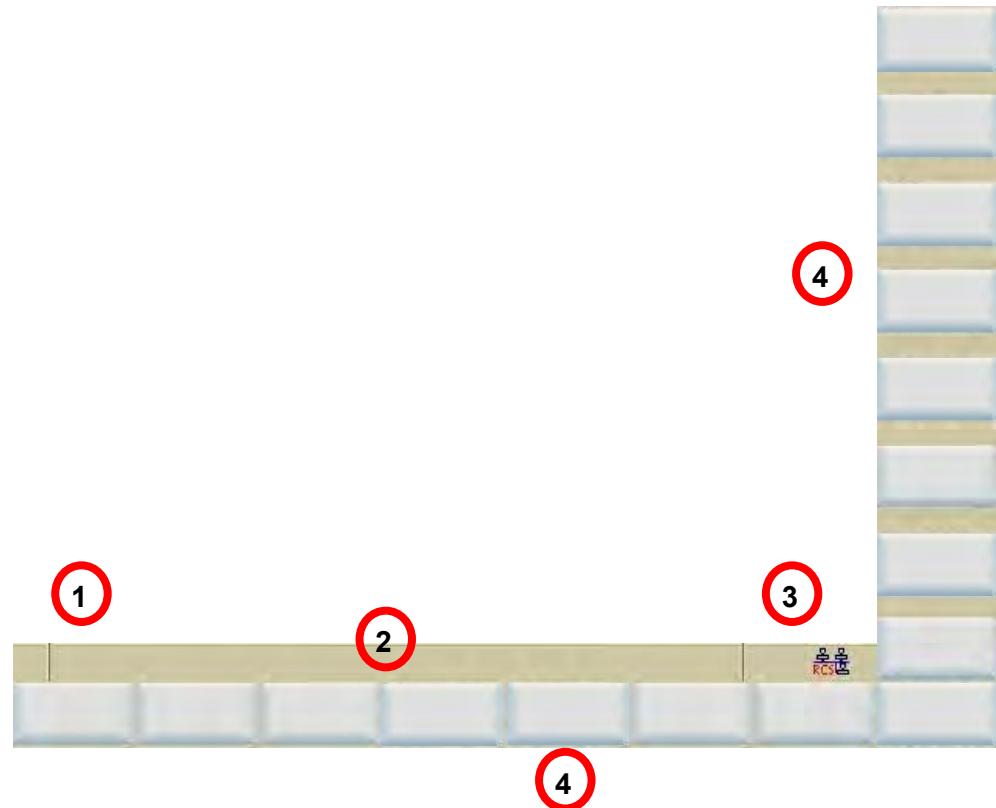
RESET	Program cancelled/default status
RUN	Program running
STOP	Program stopped

#### 4 Program control in Automatic mode

#### 5 NC messages

#### 6 Selected part program (main program)

#### Tip and softkey area



##### 1 Recall key



Pressing the recall key lets you return to the next higher menu level

##### 2 Tip line

Displays tips for the operator

##### 3 MMC status information



Data transfer running



Connection to PLC programming tool active

##### 4 Vertical and horizontal softkey bar

Notes

#### Standard softkey



Selecting this softkey to delete the content of the calculator.



Selecting this softkey to quit the screen form.



Selecting this softkey to cancel the input; the window is closed.



Selecting this softkey will complete your input and start the calculation.



Selecting this softkey will complete your input and accept the values you have entered.



The calculator function can be activated from any operating area using "SHIFT" and "=".

If the input field is already occupied by a value, the function will accept this value into the input line of the calculator.

When you press the **Input** key, the result is calculated and displayed in the calculator.

Selecting the **Accept** softkey enters the result in the input field at the current cursor position of the part program editor and closes the calculator automatically.



#### Hotkeys

This operator control can be used to select, copy, cut and deleted text using special key commands. These functions are available for the part program editor and for input fields.

CTRL	+	C	Copy
CTRL	+	B	Select
CTRL	+	X	Cut
CTRL	+	V	Paste
ALT	+	L	Switch between upper/lower case
ALT	+	H	Help system
Or Info key			

### Key definition of CNC keyboard (Vertical Format)



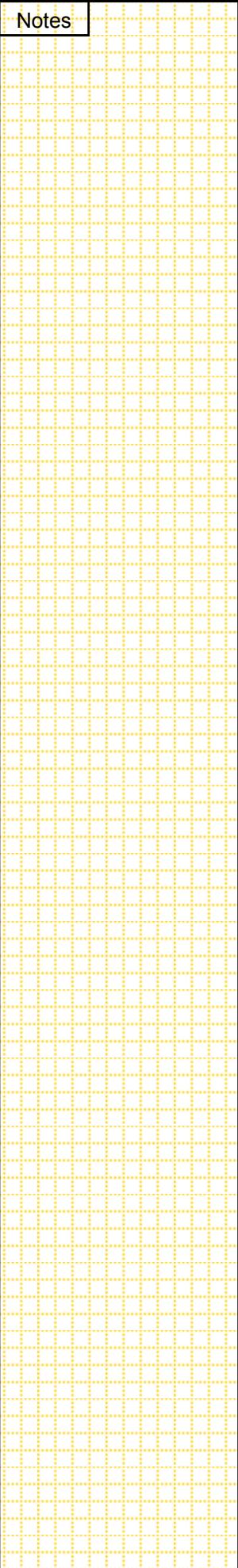
	"RECALL" key
	ETC key
	"Acknowledge alarm" key
	Without function
	Enter key
	Shift key
	Control key
	Alt key
	SPACE
	Backspace

	Clear key
	Insert key
	Tabulator
	ENTER/Input key
	"Position" operating area key
	"Program" operating area key
	"Parameter" operating area
	"Program Manager" operating area
	"Alarm/System" operating area
	Not assigned
	PageUp/PageDown keys
	Cursor keys
	Selection key/toggle key
	Alphanumeric keys Double assignment on the Shift level
	Numeric keys Double assignment on the Shift level



---

Notes



## 1 Brief description

**Module objective:**

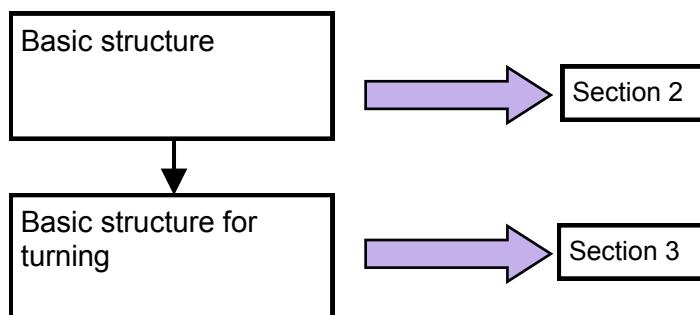
Upon completion of this module you will understand the basic structure of a NC program.

**Module description:**

We use a basic structure when writing a program to give some kind of order to the program, this will also help when performing a “Block search”.

**Module content:**

Basic structure  
Basic structure for turning  
Basic structure for milling



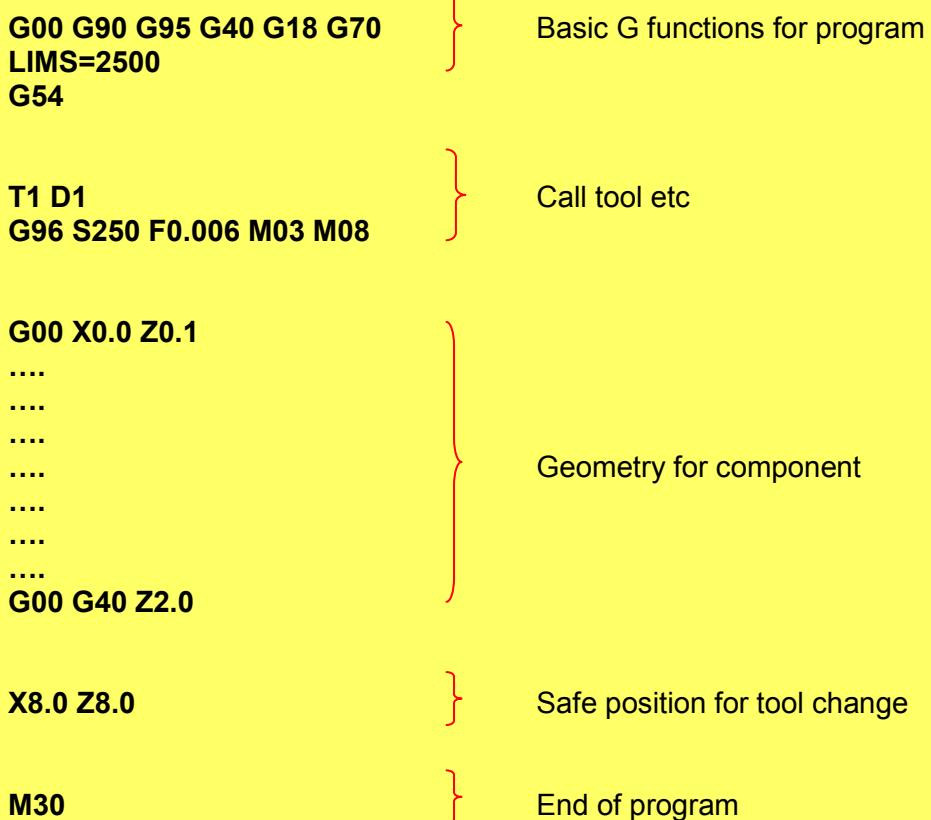
## Section 2

### Basic structure

Notes

There are five basic parts to an NC program.

1. Basic startup G functions.
2. Call tool and set spindle speed, feed, coolant.
3. Set geometry to cut component.
4. Tool to safe position
5. End of program



You will note that for each tool there is always :

Call tool etc  
Geometry for component  
Safe position for tool change

This will be the same for every tool that is written in the NC program

If you keep to a good basic structure, this will make the program safer to run and easier to perform a block search.

## Section 3

### Basic structure for turning

Notes

This is the basic structure of NC program for turning

**G00 G90 G95 G40 G18 G70  
LIMS=2500  
G54**

} Basic G functions for program

**T1 D1  
G96 S750 M03 M08**

} Call tool etc

**G00 X2.0470 Z0.1  
G01 X-0.0800 F0.006  
G00 Z0.0800  
X0.7874  
G01 Z-0.7874  
X1.1811 Z-0.9842  
X1.7716  
Z-1.9685  
X2.0470**

} Geometry for component

**G00 G40 Z2.0  
X8.0 Z8.0**

} Safe position for tool change

**T4 D1  
G95 S1500 M04 M08**

} Call tool etc

**G00 X0.0 Z0.0800  
G01 Z-0.9842 F0.0098  
G00 Z0.0800**

} Geometry for component

**G00 G40 Z0.0800  
X8.0 Z8.0**

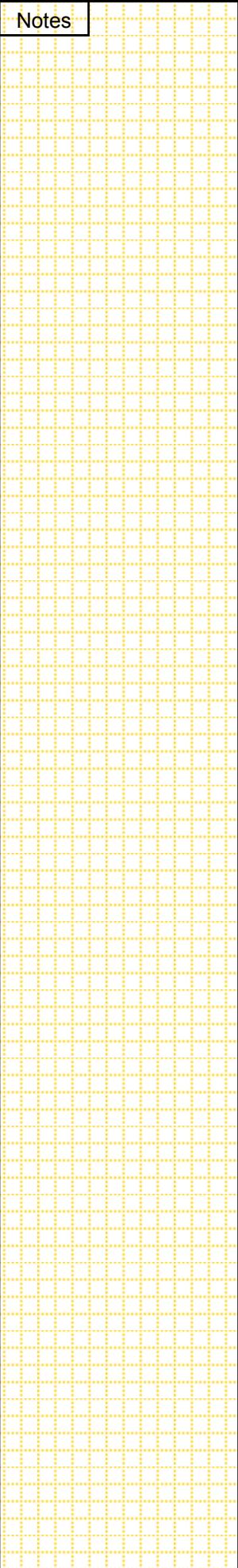
} Safe position for tool change

**M30**

} End of program

---

Notes



## 1 Brief description

**Module objective:**

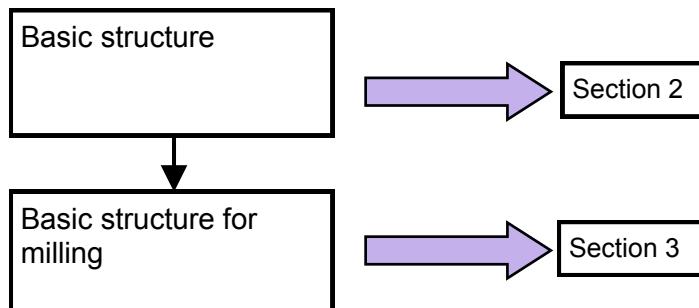
Upon completion of this module you will understand the basic structure of a NC program.

**Module description:**

We use a basic structure when writing a program to give some kind of order to the program, this will also help when performing a “Block search”.

**Module content:**

Basic structure  
Basic structure for milling



## Section 2

### Basic structure

Notes

There are five basic parts to an NC program.

1. Basic startup G functions.
2. Call tool and set spindle speed, feed, coolant.
3. Set geometry to cut component.
4. Tool to safe position
5. End of program

**G00 G90 G94 G40 G17 G70**      } Basic G functions for program

**T1  
M6  
G95 S1500 F0.006 M03 M08**      } Call tool etc

**G00 G54 X0.0 Y0.0 Z2.0**  
....  
....  
....  
....  
....  
....  
**G00 G40 Z2.0**      } Geometry for component

**X0.0 Y8.0 Z8.0**      } Safe position for tool change

**M30**      } End of program

You will note that for each tool there is always :

Call tool etc  
Geometry for component  
Safe position for tool change

This will be the same for every tool that is written in the NC program

If you keep to a good basic structure, this will make the program safer to run and easier to perform a block search.

## Section 3

### Basic structure for milling

Notes

This is the basic structure of NC program for milling

N10 G00 G90 G94 G40 G17 G70 } Basic G functions for program

N20 T1  
N30 M6  
N40 G95 S2500 F0.0098 M03 M08 } Call tool etc

N50 G00 G54 X0.0 Y0.0 Z2.0  
N60 MCALL CYCLE82(0.1968, 0.0000, 0.0800, -0.0800, 0.0000, 0.0000)

N70 X0.3937  
N80 X0.7874  
N90 X1.1811  
N100 X1.5748  
N110 X1.9685  
N120 X2.3622  
N130 MCALL } Geometry for component

N140 G00 G40 Z2.0  
N150 X0.0 Y8.0 Z8.0 } Safe position for tool change

N160 T4  
N170 M06  
N180 G95 S1500 F0.009 M05 M08 } Call tool etc

N190 G00 G54 X0.0 Y0.0 Z2.0  
N200 MCALL CYCLE82(0.1968, 0.0000, 0.0800, -0.0800, 0.0000, 0.0000)

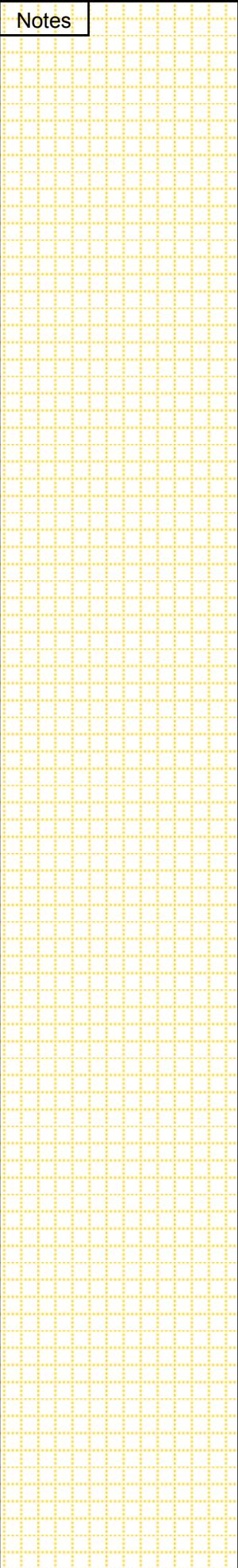
N210 X0.3937  
N220 X0.7874  
N230 X1.1811  
N240 X1.5748  
N250 X1.9685  
N260 X2.3622  
N270 MCALL } Geometry for component

N280 G00 G40 Z2.0  
N290 X0.0 Y8.0 Z8.0 } Safe position for tool change

N300 M30 } End of program

---

Notes



## 1 Brief description

**Module objective:**

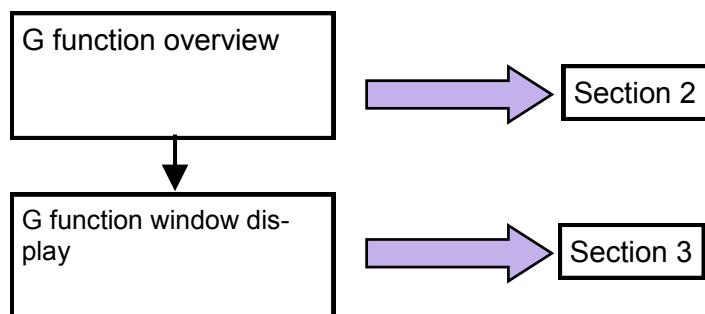
Upon completion of this module you will understand Preparatory functions (G-Codes).

**Module description:**

We use preparatory G functions according to there appropriate functional group to instruct a machine what to do via the customers NC program.

**Module content:**

G function overview  
G function window display



## Section 2

### G function overview

Notes

G function (preparatory function).

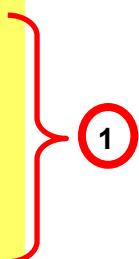
The G functions are divided into **G groups**.

Only **one** G function from each group can be written in a single block.

A G function can either be **modal** (until canceled by another function from the same group), or **non-modal** (only effective for that block it is written in).

Modal example:

```
G00 G90 G95 G40 G18
LIMS=2500
G54
T1 D1
G96 S1000 M03
G01 X0.0 Z0.0 F0.35
X20.0
Z-10.0
X30.0
Z-20.0
X40.0
G00 Z50.0 ②
M30
```



- ① These blocks are performed at a given feed rate, because G01 is modal and will not change until another code from the same group is entered
- ② At this block the speed of motion will change due to the change of the G function group, from G01 to G00.

Non-modal example:

```
G00 G90 G95 G40 G18
LIMS=2500
G54
T2 D1
G96 S1000 M03
X0.0 Z50.0
G01 Z0.0 F0.35
G04 F10.0 ③
G00 Z50.0
M30
```

- ③ This G function is only effective in this block, to use this function again you must type it in again in another block.

## Section 2

### G function overview

Notes

#### Group 1: Modally valid motion commands

Name	Meaning	Machine
G00	Rapid traverse	M/T
G01*#	Linear interpolation	M/T
G02	Circular interpolation clockwise	M/T
G03	Circular interpolation counter-clockwise	M/T
G33	Thread cutting with constant lead	M/T

#### Group 2: Non-modally valid motion, dwell

Name	Meaning	Machine
G04	Dwell time preset	M/T
G63	Tapping without synchronization	M/T
G74	Reference point approach with synchronization	M/T
G75	Fixed point approach	M/T

#### Group 3: Programmable frame, working area limitation

Name	Meaning	Machine
TRANS	Translation, programmable	M/T
ROT	Rotation, programmable	M/T
SCALE	Scaling, programmable	M/T
MIRROR	Mirroring, programmable	M/T
G25	Minimum working area limitation/spindle speed limitation	M/T
G26	Maximum working area limitation/spindle speed limitation	M/T

#### Group 6: Plane selection

Name	Meaning	Machine
G17#	Plane selection 1st - 2nd geometry axis	M/T
G18*	Plane selection 3rd - 1st geometry axis	M/T
G19	Plane selection 2nd - 3rd geometry axis	M/T

Default for turning \* / default for milling #

M = milling / T = turning

## Section 2

### G function overview

Notes

#### Group 7: Tool radius compensation

Name	Meaning	Machine
G40*#	No tool radius compensation	M/T
G41	Tool radius compensation left of contour	M/T
G42	Tool radius compensation right of contour	M/T

#### Group 8: Settable zero offset

Name	Meaning	Machine
G500*#	Deactivate all settable G54—G59 if G500 not contain a value	M/T
G54	Settable zero offset	M/T
G55	Settable zero offset	M/T
G56	Settable zero offset	M/T
G57	Settable zero offset	M/T
G58	Settable zero offset	M/T
G59	Settable zero offset	M/T

#### Group 9: Frame suppression

Name	Meaning	Machine
G53	Suppression of current zero offset: including active settable frame G54 - G59	M/T

#### Group 10: Exact stop - continuous-path mode

Name	Meaning	Machine
G60*#	Velocity reduction, exact positioning	M/T
G64	Continuous – path mode	M/T

#### Group 11: Exact stop, non-modal

Name	Meaning	Machine
G09	Velocity reduction, exact positioning	M/T

Default for turning \* / default for milling #

M = milling / T = turning

## Section 2

### G function overview

Notes

#### Group 13: Workpiece measuring inch/metric

Name	Meaning	Machine
G70	Input system inches (lengths)	M/T
G71*#	Input system metric (lengths)	M/T

#### Group 14: Workpiece measuring absolute/incremental

Name	Meaning	Machine
G90*#	Absolute dimensions input	M/T
G91	Incremental dimension input	M/T

#### Group 15: Feed type

Name	Meaning	Machine
G94#	Linear feed mm/min, inch/min	M/T
G95*	Rotational feed in mm/rev, inch/rev	M/T
G96	Constant cutting speed (type of feed as for G95) ON	T
G97	Constant cutting speed (type of feed as for G95) OFF	T

#### Group 16: Feed override on inside and outside curvature

Name	Meaning	Machine
CFC*#	Constant feed at contour	M/T
CFTCP	Constant feed in tool center point (center-point path)	M/T
CFIN	Constant feed at internal radius, acceleration at external radius	M/T

Default for turning \* / default for milling #  
M = milling / T = turning

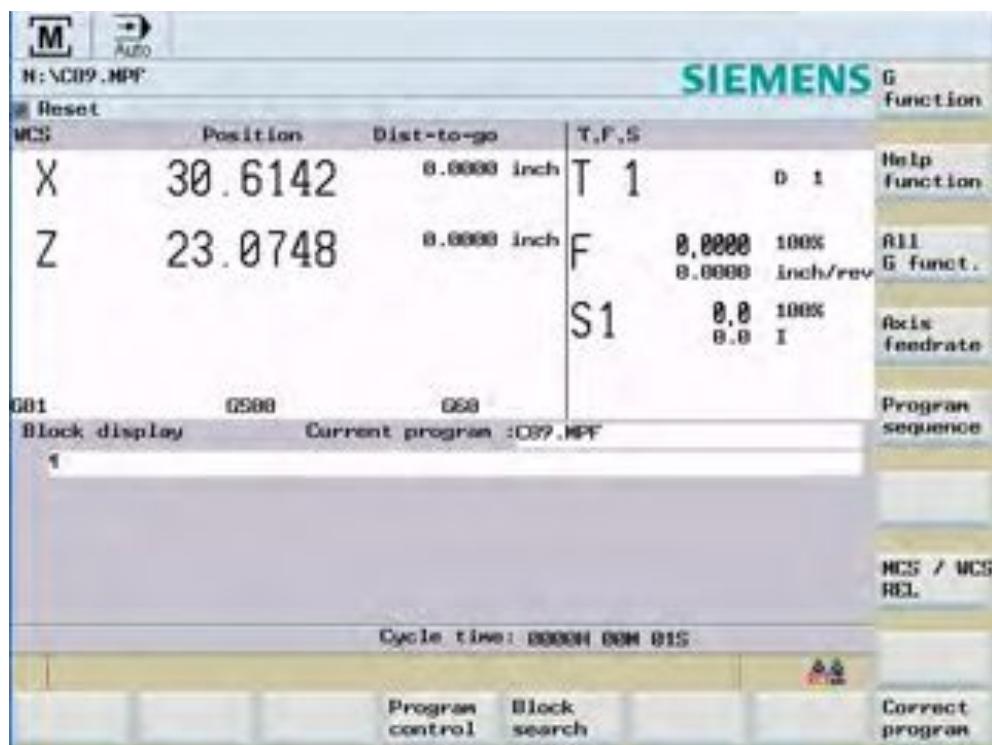
## Section 3

### G function window display

Notes

To look at the active G-codes, you can activate a window called G FUNCTION.

Follow this sequence.



G  
function

This “G function” softkey is present in modes:



## Section 3

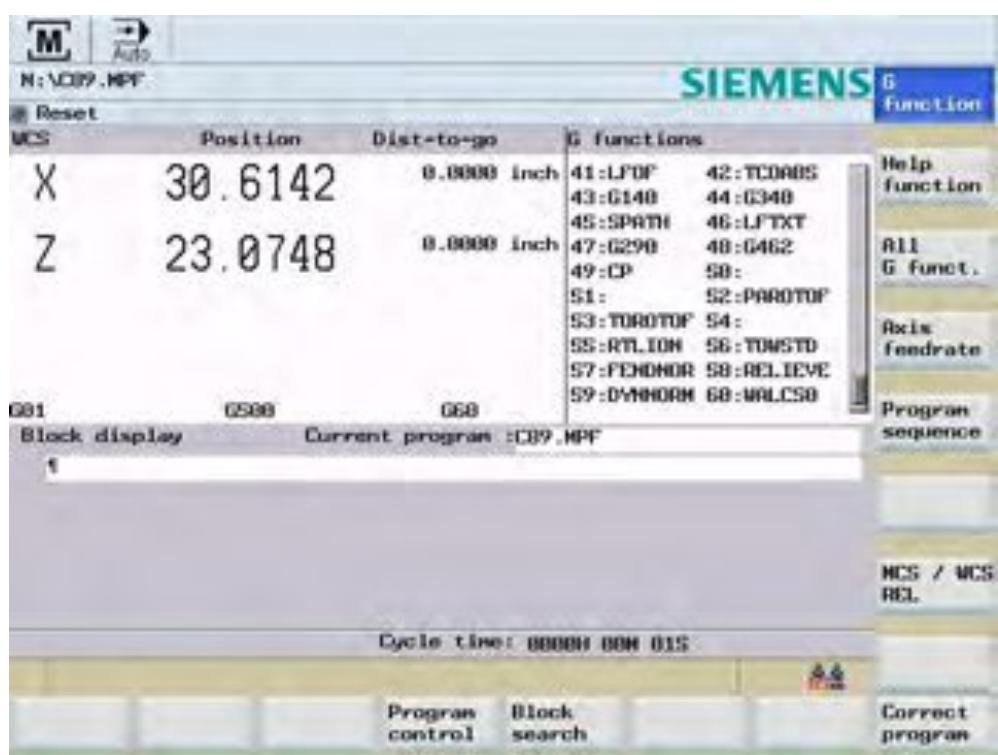
### G function window display

Notes



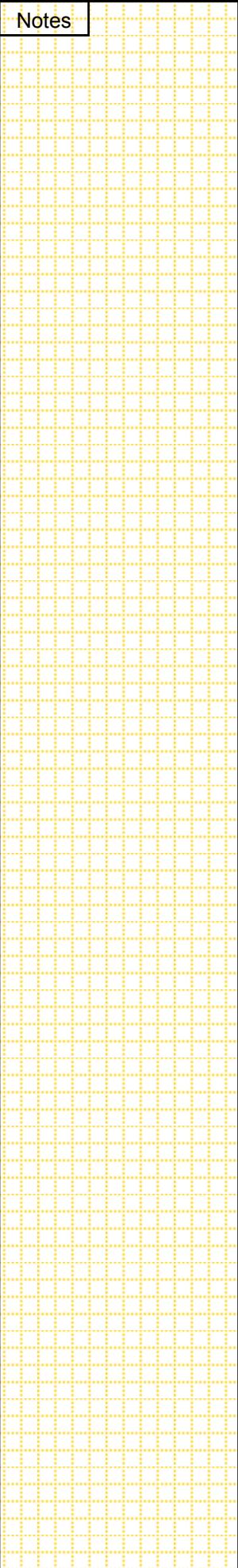
In this window, it will show you every group number and the active G-code for that group at that present time.

Use the button to scroll down or up to search for the required G-code group.



---

Notes



## 1 Brief description

**Module objective:**

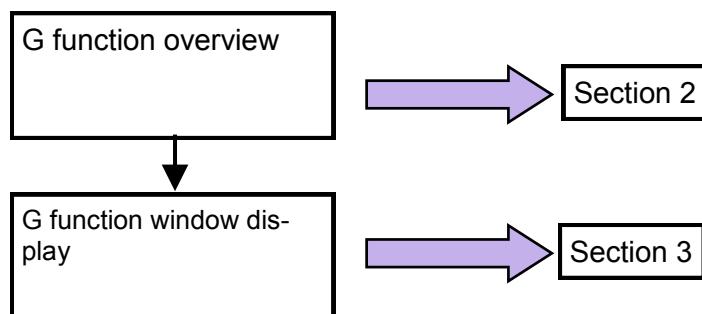
Upon completion of this module you will understand Preparatory functions (G-Codes).

**Module description:**

We use preparatory G functions according to there appropriate functional group to instruct a machine what to do via the customers NC program.

**Module content:**

G function overview  
G function window display



## Section 2

### G function overview

Notes

G function (preparatory function).

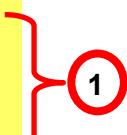
The G functions are divided into **G groups**.

Only **one** G function from each group can be written in a single block.

A G function can either be **modal** (until canceled by another function from the same group), or **non-modal** (only effective for that block it is written in).

Modal example:

```
G00 G90 G94 G40 G17  
T1 D1  
G54 S1000 M03  
G01 X0.0 Y0.0 Z0.0 F200  
X100.0  
Y100.0  
X0.0  
Y0.0  
G00 Z50.0 2  
M30
```



- 1** These blocks are performed at a given feed rate, because G01 is modal and will not change until another code from the same group is entered
- 2** At this block the speed of motion will change due to the change of the G function group, from G01 to G00.

Non-modal example:

```
G00 G90 G94 G40 G17  
T2 D1  
G54 S1000 M03  
X0.0 Y0.0 Z50.0  
G01 Z0.0 F200  
G04 F10.0 3  
G00 Z50.0  
M30
```

- 3** This G function is only effective in this block, to use this function again you must type it in again in another block.

## Section 2

### G function overview

Notes

#### Group 1: Modally valid motion commands

Name	Meaning	Machine
G00	Rapid traverse	M/T
G01*#	Linear interpolation	M/T
G02	Circular interpolation clockwise	M/T
G03	Circular interpolation counter-clockwise	M/T
G33	Thread cutting with constant lead	M/T

#### Group 2: Non-modally valid motion, dwell

Name	Meaning	Machine
G04	Dwell time preset	M/T
G63	Tapping without synchronization	M/T
G74	Reference point approach with synchronization	M/T
G75	Fixed point approach	M/T

#### Group 3: Programmable frame, working area limitation

Name	Meaning	Machine
TRANS	Translation, programmable	M/T
ROT	Rotation, programmable	M/T
SCALE	Scaling, programmable	M/T
MIRROR	Mirroring, programmable	M/T
G25	Minimum working area limitation/spindle speed limitation	M/T
G26	Maximum working area limitation/spindle speed limitation	M/T

#### Group 6: Plane selection

Name	Meaning	Machine
G17#	Plane selection 1st - 2nd geometry axis	M/T
G18*	Plane selection 3rd - 1st geometry axis	M/T
G19	Plane selection 2nd - 3rd geometry axis	M/T

Default for turning \* / default for milling #

M = milling / T = turning

## Section 2

### G function overview

Notes

#### Group 7: Tool radius compensation

Name	Meaning	Machine
G40*#	No tool radius compensation	M/T
G41	Tool radius compensation left of contour	M/T
G42	Tool radius compensation right of contour	M/T

#### Group 8: Settable zero offset

Name	Meaning	Machine
G500*#	Deactivate all settable G54—G59 if G500 not contain a value	M/T
G54	Settable zero offset	M/T
G55	Settable zero offset	M/T
G56	Settable zero offset	M/T
G57	Settable zero offset	M/T
G58	Settable zero offset	M/T
G59	Settable zero offset	M/T

#### Group 9: Frame suppression

Name	Meaning	Machine
G53	Suppression of current zero offset: including active settable frame G54 - G59	M/T

#### Group 10: Exact stop - continuous-path mode

Name	Meaning	Machine
G60*#	Velocity reduction, exact positioning	M/T
G64	Continuous – path mode	M/T

#### Group 11: Exact stop, non-modal

Name	Meaning	Machine
G09	Velocity reduction, exact positioning	M/T

Default for turning \* / default for milling #

M = milling / T = turning

## Section 2

### G function overview

Notes

#### Group 13: Workpiece measuring inch/metric

Name	Meaning	Machine
G70	Input system inches (lengths)	M/T
G71*#	Input system metric (lengths)	M/T

#### Group 14: Workpiece measuring absolute/incremental

Name	Meaning	Machine
G90*#	Absolute dimensions input	M/T
G91	Incremental dimension input	M/T

#### Group 15: Feed type

Name	Meaning	Machine
G94#	Linear feed mm/min, inch/min	M/T
G95*	Rotational feed in mm/rev, inch/rev	M/T
G96	Constant cutting speed (type of feed as for G95) ON	T
G97	Constant cutting speed (type of feed as for G95) OFF	T

#### Group 16: Feed override on inside and outside curvature

Name	Meaning	Machine
CFC*#	Constant feed at contour	M/T
CFTCP	Constant feed in tool center point (center-point path)	M/T
CFIN	Constant feed at internal radius, acceleration at external radius	M/T

Default for turning \* / default for milling #  
M = milling / T = turning

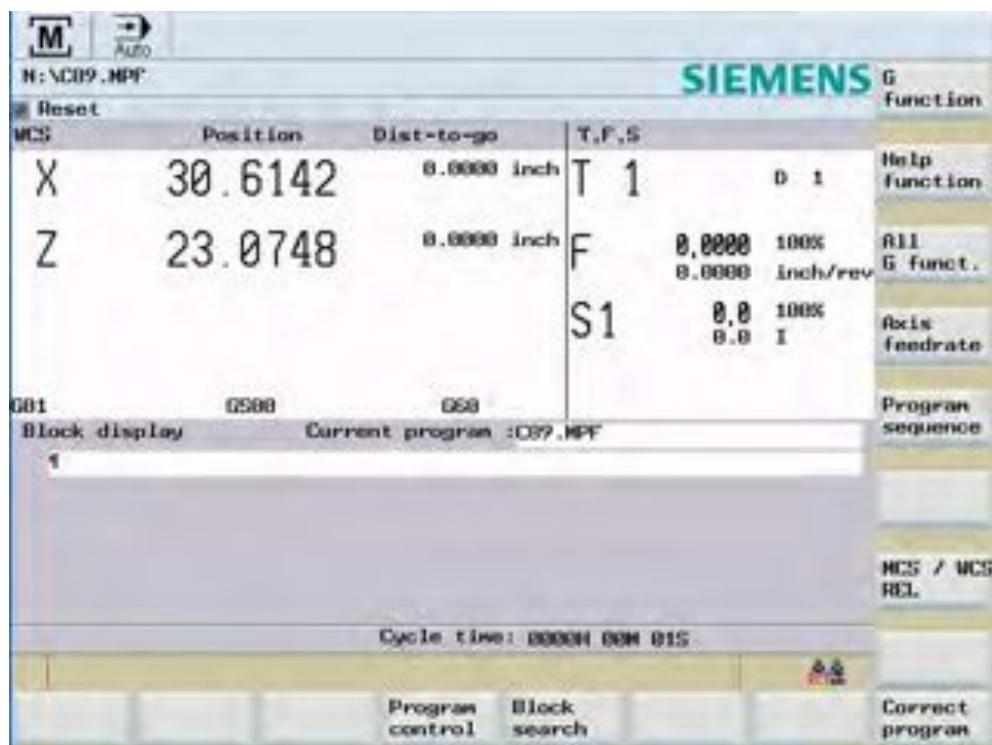
## Section 3

### G function window display

Notes

To look at the active G-codes, you can activate a window called G FUNCTION.

Follow this sequence.



G  
function

This “G function” softkey is present in modes:



## Section 3

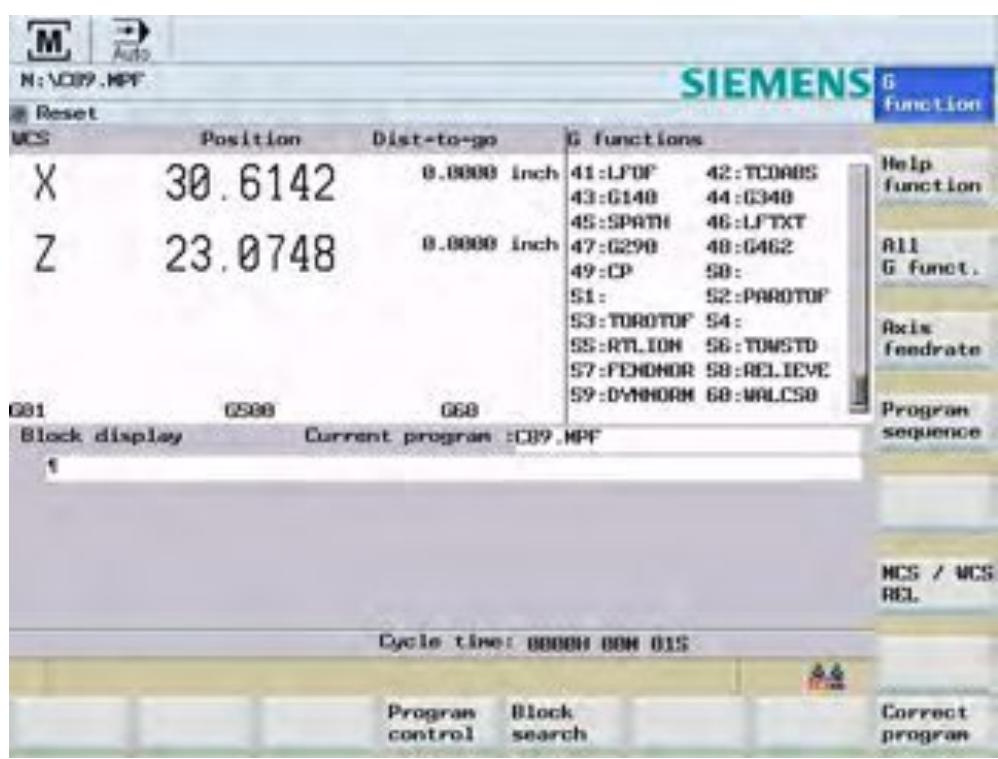
### G function window display

Notes



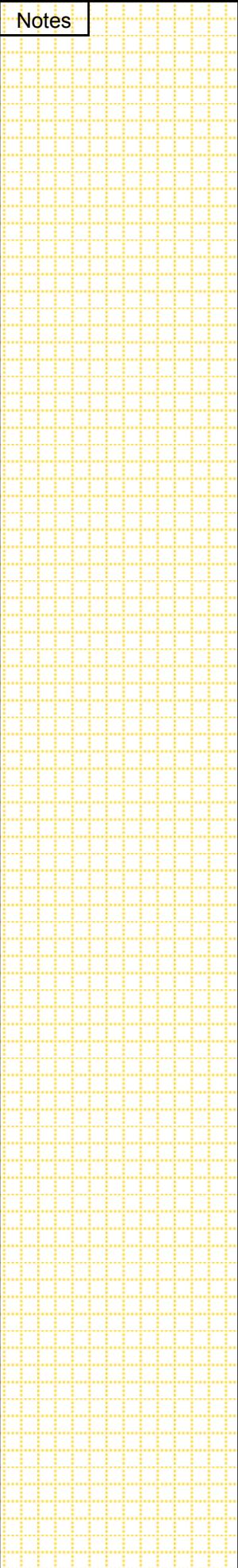
In this window, it will show you every group number and the active G-code for that group at that present time.

Use the button to scroll down or up to search for the required G-code group.



---

Notes



## 1 Brief description

### Module objective:

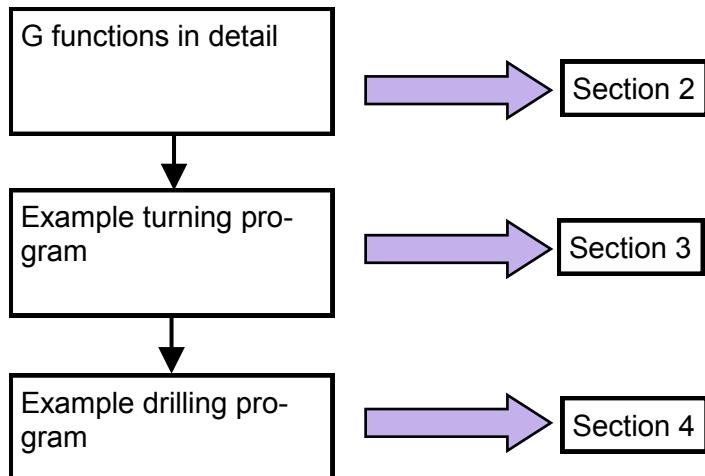
Upon completion of this module you will understand commonly used G functions (G-Codes) for turning in detail.

### Module description:

We use G functions according to their appropriate functional group to instruct a machine what to do, but a structure should be kept to.

### Module content:

G functions in detail  
Example turning program  
Example drilling program



## Section 2

### G function in detail

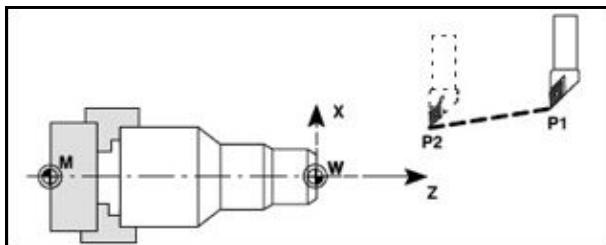
#### Rapid traverse movement G00

##### Function

You can use a rapid traverse movement to position the tool rapidly, to travel around the Workpiece or to approach tool change locations.  
Note: when there is a two axis movement, both axes interpolate to their end position, arriving at the same time.

##### Programming

G00 X.. Z..  
Or  
G0 X.. Z..



Notes

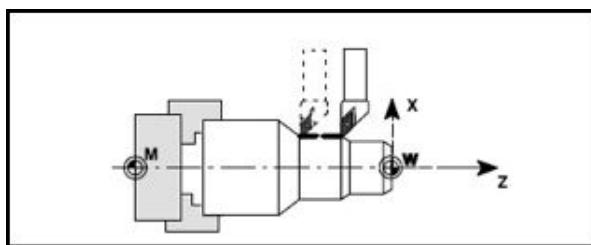
#### Linear interpolation G01

##### Function

With G01, the tool travels along a straight line.

##### Programming

G01 X.. Z.. F..  
Or  
G1 X.. Z.. F..



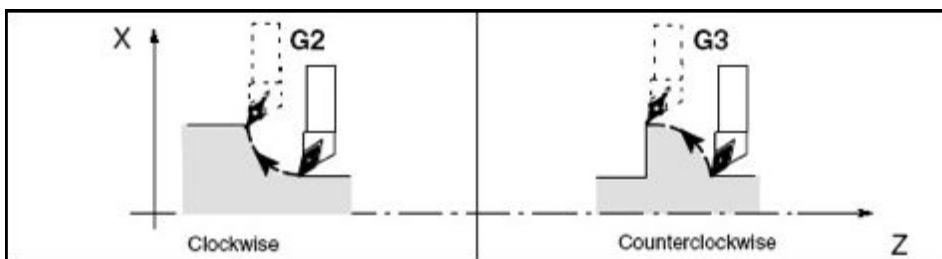
#### Circular interpolation, G02/G03

##### Function

This function allows you to program an arc either in clockwise (G02) or counter-clockwise (G03) direction.

##### Programming

G02/G03 X.. Z.. I=AC(..) K=AC(..)      Absolute centre point  
Or  
G02/G03 X.. Z.. I.. K..      Incremental centre point  
Or  
G02/G03 X.. Z.. CR=..      Circle radius CR=



## Section 2

### G function in detail

#### Dwell time G4

##### Function

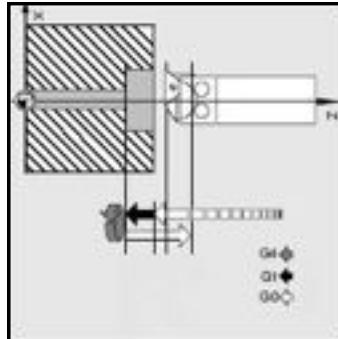
You can use G4 to interrupt workpiece machining between two NC blocks for the programmed length of time, e.g. dwell at bottom of hole.

##### Programming

G4 F..      F = Time (in seconds)

Or

G4 S..      S = Rotations



Notes

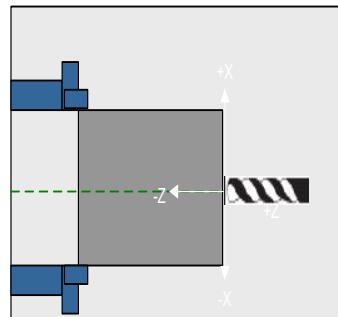
#### Plane selection G17

##### Function

This plane is selected when milling or drilling at the **end face** of the turned part.

##### Programming

G17



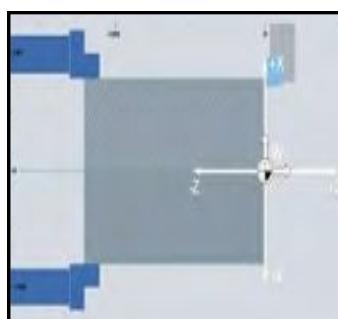
#### Plane selection G18

##### Function

This is the default plane, and is used when stock removal, threading, and grooving of the turned part.

##### Programming

G18



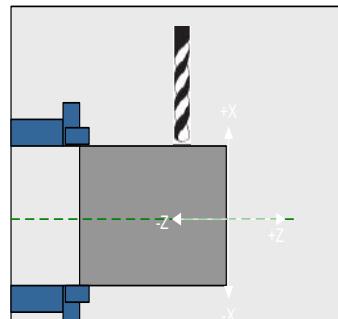
#### Plane selection G19

##### Function

This plane is selected when milling or drilling at the **peripheral surface** of the turned part.

##### Programming

G19

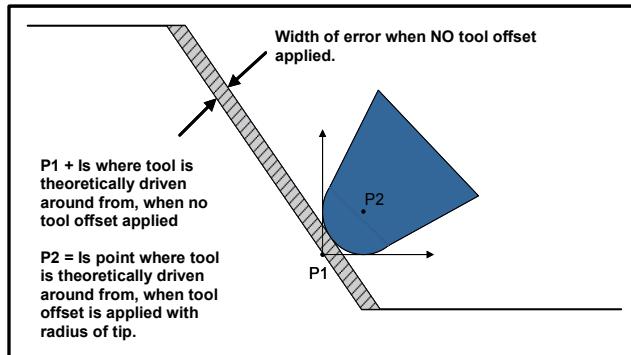


## Section 2

### G function in detail

#### Tool Compensation

Without tool compensation, there will always be an error when turning a contour, radius, and chamfer, that involves a two axis movement.



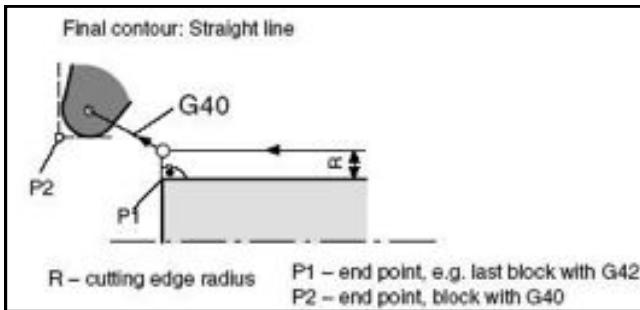
Notes

#### Tool radius compensation G40

##### Function

To deactivate tool compensation

##### Programming G40

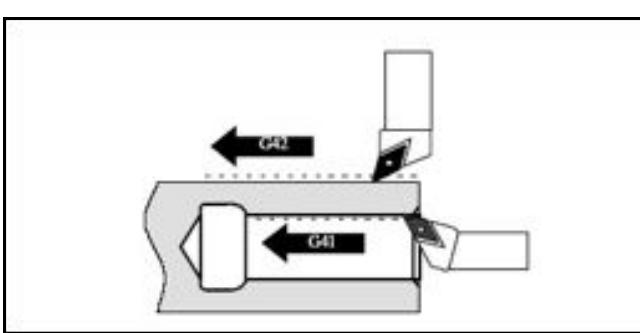


#### Tool radius compensation G41

##### Function

To activate tool compensation, with the tool operating to the left of the contour in the machining direction. Most commonly used when boring inside a hole towards the chuck

##### Programming G41



#### Tool radius compensation G42

##### Function

To activate tool compensation, with the tool operating to the right of the contour in the machining direction. Most commonly used when turning externally towards the chuck

##### Programming G42

## Section 2

### G function in detail

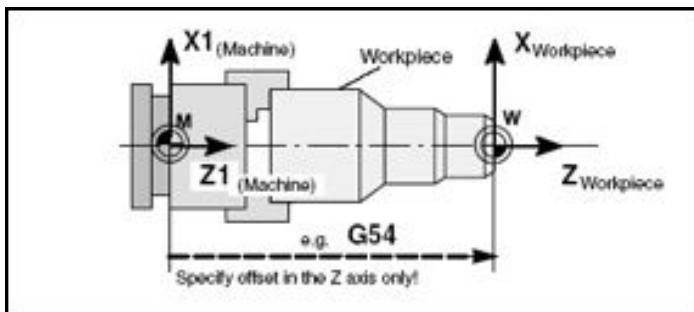
#### Zero offset G54

##### Function

The settable zero offset relates the Workpiece zero on all axes to the machine zero offset.

##### Programming

G54



#### Absolute dimension G90

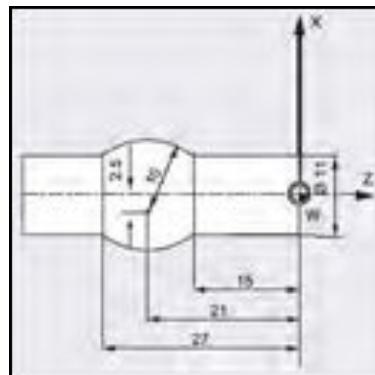
##### Function

With the G90 command all dimensions are related from your active zero workpiece.

##### Programming

G90

X.. Z..



#### Feed rate G95

##### Function

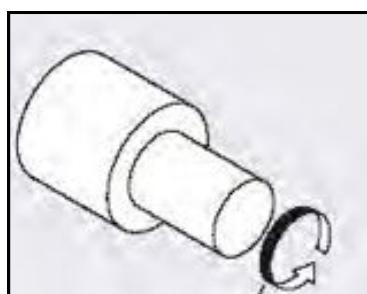
This is how the feedrate will act to the axis that is moving in relationship to the spindle

The spindle will be programmed in direct RPM and the feedrate will be MM/REV

G95 is usually used when drilling.

##### Programming

G95 S.. F..



Notes

## Section 2

### G function in detail

Notes

#### Feed rate G96

##### Function

This is how the feedrate will act to the axis that is moving in relationship to the spindle

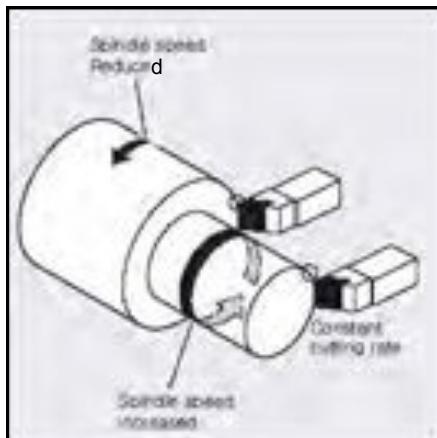
The spindle will be programmed in constant surface speed (m/min) and the feedrate will be MM/REV

G96 is usually used when turning.

##### Programming

G96 S.. F..

Note: as the tool gets closer to the centre line the spindle speed will increase. As the tool gets further away from the centre line the spindle speed will decrease.



#### Exact stop/continuous-path control: G09, G60, G64

##### Function

Will set the block end behavior movement and how to continue with the next. E.g. G09/G60 the velocity for reaching the exact end position is to decelerate the velocity to zero. With G64, the object is to avoid deceleration at the end of the block.

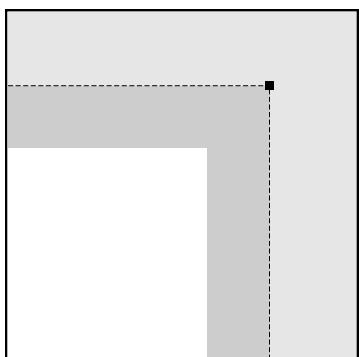
##### Programming

G09 ; Exact stop - non-modal

G60 ; Exact stop - modally effective

G64 ; Continuous path/control mode

G60



G64



## Section 3

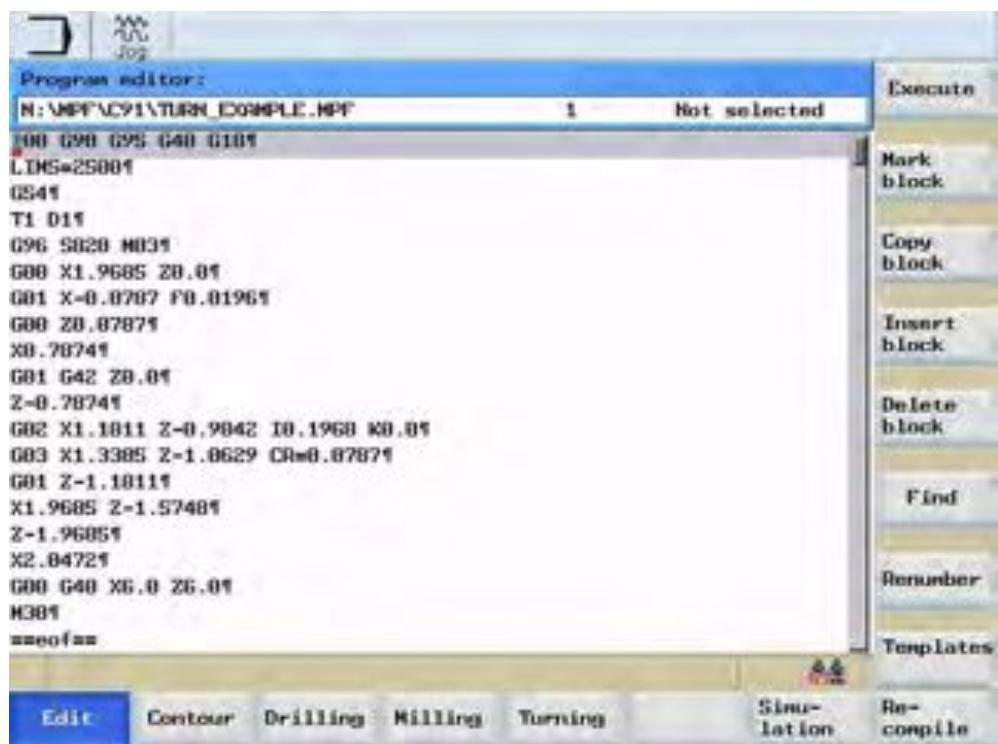
### Example turning program

Notes

```
G00 G90 G95 G40 G18          ;default G-code line
LIMS=2500
G54
T1 D1
G96 S820 M03
G00 X1.9685 Z0.0
G01 X-0.0787 F0.0196
G00 Z0.0787
X0.7874
G01 G42 Z0.0
Z-0.7874
G02 X1.1811 Z-0.9842 I0.1968 K0.0
G03 X1.3385 Z-1.0629 CR=0.0787
G01 Z-1.1811
X1.9685 Z-1.5748
Z-1.9685
X2.0472
G00 G40 X6.0 Z6.0
M30
```

;zero offset for workpiece  
;call tool  
;speed in FT/MIN feed REV/MIN  
;rapid to safe position  
;face off component  
;rapid off front face  
;rapid to diameter  
;feed onto front face with tool comp  
;clockwise arc using I and K  
;counterclockwise arc  
;feed up a slope  
;feed off of finished component  
;rapid to tool change position  
;end of program

This is the same program typed into the control

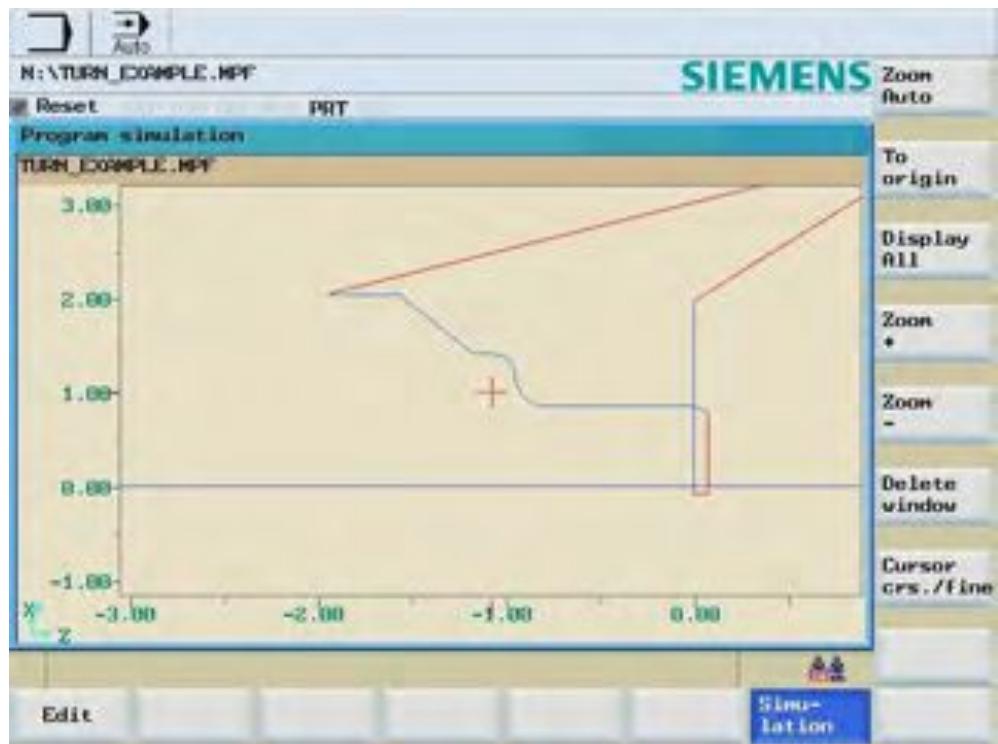


## Section 3

### Example turning program

Notes

The same program run in simulation



## Section 4

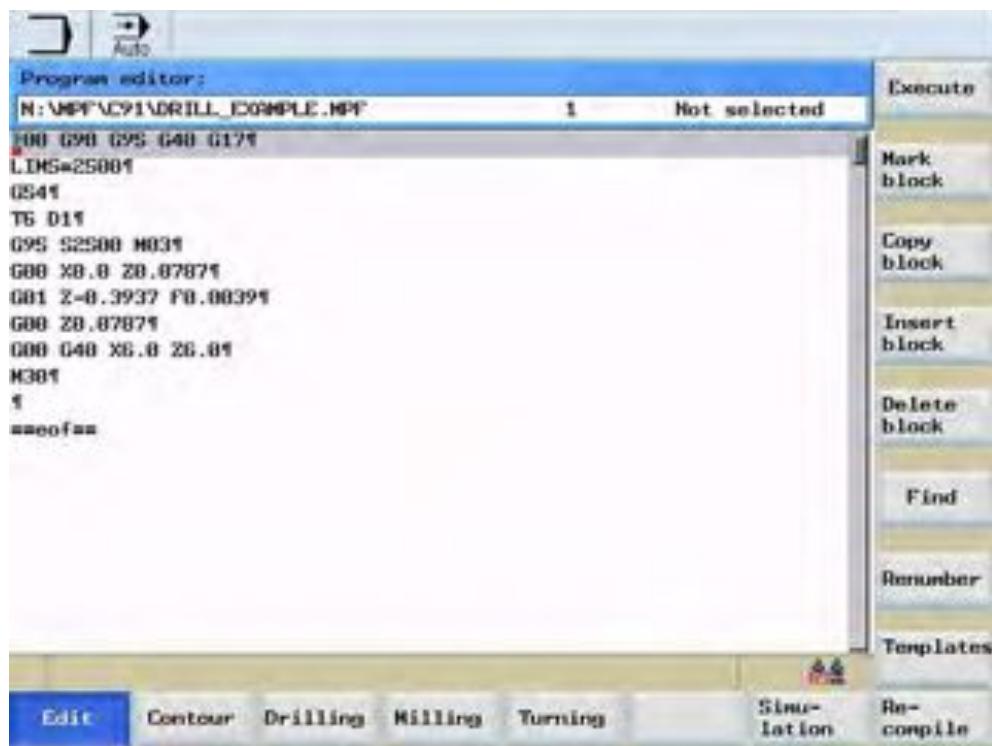
### Example drilling program

```
G00 G90 G95 G40 G17  
LIMS=2500  
G54  
T6 D1  
G95 S2500 M03  
G00 X0.0 Z0.0787  
G01 Z-0.3937 F0.0039  
G00 Z0.0787  
G00 G40 X6.0 Z6.0  
M30
```

;default G-code line  
;zero offset for workpiece  
;call tool  
;spindle speed in RPM feed REV/MIN  
;rapid to safe position  
;feed to bottom of hole  
;rapid out of hole  
;rapid to tool change position  
;end of program

Notes

This is the same program typed into the control

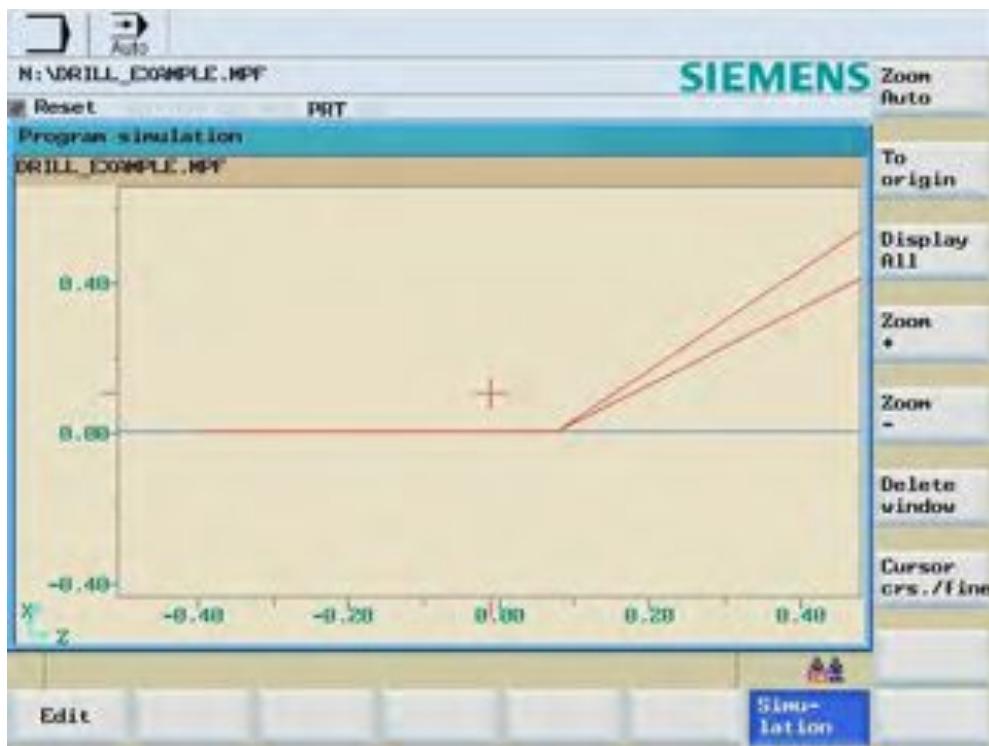


## Section 4

### Example drilling program

The same program run in simulation

Notes



## 1 Brief description

**Module objective:**

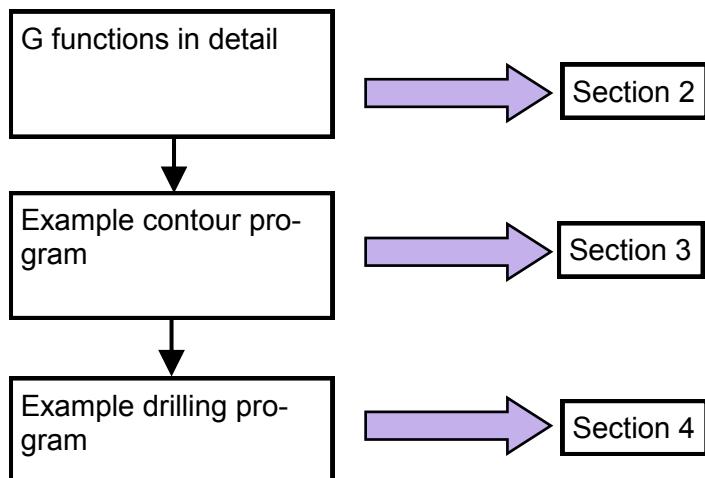
Upon completion of this module you will understand commonly used G functions (G-Codes) for milling in detail.

**Module description:**

We use G functions according to their appropriate functional group to instruct a machine what do , but a certain structure should be kept to.

**Module content:**

G functions in detail  
Example contour program  
Example drilling program



## Section 2

### G functions in detail

Notes

#### Rapid traverse movement G00

##### Function

You can use rapid traverse movement, to position the tool rapidly, to travel around the workpiece or to approach tool change locations.

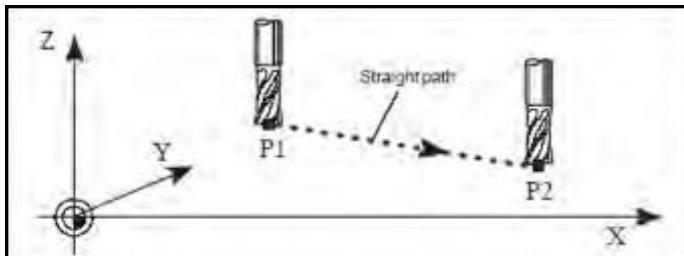
Note: when there is a two axis movement, both axes interpolate to their end position, arriving at the same time.

##### Programming

G00 X.. Y.. Z..

Or

G0 X.. Y.. Z..



#### Linear interpolation G01

##### Function

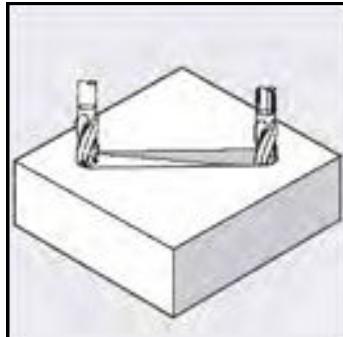
With G01, the tool travels along a straight line.

##### Programming

G01 X.. Y.. Z.. F..

Or

G1 X.. Y.. Z.. F..



#### Circular interpolation, G02/G03

##### Function

This function allows you to program an arc either in clockwise (G02) or counter-clockwise (G03) direction.

##### Programming

G02/G03 X.. Y.. Z.. I=AC(..) J=AC(..) K=AC(..)

absolute centre point

Or

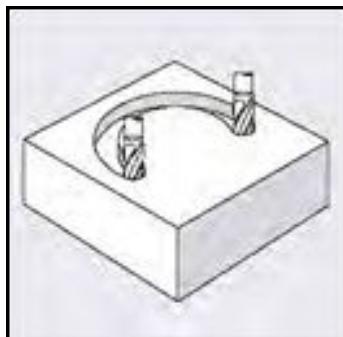
G02/G03 X.. Y.. Z.. I.. J.. K..

Incremental centre point

Or

G02/G03 X.. Y.. Z.. CR=..

Circle radius CR=



## Section 2

### G functions in detail

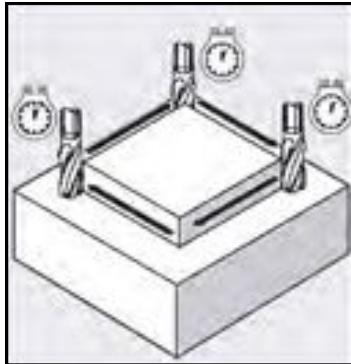
#### Dwell time G4

##### Function

You can use G4 to interrupt workpiece machining between two NC blocks for the programmed length of time, e.g. dwell at bottom of hole.

##### Programming

G4 F..      F = Time (time in seconds)  
Or  
G4 S..      S = Rotations



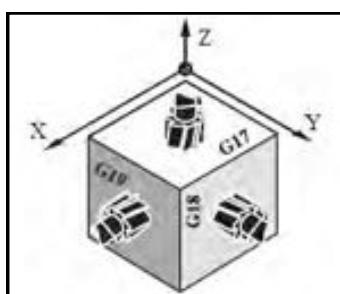
Notes

#### Plane selection G17

##### Function

To select the infeed feed axis when milling or drilling, this is the default G function.

##### Programming G17



## Section 2

### G functions in detail

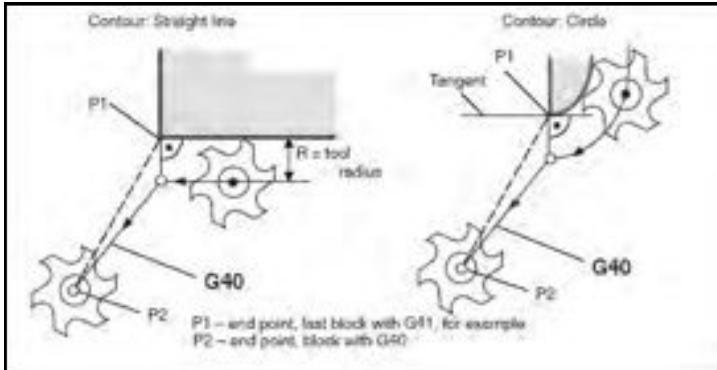
Notes

#### Tool radius compensation G40

Function

To deactivate tool compensation

Programming  
G40

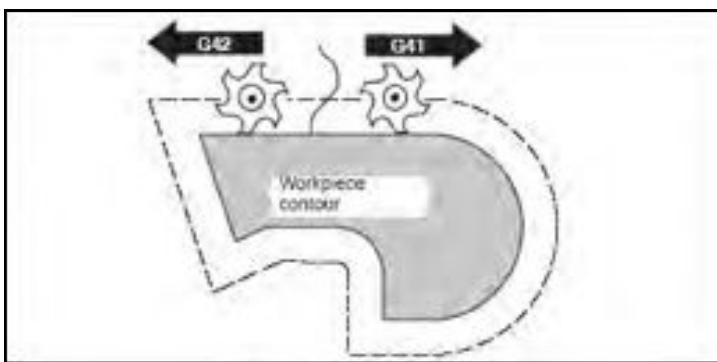


#### Tool radius compensation G41

Function

To activate tool compensation, with the tool operating to the left of the contour in the machining direction.

Programming  
G41



#### Tool radius compensation G42

Function

To activate tool compensation, with the tool operating to the right of the contour in the machining direction.

Programming  
G42

## Section 2

### G functions in detail

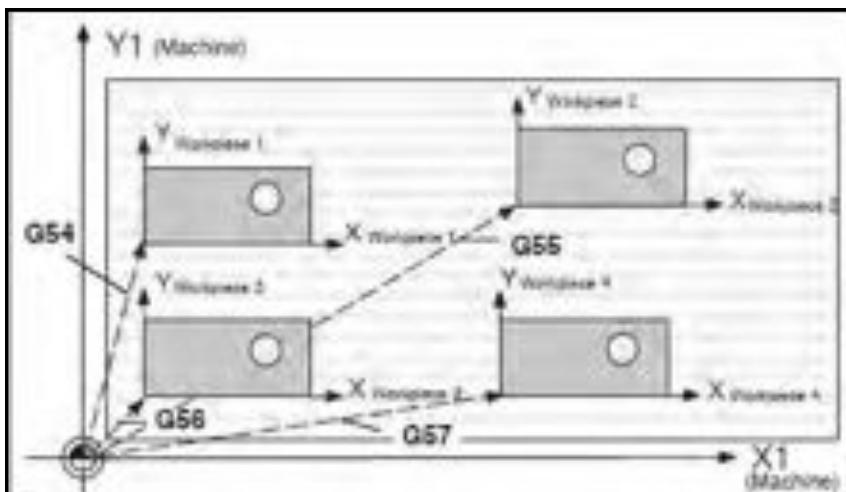
#### Zero offset G54 to G59

##### Function

The settable zero offset relates the workpiece zero on all axes to the machine zero offset.

##### Programming

G54 to G59



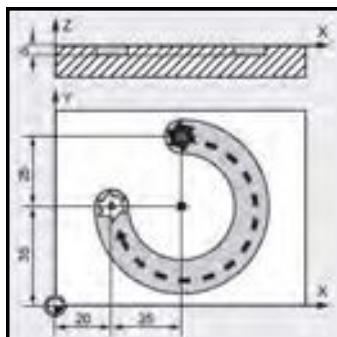
#### Absolute dimension G90

##### Function

With the G90 command all dimensions are related from your active zero offset.

##### Programming

G90  
X.. Y.. Z..



## Section 2

### G functions in detail

#### Feed rate G94

##### Function

This is how the feedrate will act to the axis that is moving in relationship to the spindle.

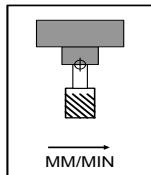
The spindle will be programmed in direct RPM and the feedrate will be MM/MIN.

Note: the axis will move without the spindle rotating.

##### Programming

G94

S.. F..



#### Feed rate G95

##### Function

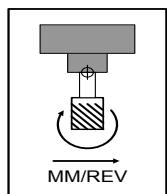
This is how the feedrate will act to the axis that is moving in relationship to the spindle.

The spindle will be programmed in direct RPM and the feedrate will be MM/REV

##### Programming

G95

S.. F..



#### Exact stop/continuous-path control: G09, G60, G64

##### Function

Will set the block end behavior movement and to continue with the next. E.g. G09/G60 the objective for reaching the exact end position is to decelerate the velocity to zero. With G64, the object is to avoid deceleration at the end of the block.

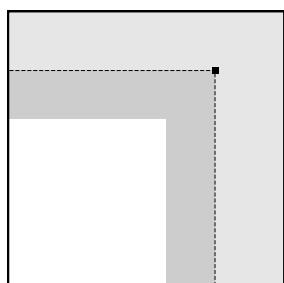
##### Programming

G09 ; Exact stop - non-modal

G60 ; Exact stop - modally effective

G64 ; Continuous path-control mode

G60



G64



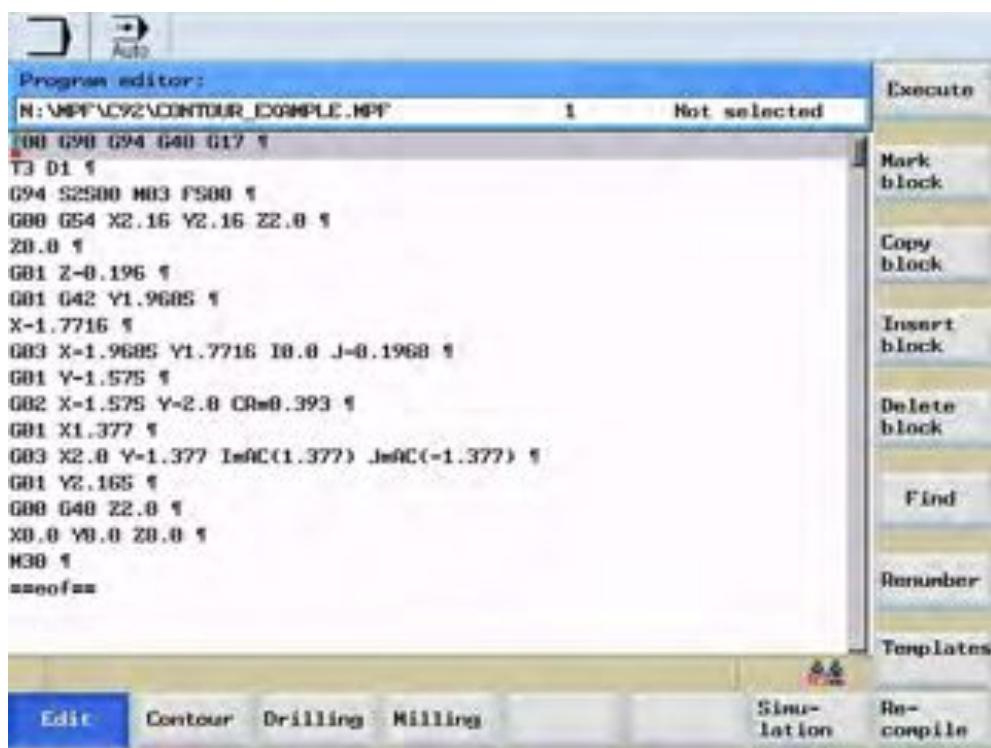
## Section 3

### Example contour program

<b>G00 G90 G94 G40 G17</b>	;Basic G-code commands
<b>T1 D1</b>	;call tool
<b>G94 S2500 M03 F20</b>	;speed in RPM feed INCH/MIN
<b>G00 G54 X2.16 Y2.16 Z2.0</b>	;start of contour
<b>Z0.0</b>	;rapid to safe position
<b>G01 Z-0.196</b>	;feed onto workpiece
<b>G01 G42 Y1.9685</b>	;apply cutter compensation
<b>X-1.7716</b>	;coordinates for contour
<b>G03 X-1.9685 Y1.7716 I0.0 J-0.168</b>	;arc using incremental start point
<b>G01 Y-1.575</b>	;coordinates for contour
<b>G02 X-1.575 Y-2.0 CR=0.393</b>	;arc using circle radius
<b>G01 X1.377</b>	;coordinates for contour
<b>G03 X2.0 Y-1.377 I=AC(1.377) J=AC(-1.377)</b>	;arc using absolute start point
<b>G01 Y2.165</b>	;coordinates for contour
<b>G00 G40 Z2.0</b>	;cancel cut comp
<b>X0.0 Y8.0 Z8.0</b>	;safe tool change position
<b>M30</b>	;end of program

Notes

This is the same program typed into the control:



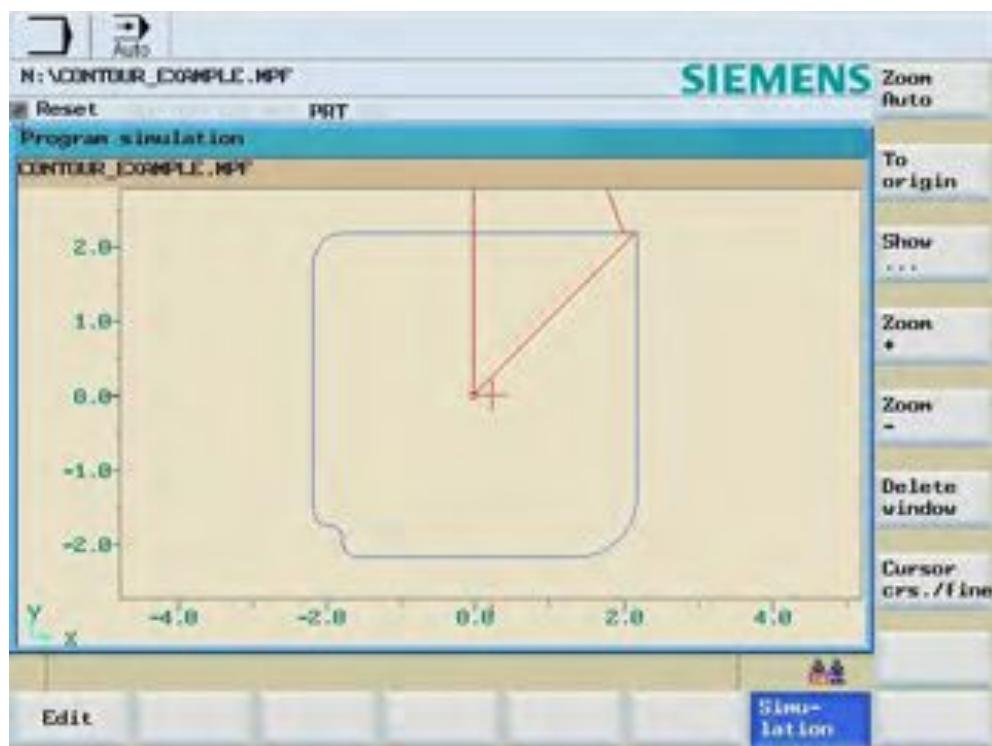
## Section 3

### Example contour program

Notes

The same program run in simulation:

Simu-  
lation



Edit

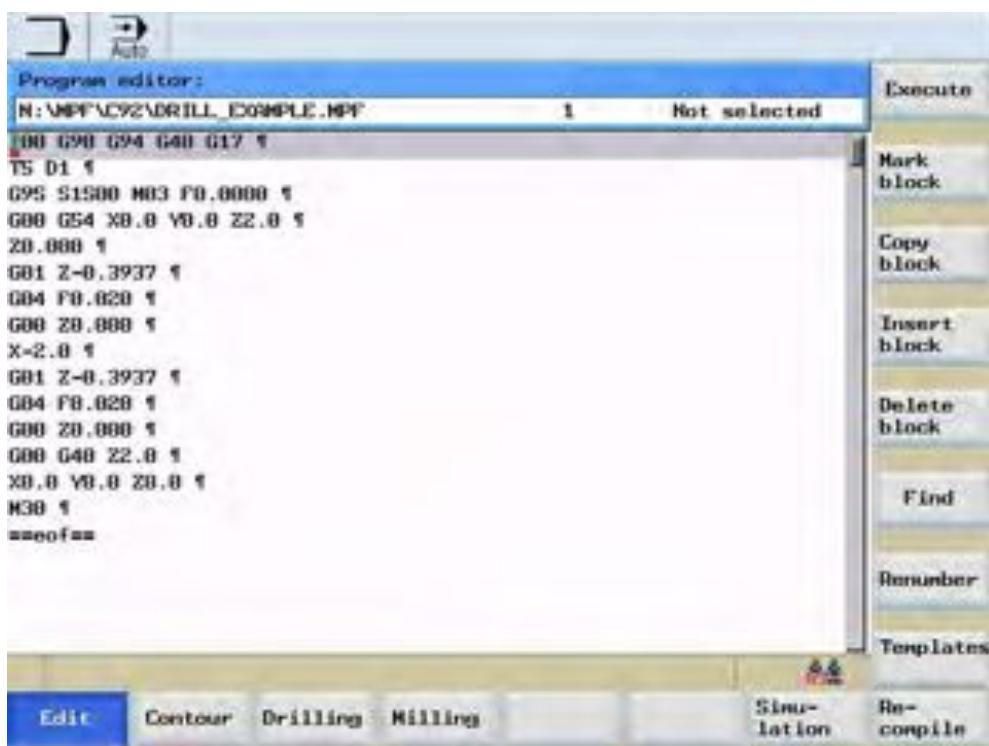
## Section 4

### Example drilling program

Notes

<b>G00 G90 G95 G40 G17</b>	;basic G-code commands
<b>T1 D1</b>	;call tool
<b>G95 S1500 M03 F0.006</b>	;spindle speed in RPM feed MM/REV
<b>G00 G54 X0.0 Y0.0 Z2.0</b>	;rapid to safe position start of contour
<b>Z0.080</b>	;rapid to safe position
<b>G01 Z-0.3937</b>	;feed onto workpiece
<b>G04 F0.020</b>	;dwell at bottom of hole
<b>G00 Z0.80</b>	;rapid out of hole
<b>X-2.0</b>	;coordinates for drilling
<b>G01 Z-0.3937</b>	;feed onto workpiece
<b>G04 F0.020</b>	;dwell at bottom of hole
<b>G00 Z0.080</b>	;rapid out of hole
<b>G00 G40 Z2.0</b>	;rapid to safe height
<b>X0.0 Y8.0 Z8.0</b>	;safe tool change position
<b>M30</b>	;end of program

This is the same program typed into the control:



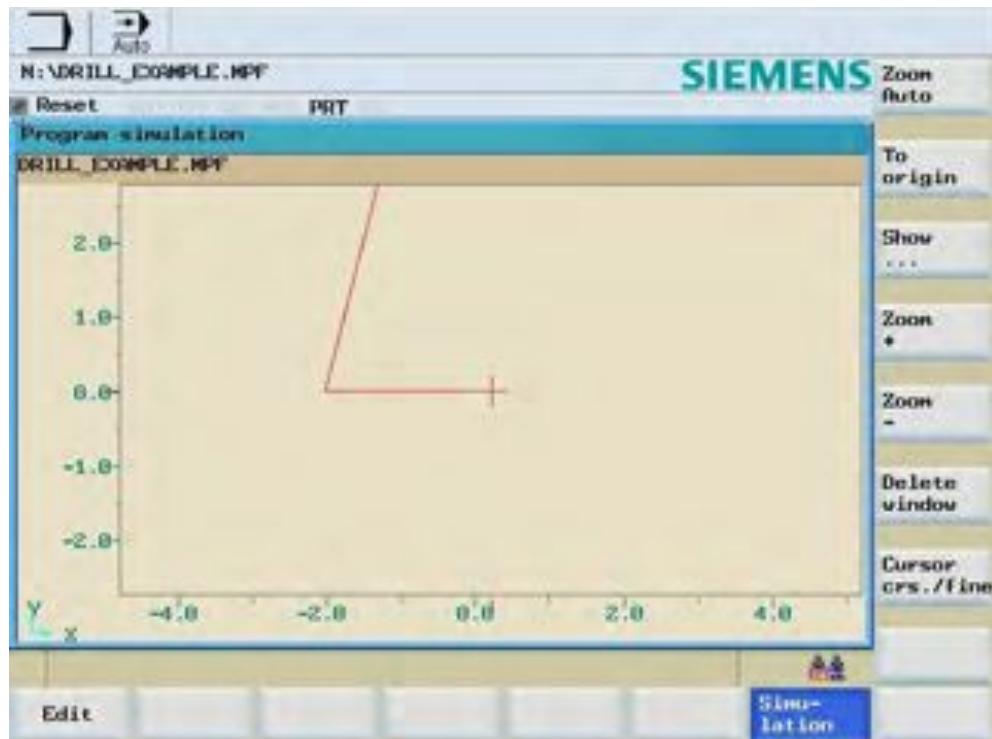
## Section 4

### Example drilling program

Notes

The same program run in simulation:

Simu-  
lation



Edit

# 1 Brief description

**Module objective:**

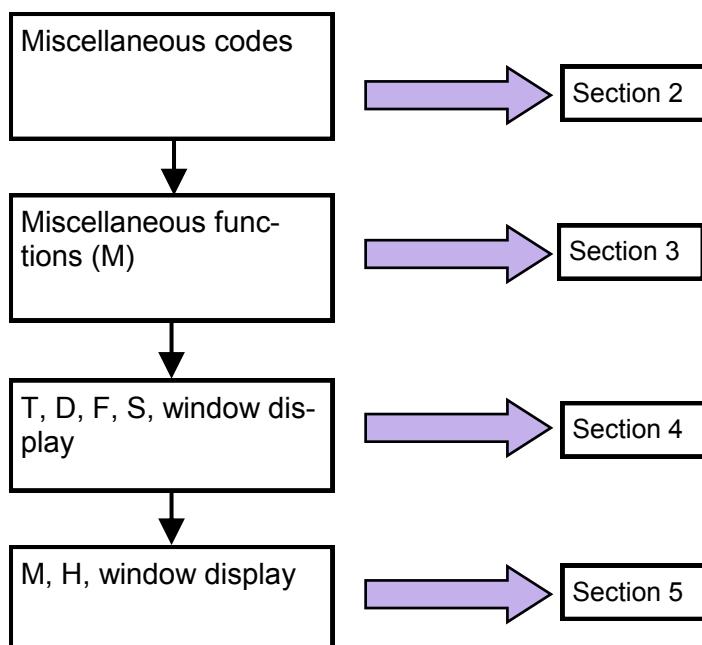
Upon completion of this module you will understand miscellaneous codes.

**Module description:**

We use miscellaneous codes as a simple switch, to instruct a machine to switch ON/OFF various function via the customers NC program.

**Module content:**

Miscellaneous codes  
Miscellaneous functions (M)  
T, D, F, S, window display  
M, H, window display



## Section 2

### Miscellaneous codes

Notes

#### Tool T

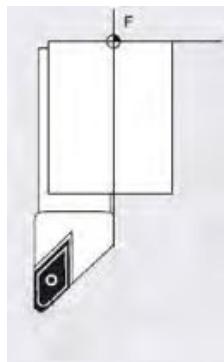
##### Function

The tool selection takes place when the T word is programmed.

Tool change is performed directly using T word.

##### Programming

T1 D1 ; tool call for "turning"



#### Tool offset number D

##### Function

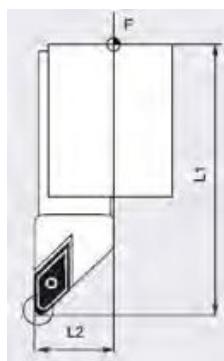
It is possible to assign 1 to 9 data fields with different tool offset blocks to a specific tool.

If no D word is written, D1 is automatically in effect.

If D0 is programmed, the offset for the tool is inactive.

##### Programming

D... ; tool offset number: 1 ... 9  
D0 ; no offset active



## Section 2

### Miscellaneous codes

Notes

#### Path Velocity F

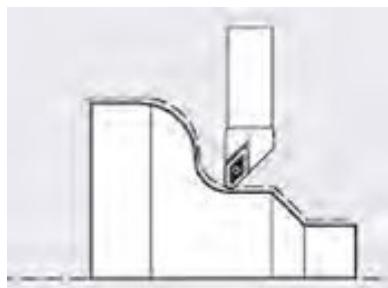
##### Function

The path velocity is determined by the programmed F word. All axes can be traversed simultaneously.

##### Programming

G01 G95 X100 F0.1 S500 ; feed rate = 0.1mm/rev

G01 G96 X100 F0.1 S100 ; feed rate = 0.1mm/rev



#### Spindle speed S

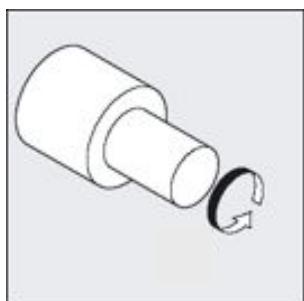
##### Function

The spindle speed is programmed in r.p.m. under the address S, provided that there is a controlled spindle on the machine.

##### Programming

S1000 M03 ; Spindle accelerates Clock-wise to 1000rpm

....  
S500 ; speed change



## Section 3

### Miscellaneous functions

Notes

The miscellaneous functions M initiates switching operations, such as "coolant ON/OFF" and other functions.

Various M functions have already been assigned to the CNC control, M functions not already assigned to the CNC control are reserved for free assignment by the machine tool manufacturer.

#### Program stop (M00, M01)

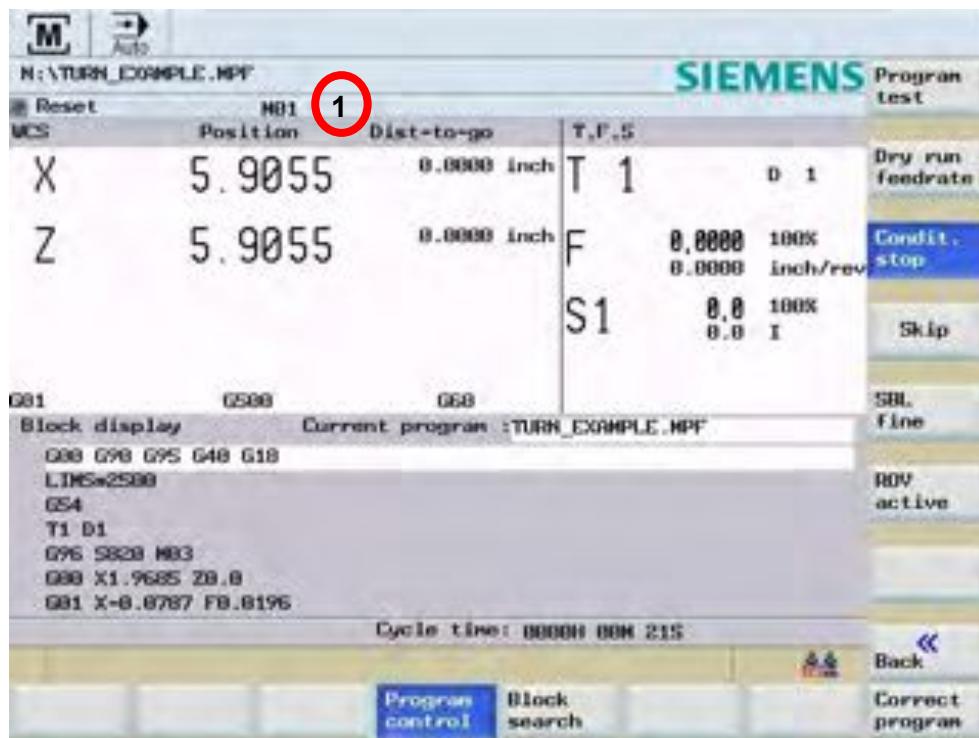
##### Function

When placed between axis movements, this command will stop the NC program until the cycle start button is pressed again.

##### Programming

M00 ; unconditional stop (you must press cycle start)  
M01 ; conditional stop  
(softkey in Auto page "program control")

M01 is conditional, can be activated or deactivated in the "AUTO" page, By following this sequence.



1

In the STATUS area the control will show you that M01 has been activated.

## Section 3

### Miscellaneous functions

#### End of program, M02, M17, M30

##### Function

A program is terminated with M2, M17 or M30 and reset to the beginning of the program. M02, M17 are usually used to return to the main program from a program call.

##### Programming

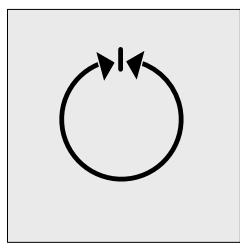
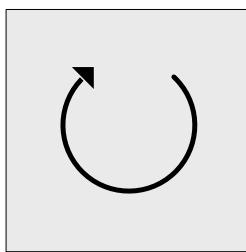
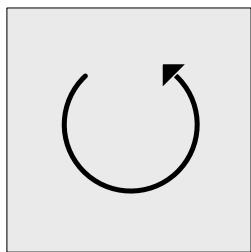
M02	; program stop and reset to beginning
M17	; program stop and reset to beginning
M30	; program stop and reset to beginning

#### Direction of spindle rotation (M03 - M04 - M05)

##### Functions

To specify the direction of rotation of the spindle or stop the rotation.

##### Programming



M03 ; clockwise direction

M04 ; counter clockwise direction

M05 ; stop the spindle

#### Tool change (M06)

##### Function

M06 is usually used on a milling machine to call up and run a cycle, this being written by the machine tool builder. The cycle will change the tool from the spindle into the magazine either by moving the magazine to the spindle or using an arm of some sort.

##### Programming

T01	
M06	tool change for a milling machine



## Section 3

### Miscellaneous functions

#### Coolant ON/OFF (M08 - M09)

##### Function

To switch ON and OFF of coolant system on machine.

##### Programming

M08 ; switch on of coolant system  
M09 ; switch off of coolant system

Notes



## Section 4

### T, D, F, S, window display

Notes

To find out which miscellaneous codes are active (T, D, F, S,)

Follow this sequence.



The screenshot shows the Siemens SINUMERIK 802D sl control interface. In the top right corner, there is a 'Function' menu. Below it, a 'T, D, F, S' window is displayed, showing the following data:

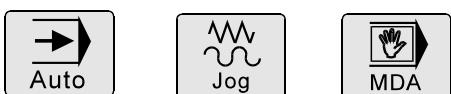
MCS	Position	Dist-to-go	T, D, F, S
X	5.9055	0.0000 inch	T 1 D 1
Z	5.9055	0.0000 inch	F 2 0.0000 100% 0.0000 inch/rev
			S1 0.0 100% 0.0 I

Below the window, the status bar displays: G91 G90 G95 G48 G18 LIMSw25000 G54 T1 D1 G96 S828 M83 G98 X1.9685 Z0.0 G91 X-0.8787 F0.8196. The cycle time is shown as 00000 000 215. At the bottom, there are buttons for Program control, Block search, and Correct program.

2

This screen will show you all active T, D, F, S, values that are being used in the NC program at that point in time.

This “T,D,F,S,” window is available in all these three modes:



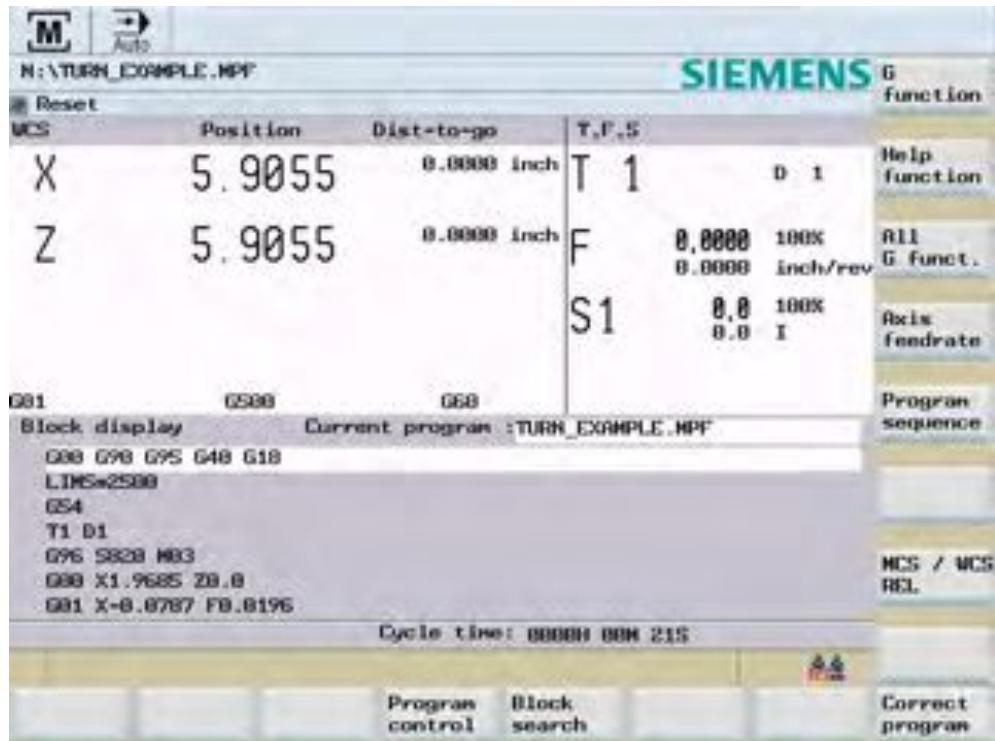
## Section 5

### M and H window display

Notes

To find out which miscellaneous functions are active (M and H)

Follow this sequence.

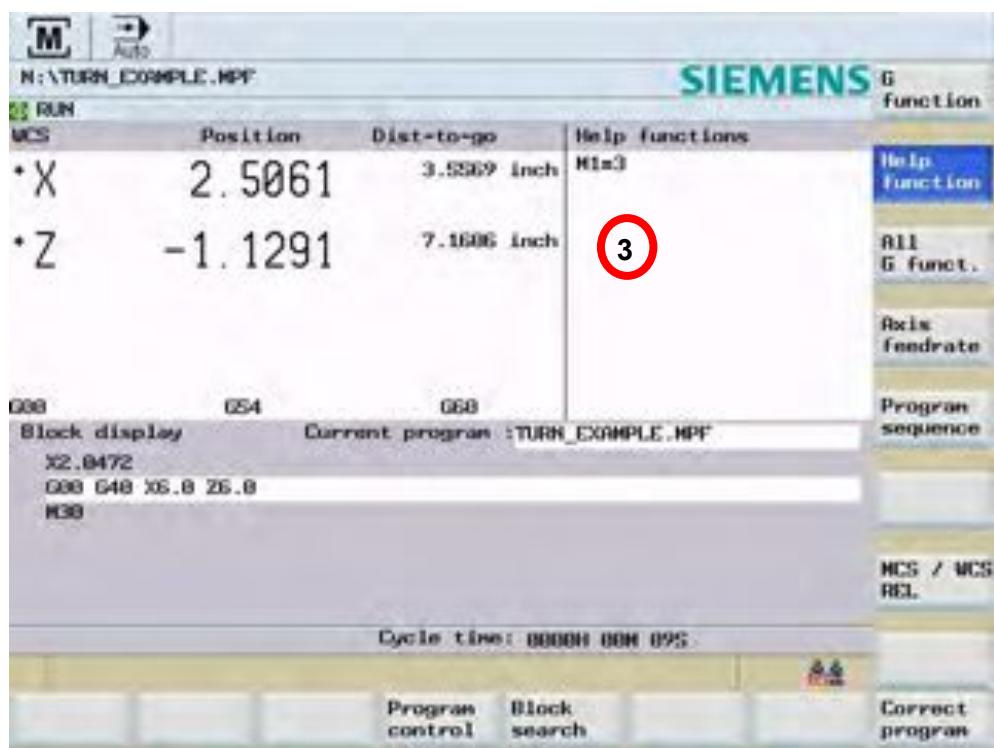


Help  
function

## Section 5

### M and H window display

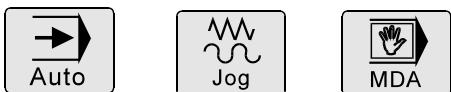
Notes



3

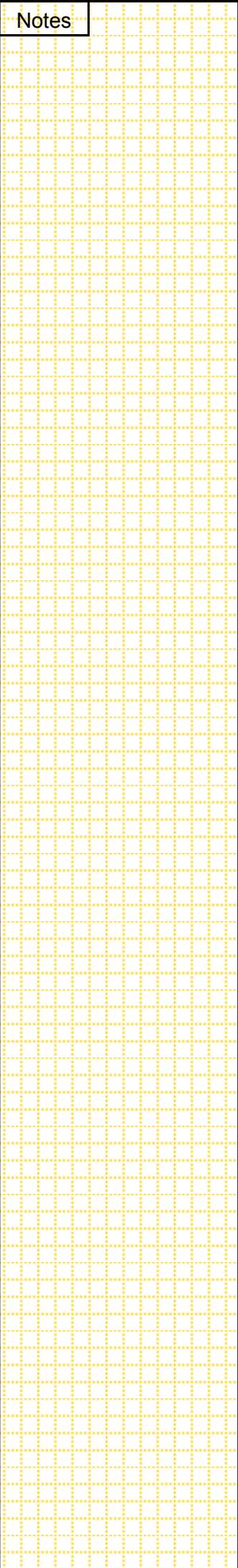
This screen will show you all active M and H values that are being used in the NC program at that point in time.

This window is available in all these three modes:



---

Notes



## 1 Brief description

**Module objective:**

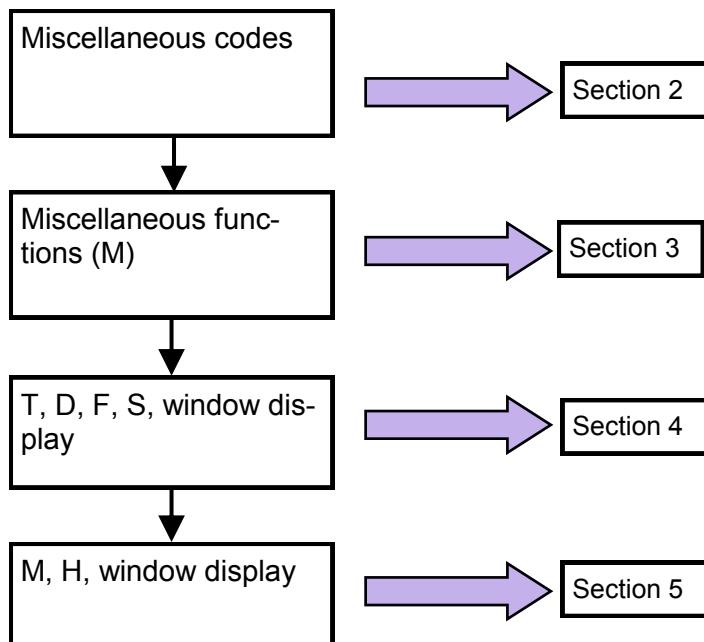
Upon completion of this module you will understand miscellaneous codes.

**Module description:**

We use miscellaneous codes as a simple switch, to instruct a machine to switch ON/OFF various function via the customers NC program.

**Module content:**

Miscellaneous codes  
Miscellaneous functions (M)  
T, D, F, S, window display  
M, H, window display



## Section 2

### Miscellaneous codes

Notes

#### Tool T

##### Function

The tool selection takes place when the T word is programmed.

However the change of tool depends:

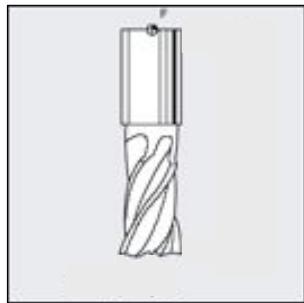
Tool change is performed either directly using T word for “turning”.

Or tool change is performed with the T word with an additional instruction M6 (see section 3 “miscellaneous functions) for milling

##### Programming

T1 D1 ; tool call for “turning”

T1  
M6 ; tool call for “milling”



#### Tool offset number D

##### Function

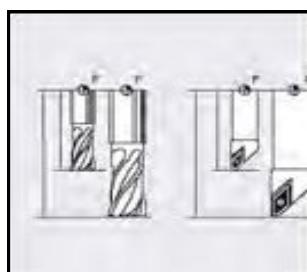
It is possible to assign 1 to 9 data fields with different tool offset blocks to a specific tool.

If no D word is written, D1 is automatically in effect.

If D0 is programmed, the offset for the tool is inactive.

##### Programming

D... ; tool offset number: 1 ... 9  
D0 ; no offset active



## Section 2

### Miscellaneous codes

Notes

#### Path Velocity F

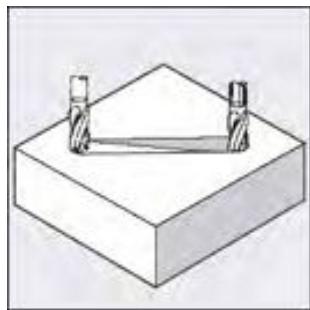
##### Function

The path velocity is determined by the programmed F word. All axes can be traversed simultaneously.

##### Programming

G01 G94 X100 F500 ; feed rate = 500mm/min

G01 G95 X100 F0.1 S500 ; feed rate = 0.1mm/rev



#### Spindle speed S

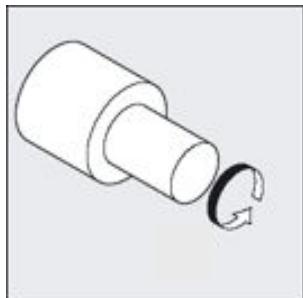
##### Function

The spindle speed is programmed in r.p.m. under the address S, provided that there is a controlled spindle on the machine.

##### Programming

S1000 M03 ; Spindle accelerates Clock-wise to 1000rpm

....  
S500 ; speed change



## Section 3

### Miscellaneous functions

Notes

The miscellaneous functions M initiates switching operations, such as "coolant ON/OFF" and other functions.

Various M functions have already been assigned to the CNC control, M functions not already assigned to the CNC control are reserved for free assignment by the machine tool manufacturer.

#### Program stop (M00, M01)

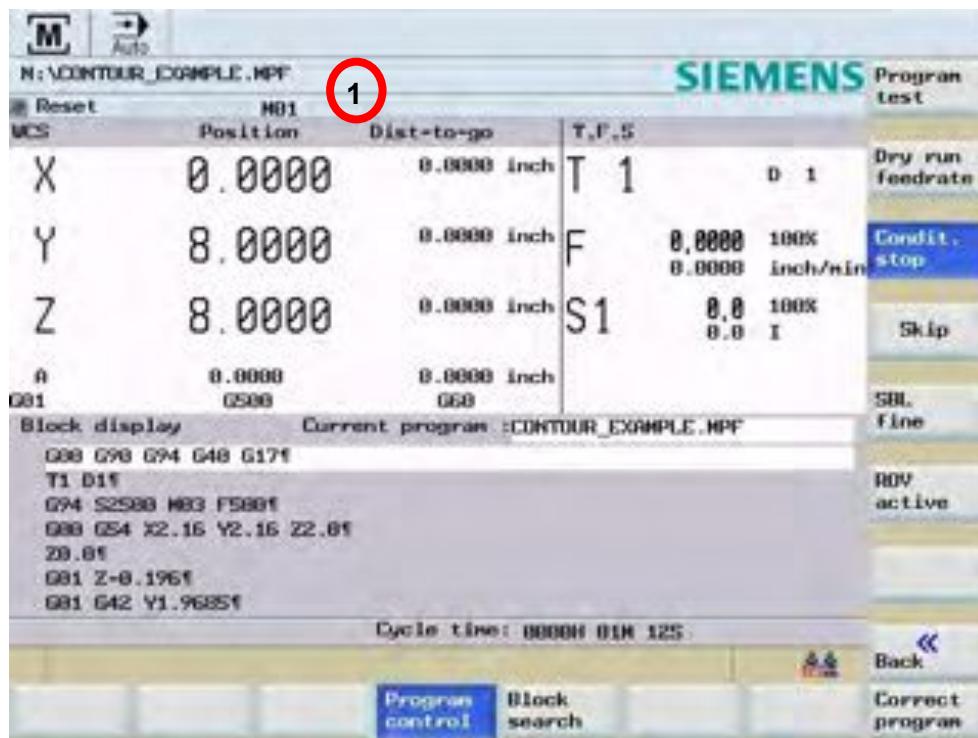
##### Function

When placed between axis movements, this command will stop the NC program until the cycle start button is pressed again.

##### Programming

M00 ; unconditional stop (you must press cycle start)  
M01 ; conditional stop  
(softkey in Auto page "program control")

M01 is conditional, can be activated or deactivated in the "AUTO" page, By following this sequence.



1

In the STATUS area the control will show you that M01 has been activated.

## Section 3

### Miscellaneous functions

#### End of program, M02, M17, M30

Notes

##### Function

A program is terminated with M2, M17 or M30 and reset to the beginning of the program. M02, M17 are usually used to return to the main program from a program call.

##### Programming

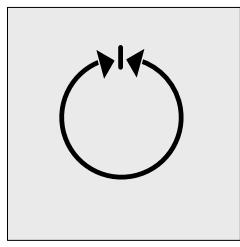
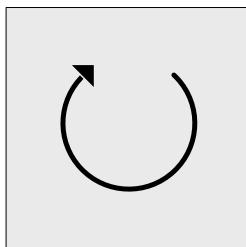
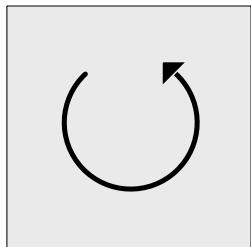
M02 ; program stop and reset to beginning  
M17 ; program stop and reset to beginning  
M30 ; program stop and reset to beginning

#### Direction of spindle rotation (M03 - M04 - M05)

##### Functions

To specify the direction of rotation of the spindle or stop the rotation.

##### Programming



M03 ; clockwise direction

M04 ; counter clockwise direction

M05 ; stop the spindle

#### Tool change (M06)

##### Function

M06 is usually used on a milling machine to call up and run a cycle, this being written by the machine tool builder. The macro will change the tool from the spindle into the magazine either by moving the magazine to the spindle or using an arm of some sort.

##### Programming

T01  
M06 tool change for a milling machine



## Section 3

### Miscellaneous functions

Notes

#### Coolant ON/OFF (M08 - M09)

##### Function

To switch ON and OFF of coolant system on machine.

##### Programming

M08	; switch on of coolant system
M09	; switch off of coolant system



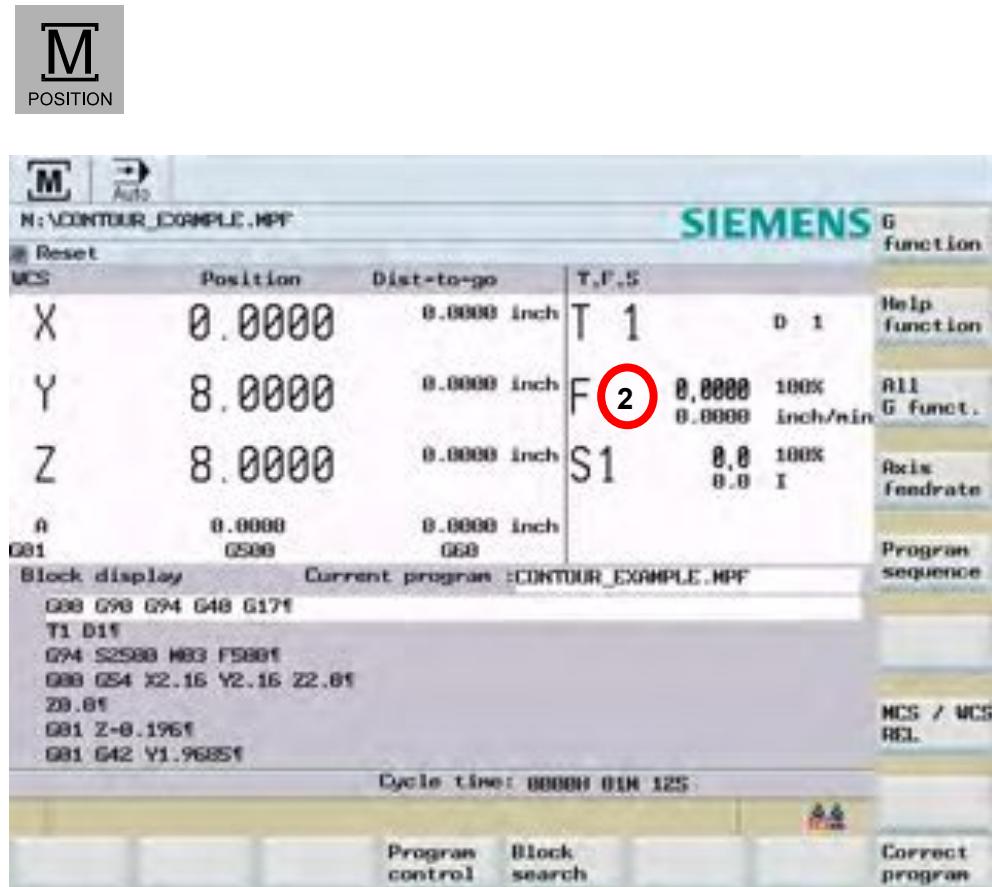
## Section 4

### T, D, F, S, window display

Notes

To find out which miscellaneous codes are active (T, D, F, S,)

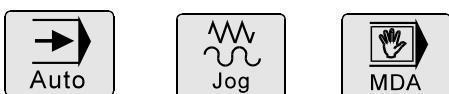
Follow this sequence.



2

This screen will show you all active T, D, F, S, values that are being used in the NC program at that point in time.

This “T,D,F,S,” window is available in all these three modes:



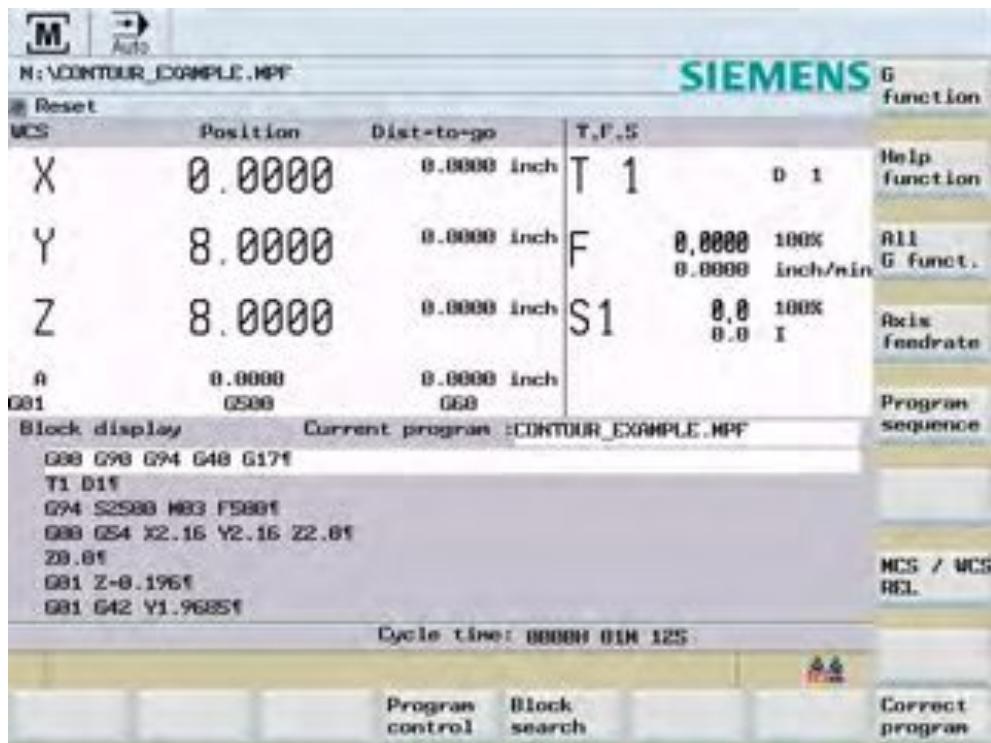
## Section 5

### M and H window display

Notes

To find out which miscellaneous functions are active (M and H)

Follow this sequence.

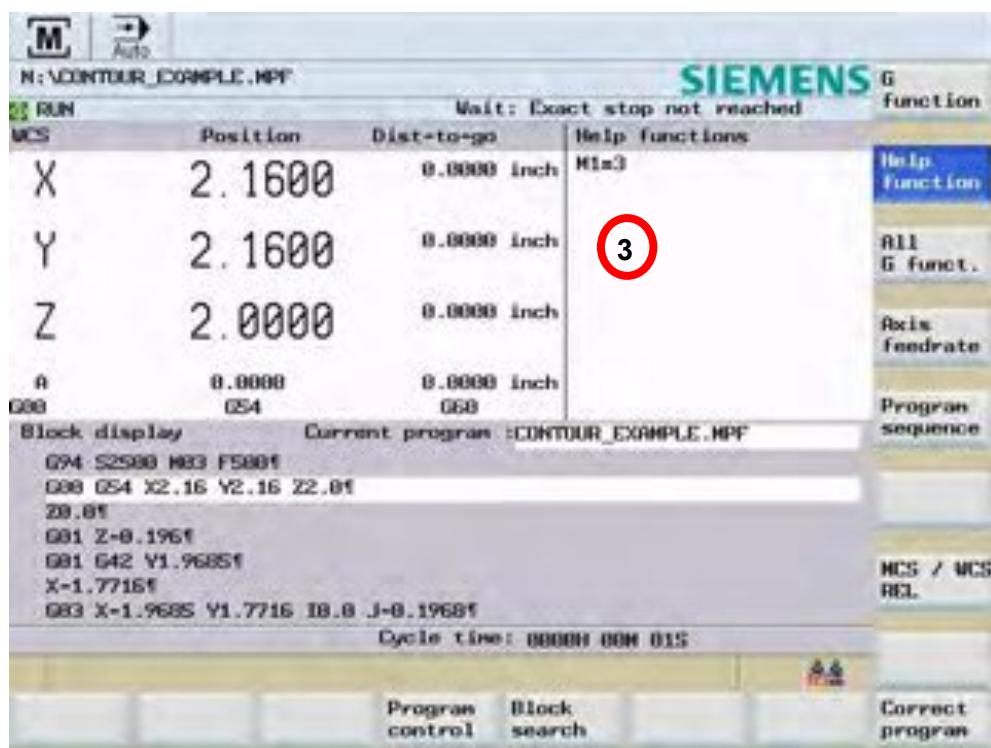


Help function

## Section 5

### M and H window display

Notes



3

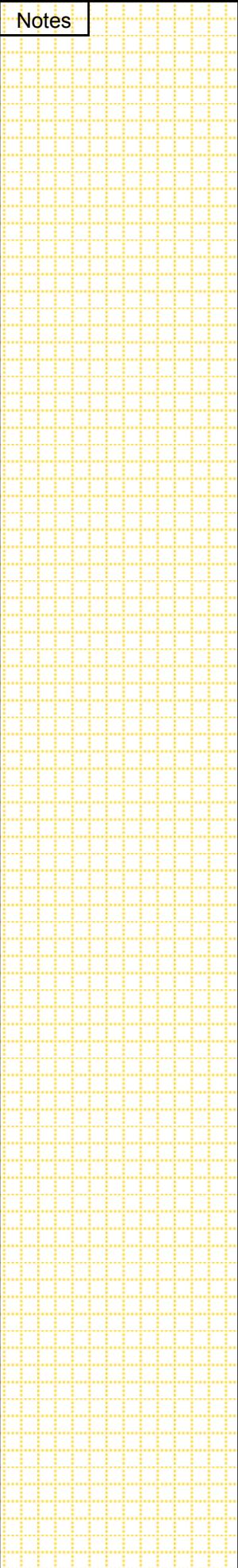
This screen will show you all active M and H values that are being used in the NC program at that point in time.

This window is available in all these three modes:



---

Notes



## 1 Brief description

**Module objective:**

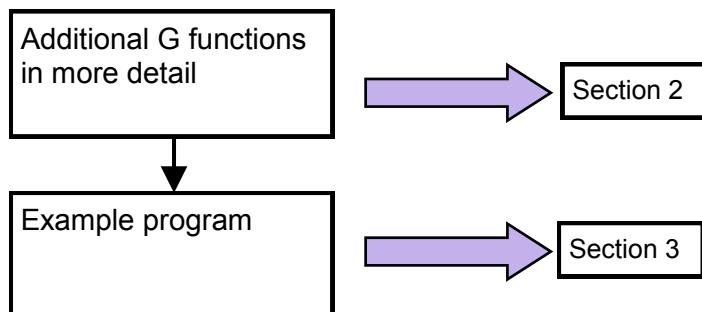
Upon completion of this module you will understand commonly used additional used G functions (G-Codes) in detail.

**Module description:**

We use G functions to instruct a machine what to do, but a structure should be kept to.

**Module content:**

Additional G functions in more detail  
Example program



## Section 2

### Additional G functions in more detail

Notes

#### Radius/diameter dimensional notation: DIAMOF, DIAMON

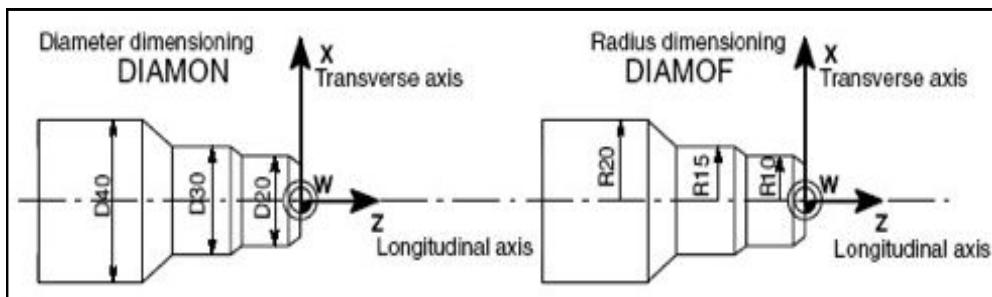
##### Function

For the machining of parts on **turning machines**, it is typical to program geometry data as diameter dimensions. When necessary, it is possible to switch to radius geometry in the program.

##### Programming

DIAMOF ; radius dimensioning  
DIAMON ; diameter dimensioning

This function is typical used on TURN/MILL machines



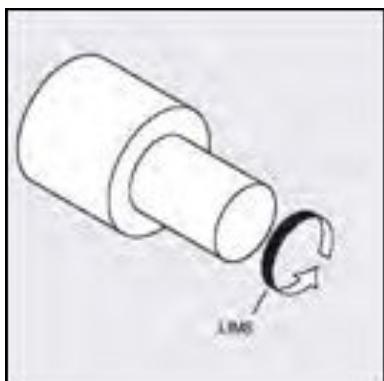
#### Upper speed limit for G96: LIMS

##### Function

When G96 is used in a program, the spindle speed will vary according to the diameter being cut. It is usual to cap the maximum spindle speed within the program for various reasons e.g. bar feed work or extremely small diameter component.

##### Programming

LIMS= ;upper speed limit for G96



## Section 2

### Additional G functions in more detail

Notes

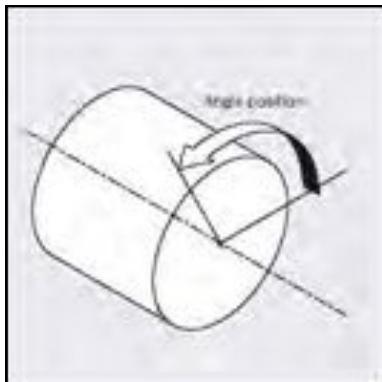
#### Spindle positioning: SPOS

##### Function

With the function SPOS= you can position the spindle in a specific angular position. The spindle is held in the position by position control.

##### Programming

SPOS= ; absolute dimension position  
SPOS= IC(..) ; incremental dimension position



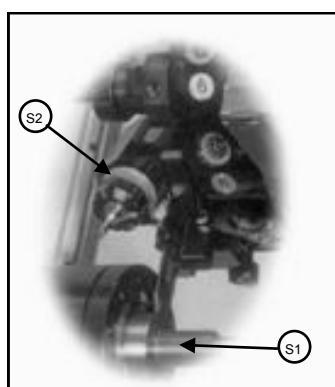
#### Master spindle: SETMS

##### Function

The master spindle is defined by machine data, and generally is the main spindle. A different spindle can be defined as master spindle in the program (driven tooling).

##### Programming

SETMS ; configured spindle is now the master spindle  
SETMS(n) ; spindle (n) is now master spindle



## Section 2

Notes

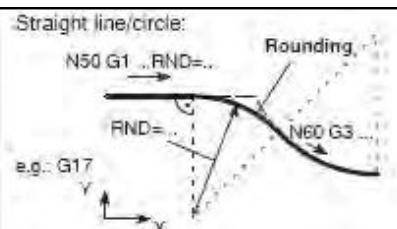
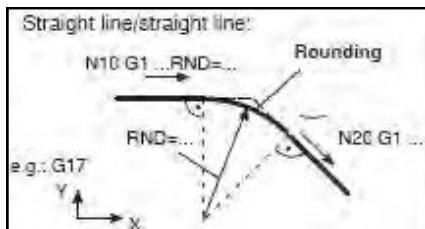
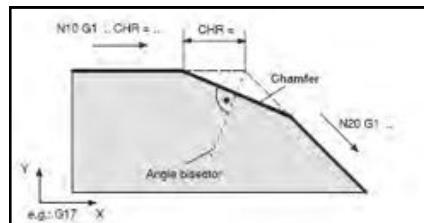
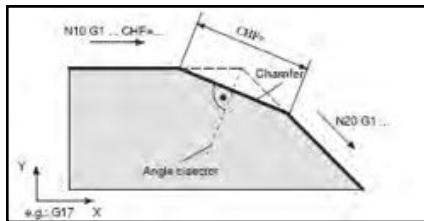
### Rounding, chamfer - RND, CHF, CHR

#### Function

You can insert a chamfer (CHF or CHR) or rounding (RND) elements into a contour intersection.

#### Programming

CHF= ; insert chamfer, value is length of chamfer  
CHR= ; insert chamfer, value is side length of chamfer  
RND= ; insert rounding, value is radius of rounding



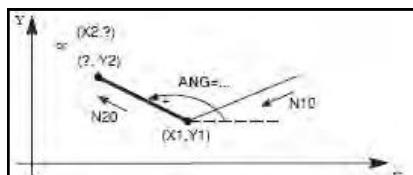
### Angle - ANG

#### Function

If only one of the end points is known for a straight line, and an angular dimension is given on the drawing, this can be used for the straight line path.

#### Programming

ANG= ; angle value for defining a straight line



## Section 2

### Unconditional program jumps

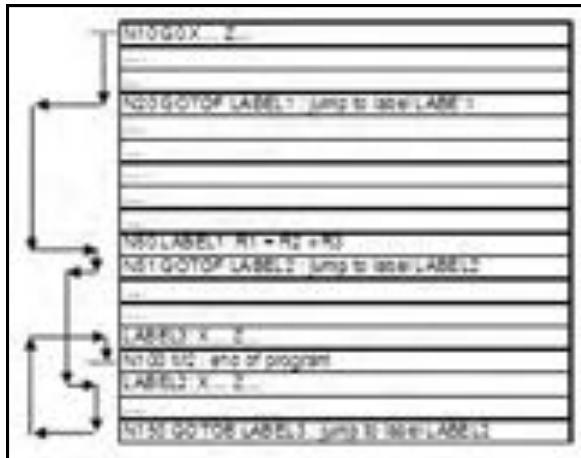
#### Function

NC programs process their blocks in the sequence in which they were written.

The jump destination can be a block with a label or with a block number. The block must be located within the program.

#### Program

GOTOF label ; jump forward to label  
GOTOB label ; jump backwards to label



### Conditional program jumps

#### Function

Jump conditions are formulated after the if instruction. If the jump condition (value not zero) is satisfied, the jump takes place.

The jump destination can be a block with a label or with a block number. The block must be located within the program.

#### Program

IF condition GOTOF label ; jump forward to label  
IF condition GOTOB label ; jump backwards to label

```
N10 IF R1==1 GOTOF LABEL1
...
N90 LABEL1:
N100 IF R1>1 GOTOF LABEL2
...
N150 LABEL2:
...
N800 LABEL3:
...
N1000 IF R45==R7+1 GOTOB LABEL3
```

Notes

#### Milling on Turning Machines

##### Milling on front face - TRANSMIT

This function is only available for SINUMERIK 802D sl plus and pro.

###### Function

This transformation function TRANSMIT allows convenient programming of milling/drilling on the front face of turned parts. A Cartesian coordinate system is used to program these machining operations.

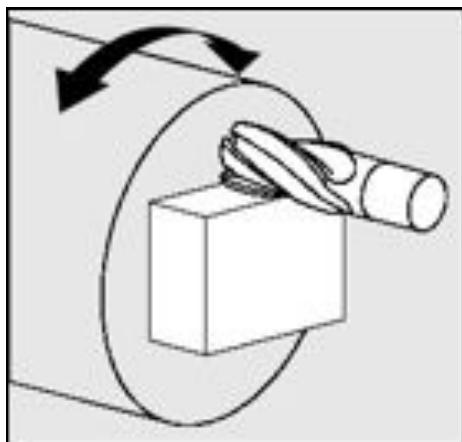
The control transforms the programmed traversing movements of the Cartesian coordinate system into traversing movements of the real machine axes. The main spindle functions here as the machine rotary axis.

In addition to the tool length offset, it is also possible to work with the tool radius offset (G41, G42).

The velocity control makes allowance for the limits defined for the rotations.

###### Programming

TRANSMIT	; activate TRANSMIT (separate block)
TRAFOOF	; deactivate (separate block)



## Section 2

### Additional G functions in more detail

Notes

#### Milling on peripheral surface - TRACYL

This function is only available for SINUMERIK 802D sl plus and pro.

##### Function

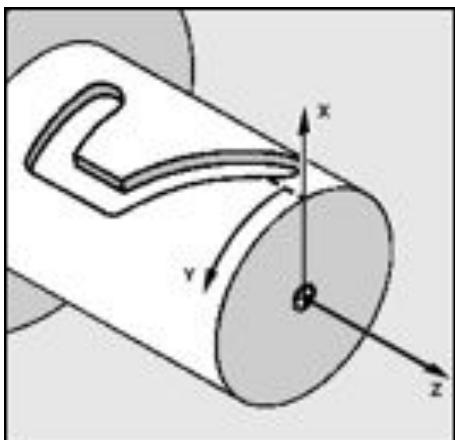
This transformation function TRACYL allows convenient programming for milling on the peripheral surface of turned parts.

The path of the grooves is programmed in the plane peripheral surface, which was logically developed for a specific machining cylinder diameter.

The control transforms the programmed traversing movements in the Cartesian coordinate X, Y, Z system into traversing movements of the real machine axes. The main spindle functions here as the machine rotary axis.

##### Programming

TRACYL	; activate TRACYL (separate block)
TRAFOOF	; deactivate (separate block)



## Section 2

### Additional G functions in more detail

This is a typical Turning program used on a turning machine, using most additional G functions.

```
TOP:  
G00 G90 G95 G40 G18 G70  
IF R1==2 GOTOF END  
LIMS=2500  
G54  
T1 D1  
SETMS  
G96 S990 M03 M08  
G00 X4.0157 Z0.0078  
G01 X-0.0787 F0.0078  
G00 Z0.0787  
X4.0157  
CYCLE95( "AAA:AAA_E", 0.13770, 0.00390, 0.00390, 0.00390, 0.01370,  
0.01390, 0.00390, 9, , ,0.01960)  
G00 G40 X6.0 Z6.0 M05  
T2 D1  
SETMS  
SPOS=0  
SETMS(2)  
G94 S800 F6 M03 M08  
DIAMOF  
TRANSMIT  
G00 G17 G54 X1.1160 Y0.0 Z0.3937  
MCALL CYCLE82( 0.39370, 0.00000, 0.07870, -0.39370, 0.00000,  
0.00390)  
HOLES2( 0.00000, 0.00000, 0.98420, 0.00000, 90.00000, 3)  
MCALL  
M05  
TRAFOOF  
DIAMON  
G00 G40 X6.0 Z6.0  
R1=R1+1  
GOTOB TOP  
END:  
M30  
*****CONTOUR*****  
AAA:  
X1.5748 Z0.0787  
Z0.0  
X3.5433 ANG=100  
Z-0.7874 RND=0.1968  
X3.9370  
Z-0.9842  
X4.0157  
M02  
AAA_E:***** CONTOUR ENDS *****
```

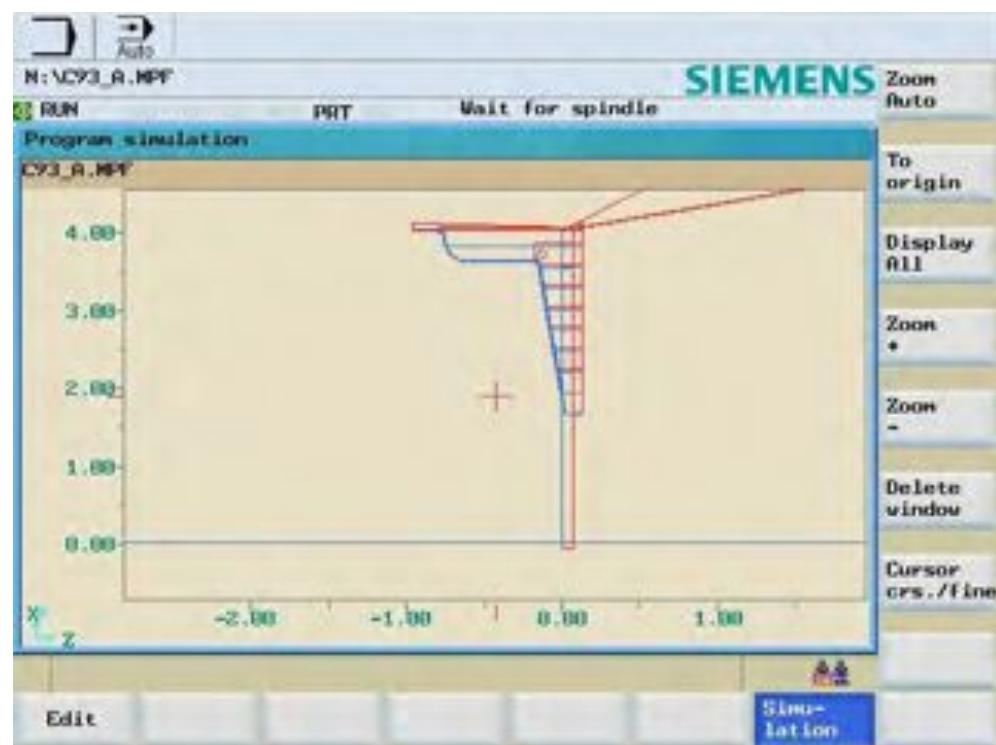
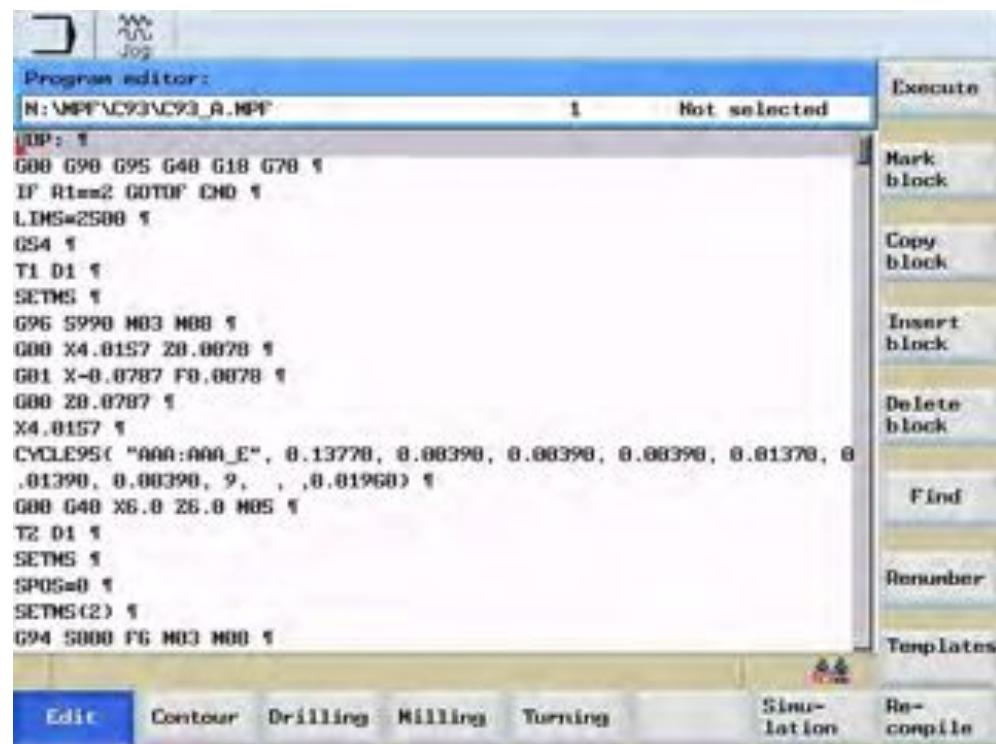
Notes

## Section 2

### Additional G functions in more detail

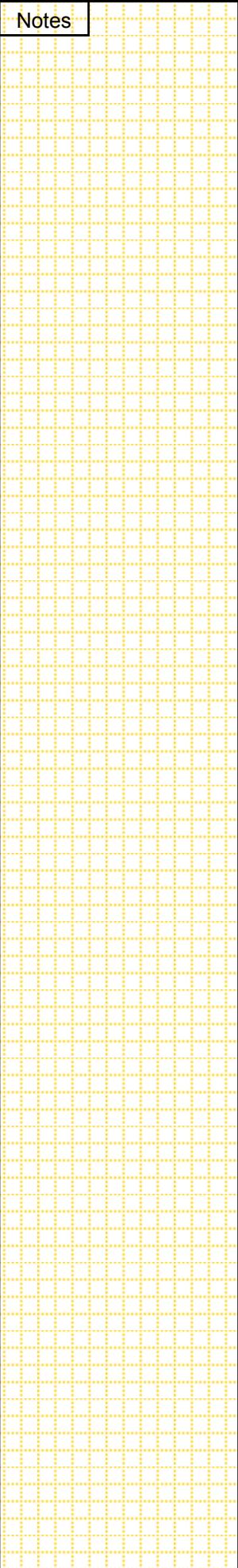
This shows the program in the “Program Editor” and then when the program is run in “Simulation”.

Notes



---

Notes



## 1 Brief description

**Module objective:**

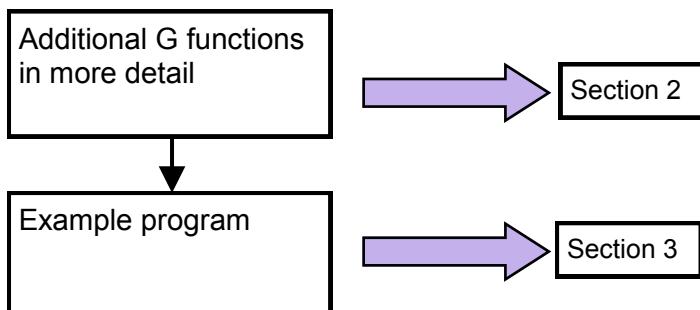
Upon completion of this module you will understand commonly used additional used G functions (G-Codes) in detail.

**Module description:**

We use G functions to instruct a machine what to do, but a structure should be kept to.

**Module content:**

Additional G functions in more detail  
Example program



## Section 2

Notes

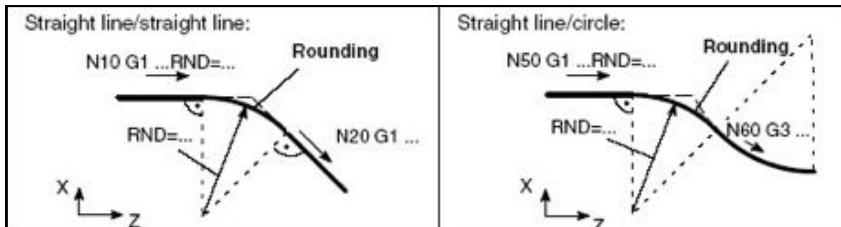
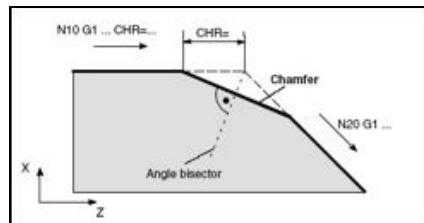
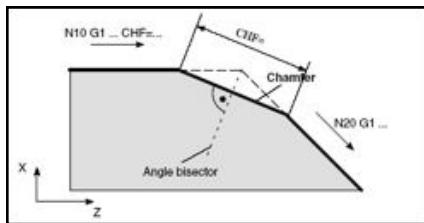
### Rounding, chamfer - RND, CHF, CHR

#### Function

You can insert a chamfer (CHF or CHR) or rounding (RND) elements into a contour intersection.

#### Programming

CHF= ; insert chamfer, value is length of chamfer  
CHR= ; insert chamfer, value is side length of chamfer  
RND= ; insert rounding, value is radius of rounding



### Angle - ANG

#### Function

If only one of the end points is known for a straight line, and an angular dimension is given on the drawing, this can be used for the straight line path.

#### Programming

ANG= ; angle value for defining a straight line



## Section 2

### Unconditional program jumps

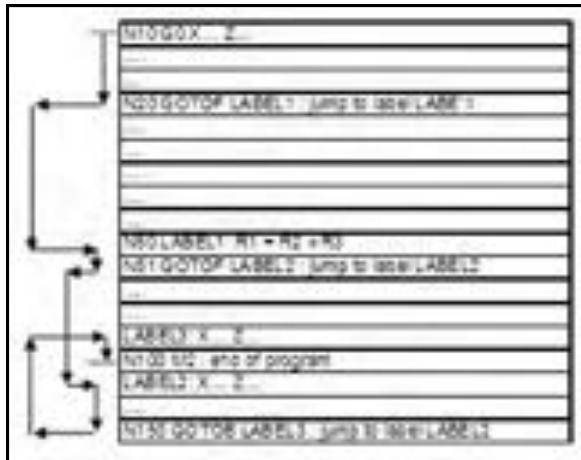
#### Function

NC programs process their blocks in the sequence in which they were written.

The jump destination can be a block with a label or with a block number. The block must be located within the program.

#### Program

GOTOF label ; jump forward to label  
GOTOB label ; jump backwards to label



### Conditional program jumps

#### Function

Jump conditions are formulated after the if instruction. If the jump condition (value not zero) is satisfied, the jump takes place.

The jump destination can be a block with a label or with a block number. The block must be located within the program.

#### Program

IF condition GOTOF label ; jump forward to label  
IF condition GOTOB label ; jump backwards to label

```
N10 IF R1==1 GOTOF LABEL1
...
N90 LABEL1:
N100 IF R1>1 GOTOF LABEL2
...
N150 LABEL2:
...
N800 LABEL3:
...
N1000 IF R45==R7+1 GOTOB LABEL3
```

Notes

#### Milling on peripheral surface - TRACYL

This function is only available for SINUMERIK 802D sl plus and pro.

##### Function

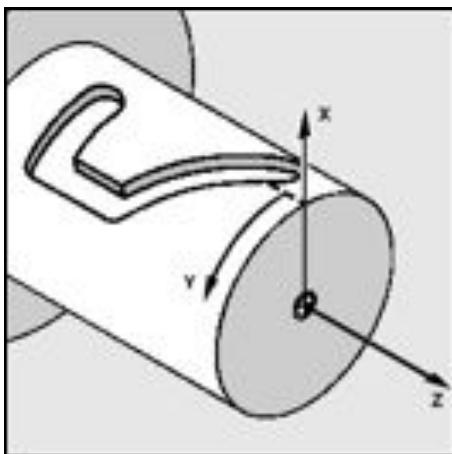
This transformation function TRACYL allows convenient programming for milling on the peripheral surface of cylindrical parts.

The path of the grooves is programmed in the plane peripheral surface, which was logically developed for a specific machining cylinder diameter.

The control transforms the programmed traversing movements in the Cartesian coordinate X, Y, Z system into traversing movements of the real machine axes. A rotary axis (rotary table) is required.

##### Programming

TRACYL ; activate TRACYL (separate block)  
TRAFOOF ; deactivate (separate block)



## Section 2

### Additional G functions in more detail

This is a typical Mill program used on a turning machine, using most additional G functions.

```
TOP:  
G00 G90 G95 G40 G17 G70  
IF R1==2 GOTOF END  
T1  
M6  
G94 S424 F10.5 M03 M08  
G90 G54 G0 X-1.3779 Y2.3622  
G0 Z0.0800  
G1 Z0  
X4.0  
Y0.8  
X-1.3779  
G0 Y-1.3779  
G1 Z-0.3937 F40  
G41 X0.1968  
Y2.7559 RND=0.3937 F10.5  
X2.8346 Y2.7559  
X3.6220 Y2.2835  
Y0.3937 CHF=0.0800  
X0.1771  
G40 Y-0.8267  
G0 Z0.0800  
Z4.0  
T2  
M6  
G94 S800 F6 M03 M08  
G00 G54 X1.0 Y0.0 Z2.0  
Z0.3937  
MCALL CYCLE82( 0.39370, 0.00000, 0.08000, -0.39370, 0.00000,  
0.00400)  
HOLES2( 0.00000, 0.00000, 1.00000, 0.00000, 90.00000, 3)  
MCALL  
G00 G40 X6.0 Y6.0 Z6.0  
R1=R1+1  
GOTOB TOP  
END:  
M30
```

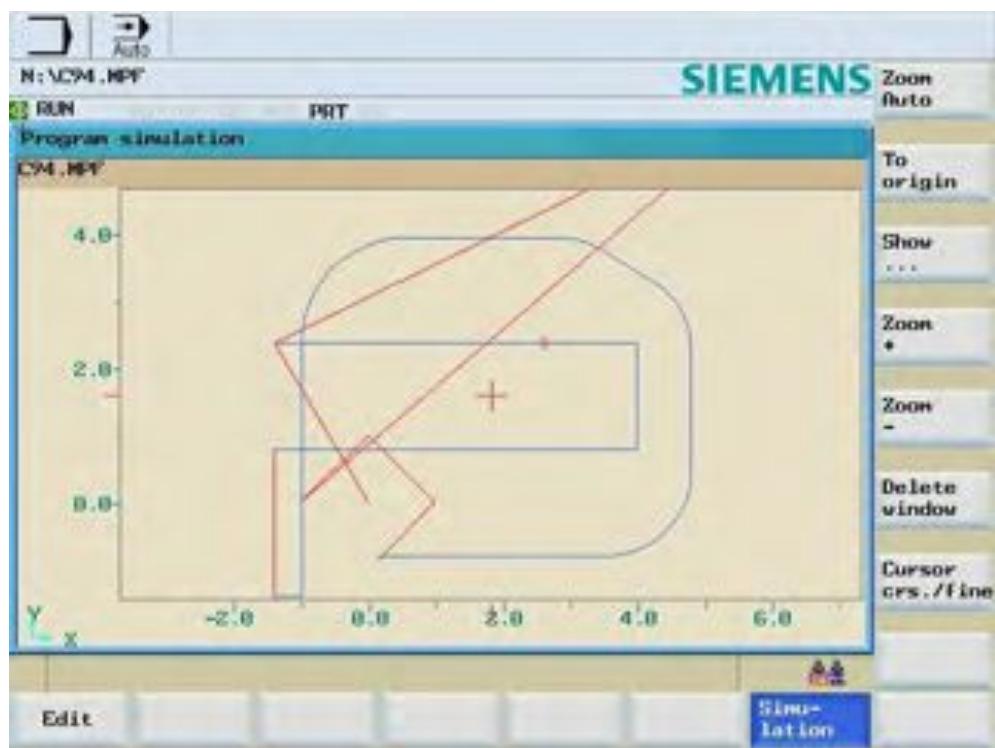
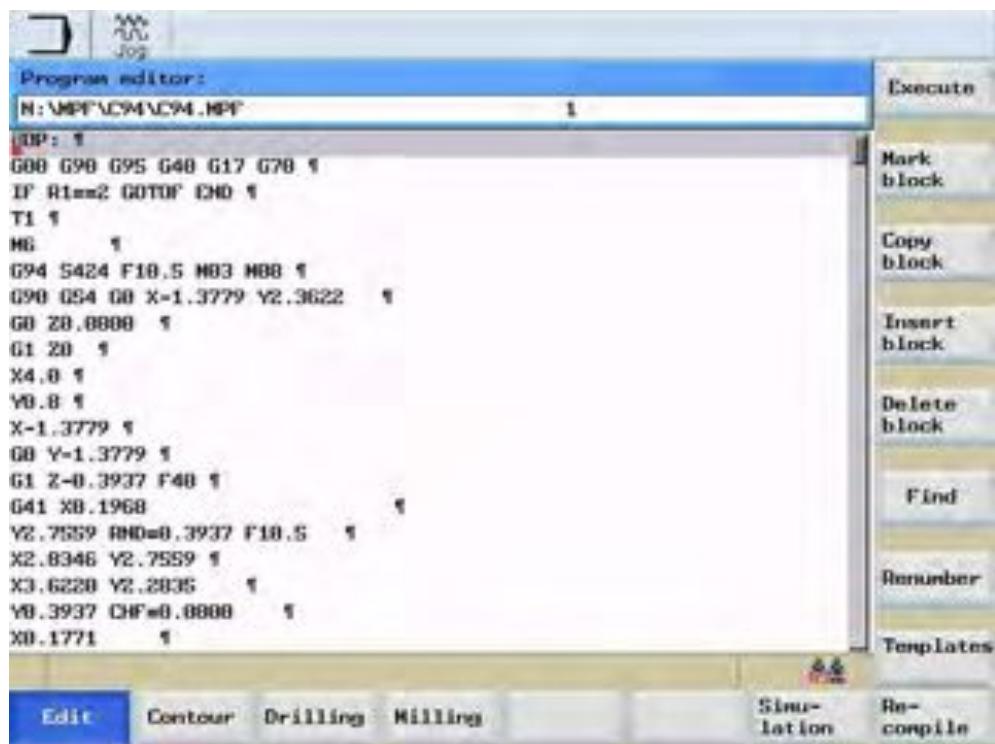
Notes

## Section 2

### Additional G functions in more detail

Notes

This shows the program in the "Program Editor" and then when the program is run in "Simulation".



## 1 Brief description

### Module objective:

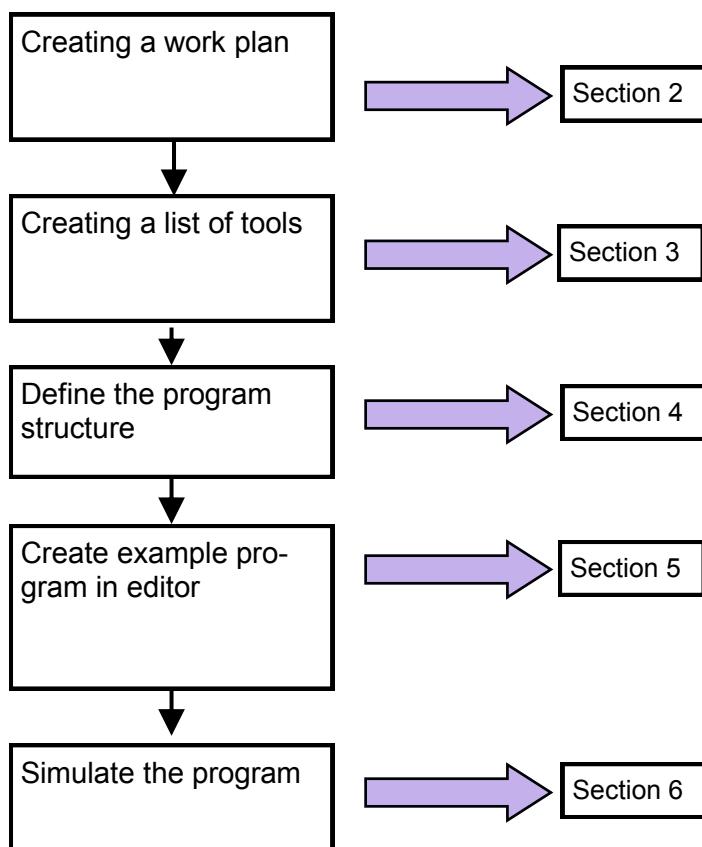
Upon completion of this module you can create a milling program.

### Module description:

This module describes how to write a part program for a milling machine, by example.

### Module content:

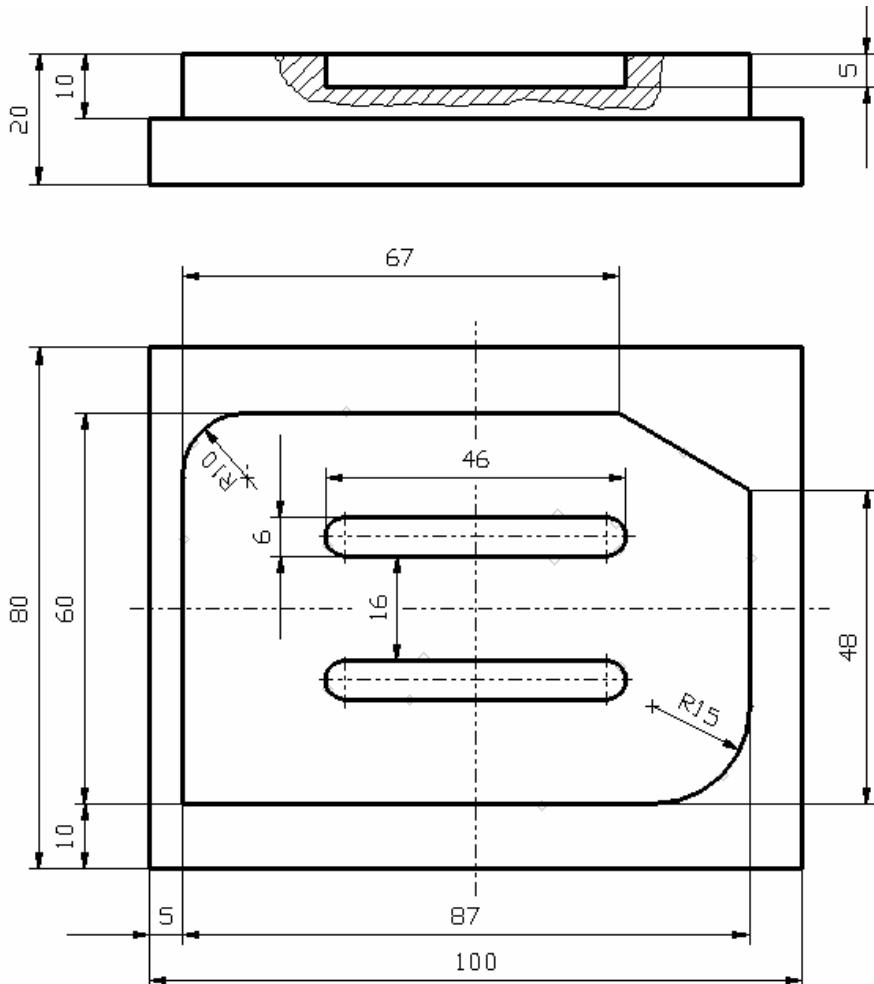
- Creating a work plan
- Creating a list of tools
- Define the program structure
- Create example program in editor
- Simulate the program



## Section 2

### Creating a work plan

An example program will be created using the following workpiece:



Notes

The work plan is used to define the steps which have to be carried out to machine the workpiece.

Working Step Nr.	Operation	Tools
1	Surface milling	End mill D2.30"
2	Contour milling	End mill D2.30"
3	Slot milling	Slot mill D0.236"

## Section 3

### Creating a list of tools

The tool list contains the technology information for each tool. This information will be used in the NC program, for the respective tool. The speeds and feeds have to be adapted dependant upon the workpiece material.

Notes

Tool / Operation	T-Nr.	Feed inch/min	Speed rpm	Spindle direction M3/M4	Coolant
End mill D2.36“ Surface milling	1	10.6“	424	M3	x
End mill D2.36“ Contour milling	1	10.6“	424	M3	x
Slot mill D0.23“ Slot milling	2	22.16“	4200	M3	x

## Section 4

### Define the program structure

Notes

The program sequence can be created from the work plan.

#### Tool 1: End mill D2.36"

- Change tool
- Select the origin (absolute/incremental)
- Activate zero offset
- Define the work plane
- Program spindle speed and direction
- Program feedrate
- Switch coolant on
- Surface milling
- Position on the outside contour
- Contour milling
- Retract tool
- Stop spindle and coolant

#### Tool 2: Slot mill D0.23"

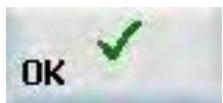
- Change tool
- Select the origin (absolute/incremental)
- Activate zero offset
- Define the working plane
- Program spindle speed and direction
- Program feedrate
- Switch coolant on
- Slot milling
- Retract tool
- Stop spindle and coolant
- Return to home position
- Program end

## Section 5

### Create example program in editor

Notes

A new program has to be generated in order to insert the ISO code.  
The program should be given the name PLATE1.  
First a new directory called PLATE1 can to be generated.  
Use the following sequence to complete this task.



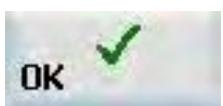
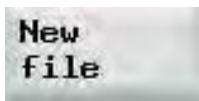
Curser over new directory name.



Or with



A new program has to be generated in order to insert the code.  
The program should be given the name PLATE1.



After generating a new program in the editor, the blocks for the surface milling, contour milling and the slot milling have to be typed in.

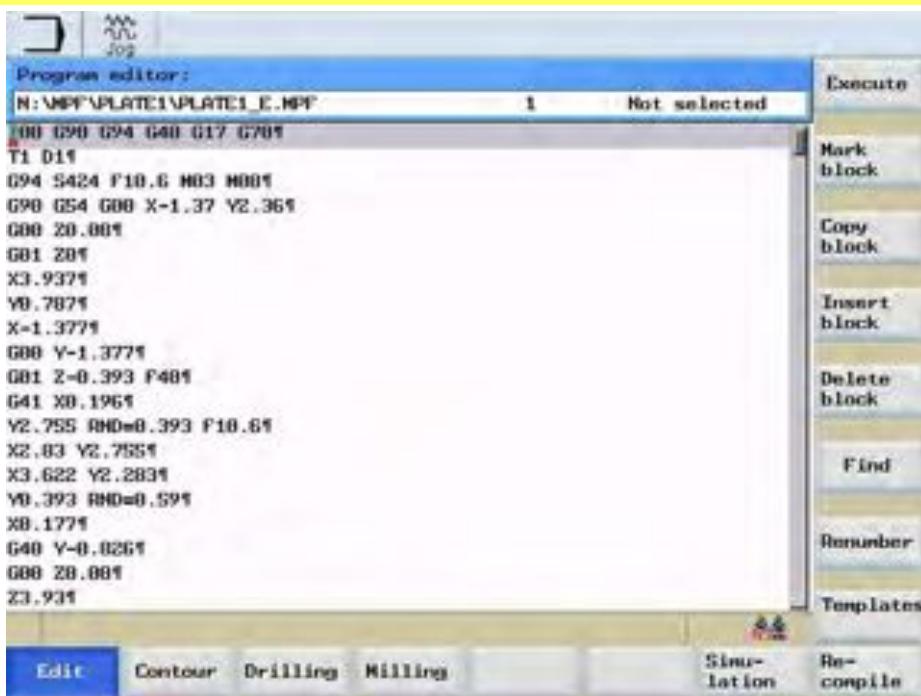
## Section 5

### Create example program in editor

Notes

This is an example of the NC blocks that will machine the component.

```
G00 G90 G94 G40 G17 G70
T1 D1 ;End Mill D2.3"
G94 S424 F10.6 M03 M08
G90 G54 G0 X-1.37 Y2.36 ; Basic settings, Start point
G0 Z0.08
G1 Z0
X3.937
Y0.787
X-1.377
G0 Y-1.377
G1 Z-0.393 F40
G41 X0.196 ; Start the radius compensation
Y2.755 RND=0.393 F10.6 ; Corner above left
X2.83 Y2.755
X3.622 Y2.283 ; Angle contour
Y0.393 RND=0.59 ; Corner below right
X0.177 ; End point 0.02" further
G40 Y-0.826 ; Cancel radius compensation
G0 Z0.08
Z3.93
T2 D1 ; Slot Mill D0.23"
G94 S4200 F6 M04 M08
G90 G54 G17 G0 X1.181 Y1.141
G0 Z0.08
G1 Z-0.196 ; Infeed into slot
X2.755 F22.16
G0 Z0.08 ; Bottom slot is finished
X2.755 Y2.401 ; Start point top slot
G1 Z-0.196 F6 ; Infeed top slot
X1.181 F22.16
G0 Z0.08 ; Top slot finished
Z4 M5 M9 ; Retract tool
X-8 Y6 ; Home position
M30 ; Program end
```

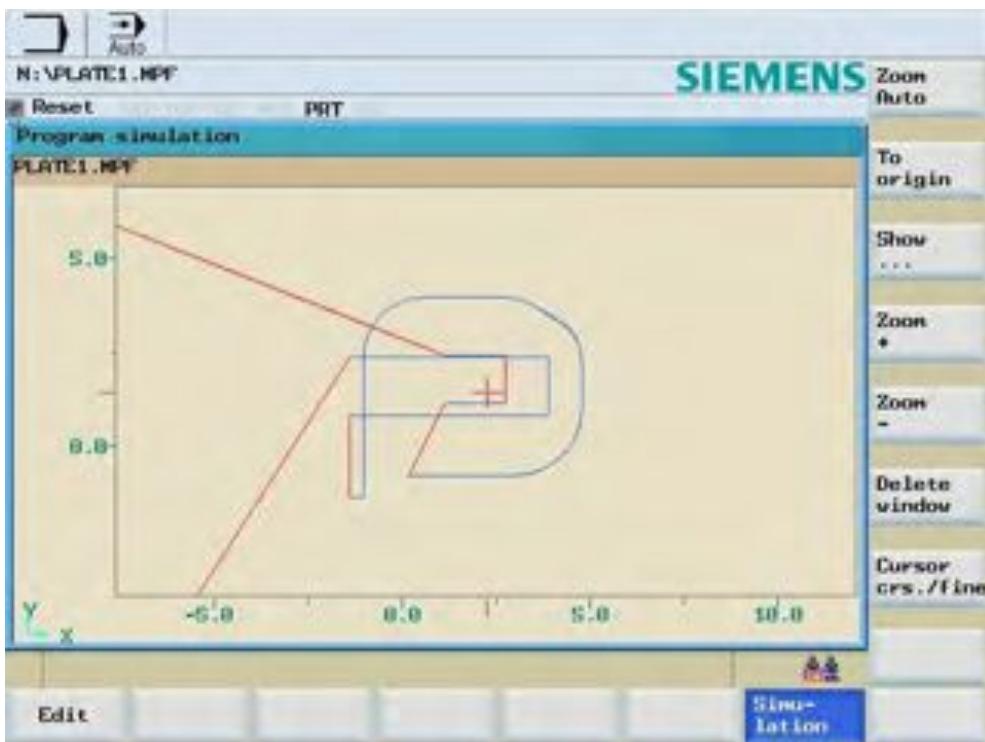
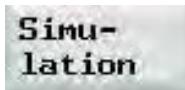


## Section 6

### Simulate the program

The program can be tested in the Simulation mode.  
The necessary tools have to exist in the tool list in order to simulate the program.

This is the sequence to test the NC program in Simulation:



Once you have simulated the NC program you can use the EDIT softkey to go back to the EDITOR and RESET button to eliminate any alarms.

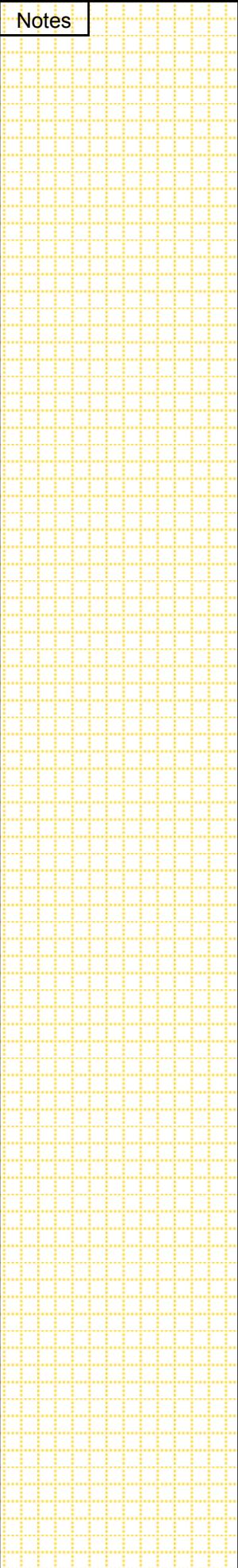
Edit



Notes

---

Notes



## 1 Brief description

**Module objective:**

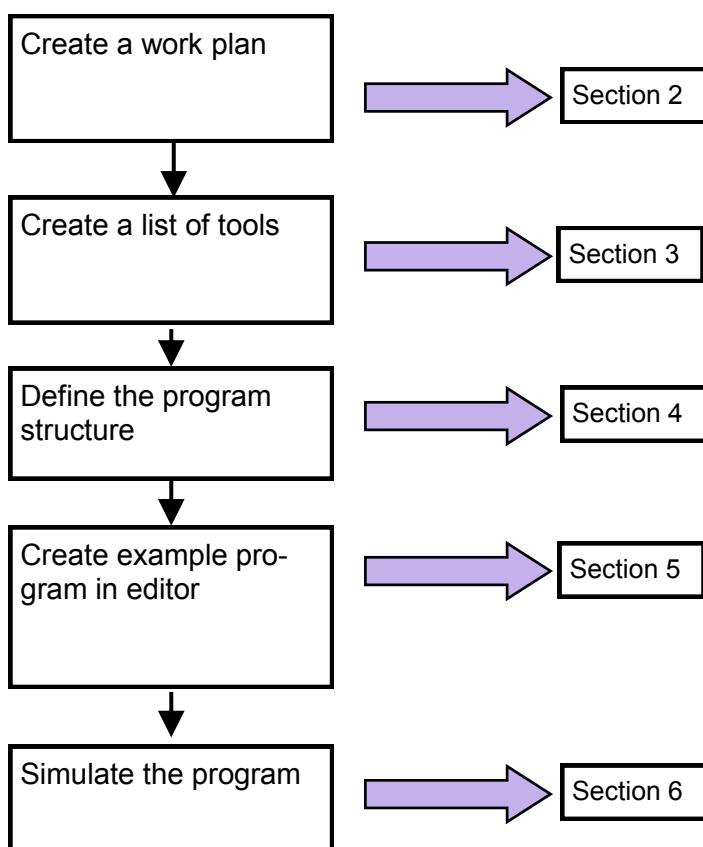
Upon completion of this module you can create a turning program

**Module description:**

This module describes how to write a part program for a turning machine, by example.

**Module content:**

- Create a work plan
- Create a list of tools
- Define the program structure
- Create example program in editor
- Simulate the program

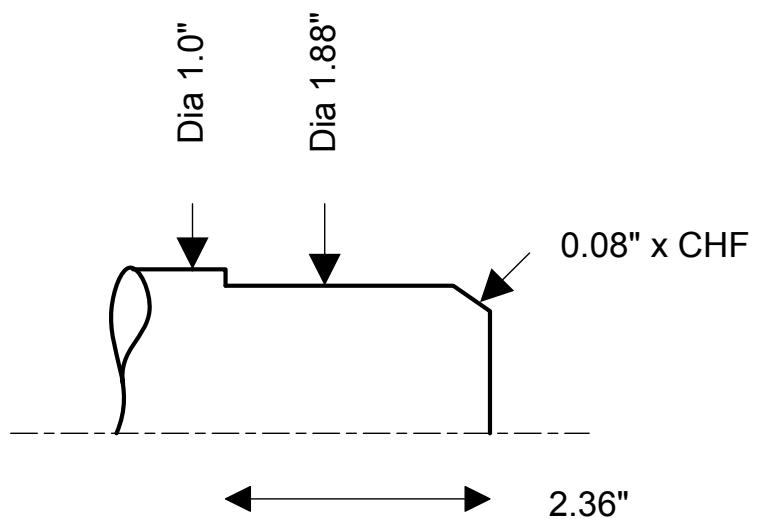


## Section 2

### Create a work plan

Notes

An example program will be created using the following workpiece:



The work plan is used to define the steps which have to be carried out to produce the workpiece.

Working step Nr.	Operation	Tools
1	Rough turn	Roughing tool
2	Finish turn	Finishing tool

## Section 3

### Create a list of tools

The tool list contains the technology information for each tool. This information will be used in the NC program, for the respective tool.  
The speeds and feeds have to be adapted dependant upon the workpiece material.

Notes

Tool / Operation	T-Nr.	Feed inch/rev	Speed ft/min	Spindle direction M3/M4	Coolant
Roughing tool R 0.031"	1	0.018	900	M3	x
Finishing tool R 0.015"	2	0.004	900	M3	x

## Section 4

### Define the program structure

Notes

The program sequence can be created from the work plan.

#### Tool 1: Roughing tool R0.031"

- Select the origin (absolut/incremental)
- Activate zero offset
- Define the working plane
- Tool change
- Program constant surface speed and spindle direction
- Program feedrate
- Switch coolant on
- Face off
- Position on the front face
- Finish turn out side
- Return to toolchange

#### Tool 2: Finishing tool R0.015"

- Select the origin (absolut/incremental)
- Activate zero offset
- Define the working plane
- Tool change
- Program constant surface speed and spindle direction
- Program feedrate
- Switch coolant on
- Face off
- Position on the front face
- Finish turn out side
- Return to toolchange

## Section 5

### Create example program in editor

Notes

A new program has to be generated in order to insert the ISO code.  
The program should be given the name C78\_TURN.  
First a new directory called C78 can to be generated.  
Use the following sequence to complete this task.

PROGRAM  
MANAGER

New

New  
directory



The new directory will be generated, and should be opened

Open

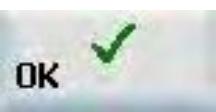
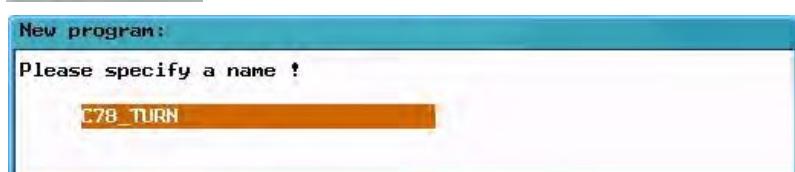
Or with



The main program is generated.

New

New  
file



After generating a new program in the editor, the NC blocks for the surface milling, contour milling and the slot milling have to be typed in.

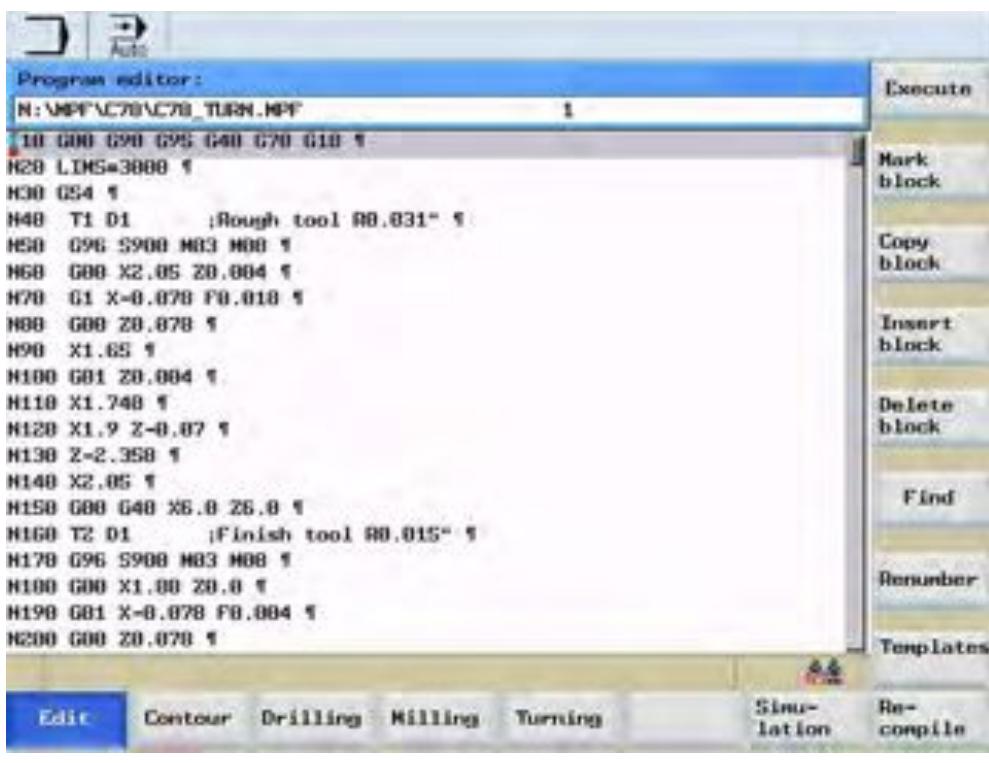
## Section 5

### Create example program in editor

Notes

This is an example of the NC blocks that will machine the component.

```
N10 G00 G90 G95 G40 G70 G18
N20 LIMS=3000
N30 G54
N40 T1 D1 ;Rough tool R0.031"
N50 G96 S900 M03 M08
N60 G00 X2.05 Z0.004
N70 G1 X-0.078 F0.018
N80 G00 Z0.078
N90 X1.65
N100 G01 Z0.004
N110 X1.740
N120 X1.9 Z-0.07
N130 Z-2.358
N140 X2.05
N150 G00 G40 X6.0 Z6.0
N160 T2 D1 ;Finish tool R0.015"
N170 G96 S900 M03 M08
N180 G00 X1.88 Z0.0
N190 G01 X-0.078 F0.004
N200 G00 Z0.078
N210 X1.65
N220 G01 Z0.0
N230 X1.88 CHR=0.08
N240 Z-2.36
N250 X2.05
N260 G00 G40 X6.0 Z6.0
N270 M30
```



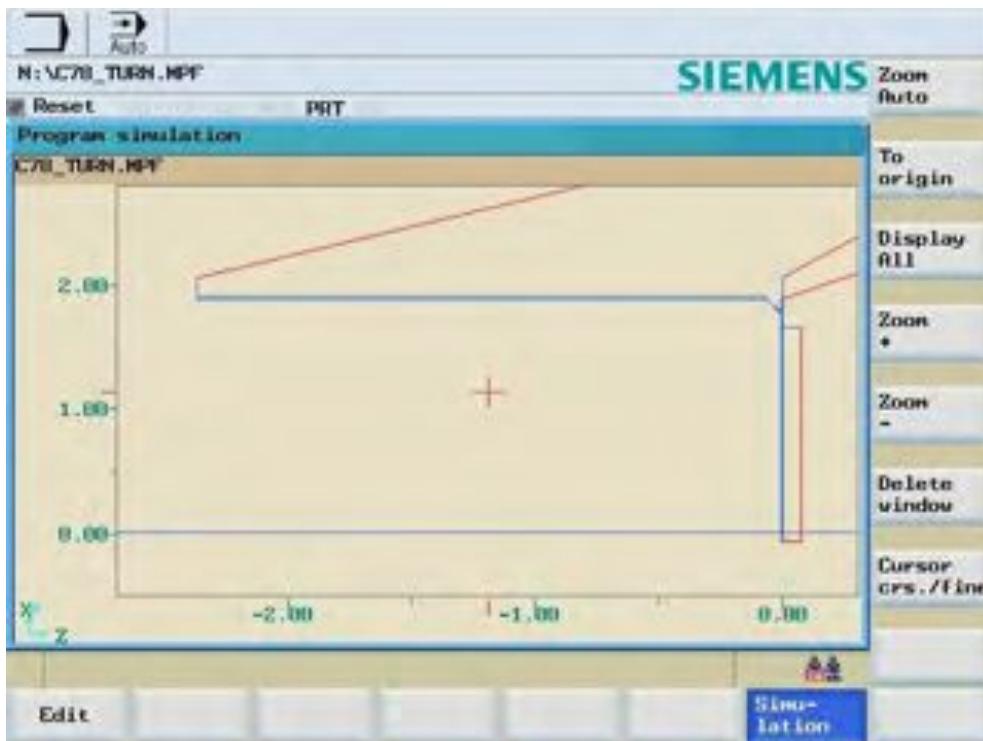
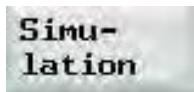
## Section 6

### Simulate the program

Notes

The program can be tested in the simulation mode.  
The necessary tools have to exist in the tool list in order to simulate the program.

This is the sequence to test the NC program in Simulation:

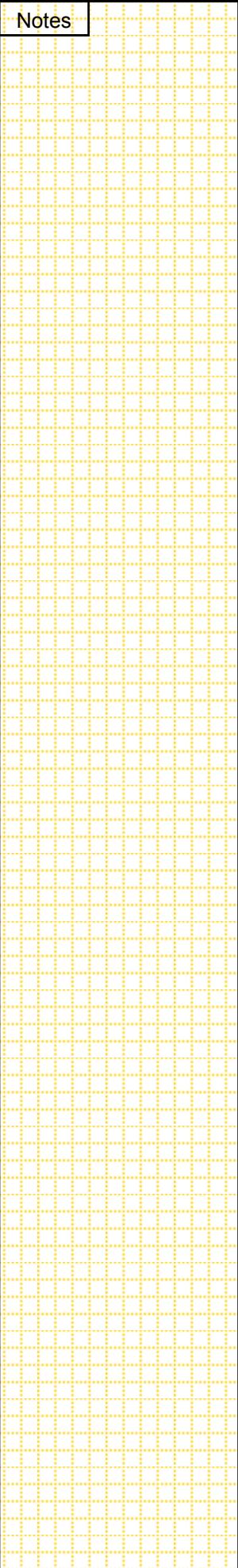


Once you have simulated the NC program you can use the EDIT softkey to go back to the EDITOR and RESET button to eliminate any alarms.



---

Notes



## 1 Brief description

**Module objective:**

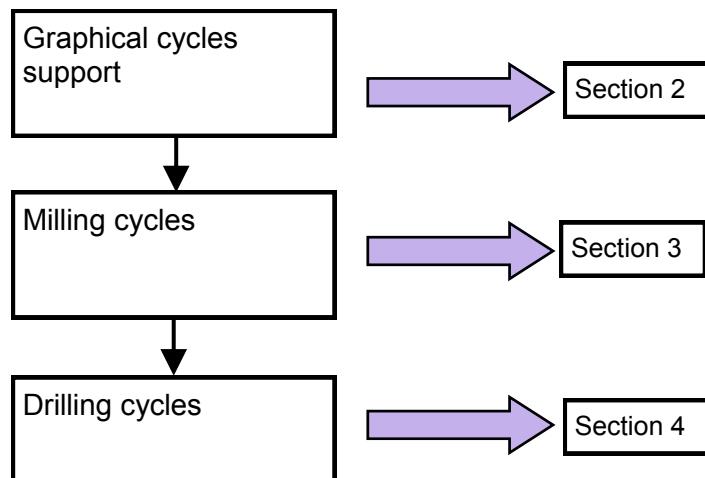
Upon completion of this module you will understand the basic principle and information required for milling cycles.

**Module description:**

Cycles are generally technology subroutines that can be used to carry out a specific machining process, such as drilling of a thread (tapping) or milling of a pocket. These cycles are adapted to individual tasks by parameter assignment.

**Module content:**

Graphical cycles support  
Milling cycles  
Drilling cycles



Notes

#### Graphical cycle support in the program editor

The program editor in the control system provides you with programming support to add cycle calls to the program and to enter parameters.

##### Function

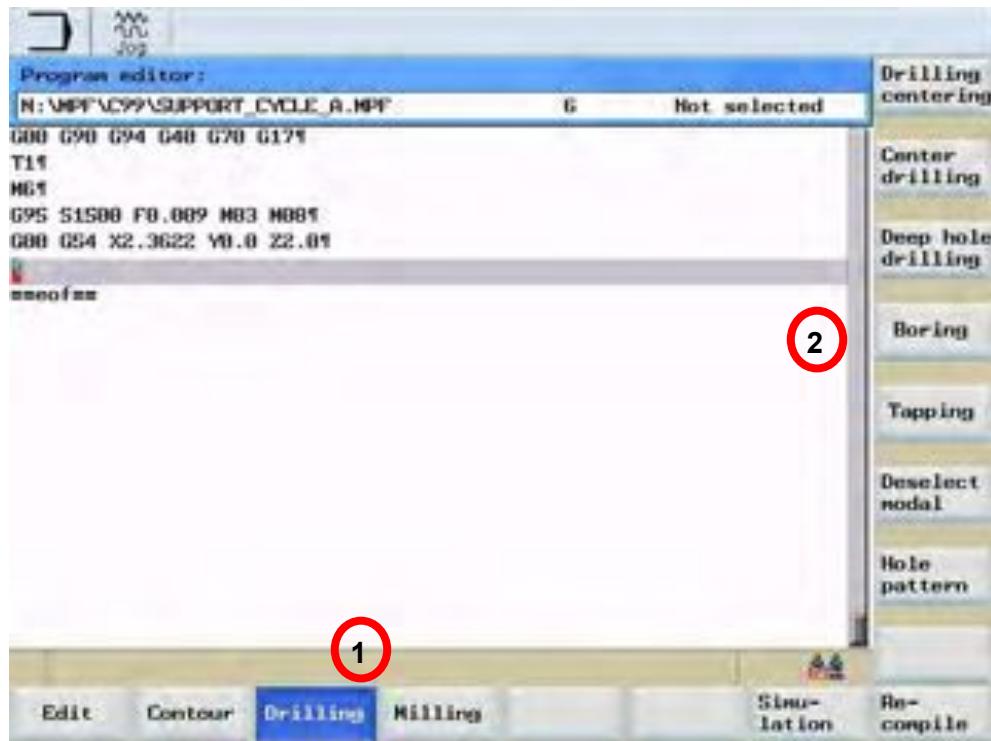
The cycle support consists of three components:

Cycle selection

Input screenforms for parameter assignment

Help screen for each cycle (is to be found in the interactive screen-form).

Once you have created an NC program, you can select a “graphical cycle support either for “drilling” or “milling”.



- 1 From the horizontal keys you can choose from milling or drilling cycles.

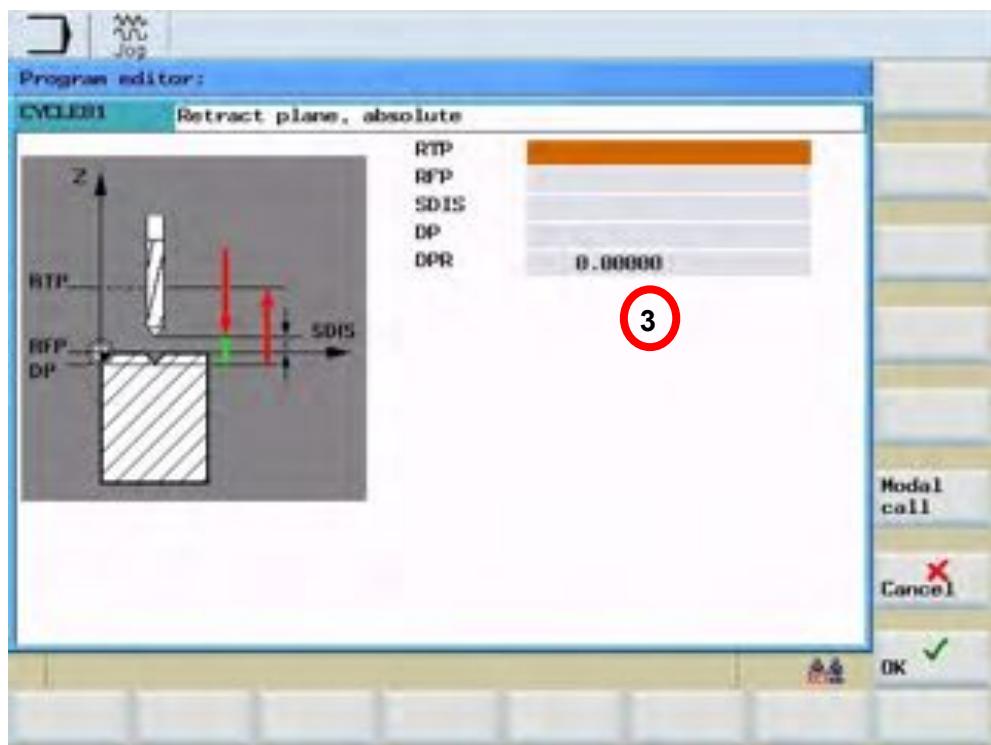
- 2 Once you have chosen milling or drilling cycles you can then use the vertical keys for the different types of cycles within that group.

## Section 2

### Graphical support cycles

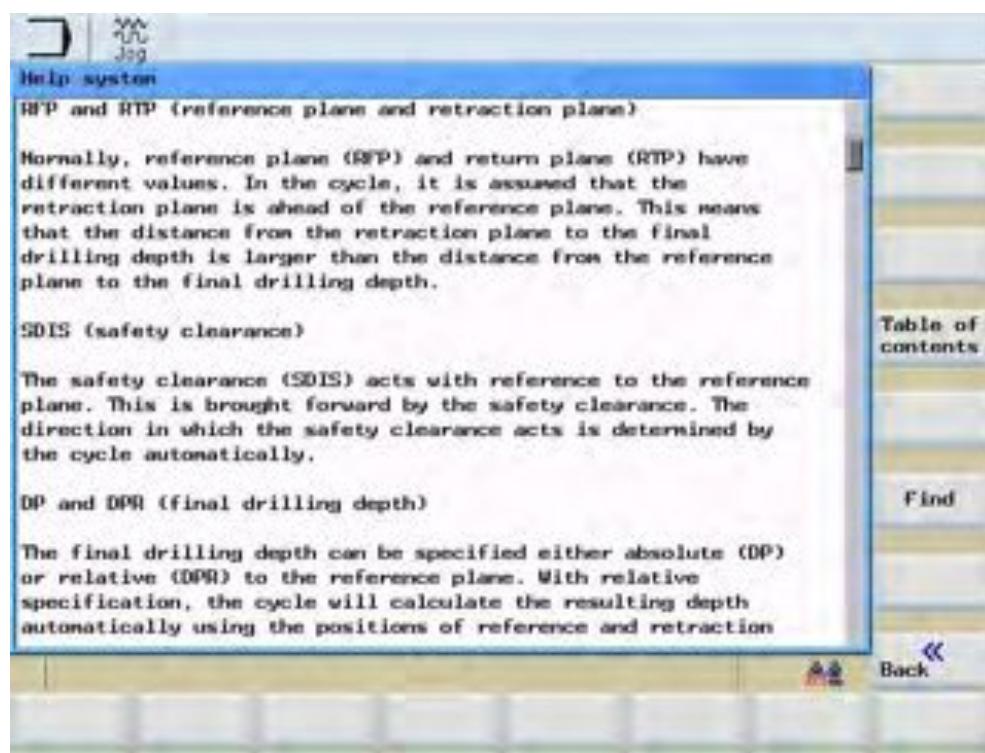
Input screenforms for parameter assignment for Cycle 81 drilling.

Notes



#### 3 Parameter assignments for the cycle

At any time you can press help the button which will give you an explanation for each of the parameters required.

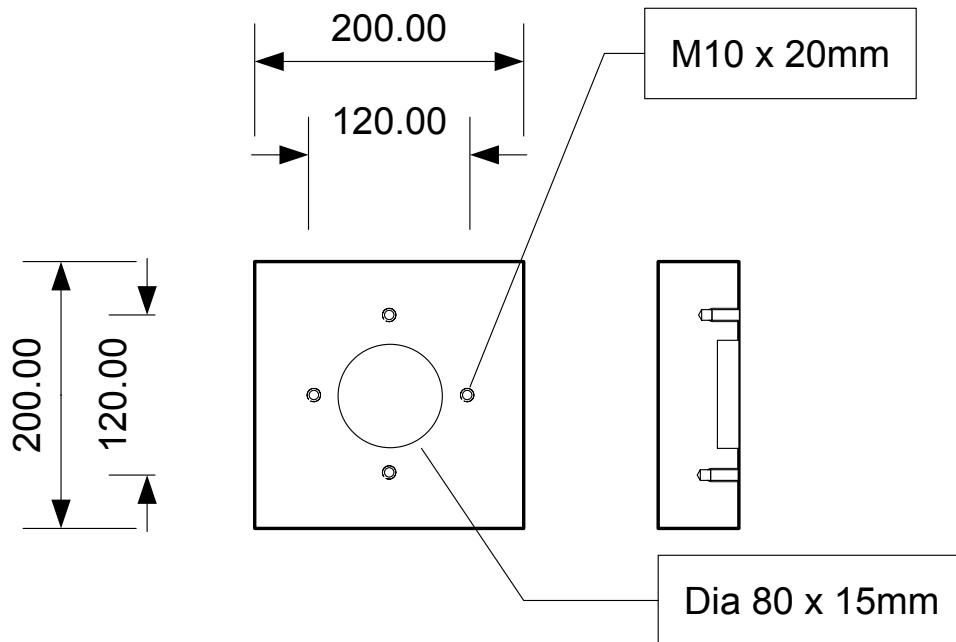


## Section 3

### Milling cycles

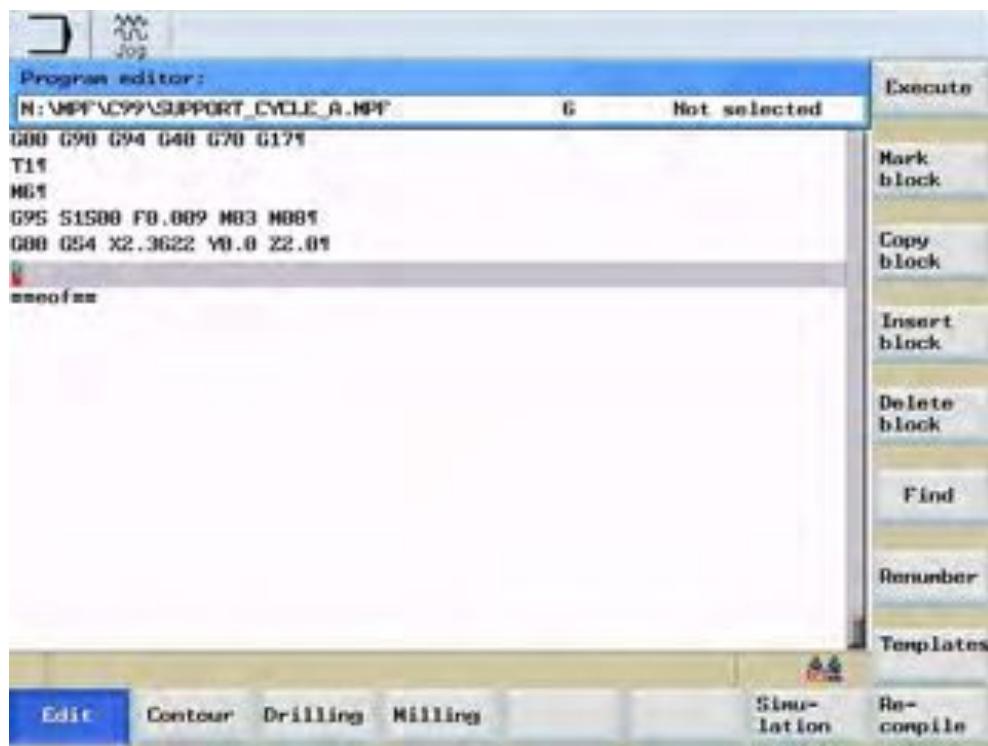
Notes

You are given the task to face and machine the pocket on your work piece.



We would create a basic program so that we can add the facing and pocket cycles.

First of all we will create the basic start of the program.

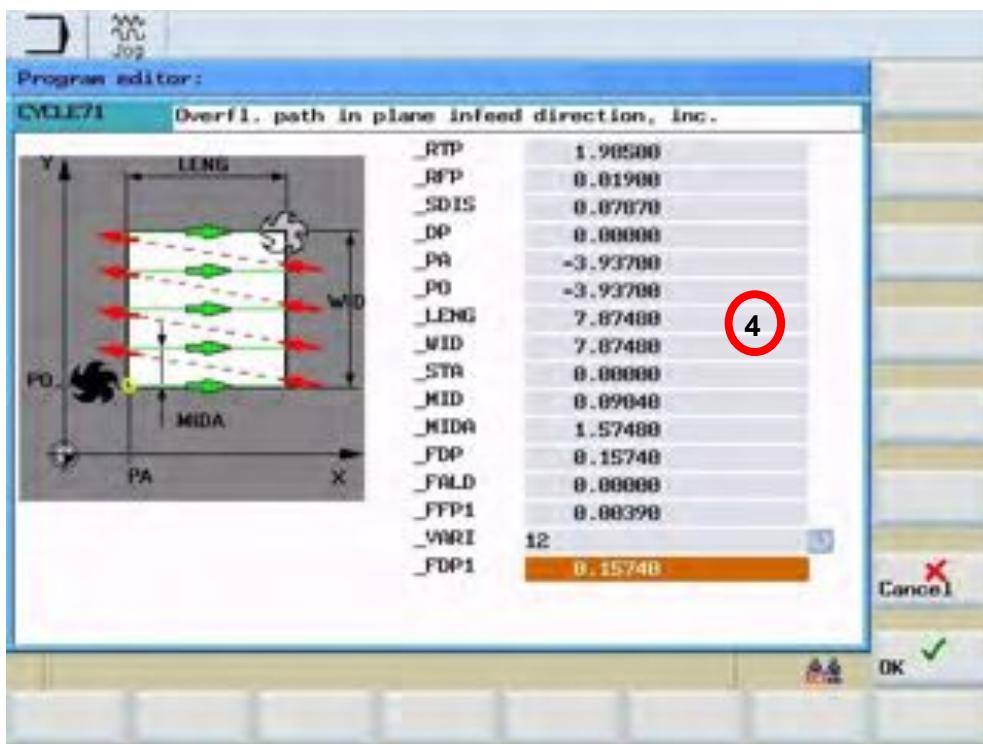
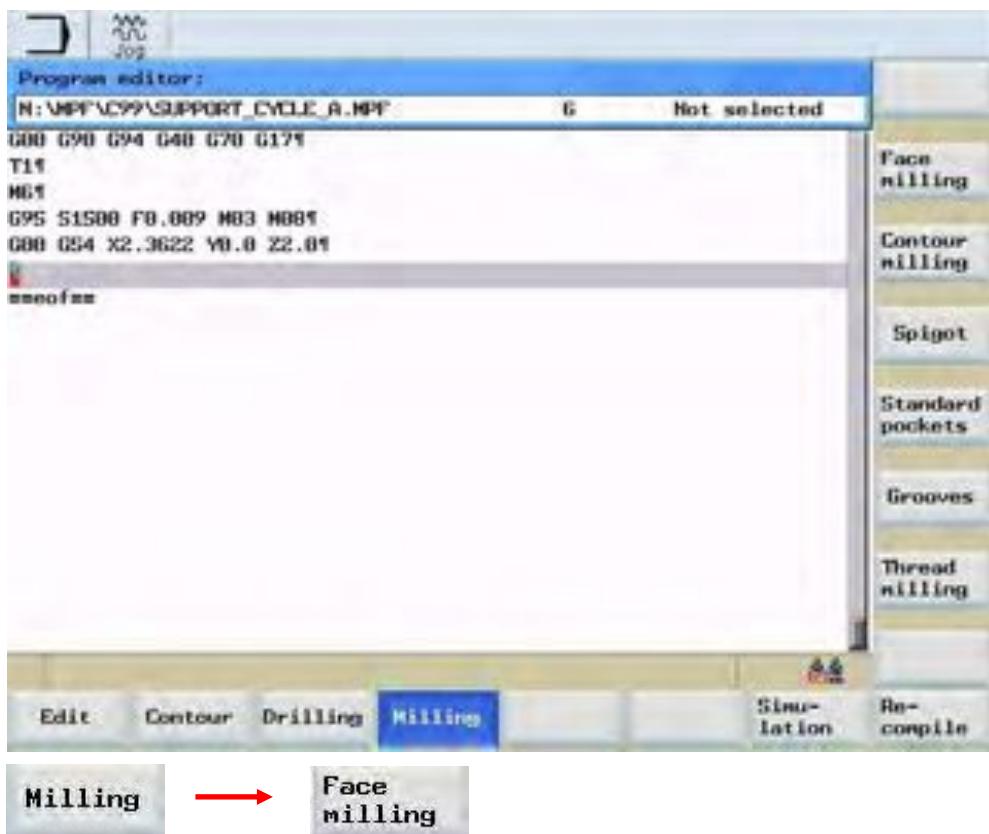


## Section 3

### Milling cycles

Then we will add the first milling cycle, as shown in the following sequence.

Notes



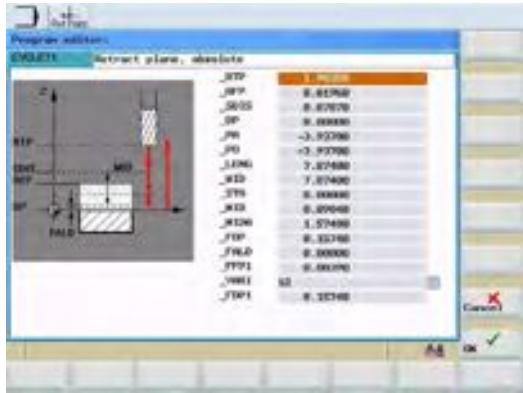
- 4 Enter data taken from your drawing.



## Section 3

### Milling cycles

Notes



Facing - CYCLE71

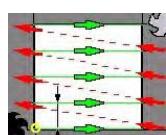
Programming

CYCLE71(RTP, RFP, SDIS, DP, DRP, DTB)

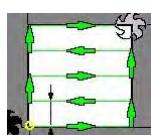
Parameters

_RTP	retraction plane (absolute)
_RFP	reference plane (absolute)
_SDIS	safety plane (enter without sign)
_DP	final drilling depth (absolute)
_PA	starting point of 1st axis (plane you are working in)
_PO	starting point of 2nd axis (plane you are working in)
_LENG	length along 1st axis, incremental with sign from start point
_WID	length along 2nd axis, incremental with sign from start point
_STA	angle between longitudinal axis and the 1st axis of the plane you are working in (enter without sign)
_MID	maximum depth of cut (enter without sign)
_MIDA	maximum width of cut by cutter
_FDP	distance past end of material in direction of cut
_FALD	maximum finish depth of cut (enter without sign)
_FFP1	feedrate for surface machining
_VARI	machining type UNITS value 1: roughing 2: finishing TENS value 1: parallel to 1st axis, unidirectional 2: parallel to 2nd axis, unidirectional 3: parallel to 1st axis, changing direction 4: parallel to 2nd axis, changing direction

1 & 2



3 & 4



\_FDP1

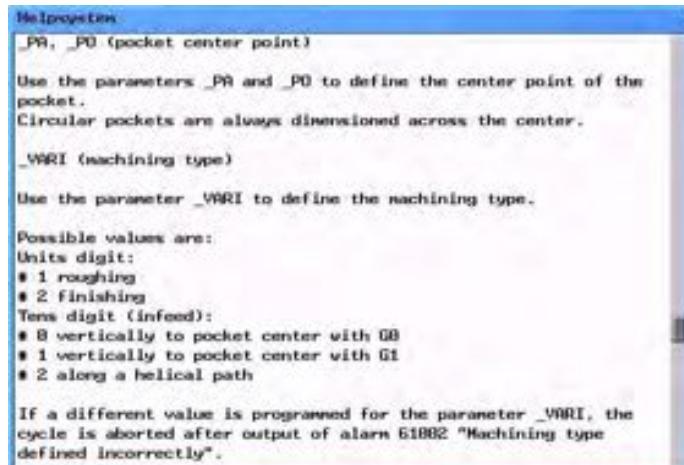
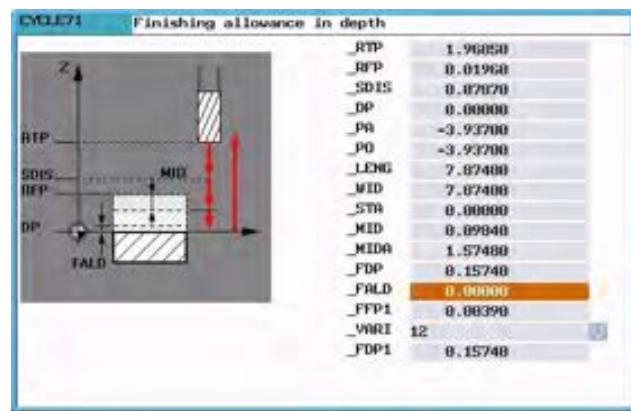
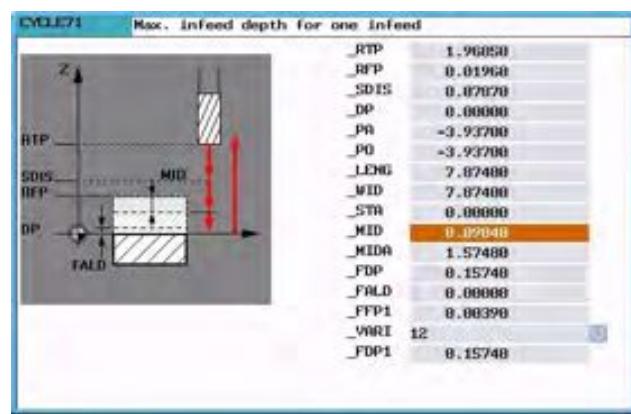
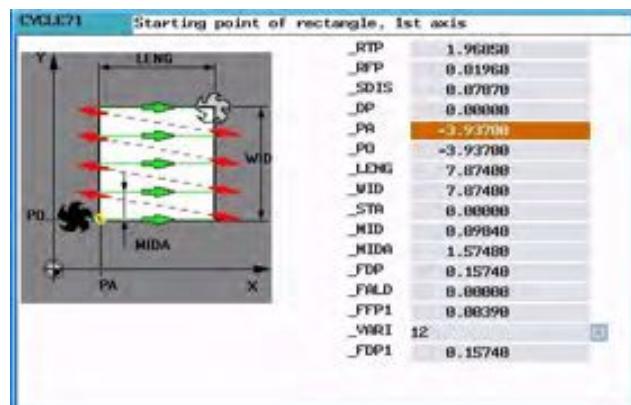
Distance of overhang by cutter along direction of cut

## Section 3

### Milling cycles

If you look below, as you cursor down the parameter windows you will get a prompt with text, which relates to the given pictures.

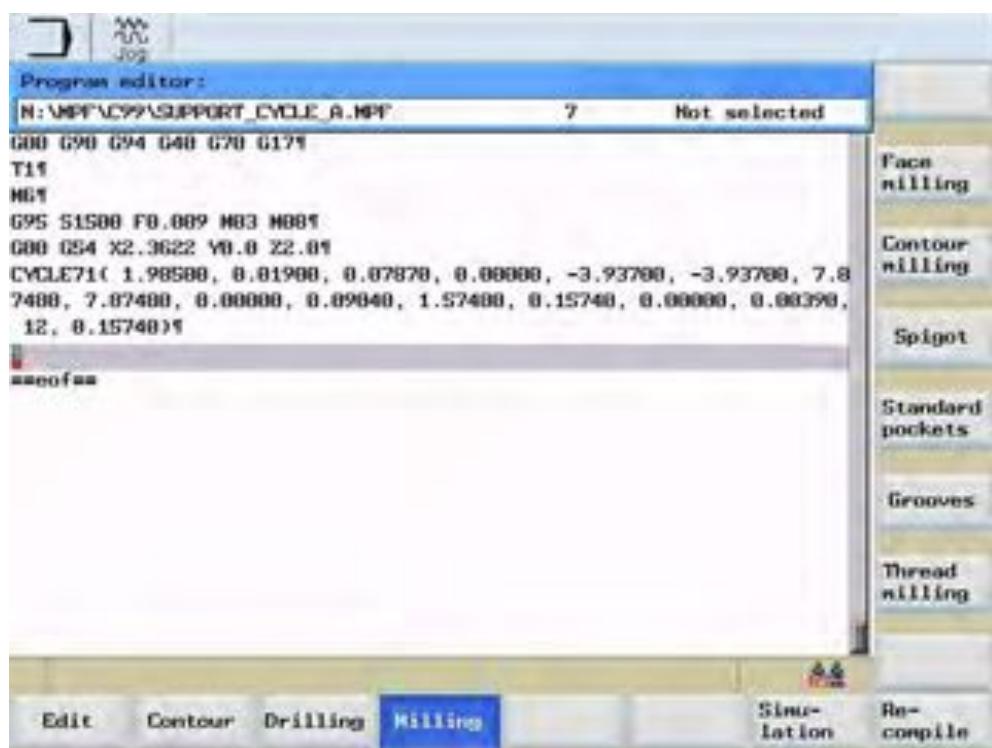
Notes



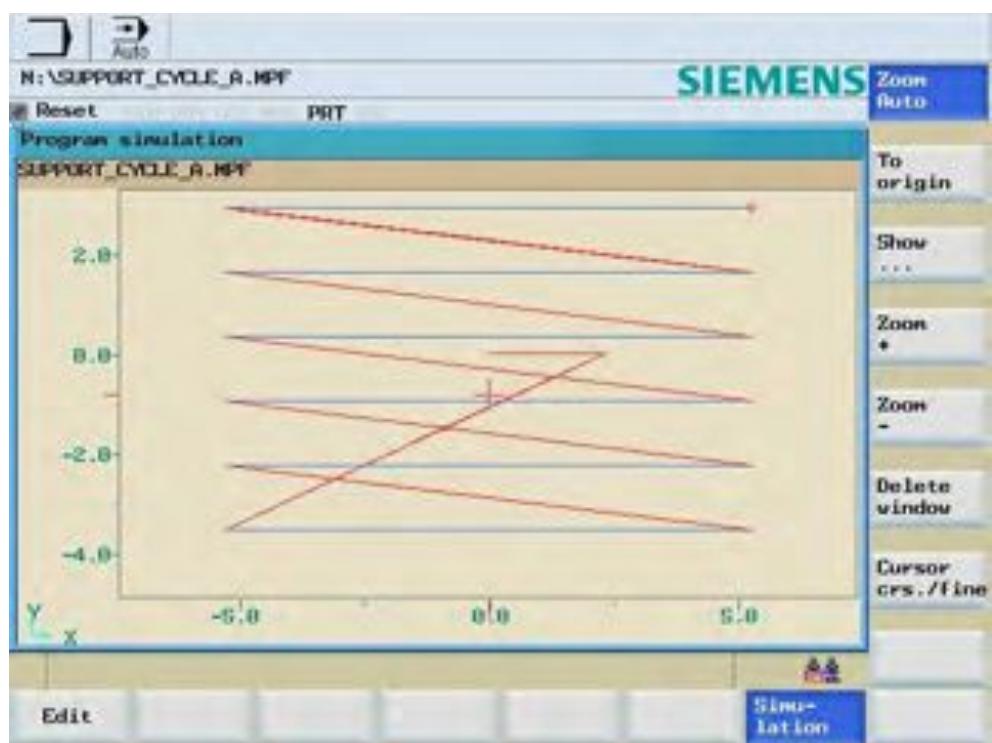
## Section 3

### Milling cycles

Notes



Your program should now look as above.  
We have now created a program to face off the component.



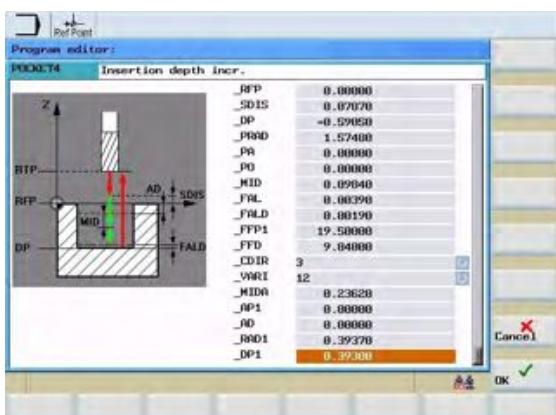
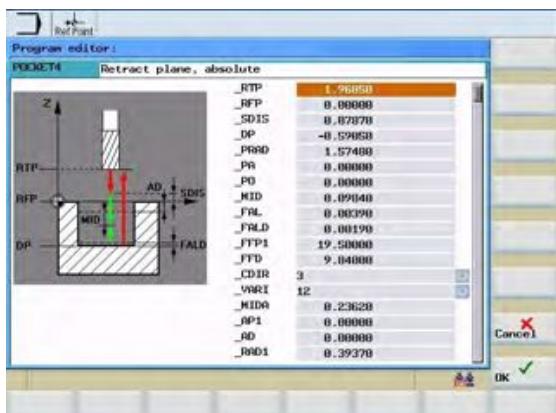
See if you can now create the rest of the program to machine the pocket.

## Section 3

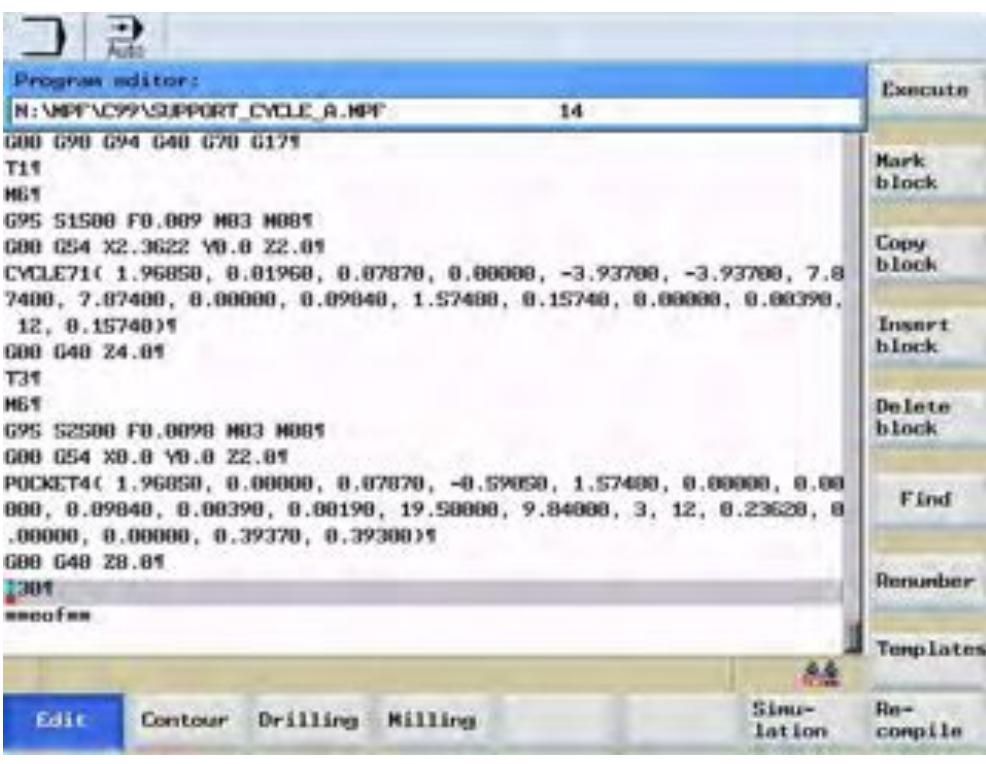
### Milling cycles

To create the rest of the program, you will have to use POCKET4 (Circular Pocket).

Notes



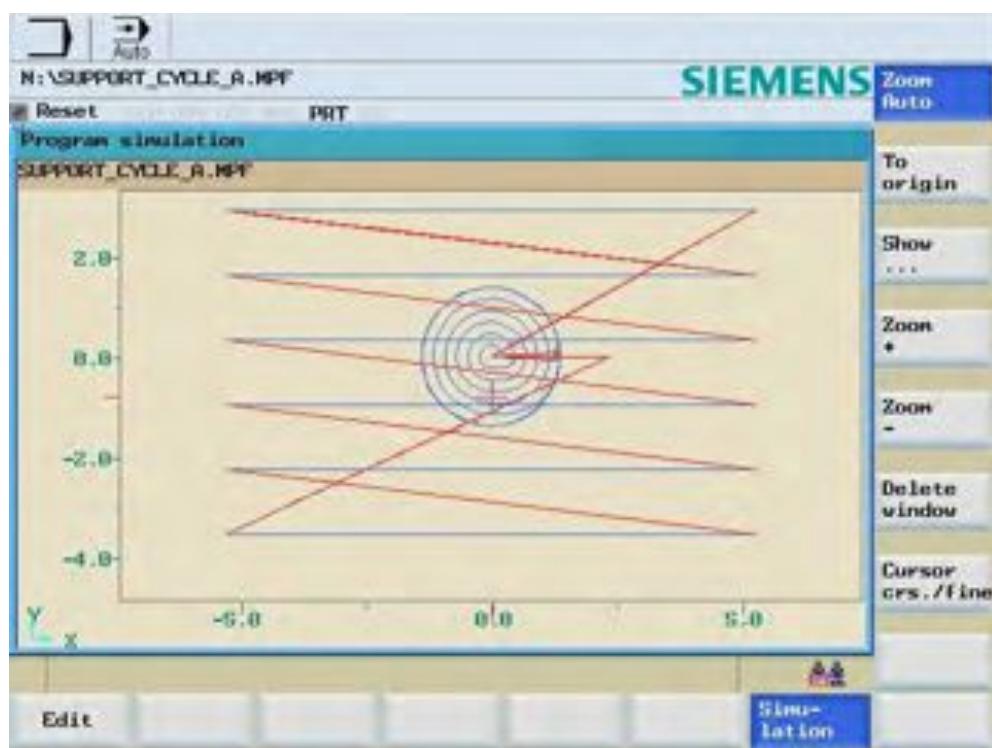
Final program and simulation.



## Section 3

### Milling cycles

Notes

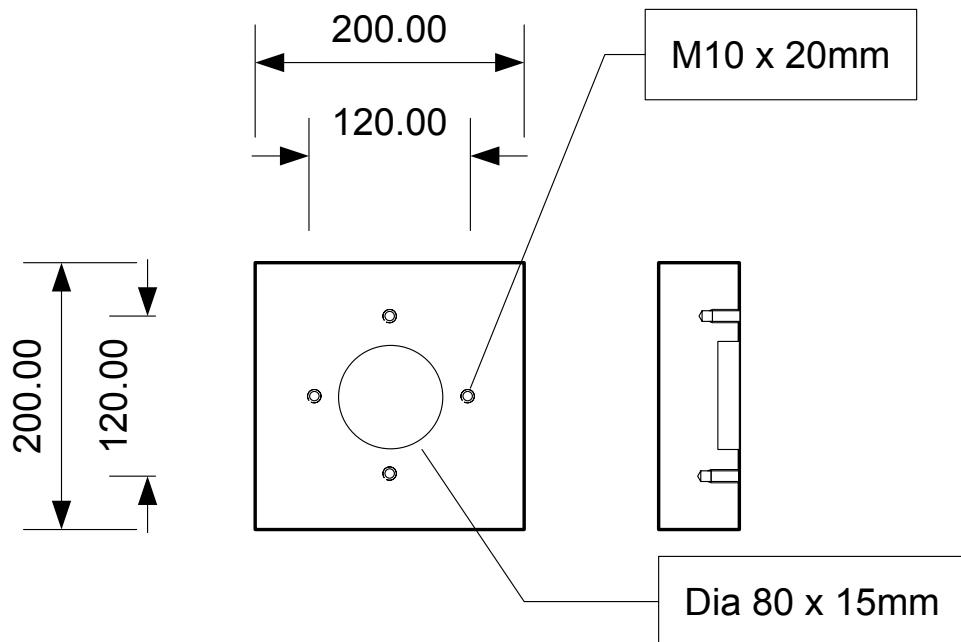


## Section 4

### Drilling cycles

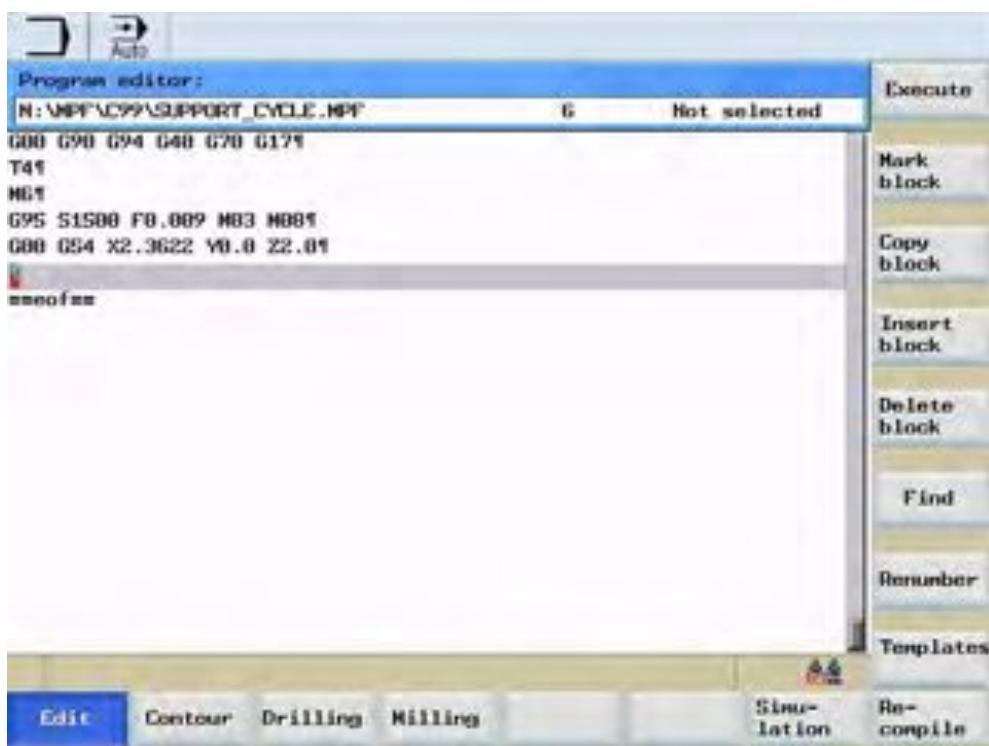
Notes

You are given the task to drill and tap 4 holes in your work piece.



We would create a basic program to CENTER DRILL, DRILL and TAP.

First of all we will create the basic start of the program.

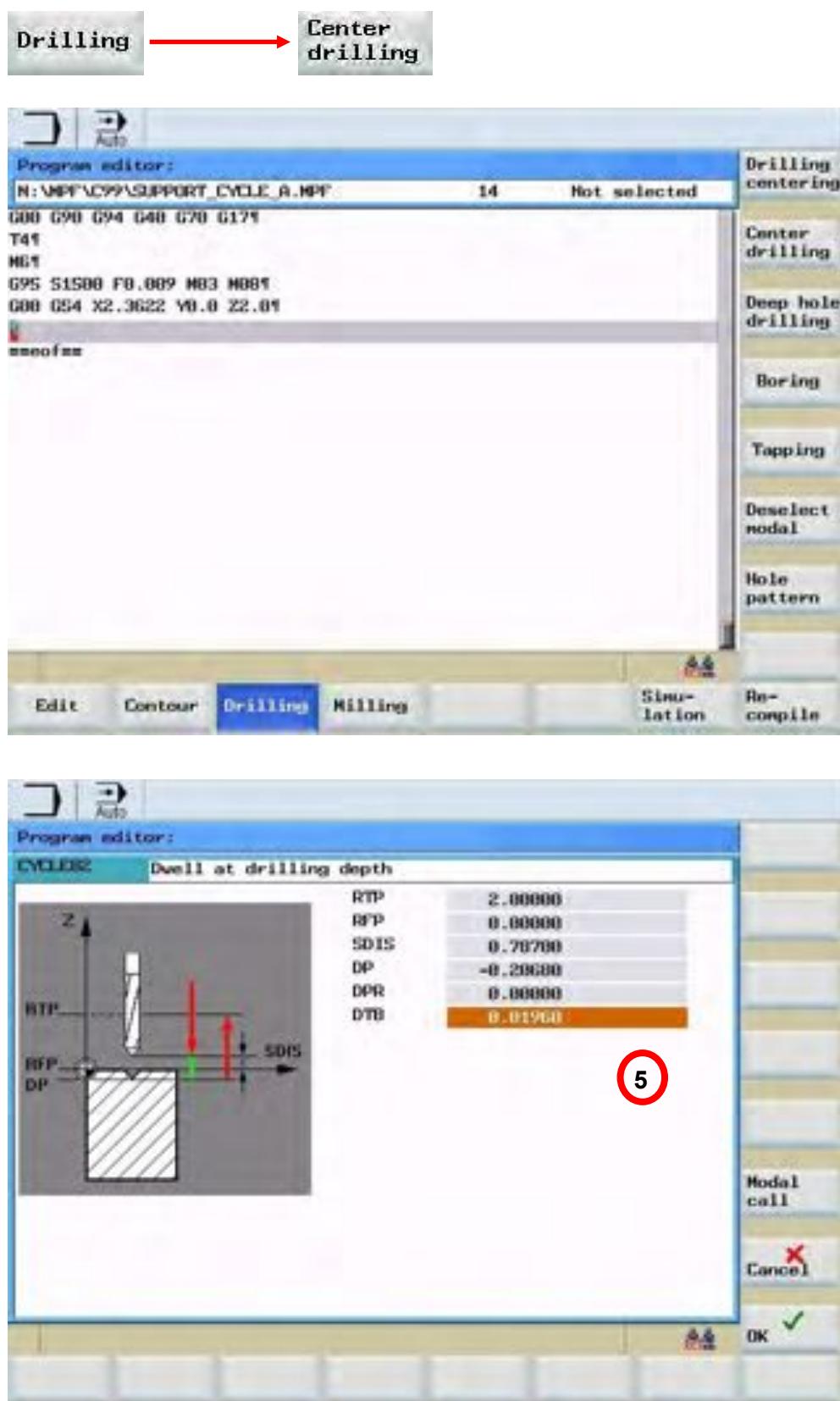


## Section 4

### Drilling cycles

Notes

Then we will add the first drilling cycle, as shown in the following sequence.



5

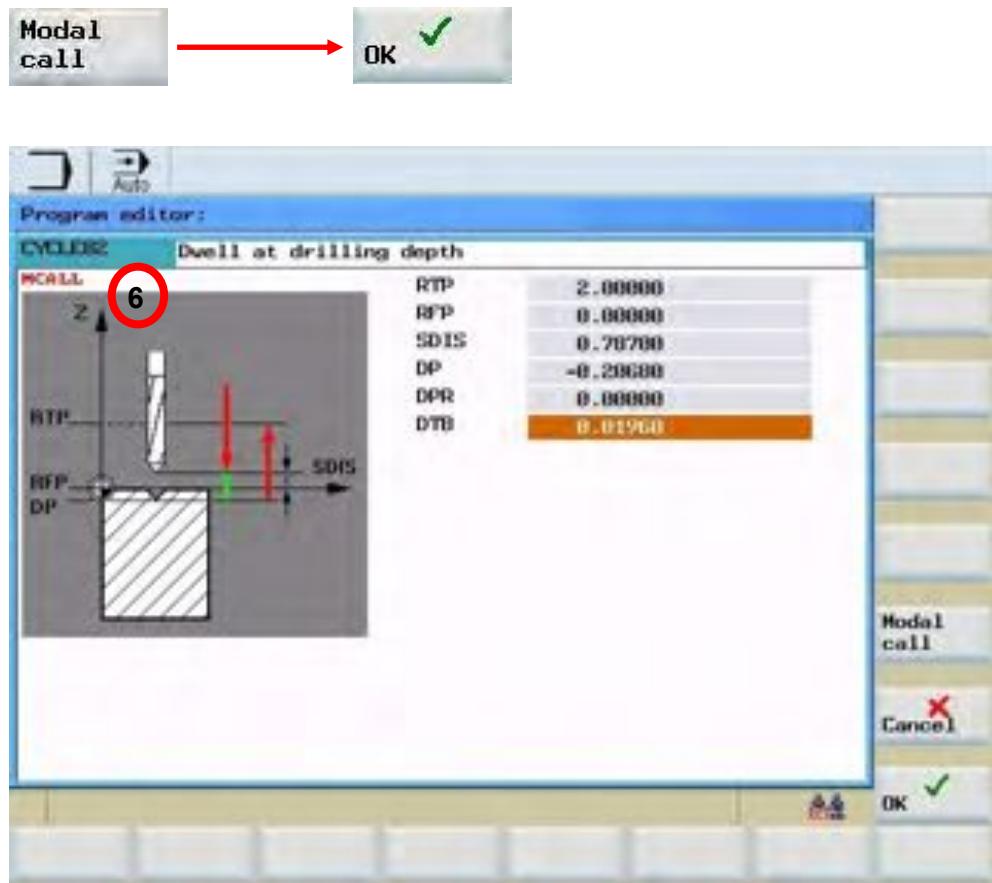
Enter data taken from your drawing.

## Section 4

### Drilling cycles

Notes

Make the drilling cycle MODAL using the following sequence.

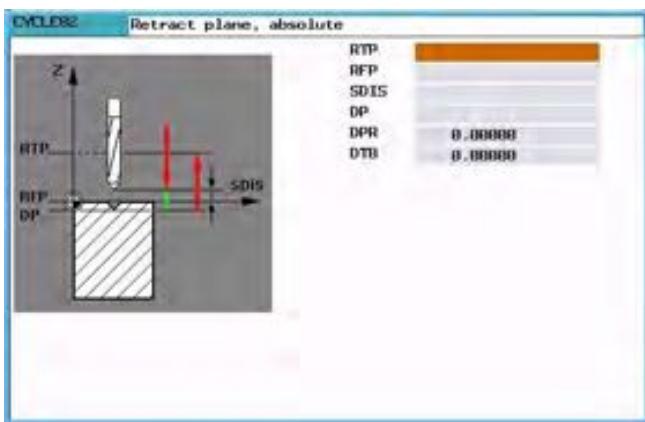


- 6 Note the MCALL in red text.



## Section 4

### Drilling cycles



Notes

Drilling, centering - CYCLE82

Programming

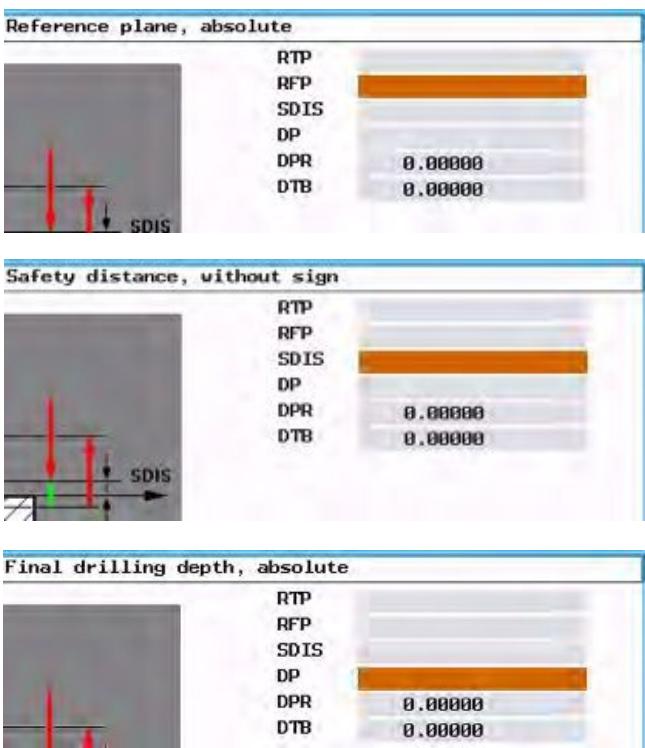
CYCLE82(RTP, RFP, SDIS, DP, DRP, DTB)

Parameters

RTP	retraction plane (absolute)
RFP	reference plane (absolute)
SDIS	safety plane (enter without sign)
DP	final drilling depth (absolute)
DPR	final drilling depth relative to the reference plane (enter without sign)
DTB	dwell time at final depth (chip breaking)

These six parameters are used through out all the drilling cycles, but as more functionality is required as are the number of parameters increased.

If you look below, as you cursor down the parameter windows you will get a prompt with text, which relates to the picture guide.

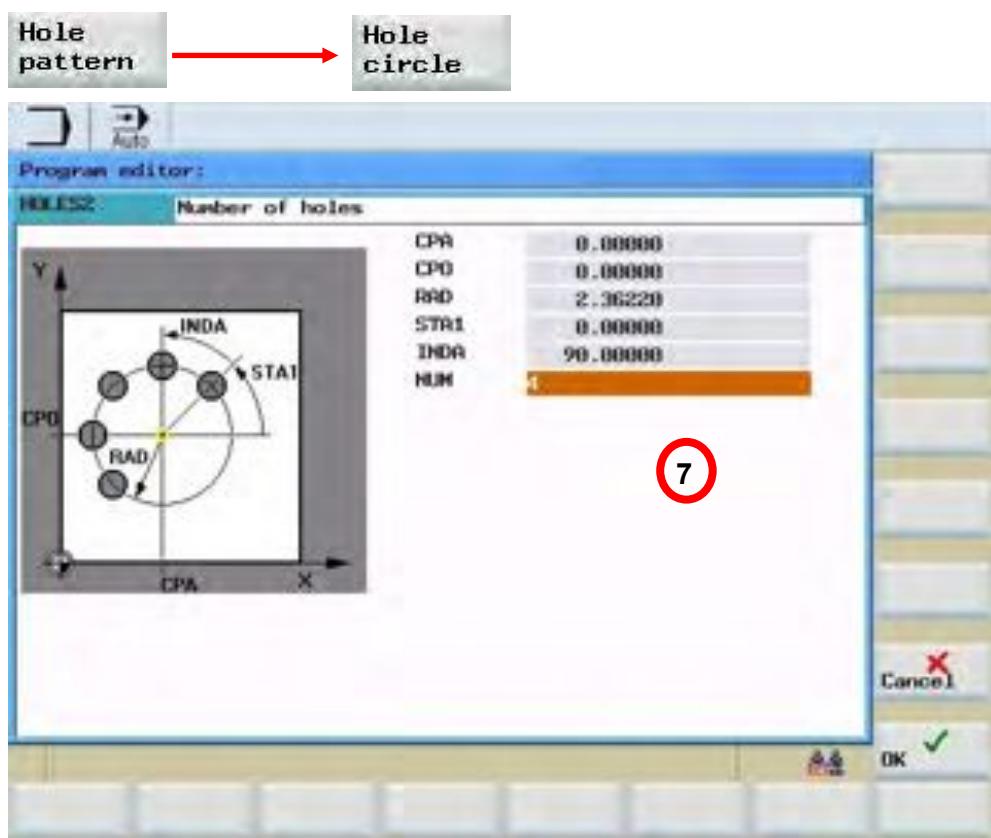


## Section 4

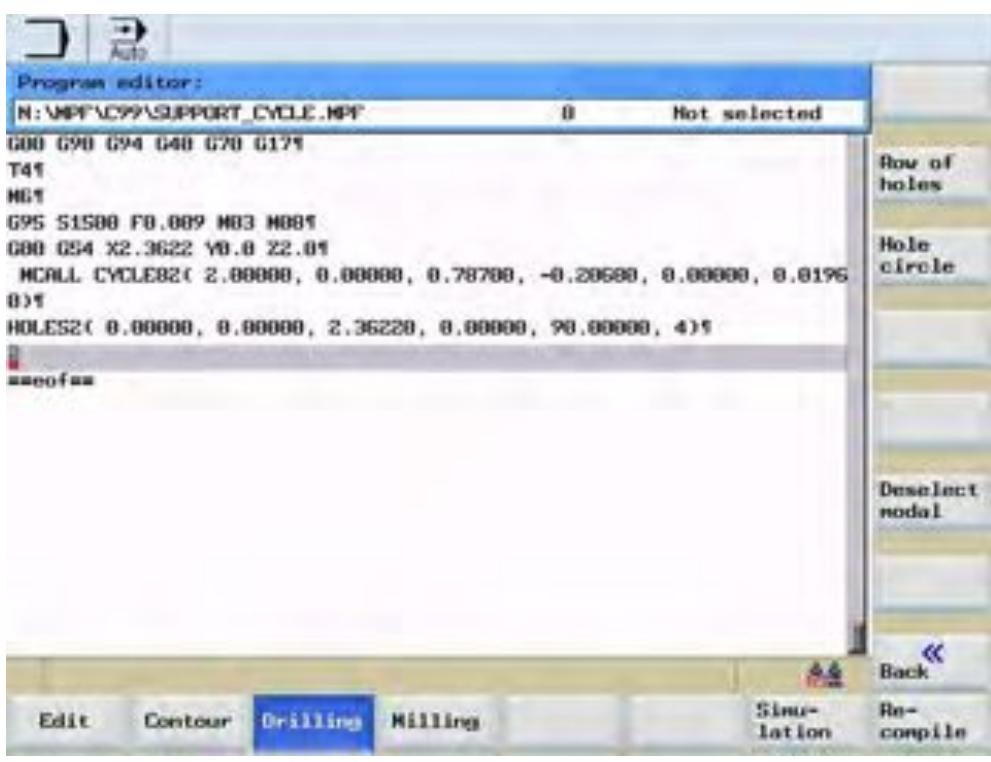
### Drilling cycles

Notes

We then add the positions of the 4 holes.



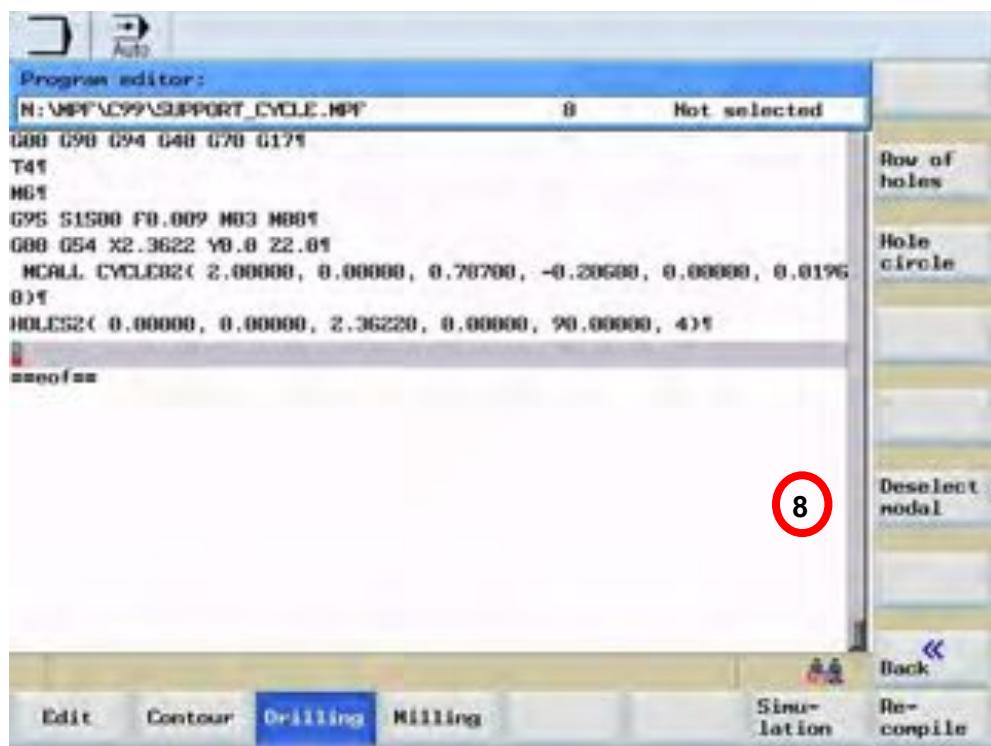
- 7 Enter data taken from your drawing



## Section 4

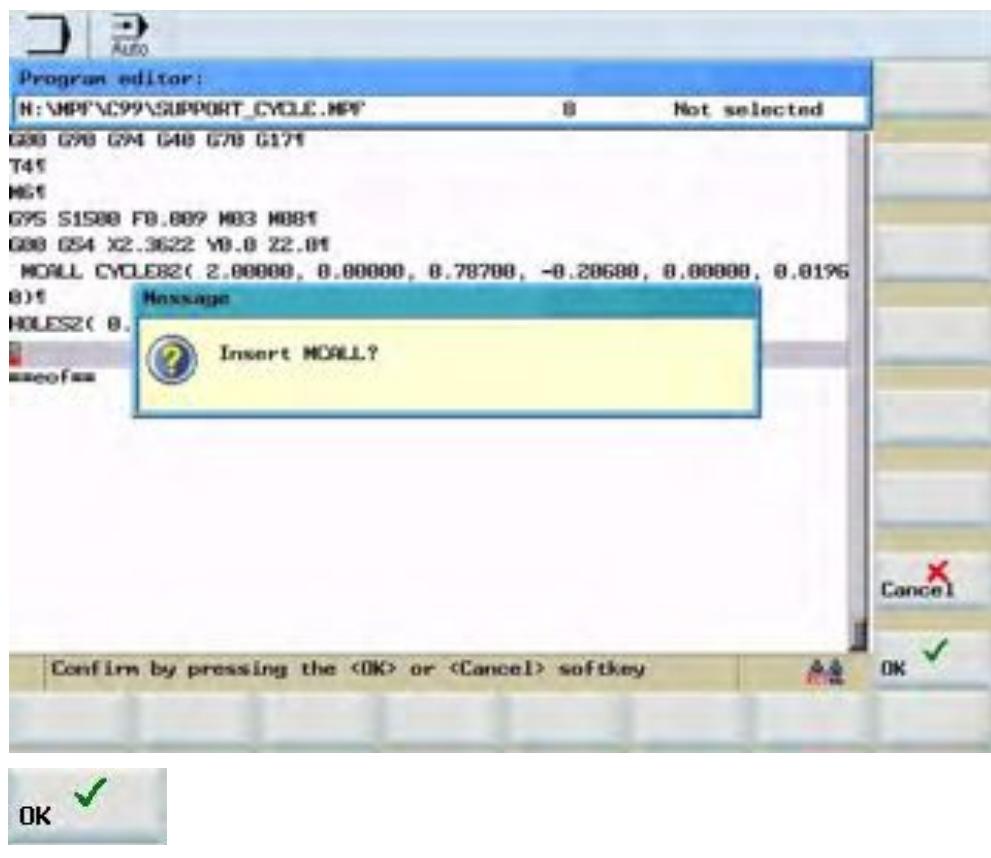
### Drilling cycles

Notes



- 8 Because the drilling cycle was a modal call, you have to deactivate the call

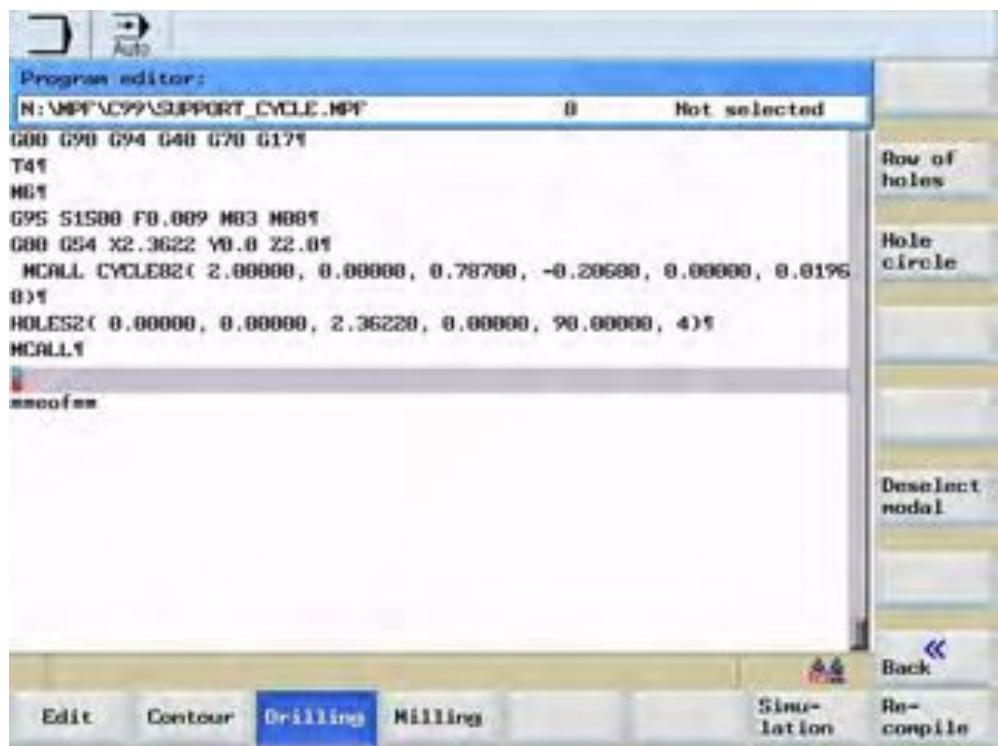
#### Deselect modal



## Section 4

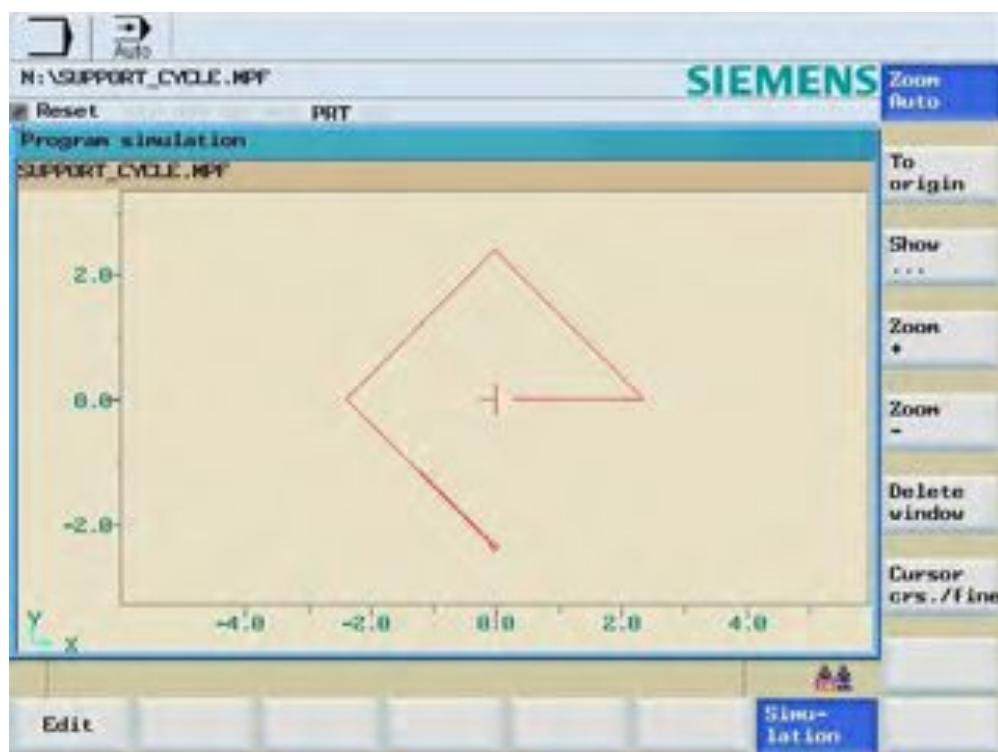
### Drilling cycles

Notes



Your program should now look as above.  
We have now created a program to center drill 4 holes.

Simu-  
lation



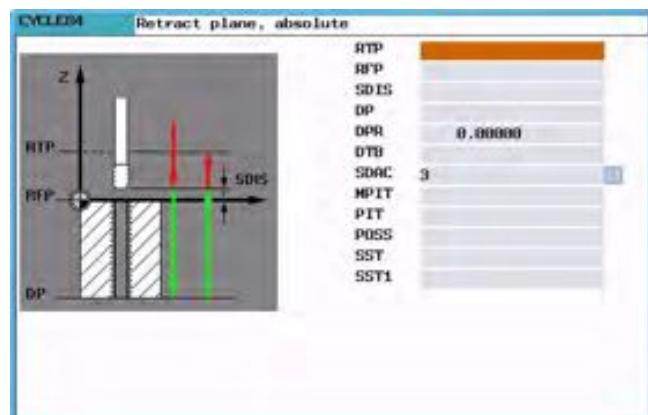
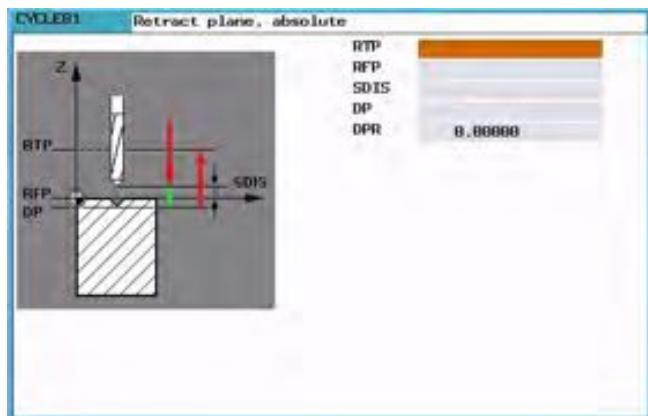
See if you can now create the rest of the program to drill and tap the 4 holes.

## Section 4

### Drilling cycles

To create the rest of the program, you will have to use CYCLE81 and CYCLE84 and HOLES2, remembering to make the correct CYCLES modal and non-modal.

Notes



Final program. For Centre Drill, Drill and Tap.

```
N:\MPF\FC99\SUPPORT CYCLE.MPF          23
MCALL CYCLE821( 1.96850, 0.00000, 0.07870, -0.20660, 0.00000, 0.10000
0?1
HOLES2( 0.00000, 0.00000, 2.36220, 0.00000, 90.00000, 405
MCALL1
G80 G40 Z4.01
T21
M61
G95 S800 F0.0078 M03 M081
G80 X2.3622 Y0.0 Z2.0
MCALL CYCLE811( 1.96850, 0.00000, 0.07870, -1.18110, 0.00000)1
HOLES2( 0.00000, 0.00000, 2.36220, 0.00000, 90.00000, 405
G80 G40 Z4.01
T31
M61
G95 S800 F0.0078 M03 M081
G80 X2.3622 Y0.0 Z2.0
MCALL CYCLE841( 1.96850, 0.00000, 0.07870, -0.70660, 0.00000, 0.10000
0, 3, 10.00000, , 000.00000, 000.00000)1
HOLES2( 0.00000, 0.00000, 2.36220, 0.00000, 90.00000, 405
MCALL1
```

## Section 4

### Drilling cycles

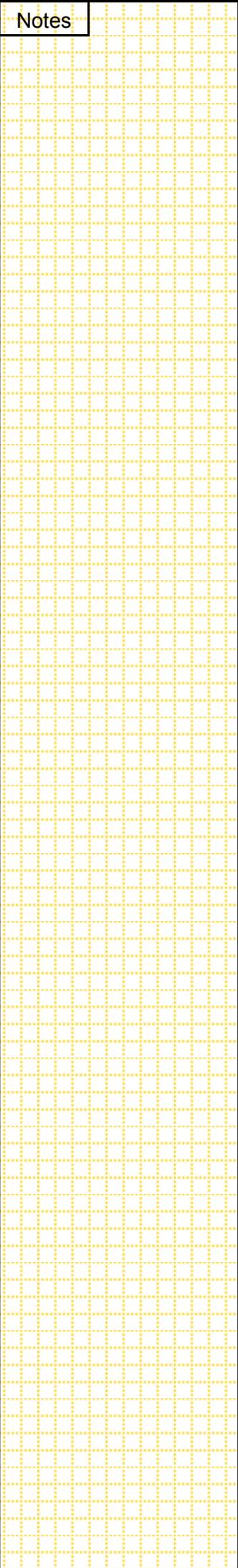
If you now put both parts of the programs together, then you would have a program that would FACE, machine POCKET, CENTER DRILL, DRILL and TAP your component.

Notes

```
G00 G90 G94 G40 G70 G17
T1
M6
G95 S1500 F0.009 M03 M08
G00 G54 X2.3622 Y0.0 Z2.0
CYCLE71( 1.96850, 0.01960, 0.07870, 0.00000, -3.93700, -3.93700,
7.87400, 7.87400, 0.00000, 0.09840, 1.57480, 0.15740, 0.00000, 0.00390,
12, 0.15740)
G00 G40 Z4.0
T3
M6
G95 S2500 F0.0098 M03 M08
G00 G54 X0.0 Y0.0 Z2.0
POCKET4( 1.96850, 0.00000, 0.07870, -0.59050, 1.57480, 0.00000,
0.00000, 0.09840, 0.00390, 0.00190, 19.50000, 9.84000, 3, 12, 0.23620,
0.00000, 0.00000, 0.39370, 0.39300)
G00 G40 Z8.0
T4
M6
G95 S1500 F0.009 M03 M08
G00 G54 X2.3622 Y0.0 Z2.0
MCALL CYCLE82( 1.96850, 0.00000, 0.07870, -0.20660, 0.00000,
0.10000)
HOLES2( 0.00000, 0.00000, 2.36220, 0.00000, 90.00000, 4)
MCALL
G00 G40 Z4.0
T5
M6
G95 S800 F0.0078 M03 M08
G00 G54 X2.3622 Y0.0 Z2.0
MCALL CYCLE81( 1.96850, 0.00000, 0.07870, -1.18110, 0.00000)
HOLES2( 0.00000, 0.00000, 2.36220, 0.00000, 90.00000, 4)
G00 G40 Z4.0
T6
M6
G95 S800 F0.0078 M03 M08
G00 X2.3622 Y0.0 Z2.0
MCALL CYCLE84( 1.96850, 0.00000, 0.07870, -0.70860, 0.00000,
0.10000, 3, 10.00000, , ,800.00000, 800.00000)
HOLES2( 0.00000, 0.00000, 2.36220, 0.00000, 90.00000, 4)
MCALL
G00 G40 X0.0 Y6.0 Z8.0
M30
```

---

Notes



## 1 Brief description

**Module objective:**

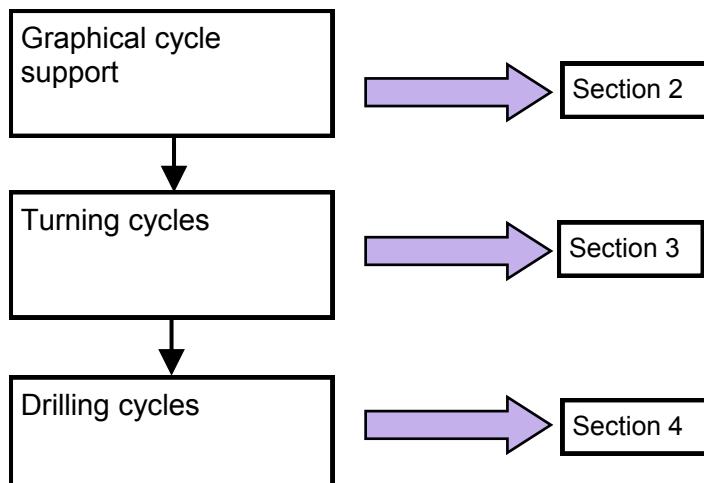
Upon completion of this module you will understand the basic principle and information required for turning cycles.

**Module description:**

Cycles are generally technology subroutines that can be used to carry out a specific machining process, such as turning of a contour or drilling. These cycles are adapted to individual tasks by parameter assignment.

**Module content:**

Graphical cycle support  
Turning cycles  
Drilling cycles



## Section 2

### Graphical cycle support

Notes

#### Graphical cycle support in the program editor

The program editor in the control system provides you with programming support to add cycle calls to the program and to enter parameters from your drawing.

##### Function

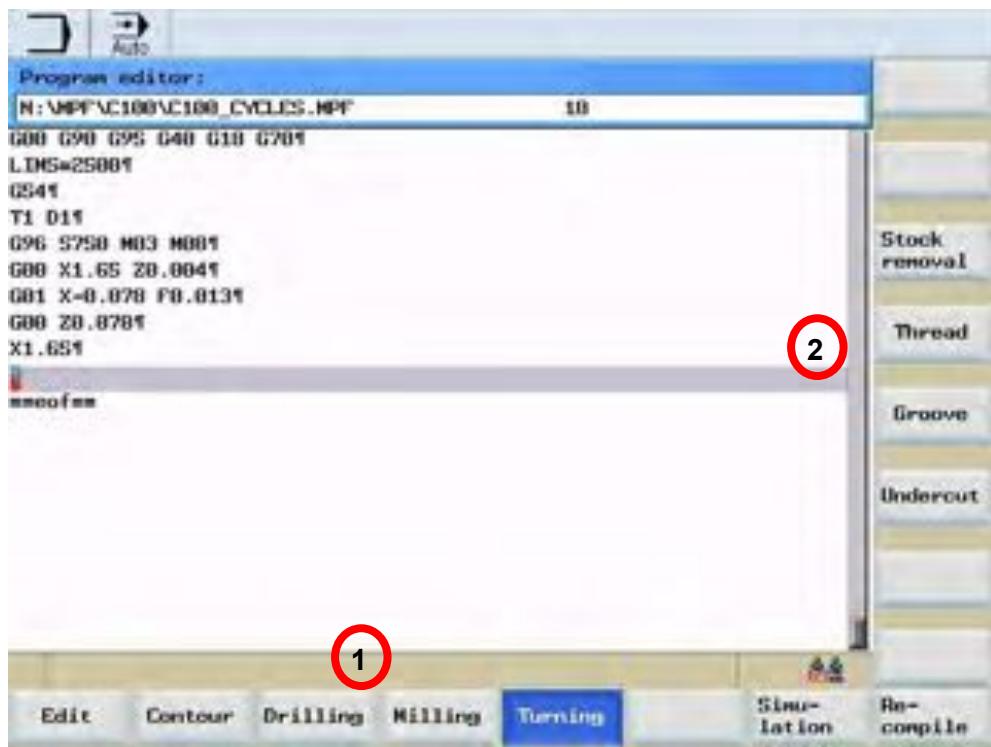
The cycle support consists of three components:

Cycle selection

Input screenforms for parameter assignment

Help screen for each cycle (is to be found in the interactive screen-form).

Once you have created an NC program, you can select a “graphical cycle support” either for “Drilling” or “Turning”.



- 1 From the horizontal keys you can choose from turning or drilling cycles.

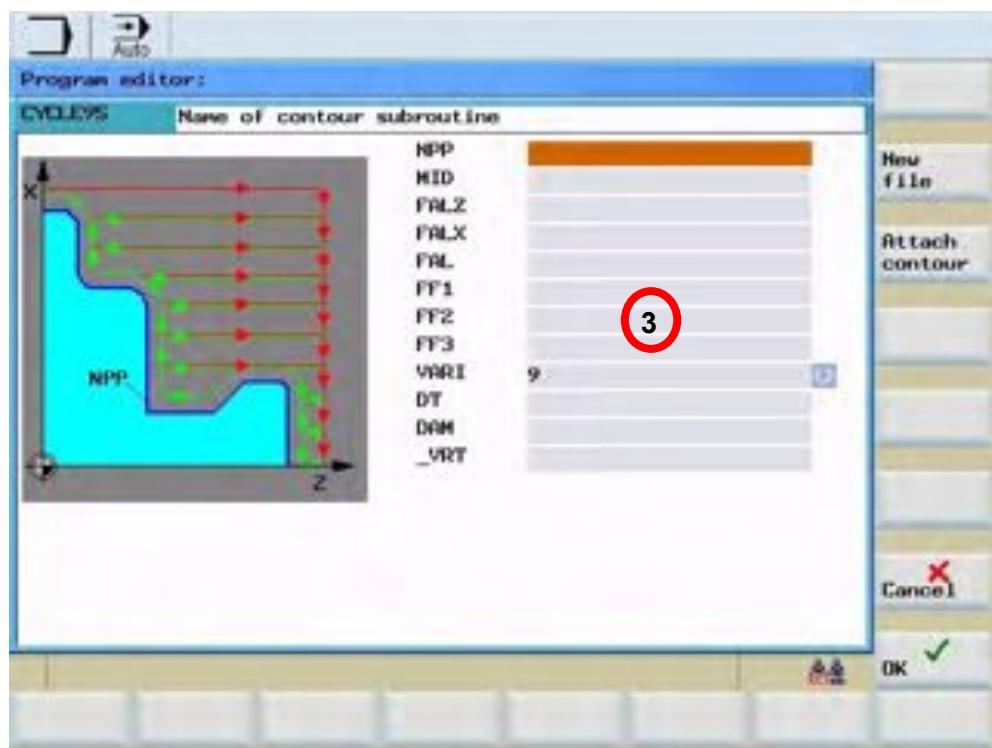
- 2 Once you have chosen turning or drilling cycles you can then use the vertical keys for the different types of cycles within that group.

## Section 2

### Graphical support cycles

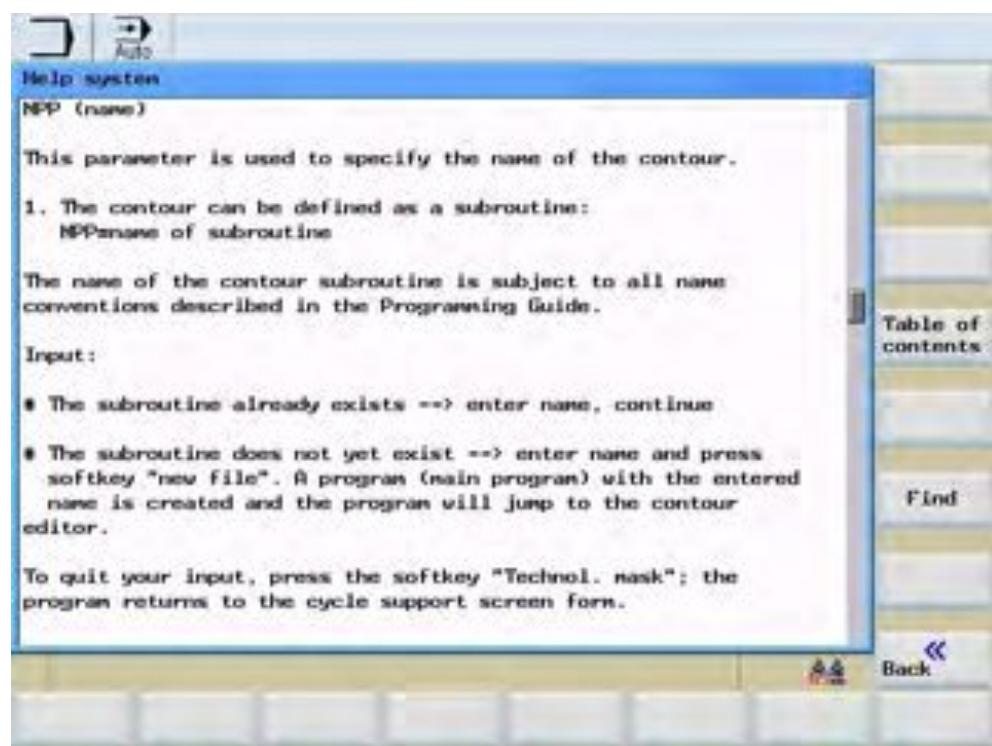
Input screenforms for parameter assignment for Cycle 95 Stock removal.

Notes



#### 3 Parameter assignments for the cycle

At any time you can press the help button which will give you an explanation for each of the parameters required.

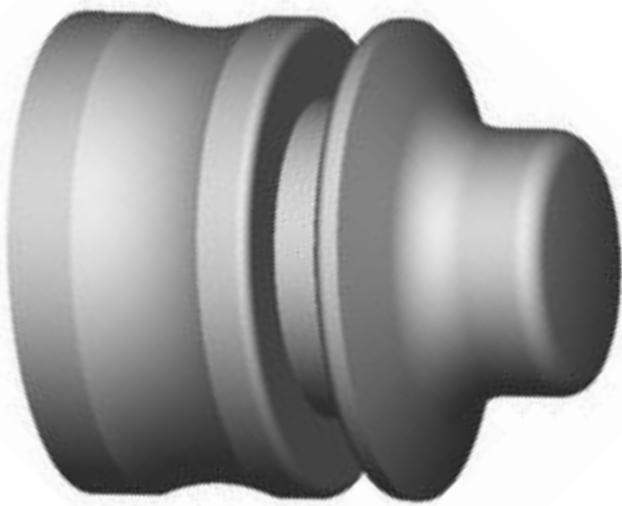


## Section 3

### Turning cycles

Notes

An example program will be created using the following workpiece:

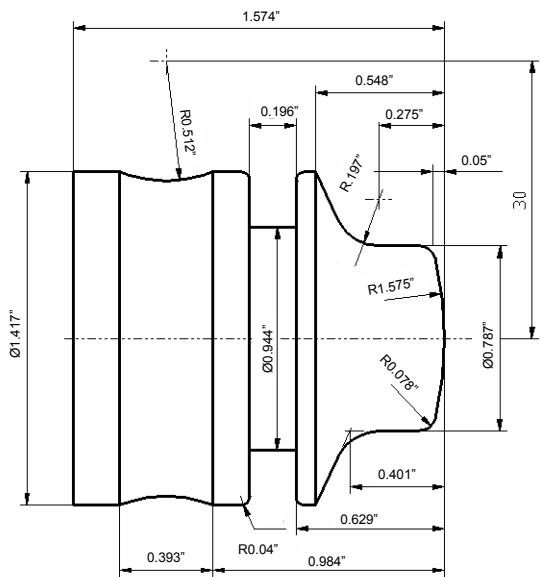


## Section 3

## Turning cycles

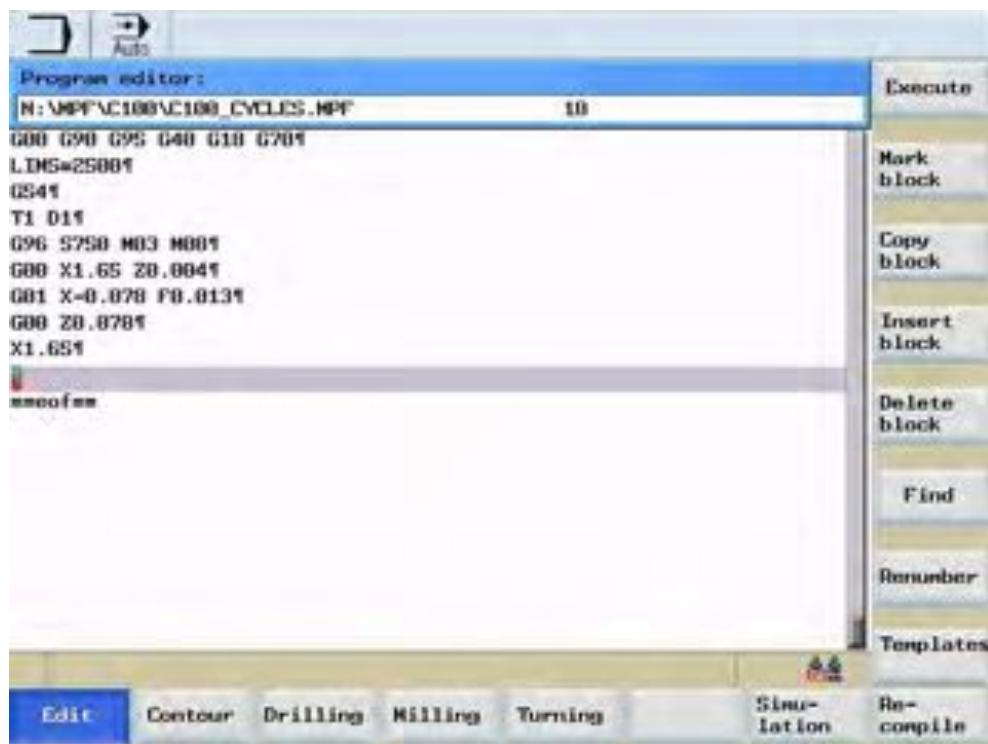
## Notes

You are given the task to rough face and rough turn .



We would create a basic program that will face the component and then add the rough turning cycle.

First of all we will create the basic start of the program, as shown below.

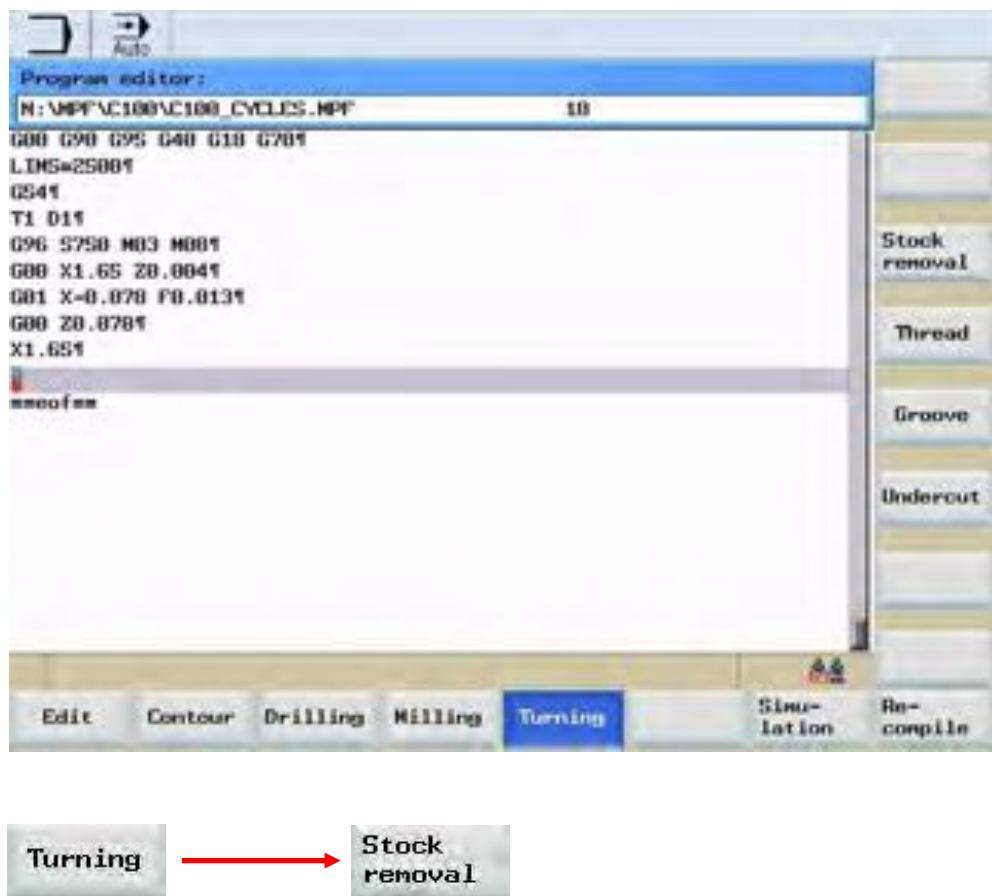


## Section 3

### Turning cycles

Notes

Then we will add the stock removal cycle, as shown in the following sequence.



The most commonly used turning cycle, is the stock removal cycle. But for this cycle to work correctly we must tell the cycle what the geometry description of the turned part is.

There are two methods to use.

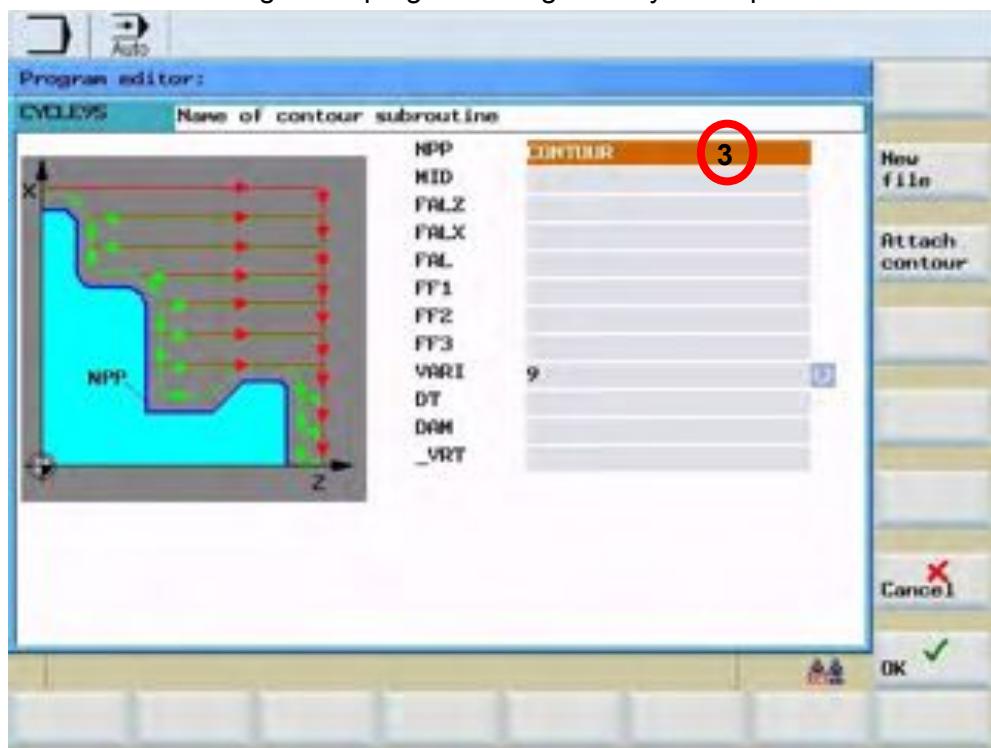
1. Contour description is stored as a sub program
2. Contour description is stored at the bottom of the main program, after the M30 code.

## Section 3

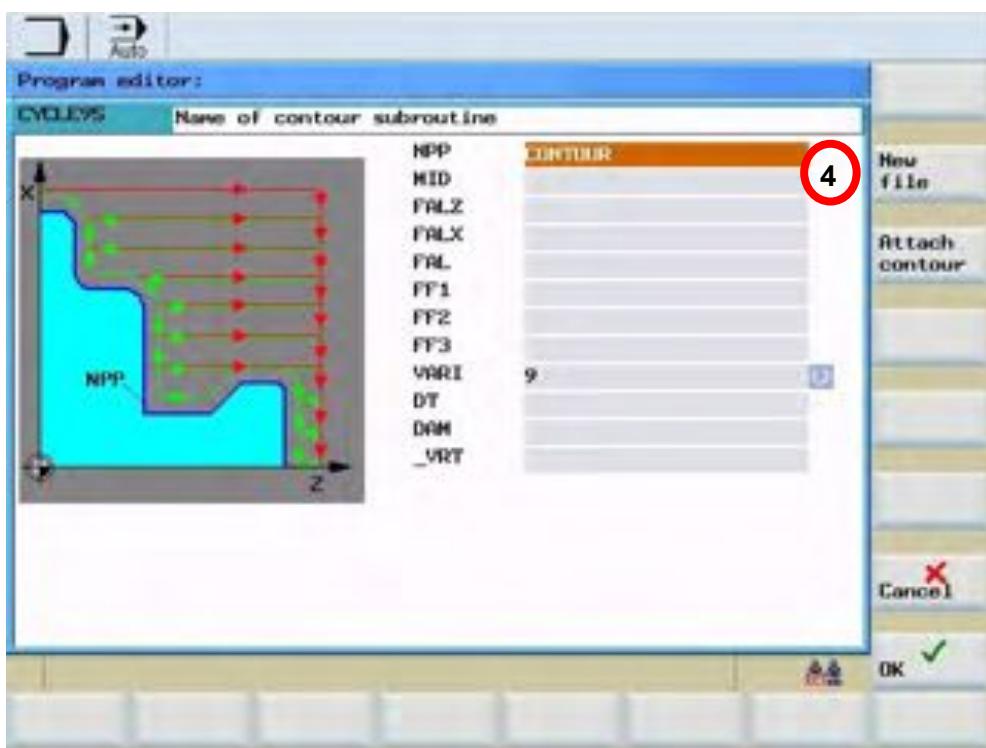
### Turning cycles

Method 1. Creating a sub program with geometry description

Notes



- 3 Type in the name of the contour so that you recognize the name for future reference, followed by:

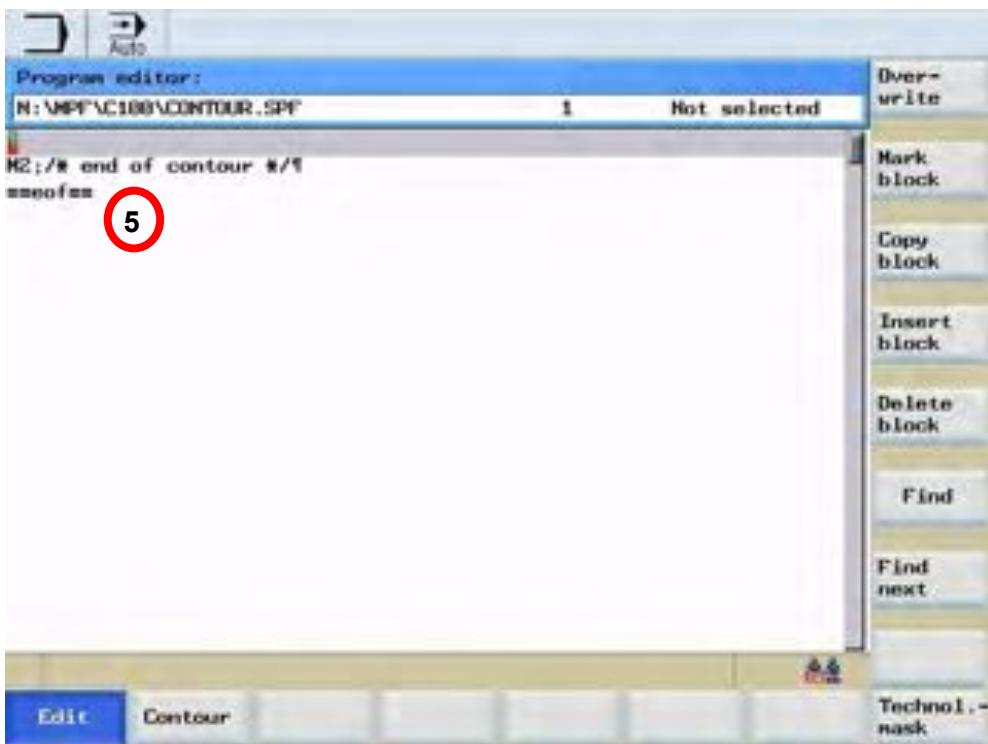


- 4 Followed by the softkey:

New  
file

## Section 3

### Turning cycles



Notes

- 5 The control automatically creates a sub program with the name that you have entered in NPP, with the cursor above the M02 code.

In the sub program, we define the contour description of the finished part, as a single pass.

We must start the contour with an X and Z coordinate.

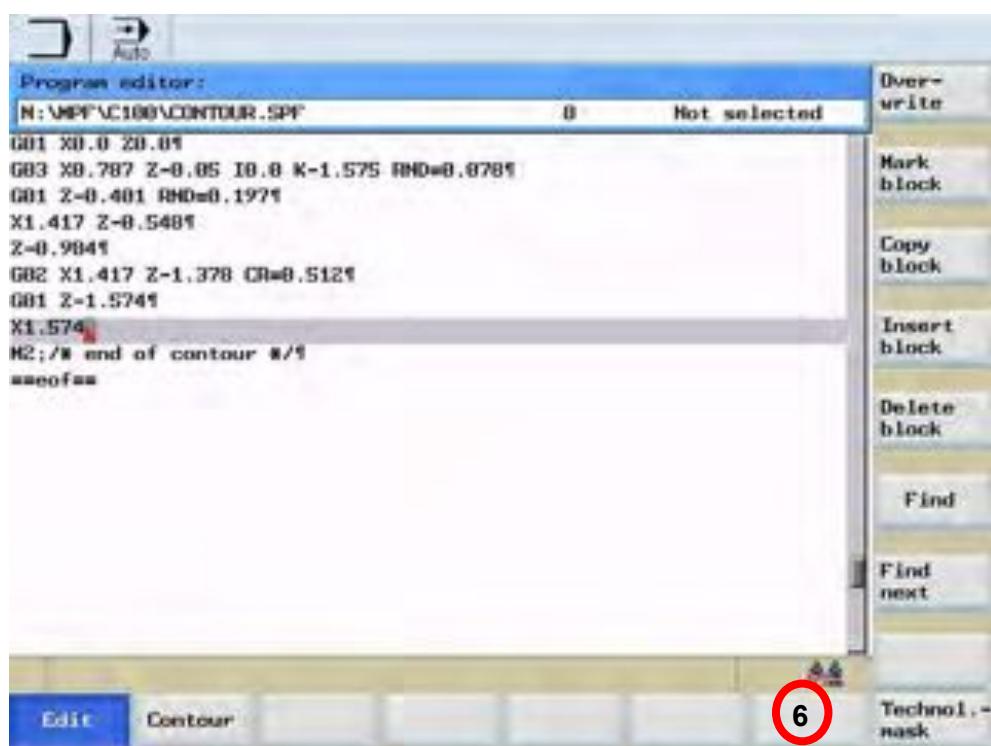
```
N10 G01 X0.0 Z0.0
N20 G03 X0.787 Z-0.05 I0 K-1.575 RND=0.078
N30 G01 Z-0.401 RND=0.197
N40 X1.417 Z-0.548
N50 Z-0.984
N60 G02 X1.417 Z-1.378 CR=0.512
N70 G01 Z-1.574
N80 X1.574
```

Now you can enter the values into the contour sub program.

## Section 3

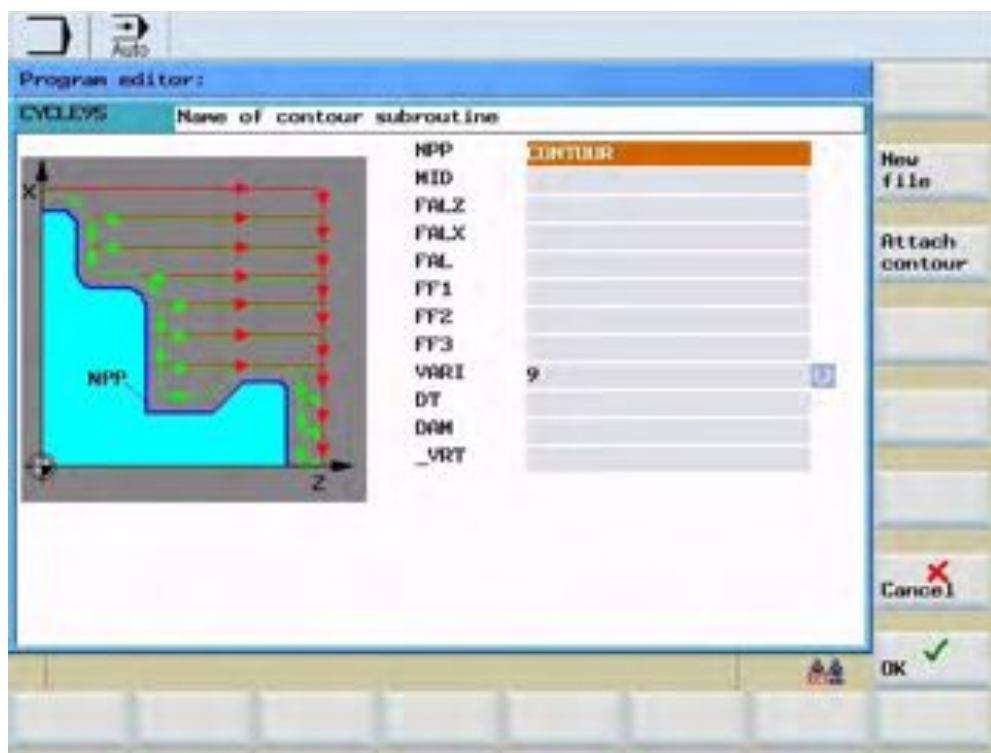
### Turning cycles

Notes



- 6 Once you have entered the description of the contour, use the following sequence to continue.

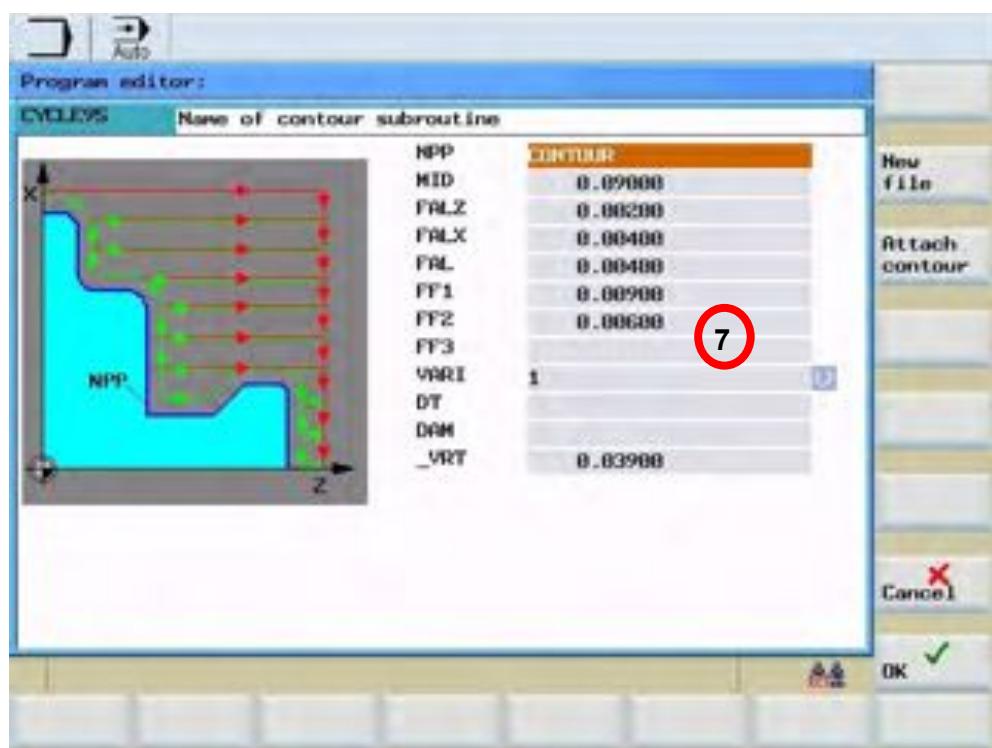
Technol.-mask



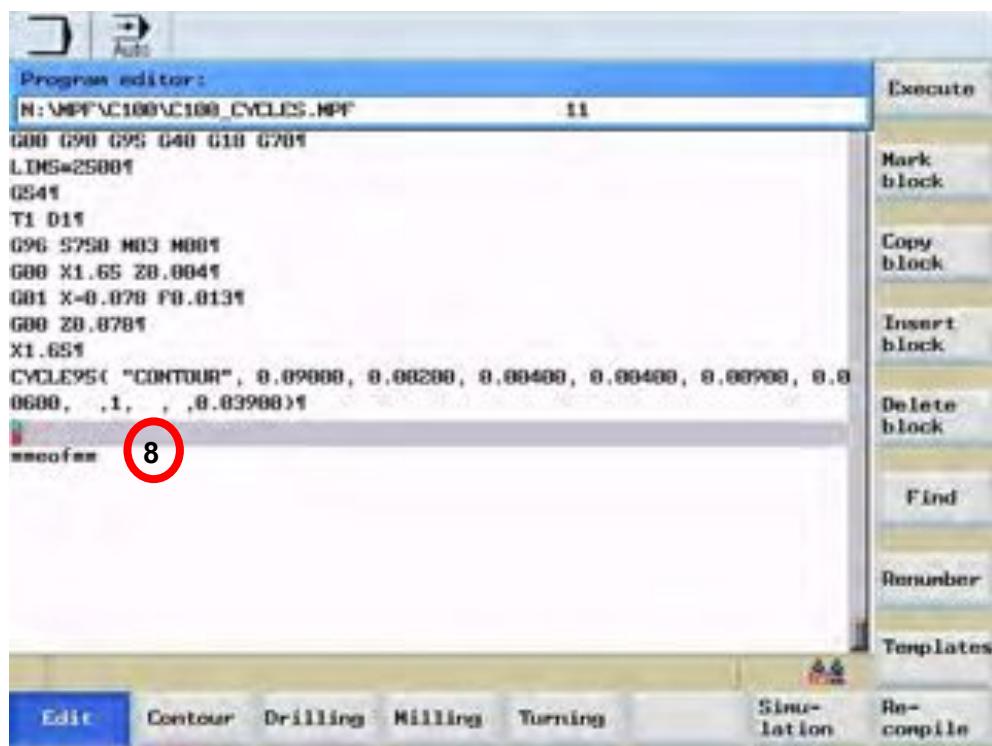
## Section 3

### Turning cycles

Notes



- 7 This is the minimum data to be entered for the stock removal cycle, this can be taken from the drawing. To continue with the following sequence.



- 8 Note Position of the cursor ready for you to continue typing the rest of the program.

## Section 3

### Turning cycles

All data must now be entered, as how to machine the workpiece to the contour.

Notes

#### Cutting with relief cut - CYCLE95

Programming

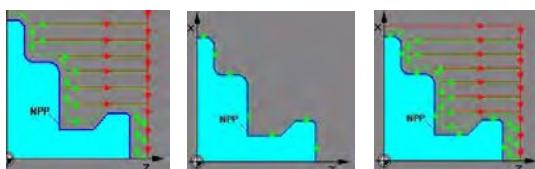
CYCLE95(NPP,MID,FALZ,FALX,FAL,FF1,FF2,FF3,VARI,DT,DAM,\_VRT)

Parameters

NPP	name of contour
MID	in feed depth of cut
FALZ	allowance left on contour for finishing (Z axis)
FALX	allowance left on contour for finishing (X axis)



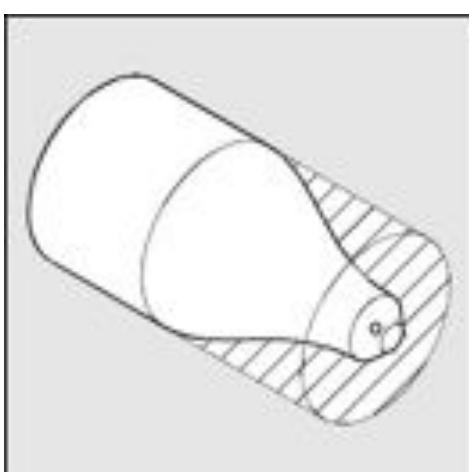
FAL	allowance left on contour profile
FF1	feedrate for roughing without undercut
FF2	feedrate for inserting into relief cut
FF3	feedrate for finishing
VARI	machining type Type 1-4 roughing cycles Type 5-8 finishing cycles Type 9-12 complete machine cycles



DT	dwell time for chip breaking when roughing
DAM	path length after which each roughing step is interrupted for chip breaking



_VRT	retraction distance from contour when roughing
------	--

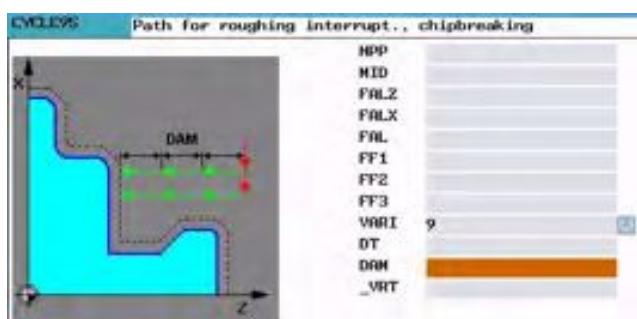
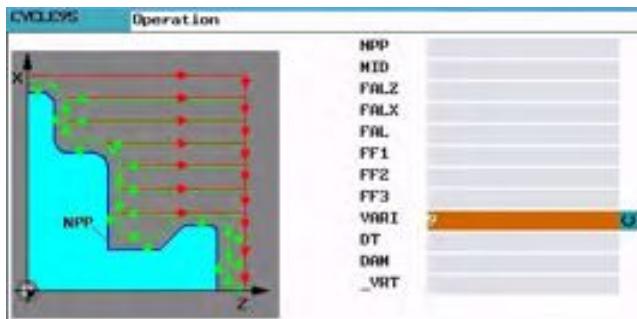
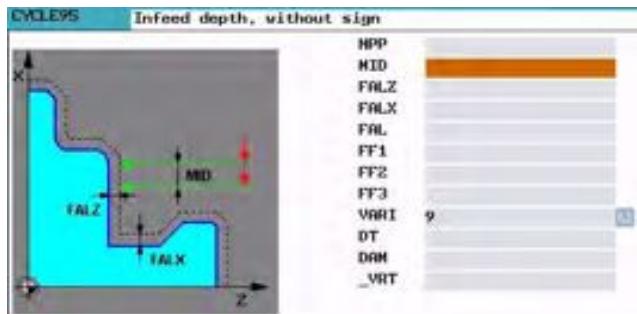


## Section 3

### Turning cycles

Notes

If you look below, as you cursor down the parameter windows you will get a prompt with text, which relates to the picture guide.



## Section 3

### Turning cycles

Notes

If you run simulation using the following sequence this is what is shown.

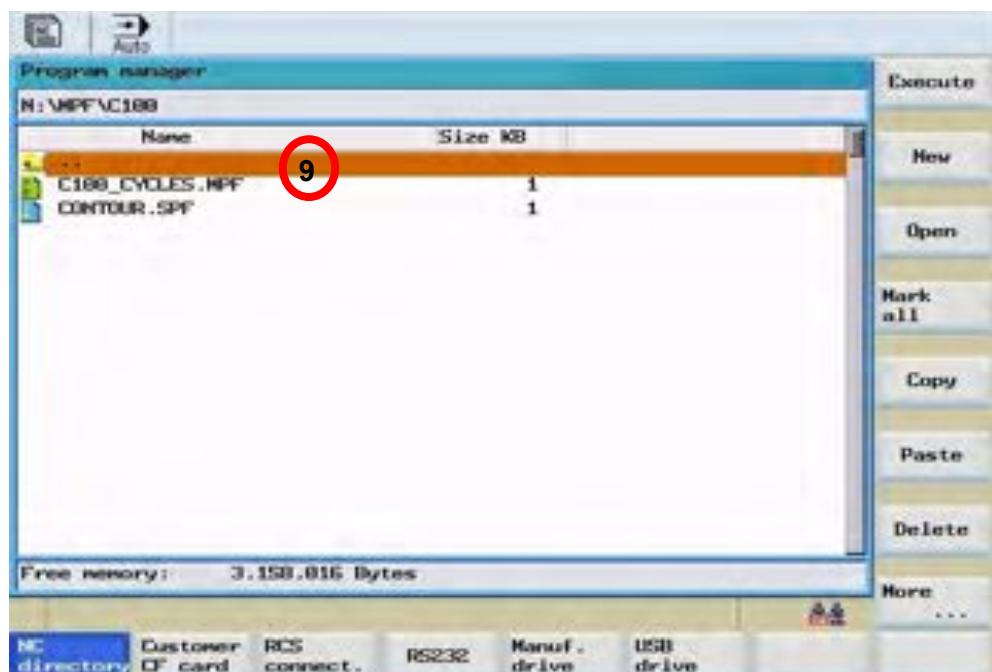


Edit

When you have finished use this sequence to show the NC program in the Editor.

If you look in program manager you will see the sub program CONTOUR.SPF

PROGRAM  
MANAGER



9

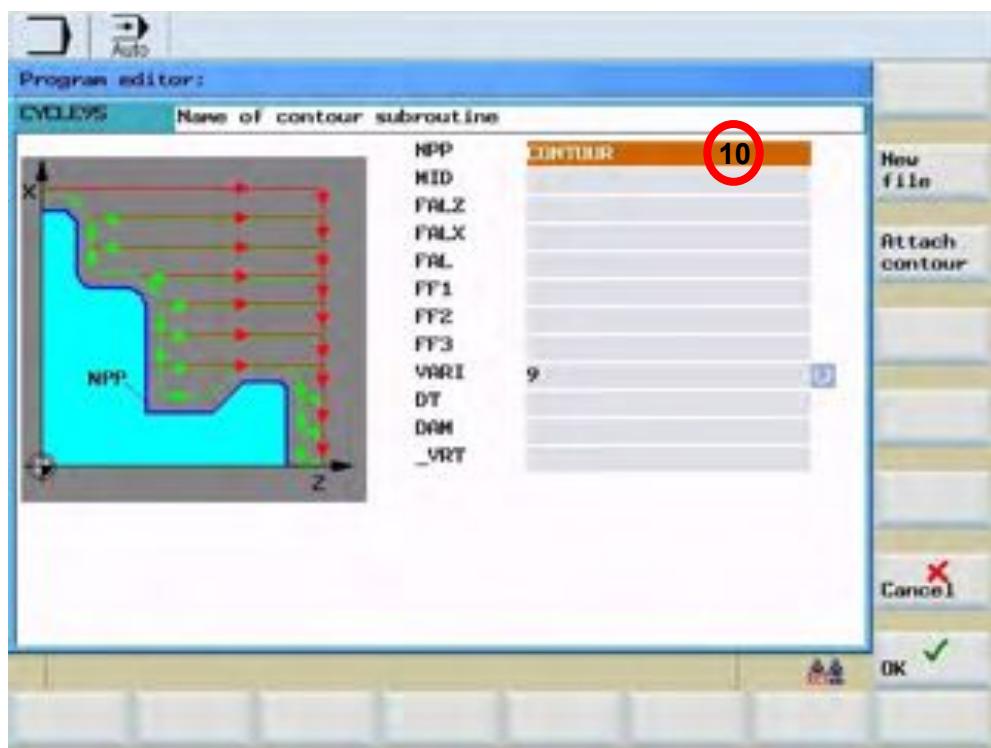
This screen shows the sub program that is being used by the main program.

## Section 3

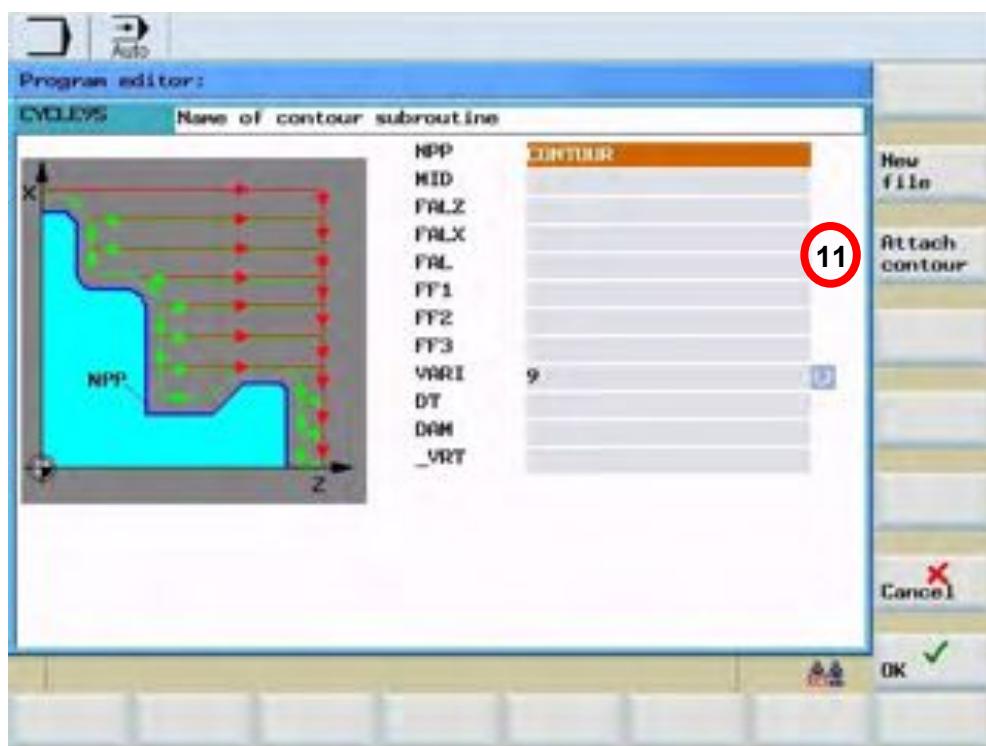
### Turning cycles

Method 2. Creating a program including geometry

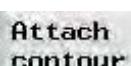
Notes



- 10 Type in the name of the contour so that you recognize the name for future reference, followed by:



- 11 Followed by the softkey:

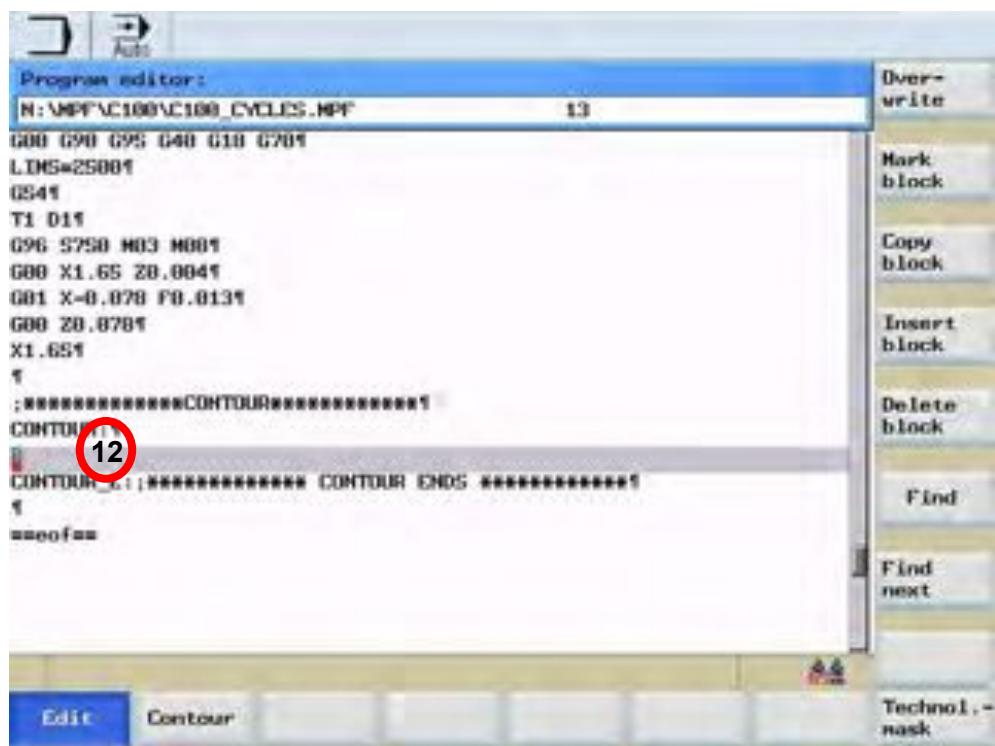


## Section 3

### Turning cycles

Notes

The control now creates an area at the bottom of your program to type your geometry description: Note where the cursor is placed.



12

The control automatically creates a description of the contour at the bottom of the NC program using the name given in NPP.

At the bottom of your program, we define the contour description of the finished part, as a single pass.

We must start the contour with an X and Z coordinate.

```
N10 G01 X0.0 Z0.0  
N20 G03 X0.787 Z-0.05 I0 K-1.575 RND=0.078  
N30 G01 Z-0.401 RND=0.197  
N40 X1.417 Z-0.548  
N50 Z-0.984  
N60 G02 X1.417 Z-1.378 CR=0.512  
N70 G01 Z-1.574  
N80 X1.574
```

## Section 3

### Turning cycles

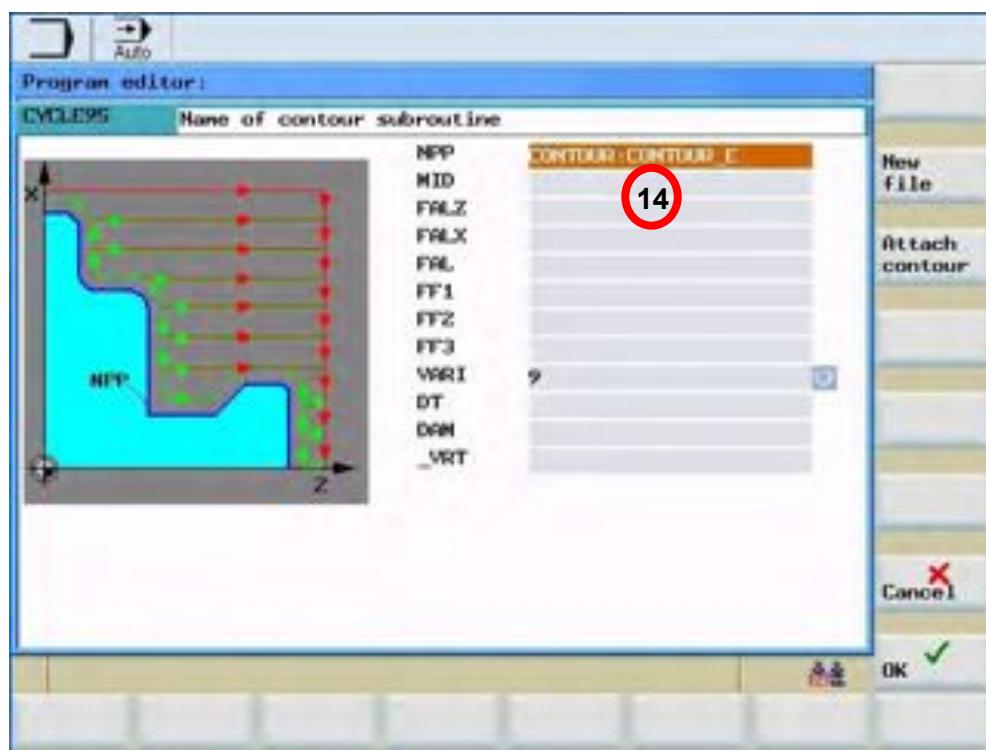
Notes

```
N:\MPF\NC100\NC100_CYCLES.MPF 22  
G241  
T1 D11  
G96 S750 M03 M001  
G00 X1.65 Z0.0045  
G01 X-0.070 F0.0139  
G00 Z0.0781  
X1.651  
!  
;*****CONTOUR*****  
CONTOUR:  
G01 X0.0 Z0.01  
G03 X0.707 Z-0.05 I0.0 Z-1.575 RND=0.0781  
G01 Z-0.401 RND=0.1971  
X1.417 Z-0.5481  
Z-0.9841  
G02 X1.417 Z-1.378 CR=0.5121  
G01 Z-1.5741  
X1.5741  
CONTOUR_E:;***** CONTOUR ENDS *****
```

- 13 Once you have entered the description of the contour, use the following sequence to continue.

Technol.-mask

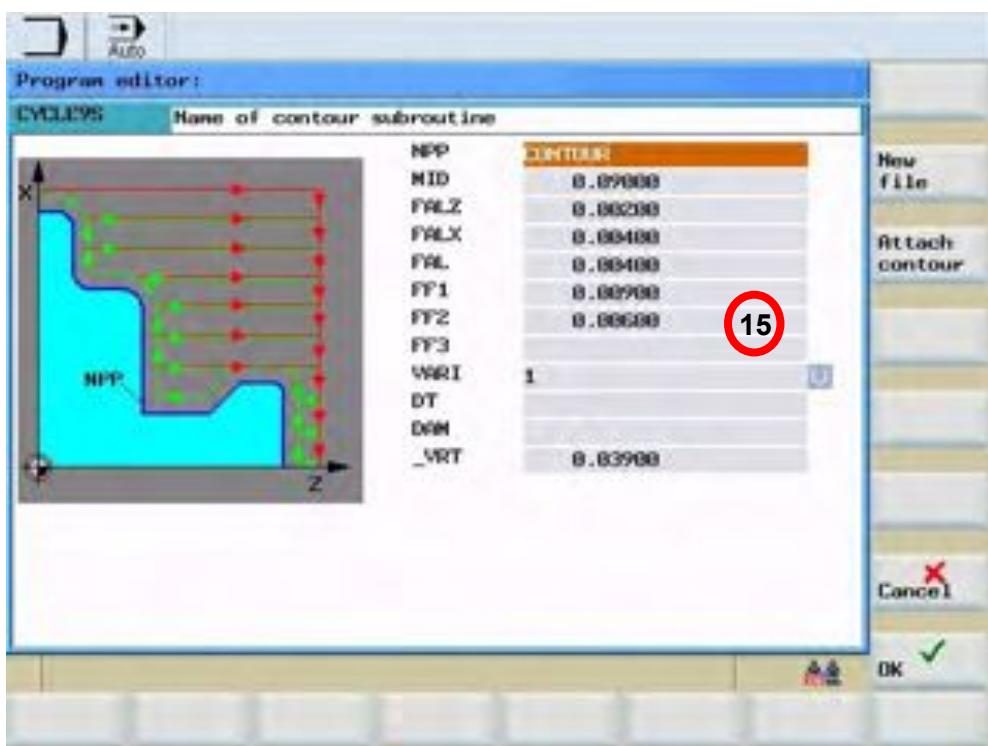
All data must now be entered, as how to machine the workpiece to the contour.



- 14 Note the name of the contour.

## Section 3

### Turning cycles



15

This is the minimum data to be entered for the stock removal cycle, this can be taken from the drawing. To continue with the following sequence.



```
N:\MPFNC100\NC100_CYCLES.MPF          22
0241
T1 D1
G96 S750 M03 M001
G00 X1.65 Z0.0045
G01 X-0.078 F0.0134
G00 Z0.0781
X1.651
;
;*****CONTOUR*****
CONTOUR:1
G01 X0.0 Z0.01
G03 X0.787 Z-0.05 I0.0 Z-1.575 RND=0.0781
G01 Z-0.481 RND=0.1971
X1.417 Z-0.5481
Z-0.9841
G02 X1.417 Z-1.378 CR=0.5121
G01 Z-1.5741
X1.5741
CONTOUR_E:;***** CONTOUR ENDS *****
```

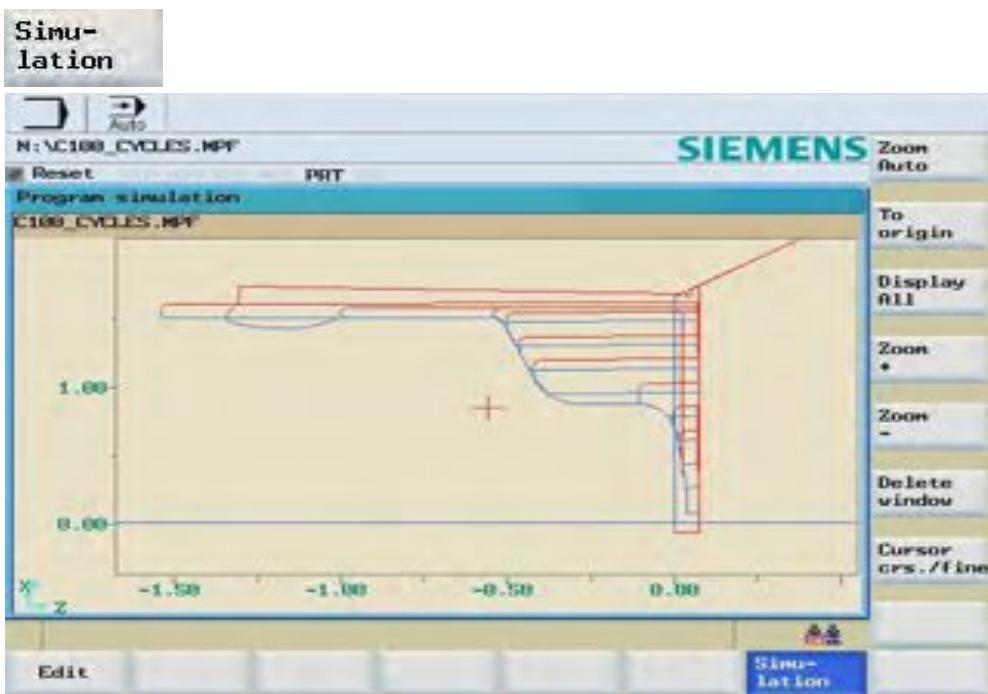
Note position of the cursor ready for you to continue typing the rest of the program.

Notes

## Section 3

### Turning cycles

If you run simulation using the following sequence this is what is shown.

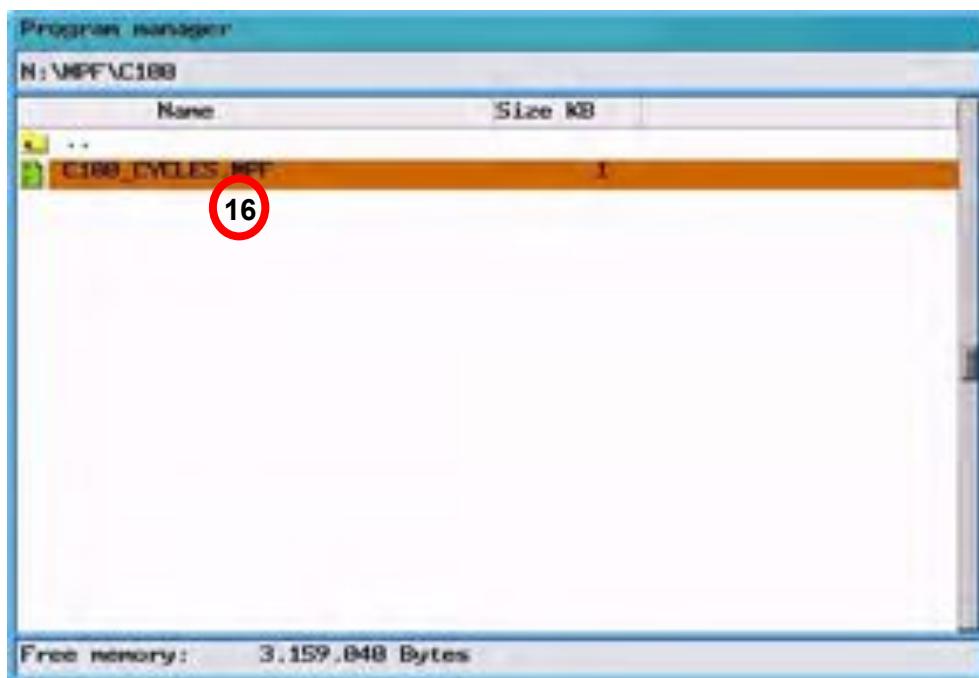


Notes

Edit

When you have finished use this sequence to show the NC program in the Editor.

If you look in program manager you will see the sub program CONTOUR.SPF



16

This screen shows that there is no sub program that is being used by the main program.

## Section 3

### Turning cycles

Notes

To add a finishing tool to the program, first add the tool change position to end of the program to the roughing tool.

**G00 G40 X6.0 Z6.0**

Then add the basic start program for the finishing tool including the finish facing off.

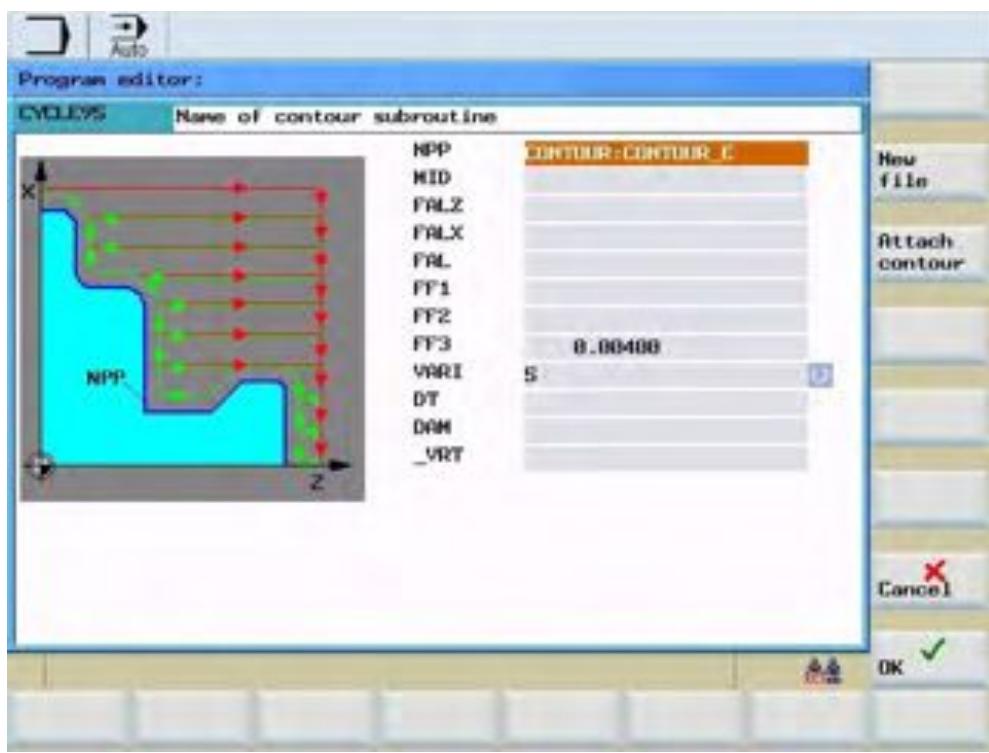
**T2 D1  
G96 S750 M03 M08  
G00 X1.65 Z0.08**

Then follow this sequence.

**Turning**

**Stock removal**

When entering the cycle data for the finish tool you only have to enter three bits of data. Contour name, Finish feedrate, and type of operation



## Section 3

### Turning cycles

Notes

To add a grooving tool to the program, first add the tool change position to end of the program to the roughing tool.

**G00 G40 X6.0 Z6.0**

Then add the basic start program for the grooving tool .

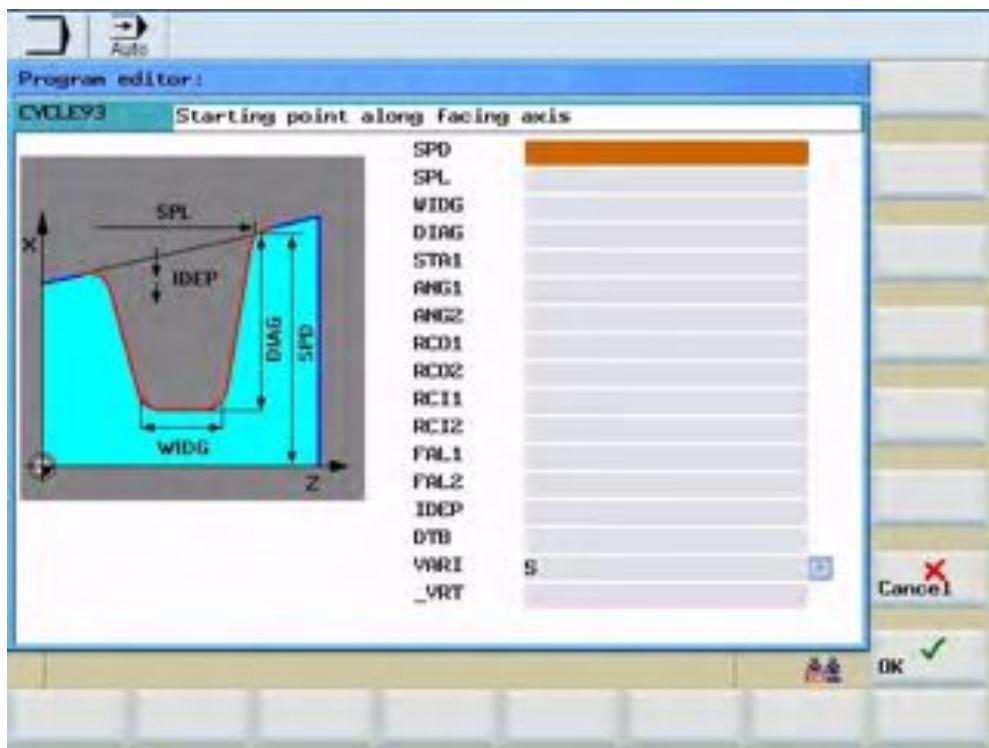
**T3 D1  
G96 S300 M03 M08  
G00 X1.574 Z0.0**

Then follow this sequence.

**Turning**

**Groove**

Enter the cycle data required from data from the drawing.



## Section 3

### Turning cycles

Notes

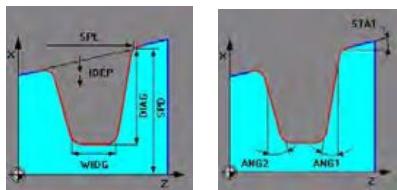
#### Groove – CYCLE93

##### Programming

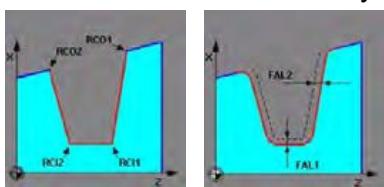
CYCLE93(SPD, SPL, WIDG, DIAG, STA1,ANG1,ANG2,RCO1,RCO2,  
RCI1,RCI2,FAL1,FAL2, IDEP, DTB, VARI, \_VRT)

##### Parameters

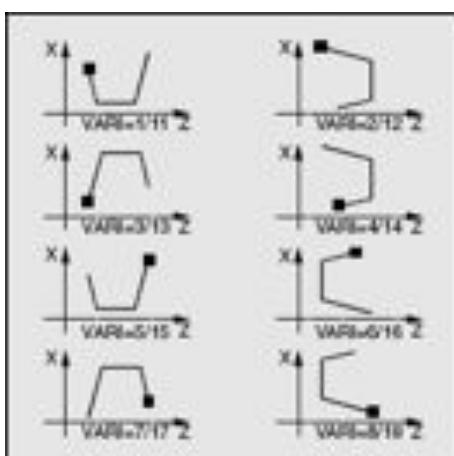
SPD	Starting point in the transverse axis
SPL	Starting point in the longitudinal axis
WIDG	Groove width (enter without sign)
DIAG	Groove depth (enter without sign)



STA1	Angle between contour and longitudinal
ANG1	Flank angle 1: on the groove side determined point
ANG2	Flank angle 2: on the other side
RCO1	Radius/chamfer 1, externally: on the starting point
RCO2	Radius/chamfer 2, externally
RCI1	Radius/chamfer 1, internally: on the starting point
RCI2	Radius/chamfer 2, internally



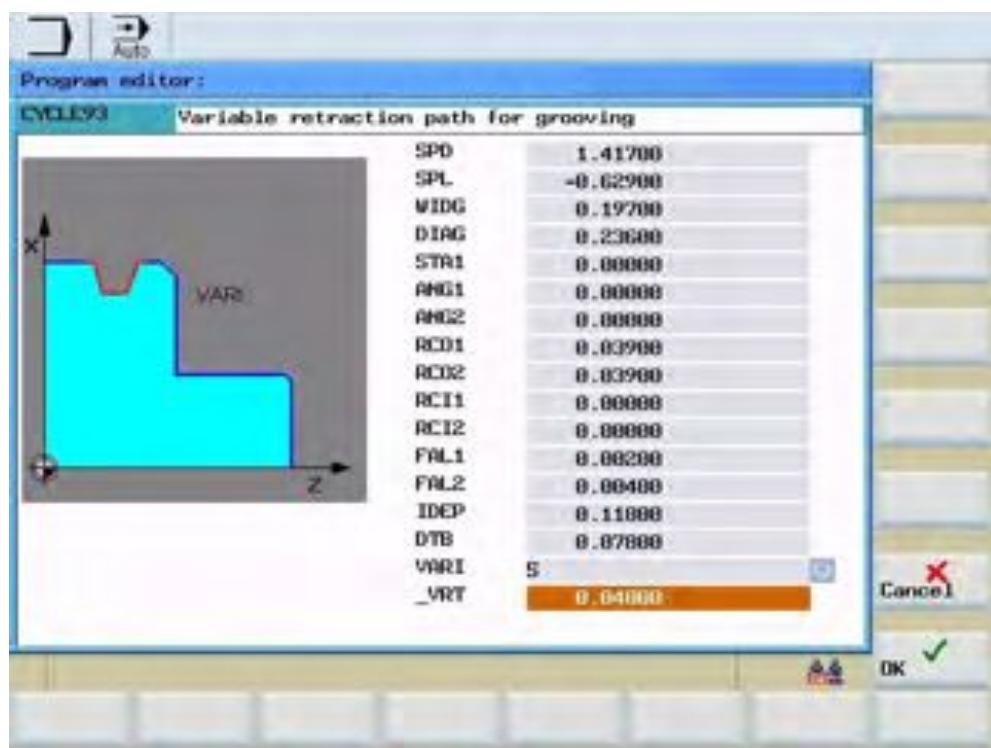
FAL1	Finishing allowance at the recess base
FAL2	Finishing allowance at the flanks
IDEP	Infeed depth (enter without sign)
DTB	Dwell time at recess base
VARI	Machining type
_VRT	Range of values: 1...8 and 11...18 retraction distance



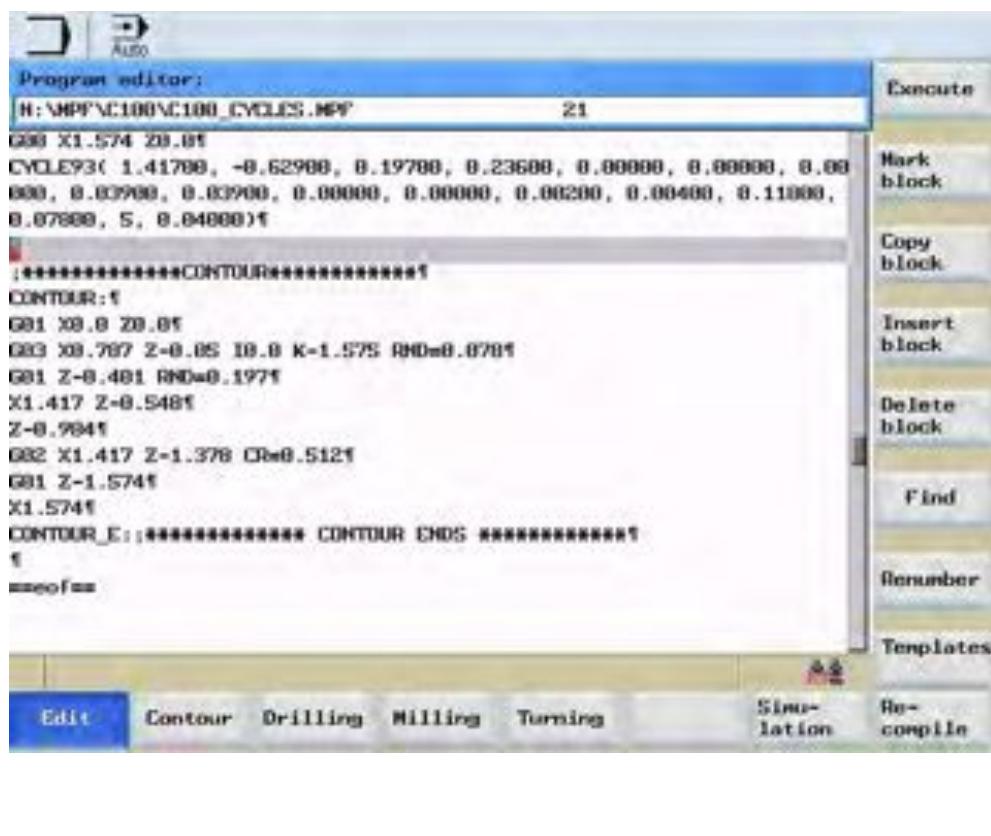
## Section 3

### Turning cycles

Notes



Then follow this sequence.



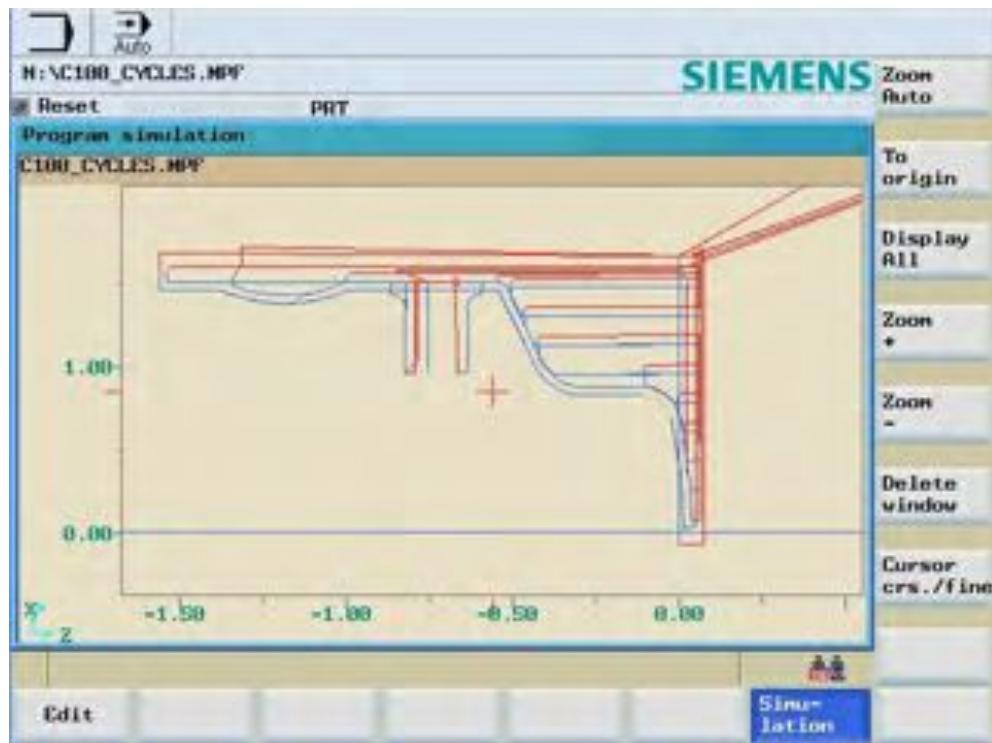
## Section 3

### Turning cycles

Notes

If you run simulation using the following sequence this is what is shown.

Simu-  
lation



Edit

When you have finished use this sequence to show the NC program in the Editor.

## Section 3

### Turning cycles

Your completed program with ROUGH, FINISH turn and GROOVE are shown below:

```
G00 G90 G95 G40 G18 G70
LIMS=2500
G54
T1 D1
G96 S750 M03 M08
G00 X1.65 Z0.004
G01 X-0.078 F0.013
G00 Z0.078
X1.65
CYCLE95( "CONTOUR:CONTOUR_E", 0.098000, 0.002000,
0.004000, 0.004000, 0.009000, 0.006000, ,1, , ,0.039000)
G00 G40 X6.0 Z6.0
T2 D1
G96 S750 M03 M08
G00 X1.65 Z0.08
CYCLE95( "CONTOUR:CONTOUR_E", , , , , ,0.004000, 5, , ,)
G00 G40 X6.0 Z6.0
T3 D1
G96 S300 M03 M08
G00 X1.574 Z0.0
CYCLE93( 1.417000, -0.629000, 0.197000, 0.236000, 0.000000,
0.000000, 0.000000, 0.039000, 0.039000, 0.000000, 0.000000,
0.002000, 0.004000, 0.118000, 0.078000, 5)
G00 G40 X6.0 Z6.0
M30
*****CONTOUR*****
CONTOUR:
N10 G01 X0.0 Z0.0
N20 G03 X0.787 Z-0.05 I0 K-1.575 RND=0.078
N30 G01 Z-0.401 RND=0.197
N40 X1.417 Z-0.548
N50 Z-0.984
N60 G02 X1.417 Z-1.378 CR=0.512
N70 G01 Z-1.574
N80 X1.574
CONTOUR_E:;***** CONTOUR ENDS *****
```

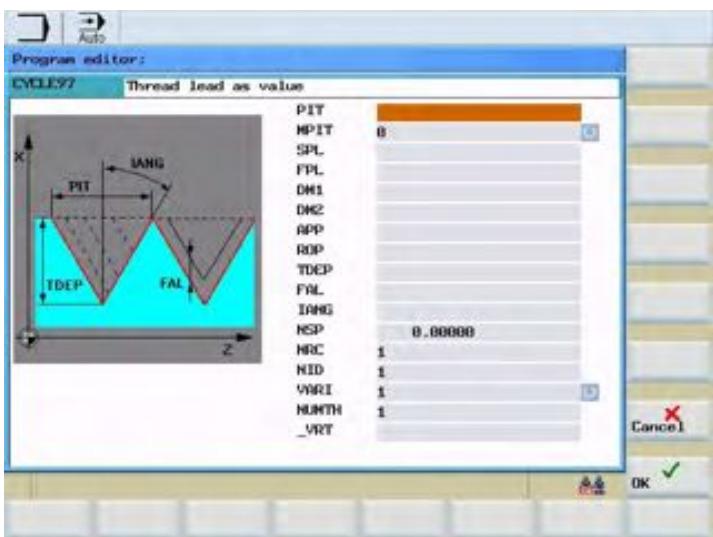
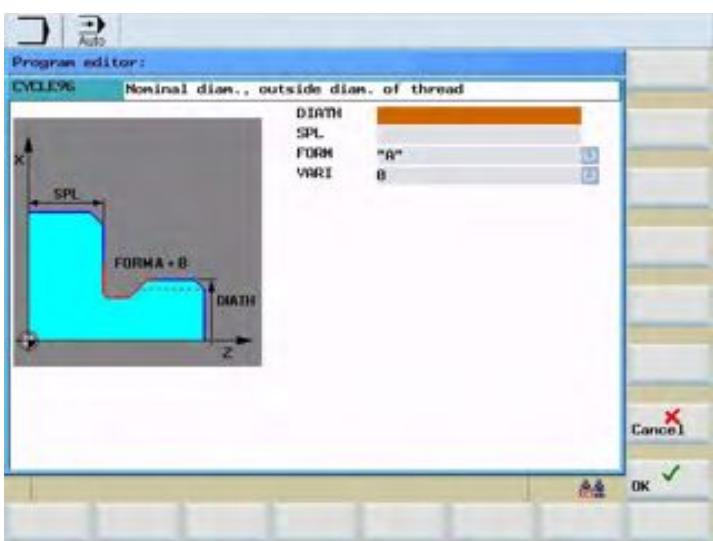
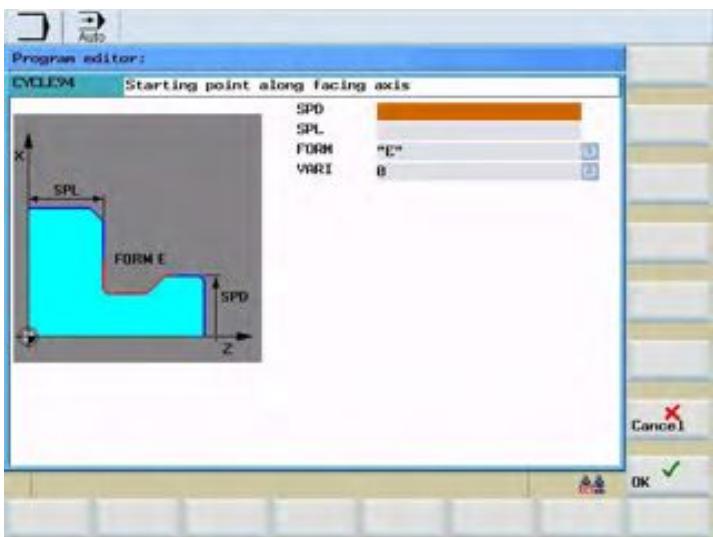
Notes

## Section 3

### Turning cycles

The three remaining cycles are for UNDERCUTTING, THREAD UNDERCUTTING, and THREADING.  
They all require data that is taken from the component drawing.

Notes



## Section 3

### Turning cycles

Notes

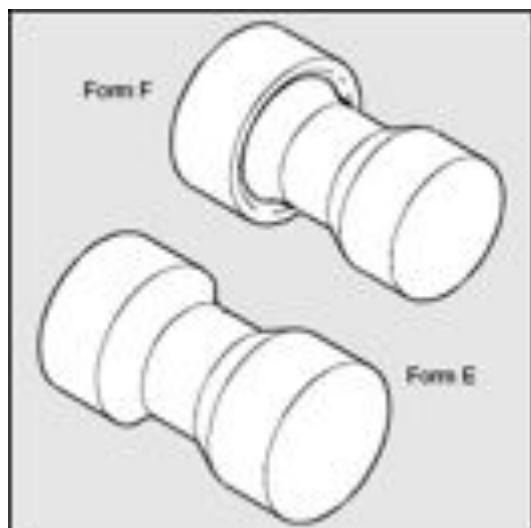
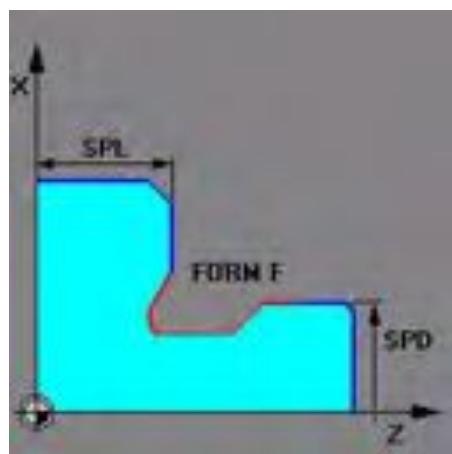
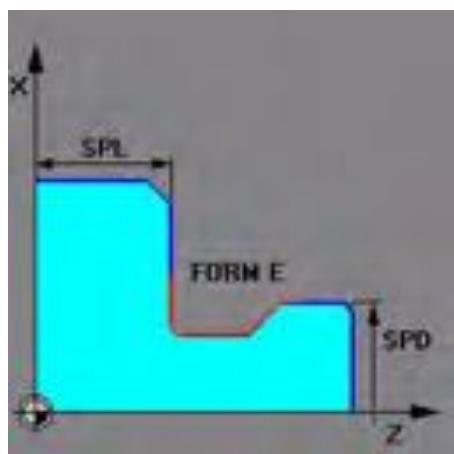
#### Undercut (forms E and F to DIN) – CYCLE94

##### Programming

CYCLE94(SPD, SPL, FORM, VARI)

##### Parameters

SPD	Starting point in the transversal axis
SPL	Starting point of the tool compensation
FORM	Definition of the form Values: E (for form E) F (for form F)
VARI	Determination of type of the machining for the Range of values: 1 ... 4



## Section 3

### Turning cycles

Notes

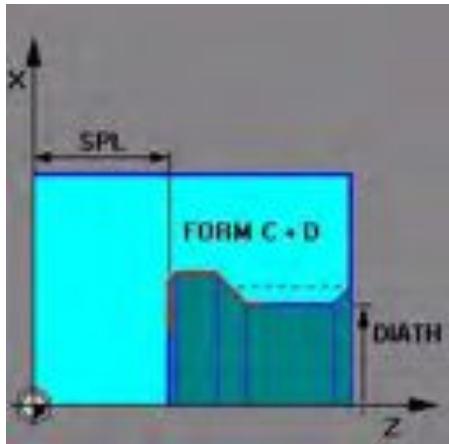
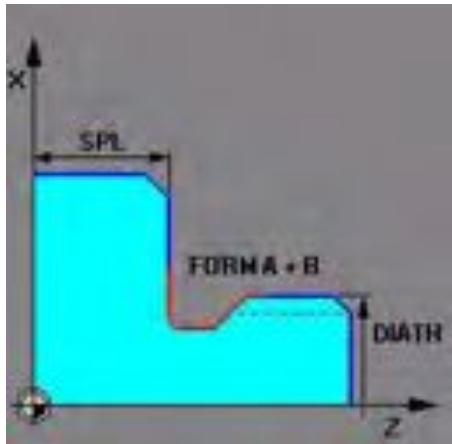
#### Thread undercut – CYCLE96

##### Programming

CYCLE96 (DIATH, SPL, FORM, VARI)

##### Parameters

DIATH	Nominal diameter of the thread
SPL	Starting point of the correction in the longitudinal axis
FORM	Definition of the form Values: A (for form A) B (for form B) C (for form C) D (for form D)
VARI	Determination of type of the machining for the Range of values: 1 ... 4



## Section 3

### Turning cycles

Notes

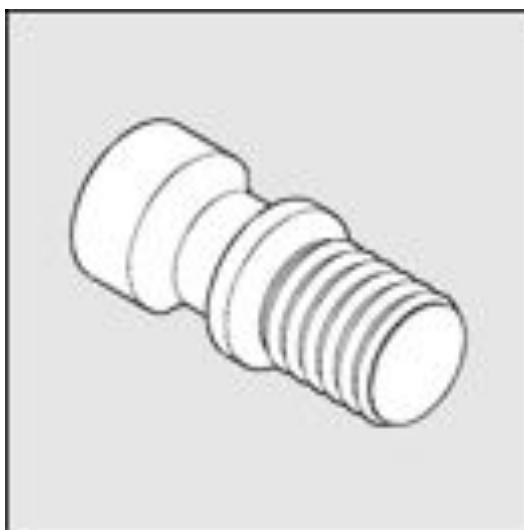
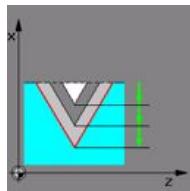
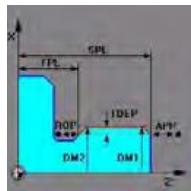
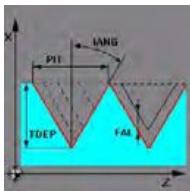
#### Thread cutting – CYCLE97

##### Programming

CYCLE97(PIT, MPIT, SPL, FPL, DM1, DM2, APP, ROP, TDEP, FAL, IANG, NSP, NRC, NID, VARI, NUMT, \_VRT)

##### Parameters

PIT	Thread pitch as a value (enter without sign)
MPIT	Thread pitch as a thread size Range of values: 3 (for M3) ... 60 (for M60)
SPL	Thread starting point in the longitudinal axis
FPL	Thread end point in the longitudinal axis
DM1	Thread diameter at the starting point
DM2	Thread diameter at the end point
APP	Run-in path (enter without sign)
ROP	Run-out path (enter without sign)
TDEP	Thread depth (enter without sign)
FAL	Finishing allowance (enter without sign)
IANG	Infeed angle Range of values: "+" (for flank infeed at the flank) "−" (for alternating flank infeed)
NSP	Starting point offset for the first thread turn
NRC	Number of roughing cuts (enter without sign)
NID	Number of idle passes (enter without sign)
VARI	Determination of type of the machining for the Range of values: 1 ... 4
NUMT	Number of thread turns (enter without sign)
_VRT	retraction distance

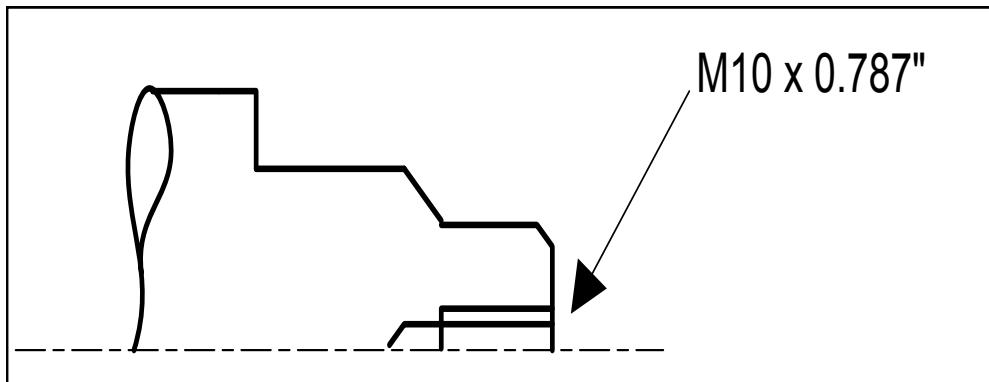


## Section 4

### Drilling cycles

Notes

You are given the task to drill and tap a hole in your work piece.



We would create a basic program to DRILL, and TAP.

First of all we will create the basic start of the program.

The screenshot shows a CNC program editor window. The title bar reads "Program editor: N:\MPF\NC100\NC100\_CYCLES\_DRILL.MPF". The main area displays the following G-code:

```
G00 G90 G95 G40 G17 G701  
L015=25001  
G241  
T6 D11  
G95 S000 F0.01 M03 M001  
G00 X0.0 Z0.0701  
**eof**
```

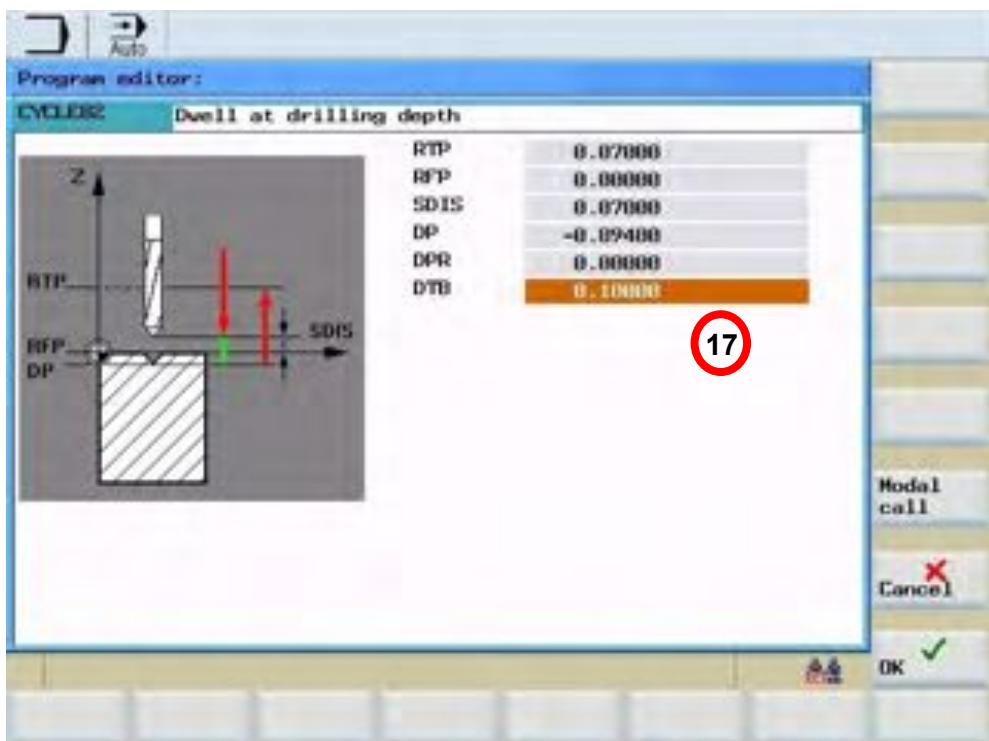
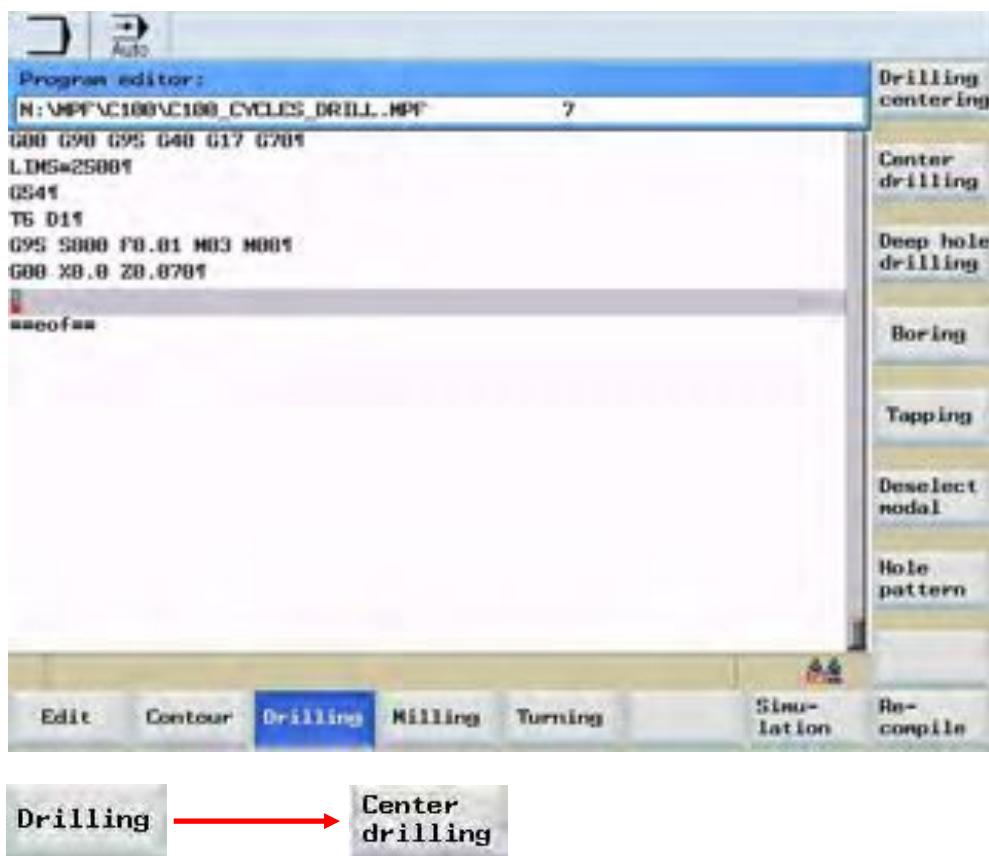
A context menu is open on the right side of the screen, listing options: Execute, Mark block, Copy block, Insert block, Delete block, Find, Renumber, and Templates. The "Copy block" option is highlighted.

## Section 4

### Drilling cycles

Notes

Then we will add the drilling cycle, as shown in the following sequence.

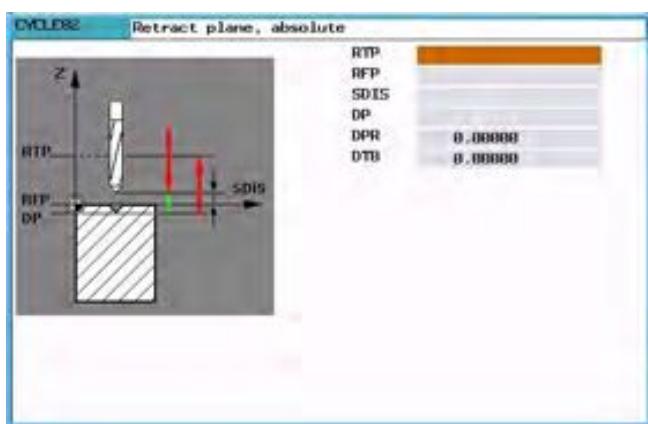


- 17 This is the data to be entered for the drill cycle, this can be taken from the drawing. To continue with the following sequence.



## Section 4

### Drilling cycles



Notes

Drilling , centering - CYCLE82

Programming

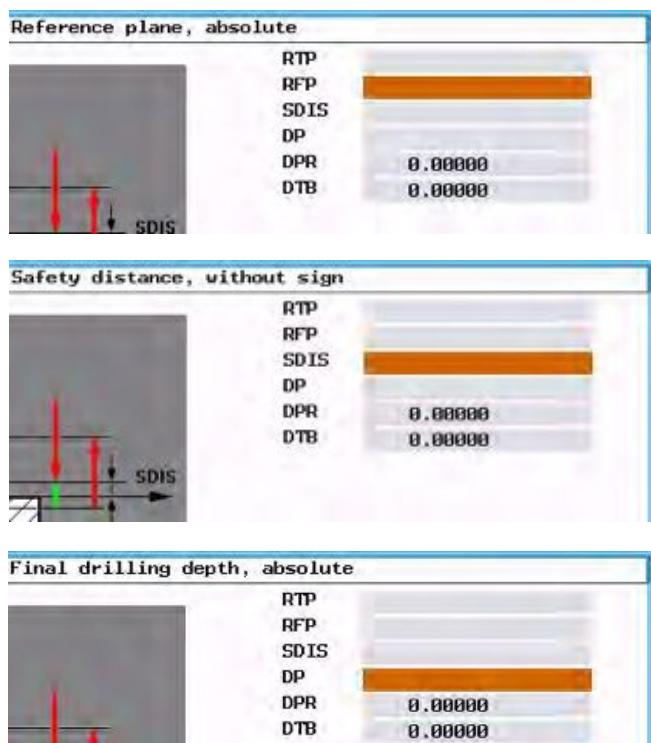
CYCLE82(RTP, RFP, SDIS, DP, DRP, DTB)

Parameters

RTP	retraction plane (absolute)
RFP	reference plane (absolute)
SDIS	safety plane (enter without sign)
DP	final drilling depth (absolute)
DPR	final drilling depth relative to the reference plane (enter without sign)
DTB	dwell time at final depth (chip breaking)

These six parameters are used through out all the drilling cycles, but as more functionality is required as are the number of parameters increased.

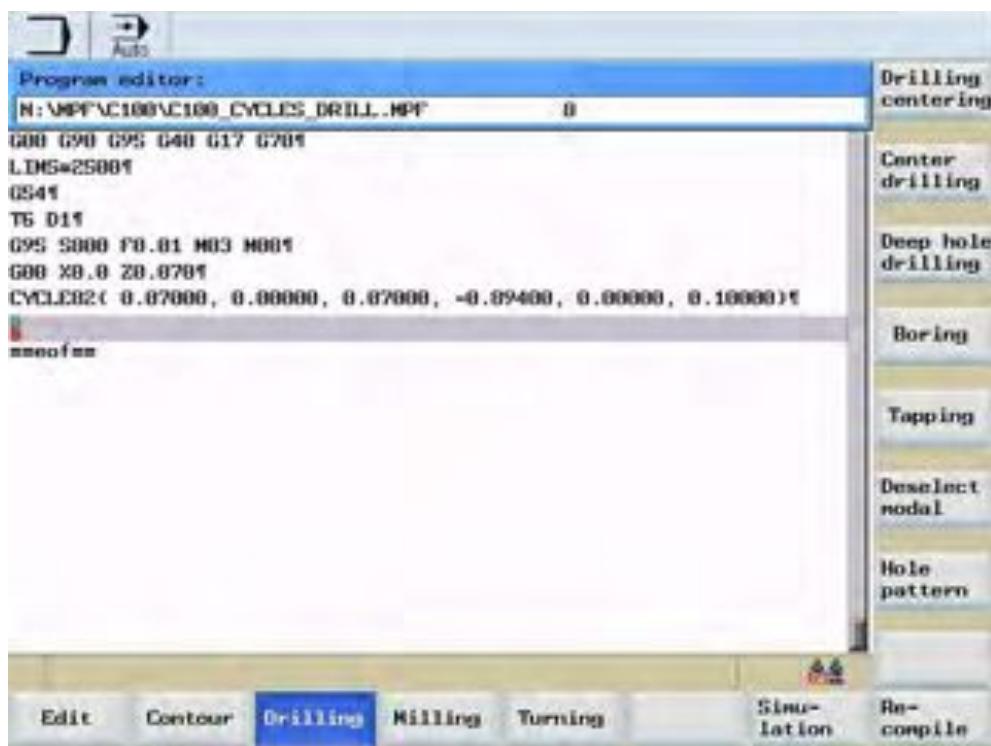
If you look below, as you cursor down the parameter windows you will get a prompt with text, which relates to the picture guide.



## Section 4

### Drilling cycles

Notes



Your program should now look as above.  
We have now created a program to drill the hole.

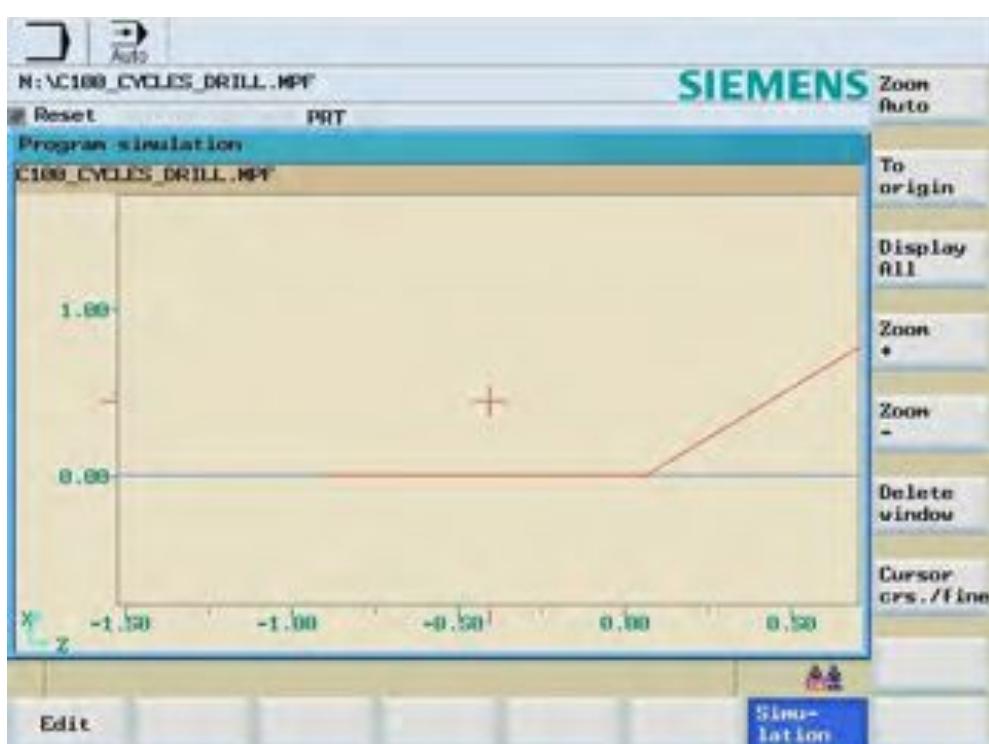
To check the program follow this sequence.



This allows the simulation to show the correct working plane for drilling.

## Section 4

### Drilling cycles



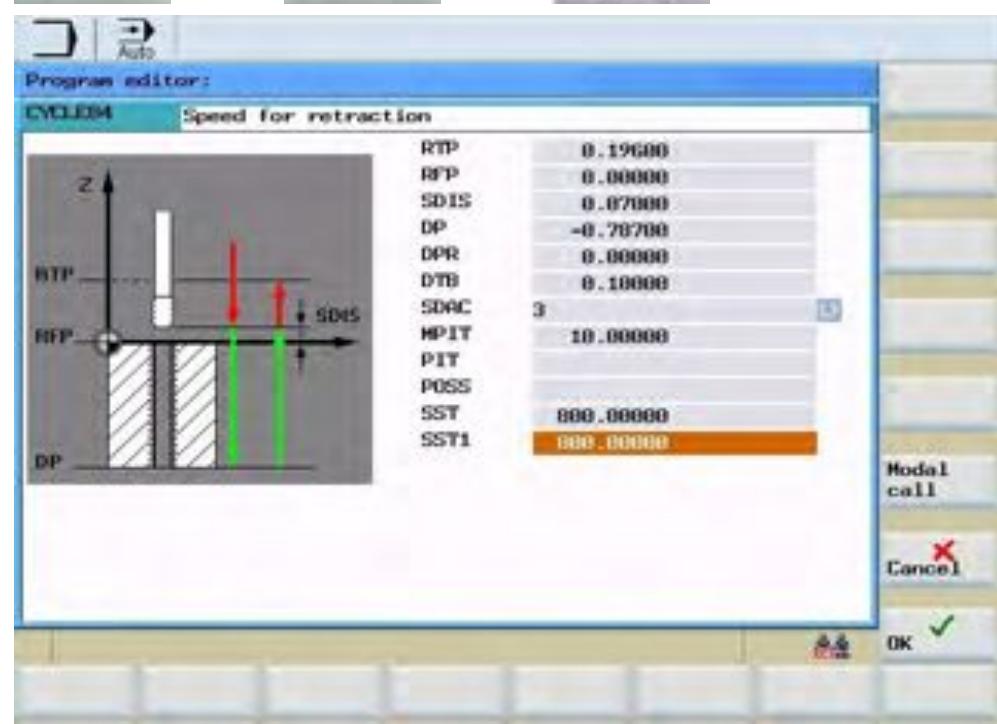
Notes

Edit

See if you can now create the rest of the program to tap the hole.

To create the rest of the program, you will have to use cycle84.

Drilling → Tapping → Rigid tapping

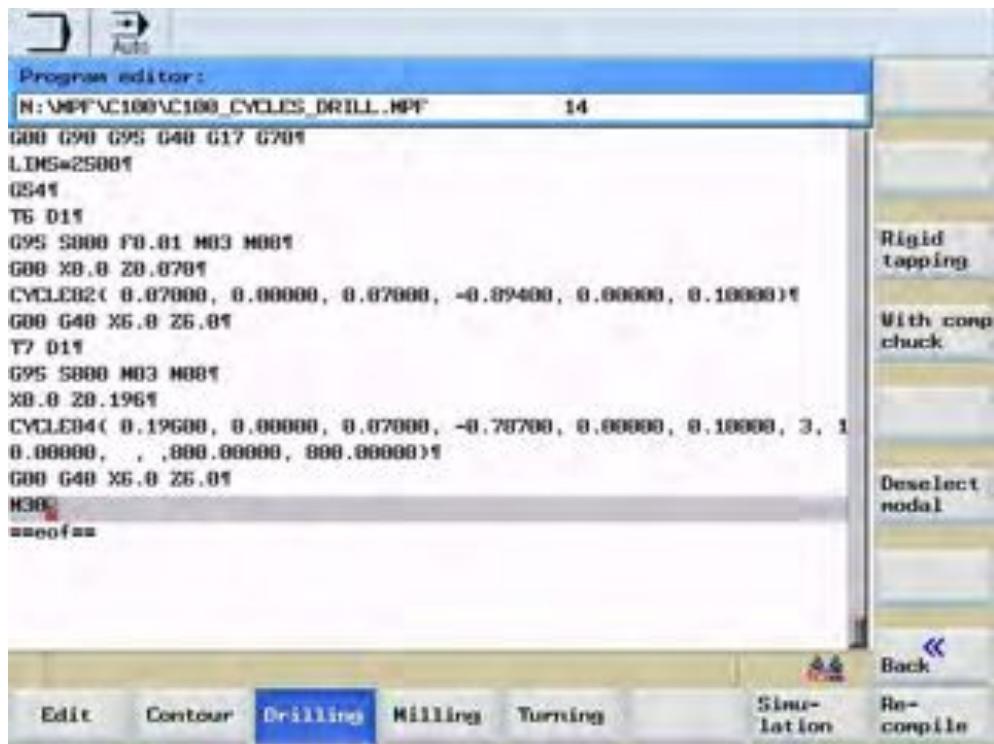


## Section 4

### Drilling cycles

Notes

Final program: For Drilling and Tapping.



**G00 G90 G95 G40 G17 G70**  
LIMS=2500  
G54  
T6 D1  
G95 S800 F0.01 M03 M08  
G00 X0.0 Z0.078  
CYCLE82( 0.078000, 0.000000, 0.078000, -0.894000, 0.000000,  
0.500000)  
G00 G40 X6.0 Z6.0  
T7 D1  
G95 S800 M03 M08  
G00 X0.0 Z-0.196  
CYCLE84( 0.196000, 0.000000, 0.078000, -0.787000, 0.000000,  
0.100000, 3, 10.000000, , ,800.000000, 800.000000)  
G00 G40 X6.0 Z6.0  
M30

## 1. Description

### Module objective:

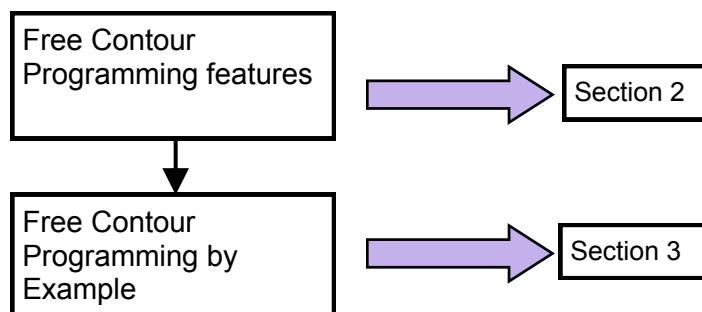
Upon completion of this module you use Free Contour Programming, by example.

### Module description:

Free contour programming is a support tool for the editor. The contour programming function enables you to create simple and complex contours.

### Module content:

Free Contour Programming Features  
Free Contour Programming by Example



## Section 2

### Free Contour Programming Features

Notes

#### Functionality

Free contour programming is a support tool for the editor. The contour programming function enables you to create simple and complex contours. An integrated contour calculator (geometry processor) calculates any missing parameters for you, provided that they can be calculated from other parameters. You can link together contour elements. Contour transition elements "radius" and "chamfer" are also provided. The programmed contours are transferred to the edited part program.

#### Contour elements

The following are contour elements:

Start point



Straight line in the vertical direction



Straight line horizontal in the direction



Oblique straight line



Circular arc

## Section 2

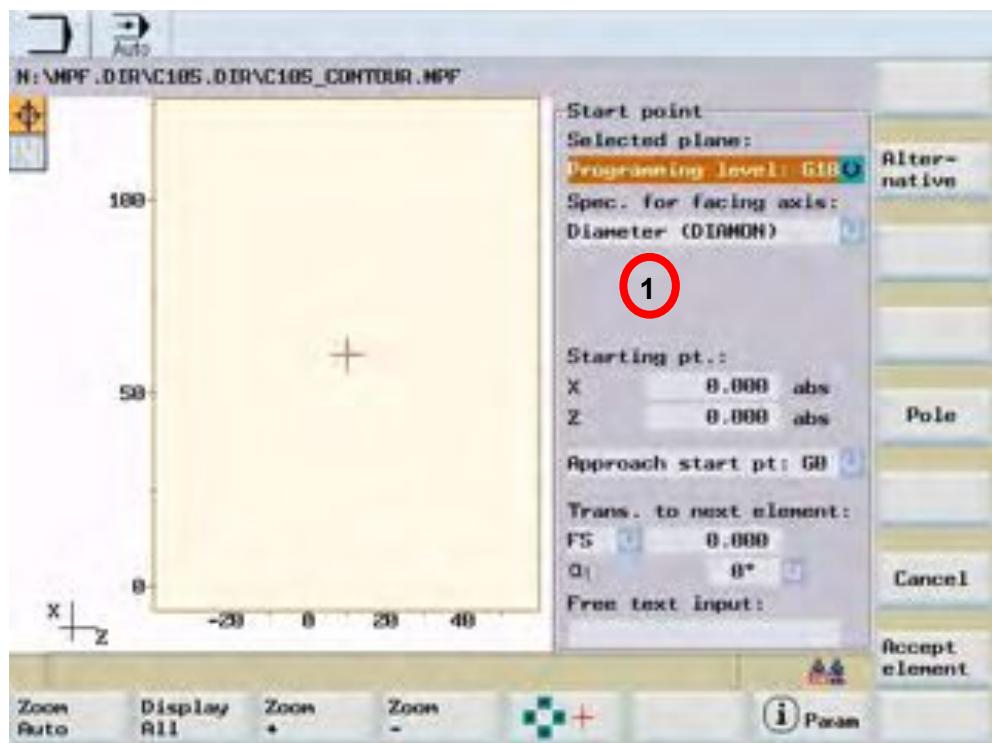
### Free Contour Programming Features

Notes

#### Define a start point

When entering a contour, begin at a position which you already know and enter it as the starting point. The sequence of operations for defining the start point of a contour is as follows.

You have opened a part program and selected soft key “Contour” to program a new contour. The input screen for specifying the start of the contour is displayed as below:



1

As you cursor down the “start point” field they will be highlighted in a dark brown field.

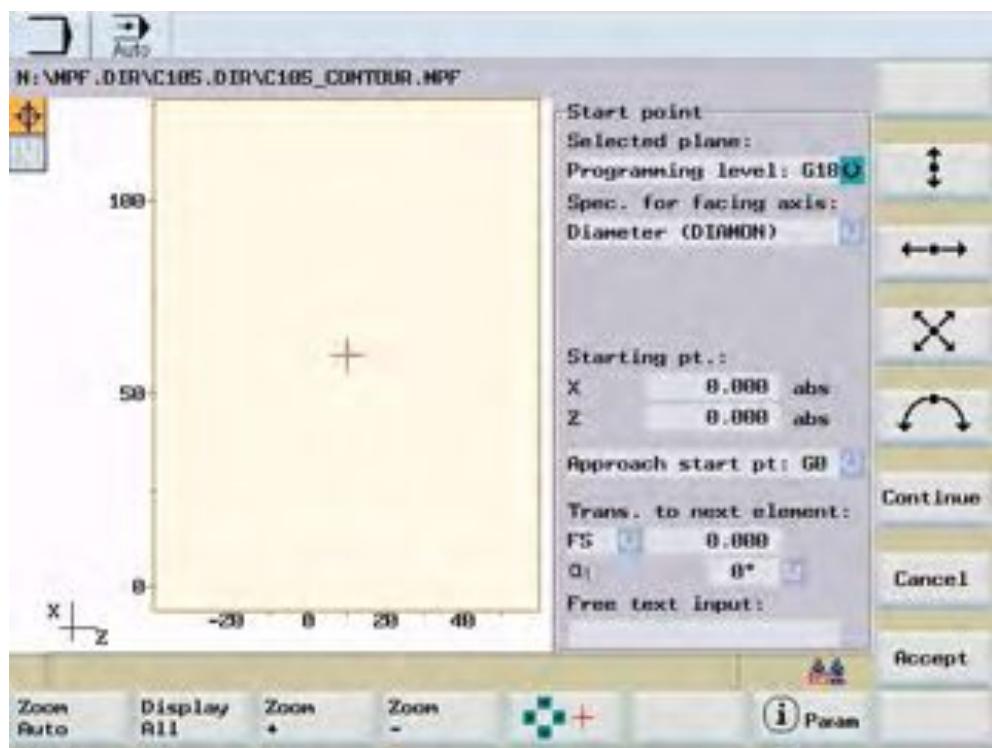
## Section 2

### Free Contour Programming Features

Notes

#### Soft keys and parameters.

Once you have defined the contour start point, you can begin programming the individual contour elements from the main screen below:



#### Vertical soft keys

The following contour elements are available for programming contours:



Straight line in the vertical direction.



Straight line in the horizontal direction.



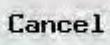
Oblique line in the X/Y direction. Enter the end point of the line using coordinates or an angle.



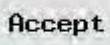
Arc with any direction of rotation.

**Continue**

The “Continue” soft key accesses the “Pole” sub screen softkey.



By selecting the “Cancel” soft key you can return to the main screen without transferring the last edited values.



When you select “Accept” soft key you close the contour screen and return to the program editor.

## Section 2

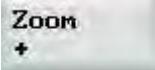
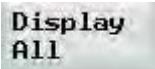
### Free Contour Programming Features

Notes

#### Horizontal soft keys



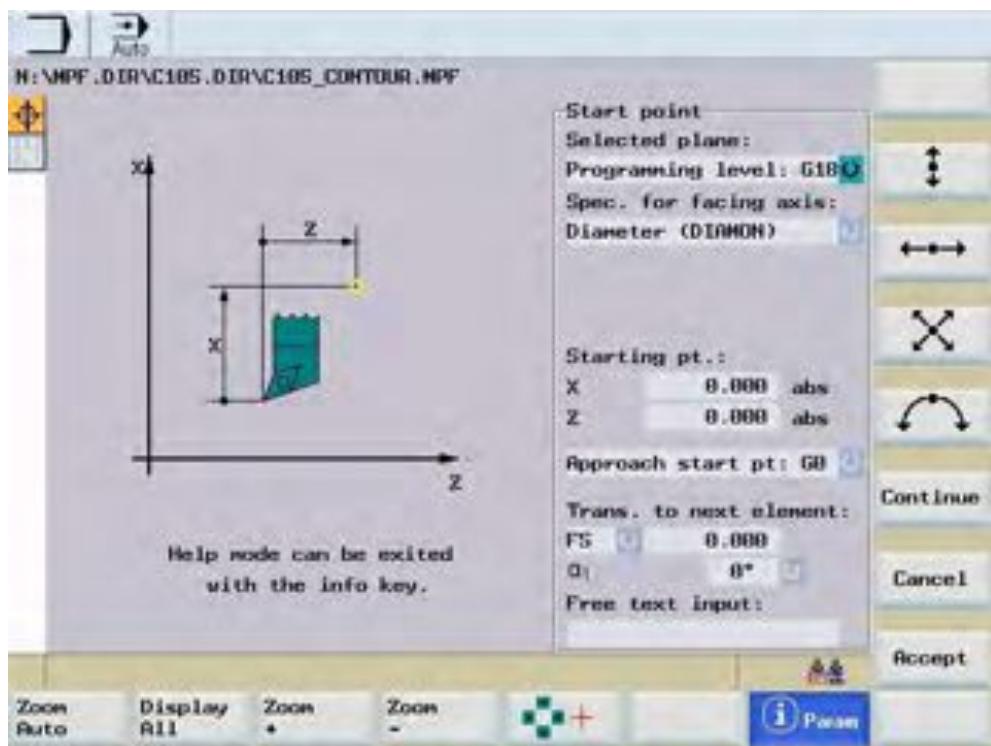
You can zoom in and out of the graphic with the first four horizontal soft keys.



When you select this soft key, you can move the red cross-hair with cursor keys and choose a picture detail to display.



If you press this soft key, help graphics are displayed in addition to the relevant parameter information (as seen below). Press the soft key again to exit help mode.

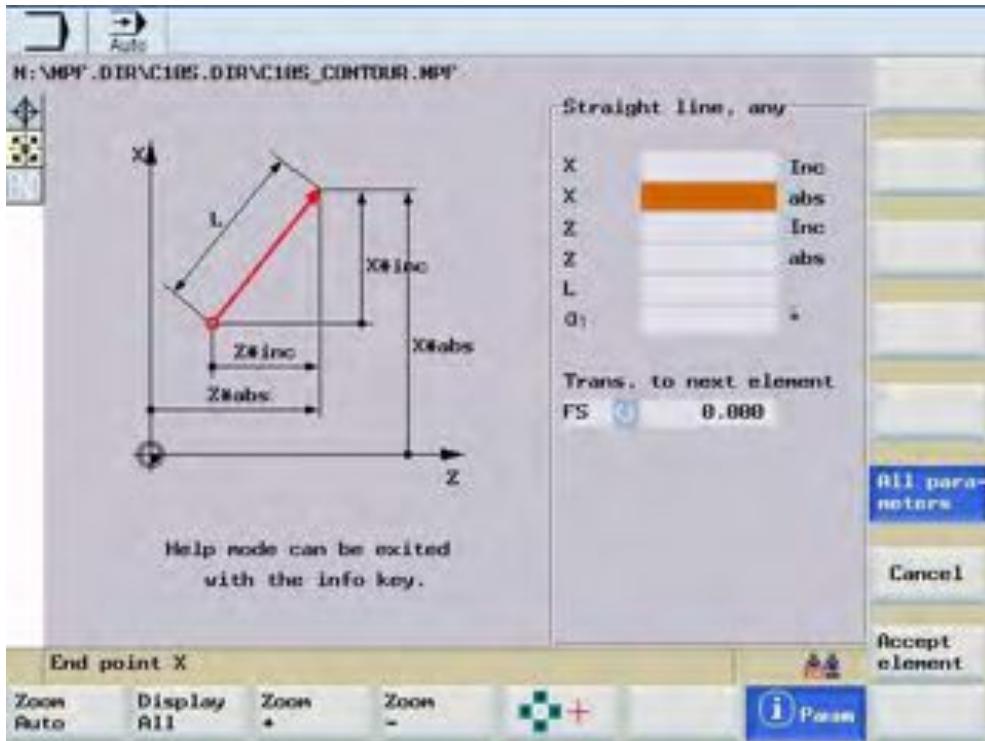


## Section 2

### Free Contour Programming Features

#### Parameter description of straight line /circle contour elements

Parameters for contour element “Straight line”



Notes

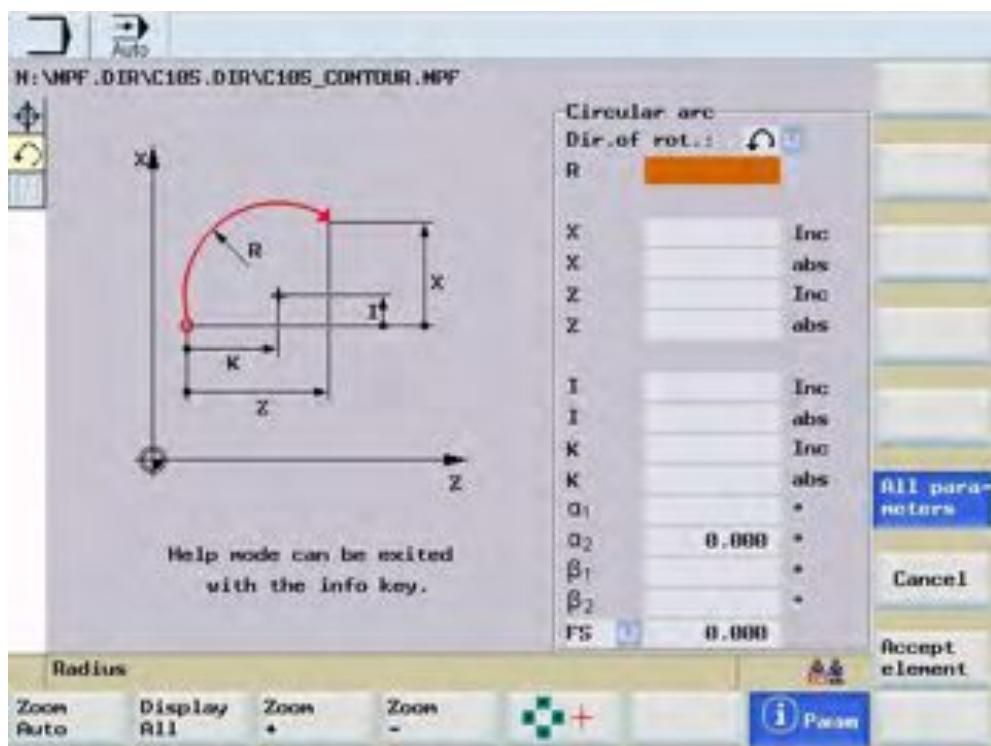
Parameter	Contour element “Straight line”
X inc X abs	Incremental end position in X direction Absolute end position in X direction
Z inc Z abs	Incremental end position in Z direction Absolute end position in Z direction
L	Length of straight line
α1	Pitch angle with reference to X axis
α2	Angle to preceding element; tangential transition: $\alpha_2=0$
Transition to next element	Transition element to next contour is a chamfer (FS) Transition element to next contour is a radius . FS=0 or R=0 means no transition element

## Section 2

### Free Contour Programming Features

Notes

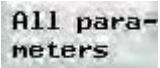
Parameters for contour element “Circle, Arc”



Parameter	Contour element “Circle, Arc”
Direction of rotation	In clockwise or counter-clockwise direction
R	Radius of circle
X inc X abs	Incremental end position in X direction Absolute end position in X direction
Z inc Z abs	Incremental end position in Z direction Absolute end position in Z direction
I	Position of circle center point in X direction (abs. or incr.)
K	Position of circle center point in Z direction (abs. or incr.)
L	Length of straight line
α1	Pitch angle with reference to X axis
α2	Angle to preceding element; tangential transition: $\alpha_2=0$
β1	End angle with reference to X axis
β2	Angle of aperture of circle
Transition to next element	Transition element to next contour is a chamfer (FS) Transition element to next contour is a radius . FS=0 or R=0 means no transition element

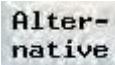
Notes

#### Additional soft keys

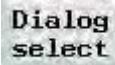


Select the "All parameters" soft key to display a selection list of all the parameters for the contour element.

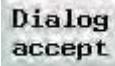
If you leave any parameter input fields blank, the control assumes that you do not know the correct value and attempts to calculate them from the values of the other parameters.



The "Alternative" soft key is displayed only in cases where the cursor is positioned on an input field with several switchover settings.



Some parameter configurations can produce several different contour lines. In such cases, you will be asked to select a dialog. By pressing the "Dialog select" soft key, you can select the available option in the graphic display area.



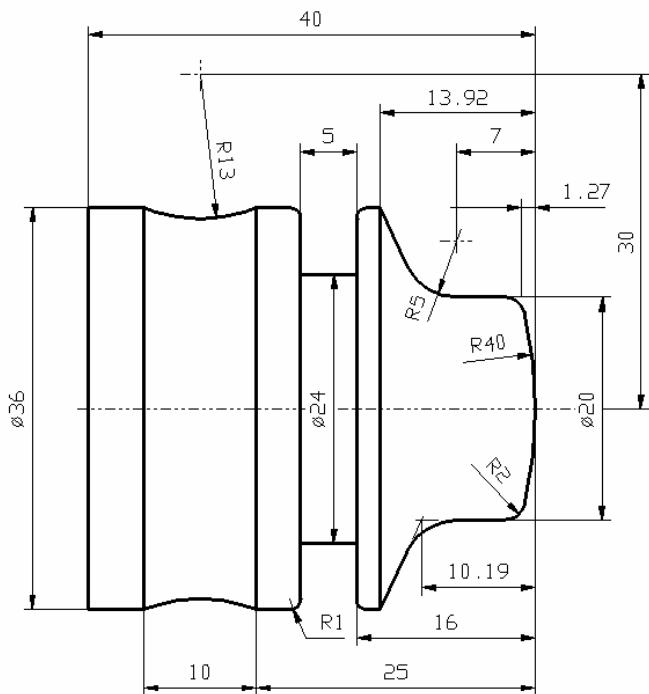
Select the "Dialog accept" soft key to conform your choice of available contours (see above).

## Section 3

### Free Contour Programming by example

Notes

An example program will be created using the following workpiece:



Create the start of the program as shown below.

Program editor:

```
N:\MPF\VC105\VC105_CONTOUR.MPF          9    Not selected
G00 G90 G95 G40 G10 G715
LIMS=25000
T1 D14
G96 S250 M03 M081
G00 X42.0 Z0.15
G01 X-2.0 F8.351
G00 Z22.01
X42.01
**eof**
```

Execute

- Mark block
- Copy block
- Insert block
- Delete block
- Find
- Renumber
- Templates

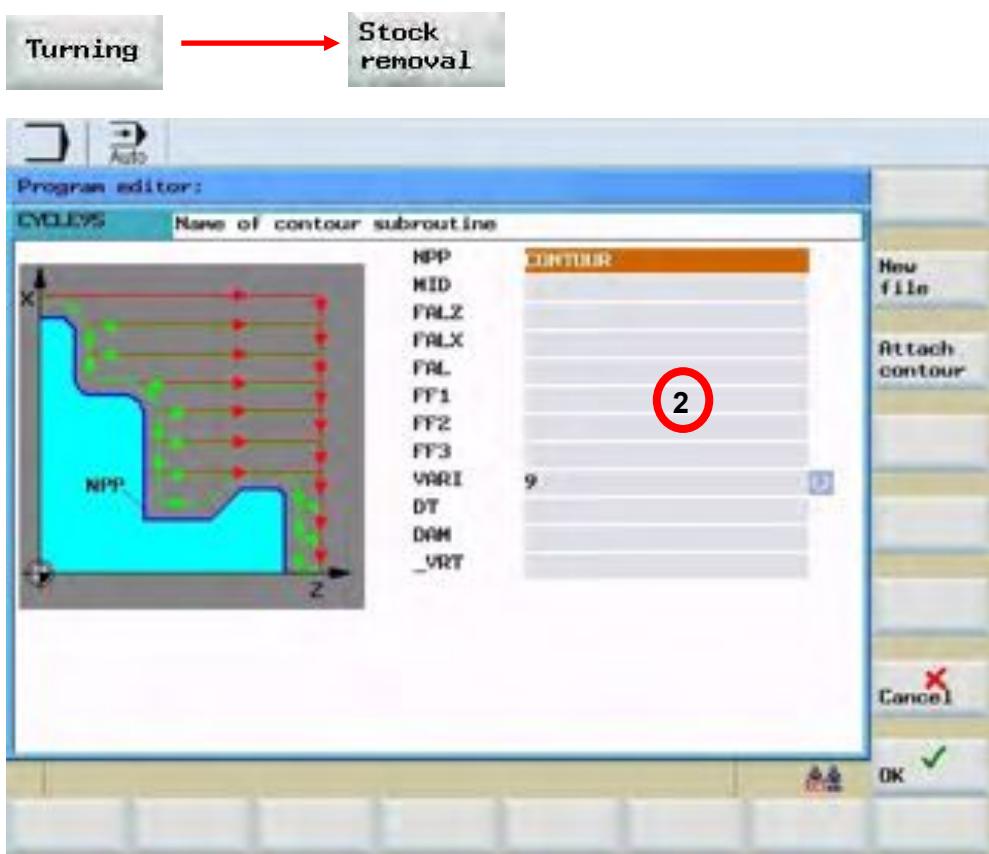
Edit Contour Drilling Milling Turning Simulation Re-compile

## Section 3

### Free Contour Programming by example

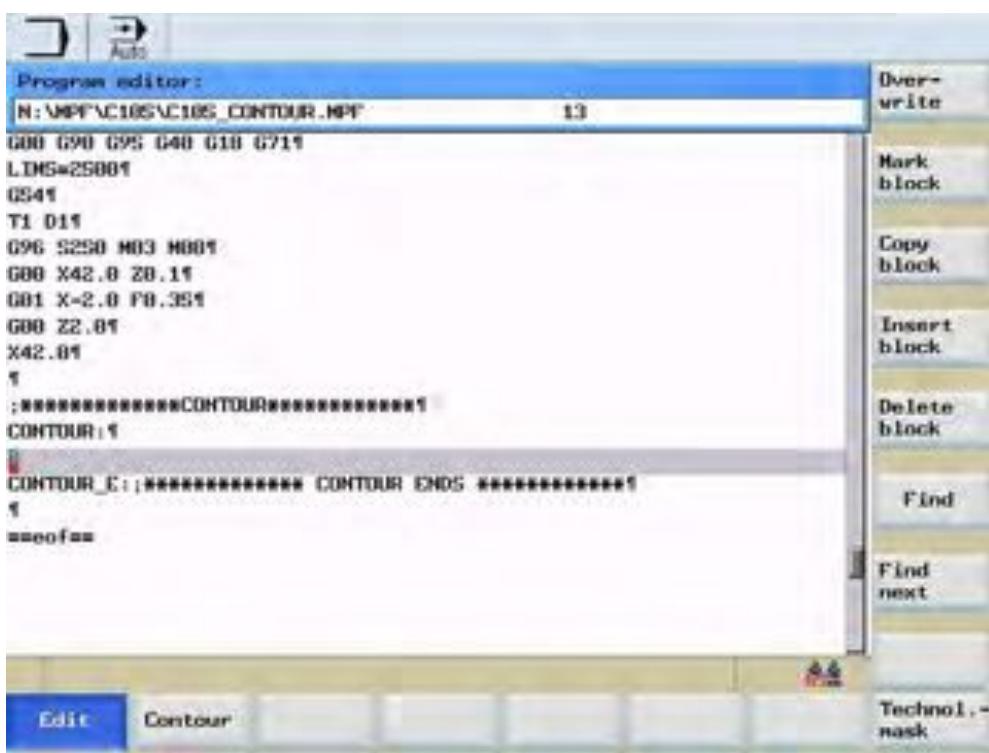
Notes

Follow the sequence to create the contour for the stock removal cycle.



- 2 Enter name of contour (NPP)

Attach contour



## Section 3

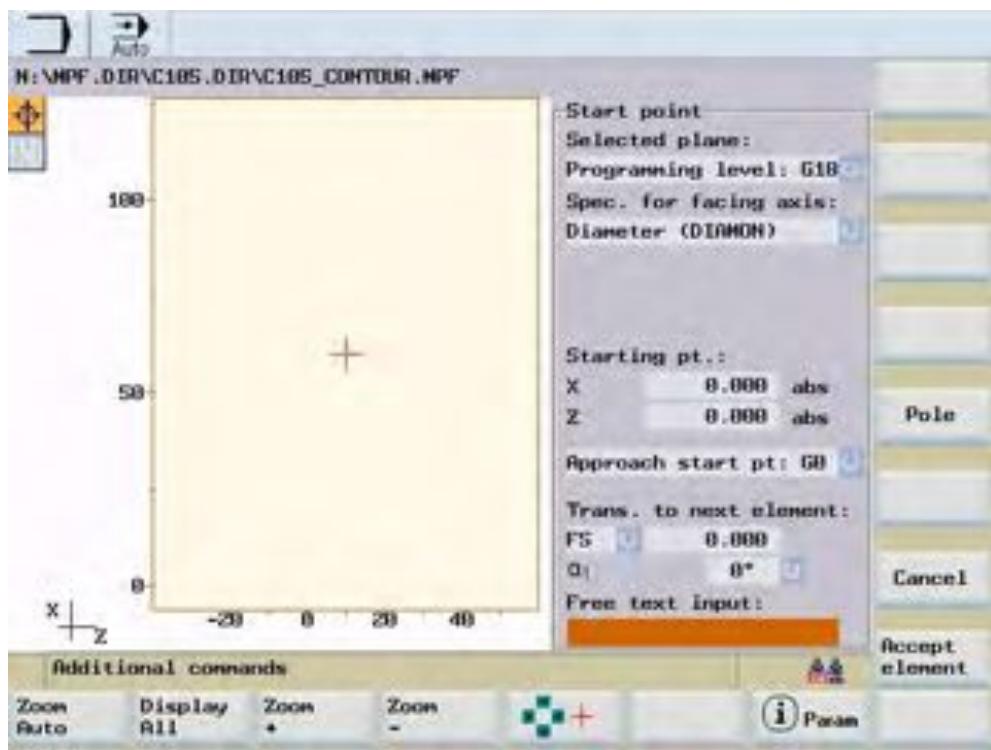
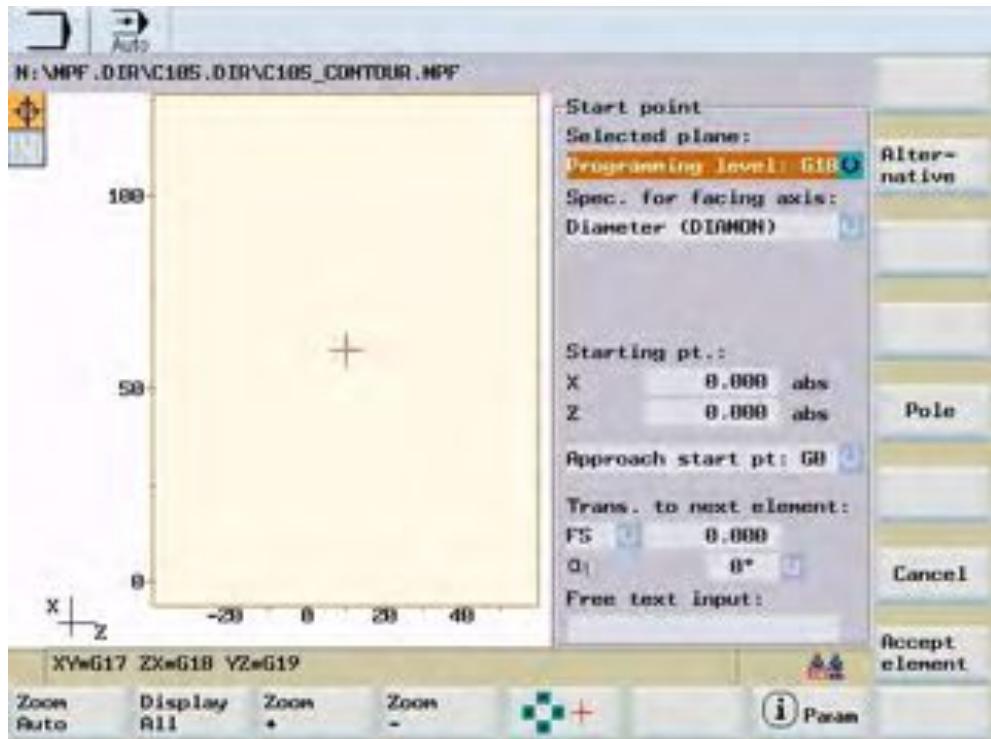
### Free Contour Programming by example

Notes

To program the contour follow this sequence.

Contour

The first element that must be entered is the start point.



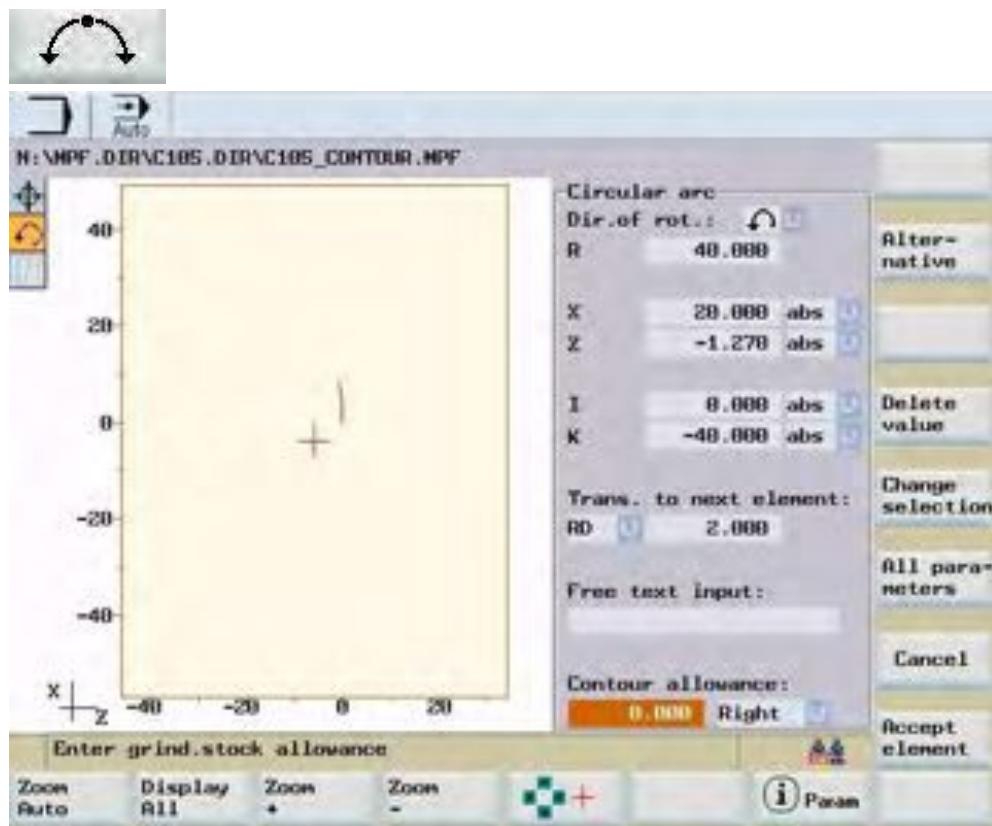
Accept element

## Section 3

### Free Contour Programming by example

Now follow the next 8 elements:

Notes



Accept element

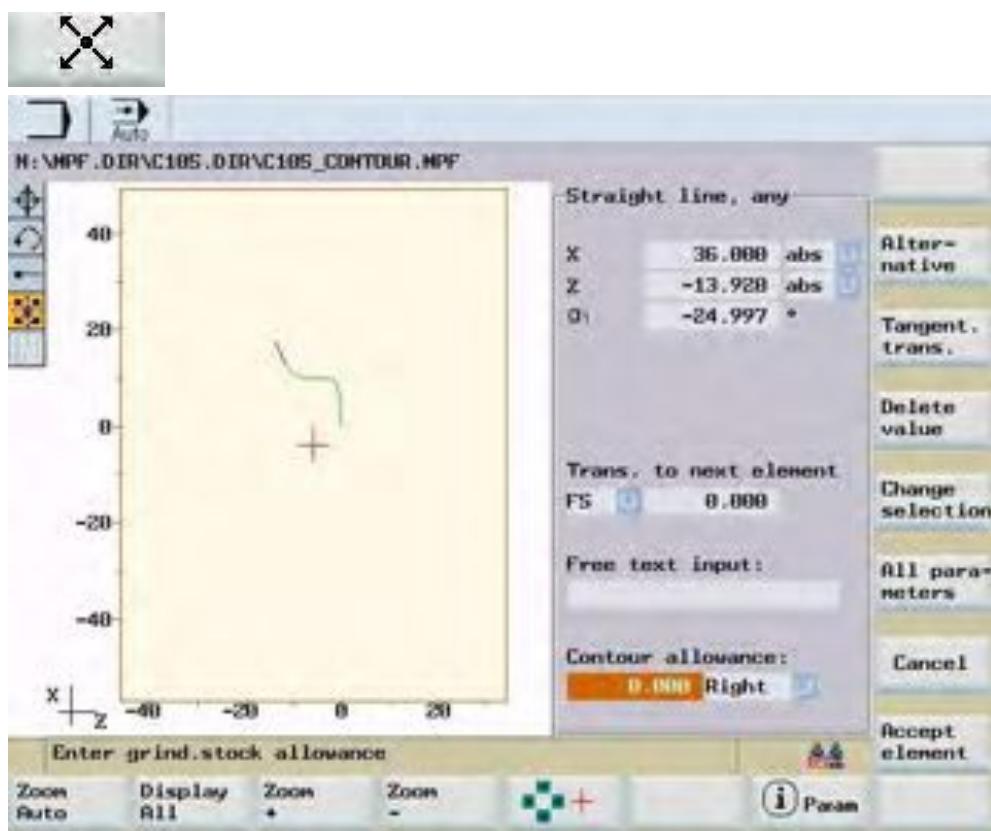


Accept element

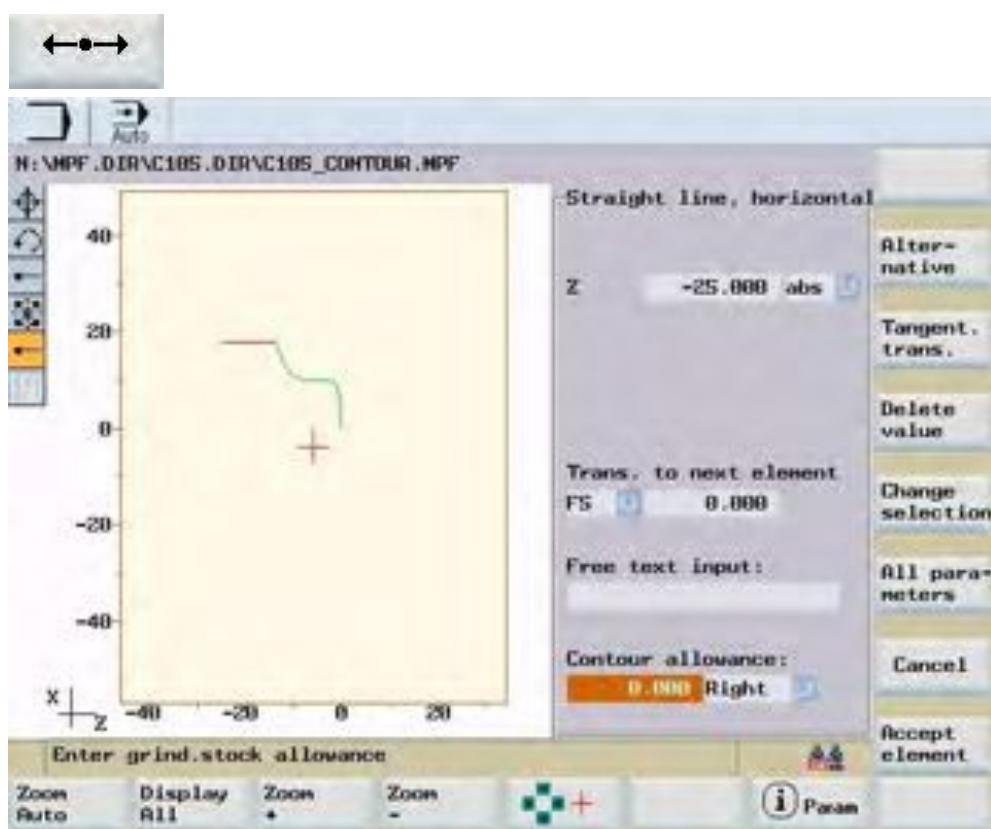
## Section 3

### Free Contour Programming by example

Notes



Accept element

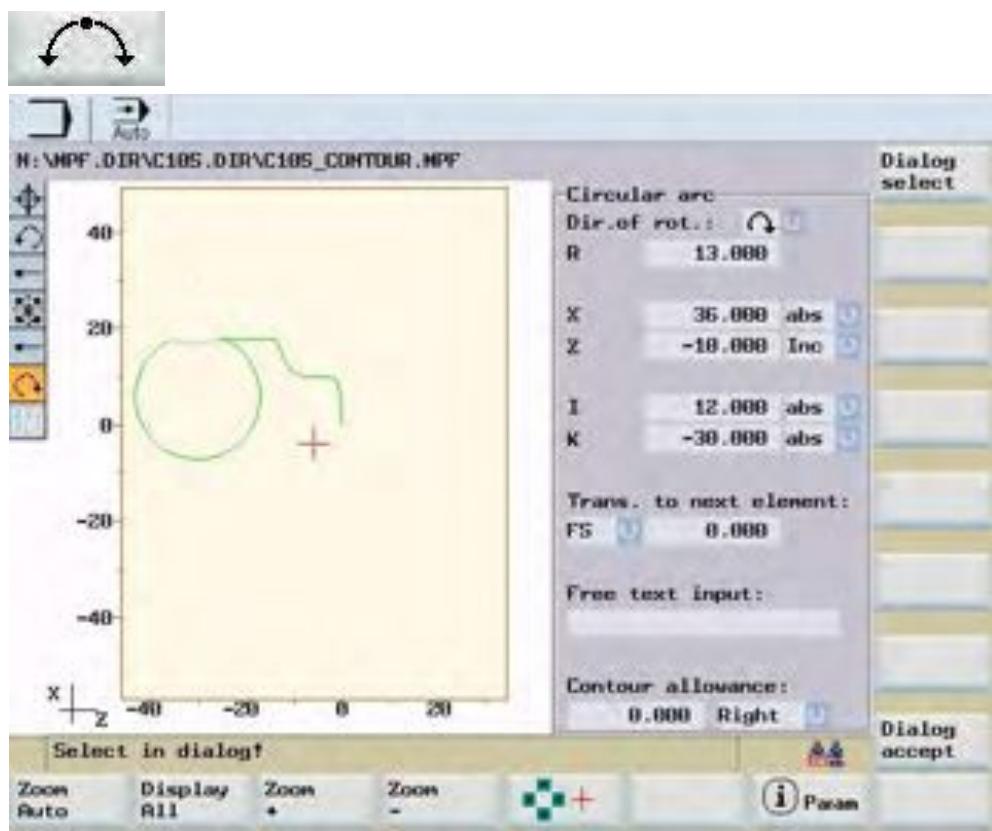


Accept element

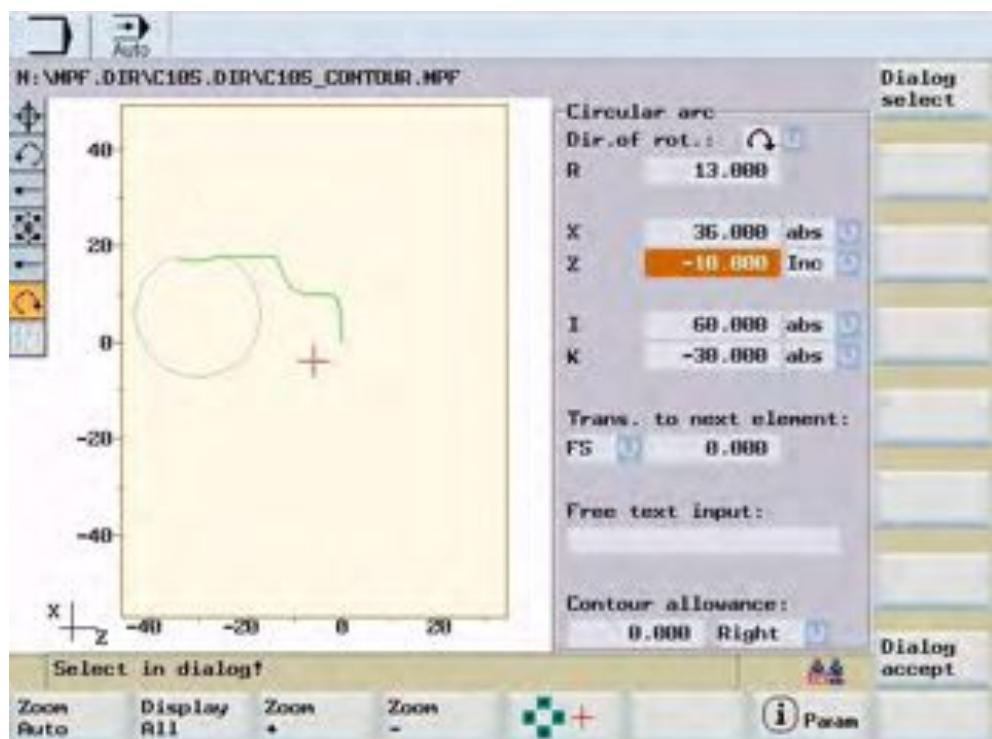
## Section 3

### Free Contour Programming by example

Notes



Dialog  
select

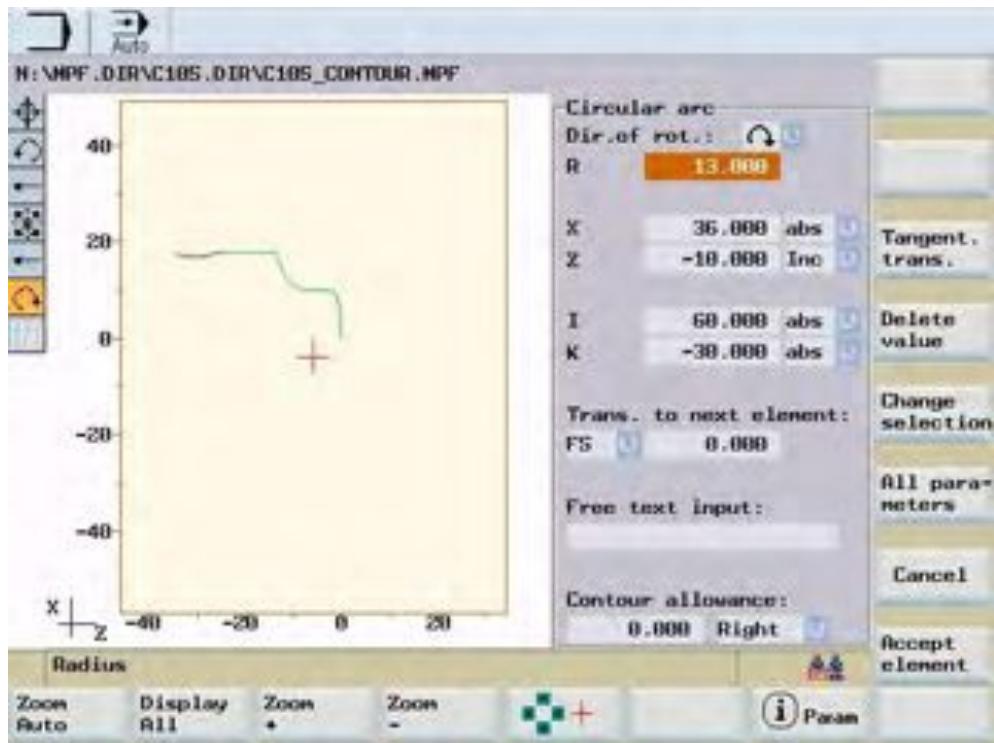


Dialog  
accept

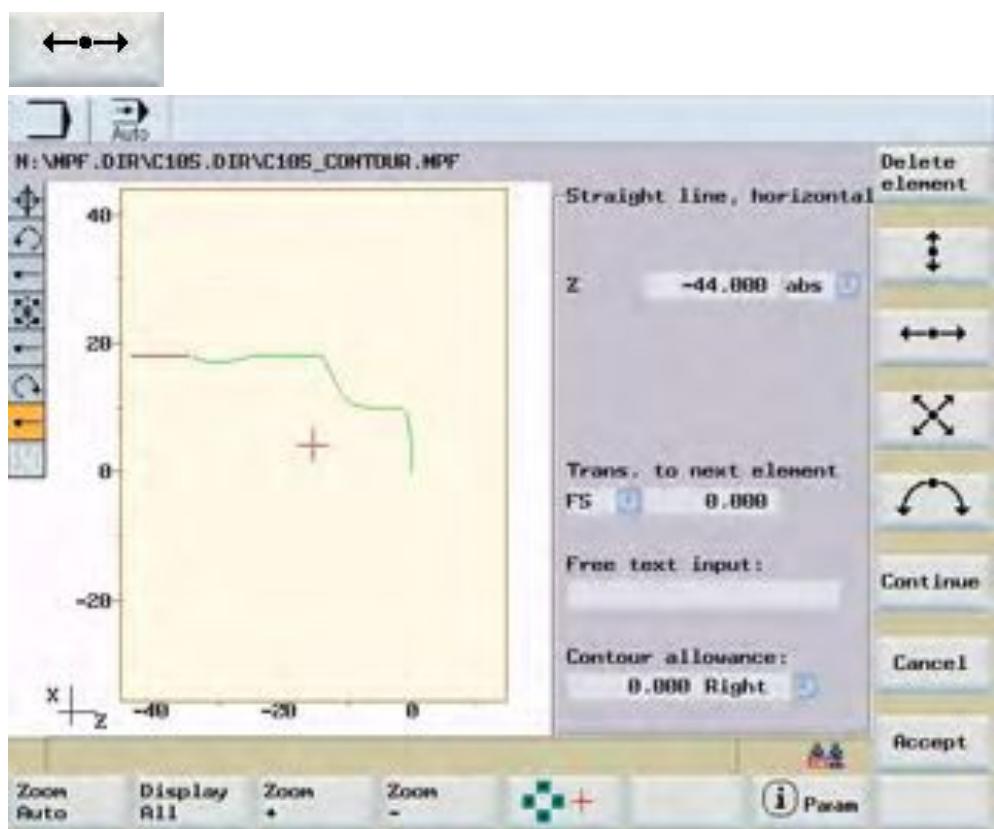
## Section 3

### Free Contour Programming by example

Notes



Accept element

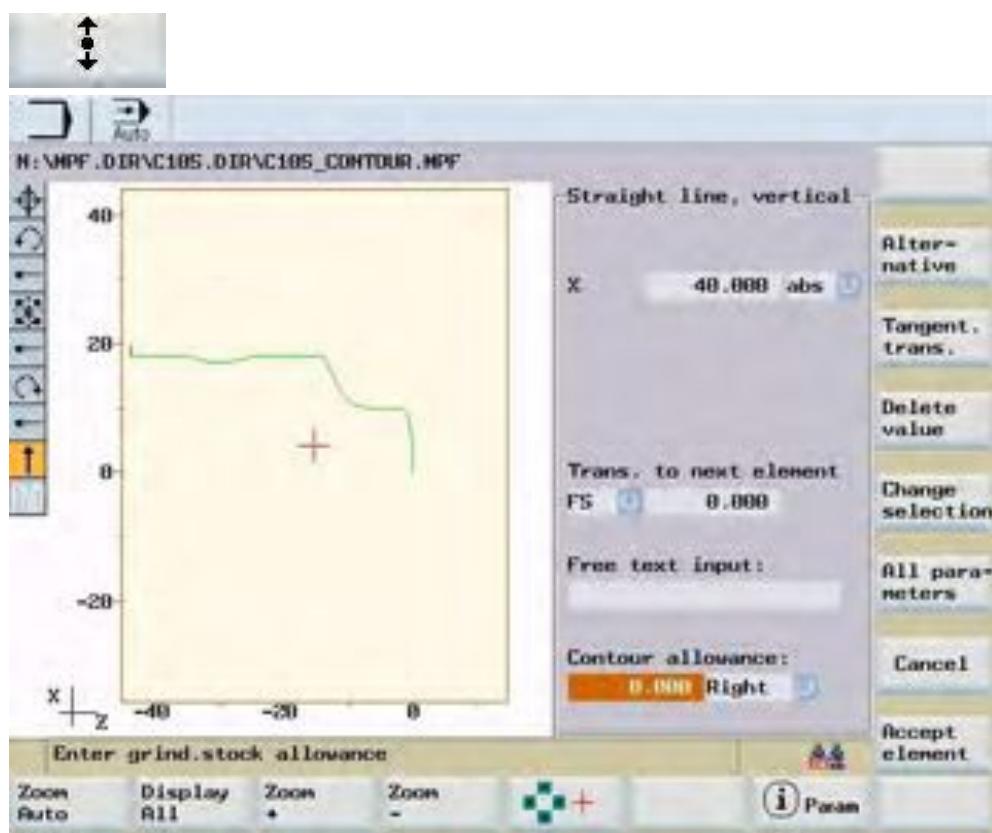


Accept element

## Section 3

### Free Contour Programming by example

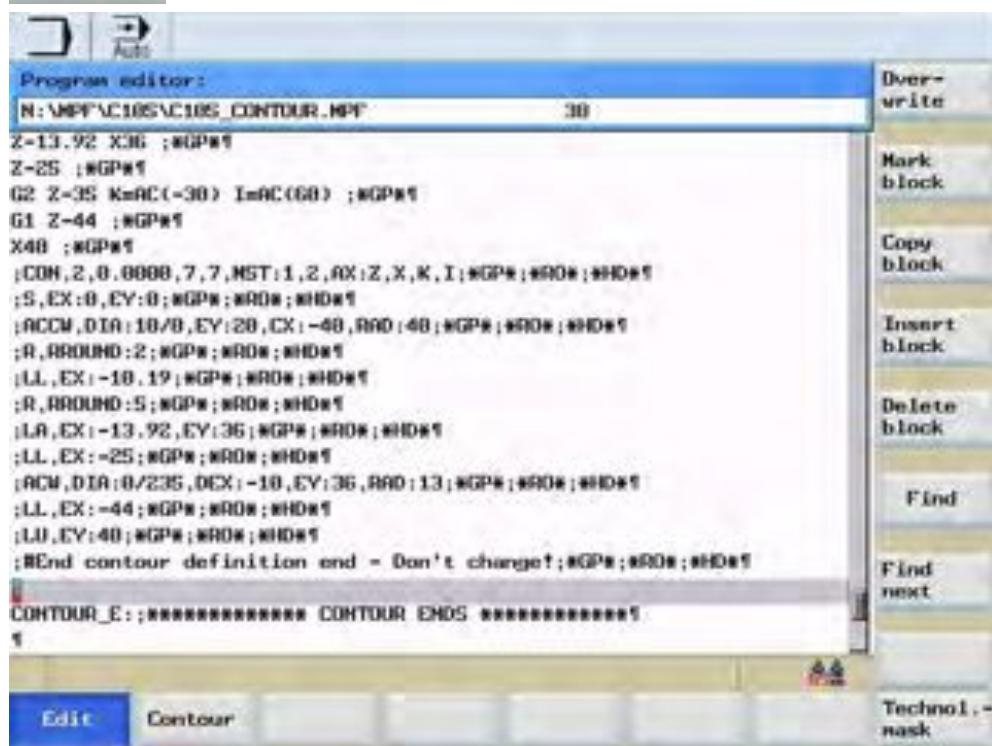
Notes



Accept element

Use the following sequence to enter the contour definition into the NC program.

Accept

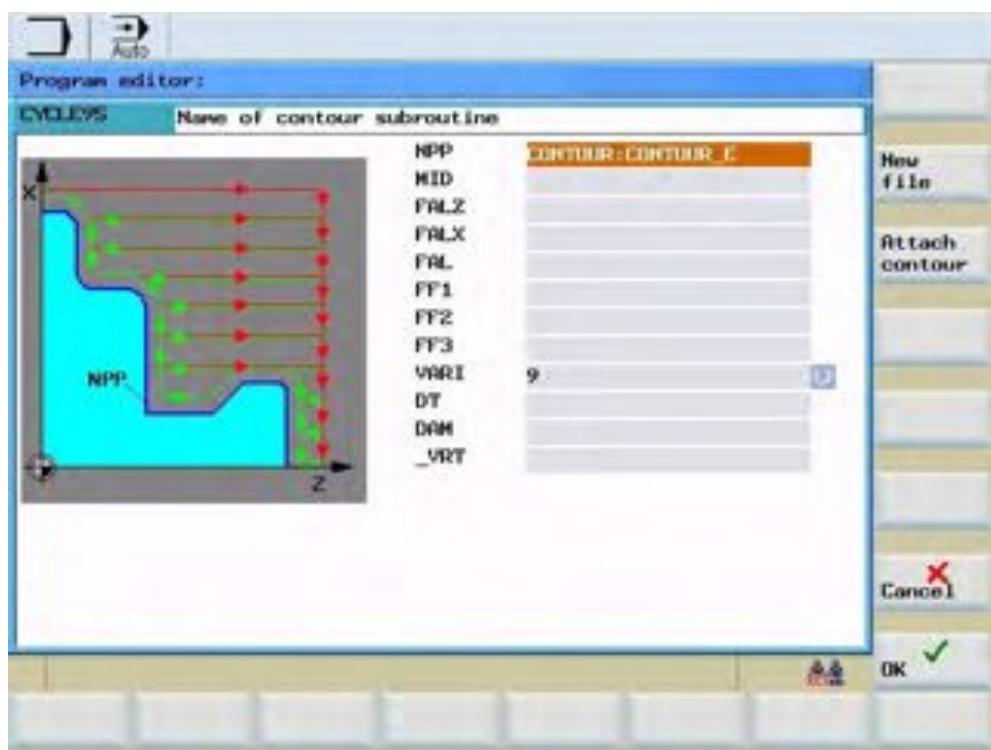


Technol.-mask

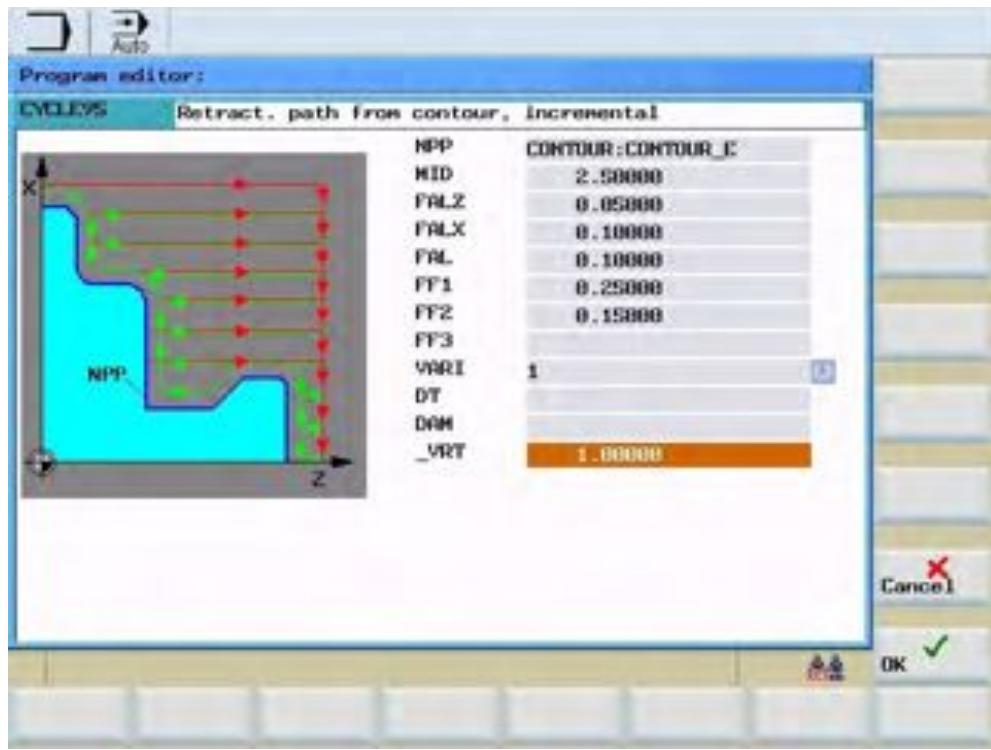
## Section 3

### Free Contour Programming by example

Notes



Enter data into the stock removal parameter mask



OK ✓

## Section 3

### Free Contour Programming by example

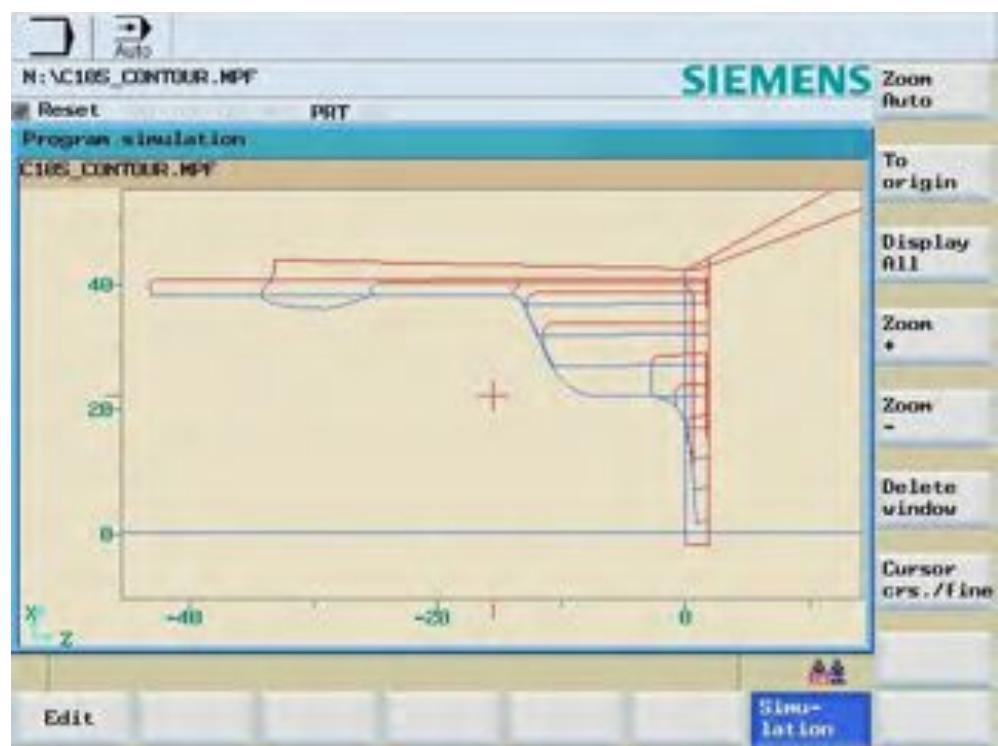
Notes

```
G00 Z22.0F  
X42.0F  
CYCLE95C("CONTOUR:CONTOUR_E", 2.50000, 0.05000, 0.10000, 0.10000, 0.  
25000, 0.15000, ,1, , ,1.00000)  
;*****CONTOUR*****  
CONTOUR:  
#7_DigK contour definition begin - Don't change! ;#GP#;#RD#;#HD#  
G18 G90 DIAMON;#GP#  
G0 Z0 X0 G01 ;#GP#  
G3 Z-1.27 X20 K=AC(-48) I=AC(8) RHD=2 ;#GP#  
G1 Z-10.19 RHD=S ;#GP#  
Z-13.92 X36 ;#GP#  
Z-25 ;#GP#  
G2 Z-35 K=AC(-30) I=AC(68) ;#GP#  
G1 Z-44 ;#GP#  
X40 ;#GP#  
;DNH,2,0.0000,7,7,MST:1,2,0X:Z,X,K,I:#GP#;#RD#;#HD#  
;S,EX:0,EY:0:#GP#;#RD#;#HD#
```

Edit    Contour    Drilling    Milling    Turning    Simulation    Re-compile

Check the program using Simulation.

Simu-  
lation



Edit

## 1. Description

**Module objective:**

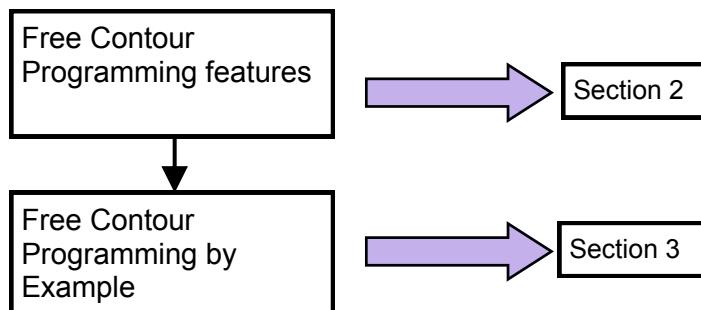
Upon completion of this module you can use Free Contour Programming, by example.

**Module description:**

Free contour programming is a support tool for the editor. The contour programming function enables you to create simple and complex contours.

**Module content:**

Free Contour Programming Features  
Free Contour Programming by Example



## Section 2

### Free Contour Programming Features

Notes

#### Functionality

Free contour programming is a support tool for the editor. The contour programming function enables you to create simple and complex contours. An integrated contour calculator (geometry processor) calculates any missing parameters for you, provided that they can be calculated from other parameters. You can link together contour elements. Contour transition elements "radius" and "chamfer" are also provided. The programmed contours are transferred to the edited part program.

#### Contour elements

The following are contour elements:

Start point



Straight line in the horizontal direction



Straight line in the vertical direction



Oblique straight line



Circular arc

## Section 2

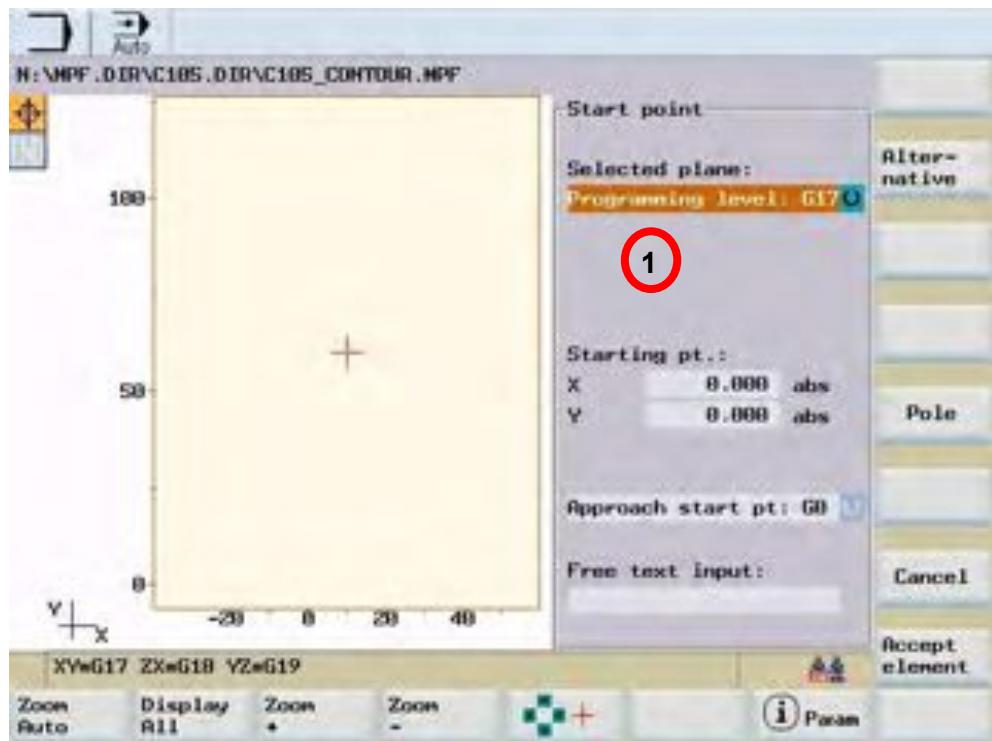
### Free Contour Programming Features

Notes

#### Define a start point

When entering a contour, begin at a position which you already know and enter it as the starting point. The sequence of operations for defining the start point of a contour is as follows.

You have opened a part program and selected soft key “Contour” to program a new contour. The input screen for specifying the start of the contour is displayed as below:



1

As you cursor down the “start point” field they will be highlighted in a dark brown field.

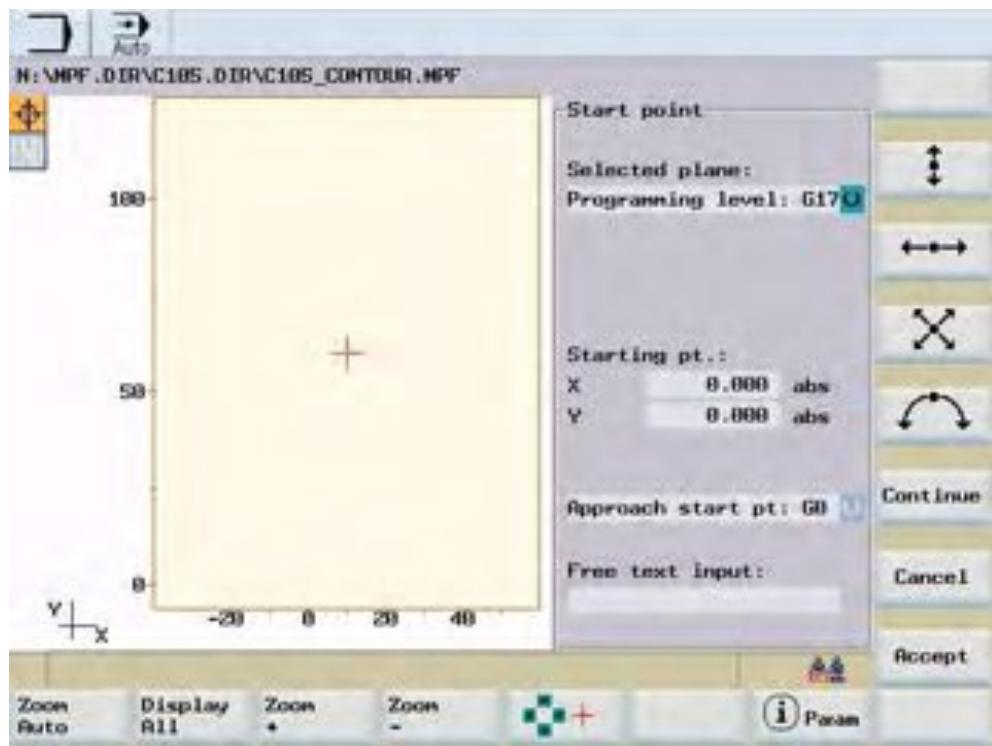
## Section 2

### Free Contour Programming Features

Notes

#### Soft keys and parameters.

Once you have defined the contour start point, you can begin programming the individual contour elements from the main screen below:



#### Vertical soft keys

The following contour elements are available for programming contours:



Straight line in the vertical direction.



Straight line in the horizontal direction.



Oblique line in the X/Y direction. Enter the end point of the line using coordinates or an angle.



Arc with any direction of rotation.

**Continue**

The “Continue” soft key accesses the “Pole” sub screen softkey.

**Cancel**

By selecting the “Cancel” soft key you can return to the main screen without transferring the last edited values.

**Accept**

When you select “Accept” soft key you close the contour screen and return to the program editor.

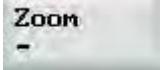
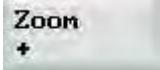
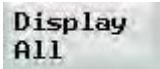
## Section 2

### Free Contour Programming Features

#### Horizontal soft keys



You can zoom in and out of the graphic with the first four horizontal soft keys.

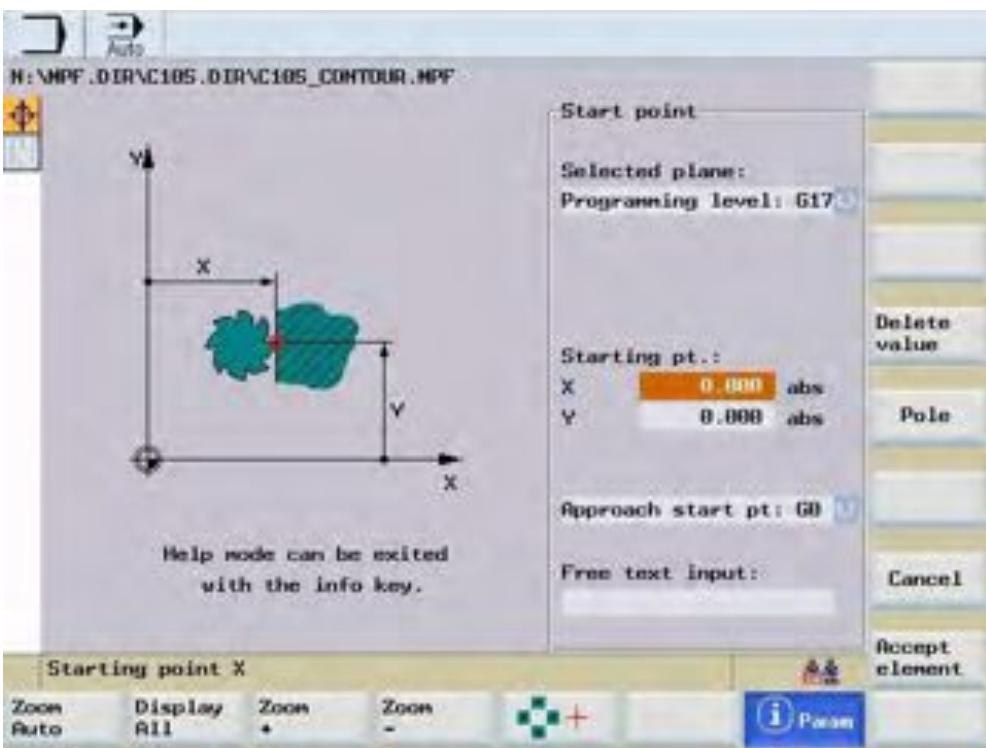


When you select this soft key, you can move the red cross-hair with cursor keys and choose a picture detail to display.



If you press this soft key, help graphics are displayed in addition to the relevant parameter information (as seen below). Press the soft key again to exit help mode.

Notes

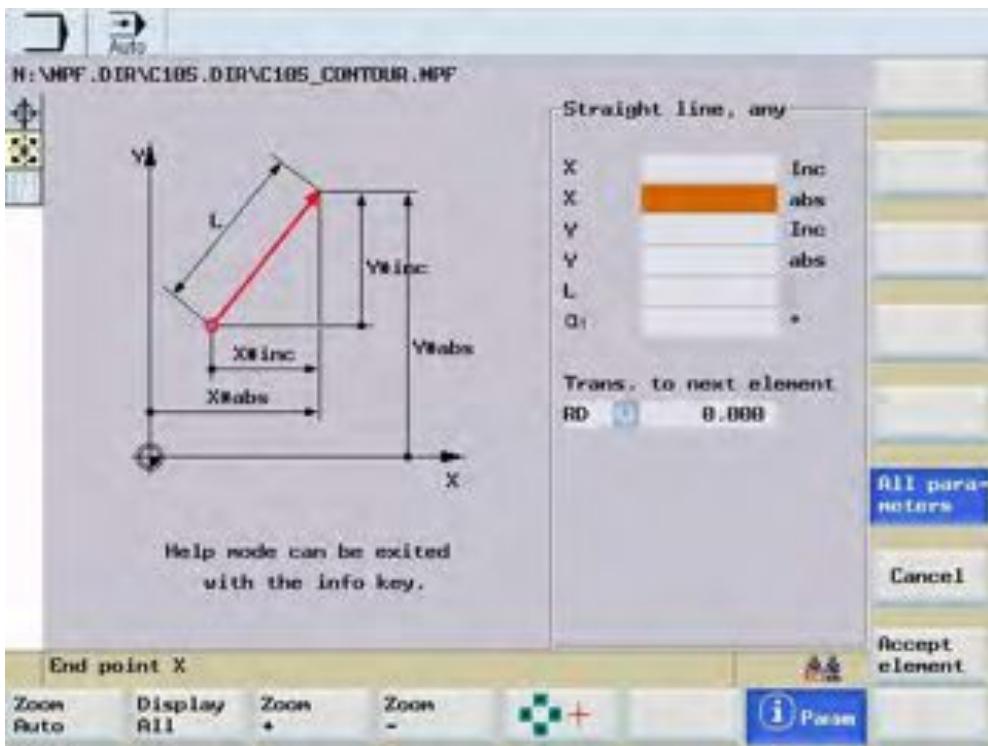


## Section 2

### Free Contour Programming Features

#### Parameter description of straight line /circle contour elements

Parameters for contour element “Straight line”



Notes

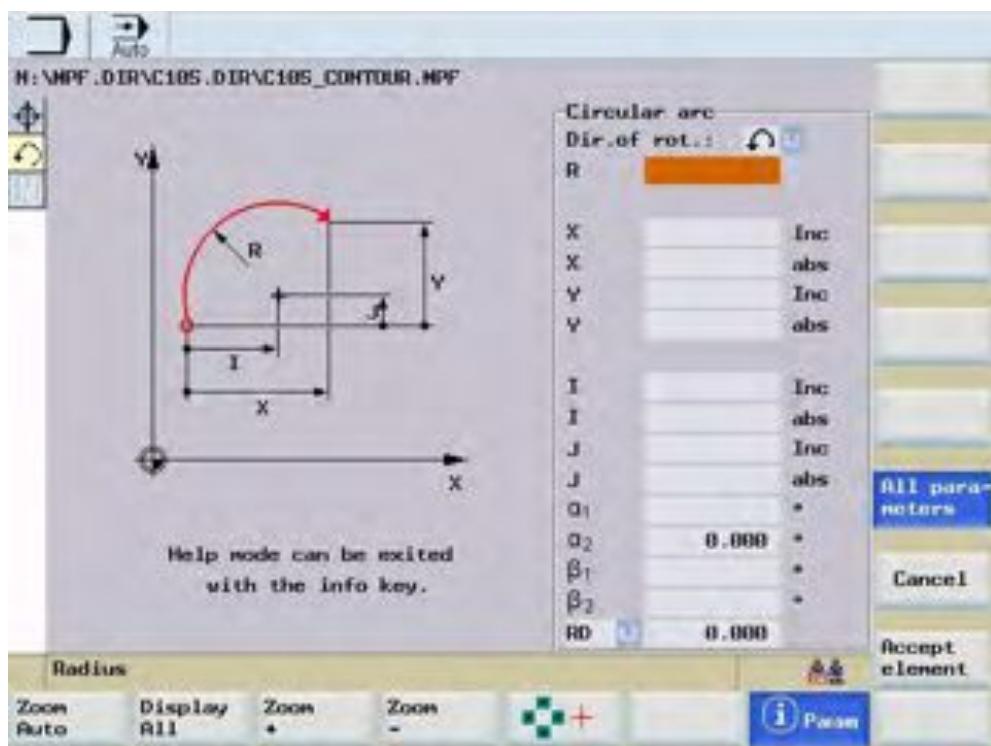
Parameter	Contour element “Straight line”
X inc	Incremental end position in X direction
X abs	Absolute end position in X direction
Y inc	Incremental end position in Y direction
Y abs	Absolute end position in Y direction
L	Length of straight line
α1	Pitch angle with reference to X axis
α2	Angle to preceding element; tangential transition: $\alpha_2=0$
Transition to next element	Transition element to next contour is a chamfer (FS) Transition element to next contour is a radius . FS=0 or R=0 means no transition element

## Section 2

### Free Contour Programming Features

Notes

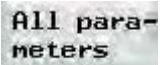
Parameters for contour element “Circle, Arc”



Parameter	Contour element “Circle, Arc”
Direction of rotation	In clockwise or counter-clockwise direction
R	Radius of circle
X inc X abs	Incremental end position in X direction Absolute end position in X direction
Y inc Y abs	Incremental end position in Y direction Absolute end position in Y direction
I	Position of circle center point in X direction (abs. or incr.)
J	Position of circle center point in Y direction (abs. or incr.)
L	Length of straight line
α1	Pitch angle with reference to X axis
α2	Angle to preceding element; tangential transition: $\alpha_2=0$
β1	End angle with reference to X axis
β2	Angle of aperture of circle
Transition to next element	Transition element to next contour is a chamfer (FS) Transition element to next contour is a radius . FS=0 or R=0 means no transition element

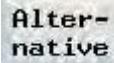
Notes

#### Additional soft keys

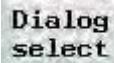


Select the “All parameters” soft key to display a selection list of all the parameters for the contour element.

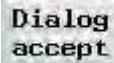
If you leave any parameter input fields blank, the control assumes that you do not know the correct value and attempts to calculate them from the values of the other parameters.



The “Alternative” soft key is displayed only in cases where the cursor is positioned on an input field with several switchover settings.



Some parameter configurations can produce several different contour lines. In such cases, you will be asked to select a dialog. By pressing the “Dialog select” soft key, you can select the available option in the graphic display area.

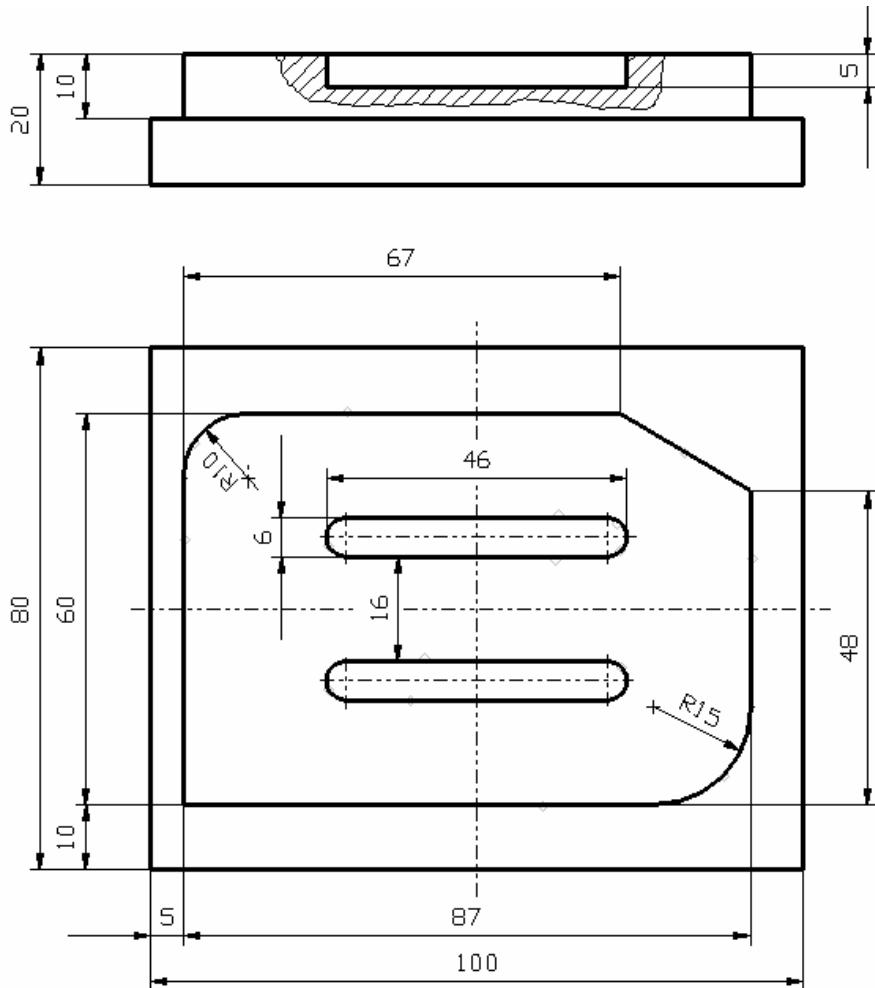


Select the “Dialog accept” soft key to conform your choice of available contours (see above).

## Section 3

### Free Contour Programming by example

An example program will be created using the following workpiece:



Notes

Create the start of the program as shown below.

```
G00 G90 G94 G40 G17 G715
G95 S1000 F0.35 M03 M009
G80 G54 X-35.0 Y-35.0 Z50.01
Z2.01
G81 Z-10.01
M00
**eof**
```

## Section 3

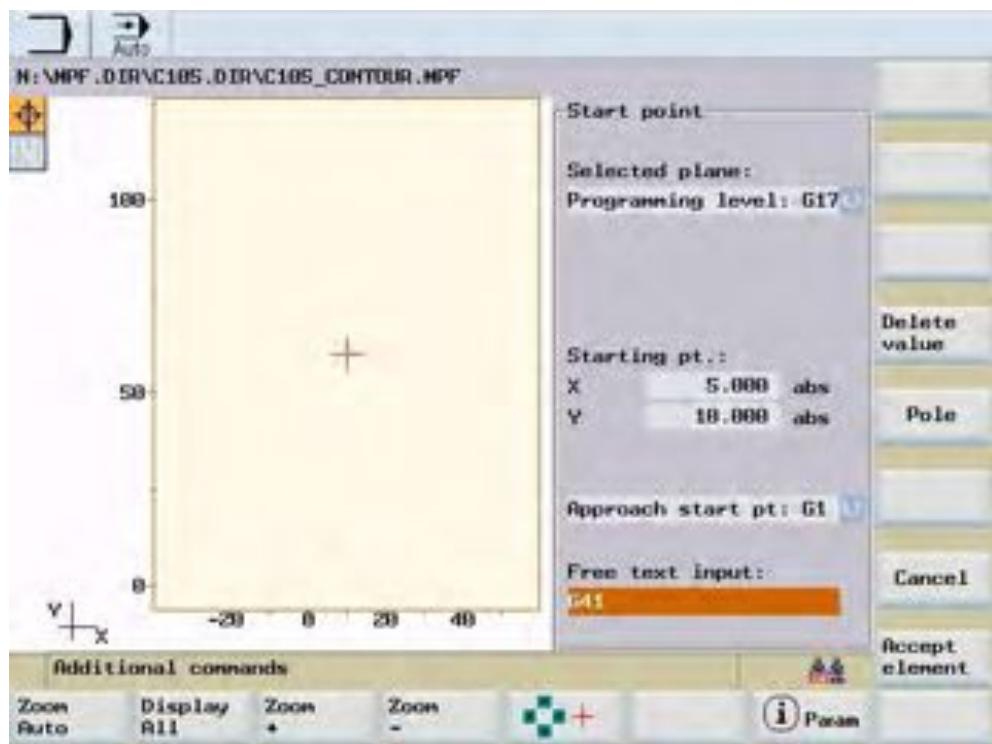
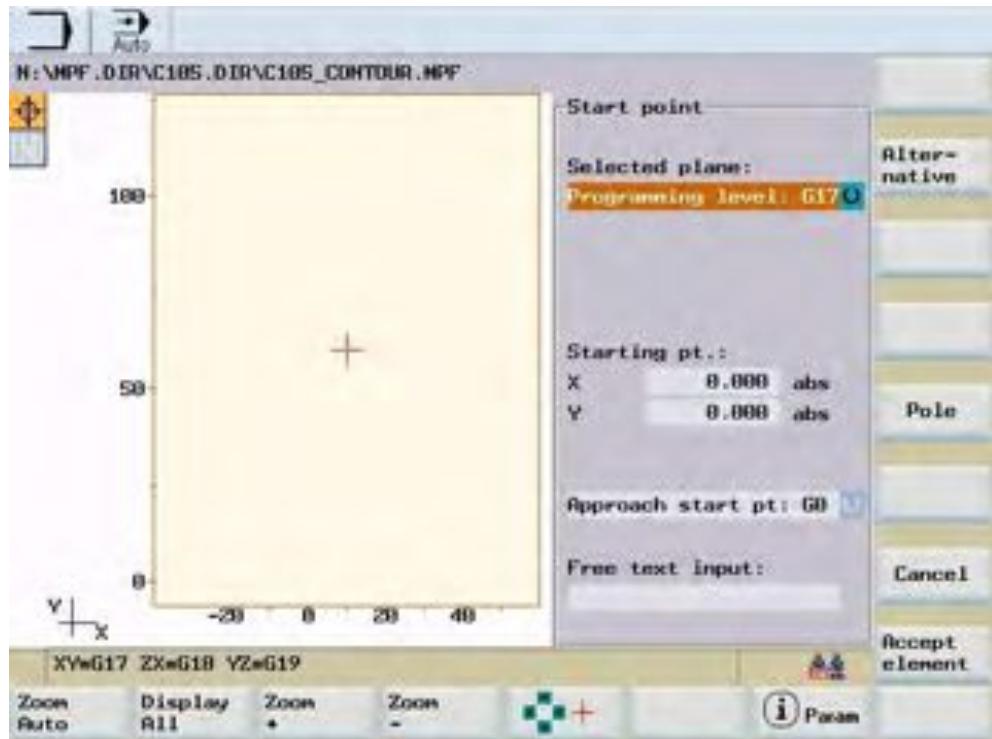
### Free Contour Programming by example

Notes

To program the contour follow this sequence.

Contour

The first element that must be entered is the start point.



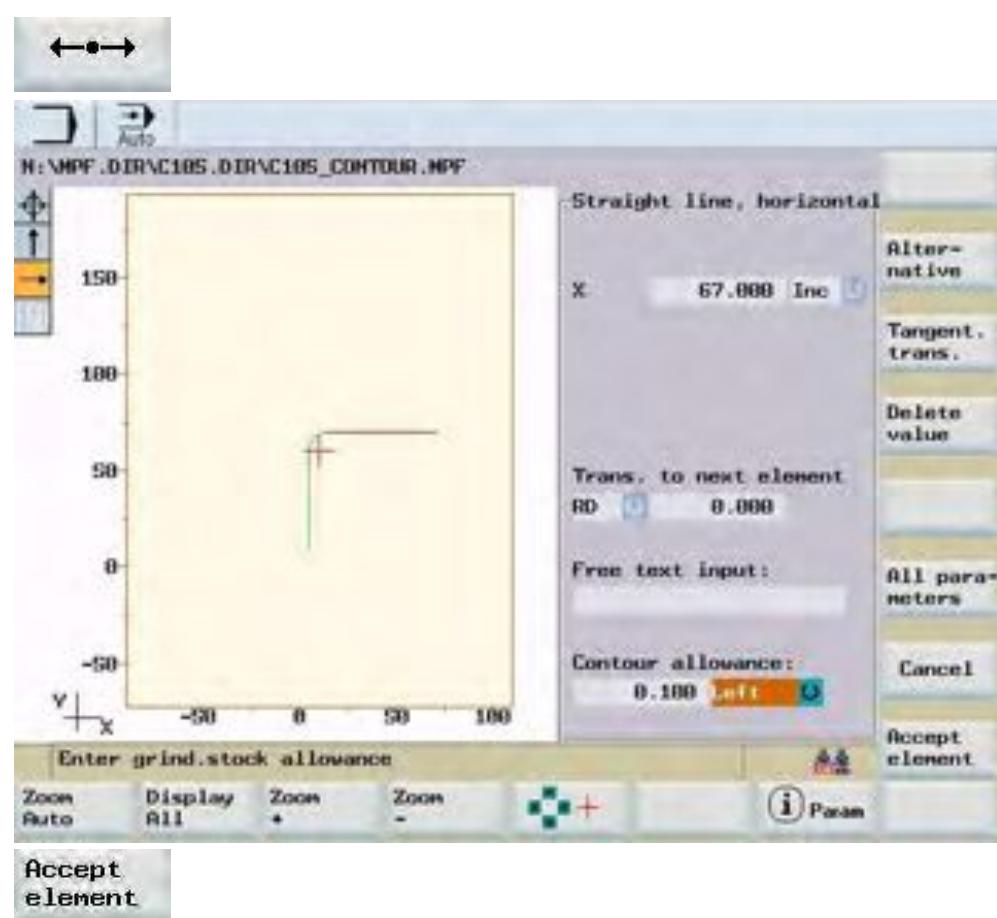
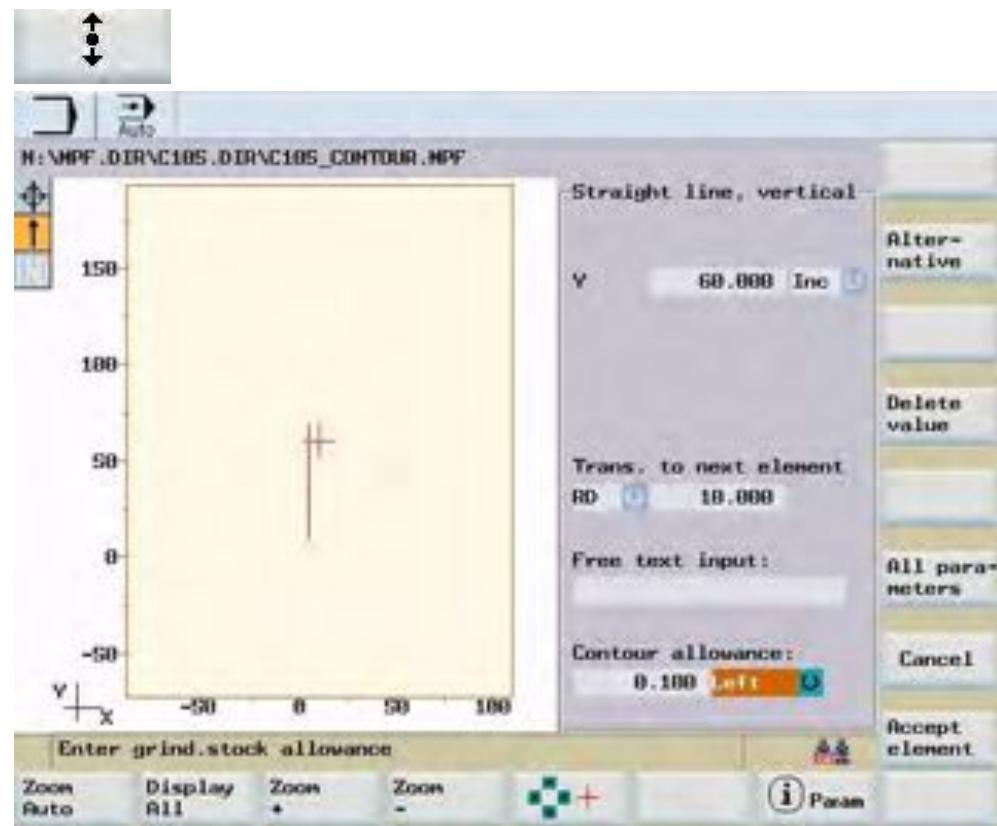
Accept element

## Section 3

### Free Contour Programming by example

Now follow the next 6 elements:

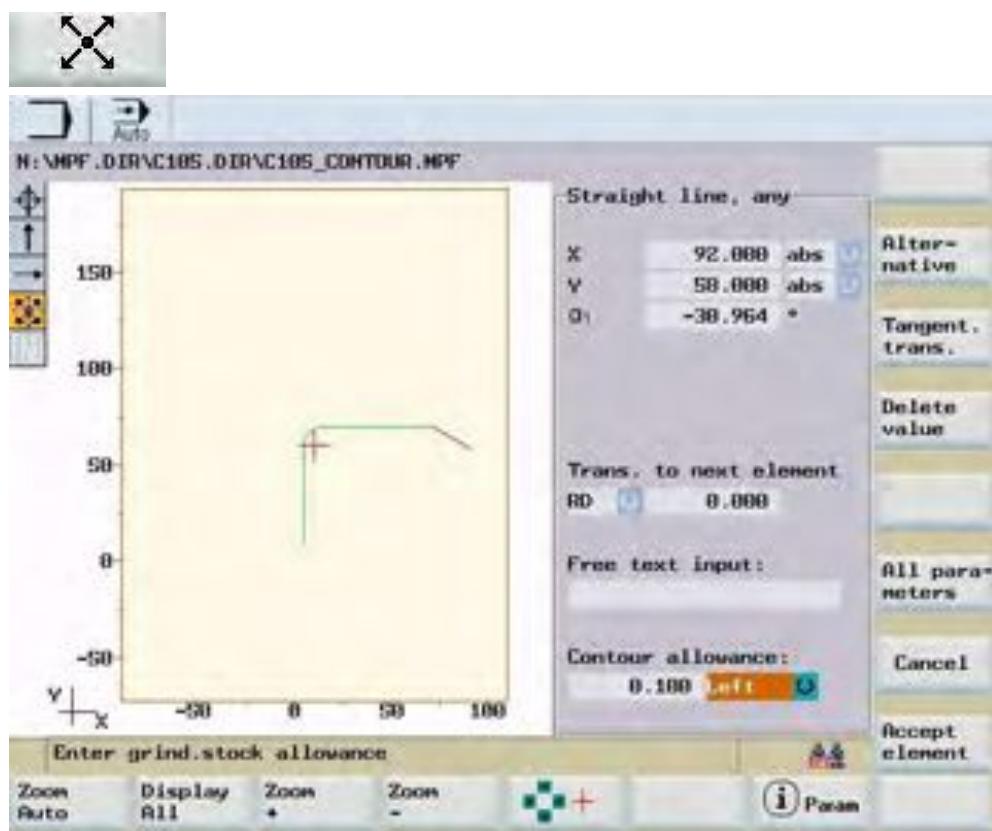
Notes



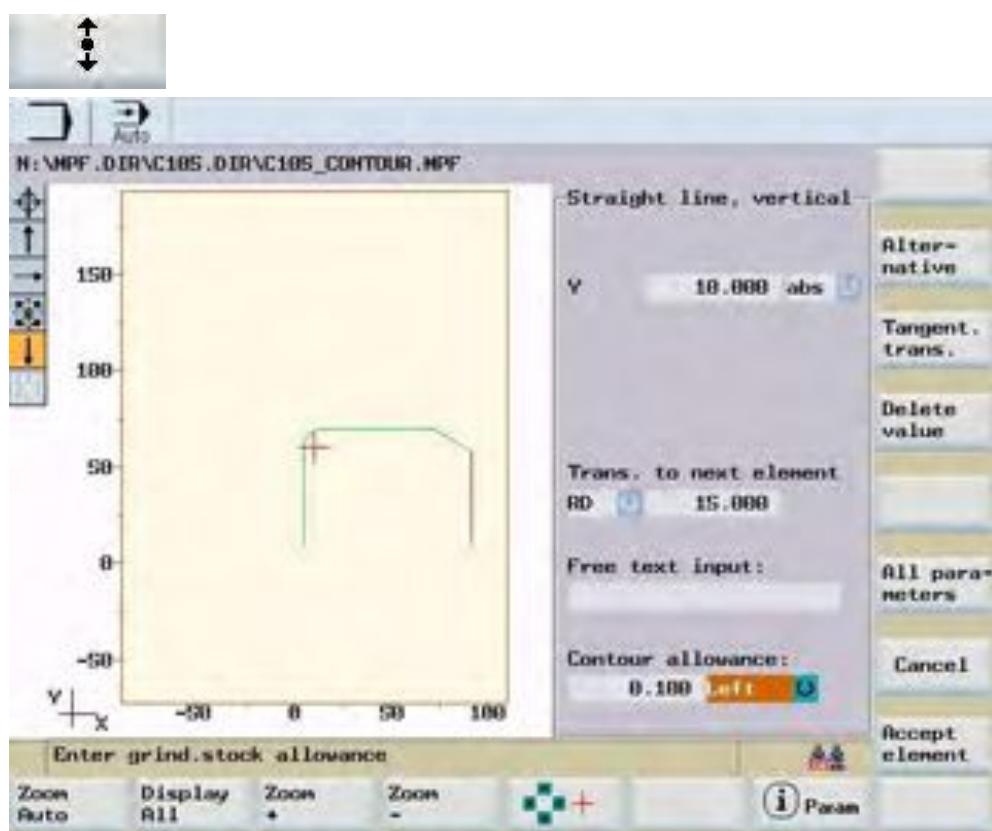
## Section 3

### Free Contour Programming by example

Notes



Accept element

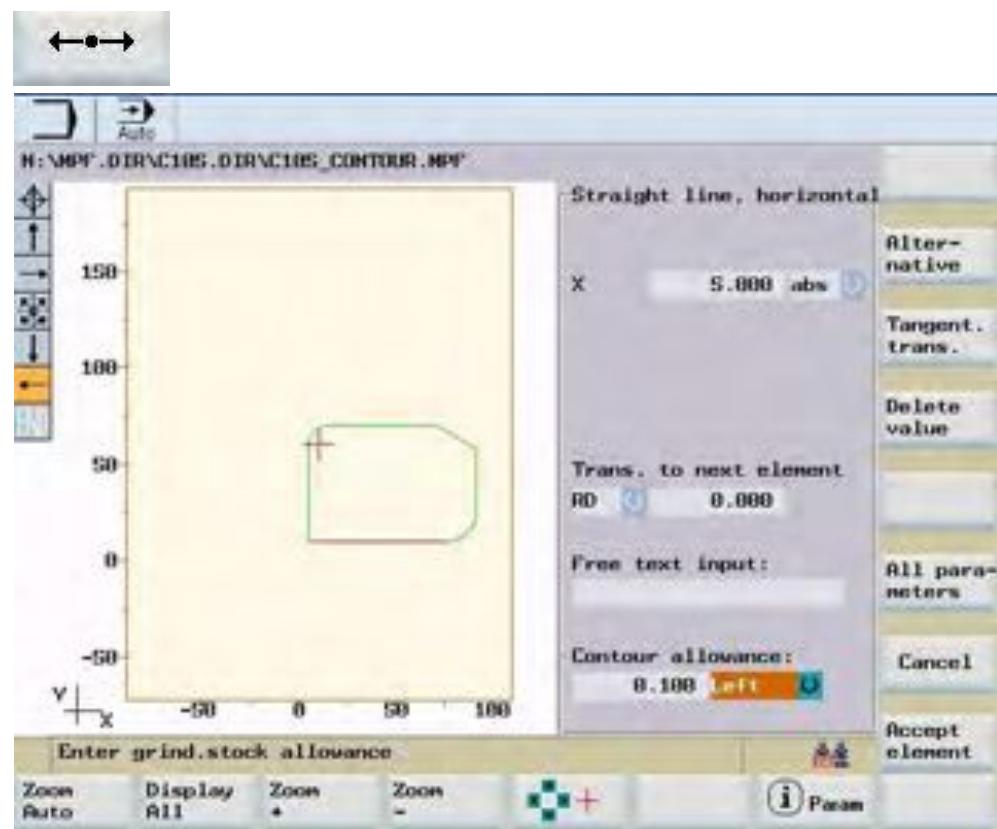


Accept element

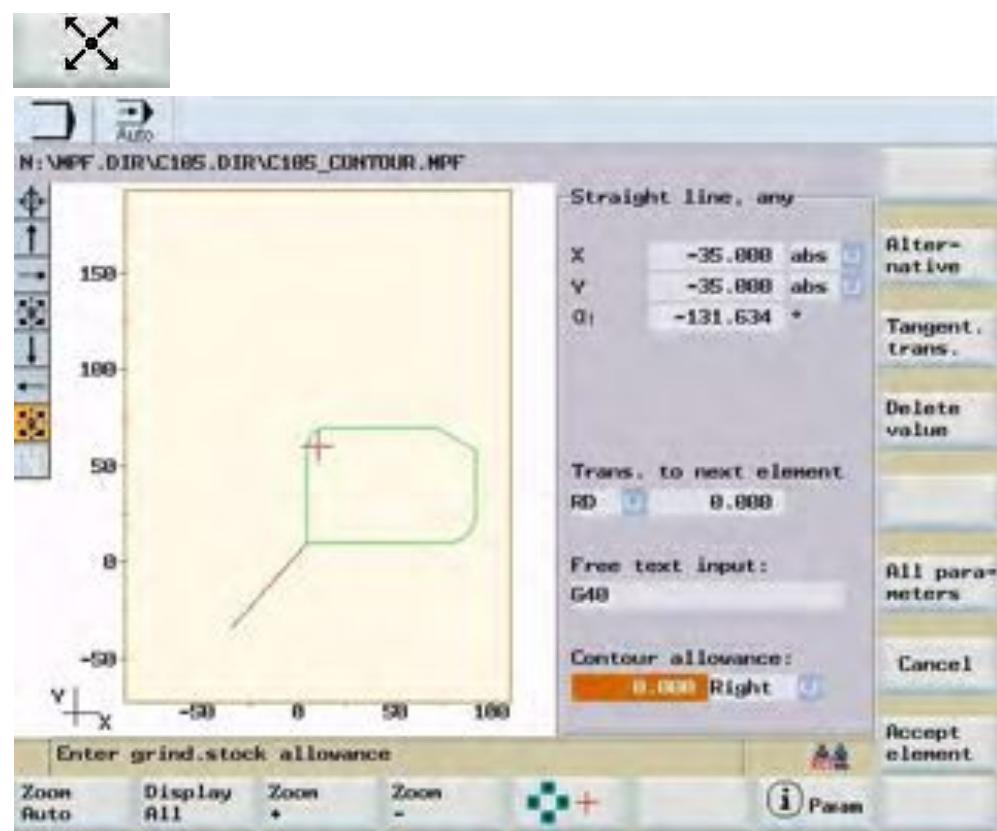
## Section 3

### Free Contour Programming by example

Notes



Accept element

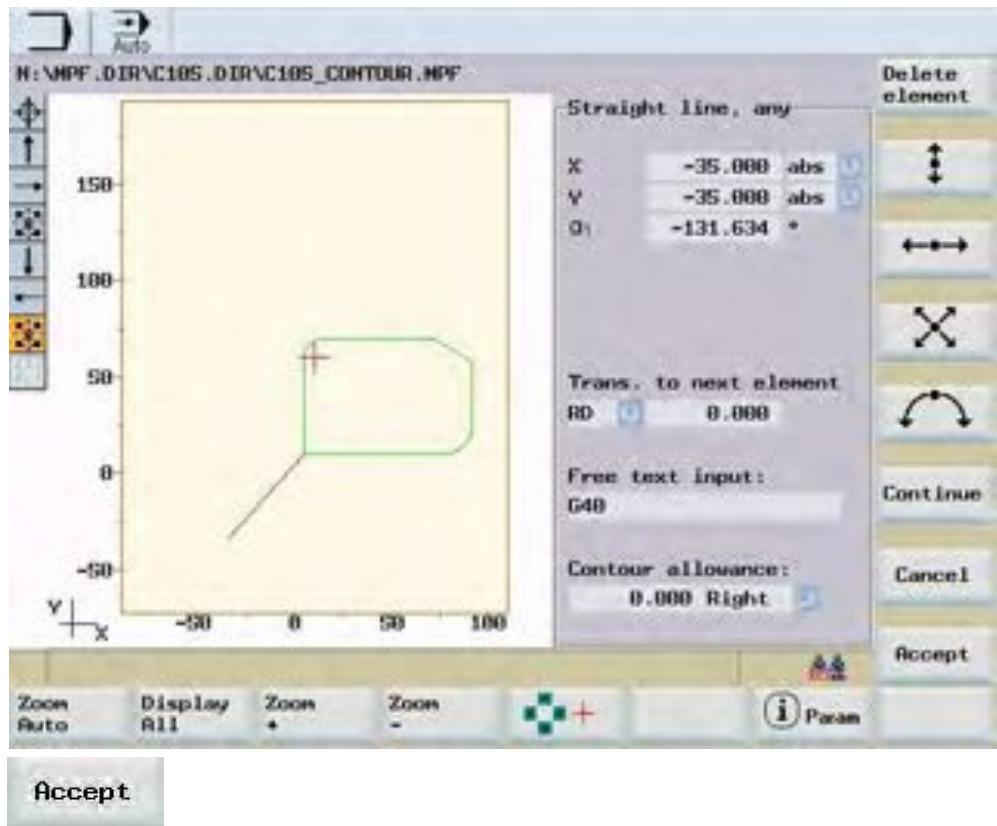


Accept element

# Section 3

# Free Contour Programming by example

Use the following sequence to enter the contour definition into the NC program.



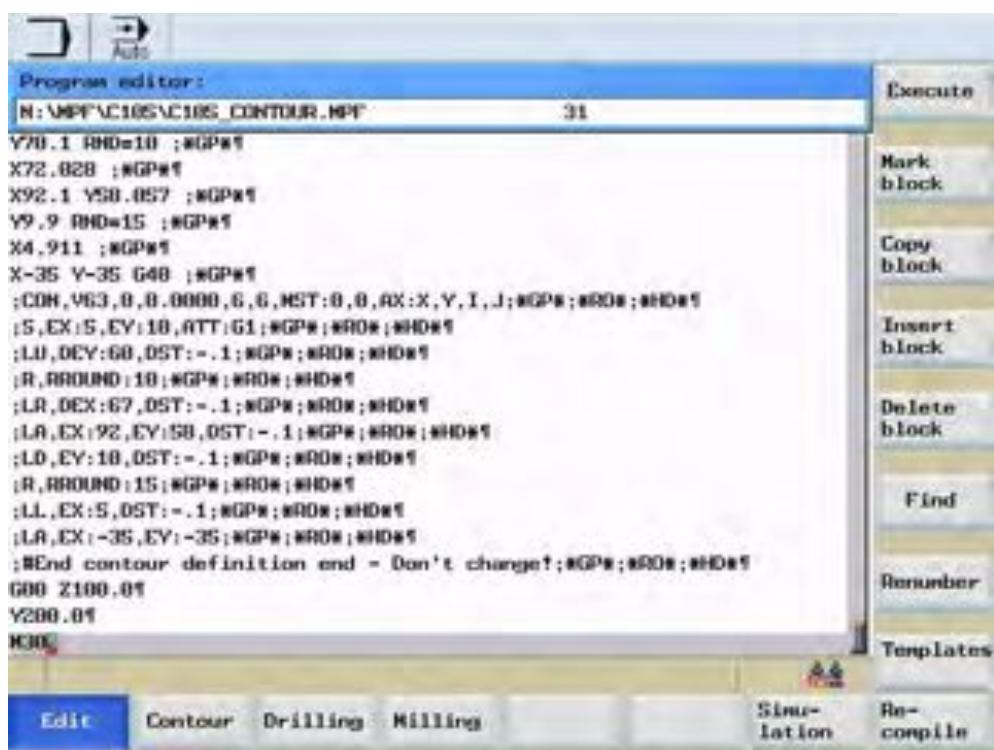
## Notes

## Section 3

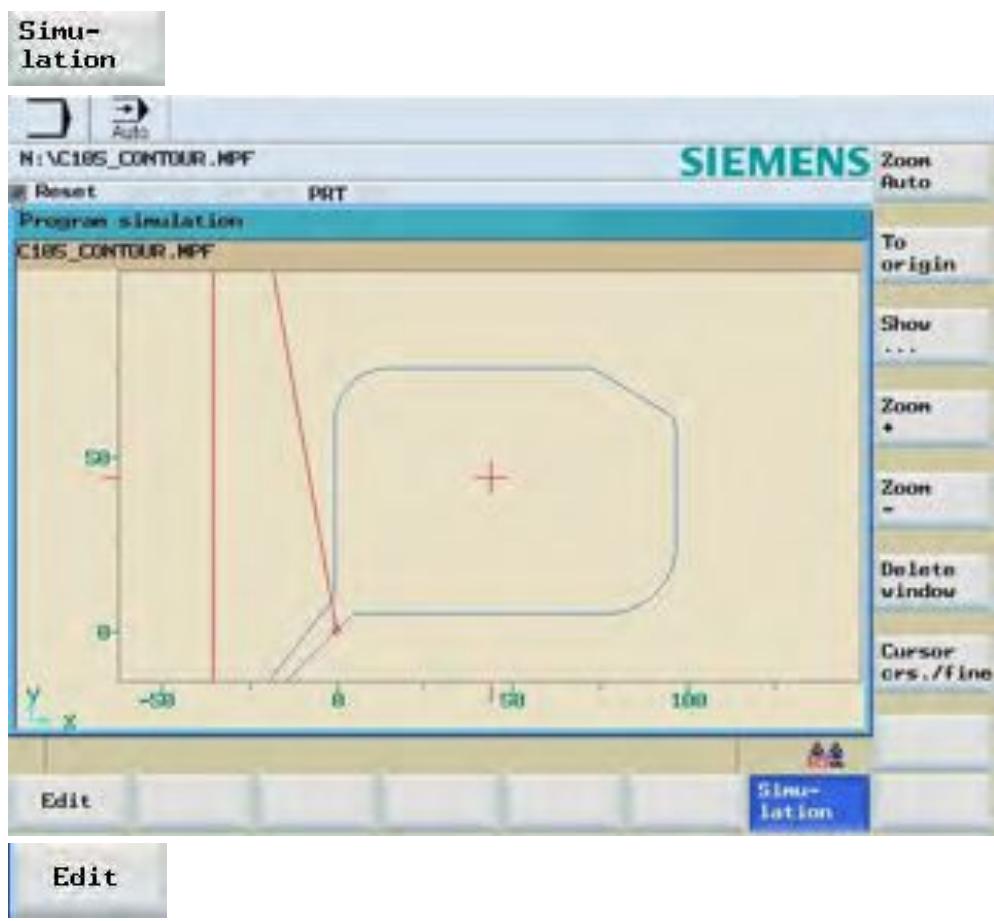
### Free Contour Programming by example

Notes

Add the end of the program.

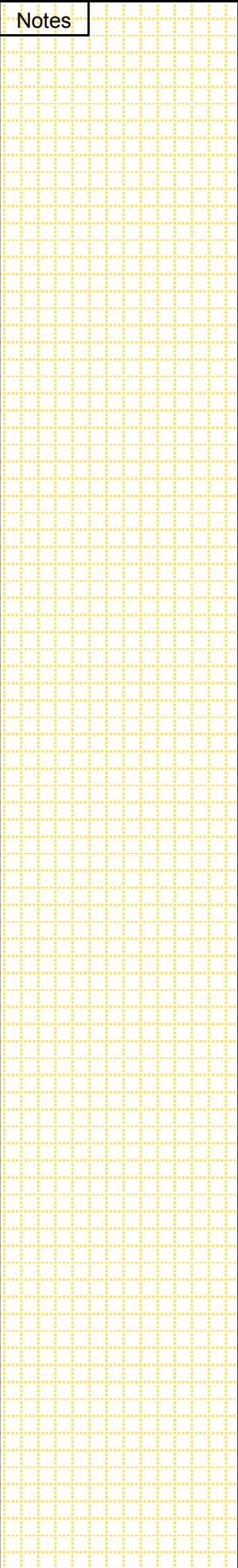


Check the program using Simulation.



---

Notes



## 1 Brief description

**Module objective:**

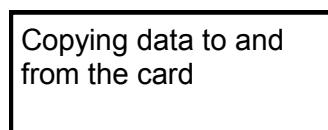
Upon completion of this module you can copy and paste programs between the CF card and the CNC controller.

**Module description:**

It is possible to use a CF (Compact Flash) card in the controller as additional storage for part programs or for storing backups of the machines commissioning data. This module describes the usage of the card in the control.

**Module content:**

Copying data to and from the card



## Section 2

### Copying data to and from the card

#### 2.1 Copying data to and from the card

All data can be saved to a CF card using copy and paste functionality. To use the feature a CF card should be inserted into the controller, as can be seen below:



Data can be transferred to the card in the “Program Manager” area of the control, also in the “System area”.



Program Manager

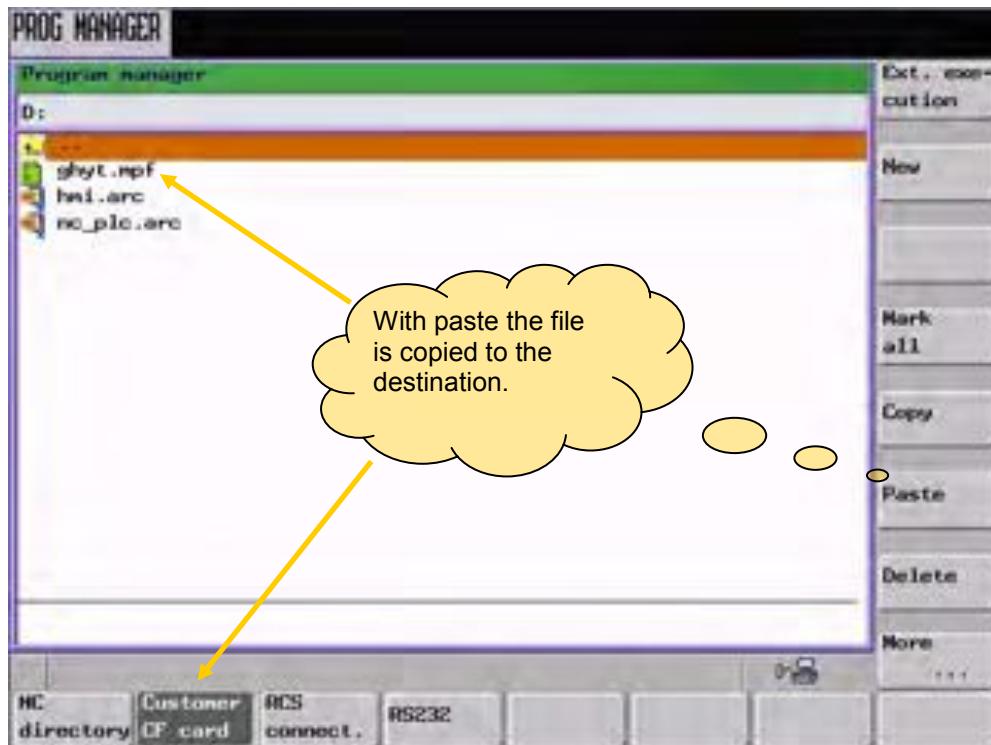
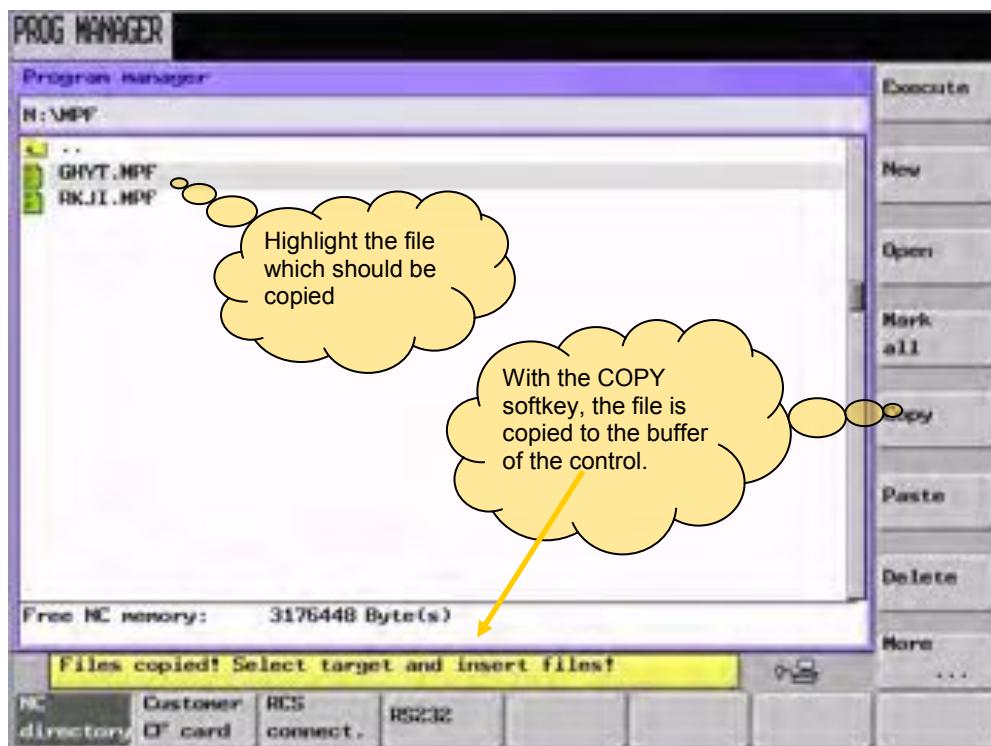
File functions  
CF Card

## Section 2

### Copying data to and from the card

To copy data to the card in the program manager perform the following, reverse the process to copy from card to NC.

Notes





## 1 Brief description

**Module objective:**

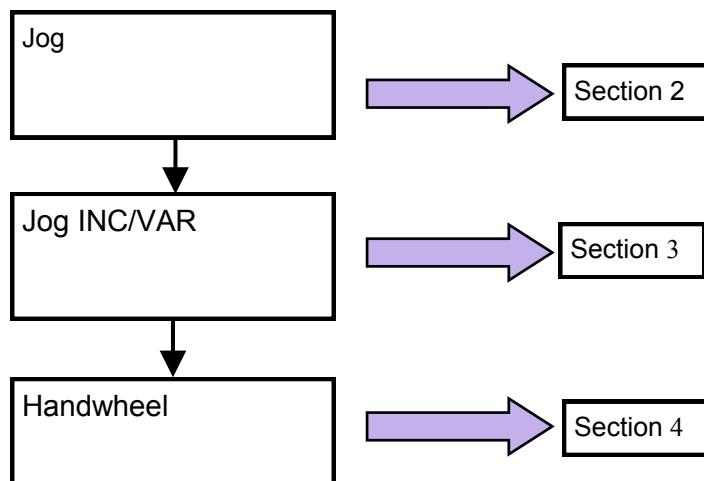
Upon completion of this module you can jog all axis within the confines of the machine, using three methods: Jog, Jog INC/VAR, Handwheel

**Module description:**

We use Axis Control/Jog to move axis, that we can setup tools, change tools / inserts, and gain approved access to the workpiece.

**Module content:**

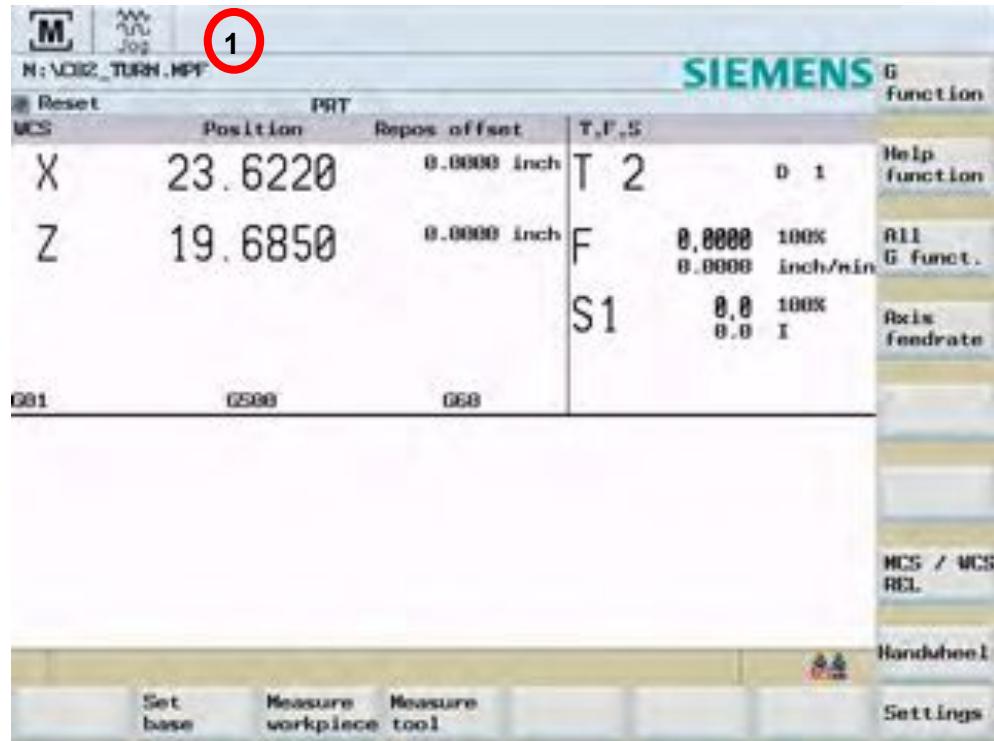
Jog  
Jog INC/VAR  
Handwheel



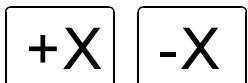
## Section 2

### Jog

To JOG the axis freely around within the confines of the machine, follow this sequence.



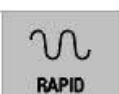
- 1 In the STATUS area the control will show you that it is in JOG mode



To Jog the axis use the push button for the required axis and the direction to be moved:



If you use the **rapid override** button at the same time as the selected axis, the axis will move at rapid speed until the buttons are released.



The use of the feed override switch allows you to regulate the feed of the chosen axis.

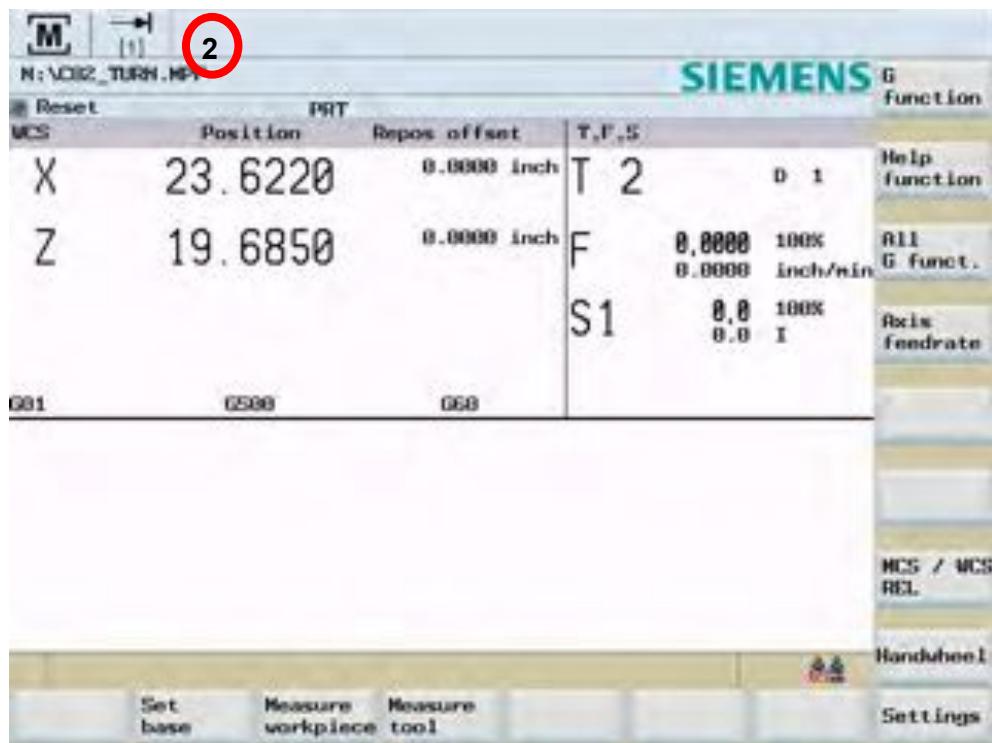
Notes

## Section 3

### Jog 1 INC

To INCREMENTALLY move the axis around within the confines of the machine, follow this sequence.

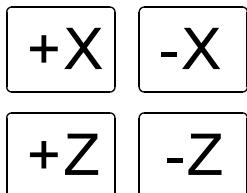
Notes



2

In the STATUS area the control will show you what INC is being used.

Depending on how many time you press the [VAR] button



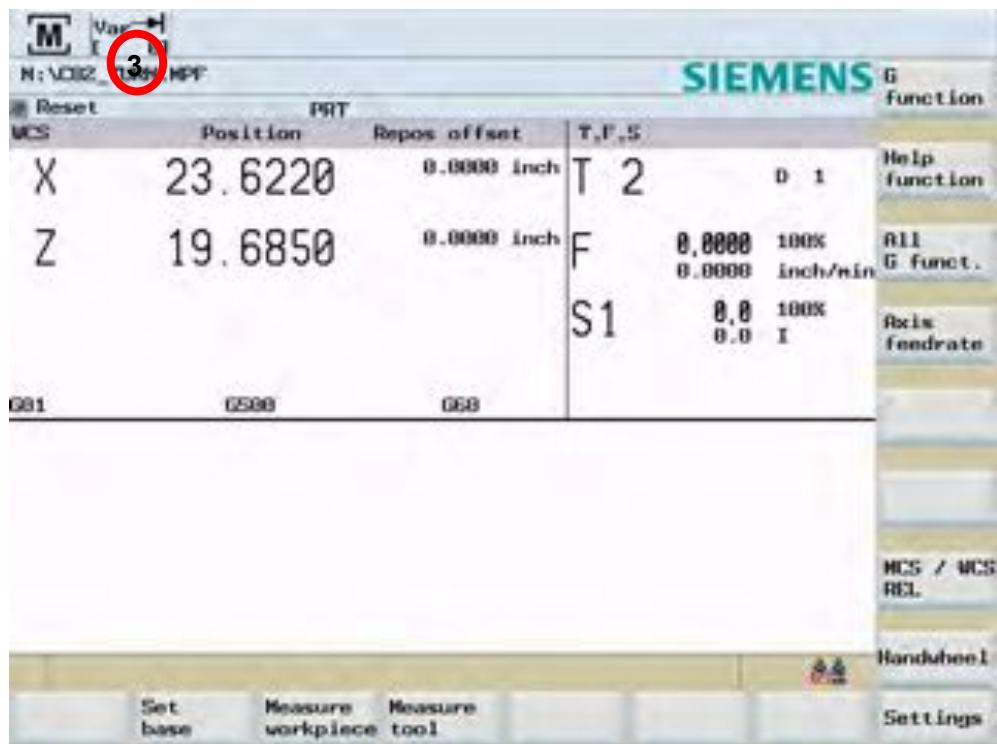
Use the axis pushbutton for the required axis, and direction to be moved. When pressed the chosen axis will move by 1 increment per press.

## Section 3

### Jog VAR

Notes

To INCREMENTALLY move the axis around within the confines of the machine, follow this sequence.



- 3 In the STATUS area the control will show you that it is in JOG VAR mode.

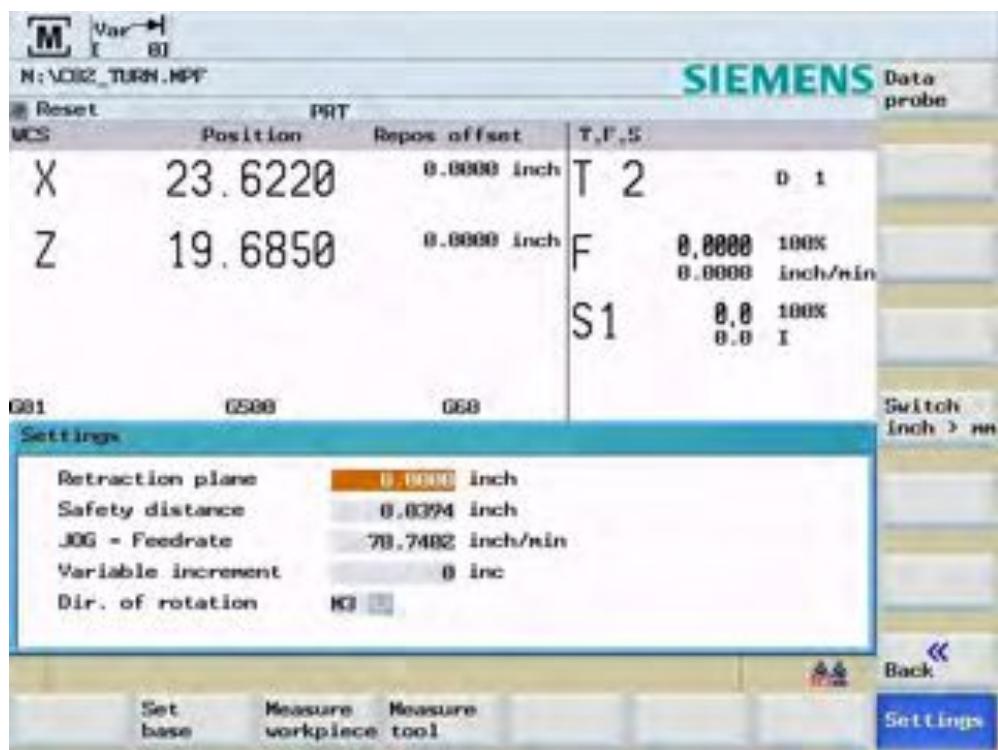
Use the SETTINGS softkey to set the VARIABLE value



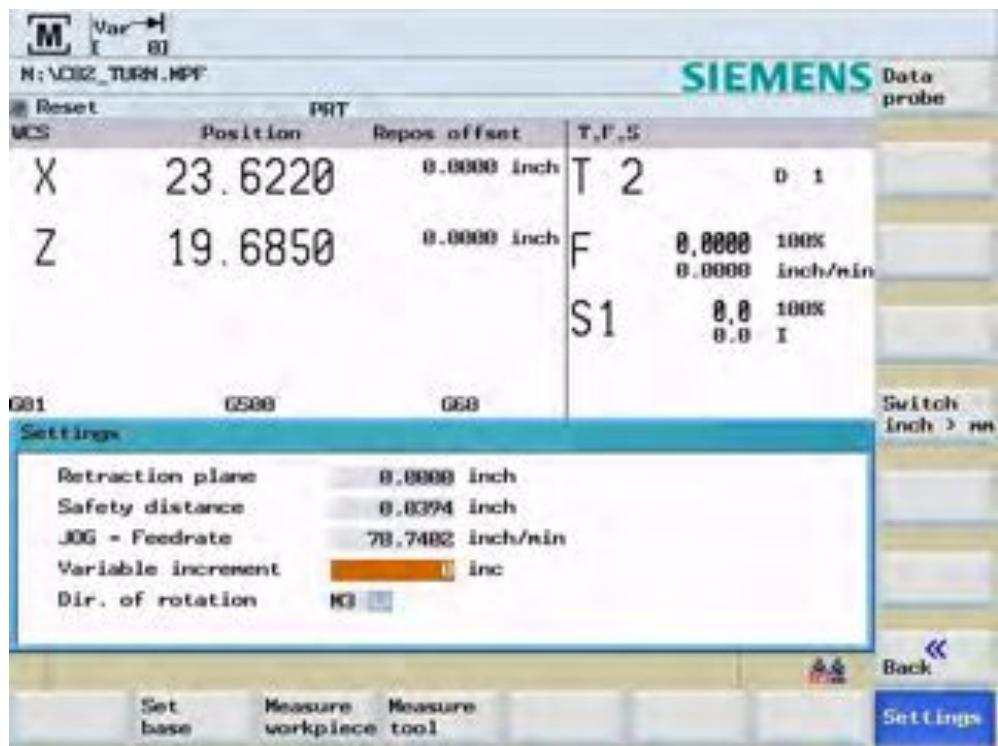
## Section 3

### Jog VAR

Notes



Cursor down to VARIABLE INCREMENT using button.



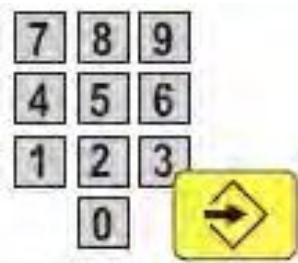
## Section 3

### Jog VAR

Enter the value for VAR INC

i.e. 0.5 increments

Type **500** using the numerical buttons



Followed by



The screenshot shows the SIMATIC Manager software interface. In the 'Settings' menu, the 'Variable increment' field is highlighted with a red circle and contains the value '0.05 inc'. Other settings shown include Retraction plane (0.0000 inch), Safety distance (0.0394 inch), JOG = Feedrate (79.7482 inch/min), and Dir. of rotation (K3). The status bar at the bottom left shows 'Var [ 500 ]'.

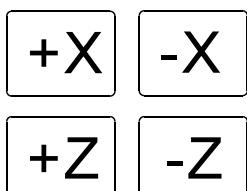
4

The value that has been entered will be shown here.



Note: the status area shows that "Incremental Variable 500" has now been set.

When you use the push buttons, the axis will JOG at **0.05** increment per press.



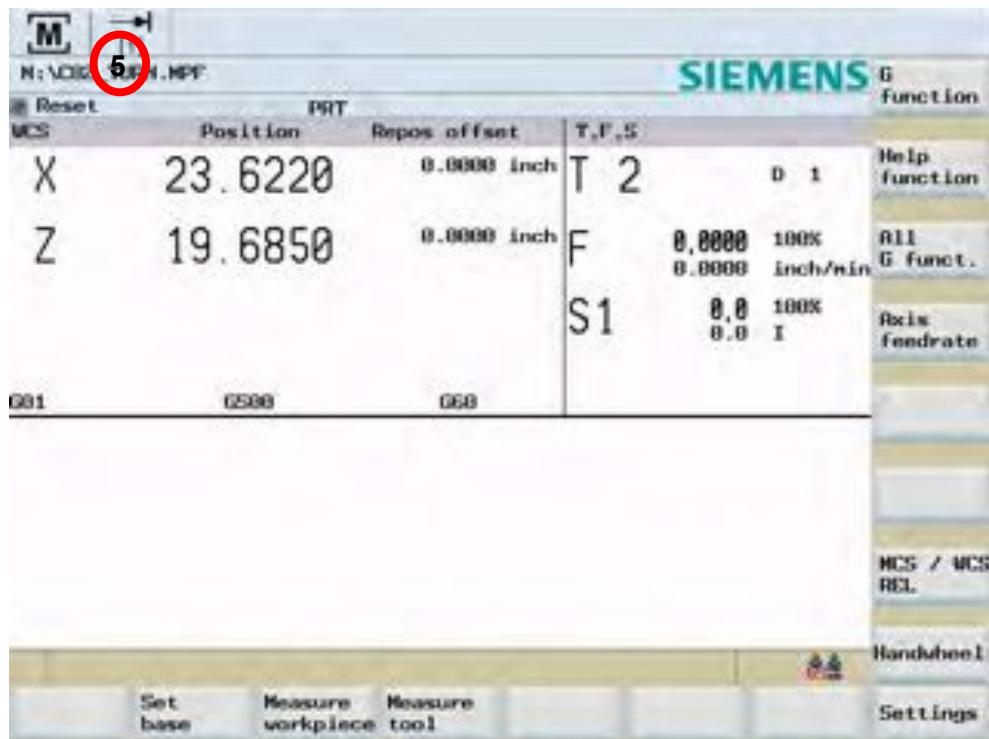
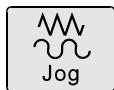
Notes

## Section 4

### Hand wheel

Notes

To move the axis around within the confines of the machine, using the hand wheel follow this sequence.



5

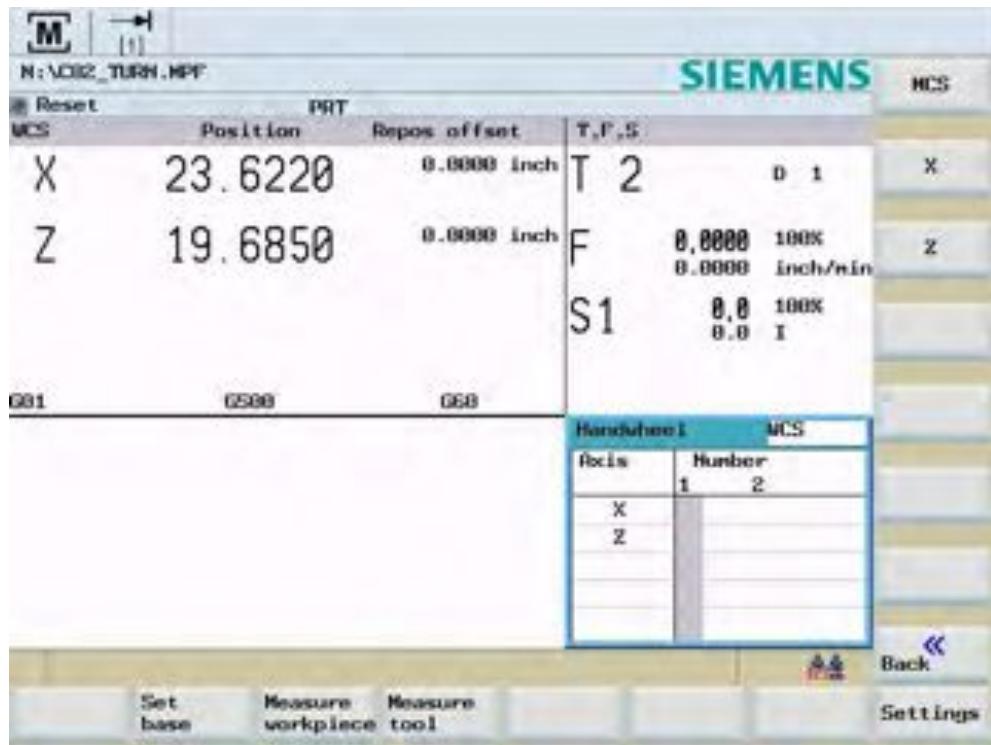
In the STATUS area the control will show you what INC is being used.  
Depending on how many time you press the [VAR] button

Handwheel

## Section 4

### Hand wheel

Notes



Using the soft keys

X

Z

Select the axis you wish to move using the hand wheel

i.e. **X axis**

X

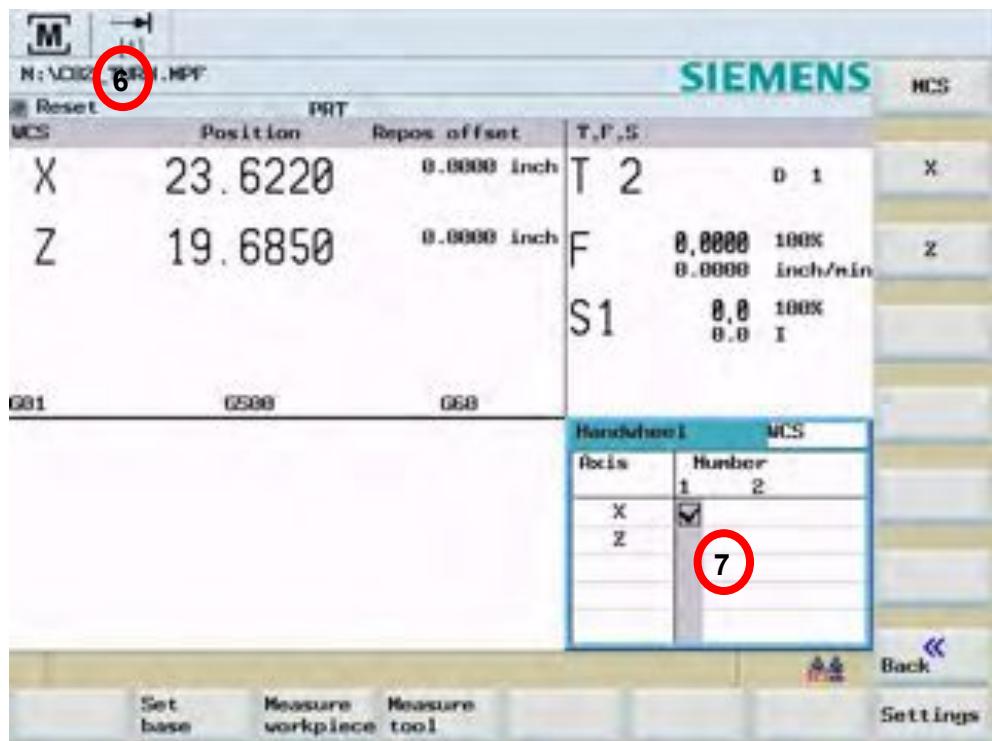
A tick box will now be shown by the side of the axis, in the **hand wheel** window.



## Section 4

### Hand wheel

Notes



6 In the STATUS area the control will show you what INC is being used.

7 In the Hand wheel area the control will show you which axis has been selected.

When the hand wheel is used the X axis will move at 0.001mm per increment.

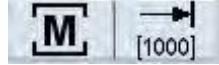
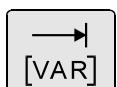


## Section 4

### Hand wheel

Notes

- 8 To change the INCREMENT for the hand wheel, use the following button.



- 9 To change the axis use the soft keys.



The screenshot shows the Siemens SINUMERIK 802D SI control interface. At the top, there is a status bar with the file path "N:\V002\_TURB.HPF". Below it is a tool palette with icons for "Reset", "Position", "Repos offset", "T,F,S", and "HCS". The main area displays a coordinate system with X=23.6220 and Z=19.6850. A sub-menu titled "Handwheel" is open, showing a table with two rows:

Axis	Number
X	1
Z	2

The "Z" row is highlighted with a red circle containing the number "9". At the bottom of the screen, there are buttons for "Set base", "Measure workpiece", "Measure tool", "Back", and "Settings".

## Section 4

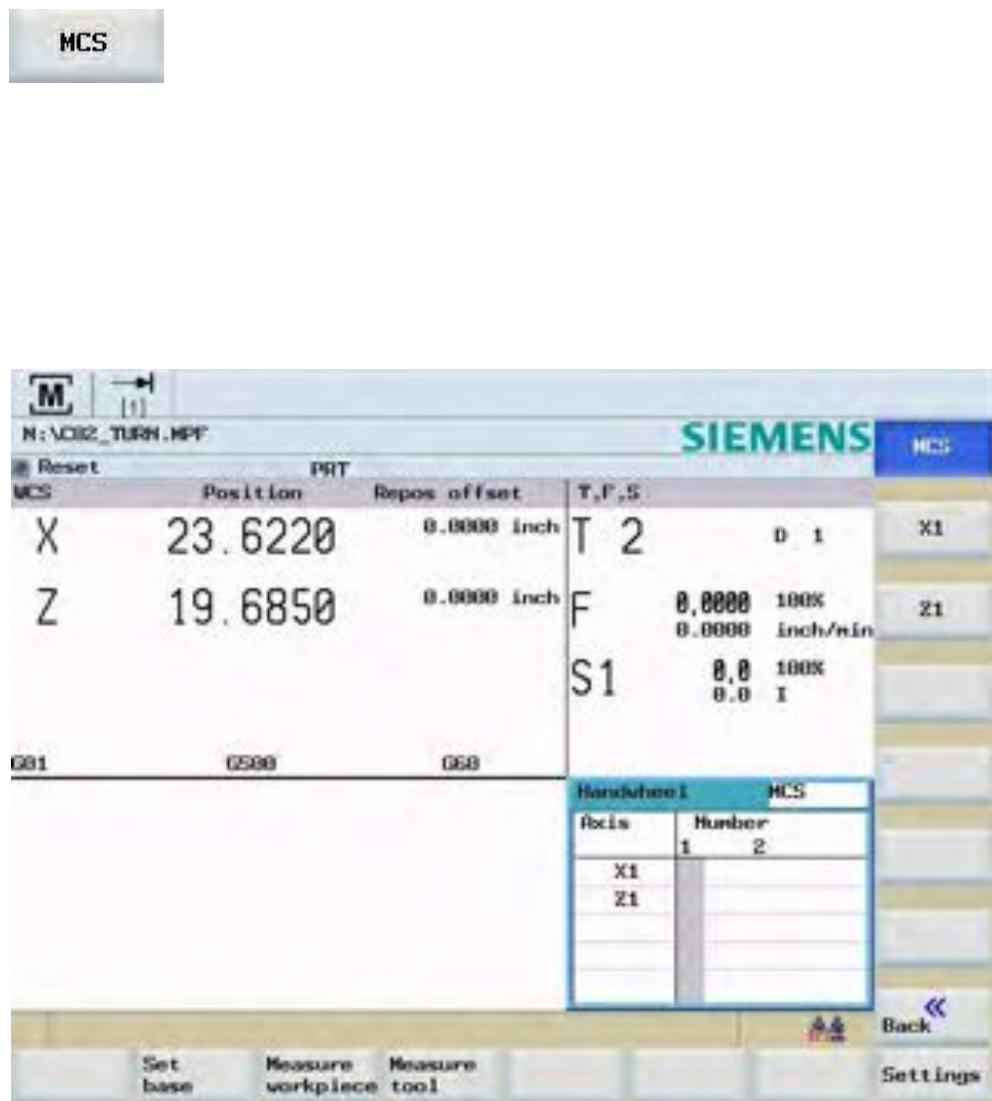
### Handwheel

Notes

You can use the hand wheel in MCS mode, which allows you to move the axis with the figures on the screen relative to the machine and not the work piece.

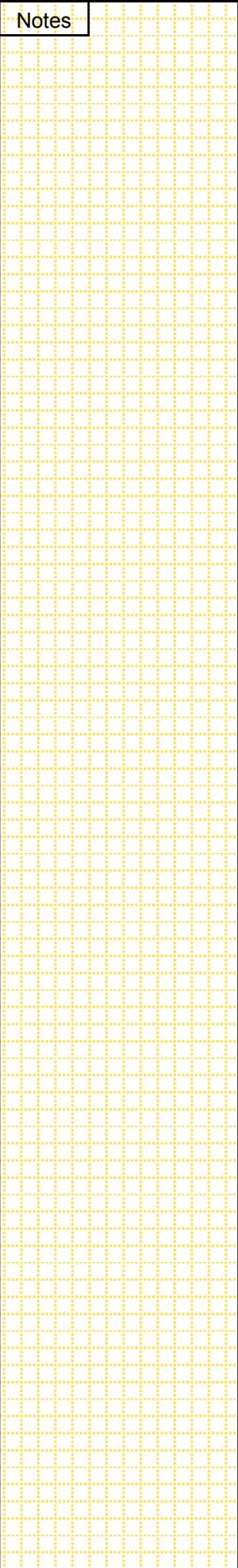
This will show the axis as  
X1  
Z1

Use the following softkey.



---

Notes



## 1 Brief description

**Module objective:**

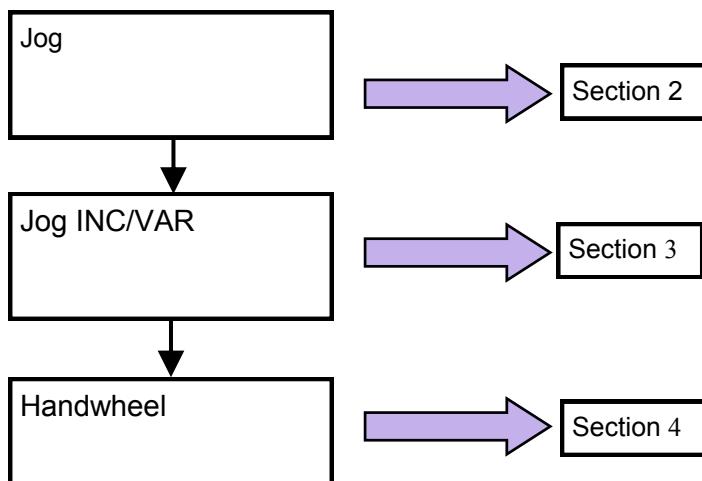
Upon completion of this module you can jog all axis within the confines of the machine, using three methods: Jog, Jog INC/VAR, Handwheel

**Module description:**

We use Axis Control/Jog to move axis, that we can setup tools, change tools / inserts, and gain approved access to the workpiece.

**Module content:**

Jog  
Jog INC/VAR  
Handwheel

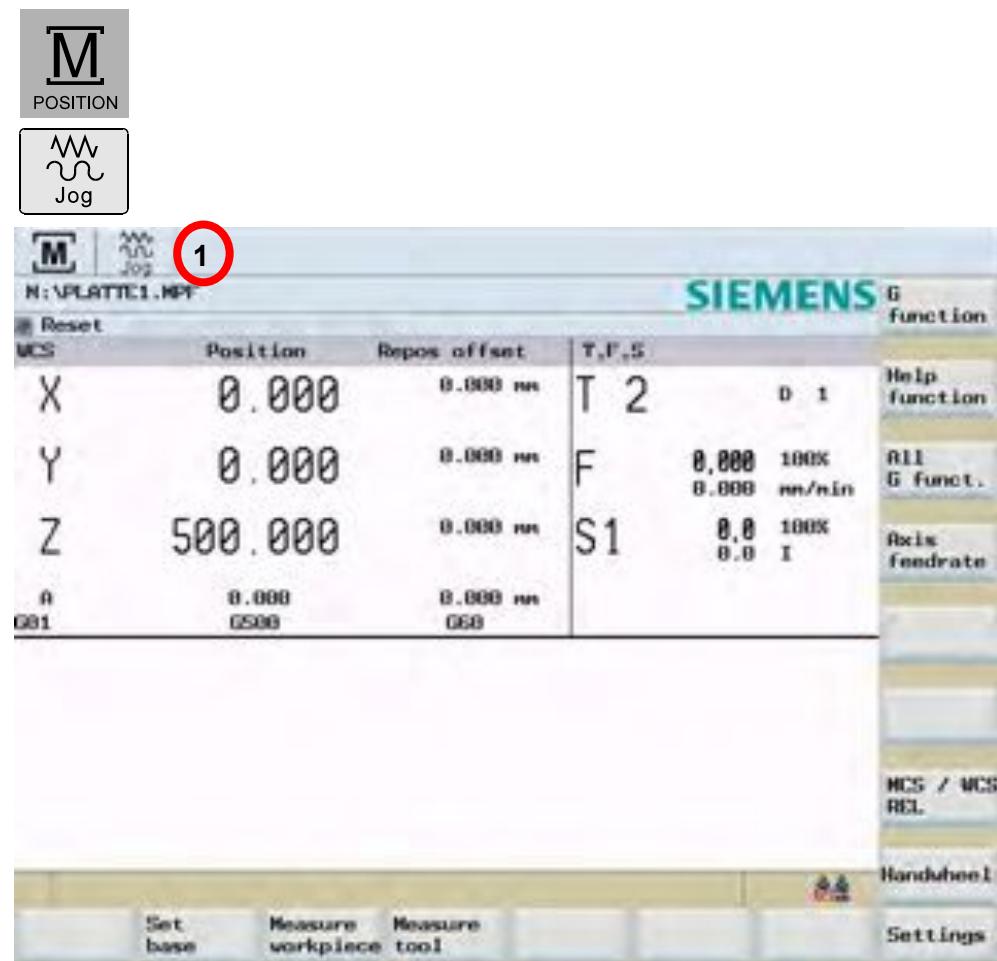


## Section 2

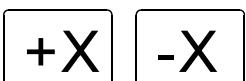
### Jog

To JOG the axis freely around within the confines of the machine, follow this sequence.

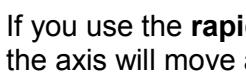
Notes



- 1 In the STATUS area the control will show you that it is in JOG mode



To Jog the axis use the push button for the required axis and the direction to be moved:



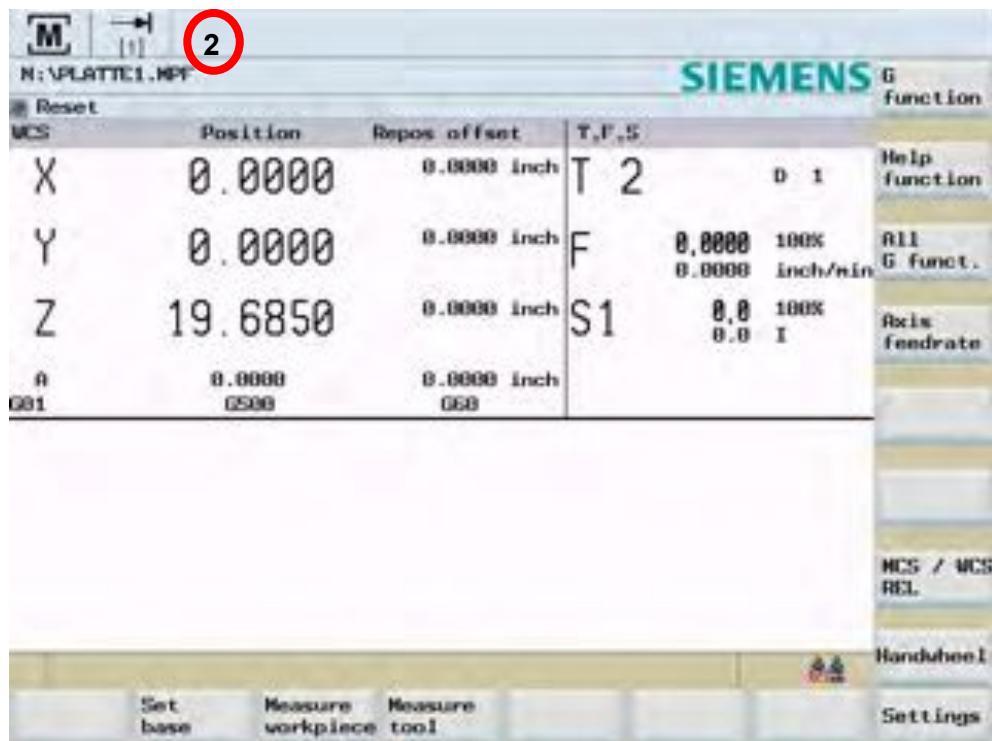
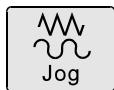
The use of the feed override switch allows you to regulate the feed of the chosen axis.

## Section 3

### Jog 1 INC

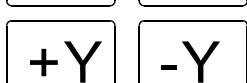
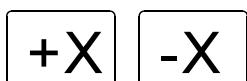
To INCREMENTALLY move the axis around within the confines of the machine, follow this sequence.

Notes



2

- In the STATUS area the control will show you what INC is being used.  
Depending on how many time you press the [VAR] button



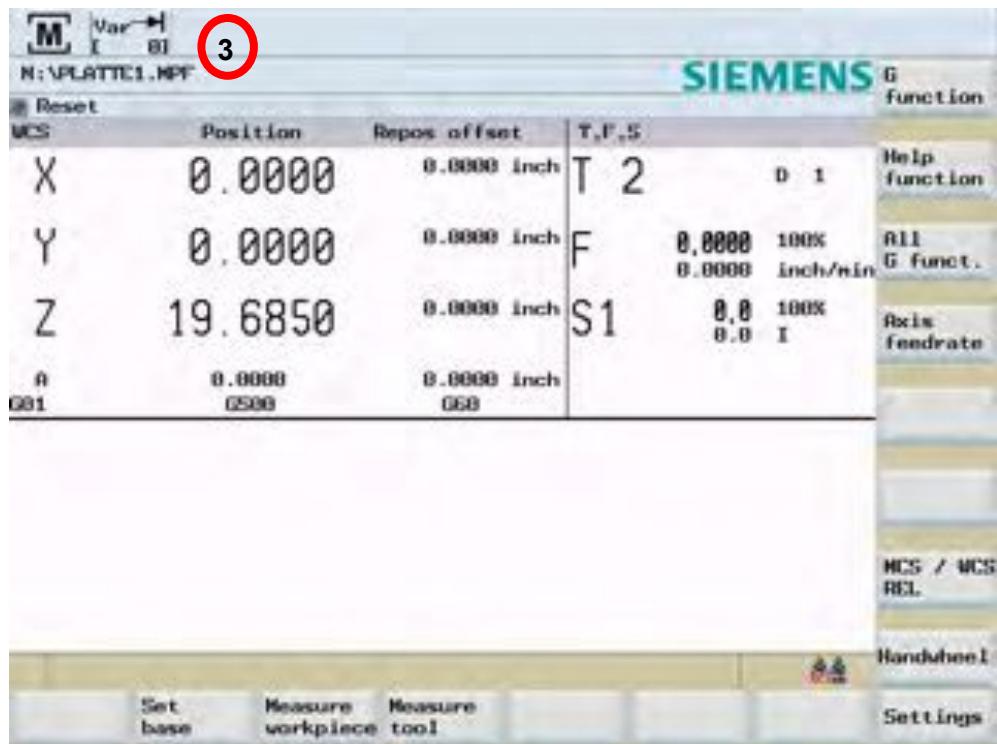
Use the axis pushbutton for the required axis, and direction to be moved. When pressed the chosen axis will move by 1 increment per press.

## Section 3

### Jog VAR

Notes

To INCREMENTALLY move the axis around within the confines of the machine, follow this sequence.



- 3 In the STATUS area the control will show you that it is in JOG VAR mode.

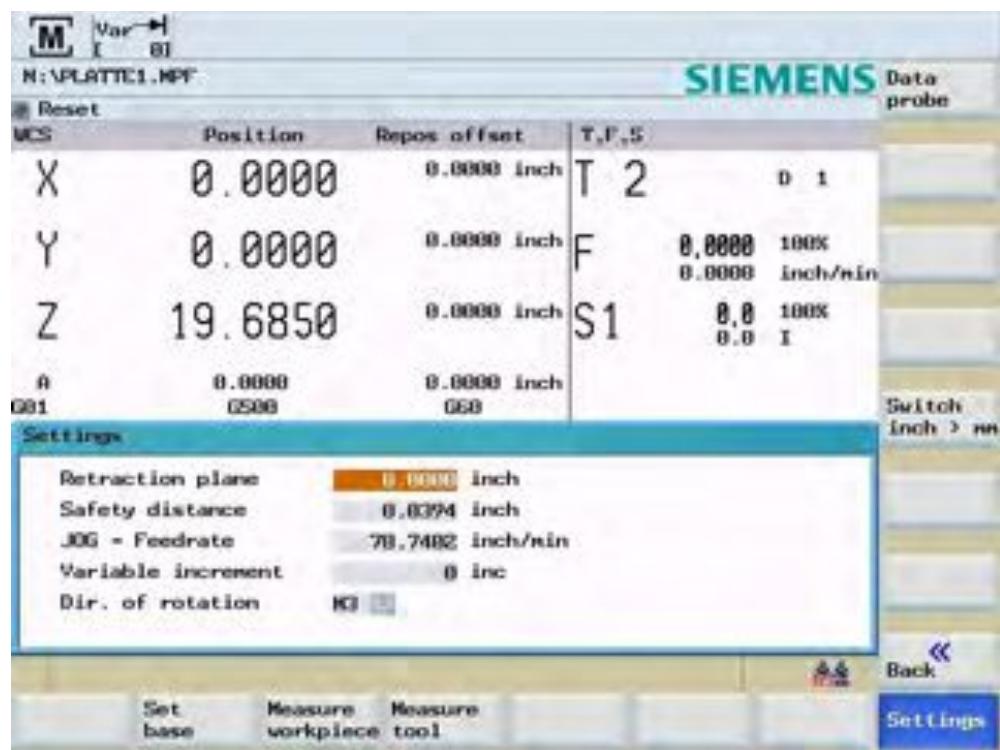
Use the SETTINGS softkey to set the VARIABLE value

Settings

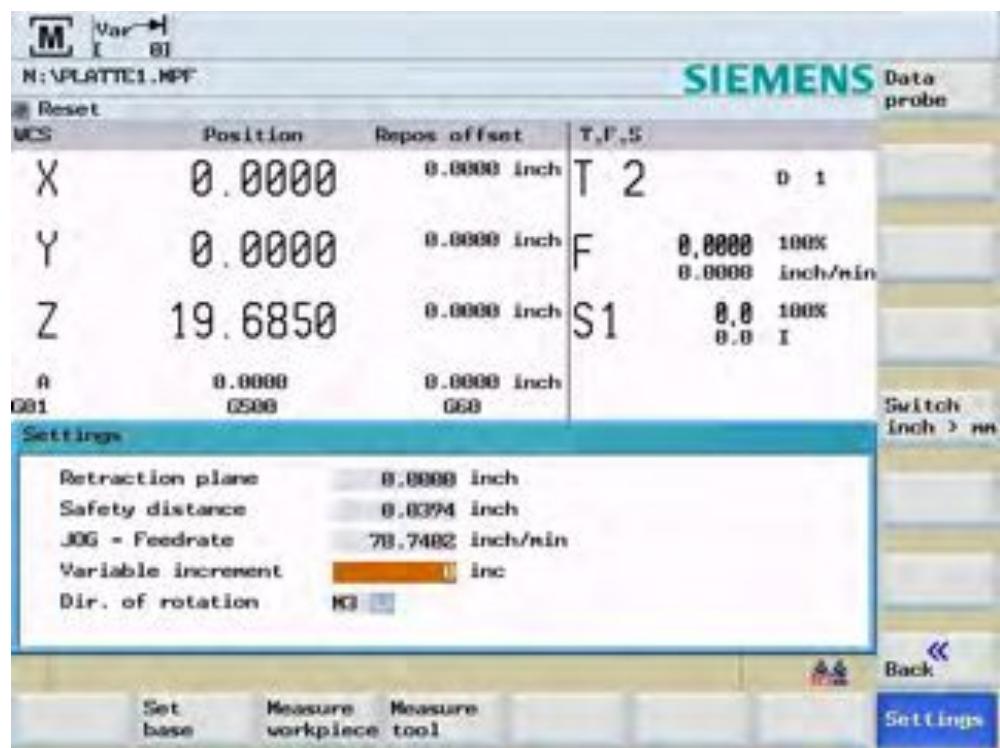
## Section 3

### Jog VAR

Notes



Cursor down to VARIABLE INCREMENT using button.



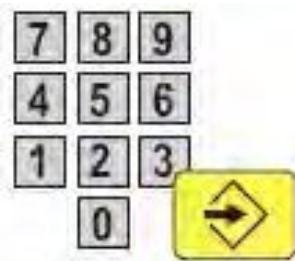
## Section 3

### Jog VAR

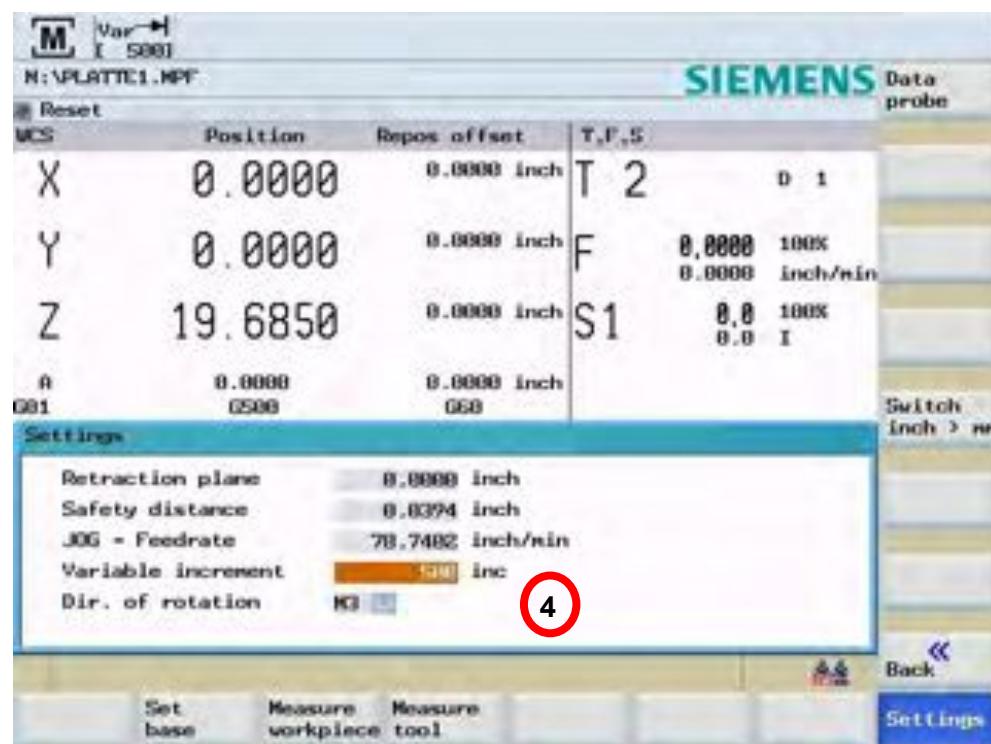
Enter the value for VAR INC

i.e. 0.5 increments

Type **500** using the numerical buttons



Followed by



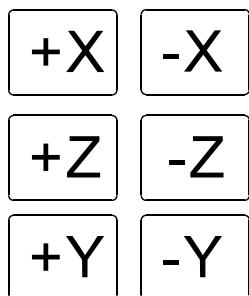
4

The value that has been entered will be shown here.



When you have entered the data in SETTINGS  
press the soft key BACK.

When you use the push buttons, the axis will JOG at **0.05** increment per press.



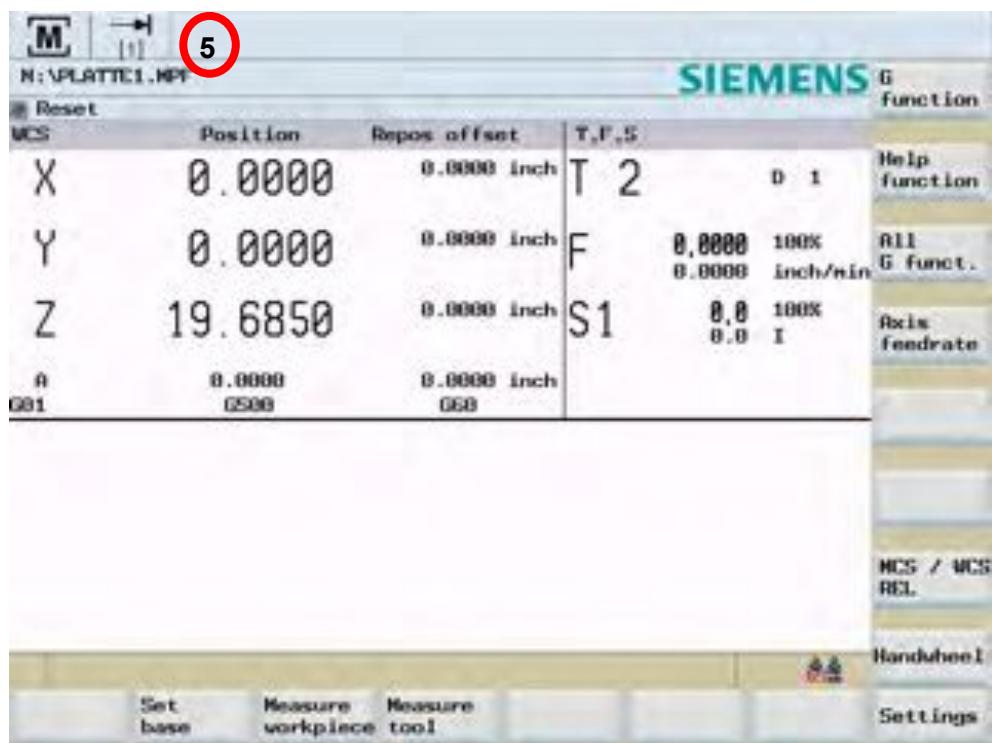
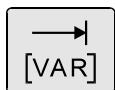
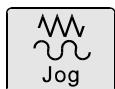
Notes

## Section 4

### Hand wheel

Notes

To move the axis around within the confines of the machine, using the hand wheel follow this sequence.



5

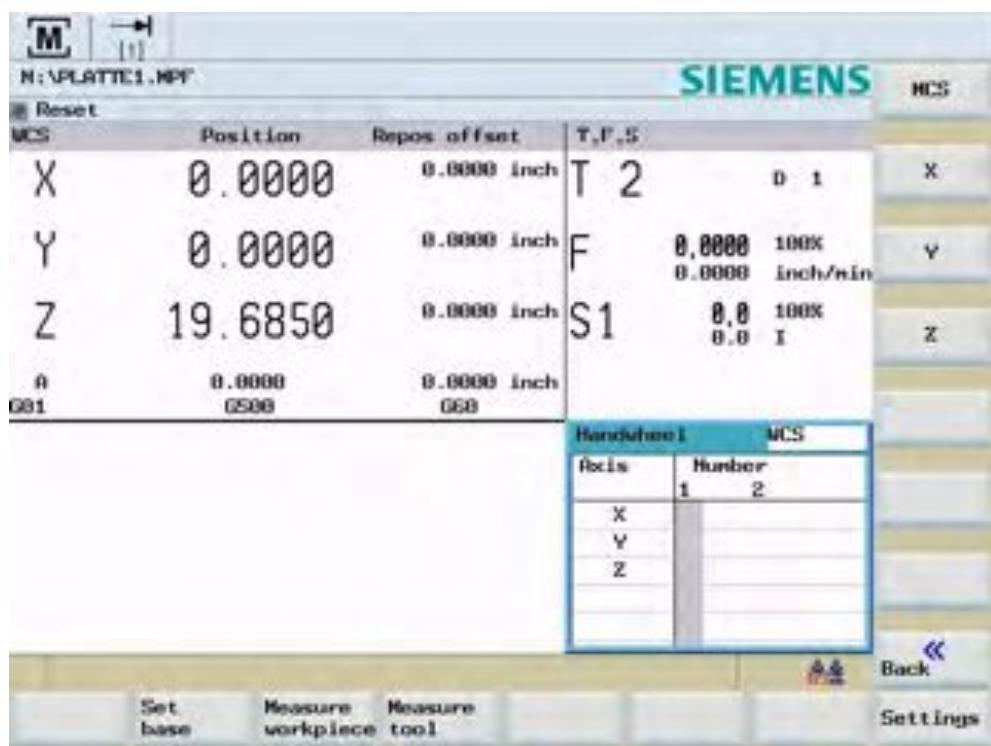
- In the STATUS area the control will show you what INC is being used.  
Depending on how many time you press the [VAR] button

Handwheel

## Section 4

### Hand wheel

Notes



Using the soft keys

X

Y

Z

Select the axis you wish to move using the hand wheel

i.e. **X axis**

X

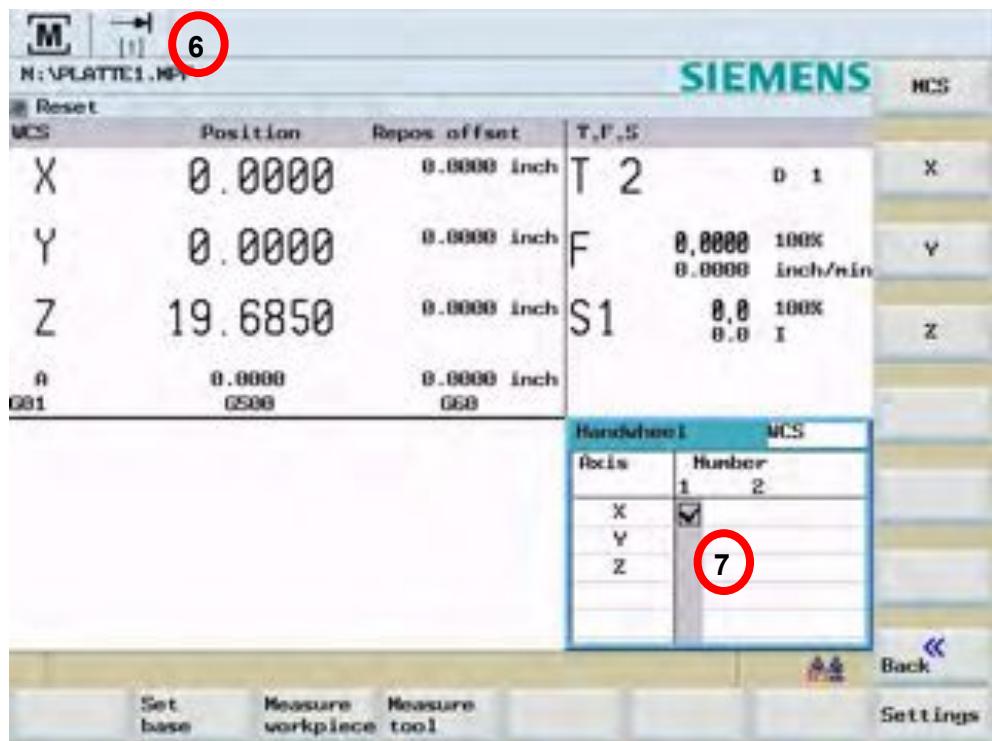
A tick box will now be shown by the side of the axis, in the **hand wheel** window.



## Section 4

### Hand wheel

Notes



6 In the STATUS area the control will show you what INC is being used.

7 In the Hand wheel area the control will show you which axis has been selected.

When the hand wheel is used the X axis will move at 0.001mm per increment.

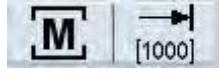
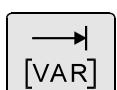


## Section 4

### Hand wheel

Notes

- 8 To change the INCREMENT for the hand wheel, use the following button.



- 9 To change the axis use the soft keys.



The screenshot shows the Siemens SINUMERIK 802D sl software interface. At the top, there is a menu bar with 'File', 'Edit', 'View', 'Tool', 'Position', 'Axis', 'MCS', 'Help', and 'About'. Below the menu is a toolbar with icons for 'Set base', 'Measure workpiece', 'Measure tool', and 'Settings'. The main area displays a table of current values:

MCS	Position	Repos offset	T,F,S	MCS
X	0.0000	0.0000 inch	T 2 D 1	X
Y	0.0000	0.0000 inch	F 0.0000 100%	Y
Z	19.6850	0.0000 inch	S1 0.0 100%	Z
A	0.0000	0.0000 inch		
G91	0.0000	0.0000 inch		

To the right of the table is a 'Handwheel' MCS panel with a table:

Axis	Number
X	1
Y	2
Z	9

The 'Z' row in this table is circled in red, indicating it is selected. The bottom of the screen features a footer with 'Back' and 'Settings' buttons.

## Section 4

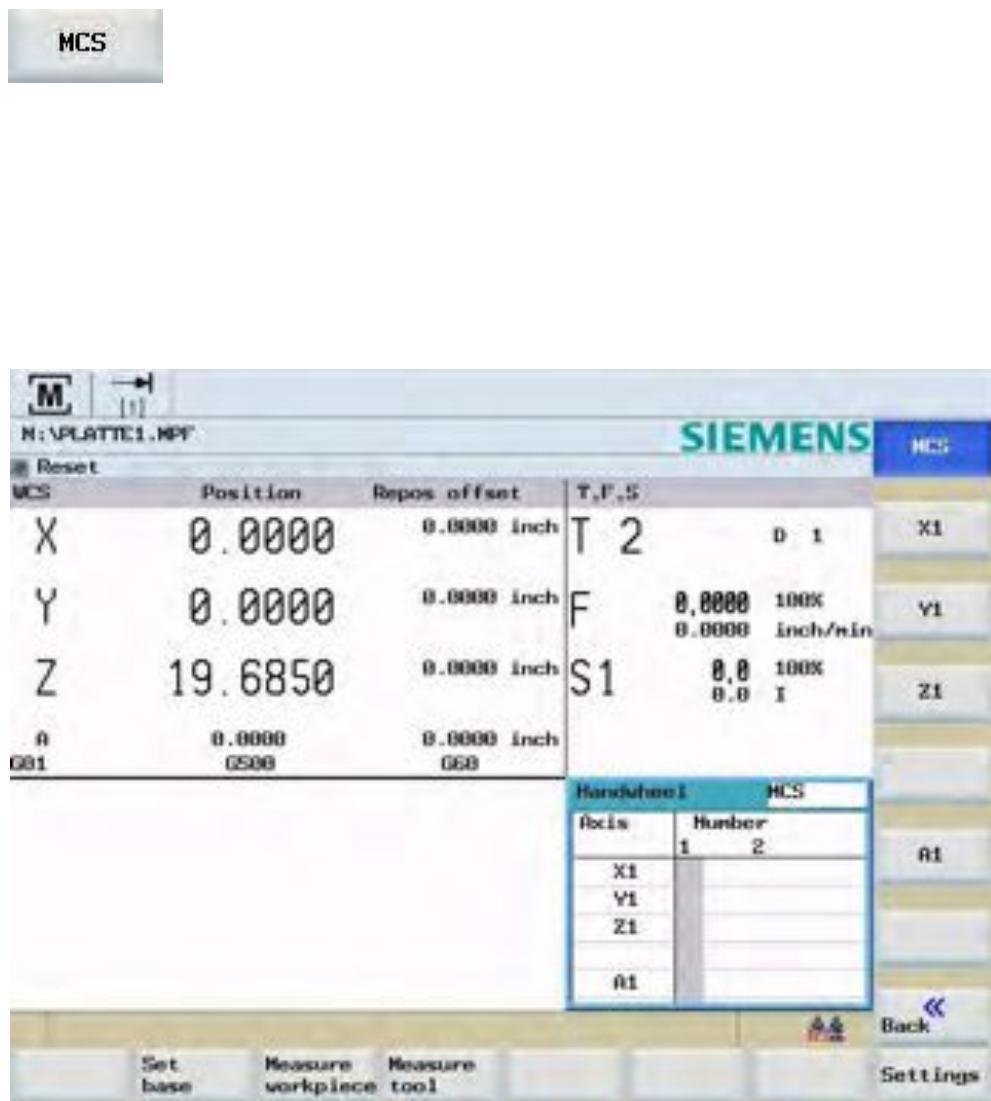
### Handwheel

Notes

You can use the hand wheel in MCS mode, which allows you to move the axis with the figures on the screen relative to the machine and not the work piece.

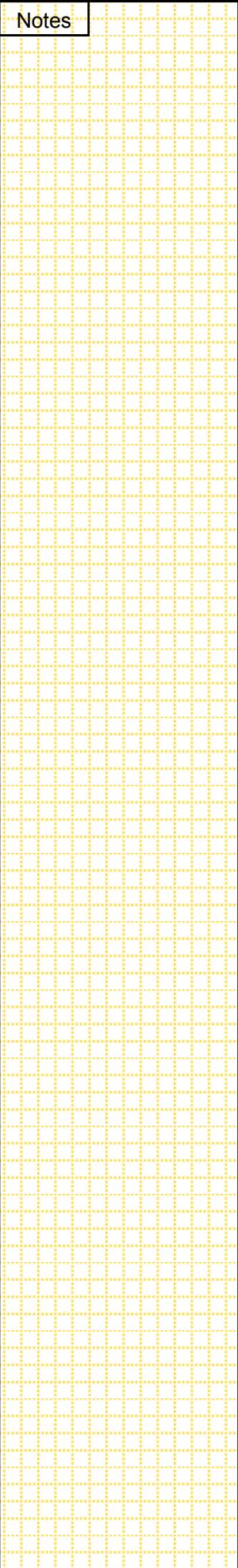
This will show the axis as  
X1  
Y1  
Z1

Use the following softkey.



---

Notes



## 1 Brief description

**Module objective:**

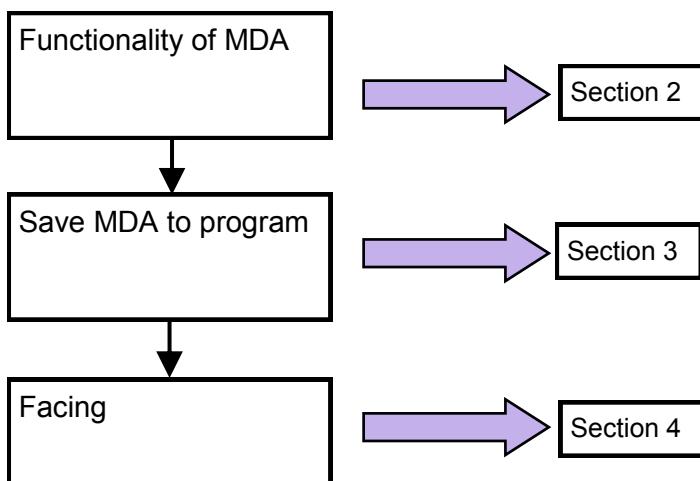
Upon completion of this module you can execute commands in the MDA mode

**Module description:**

You can enter program commands in MDA and immediately execute them. This allows the operator to activate functions on the machine normally for setting up purposes outside the NC program

**Module content:**

Functionality of MDA  
Save MDA to program  
Facing



## Section 2

### Functionality of MDA

The following functions are possible in MDA mode, executing immediately program blocks in the machine area. It is possible to execute M functions, especially for tool changing or activating clamping devices outside of the NC program.

You can activate MDA in the following sequence:



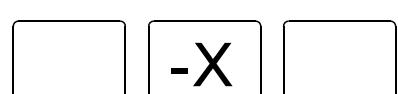
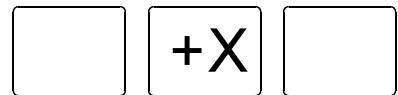
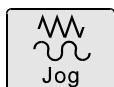
To scratch on the workpiece (Set zero offset, or measure tools manually)  
Jog and MDA can be used with each other.

In MDA you can then call the tool with spindle speed and direction, being careful not to program M02 as this will have the effect of resetting the given spindle speed and direction.

After NC start you can return to Jog mode and position the axis accordingly.

Example shows text typed into the MDA area and sequence for moving the axis:

**T1 D1  
G95 S200 M4**



Notes

## Section 3

### Save MDA to program

Notes

The content of the MDA buffer can be saved as an NC program, this can later be used again.

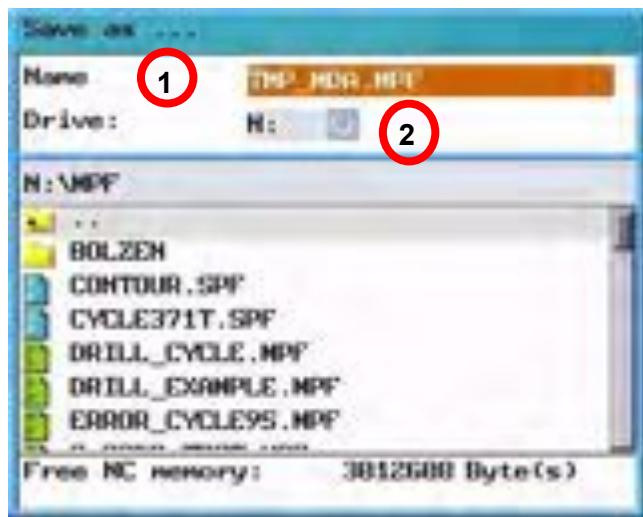
Pre-conditions:

- MDA-mode is active
- Program blocks exist in the buffer.

To save MDA to a program select the following sequence:

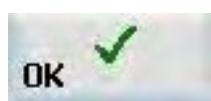


Save  
MDI prog.



- 1 In the window "Save as" you can input the name of the program in which the blocks will be saved.
- 2 The destination can be determined with the Select key (Drive N:), the CF-card (Drive D:) or the RCS-connection can be selected.

The data will be saved with the following sequence:

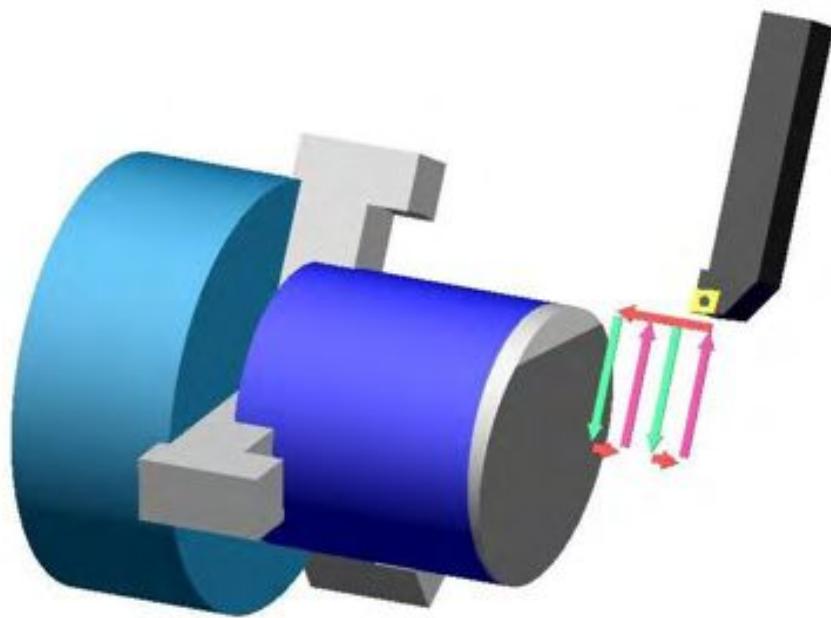


## Section 4

### Facing

Notes

With the facing function in MDA, the face or periphery of the workpiece can be prepared for further operations.



The following example shows the sequence to perform this task:



The retract plane and safety distance should be entered in the Settings dialog.

Settings



3

"Variable increment" This parameter should be kept to zero when not being used.

## Section 4

### Facing

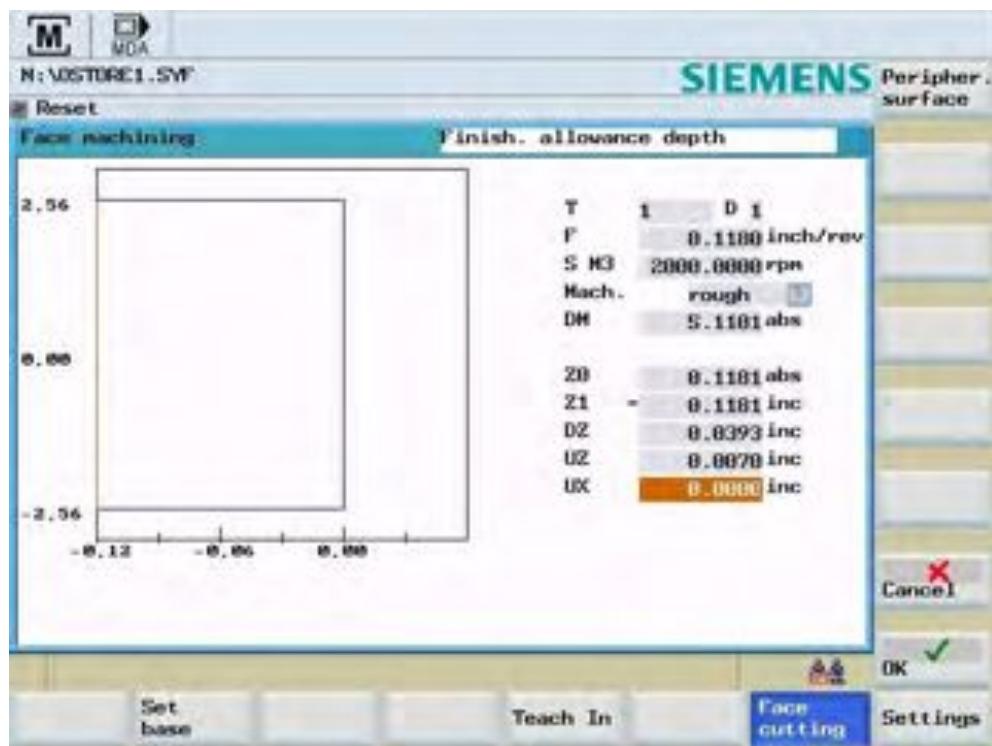
Notes

You can return to the previous page with the following sequence:

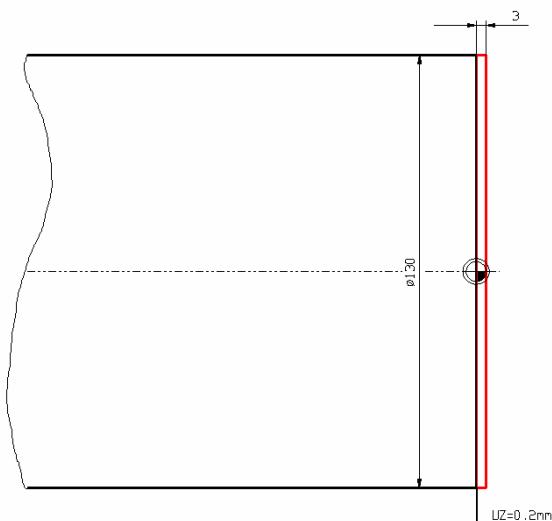


Face  
cutting

After parametrising the setting data, you can parameterise the MDA Facing cycle with the following data:



The values in the above dialog are for the following example:



## Section 4

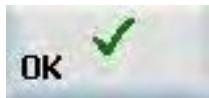
### Facing

Notes

Description of the parameters in the FACE TURNING working window.

Parameter	Explanation
<b>Tool</b>	Selected tool to be used in the working cycle.
<b>Feed F</b>	Selected path feedrate, in mm/min or mm/rev.
<b>Spindle S rpm</b>	selected spindle speed
<b>Mach</b>	Definition of the surface quality. You can select between roughing or finishing.
<b>Diameter DN</b>	input of the blank diameter of the part.
<b>Z0</b> Blank dimension	Input of the Z position.
<b>Z1</b> Cutting dimension	Cutting dimension, incremental.
<b>DZ</b> Cutting dimension	Input of the cutting length in the Z direction. This dimension is always specified in increments and is referred to the workpiece edge.
<b>UZ</b> Max. infeed	Stock allowance in the Z direction.
<b>UX</b> Max. infeed	Stock allowance in the X direction.

The following sequence accepts the parameters for the MDA facing cycle, and starts the machining sequence in MDA:



## 1 Brief description

**Module objective:**

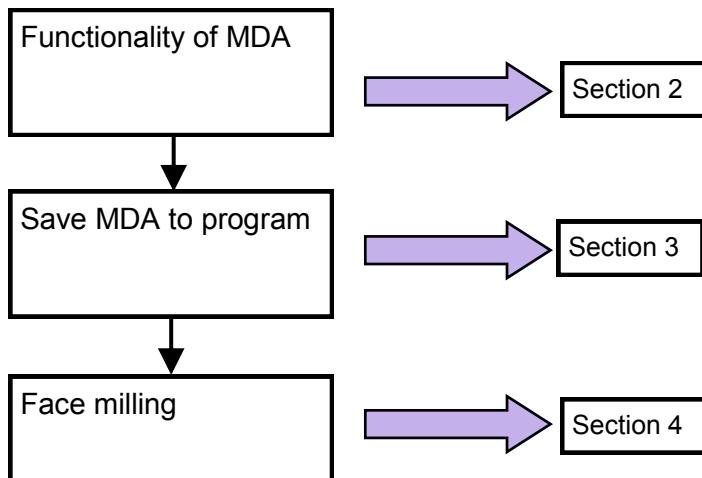
Upon completion of this module you can execute commands in the MDA mode.

**Module description:**

You can enter program commands in MDA and immediately execute them. This allows the operator to activate functions on the machine normally for setting up purposes outside the NC program.

**Module content:**

Functionality of MDA  
Save MDA to program  
Face milling



## Section 2

### Functionality of MDA

The following functions are possible in MDA mode, executing immediately program blocks in the Machine area. It is possible to execute M functions, especially for tool changing or activating clamping devices outside of the NC program.

You can activate MDA in the following sequence:



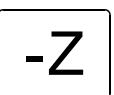
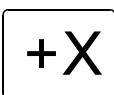
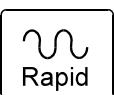
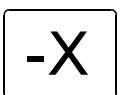
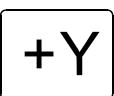
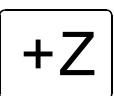
To scratch on the workpiece (Set zero offset, measure tools manually)  
Jog and MDA can be used with each other.

In MDA you can then call the tool with spindle speed and direction. Be careful not to program M02 as this will have the effect of resetting the given spindle speed and direction.

After NC start you can return to Jog and position the axis accordingly.

Example shows text typed into the MDA area and sequence for moving the axis:

T1  
M6  
X0 Y0 S2000 M3



Notes

## Section 3

### Save MDA to program

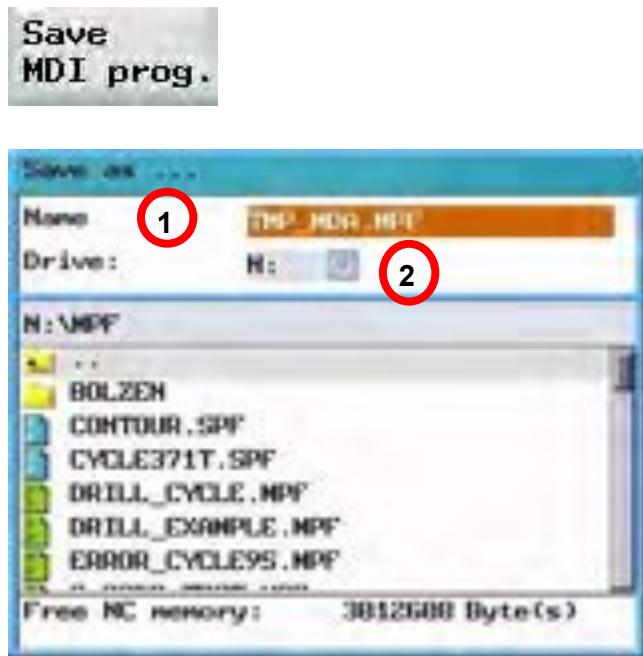
Notes

The content of the MDA buffer can be saved as an NC program, this can later be used again.

Pre-conditions:

- MDA-mode is active
- Program blocks exist in the buffer.

To save MDA to a program select the following sequence:



- 1 In the window "Save as" you can input the name of the program in which the blocks will be saved.
- 2 The destination can be determined with the Select key (Drive N:), the CF-card (Drive D:) or the RCS-connection can be selected.

The data will be saved with the following sequence:

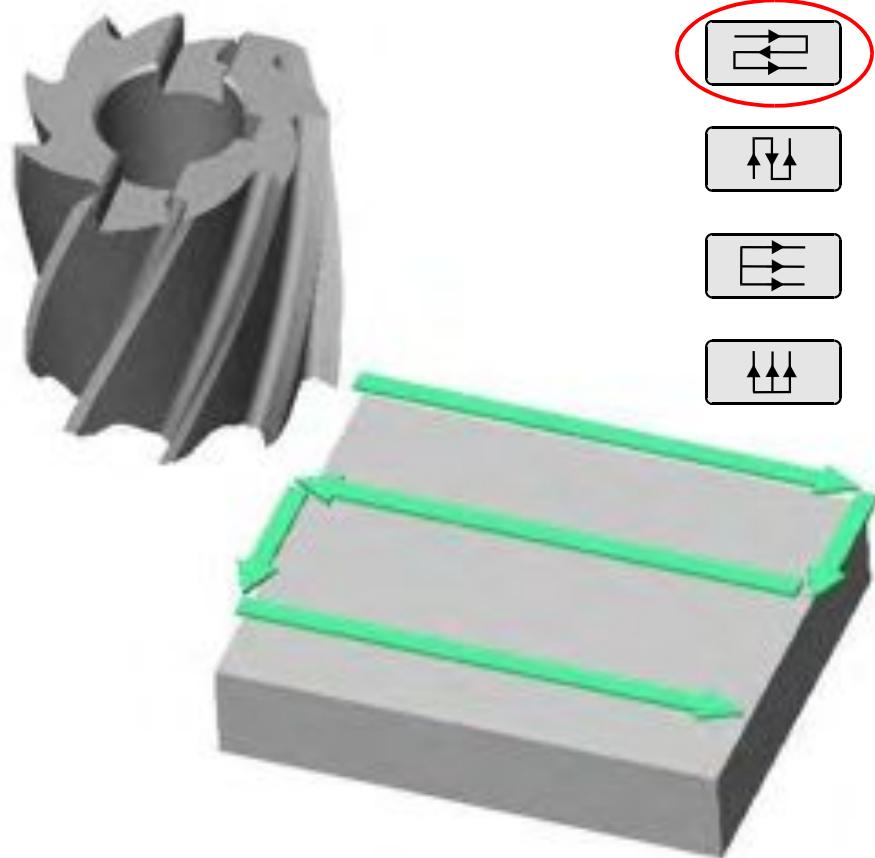


## Section 4

### Face milling

The “Face milling” function is used for surface preparation prior to machining in Automatic.

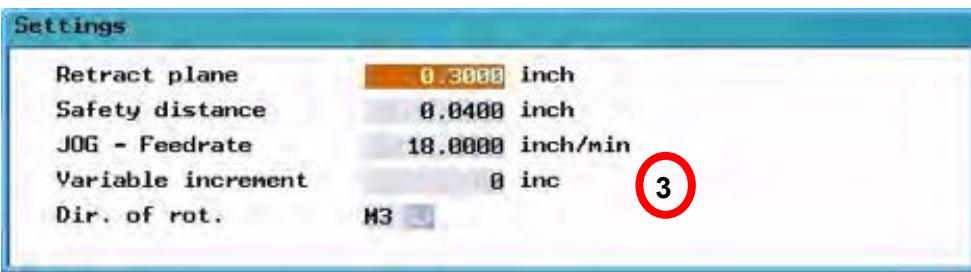
Notes



Face  
cutting

The return plane and safety distance should be entered in the Settings dialog.

Settings



3

“Variable increment“ This parameter should be kept to zero when not being used.

## Section 4

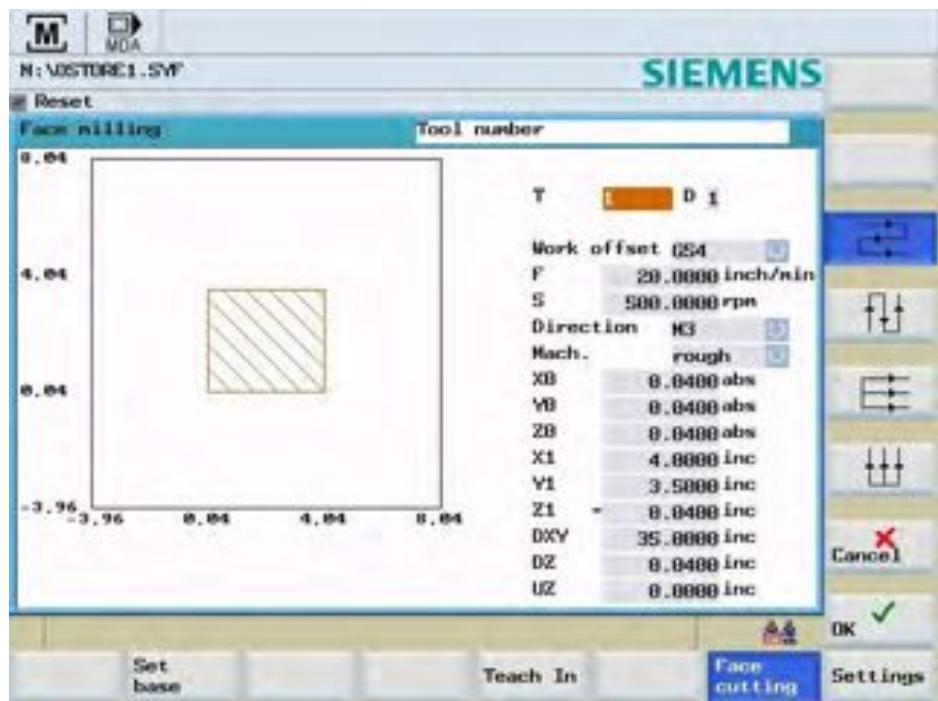
### Face milling

Notes

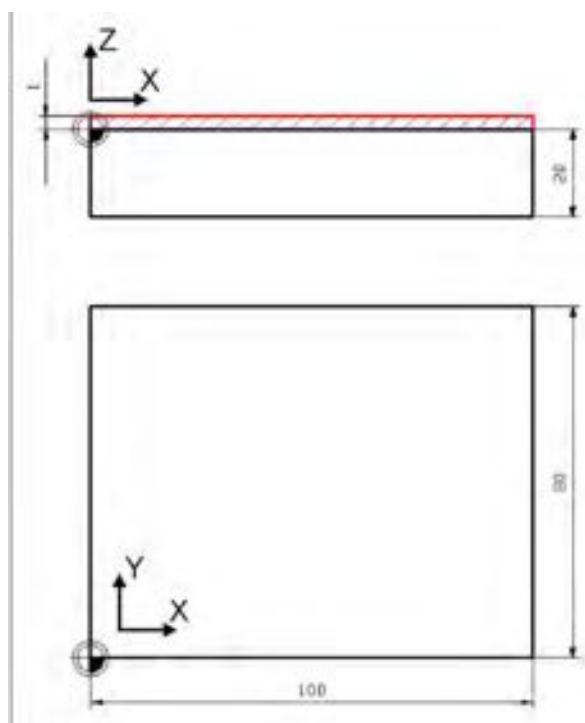
You can return to the previous page with the following sequence:



Face  
cutting



The values in the above dialog are for the following example:



## Section 4

### Face milling

Notes

Description of the parameters in the FACE TURNING working window.

Parameter	Explanation
<b>Tool</b>	Selected tool to be used in the working cycle.
Work offset	Work offset to be selected in the program
Feed <b>F</b>	Selected path feedrate, in mm/min or mm/rev.
Spindle <b>S</b> rpm	selected spindle speed
Direction	Select the direction of rotation of the spindle.
<b>Mach</b>	Definition of the surface quality. You can select between roughing or finishing.
<b>X0, Y0, Z0</b>	select the geometry of the workpiece.
<b>X1, Y1</b>	
Blank dimensions	
<b>Z1 Finish dimension</b>	Finishing dimension in the Z
<b>DXY Max. infeed</b>	Input the amount of the infeed motion (X, Y).
<b>DZ Max. infeed</b>	Input the amount of the infeed motion (Z)
<b>UZ</b>	Input the stock allowance when roughing.

The following sequence accepts the parameters for the MDA facing cycle, and starts the machining sequence in MDA:



## 1 Brief description

### Module objective:

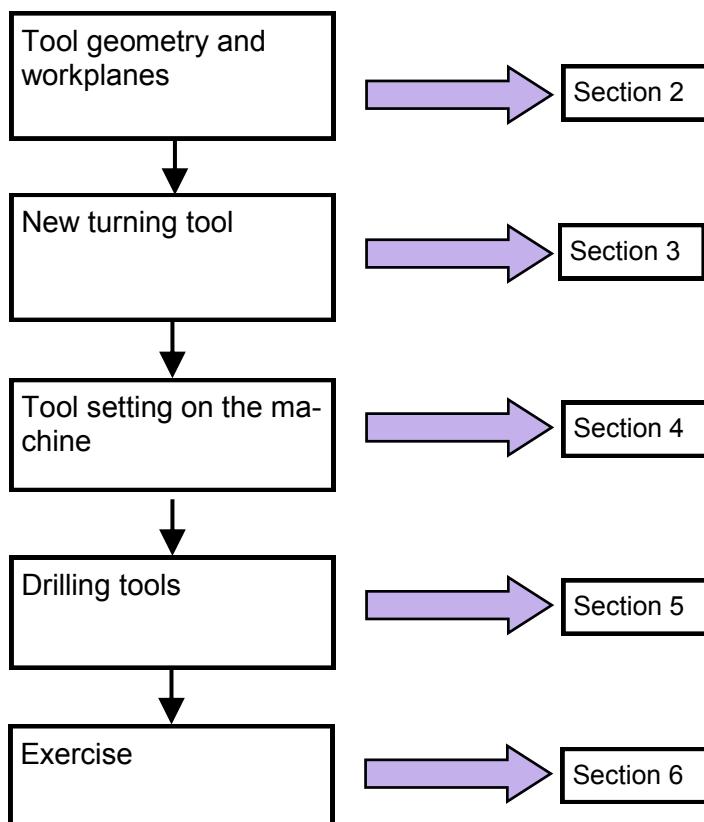
Upon completion of this module you can define new tools and also parameterise them.

### Module description:

A description of the tools which are supported by the controller and also how they are parameterised can be found in this module

### Module content:

Tool geometry and workplanes  
New turning tool  
Tool setting on the machine  
Drilling tools



## Section 2

### Tool geometry and workplanes

Notes

When working on a turning machine with turning tools, the workplane is determined from the Z– axis (=1st. Axis in plane) and the X-axis (=2nd. Axis in plane).

A simple turning machine has no Y axis. When working with driven tools, the necessary Y axis functionality is achieved with an interpolation of the C axis (Virtual Y axis functionality is achieved).

Therefore the following is the workplane which is active on a turning machine.

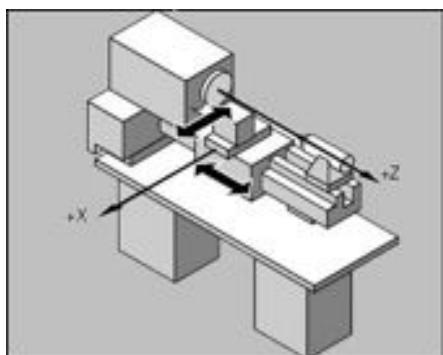


To activate this in the part program G18 has to be programmed.

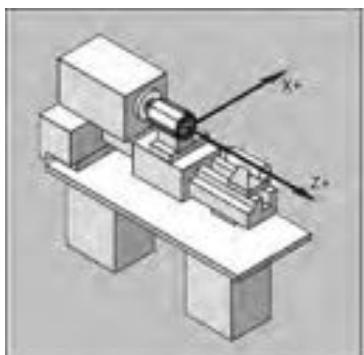
The Y axis (Blue arrow) is not available on simple turning machines.

It is important for the programming of a turning machine to observe first whether the tool is in front of the turning centre or behind.

Turning in front of the turning centre line



Turning behind the turning centre line



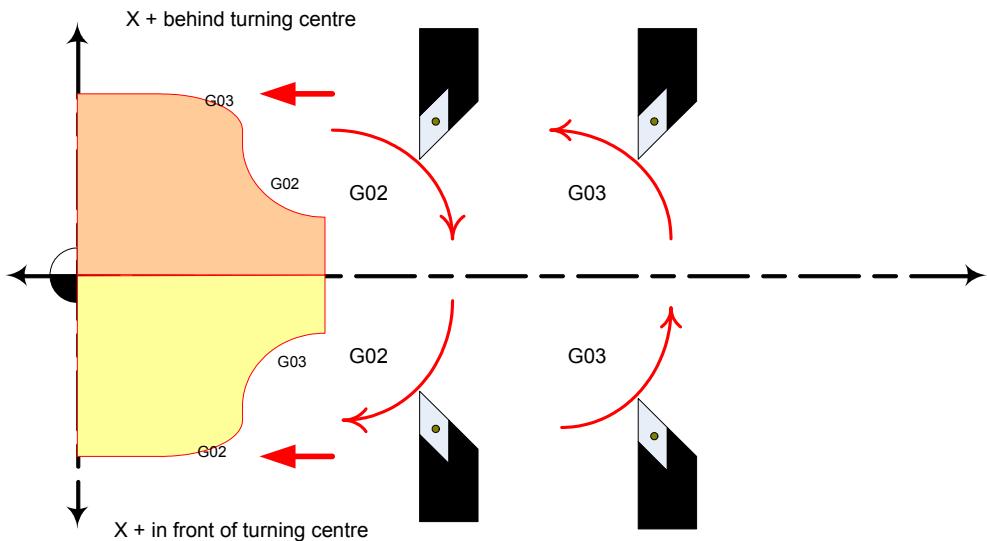
## Section 2

### Tool geometry and workplanes

Notes

When working with turning machines which are having their tools in front of the turning centre the G18 coordinate system puts the Y axis in the direction of the floor as opposed to the up direction when the turning tools are behind the turning centre. This affects the direction of circles and also the radius compensation of tools in the plane and care for this should be taken.

The following graphic should help to explain this phenomenon:



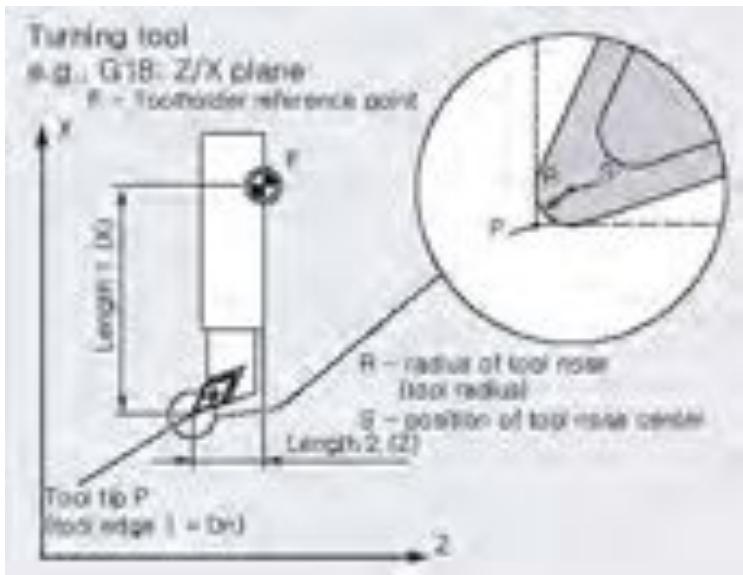
## Section 2

### Tool geometry and workplanes

Notes

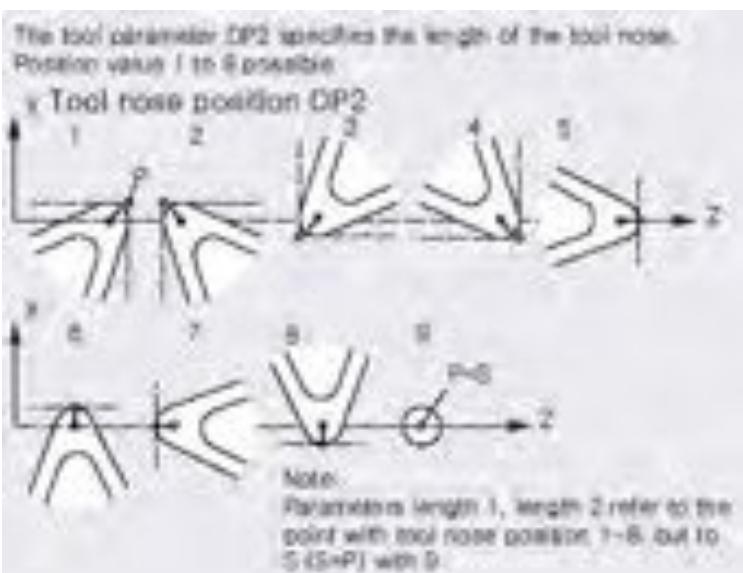
The tool length offset is applied to the axis dependent upon the plane which is active, with G18:

- Length 1: - Calculated in the X direction
- Length 2: - Calculated in the Z direction
- Radius: - Z/X-Plane



In parallel to the length and radius offsets, the orientation of the turning tool has to be given into the control. This Orientation is necessary for the cycles and radius compensation functions to correctly calculate the contour intersections without collisions.

The following values are possible:



## Section 3

### New turning tool

Notes

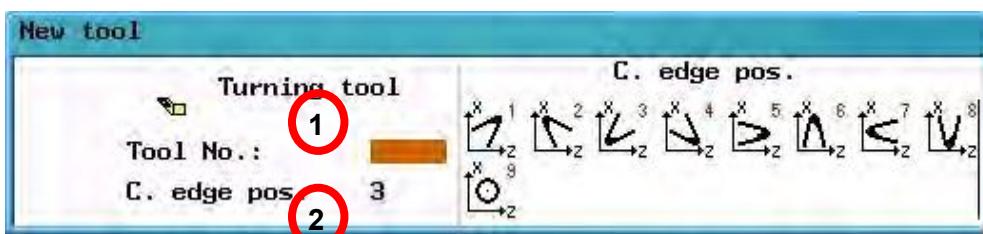
The geometry of the tool has to be entered into the tool offset screen before it can be used on the machine in a part program.

The following sequence shows how to perform this task.

OFFSET  
PARAM

New  
tool

Turning  
tool



- 1 Enter the tool number at this position.
- 2 Enter the tool type at this position.

OK ✓

After this procedure the tool appears in the tool list.  
The length and radius of the tool can now be entered in the left geometry fields.

Example:

T	D <sub>Σ</sub>	Geometry		
		Length1	Length2	Radius
1	3	3.8431	1.7211	0.0000

**Important:**

After each value is entered this key must be pressed.



## Section 4

### Tool setting on the machine

Notes

It is possible to determine the length of the tool in X and Z simply, directly on the machine.

A prerequisite for this, is a workpiece with known position and size (diameter) located in the chuck.

To scratch on the surface of the workpiece, the spindle should first be started in MDA.

The following sequence can be followed.

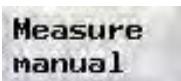
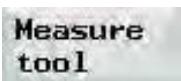


T1  
G95 S500 M4

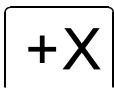
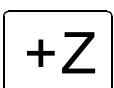


Do not program in MDA the M02 function as this will result in the spindle stopping immediately.

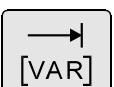
To move the axis manually you first have to switch to JOG mode.



With the axis keys you can move the tool to the workpiece.



For an exact measurement when close to the component, the INC mode should be used:

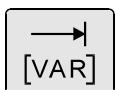


## Section 4

### Tool setting on the machine

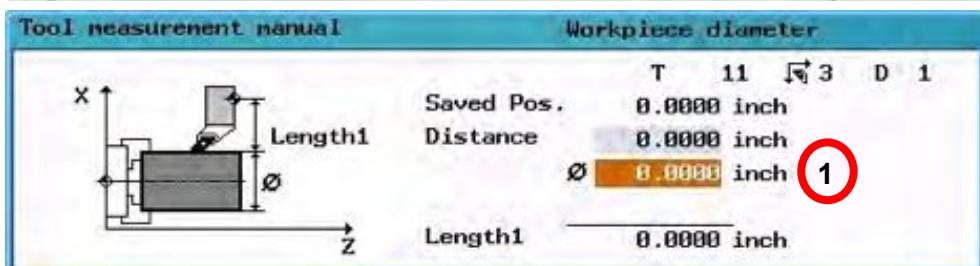
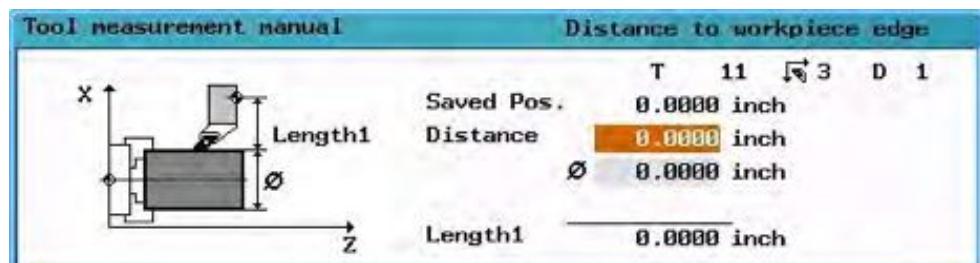
Notes

Using the VAR key repeatedly, can change the size of the INC to be used, as seen in the following sequence:



Move the tool tip to a known diameter.

To set Length 1 follow the sequence:



1

Enter a diameter at this position.

Save  
position

Set  
length1

To show that the data has been set, this text will be shown.

Compensation data have been activated!

## Section 4

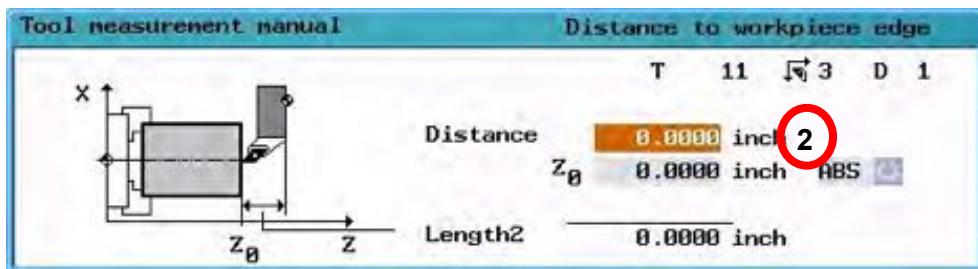
### Tool setting on the machine

Notes

Move the tool tip to a known datum on the face.

To set Length 2 follow the sequence:

Length2



2

Enter a length at this position if using a setting block.

Set  
length2

To show that the data has been set, this text will be shown.

Compensation data have been activated!

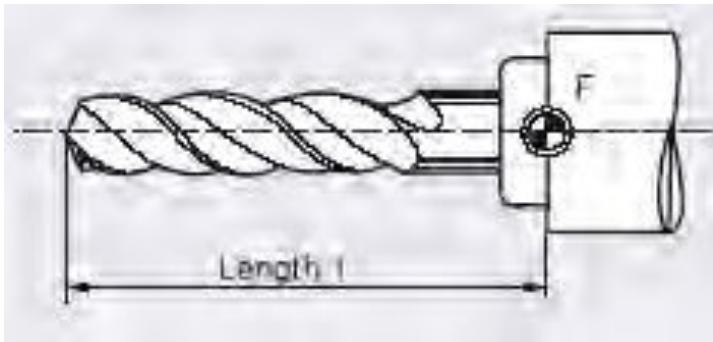
## Section 5

### Drilling tools

Notes

A difference between turning and drilling tools exists in the control, drills have only a length in the control.

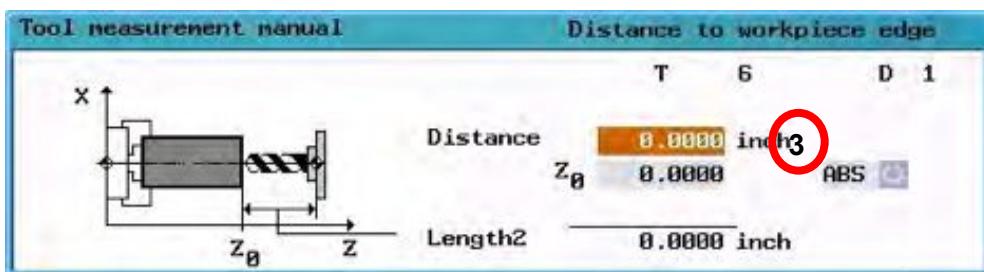
The length can be determined with the same steps as in the last section.



For all simple drilling operations, G17 must be selected in the NC program.

Move the tool tip onto a known datum on the face.

To set Length 1 follow the sequence:



3

Enter a length at this position if using a setting block.

**Set  
length2**

To show that the data has been set, this text will be shown.

**Compensation data have been activated!**

## Section 6

### Exercise

Notes

Create six tools.

- 1/ Rough turning tool. Tool number 1. External. Tip radius 0.8mm
- 2/ Finish turning tool. Tool number 2. External. Tip radius 0.4mm
- 3/ Grooving tool. Tool number 3. External. 3.1mm wide
- 4/ Threading tool. Tool number 4. External.
- 5/ Drilling tool. Tool number 5. 6.5mm diameter
- 6/ Tapping tool. Tool number 6. 8mm diameter

Below shows what the tool offset page will look like.

Type	T	Ø	Geometry			Wear			D	I	O >>
			Length1	Length2	Radius	Length1	Length2	Radius			
1	1	1	0.0000	0.0000	0.8315	0.0000	0.0000	0.0000	3		Measure tool
2	1	2	0.0000	0.0000	0.8157	0.0000	0.0000	0.0000	3		Delete tool
3	2	3	0.0000	0.0000	0.8039	0.0000	0.0000	0.0000	3		Extend
4	1	4	0.0000	0.0000	0.8000	0.0000	0.0000	0.0000	3		Edges
5	1	5	0.0000	0.0000	0.1290	0.0000	0.0000	0.0000	3		Find
6	1	6	0.0000	0.0000	0.1575	0.0000	0.0000	0.0000	3		New tool
7	1	7	0.0000	0.0000	0.1575	0.0000	0.0000	0.0000	3		User data

Type	T	Ø	Geometry			Wear			D	I	O >>
			Length1	Length2	Radius	Length1	Length2	Radius			
1	1	1	0.0000	0.0000	0.8315	0.0000	0.0000	0.0000	2		Measure tool
2	1	2	0.0000	0.0000	0.8157	0.0000	0.0000	0.0000	2		Delete tool
3	2	3	0.0000	0.1181	0.8039	0.0000	0.0000	0.0000	2		Extend
4	1	4	0.0000	0.0000	0.8000	0.0000	0.0000	0.0000	2		Edges
5	1	5	0.0000	0.0000	0.1290	0.0000	0.0000	0.0000	2		Find
6	1	6	0.0000	0.0000	0.1575	0.0000	0.0000	0.0000	2		New tool
7	1	7	0.0000	0.0000	0.1575	0.0000	0.0000	0.0000	2		User data

## 1 Brief description

**Module objective:**

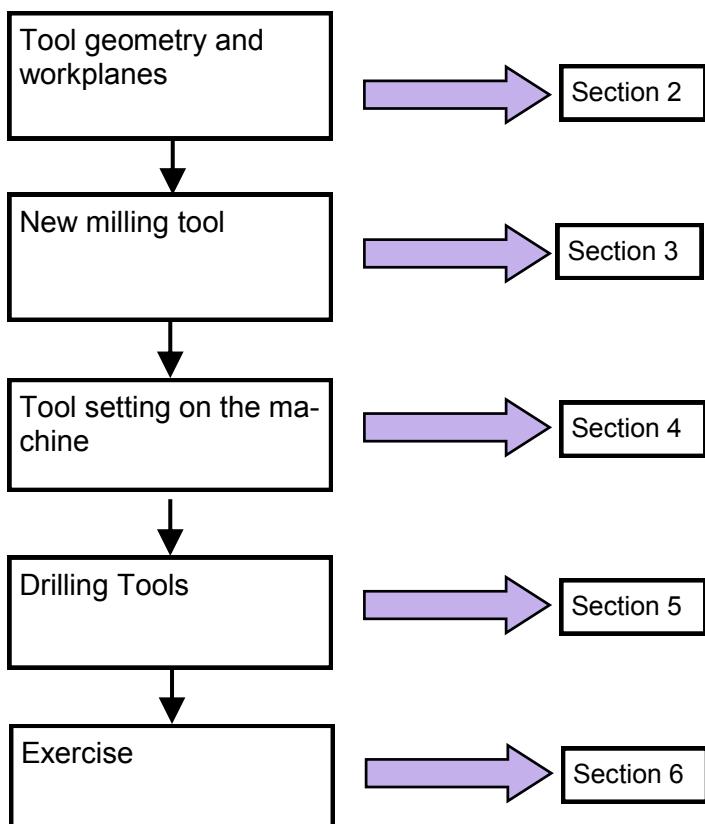
Upon completion of this module you can define new tools and also parameterise them.

**Module description:**

A description of the tools which are supported by the controller and also how they are parameterised can be found in this module.

**Module content:**

Tool geometry and workplanes  
New milling tool  
Tool setting on the machine  
Drilling tools



## Section 2

### Tool geometry and workplanes

#### 2.1 Tool geometry and workplanes

Notes

Work plane for milling.

When working on a milling machine with milling tools, the work plane is determined from the X-axis (=1st. axis in plane) and the Y-axis (=2nd. axis in plane).

The Z axis is the infeed axis and is perpendicular to the plane X/Y plane.

Therefore the following is the workplane which is active on a milling machine.



To activate this, in the part program G17 has to be programmed.

The G18 or G19 plane can be selected when a spindle attachment (rotary head etc.) is used which places the orientation of the spindle into the X or Y axis.

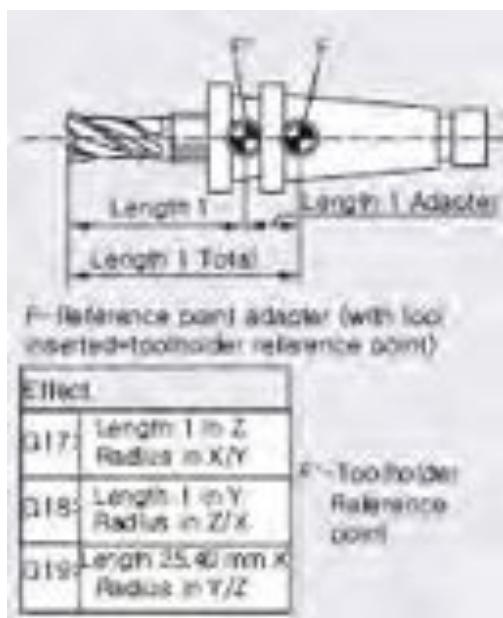
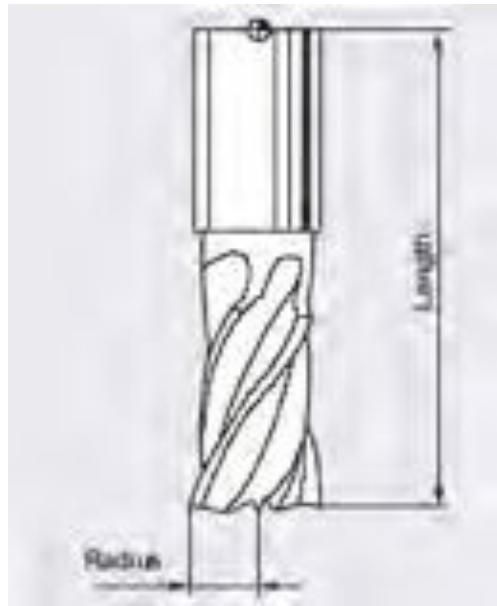
## Section 2

### Tool geometry and workplanes

The tool length offset is applied to the axis dependent upon the plane which is active, with G17:

- Length 1: - Calculated in the Z direction  
Radius: - XY-plane

Notes



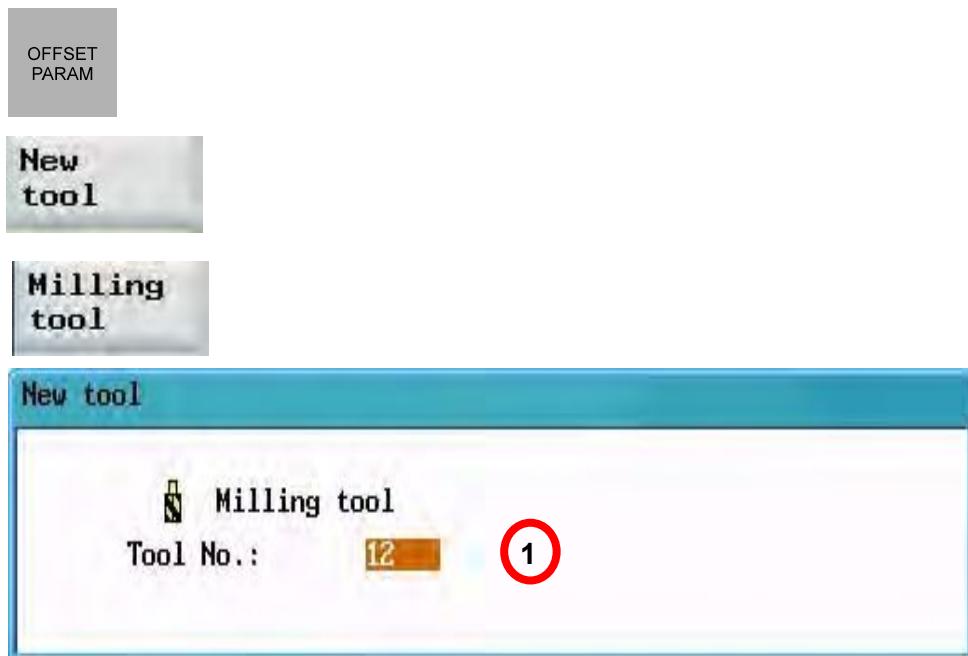
## Section 3

### New milling tool

Notes

The geometry of the tool has to be entered into the tool offset screen before it can be used on the machine in a part program.

The following sequence shows how to perform this task.



- 1 Enter the tool number at this position.



After this procedure the tool appears in the tool list.  
The length and radius of the tool can now be entered in the left geometry fields, manually if need arises.

Example:

Type	T	D <sub>Σ</sub>	Geometry
			Length1      Radius
■	1	1	1.5670      0.1969

**Important:**

After each value is entered this key must be pressed.



## Section 4

### Tool setting on the machine

Notes

It is possible to determine the length of the tool in Z simply and directly on the machine.

A prerequisite for this, is a workpiece with known position and size (diameter) located in the chuck.

To scratch the surface of the workpiece, the spindle should first be started in MDA.

The following sequence can be followed.

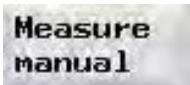
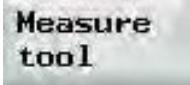
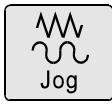


T1  
M6  
G95 S2000 M3

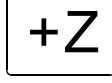
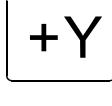
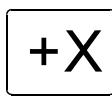


Do not program in MDA the M2 instruction as this will result in the spindle stopping immediately.

To move the axis manually you first have to switch to JOG mode.



With the axis keys you can bring the tool to the workpiece.



For an exact measurement close to the component, the INC mode should be used:



## Section 4

### Tool setting on the machine

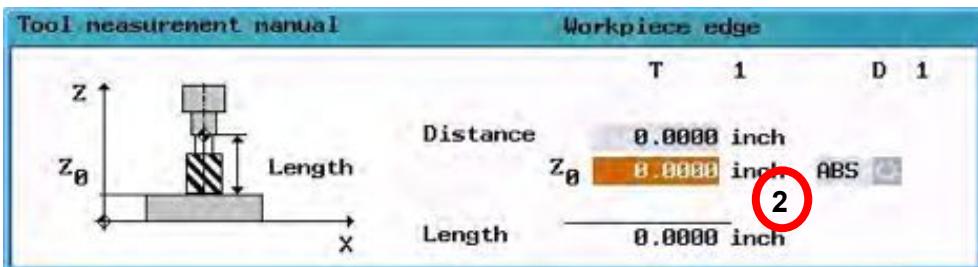
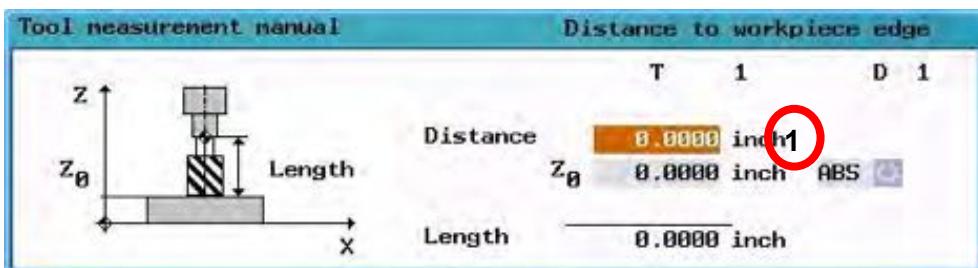
Notes

Using the VAR key repeatedly, can change the size of the INC to be used, as seen in the following sequence:

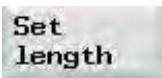


Move the tool tip to a known diameter.

To set Length 1 follow the sequence:



- 1 Enter a length of the setting block at this position if using a setting block.
- 2 Enter the height from the reference plane at this position.



To show that the data has been set, this text will be shown.

**Compensation data have been activated!**

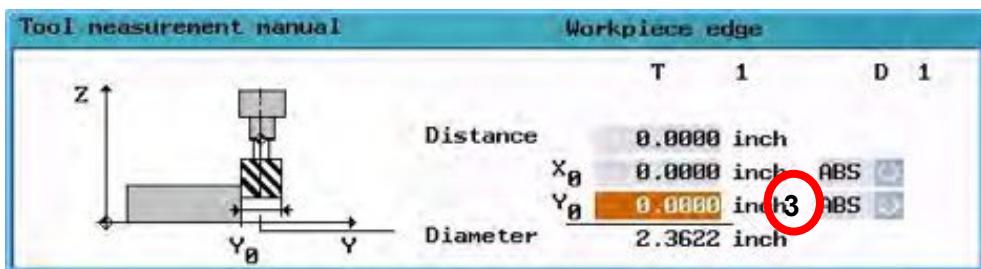
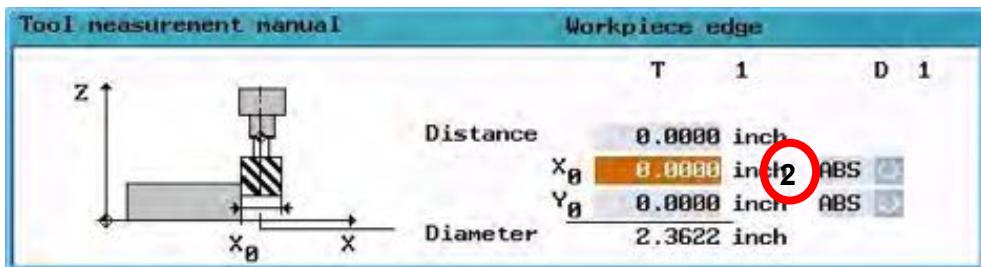
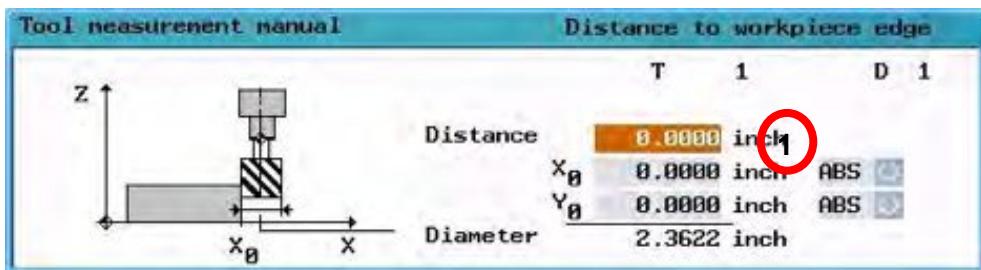
## Section 4

### Tool setting on the machine

Notes

You can also measure the diameter of the cutter using the following sequence.

Diameter



- 1 Enter a length of the setting block at this position if using a setting block.
- 2 Enter the length from the reference plane in the X axis at this position.
- 3 Enter the length from the reference plane in the Y axis at this position.

Set  
diamet.

To show that the data has been set, this text will be shown.

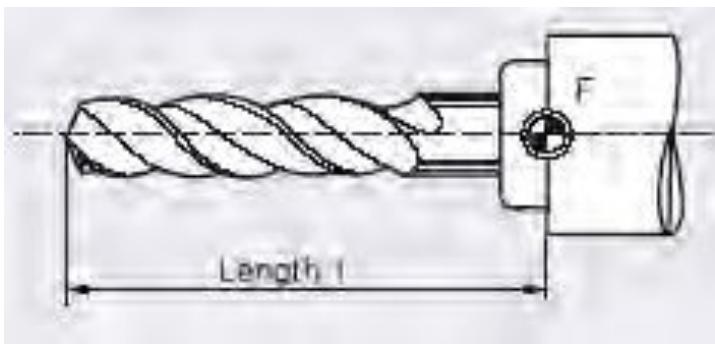
Compensation data have been activated!

## Section 5

### Drilling tools

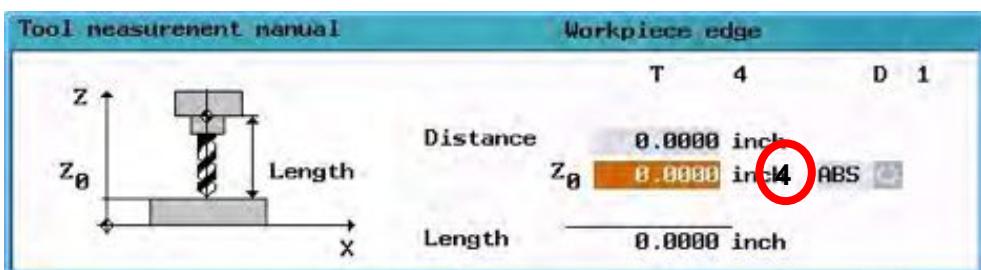
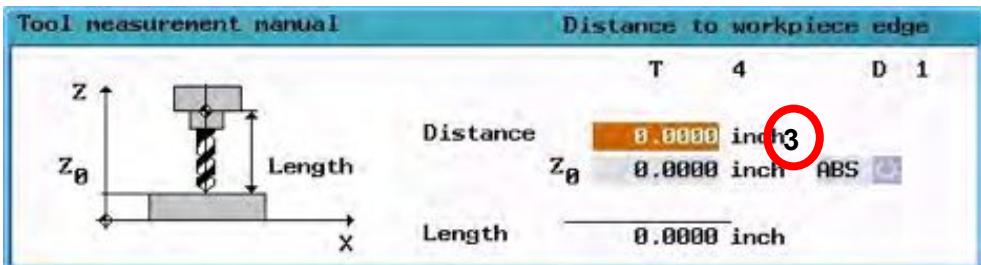
Notes

The only difference between a milling cutter and a drill is that a drill has no diameter to set, drills have only a length in the control.  
The length can be determined with the same steps as in the last section.



Move the tool tip onto a known datum on the face.

To set Length 1 follow this sequence.



- 3 Enter a length of the setting block at this position if using a setting block.
- 4 Enter the height from the reference plane at this position.

Set  
length

To show that the data has been set, this text will be shown.

Compensation data have been activated!

## Section 6

### Exercise

Notes

Create six tools.

- 1/ 60mm Diameter cutter
- 2/ 6mm Diameter cutter
- 3/ 10mm Diameter cutter
- 4/ 12mm centre drill
- 5/ 8.2mm drill
- 6/ M10 Tap

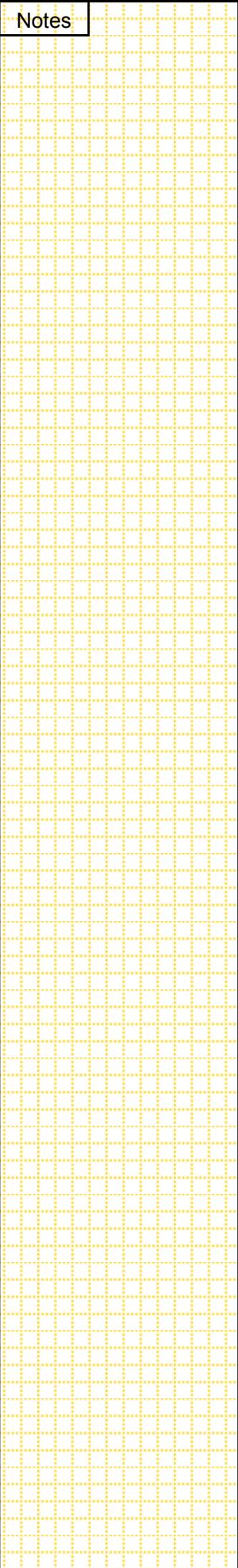
Below shows what the tool offset page will look like.

The screenshot shows a CAD software interface with a tool offset table. The table has columns for Type, T, D, Geometry, and Wear. The first row is highlighted in orange. A context menu is open on the right side of the table, listing options: Measure tool, Delete tool, Extend, Edges, Find, New tool, and User data. The 'Tool list' tab is selected at the bottom left. The top bar includes icons for zoom and orientation, and the number '102'. The right side of the screen features a grid for notes.

Type	T	D	Geometry	Wear	I	D	I	D	I
1	1	1	0.0000	1.1811	0.0000	0.0000	0.0000	0.0000	0.0000
2	1	1	0.0000	0.1181	0.0000	0.0000	0.0000	0.0000	0.0000
3	1	1	0.0000	0.1969	0.0000	0.0000	0.0000	0.0000	0.0000
4	1	1	0.0000	0.2362	0.0000	0.0000	0.0000	0.0000	0.0000
5	1	1	0.0000	0.1614	0.0000	0.0000	0.0000	0.0000	0.0000
6	1	1	0.0000	0.1969	0.0000	0.0000	0.0000	0.0000	0.0000

---

Notes



## 1 Brief description

**Module objective:**

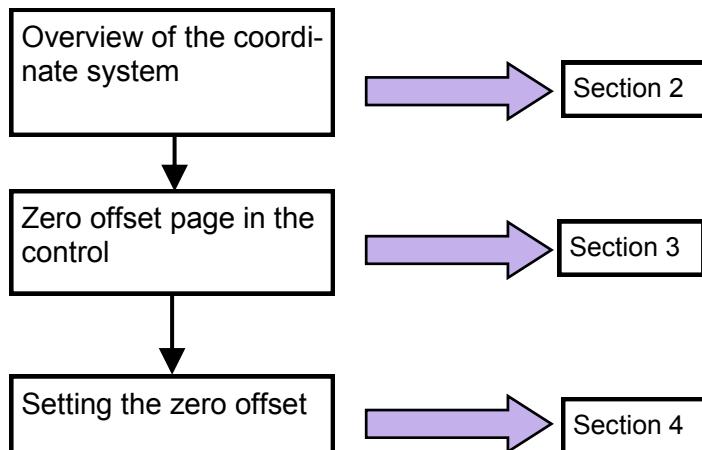
Upon completion of this module you will be able to set the zero offset on a turning machine

**Module description:**

This module shows how to set the zero offset on a turning machine, and helps you to understand it's purpose.

**Module content:**

Overview of the coordinate system  
Zero offset page in the control  
Setting the zero offset



## Section 2

### Overview of the coordinate system

Notes

#### Description of the zero offset:

The zero offset determines the difference between the Machine Coordinate System (MCS) and the Workpiece Coordinate System (WCS).

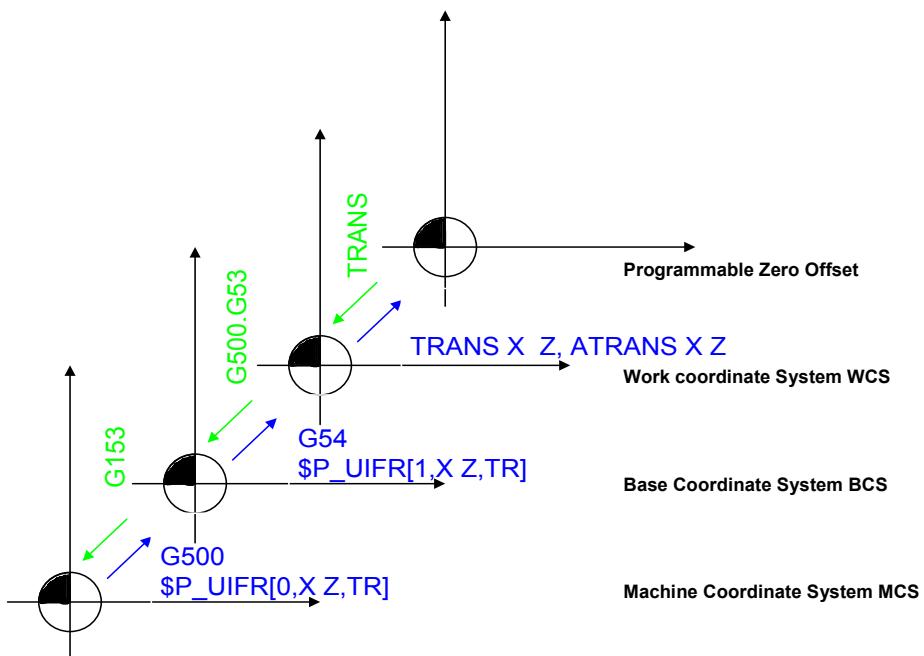
For programming, the dimensions are with reference to the Workpiece Coordinate System.

The exact position of the workpiece on the machine is not known at the time of writing the workpiece program.

The distance between the machine zero point and the workpiece zero point must therefore be obtained.

This geometric difference is known as the zero offset.

The following diagram gives an overview of the offset possibilities of the control:



Description of the single components:

#### Machine Coordinate System (MCS):

The machine coordinate system is determined by the manufacturer of the machine, the value (0) is determined at the 1st setup by the builder and should only be changed by experienced service personnel.

The G code G153 can be used blockwise to perform a move in the MCS at any time in the program.

**NOTE: The instruction G153 is only blockwise active!**

Notes

#### **Base Coordinate System (BCS):**

The Base Coordinate System works between the Machine Coordinate System and the Workpiece Coordinate System. If the value of the Base Coordinate System is changed, then the Workpiece Coordinate System will move respectively.

The Base Coordinate System is activated with G500. When G500 is programmed the modally active (G54) will be deactivated.

With G54 active a movement can be programmed with respect to the BCS by using the blockwise G code G53.

#### **Workpiece Coordinate System (WCS):**

The workpiece coordinate system is the offset between the BCS and the origin of the component.

When the BCS system has no value, then the WCS is the offset to the MCS system.

7 Zero offsets are therefore available to the operator/programmer.

G500 - Basic Zero Offset

G54 - 1st. Zero Offset

G55 - 2nd. Zero Offset

G56 - 3rd. Zero Offset

G57 - 4th. Zero Offset

G58 - 5th. Zero Offset

G59 - 6th. Zero Offset

## Section 3

### Zero offset page in the control

An offset page is available in the control, where the operator of the machine can input and check the zero offset values.

The following sequence shows how to perform this task.

OFFSET  
PARAM

Work  
offset

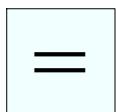
	X	inch	Z	inch	C	*	X	?	Z	?	C	?
Base	0.000		0.000		0.000		0.000		0.000		0.000	
054	0.000		0.000		0.000		0.000		0.000		0.000	
055	0.000		0.000		0.000		0.000		0.000		0.000	
056	0.000		0.000		0.000		0.000		0.000		0.000	
057	0.000		0.000		0.000		0.000		0.000		0.000	
058	0.000		0.000		0.000		0.000		0.000		0.000	
059	0.000		0.000		0.000		0.000		0.000		0.000	
Program	0.000		0.000		0.000		0.000		0.000		0.000	
Scale	1.000		1.000		1.000							
Mirror	0		0		0							
Total	0.000		0.000		0.000		0.000		0.000		0.000	

When working with a turning machine an offset in X is normally not necessary.

A BCS offset can be used to offset the front of the chuck from the MCS. A WCS offset can then be used for the origin of the workpiece.

All values can be edited on the offset page.

To simplify the input a pocket calculator is integrated into the control:

 By pressing the equals key in the respective input field, the calculator will start.

Notes

## Section 3

### Zero offset page in the control

Notes

After pressing the equals key the value from the input field will automatically be inserted into the calculator for further processing.



The calculation takes place with the following sequence.



And the calculation is transferred to the zero offset page with the following sequence.



	X	inch Z	inch C	°
Base	0.0000	0.0000	0.0000	
G54	0.0000	251.3670	1 0.0000	
G55	0.0000	0.0000	0.0000	
G56	0.0000	0.0000	0.0000	
G57	0.0000	0.0000	0.0000	
G58	0.0000	0.0000	0.0000	
G59	0.0000	0.0000	0.0000	

- 1 The calculated value will be transferred into the input field.

With the parameter field

X2 Z2 C2

A coordinate rotation can also be calculated.

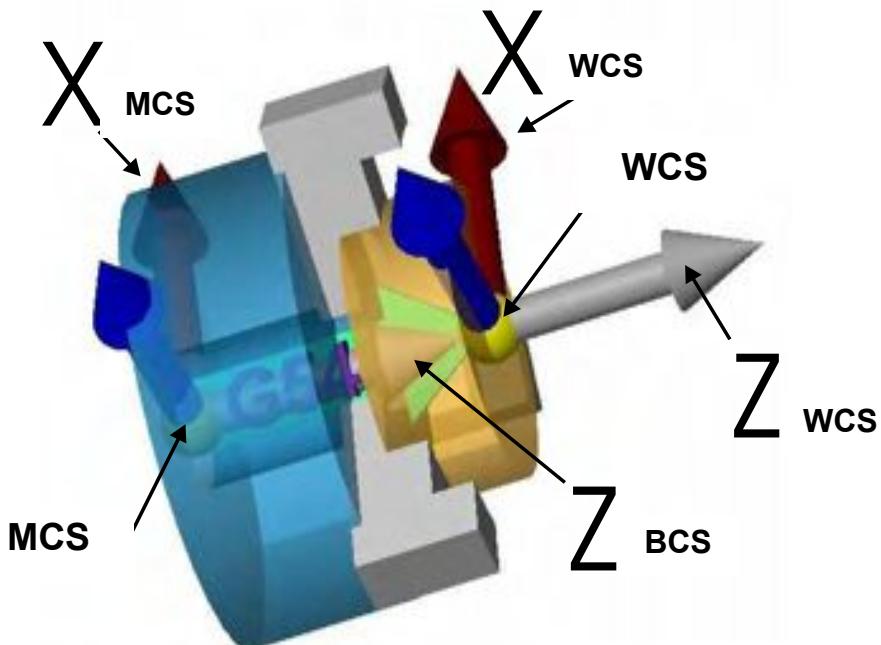
## Section 4

### Setting the zero offset

Notes

The zero offset on a turning machine is normally only used in the Z axis, the X axis stays at 0 (spindle centre line).

Therefore: from left to right in the diagram below the BCS offsets the MCS and the WCS offsets the BCS.

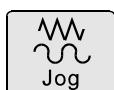


The scratch method is used to set the zero offset, the spindle should therefore be started in MDA.

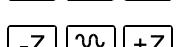
The following sequence shows how to perform this task.



**T1 D1  
S200 M4**



Touch the workpiece using the axis direction buttons.



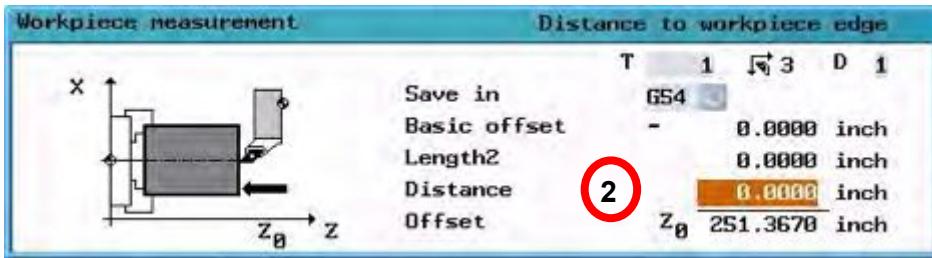
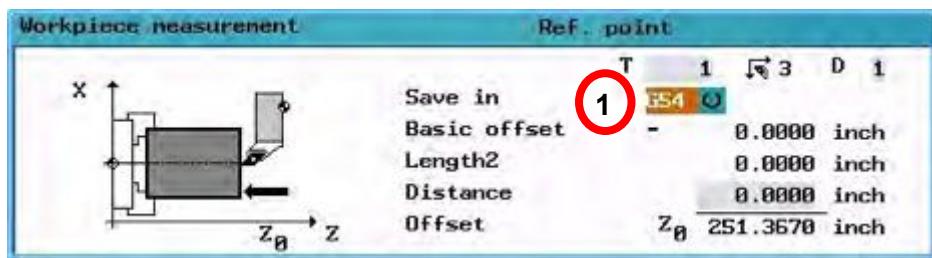
## Section 4

### Setting the zero offset

Notes

The following sequence shows how to set the zero offset automatically on the machine.

#### Measure workpiece



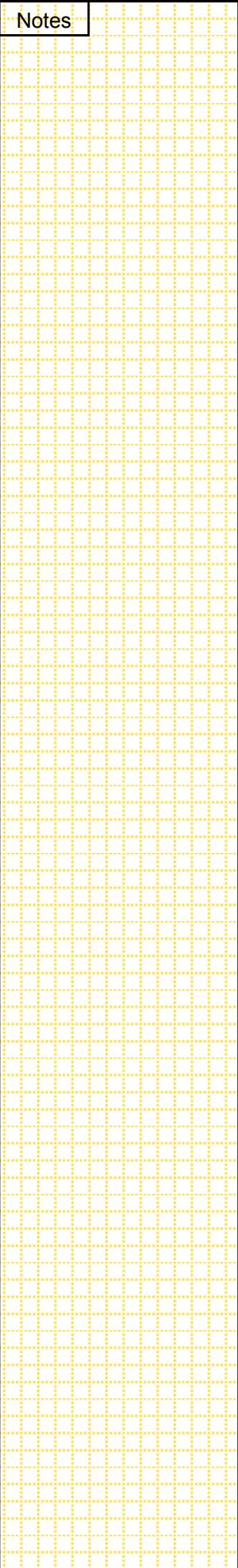
- 1 You can select the required offset with the Toggle field using the SELECT softkey, into which zero offset the value should be stored.
- 2 Enter a length at this position if using a setting block.

The value will be calculated and stored with the following sequence.

#### Set work offset

---

Notes



## 1 Brief description

**Module objective:**

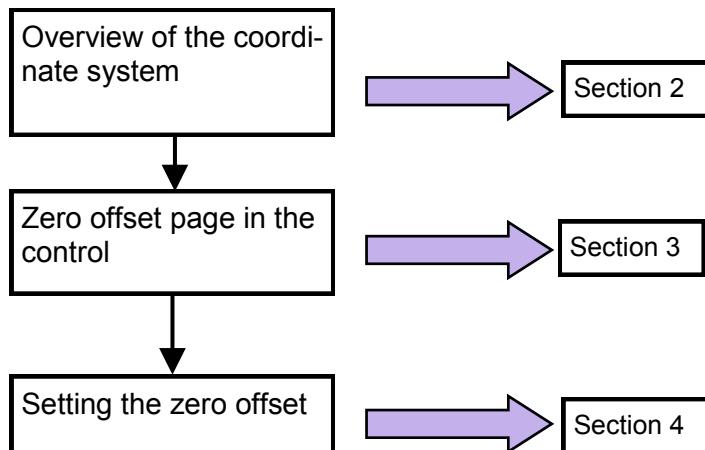
Upon completion of this module you will be able to set the zero offset on a milling machine.

**Module description:**

This module shows how to set the zero offset on a milling machine, and helps you to understand its purpose.

**Module content:**

Overview of the coordinate system  
Zero offset page in the control  
Setting the zero offset



## Section 2

### Overview of the coordinate system

Notes

#### Description of the zero offset:

The zero offset determines the difference between the Machine Coordinate System (MCS) and the Workpiece Coordinate System (WCS).

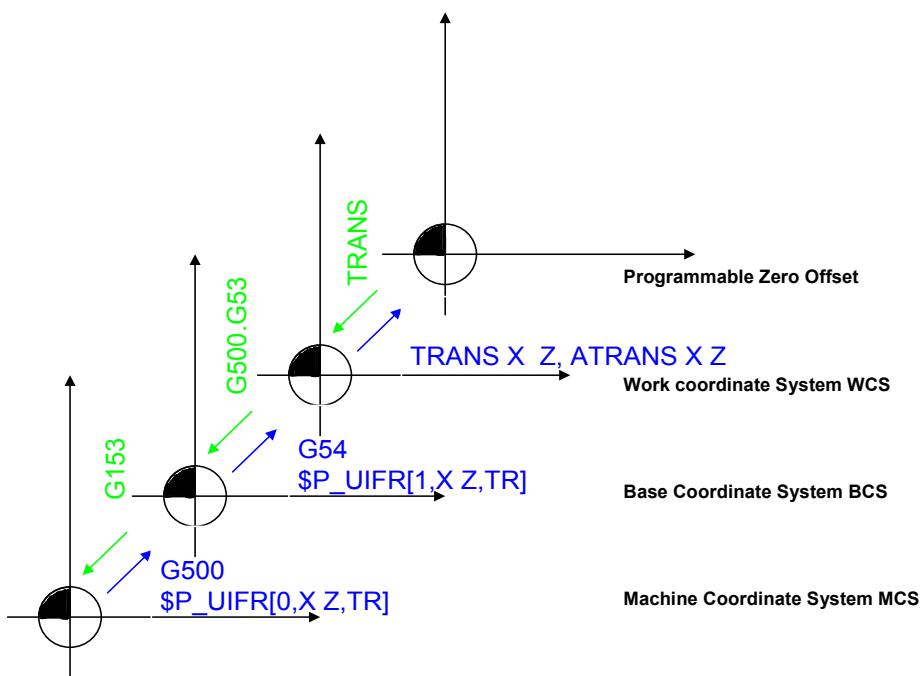
For programming, the dimensions are with reference to the Workpiece Coordinate System.

The exact position of the workpiece on the machine is not known at the time of writing the workpiece program.

The distance between the machine zero point and the workpiece zero point must therefore be obtained.

This geometric difference is known as the zero offset.

The following diagram gives an overview of the offset possibilities of the control:



Description of the single components:

#### Machine Coordinate System (MCS):

The machine coordinate system is determined by the manufacturer of the machine, the zero value is determined at the 1st setup by the machine tool builder and should only be changed by experienced service personnel. The G code G153 can be used blockwise to perform a move in the MCS at any time in the program.

**NOTE: The instruction G153 is only blockwise active!**

## Section 2

### Overview of the coordinate system

Notes

#### Basic Coordinate System (BCS):

The Basic Coordinate System works between the Machine Coordinate System and the Workpiece Coordinate System. If the value of the Basic Coordinate System is changed, then the Workpiece Coordinate System will move respectively.

The Basic Coordinate System is activated with G500. When G500 is programmed the modally active (G54) will be deactivated.

With G54 active a movement can be programmed with respect to the BCS by using the blockwise G code G53.

#### Workpiece Coordinate System (WCS):

The workpiece coordinate system is the offset between the BCS and the origin of the component.

When the BCS system has no value, then the WCS is the offset to the MCS system.

7 Zero offsets are therefore available to the operator/programmer.

G500 - Basic Zero Offset

G54 - 1st. Zero Offset

G55 - 2nd. Zero Offset

G56 - 3rd. Zero Offset

G57 - 4th. Zero Offset

G58 - 5th. Zero Offset

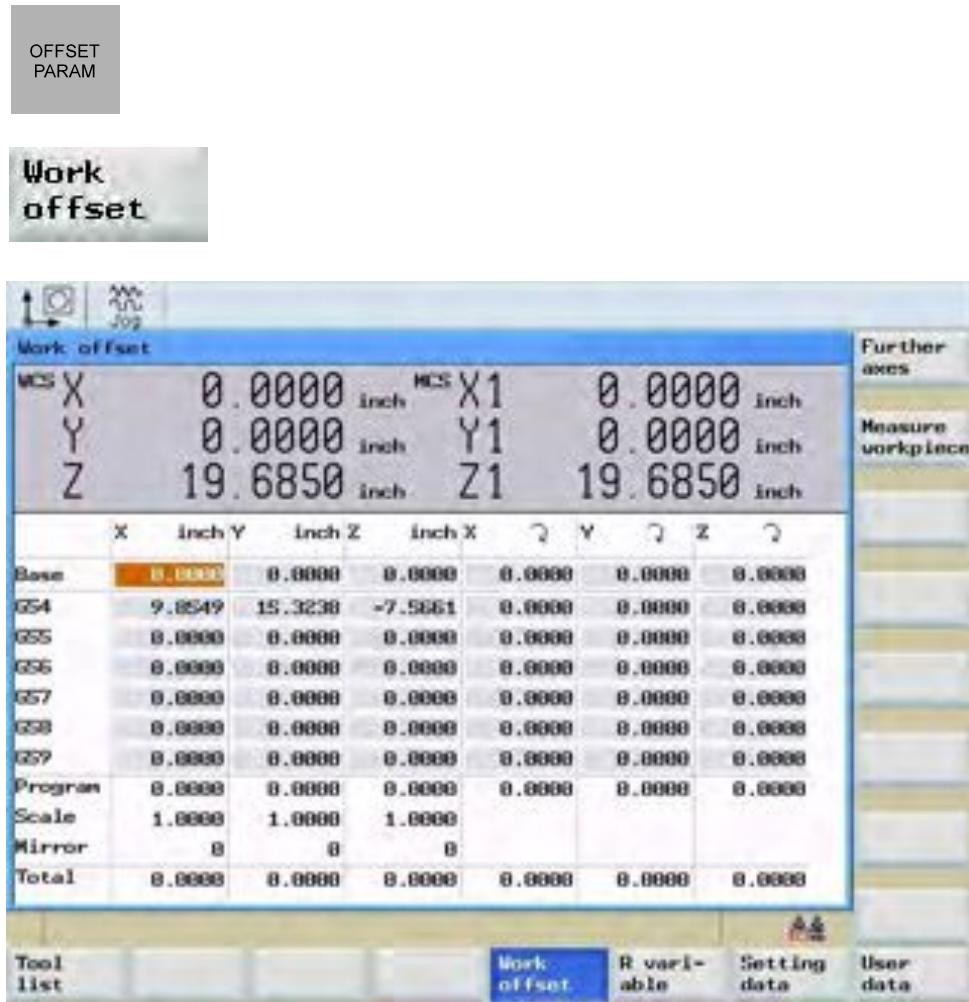
G59 - 6th. Zero Offset

# Section 3

## **Zero offset page in the control**

An offset page is available in the control, where the operator of the machine can input and check the zero offset values.

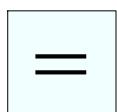
The following sequence shows how to perform this task.



When working with a milling machine a BCS offset can be used to offset the surface of the clamping device from the MCS. A WCS offset can then be used to the origin of the workpiece.

All values can be edited on the offset page.

To simplify the input a pocket calculator is integrated into the control:



By pressing the equals key in the respective input field, the calculator will start.

## Section 3

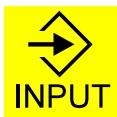
### Zero offset page in the control

Notes

After pressing the equal key the value from the input field will automatically be inserted into the calculator for further processing.



The calculation takes place with the following sequence.



And the calculation is transferred to the zero offset page with the following sequence.



	X inch	Y inch	Z inch
Base	0.0000	0.0000	0.0000
GS4	10.3150	12.4500	0.1700
GS5	0.0000	0.0000	0.0000
GS6	0.0000	0.0000	0.0000
GS7	0.0000	0.0000	0.0000
GS8	0.0000	0.0000	0.0000
GS9	0.0000	0.0000	0.0000

- 1 The calculated value will be transferred into the input field.

With the parameter field

X ↗ Y ↗ Z ↗

A coordinate rotation can also be calculated.

## Section 4

### Setting the zero offset

Notes

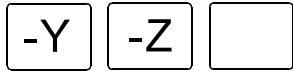
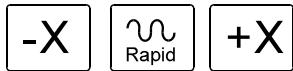
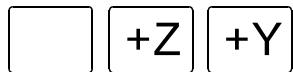
On a milling machine a value in all three geometry axis has to be set.

The offset is set using the scratch function in the control, to use scratch the spindle should first be started in MDA.



**T1 ; manual 3D-Dial indicator**

**M6**



Drive to the surface of the workpiece until indication.

## Section 4

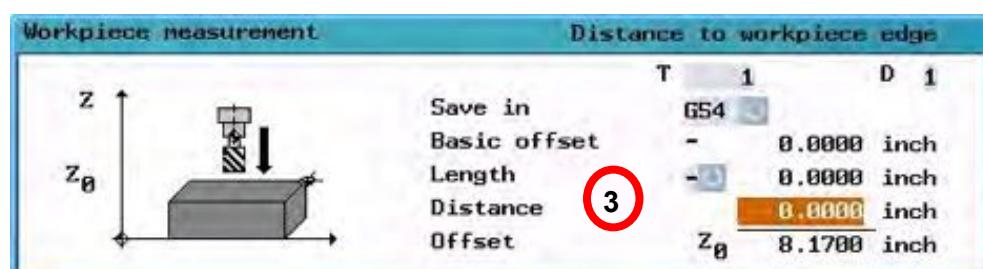
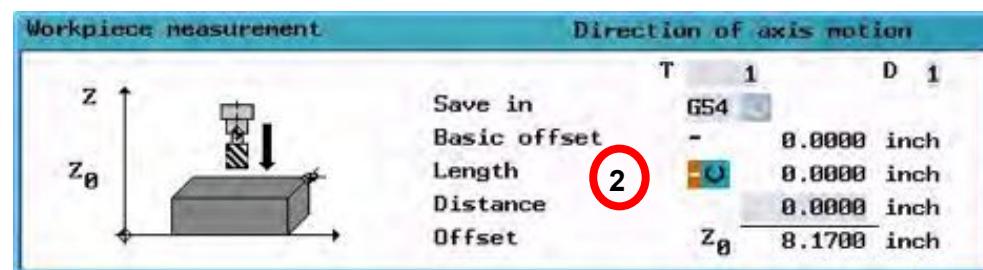
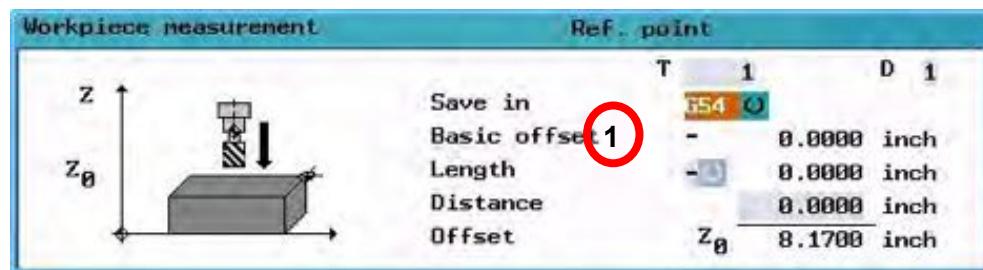
### Setting the zero offset

The following sequence shows how set the zero offset automatically on the machine.

Notes

Measure workpiece

Z



- 1 You can select the required offset with the Toggle field using the SELECT softkey, into which zero offset the value should be stored.
- 2 You can select the required direction with the Toggle field.
- 3 Enter a length at this position if using a setting block.

The value will be calculated and stored with the following sequence.

Set work offset

## Section 4

### Setting the zero offset

Notes

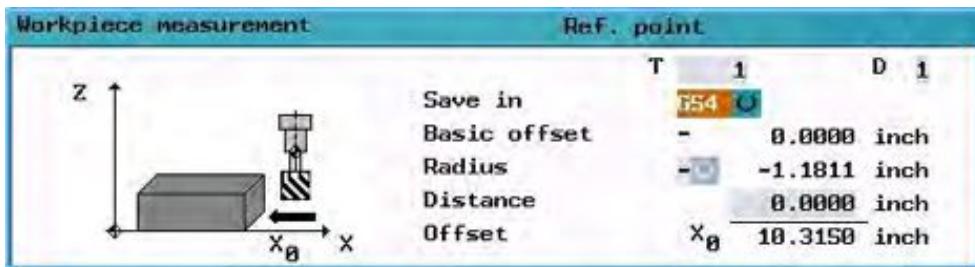
This should then be repeated for the X and Y axes respectively.

X

+Z  +Y

-X  Rapid  +X

-Y  -Z



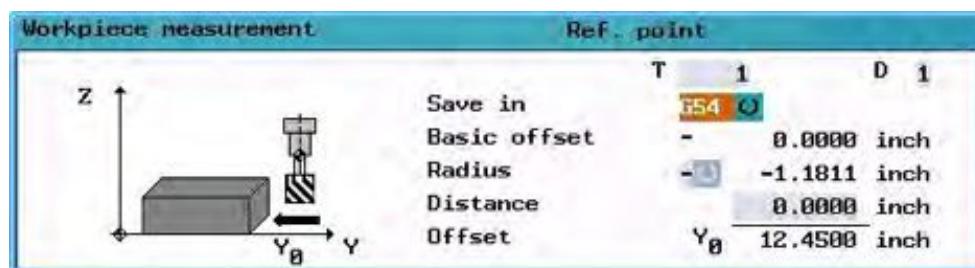
Set work offset

Y

+Z  +Y

-X  Rapid  +X

-Y  -Z



Set work offset

Back

## 1 Brief description

**Module objective:**

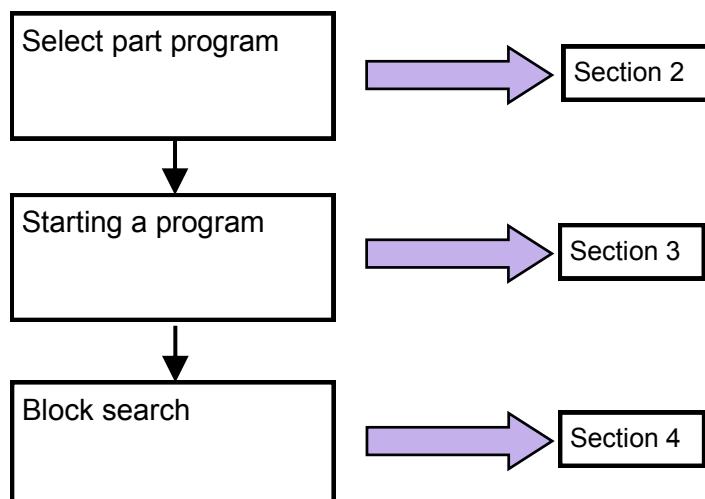
Upon completion of this module you can select and run an NC part program

**Module description:**

This module describes the steps which have to be taken to select and run an NC part program.

**Module content:**

Select part program  
Starting a program  
Block Search



## Section 2

### Select part program

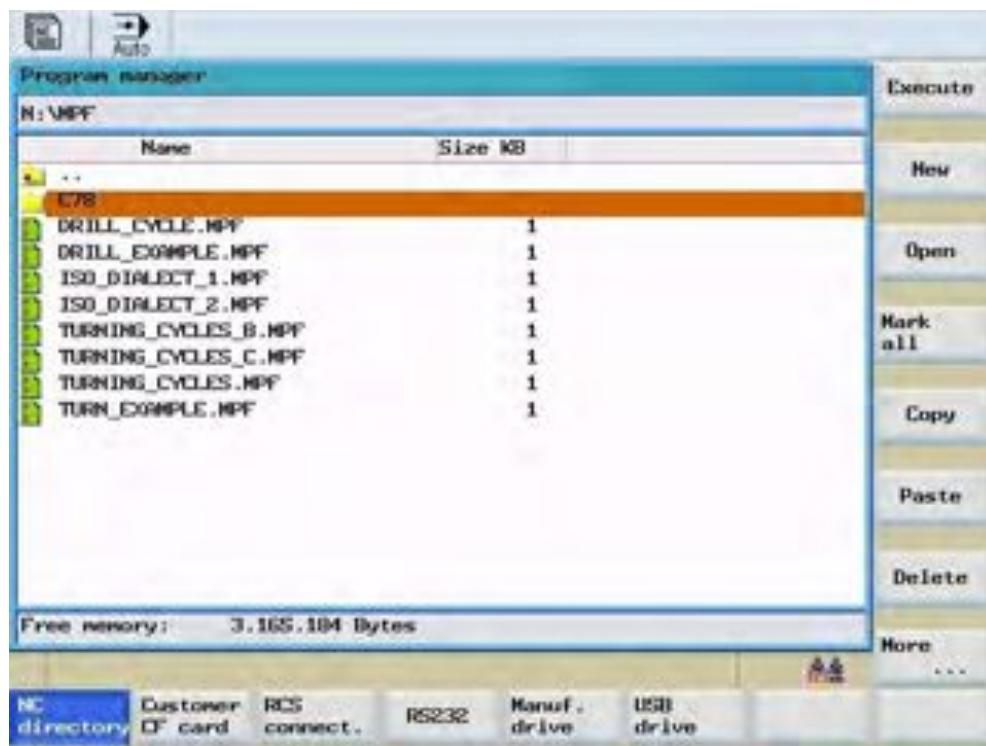
Before an NC program can be started in automatic, it must first be selected.

The program is selected in the “Program Manager” using the following sequence.



First the relevant directory has to be opened.

The required directory can be highlighted with the cursor keys.



The directory can be opened with the following keys.



or

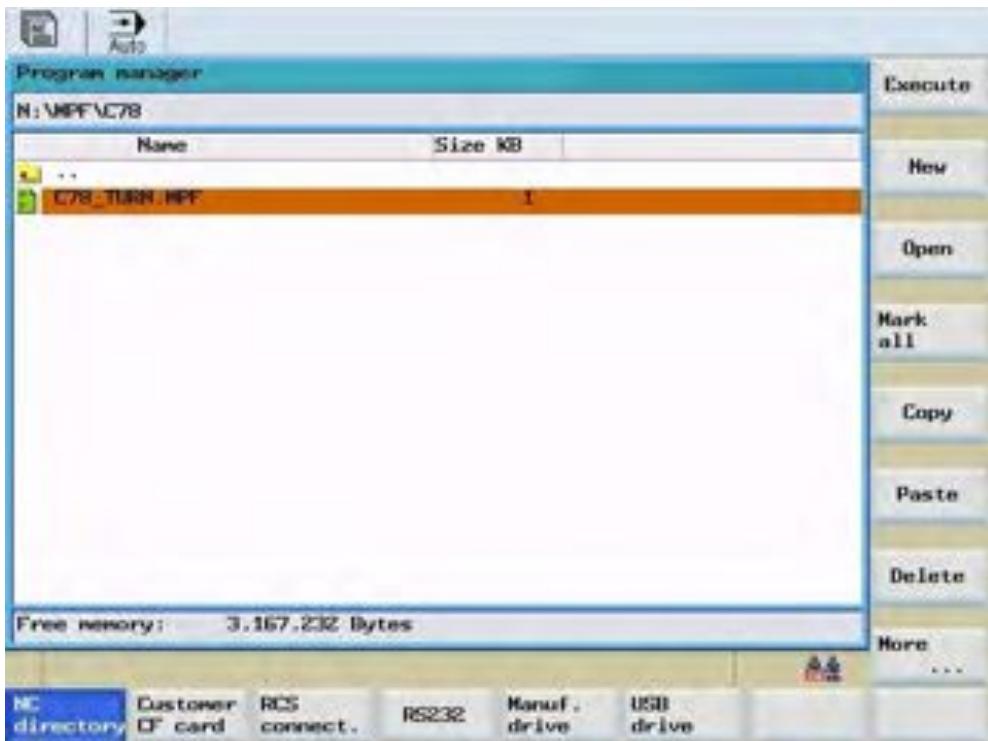


Notes

## Section 2

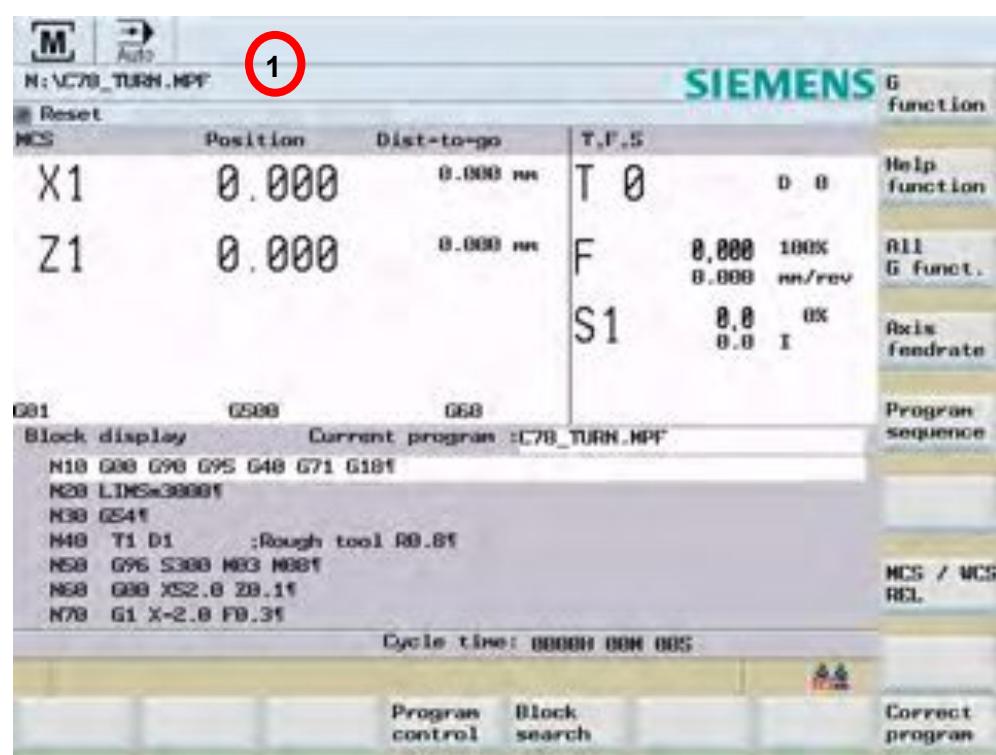
### Select part program

The program now has to be selected, use the following sequence.  
The required program can be highlighted with the cursor keys.



**Execute**

1 This softkey will transfer the required NC program into Automatic area and will be shown on the screen at this position.



Notes

## Section 3

### Starting a Program

Notes

Requirements prior to NC-Start.

The Automatic mode has to be selected prior to NC-Start.



The machine builder may also state prerequisites, read the handbook of the machine tool builder

It is possible with Single Block, to process one block at a time. To execute the whole part program this requires subsequent presses of the NC Start key.

The SINGLE BLOCK key can be found on the machine control panel:



Whether or not SINGLE BLOCK is active can be seen on the Status line of the control, as in the following diagram:

RESET SKP DRY ROV M01 PRT **SBL**

The SINGLE BLOCK function can be deactivated by pressing the key a second time.

The machine Stops at the end of the current block when the SINGLE BLOCK key is pressed in program.

The feedrate override can be used additionally to control the velocity of the axis.



### NC-Start

Taking care of the above mentioned functions you can continue with the Cycle Start key.

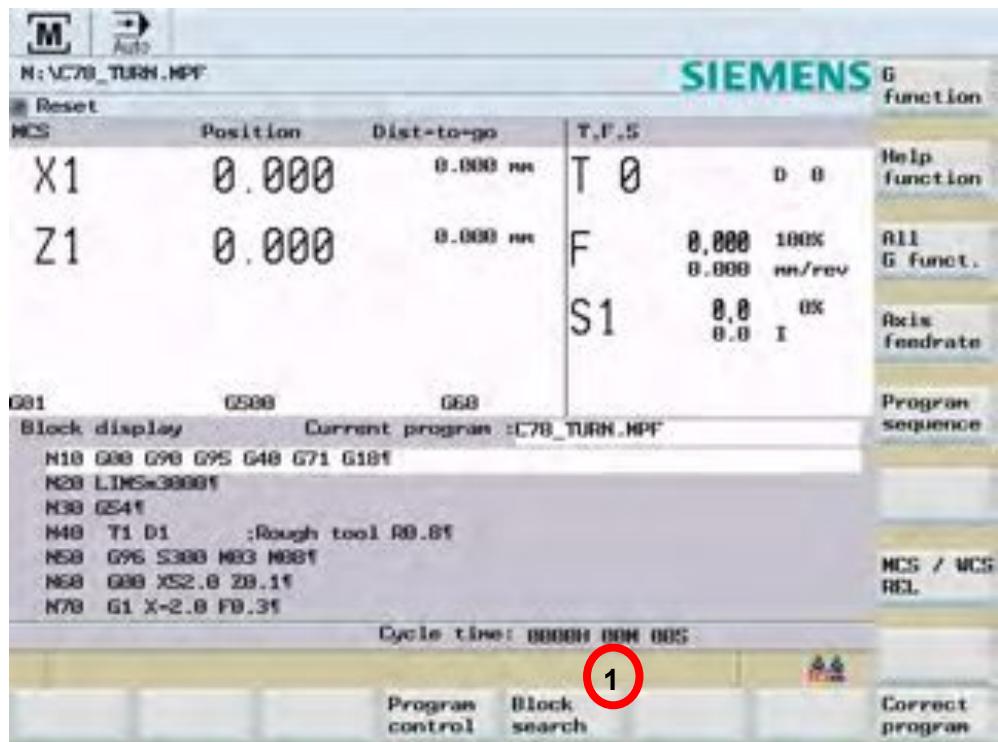


## Section 4

### Block Search

Notes

A block search is performed, so that an operator can begin at a suitable point other than the start of the NC program. This is achieved by following the sequence.



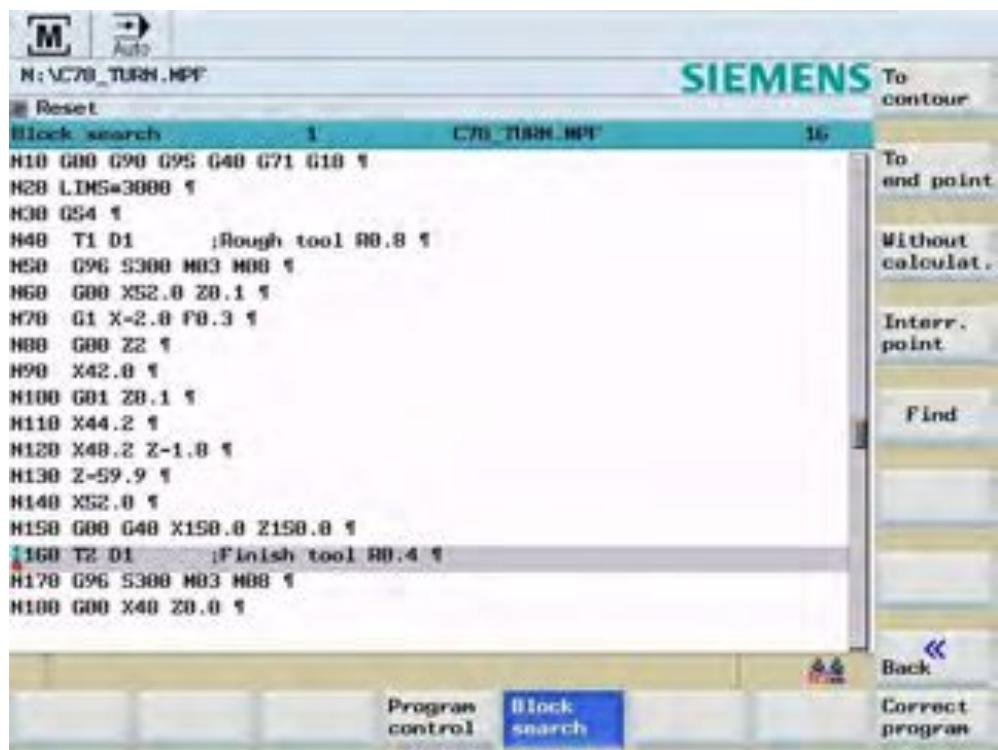
Block search

1

This softkey will allow you to choose which part of the program to start from.



This button can be used to cursor down to the required position in the program.



# Section 4

## Block Search

**Find**

Or this soft key can be used to find a specific point in the program e.g. block number or tool number.

## Notes

## To contour

Followed by “to contour” then “cycle start”.



1

M		01008	Channel 1 continue program with NC start								
N:\C78_TURN.MPF		SIEMENS				G function					
Stop		Wait: Axis feed override is 0									
MCS	Position	Dist-to-go	T,F,S								
X1	300.000	0.000 mm	T 1	D 1		Help function					
Z1	500.000	0.000 mm	F 0.000	100X		All G funct.					
			F 0.000	mm/min		Axis feedrate					
			S1 0.0	0X							
				159.2 I							
008	054	068				Program sequence					
Block display		Current program :C78_TURN.MPF									
N159 G90 G49 X150.0 Z150.0											
N160 T2 D1 ;Finish tool R0.45											
N170 G96 S300 M03 M88T											
N180 G90 X48 Z0.0F											
N199 G91 X-2.8 F0.1F											
N200 G90 Z2.0F											
N218 X42.0F											
Cycle time: 0000H 00M 00S											
		Program control	Block search			Correct program					



You have to press the cycle start button a second time to acknowledge the above alarm so that the program will execute in Auto mode.

M		Auto		SIEMENS				G function						
RUN		Wait: Axis feed override is 0												
MCS	Position	Dist-to-go	T,F,S					Help function						
-X1	194.185	-119.185 mm	T	1		0	1	All G funct.						
-Z1	335.399	-105.399 mm	F	11888.004	100%	11888.004	mm/min	Axes feedrate						
			S1	0.0	0%	159.2	I							
008	654	668	Program sequence											
Block display		Current program :C78_TURN.MPF												
N158 G80 G48 X158.0 Z158.01														
M168 T2 D1 ;Finish tool R8.45														
M179 G96 S3000 M03 M08T														
M180 G80 X48 Z0.01														
M190 G01 X-2.8 F9.1F														
M200 G00 Z2.0F														
M210 X42.0F														
Cycle time: 0000H 00W 01S														
Program control		Block search			Correct program									

## 1 Brief description

**Module objective:**

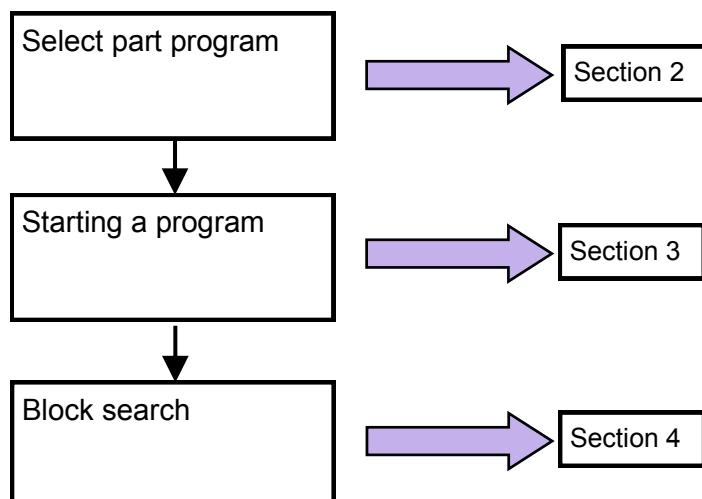
Upon completion of this module you can select and run an NC part program

**Module description:**

This module describes the steps which have to be taken to select and run an NC part program.

**Module content:**

Select part program  
Starting a program  
Block Search



## Section 2

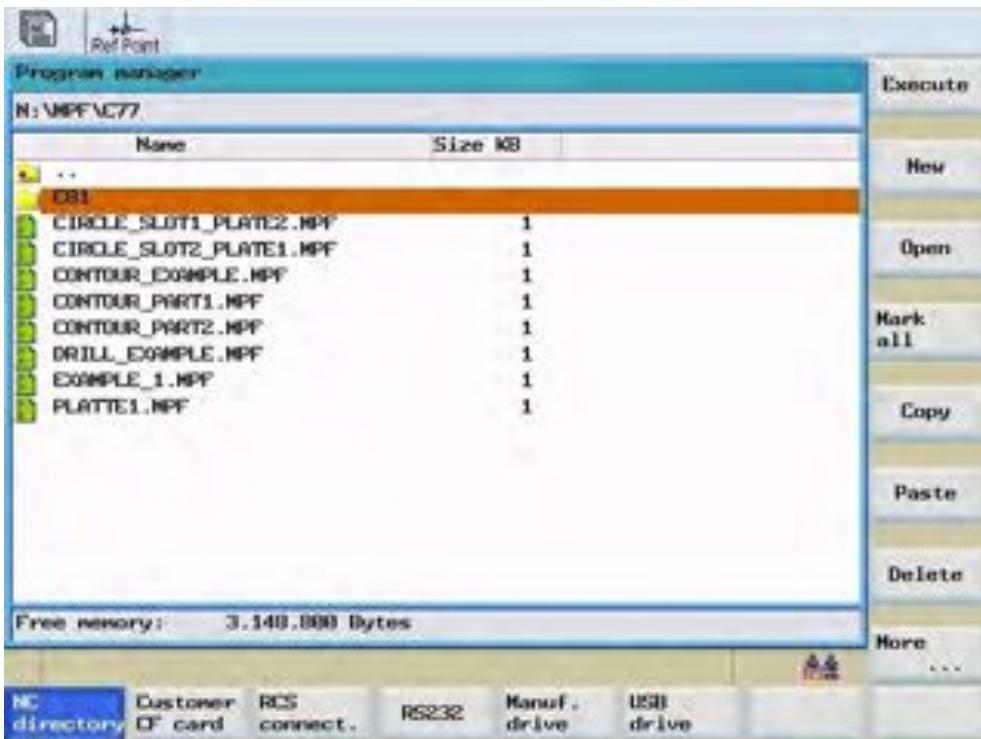
### Select part program

Before an NC program can be started in automatic, it must first be selected.

The program is selected in the “Program Manager” using the following sequence.



First the relevant directory has to be opened.  
The required directory can be highlighted with the cursor keys.



The directory can be opened with the following keys.



or

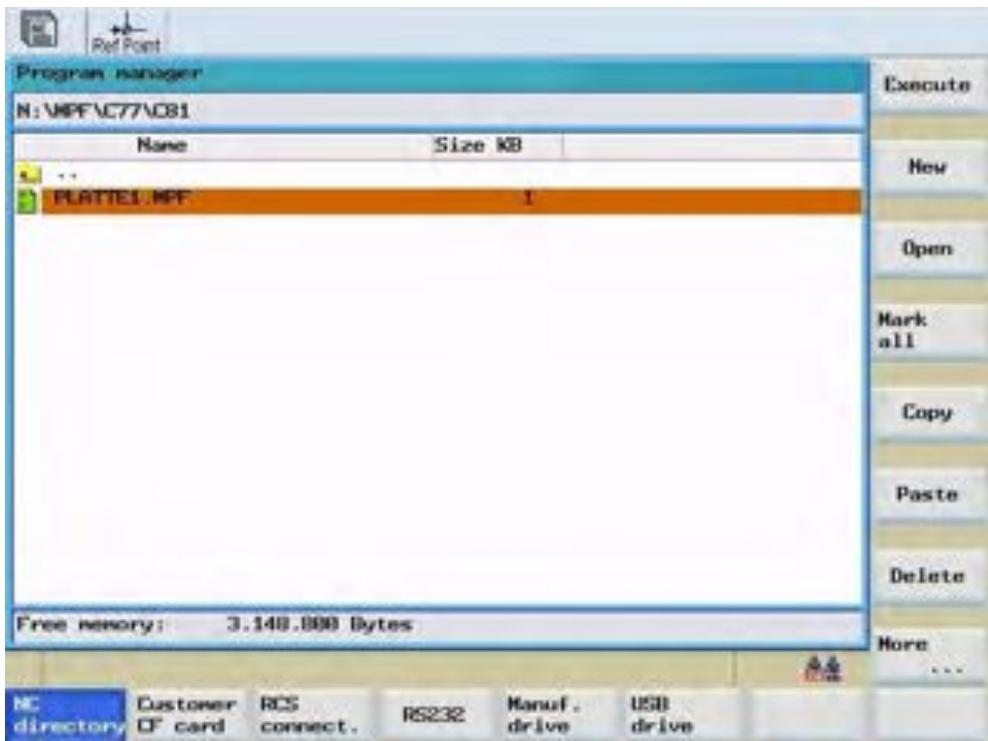


Notes

## Section 2

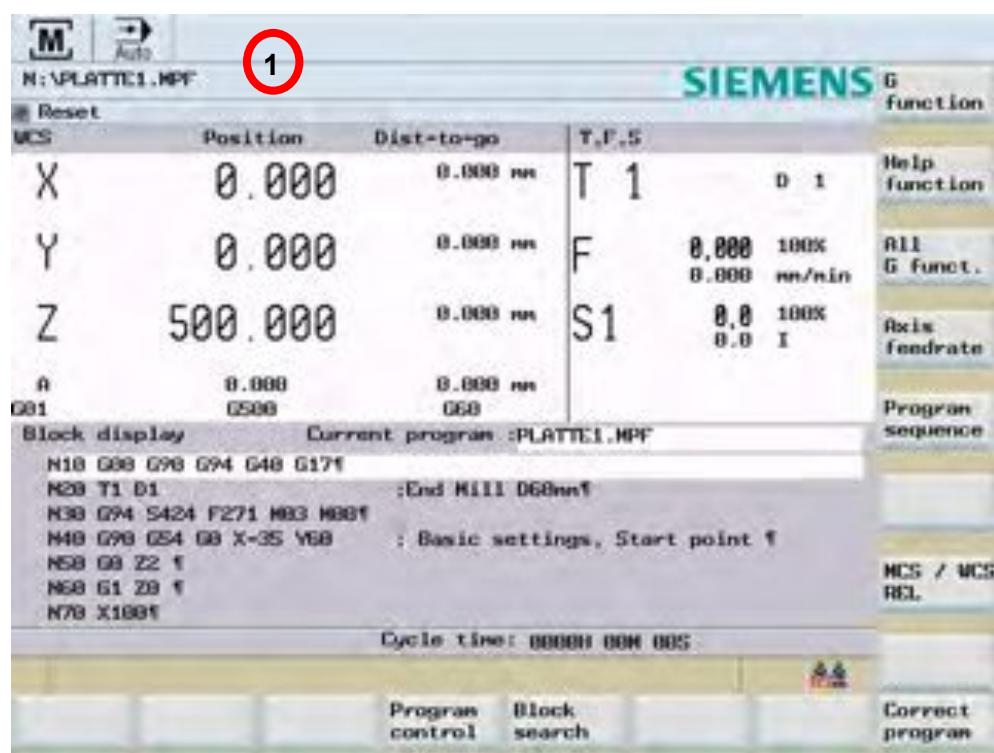
### Select part program

The program now has to be selected, use the following sequence.  
The required program can be highlighted with the cursor keys.



**Execute**

1 This softkey will transfer the required NC program into Automatic area and will be shown on the screen at this position.



Notes

## Section 3

### Starting a Program

Notes

Requirements prior to NC-Start.

The Automatic mode has to be selected prior to NC-Start.



The machine builder may also state prerequisites, read the handbook of the machine tool builder

It is possible with Single Block, to process one block at a time. To execute the whole part program this requires subsequent presses of the NC Start key.

The SINGLE BLOCK key can be found on the machine control panel:



Whether or not SINGLE BLOCK is active can be seen on the Status line of the control, as in the following diagram:

RESET SKP DRY ROV M01 PRT **SBL**

The SINGLE BLOCK function can be deactivated by pressing the key a second time.

The machine Stops at the end of the current block when the SINGLE BLOCK key is pressed in program.

The feedrate override can be used additionally to control the velocity of the axis.



### NC-Start

Taking care of the above mentioned functions you can continue with the Cycle Start key.

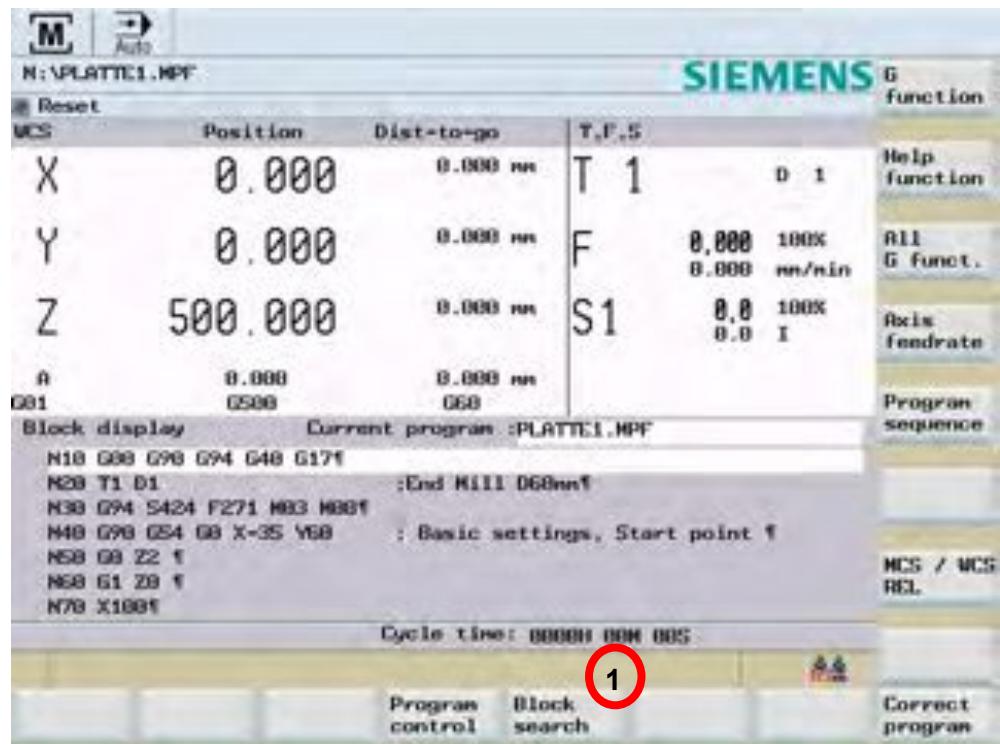


## Section 4

### Block Search

Notes

A block search is performed, so that an operator can begin at a suitable point other than the start of the NC program. This is achieved by following the sequence.

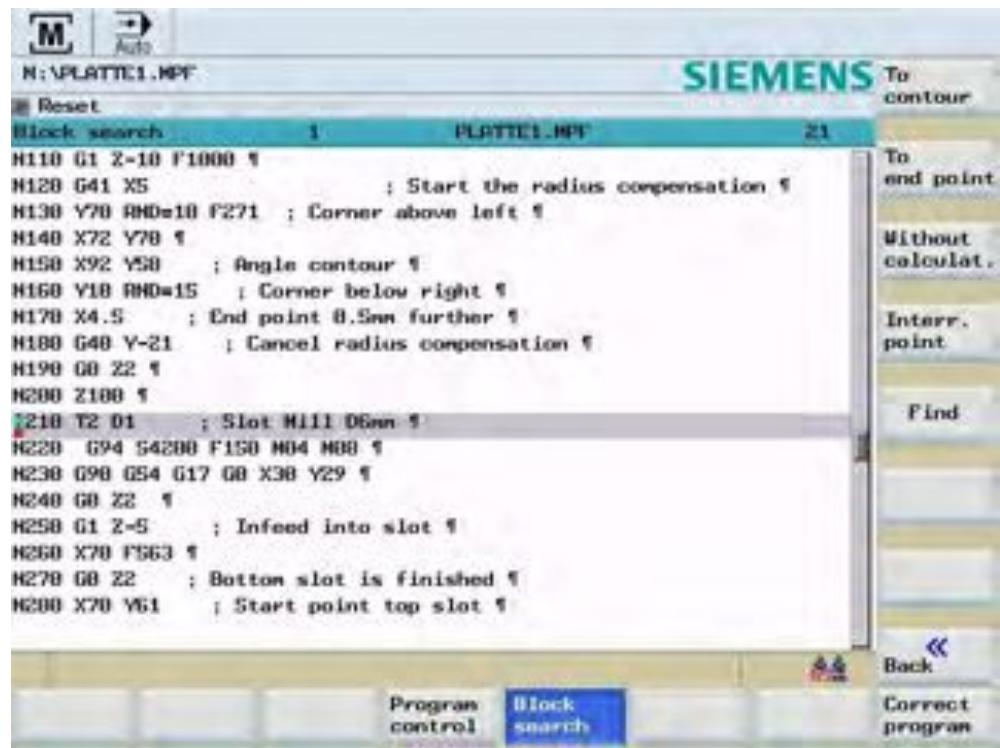


Block  
search

- 1 This softkey will allow you to choose which part of the program to start from.



This button can be used to cursor down to the required position in the program.



## Section 4

### Block Search

Find

Or this soft key can be used to find a specific point in the program e.g. block number or tool number.

Notes

To contour

Followed by “to contour” then “cycle start”.



The screenshot shows the Siemens SINUMERIK 802D SI control interface. The top status bar indicates 'M' (Machine On), 'Auto' mode, and '010208 Channel 1 continue program with NC start'. The main screen displays a G-code program for a slot mill operation. The 'To contour' function has been selected, and the 'Cycle Start' button is highlighted in green. The program details include coordinates X -6358.001, Y -9886.315, Z 5381.321, and various feed rates and tool data. A note in the bottom right corner says 'HCS / MCS REL.' The bottom navigation bar includes 'Program control', 'Block search', and 'Correct program' buttons.



You have to press the cycle start button a second time to acknowledge the above alarm so that the program will execute in Auto mode.

This screenshot shows the same Siemens SINUMERIK 802D SI control interface as the previous one, but with a different status message: 'Wait: Exact stop not reached'. The 'To contour' function and the 'Cycle Start' button remain highlighted. The program details are identical to the previous screenshot. The bottom navigation bar includes 'Program control', 'Block search', and 'Correct program' buttons.

## Section 1 Brief description

**Module objective:**

Upon completion of this module you can:-

Connect to the controller and transfer data using The RCS802 Data Transfer Tool.

Understand the use of licences

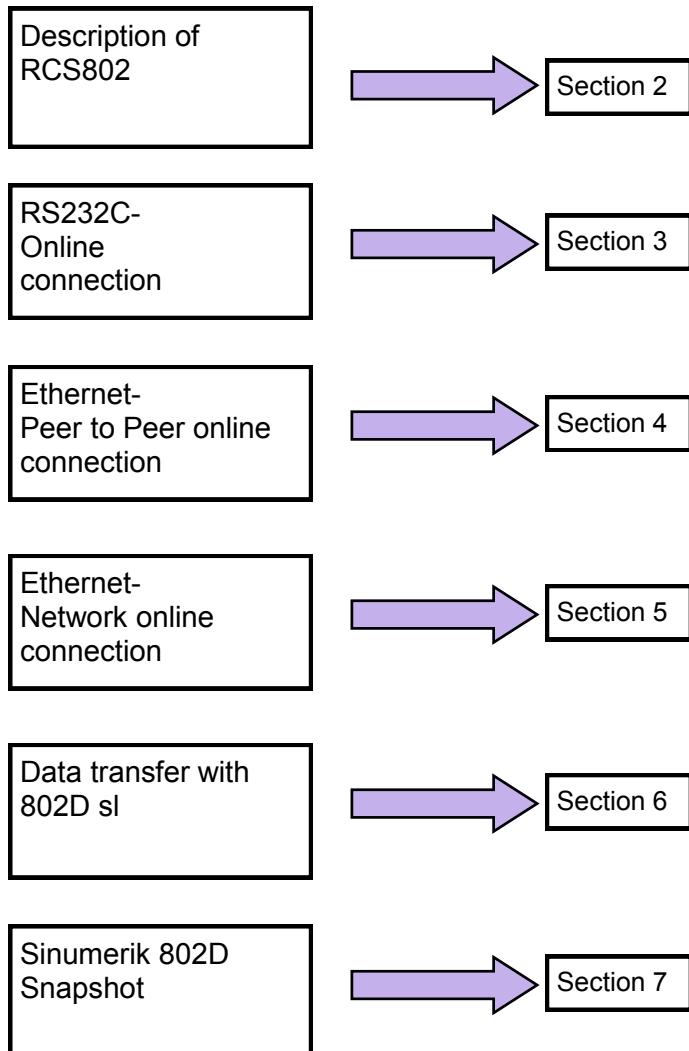
Use Sinumerik 802D Snapshot.

**Module description:**

Data resident on the controller can be saved to an external Laptop/PC for backup purposes. This data can be transferred back to the controller as and when required.

Alternatively, data created off-line such as part programs can be transferred to the controller in the same way.

This module describes the three methods of connecting to the controller and shows an example of how the RCS802 Data Transfer Tool can be used to save/restore data.

**Module content:**

# Section 2

## Description of RCS802.

The RCS802 program is available as part of the toolbox supplied with the controller.

The software can be used in several ways:-commissioning/service work, data transfer and remote diagnosis.

Connection is via Ethernet or RS232.

There are two Ethernet connection methods:-Peer to Peer & Network.

The following pages will describe how to make a connection with each of the above protocols.

## Section 3

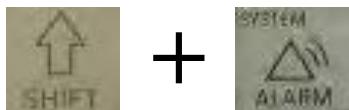
## **RS232C online connection**

The following sequence of operations should be carried out to obtain an online serial connection to the controller.

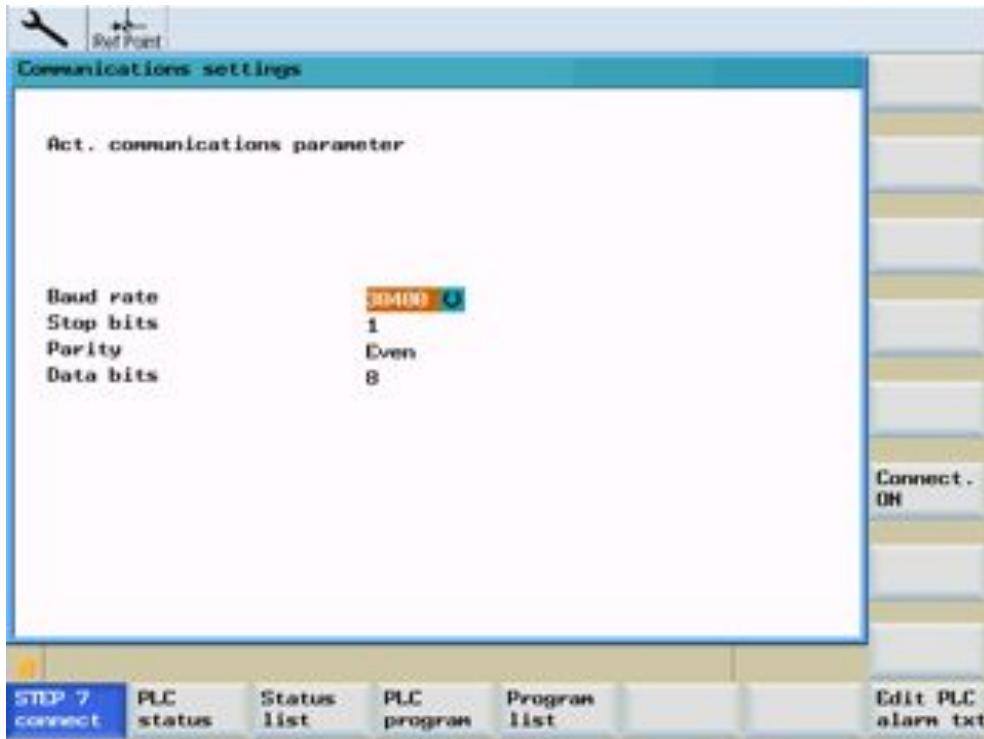
The PG/PC should be connected to the 802D sl controller (Socket X8) with a standard RS232 cable.

The communication port on the controller must be activated.

Access the System area with the following simultaneous key presses:-



Now select the “PLC” softkey



This opens the page where the communication parameters are set up. When the parameters have been set, the connection is activated with the “Connect, ON” softkey.



## Notes

## Section 3

### RS232C online connection

When the connection is activated the “Connect. ON” softkey changes to “Connect. OFF”.

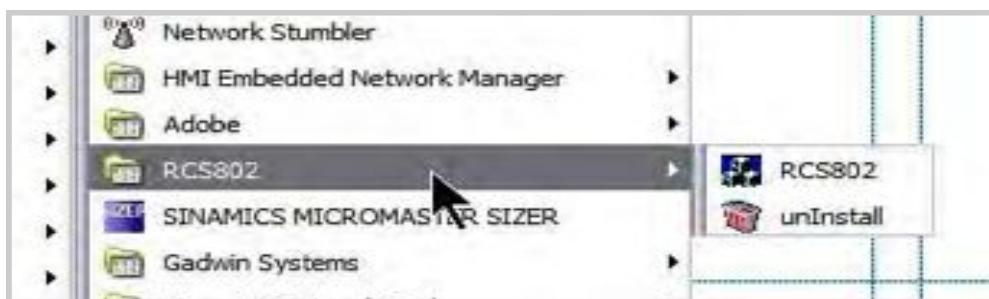
Notes

Connect.  
OFF

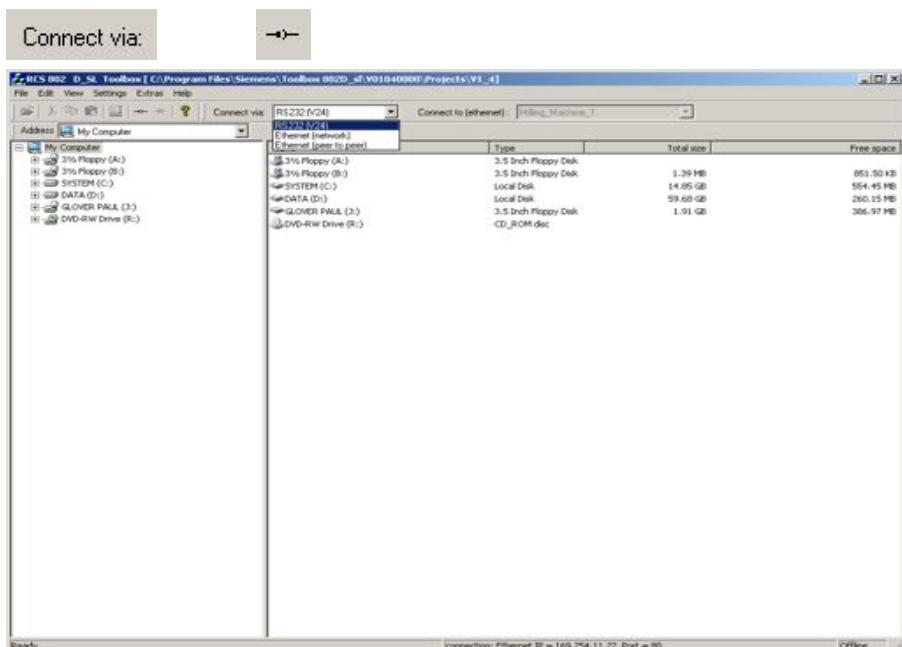
Also a connection icon appears at the bottom right of the screen.



The RCS802 program should now be opened.



Choose the “RS232[V24]” option from the “Connect via” drop down menu. Then select the “connect” button



Now the relevant communication port of the PG/PC can be selected and then configured.

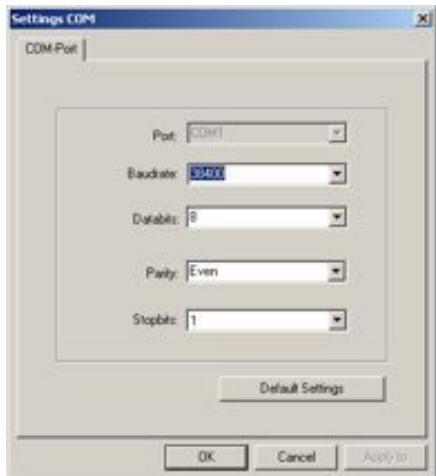


## Section 3

### RS232C online connection

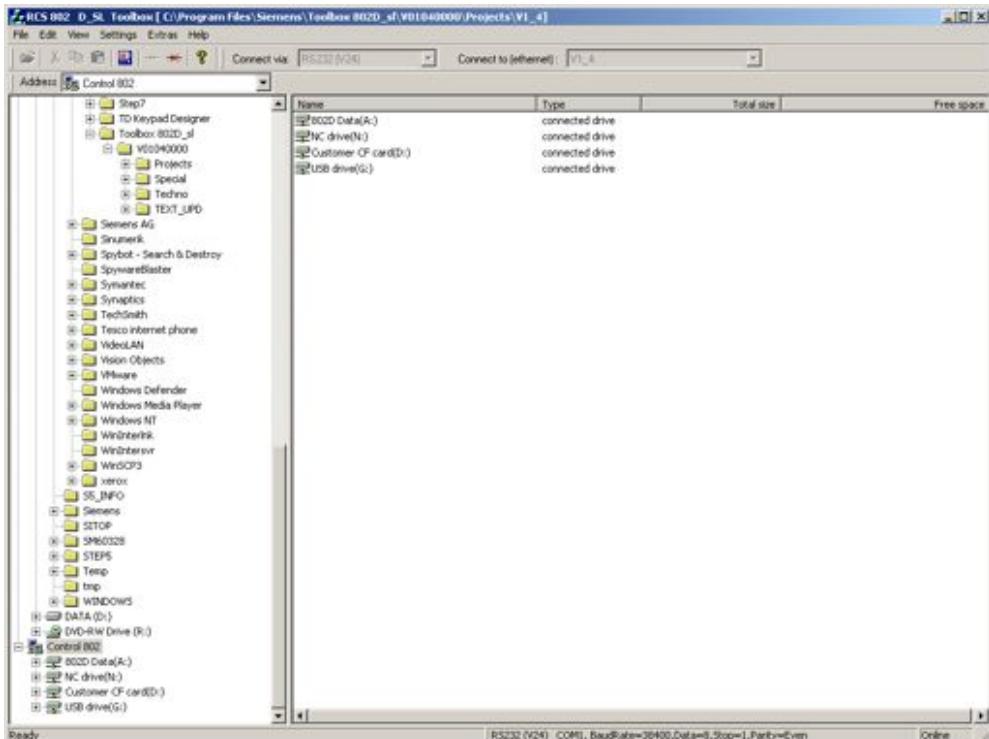
The settings should be changed to match the controller settings set on page 2.

Notes



Now select the "OK" button (Twice)

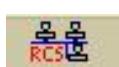
OK



The connection should now be complete. This is confirmed by the appearance of the 802D sl drives in the bottom of the left hand window



Another indication of a successful connection is the appearance of the following icon in the bottom right hand of the controller screen.



## Section 4

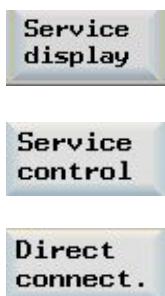
### Ethernet-Peer to Peer online connection

To use the Peer to Peer method of connection first connect the PG/PC to socket X5 on the controller using a crossover Ethernet cable.  
The connection on the controller is activated first. This is done from the "System" area.

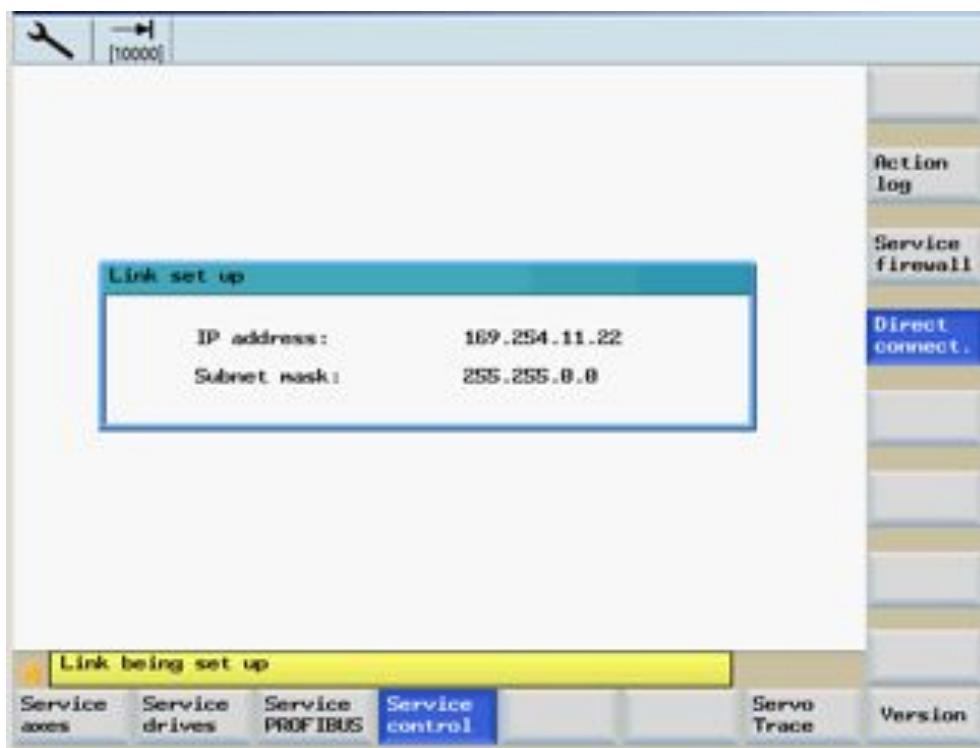
Notes



The following sequence of softkey presses is then followed.



The following page is then shown as the port is activated.



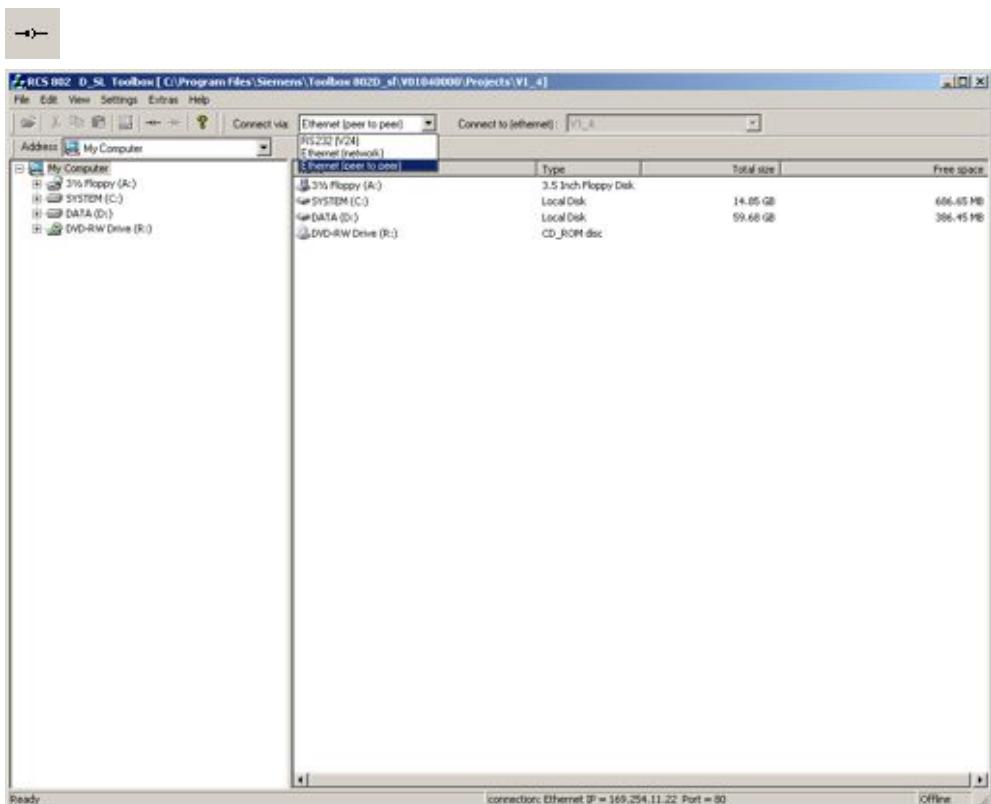
The IP Address and subnet mask are default and are independent of the settings in the "Service network" area which are disabled when the "Peer to Peer" mode is selected.

## Section 4

### Ethernet-Peer to Peer online connection

The RCS802 software is now opened and the “Ethernet [peer to peer]” option selected from the drop down menu.

The “Connect” button is then selected.



Notes

The dialogue box is for information only and shows the default IP settings of the controller. No changes can be made.

The procedure is completed with the “OK” button.



If the connection is successful, the drives of the controller can be seen in the left hand window of the RCS802 Software, also the following icon appears in the bottom right of the controller screen.



## Section 5

### Ethernet-Network online connection

#### Licences

This method of connection is primarily for “Remote Diagnostics” use.

To function correctly a licence is required.

To check whether a licence is Installed the “Automatic Licence Manager” software is opened from the Start Menu.



#### Notes

The example above shows that the RCS802 Licence is stored on the C Drive along with the licences of other software applications.

The licences can be transferred to/from other media such as floppy drives, memory sticks etc via the “Licence Key” Menu.

#### Making a connection

To set the controller up to communicate, enter the “System” Area:-



The password needs to be active to a minimum of level 3 (Customer).

**Module C21** explains the various protection levels



A user must also be logged in. There are three default users, SIEMENS, SERVICE and PEERTOPEER.

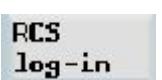
The SIEMENS user is not available for general use.

The PEERTOPEER user is automatically logged in when the “Ethernet-PeerToPeer” connection described in section 4 is selected.

The SERVICE user is available for use and will be used for this module.

The password for this user is:-\$SERVICE\$.

To log into an existing account the “RCS log-in” softkey is used

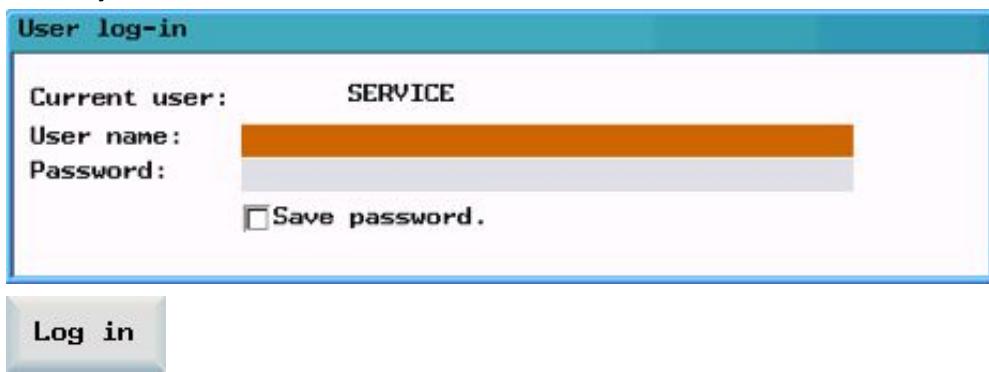


## Section 5

### Ethernet-Network online connection

The user name and password are entered followed by the “Log in” Softkey.

Notes

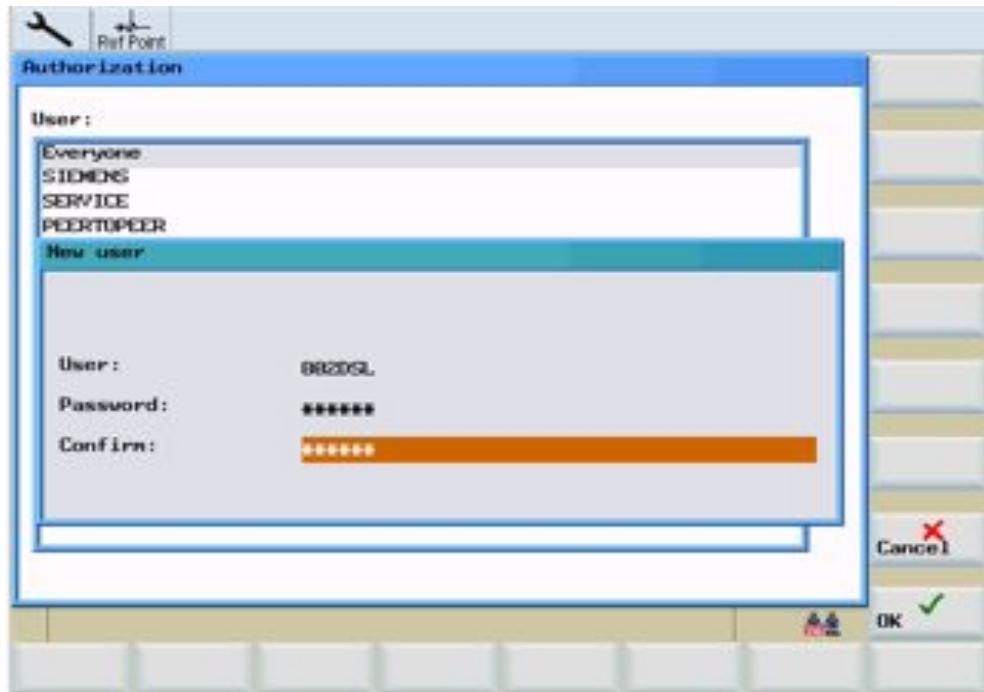


Log in

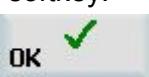
Use the “Back” softkey to return to the “System” page.



If a new user is to be added, follow the following softkey sequence:-



A user name and password can now be entered followed by the “OK” softkey.

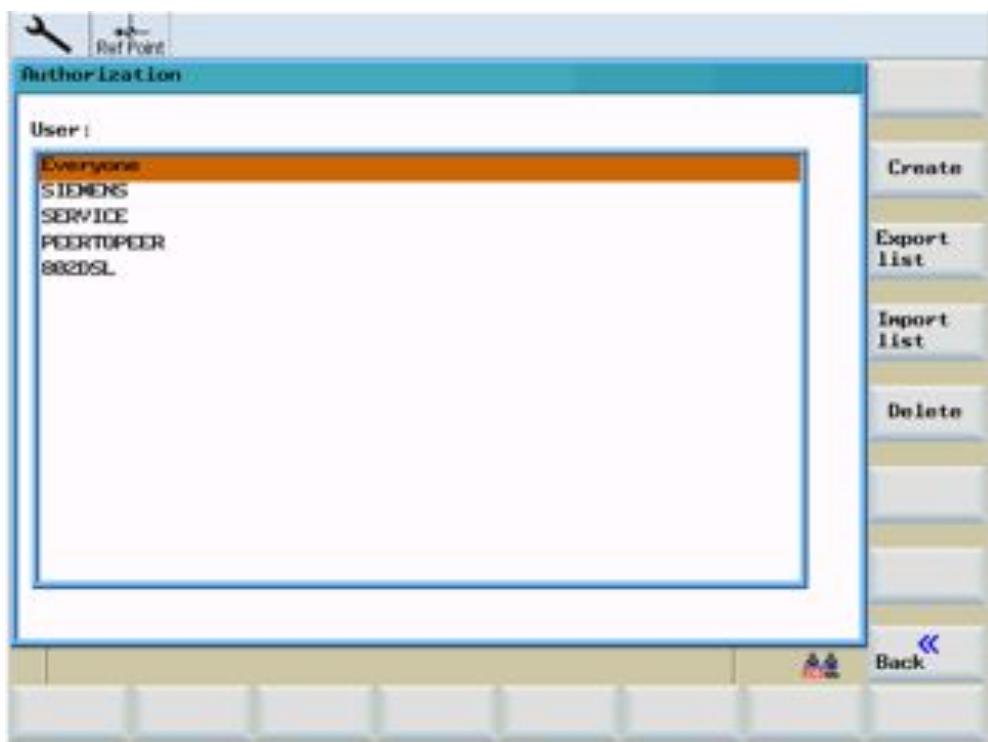


## Section 5

### Ethernet-Network online connection

The newly created user now appears in the list.

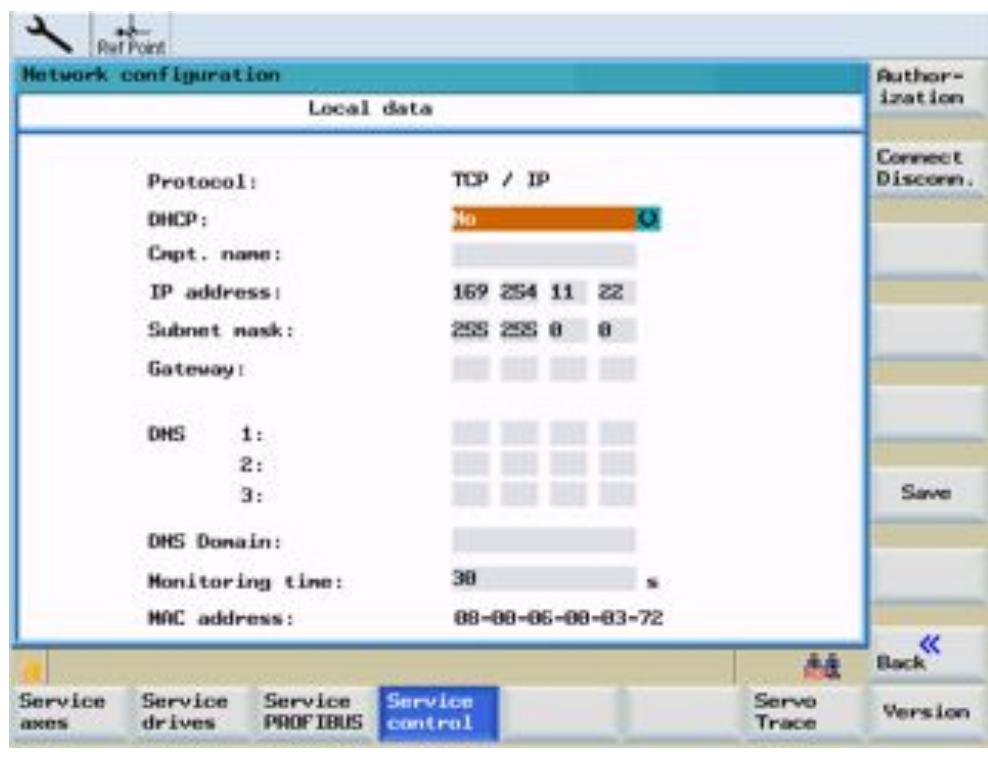
Notes



To continue setting up the control select the "Back" softkey



This page is used for entering network settings. These settings will depend on the type of remote access required. For a simple connection between the controller and a PC/PG we are interested in the IP address and Subnet mask. More detailed information will be required for connection via factory networks etc.



## Section 5

### Ethernet-Network online connection

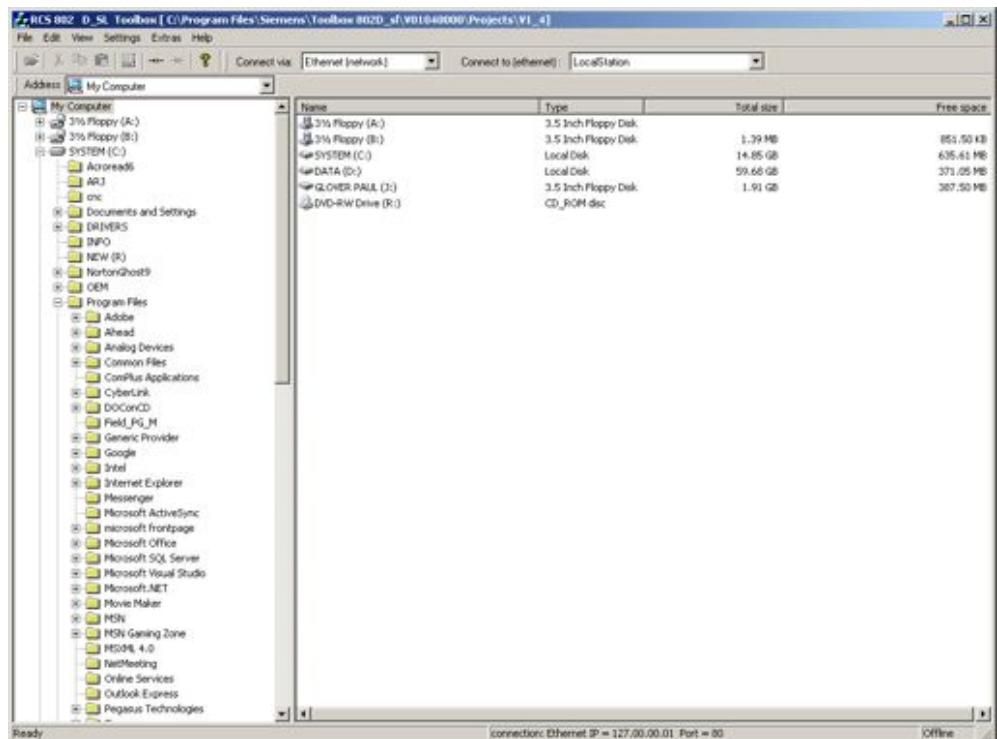
Any changes to the network settings are activated with the “Save” then “OK” softkeys.

Save

OK ✓

Notes

Now that the controller is setup the RCS802 program is opened. From the drop down menu the option “Ethernet [network]” is selected.



When the “Connect” button is selected a dialogue box will appear. Then the “Configure” button must be selected



## Section 5

### Ethernet-Network online connection



Notes

This is where the connection can be named and the IP address of the controller we are connecting to entered.  
Select the “Add” button and enter the Station name and IP address followed by the “Save” button and the “OK” button.

Add



Save



OK

## Section 5

### Ethernet-Network online connection

In the dialogue box, enter the log on name and password of the user which is currently logged in on the controller. Then select “OK”

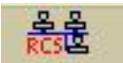
Notes



A successful connection is shown by the 802D sl drives appearing in the left hand window of the RCS802 program.



Another indication of a successful connection is the appearance of the following icon in the bottom right hand of the controller screen.



## Section 6

### Data Transfer with RCS802

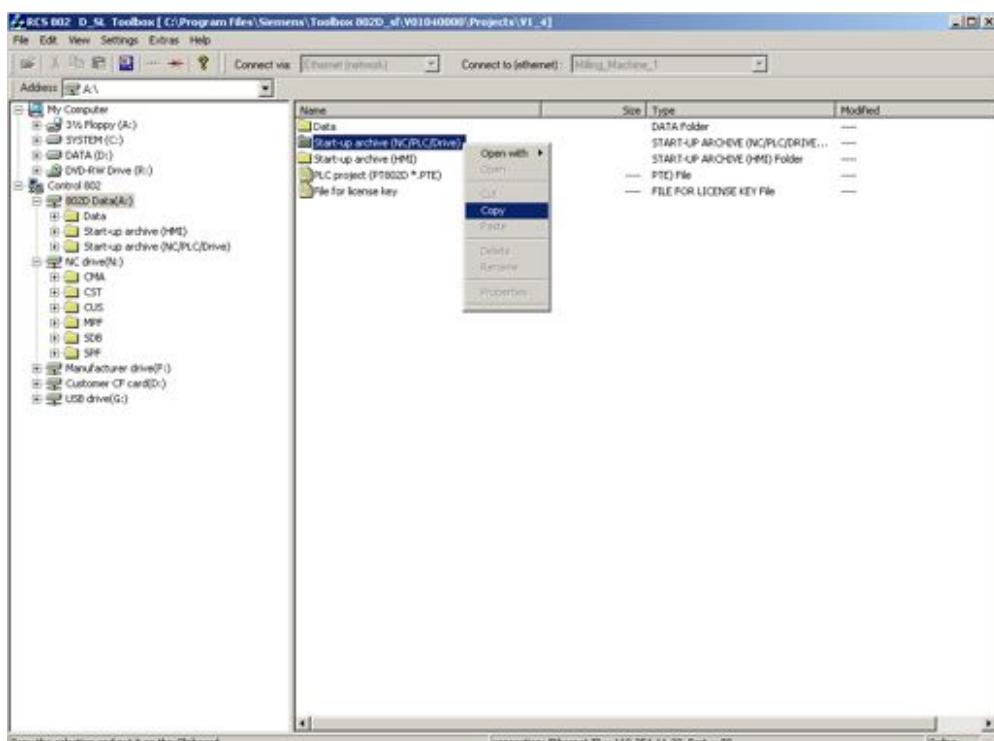
#### Saving data to external PG/PC

Notes

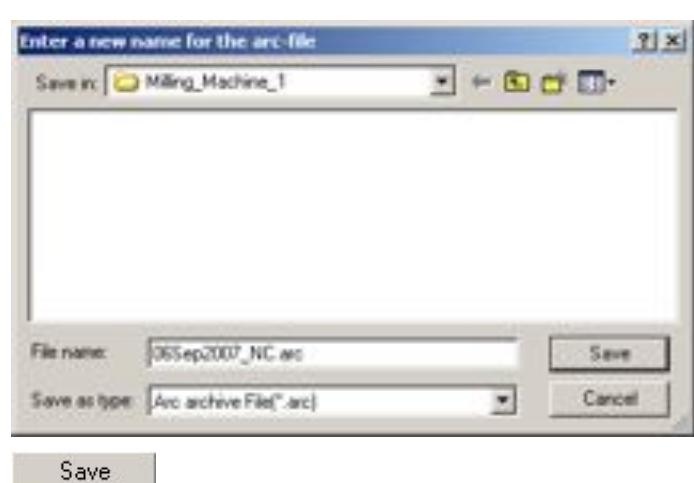
Data such as programs can be transferred to and from the controller using the RCS802 software.

The directory structure and functionality of the software is similar to Windows Explorer.

Example:-To create an NC/PLC/DRIVES series start up file, right click on the folder named “Start-up archive (NC/PLC/Drive)” and select the copy option. Now select a destination on the PC/PG, right click and paste the file.



If you need to change the name of the archive from default you can do this here by selecting “Yes”. Enter the desired file name followed by “Save”



## Section 6

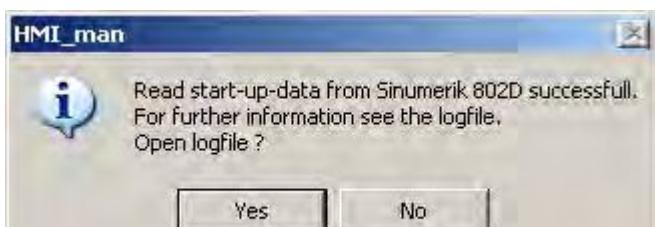
### Data Transfer with RCS802

The progress of the data transfer is displayed

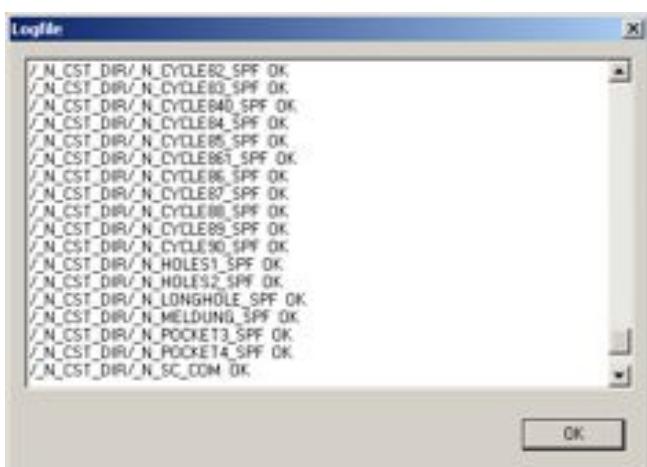
Notes



Once the transfer is complete a logfile of the transferred files can be viewed.



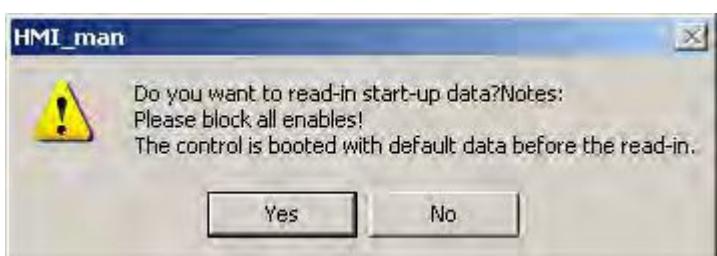
Yes



OK

### Restoring Data with external PG/PC

The control can be restored using the same method. The Archive file is copied from the PG/PC to the "Start-up archive (NC/PLC/Drive)" folder of Drive A on the controller.



Yes

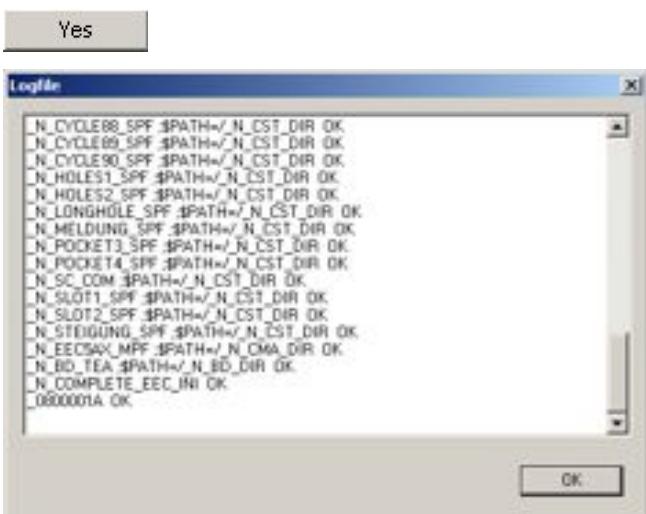
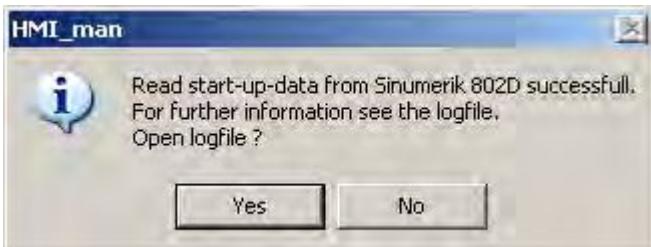
## Section 6

### Data Transfer with RCS802



Notes

Allow data transfer to complete then view the logfile if required



OK

After the "OK" button is selected the controller will reboot

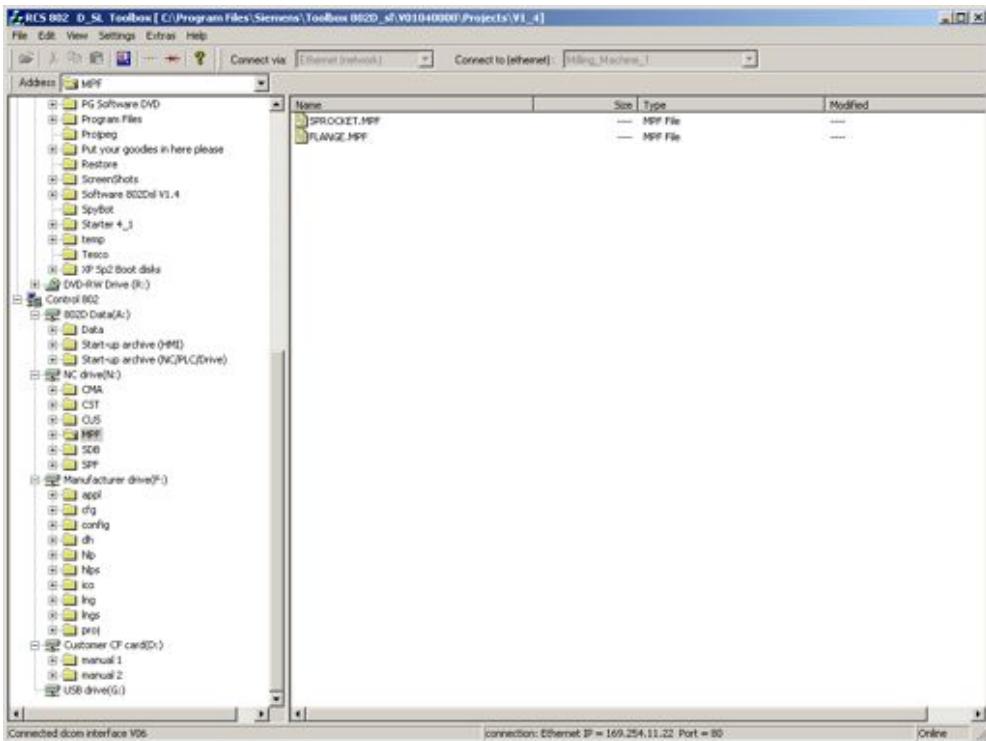


Certain dialogue boxes will be shown on the controller which are quite normal. The procedure finishes with an NCK reset and resets the password as well as the RCS802 losing connection with the controller.

## Section 6

### Data Transfer with RCS802

In this example an NCK archive was created and restored. Individual data such as Tool Offsets, Part Programs etc can be copied and pasted to and from the controller. The screen below shows the contents of the MPF directory. This is reached by expanding the menu tree in the left hand Window. Other data is found in the same way.



Notes

As well as the three drives on the controller (Drives A, N & F), data can be transferred to and from the other drives (Drives D & G) Provided that the media has been inserted i.e. CF card in drive D and USB memory device in drive G

## Section 7

### Sinumerik 802D Snapshot

The RCS802 software package also has a function called "Snapshot" Like the "Ethernet [Network]" option, this is dependant on a license (See page 7)

If a licence is detected the "Sinumerik 802D Snapshot" button is available on the toolbar. No licence and the button will be greyed out.



Selecting this button will open up a window which will show exactly the same screen that the controller is showing.

When the mouse pointer is on the Snapshot screen and right click is selected, a list is displayed which can be used to replicate key board functions.

Changing a screen via Snapshot will result in the screen changing on the controller.

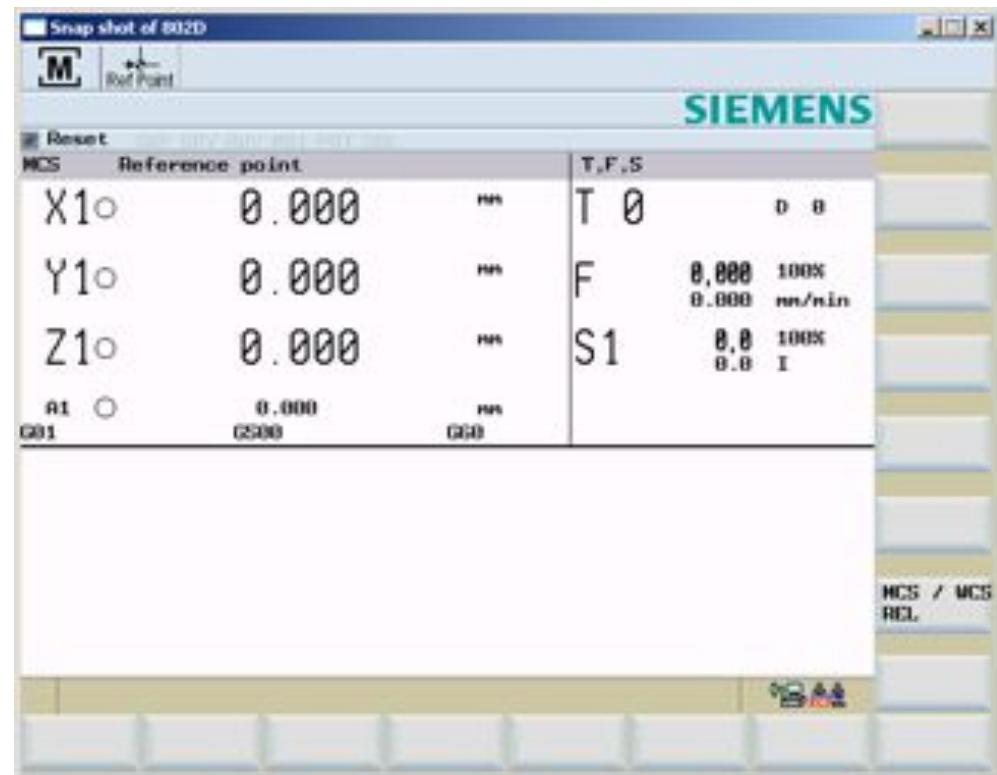
This is very useful for Remote Diagnosis and service/commissioning work.

## Section 7

### Sinumerik 802D Snapshot

Click the “Sinumerik 802D Snapshot” button to show a real-time display of the controller screen

Notes



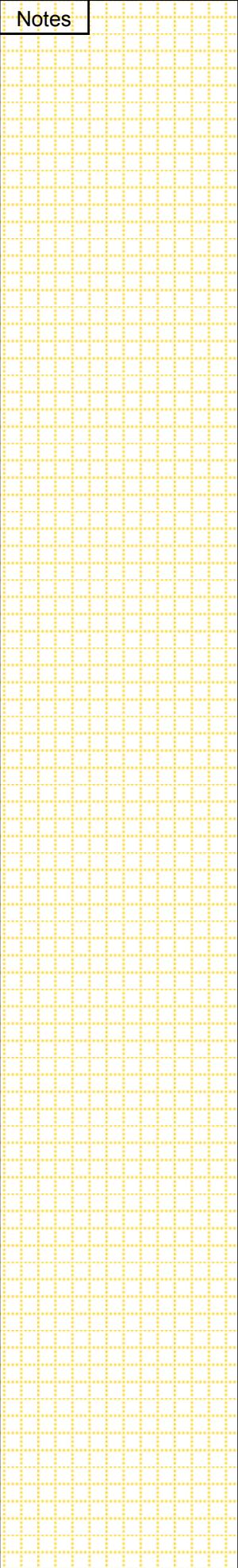
Right click the mouse over the Snapshot screen to get a list of keyboard functions.

Window settings	
Position	<ALT X>
Offset Param	<ALT C>
Program	<ALT V>
Program Manger	<ALT B>
System	<ALT N>
Alarm	<ALT M>
Customer	<ALT Y>
Caps lock	<ALT L>
Chinese Input	<ALT S>
Help information	<ALT H>
Recall	<F9>
More	<SHIFT F9>
Calculator	<CTRL A>
Alarm Cancel	<ALT A>

End of Module C1

Blank Page

Notes



## 1 Brief description

### Module objective:

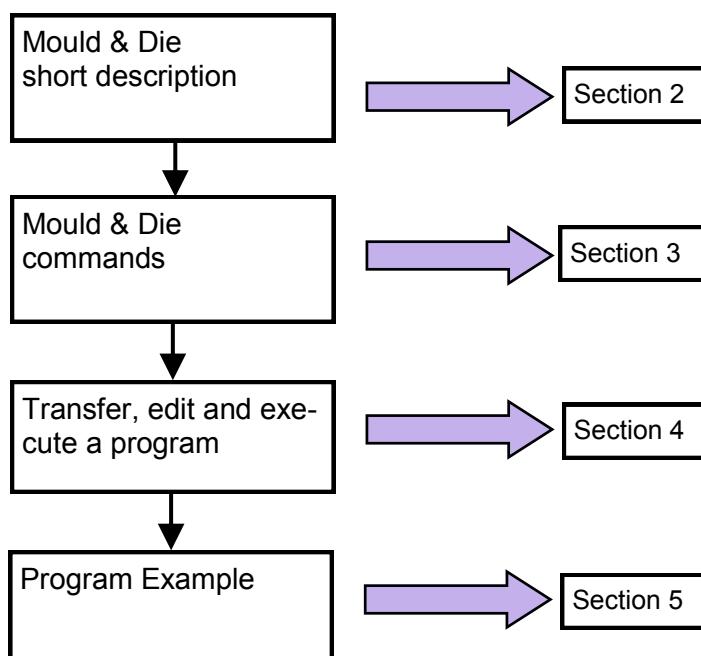
Upon completion of this module you can transfer, edit / program and execute a Mould & Die program on the 802Dsl.

### Module description:

Mold & Die functions and commands allow to execute especially large programs with G1 blocks.

### Module content:

Mould & Die short description  
Mould & Die commands  
Transfer, edit and execute program  
Program Example



## Section 2

### Mould & Die short description

#### Introduction

Notes

#### CAD -> CAM -> CNC process chain

Many CNC programs for freeform surface machining come from CAM systems. The CAM system obtains the workpiece geometry from a CAD system.

When executing CAM programs in the HSC range for Mold&Die, the control has to process high feedrates with short NC/G1 blocks with single points. The user expects good **surface finish** with great **accuracy** in the µm range with high **machining feedrates** up to 10 m/min. By applying different machining strategies, the user can „fine tune“ the program with the aid of Sinumerik application commands.

When **roughing**, **speed** is the priority due to smoothing the contour.

When **finishing**, **accuracy and surface quality** is the priority.

In both cases, specifying a tolerance ensures that the machining contour is observed in order to achieve the desired surface finish. When defining the tolerance value for smoothing the contour, the operator must have knowledge of the subsequent CAM program.

## Section 3

### Mould & Die commands

#### List of G Codes

The following G Codes are relevant for Mold & Die

#### G Code Group4 : FIFO

STARTFIFO / STOPFIFO / FIFOCTRL

#### Recommended function : FIFOCTRL

Preprocessing memory control - Avoids that the FIFO buffer runs empty

#### G Code Group10 : EXACT STOP - CONTINUOUS PATH MODE

##### LOOK AHEAD

G60 / G64 / G641 / G642

#### Recommended function : G642

##### Use of LOOK AHEAD:

The purpose of continuous-path control is to increase speed and harmonize traversing performance. In path control function this is achieved with two functions.

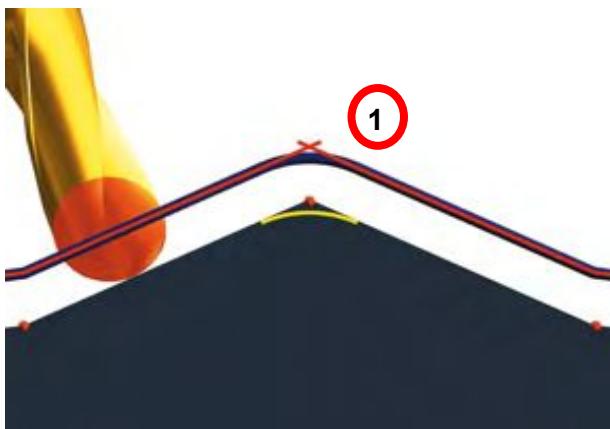
##### Look ahead - Look ahead speed control :

The control system calculates several NC blocks ahead and determines a modal speed profile. The way in which this speed control is calculated can be set with functions G64 or G642.

#### Recommended function : Corner rounding G642

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

Insertion of Circle or Spline elements.



1

Corner Rounding G642

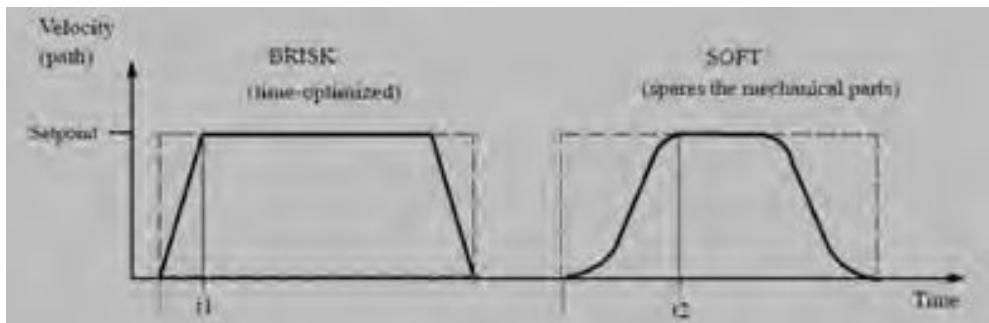
Notes

**G Code Group21 : ACCELERATION PROFILE**  
BRISK / SOFT / DRIVE**Recommended function : SOFT**

Soft smoothed path acceleration

To make acceleration as gentle on the machine as possible, the acceleration profile of the axes can be influenced by means of the commands **Soft** and **Brisk**. If **Soft** is activated, the acceleration behavior does not change abruptly but is increased by a linear characteristic. This reduces the load on the machine. It also has a beneficial effect on the surface quality of workpieces, since machine resonance is excited far less frequently.

The axis slides travel with constant acceleration until the feedrate is reached. SOFT acceleration enables higher path accuracy and less wear and tear on the machine.



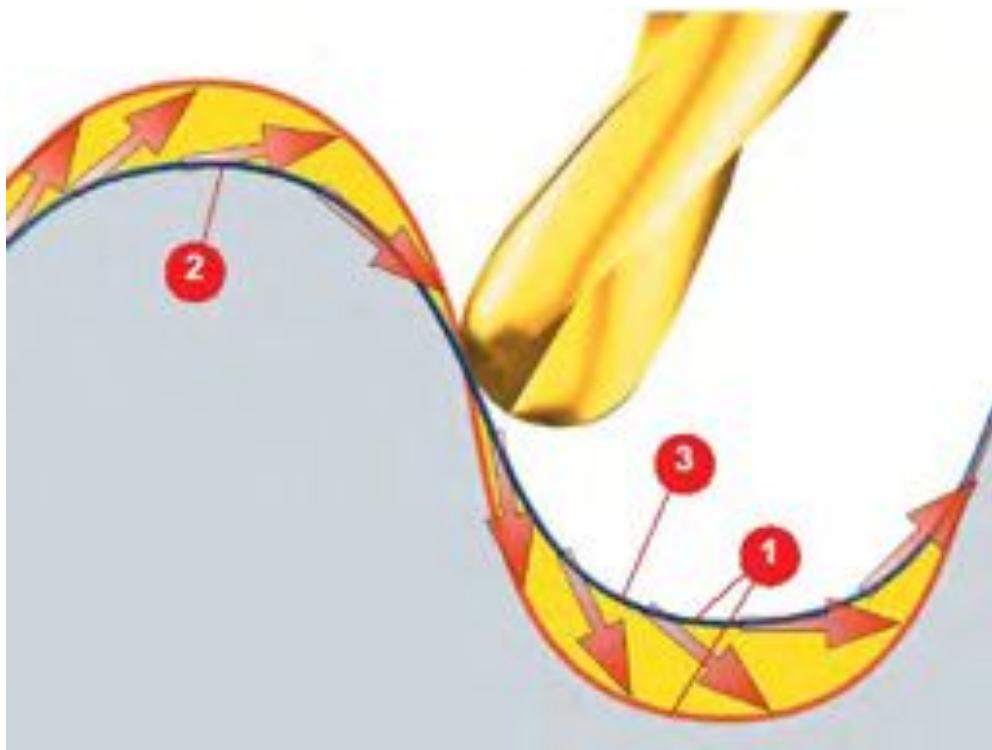
**G Code Group24 : FEEDFORWARD CONTROL**  
FFWOF / FFWON**Recommended function : FFWON**

Feed forward control ON

**Feedforward control function FFWON**

Following errors cause contour violation. **1** The inertia in the system means that the cutter tends to leave the setpoint contour - **2** - tangentially, i.e. the actual contour - **3** - that is produced deviates from the setpoint contour. Following errors are due to a combination of the system -- positioning control - and the speed.

Feedforward control **FFWON** reduces speed - dependent following errors when contouring almost to zero. Traversing with feedforward control permits higher path accuracy and thus improved machining results.



**G Code Group30 — COMPRESSOR ON/OFF**

COMPOF / COMPON / COMPCURV / COMPCAD

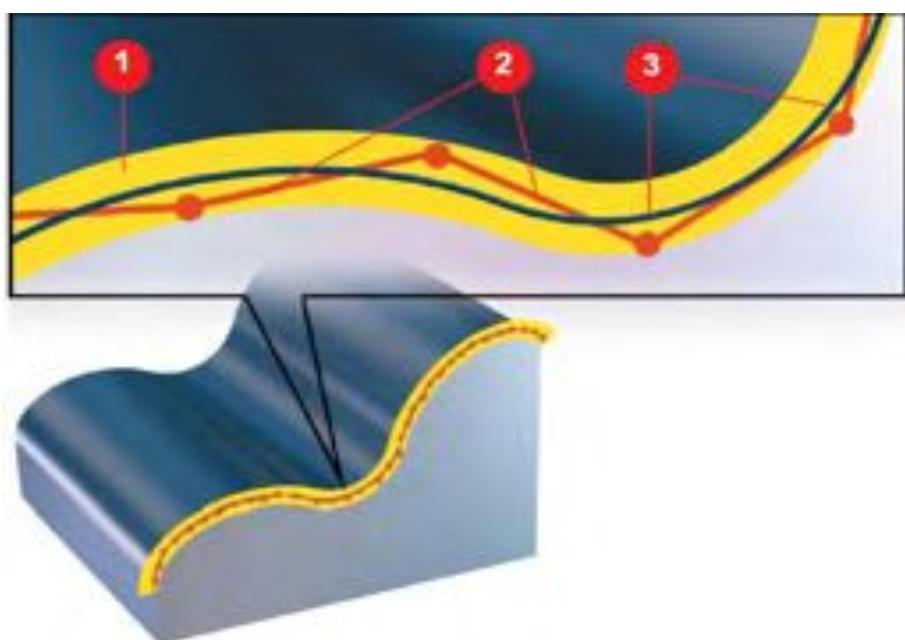
This function is only available for SINUMERIK 802D sl pro.

**Recommended function : COMPCAD**

CAD/CAM systems usually deliver linear blocks which observe the parameterized accuracy. With complex contours, this results in a substantial data quantity and – in some cases short path sections. These short path sections limit the machining speed. By using the compressor, it is possible to summarize short path blocks in a path section. The number of blocks to be executed is compressed. By using the COMPCAD Gcode, you can select a compression optimized with reference to the surface quality and the velocity, whereby the accuracy of the interpolation can be defined via machine data \$MA\_COMPRESS\_POS\_TOL - MD33100 .

According to the specified tolerance band - **1** - the compressor combines a sequence of G1 commands - **2** - and compresses them into a spline - **3** - which is directly calculated by the control system. This makes the surface much smoother, since the machine axes can move more harmoniously and machine resonance is avoided. This allows higher traversing speeds and reduces the load on the machine.

This compression operation can only be executed on linear blocks (G1). It is interrupted by any other type of NC instruction, e.g., an auxiliary function output, but not by parameter calculations. Only those blocks containing nothing more than the block number, G1, axis addresses, feed and comments are compressed. All other blocks are executed unchanged (no compression).



## Section 3

### Mould & Die commands

#### Programming Commands

Notes

**Programming functions :**  
Tolerance of the axes used for machining.

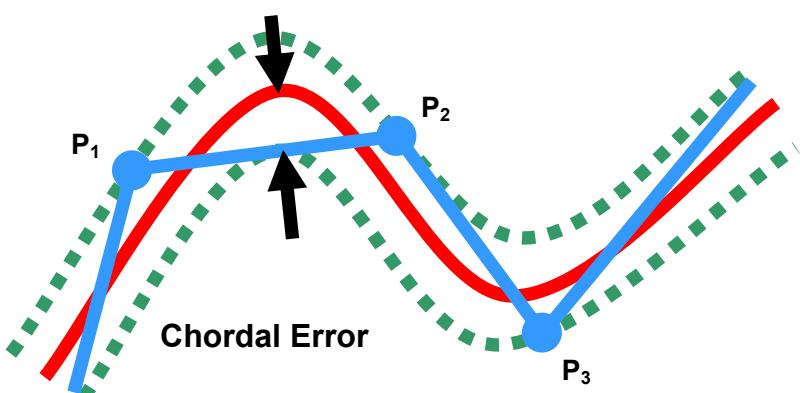
\$MA\_COMPRESS\_POS\_TOL[ AXIS]= Tolerance Value

AXIS MD 33100

2

The tolerance value applies with G642 and COMPCAD. The tolerance value is written to the relevant machine data. This tolerance value should be the **same** or a **a little bit higher** 10-20%, than the chord tolerance band of the CAM system.

#### Freeform Surface:



— Ideal Cutter Path

●—● Linearized Cutter Path

■ ■ ■ Tolerance Band

2

To change this machine data the Manufacturing password has to be set

## Section 4

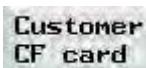
### Transfer, edit and execute program

#### Transfer and copy a program

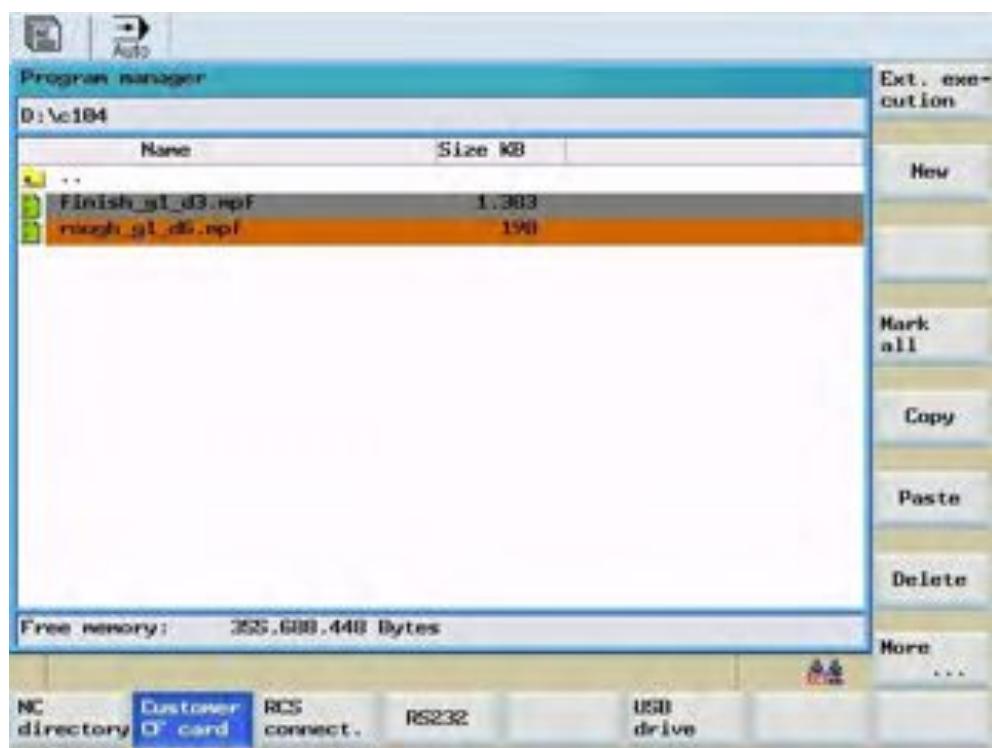
Notes

With **802Dsl pro** it is possible to copy a program <= 3MB directly into the controller - program manager.

Follow the sequence below.



Highlight the files you wish to transfer to the NC



Copy

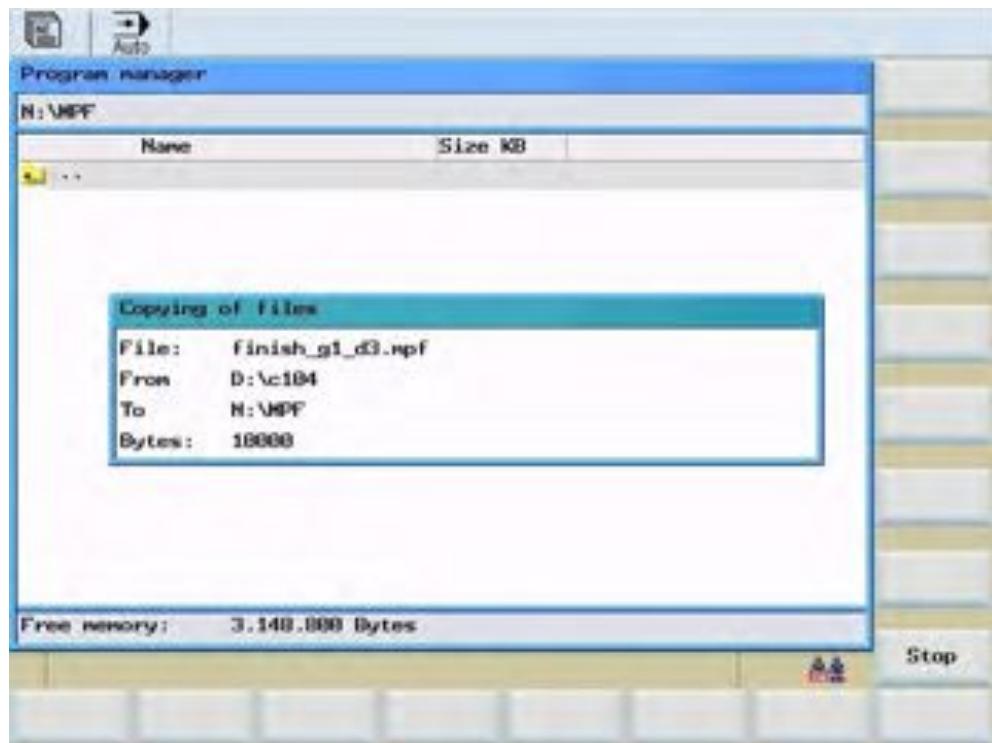
## Section 4

### Transfer, edit and execute program

Notes

NC  
directory

Paste



Note: The NC Memory for the operator on the **802Dsl pro** is max 3MB.  
Programs larger than 3MB could only be stored and executed from the CF Card.

## Section 4

### Transfer, edit and execute program

#### Edit a program

With **802Dsl pro** it is possible to edit every program <= 3MB directly on the controller in the menu program manager. Programs > 3MB could only be edited and changed on an extern medium i.e. PC.

Use the following sequence to open and edit programs in the NC directory.



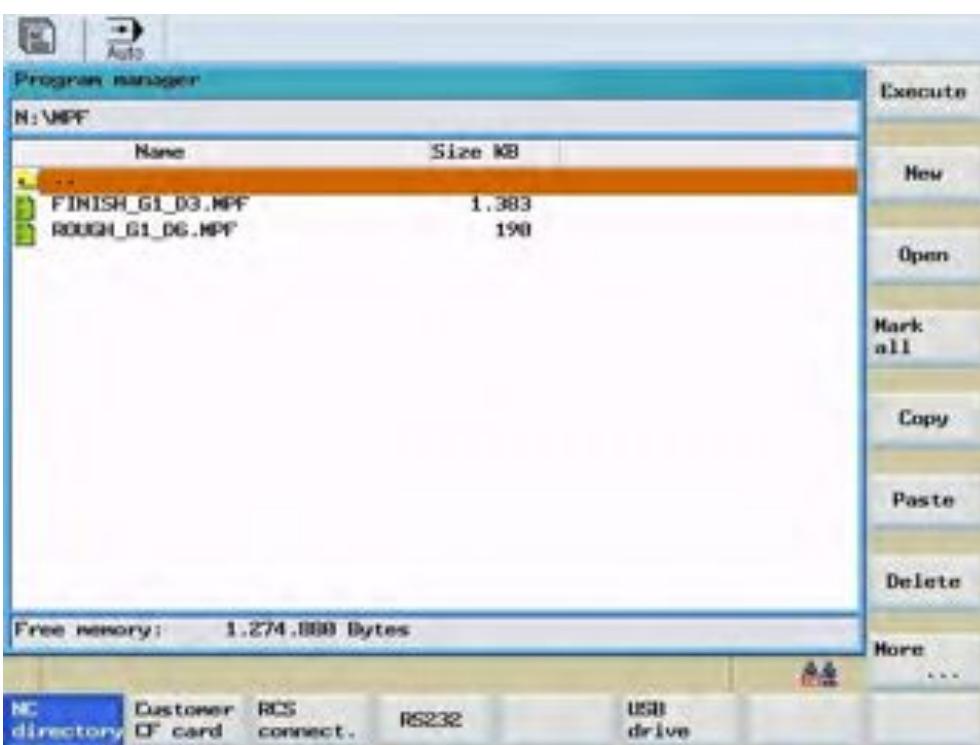
Highlight the file using the following buttons.



The directory and, or file can be opened with the following keys.



or



Notes

## Section 4

### Transfer, edit and execute program

Notes

The screenshot shows the SINUMERIK 802D SI Program Editor interface. On the left, a list of G-code commands is displayed:

```
N:\MPF\ROUGH_G1_DG.MPF
T1 T1 D1 1
N2 M6 1
H3 G24 1
H4 S+4500 H3 1
H5 MSGC " Roughing end mill no arcs, Chord 0.05, Face cycle " 3 5
H6 MSGC " End Mill 6mm " 3 5
H7 PIPCTRL 1
H8 FFWDH 1
H9 SOFT 1
H10 COMPARD 1
H11 G84Z 1
H12 $NR_COMPRESS_POS_TOL(X)= 0.05 1
H13 $NR_COMPRESS_POS_TOL(Y)= 0.05 1
H14 $NR_COMPRESS_POS_TOL(Z)= 0.05 1
H15 G0 X50.899 Y=57.933 1
H16 G0 X50.899 Y=57.933 Z10.15 1
H17 G0 Z5.15 1
H18 G1 Z0.15 F2500 1
H19 G1 X49.986 Y=51. 1
H20 G1 X50.014 1
```

At the bottom of the editor window, there are tabs for "Edit" and "Simu-lation". A context menu is open on the right side of the screen, listing options: Execute, Mark block, Copy block, and Find.

## Section 4

### Transfer, edit and execute program

#### Execute a program

With **802Dsl** pro it is possible to execute a program in two ways . Execute from the controller NC Memory and execute from CF Card

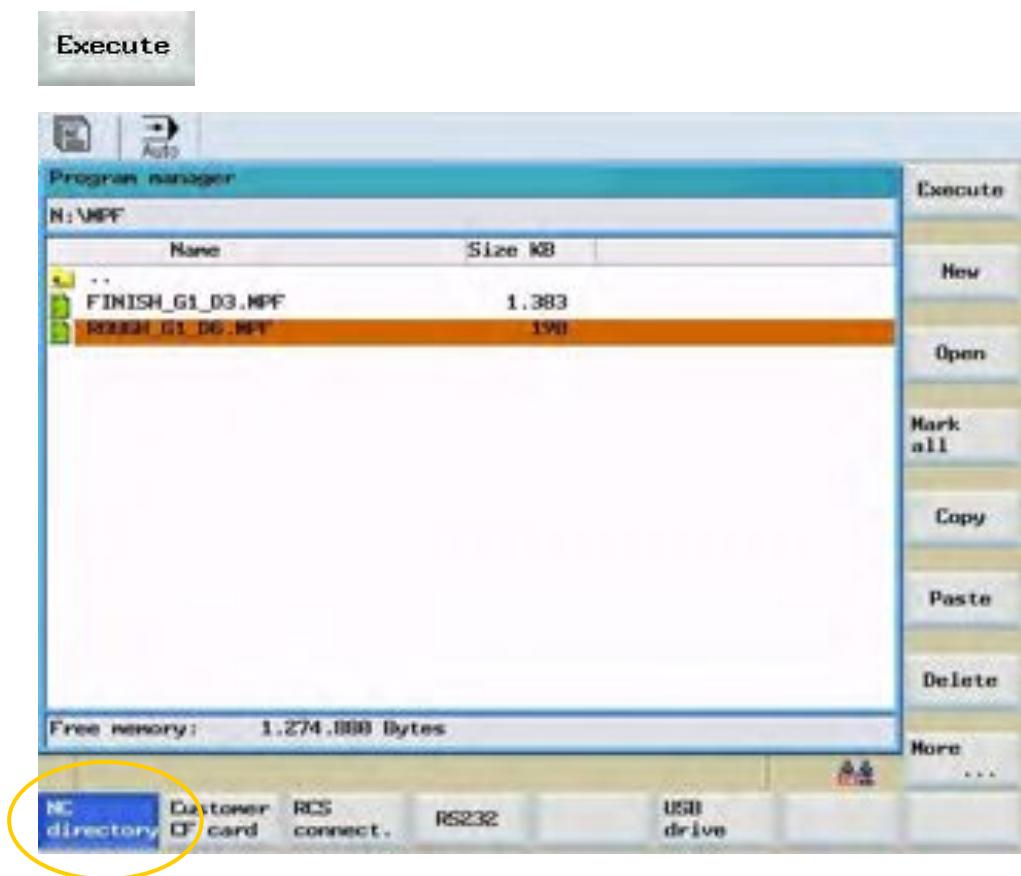
Use the following sequence to open and execute programs in the NC directory.



Highlight the file using the following buttons.



The directory and, or file can be executed with the following key.



When in the NC DIRECTORY the program is executed directly from the NC Memory.

Notes

## Section 4

### Transfer, edit and execute program

Notes

PROGRAM  
MANAGER

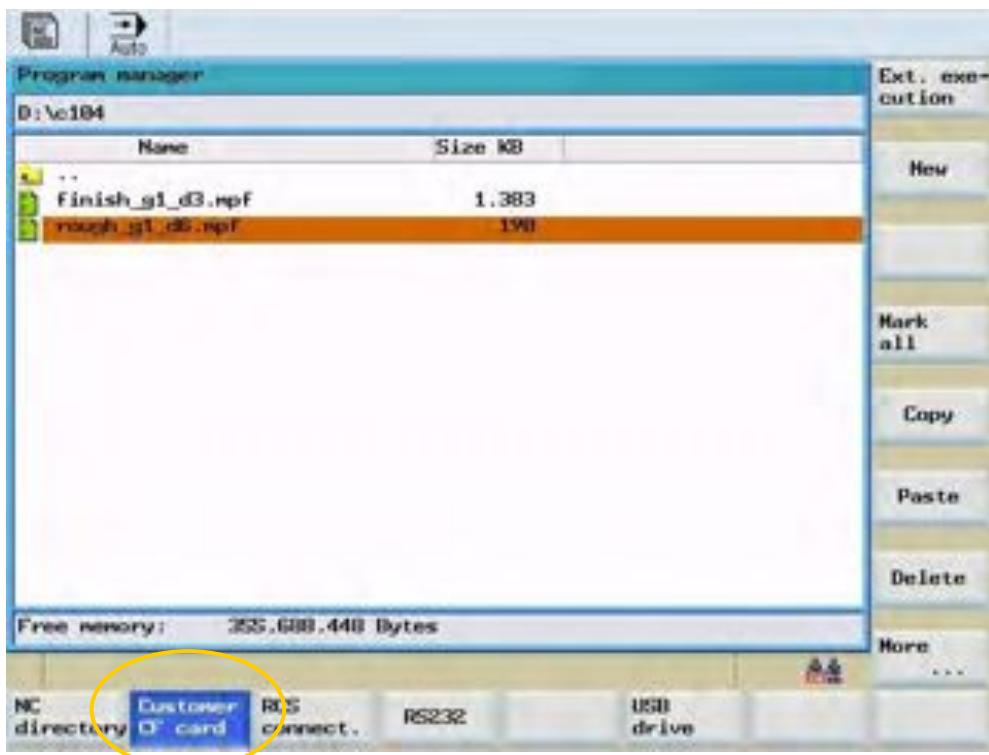
Customer  
CF card

Highlight the file using the following buttons.



The directory and, or file can be executed with the following key.

Ext. exe-  
cution



When in the CUSTOMER CF CARD the program is executed from the CF Memory card.

## Section 5

### Program Example

#### Program Roughing

##### Short Description:

Tool : End Mill 6mm  
CAM Chordal Tolerance : 0.05 mm  
Program Size : 186 KB - 7778 lines



Notes

#### Program Header:

Program editor:  
N:\MPF\ROUGH\_G1\_D6.MPF      1      Not selected

```
N1 T1 D1 1
N2 M6 1
N3 G24 1
N4 S+4500 N3 1
N5 MSG1 "Roughing end mill no arcs, Chord 0.05, Face cycle = 1 1"
N6 MSGC "End Mill 6mm " 1 1
N7 P100T1 1
N8 FFMOH 1
N9 SOFT 1
N10 COMPCAD 1
N11 G642 1
N12 SHA_COMPRESS_POS_TOL(X)= 0.05 1
N13 SHA_COMPRESS_POS_TOL(Y)= 0.05 1
N14 SHA_COMPRESS_POS_TOL(Z)= 0.05 1
N15 G0 X50.899 Y=57.933 1
N16 G0 X50.899 Y=57.933 Z10.15 1
N17 G0 Z5.15 1
N18 G1 Z9.15 F25000 1
N19 G1 X49.986 Y=51. 1
N20 G1 X50.014 1
```

Edit      Simulation

End\_MILL.Ges      SIEMENS G Function

MCS	position	Dist-to-go	G functions
-X	50.396	-0.305 nm	21:SOFT 22:CUT2D 23:CDOF 24:FFMOH
*Y	-54.057	2.353 nm	25:ORINAS 26:RME 27:ORIC 28:VALTHON
-Z	0.150	0.000 nm	29:DIAMOF 30:COMPCAD 31:G810 32:G820 33:FTOCDF 34:OSOF 35:SPOF 36:POCLAVO 37:PHORM 38:SP1F1 39:CPRECDF 40:CUTCONO
A	0.000	0.000 nm	
G01	054	054	054

Block display      Current program :ROUGH\_G1\_D6.MPF  
N16 G0 X50.899 Y=57.933 Z10.15  
N17 G0 Z5.15  
N18 G1 Z9.15 F25000  
N19 G1 X49.986 Y=51.  
N20 G1 X50.014  
N21 G1 X51. . Y=47.877  
N22 G1 X=51.4

Cycle time: 00000 000 015

Program control      Block search      Correct program

## Section 5

### Program Example

#### Program Finishing

##### Short Description:

Tool : Ball Nose 3mm

CAM Chordal Tolerance : 0.005 mm

Program Size : 1,35 MB - 50338 lines



Notes

#### Program Header:

```
N:\MPFF\FINISH_G1_03.MPF
H1 T2 01 1
H2 M6 1
H3 G54 1
H4 S+4500 M3 1
H5 MSGC "Finishing chord 0.005 " 3 1
H6 MSGC "Ball Mill 03" 3 1
H7 F1000R1 1
H8 FFMON 1
H9 SOFT 1
H10 COMPCRD 1
H11 G642 1
H12 SNA_COMPRESS_POS_TOL(X)= 0.005 1
H13 SNA_COMPRESS_POS_TOL(Y)= 0.005 1
H14 SNA_COMPRESS_POS_TOL(Z)= 0.005 1
H15 HEMCOHP 1
H16 G0 X26.499 Y-12.096 1
H17 G0 X26.499 Y-12.096 Z10. 1
H18 G0 Z-11. 1
H19 G1 Z-16. F500 1
H20 G1 Y-12.079 Z-15.666 F2000 1
```

Edit Contour Drilling Milling Simulation Re-compile

MCS	Position	Dist-to-go	G functions
X	26.499	0.000 mm	21:SOFT 22:CUT2D 23:CDOF 24:FFMON 25:DRINKS 26:INCR 27:ORIG 28:MM_XMON 29:DIAMONF 30:COMPCRD 31:G010 32:GOZO 33:PTOCDF 34:OSDF 35:SPOF 36:POELMDO 37:PHORM 38:SP1P1 39:CPRECDF 40:CUTCONO
Y	-12.096	0.000 mm	
Z	396.887	-396.887 mm	
A	0.000	0.000 mm	
G98	G54	0642	

Block display Current program: FINISH\_G1\_03.MPF

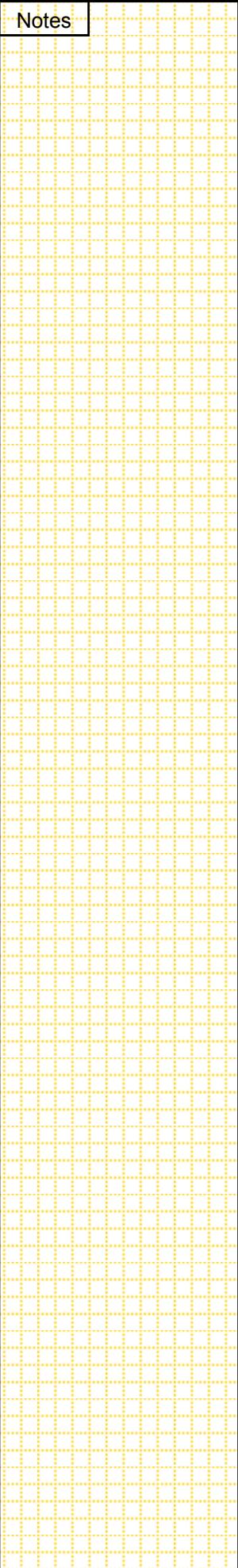
N16 G0 X26.499 Y-12.096 1  
N17 G0 X26.499 Y-12.096 Z10. 1  
N18 G0 Z-11. 1  
N19 G1 Z-16. F500 1  
N20 G1 Y-12.079 Z-15.666 F2000 1  
N21 G1 Y-12.043 Z-15.251 1  
N22 G1 Y-11.993 Z-14.838 1

Cycle time: 00000 000 015

Program control Block search Correct program

---

Notes



## 1 Brief description

### Module objective:

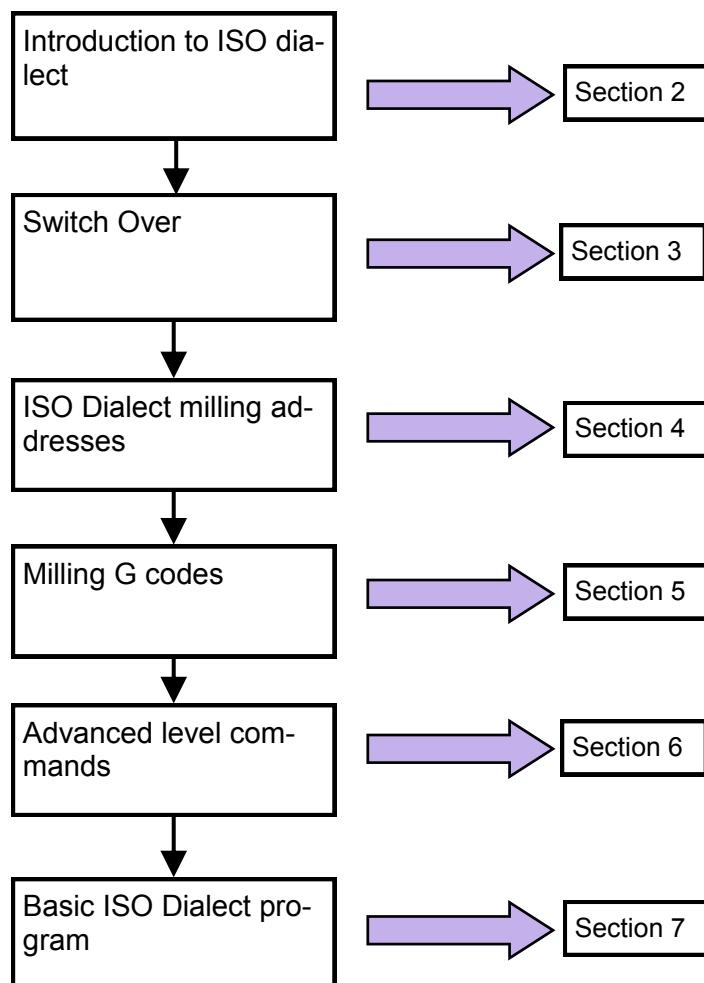
Upon completion of this module you will see that it is possible with 802D sl, ISO Dialect programming for milling.

### Module description:

ISO Dialect allows a non-Siemens NC program to be run within the control. This module describes the functionality offered by standard functions. Differences and additions implemented by the machine tool manufacturer are documented by the machine-tool manufacturer.

### Module content:

- Introduction to ISO dialect
- Switch Over
- ISO Dialect milling addresses
- Milling G codes
- Advanced level commands
- Basic ISO Dialect program



## Section 2

### Introduction to ISO Dialect

#### Siemens Mode

The following conditions apply when Siemens mode is active:

Siemens G functions are interpreted on the control by default.

It is not possible to extend the Siemens programming system with ISO Dialect functions because some of the G functions have different meanings.

#### ISO Dialect mode

The following conditions apply when ISO Dialect mode is active:

Only ISO dialect G codes can be programmed, not Siemens G functions.

It is not possible to use a mixture of ISO Dialect code and Siemens code in the same NC block.

If further Siemens functions are to be used, it will be necessary to switch to Siemens mode first.

Notes

## Section 3

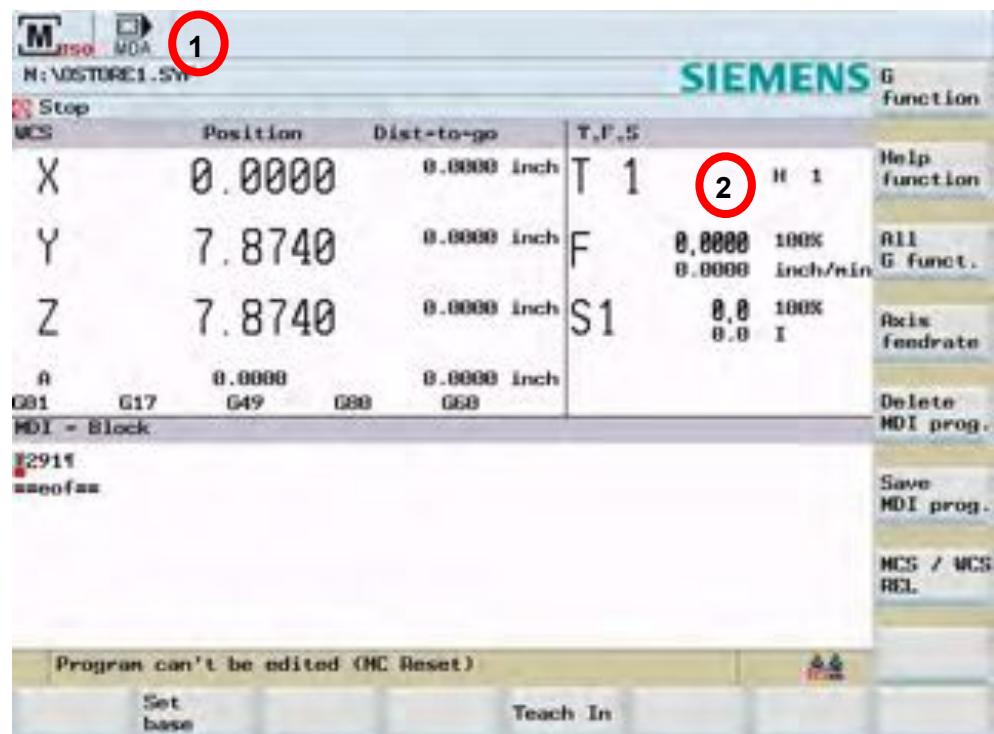
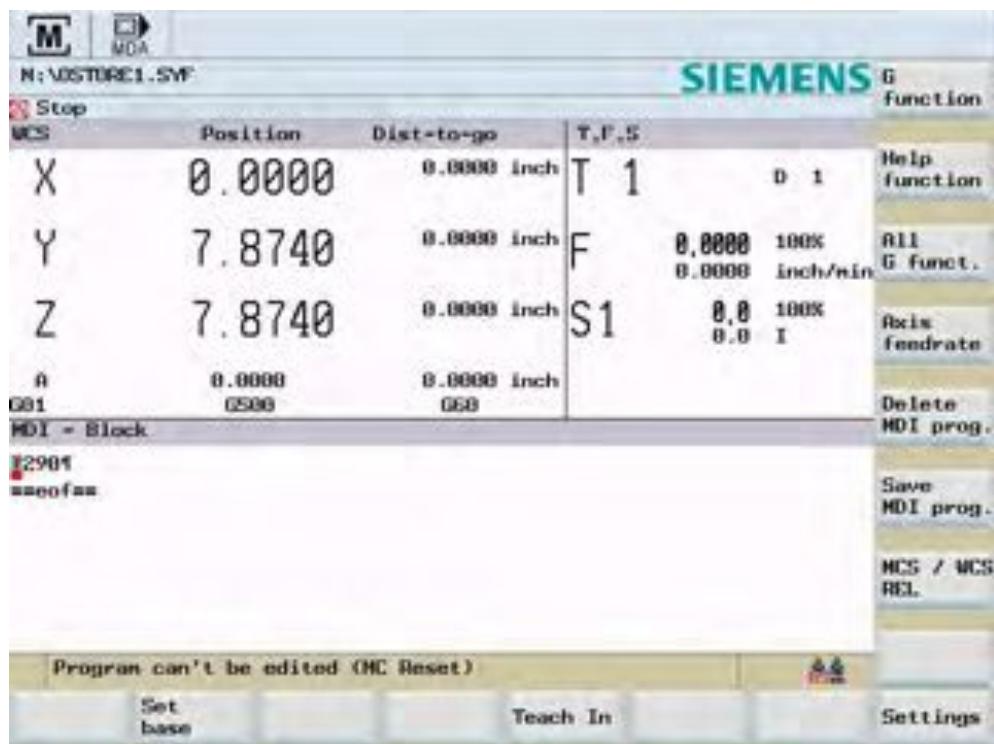
### Switch Over to ISO Dialect

Notes

The following two G functions are used to switch between Siemens Mode and ISO dialect Mode:

G290 - Siemens NC programming language active

G291 - ISO Dialect NC programming language active



1 In ISO dialect, you will see G291 in top left hand corner

2 In ISO dialect, you will see G291 H value in the T,F,S window.

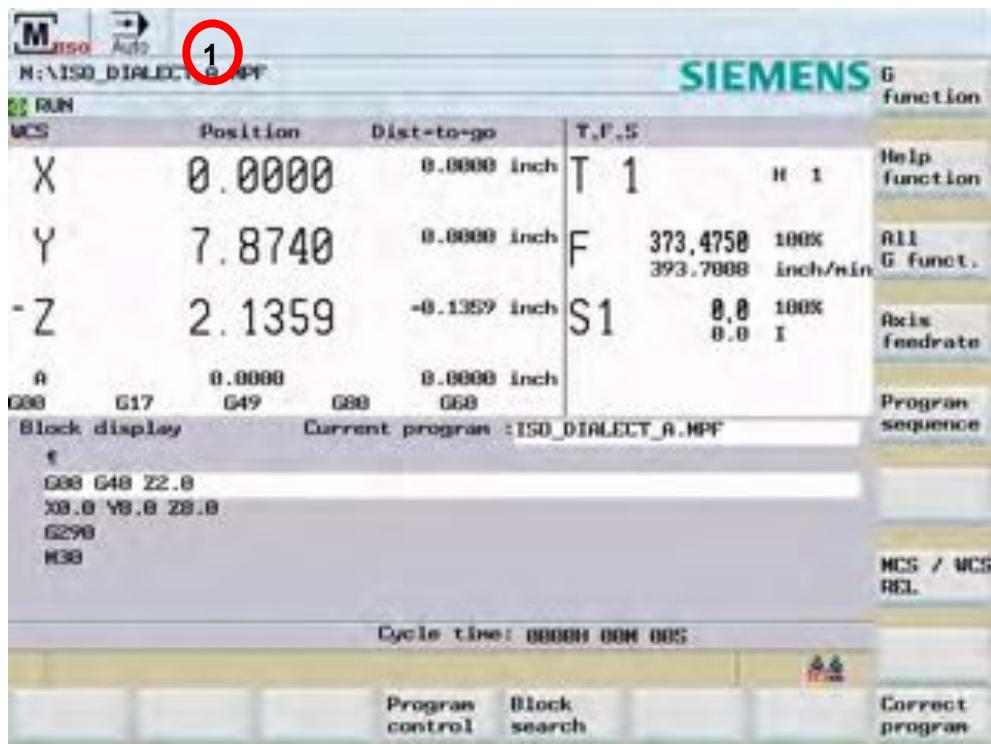
## Section 3

### Switch Over to ISO Dialect

Notes

When writing the NC program the two G functions are placed at the top and bottom of the program

```
G291 ;ISO dialect Mode  
G00 G90 G94 G40  
G21 G17  
T1  
M6  
...  
...  
...  
...  
G00 G40 Z2.0  
X0.0 Y8.0 Z8.0  
G290 ;Siemens mode  
M30
```



- 1 In ISO dialect, you will see G291 in top left hand corner

Note: if you press "RESET" button

halfway through running an NC program, the control will revert back to Siemens mode, as G290 is the default code.

## Section 4

### ISO Dialect milling addresses

These are the different addresses that are used in ISO dialect.

Notes

Address	Meaning
F	Feed G94 (mm/inch per min)
F	Feed G95 (mm/inch per rev)
F	Thread pitch
C	Chamfer
R	Radius
Q	
I, J, K	Interpolation parameters
X	G4 Time unit
A	Contour Angle

## Section 5

### Milling G codes

Notes

Here is a list of ISO dialect G codes that are fundamentally different to G functions in Siemens mode.

#### Circular interpolation, G02/G03

##### Function

This function allows you to program an arc either clockwise (G02) or counter-clockwise (G03) direction.

##### Programming

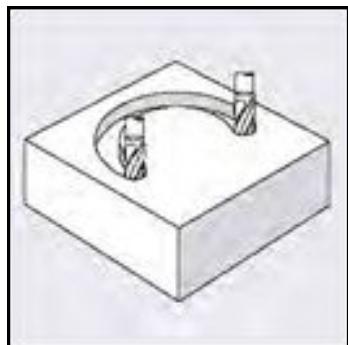
G02/G03 G90 X.. Y.. Z.. I.. J.. K.. F.. Absolute end point

Or

G02/G03 G91 X.. Y.. Z.. I.. J.. K.. F.. Incremental end point

Or

G02/G03 X.. Y.. Z.. R.. F.. Radius of arc



#### Dwell time G4

##### Function

You can use G4 to interrupt workpiece machining between two NC blocks for the programmed length of time, e.g. dwell at bottom of hole

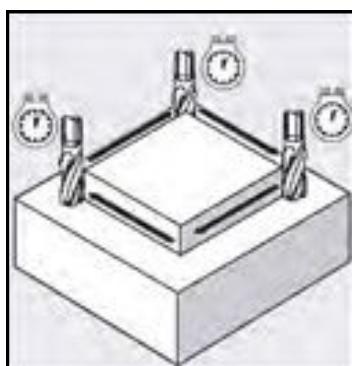
In the feed per minute mode (G94) the dwell time unit is in seconds, while in the feed per revolution (G95) the dwell time unit is in spindle revolutions.

##### Programming

G4 G94 X..      X = Time

Or

G4 G95 X..      X = Rotations



## Section 5

### Milling G codes

Notes

#### Tool length offset (G43, G44, G49)

##### Function

The tool length offset function adds or subtracts the amount stored in the tool offset data memory to or from the Z coordinate values specified in the program to offset the programmed paths according to the length of a cutting tool.

##### Commands

In the execution of the tool length offset function, addition or subtraction of the offset data is determined by the specified G code and the direction of offset by the H code.

##### Programming

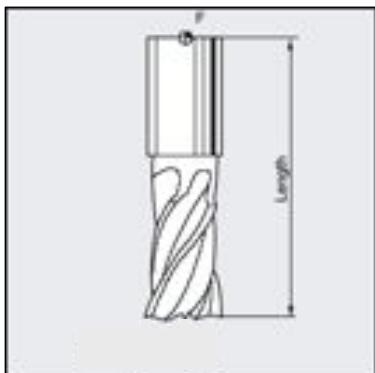
G01 G43 Z... H... ;tool offset is added to Z axis position

Or

G01 G44 Z... H... ;tool offset is subtracted from Z axis position

Or

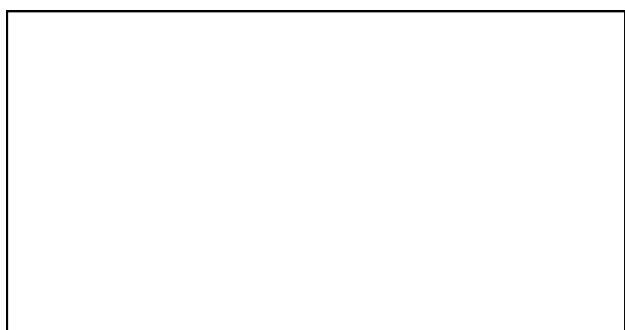
G49 ;cancels the tool offset mode



#### Return value (G98, G99)

##### Function

When using canned cycles, the retraction level for the Z axis is determined through G98/G99. G98/G99 are modal G codes. G98 is usually set as power-on default.



##### Programming

G85 G98 Z... R... F...

Or

G85 G99 Z... R... F...

## Section 6

### Advanced level commands

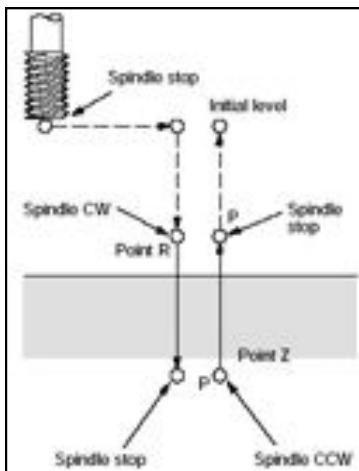
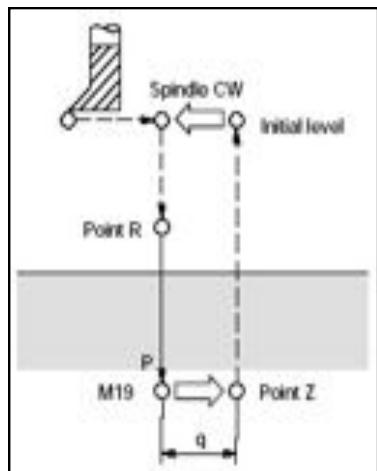
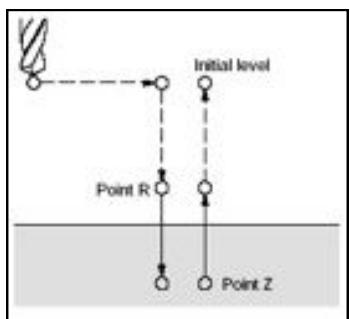
Notes

#### Canned cycles (G73 to G89)

##### Function

By using canned cycles, it is easier for the programmer to create programs. By means of canned cycles, machining operations frequently used can be determined in a single block through a G function. Normally more than one block is required when programming without canned cycles. Using canned cycles can also shorten the program in order to save memory.

G Code	Description
G73	Deep hole drilling with chip break
G74	Counterclockwise tapping cycle
G76	Fine drilling cycle
G80	Cycle off
G81	Counterbore drilling cycle
G82	Countersink drilling cycle
G83	Deep hole drilling cycle with swarf removal
G84	Clockwise tapping cycle
G85	Drilling cycle
G86	Drilling cycle, retract using G00
G87	Back boring cycle
G89	Drilling cycle, retract using G01



## Section 6

### Advanced level commands

Notes

#### Subprogram call up function (M98, M99)

##### Function

This function can be used when subprograms are stored in the part program memory, subprograms registered to the memory with program numbers assigned can be called up and executed as many times as required.

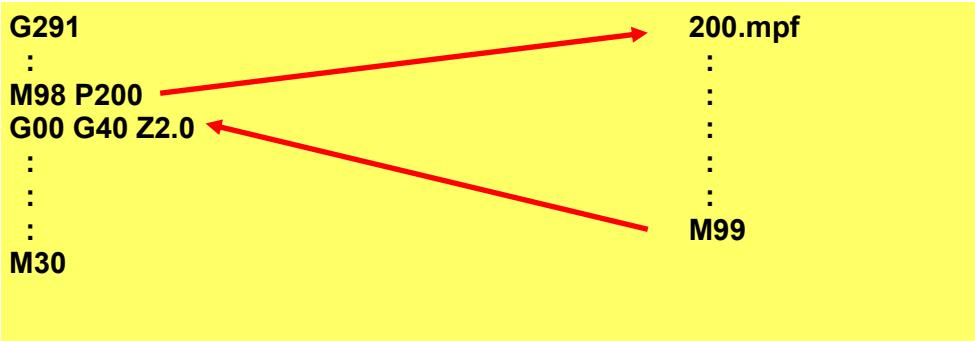
The created subprograms should be stored in the part program memory before they are called up.

##### Commands

M98	;subprogram call up
Pxxxx	;program number
Lyyyy	;number of program runs
M99	;end of subprogram

##### Programming

M98 Pyyyyxxxxx ;  
Or  
M98 Pxxxx Lyyyy ;



#### Chamfering and corner rounding commands (R, C)

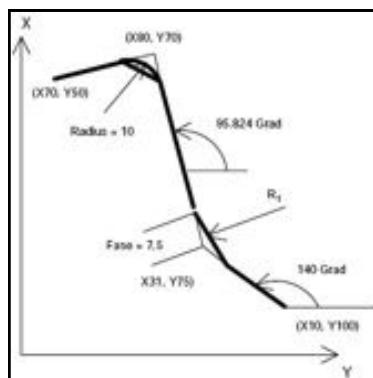
##### Function

It is possible to insert chamfering and corner rounding blocks automatically between:

- linear interpolation and linear interpolation
- linear interpolation and circular interpolation
- circular interpolation and linear interpolation
- circular interpolation and Circular interpolation

##### Programming

C... ;chamfering  
R... ;corner rounding

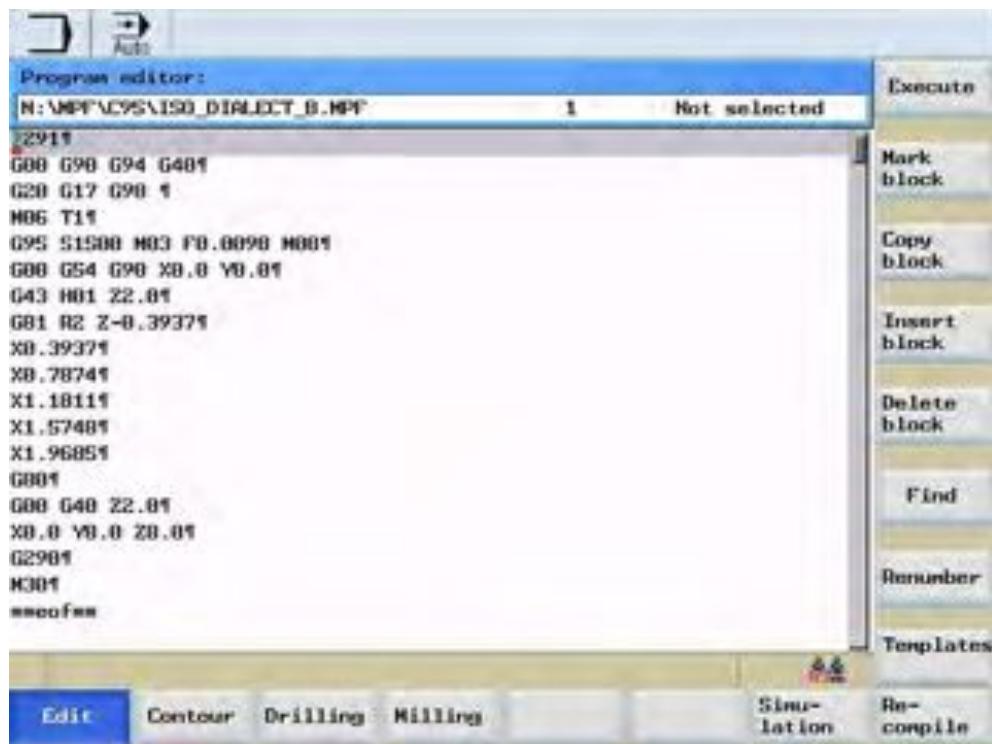


## Section 7

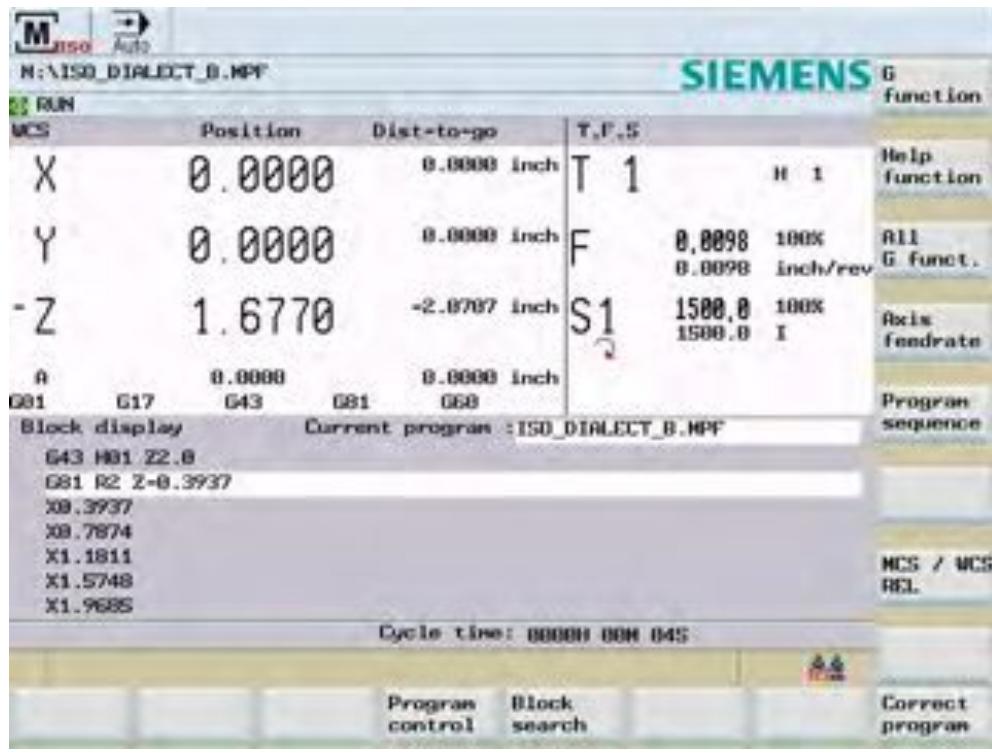
### Basic ISO Dialect program

Notes

An ISO Dialect program using the same basic structure as a Siemens program, but using instead an ISO Dialect drilling cycle (G81).



The above program running in auto mode  
note the ISO in red text, below.



## 1 Brief description

### Module objective:

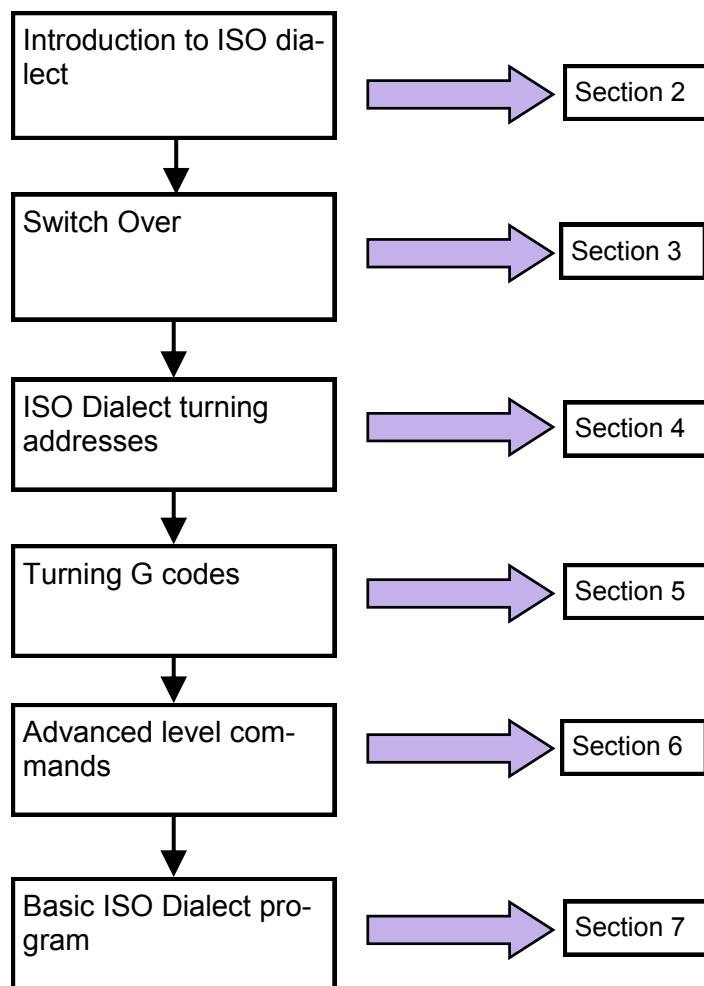
Upon completion of this module you will see that it is possible with 802D sl, ISO Dialect programming for turning - \*A, B, C G code system).

### Module description:

ISO Dialect allows a non - Siemens NC program to be run within the control. This module describes the functionality offered by standard functions. Differences and additions implemented by the machine tool manufacturer are documented by the machine-tool manufacturer.

### Module Content:

- Introduction to ISO dialect
- Switch Over
- ISO Dialect turning addresses
- Turning G codes
- Advanced level commands
- Basic ISO Dialect program



## Section 2

### Introduction to ISO Dialect

Notes

#### Siemens Mode

The following conditions apply when Siemens mode is active:

Siemens G functions are interpreted on the control by default.

It is not possible to extend the Siemens programming system with ISO Dialect functions because some of the G functions have different meanings.

#### ISO Dialect mode

The following conditions apply when ISO Dialect mode is active:

Only ISO dialect G codes can be programmed, not Siemens G functions.

It is not possible to use a mixture of ISO Dialect code and Siemens code in the same NC block.

If further Siemens functions are to be used, it will be necessary to switch to Siemens mode first.

#### Selection of G code system A, B, C

ISO dialect T distinguishes between G code A, B, C. G code system B is the default setting

#### G code system A

If G code system A is active, G91 is not available. In this case, incremental axes movement for axis X and Z is programmed by address U and W. U and W are not available as axis designation in this case resulting in an axes number of 6

Address H is used for programming incremental movement of the C axis in G code system A.

#### Canned cycles of G code A, B, C

All three G code groups have a set of canned cycles but the G code names given to the canned cycles are different to the G code A, B, C.

#### Notice

If not otherwise noted, the manual in hand describes G code system B.

## Section 3

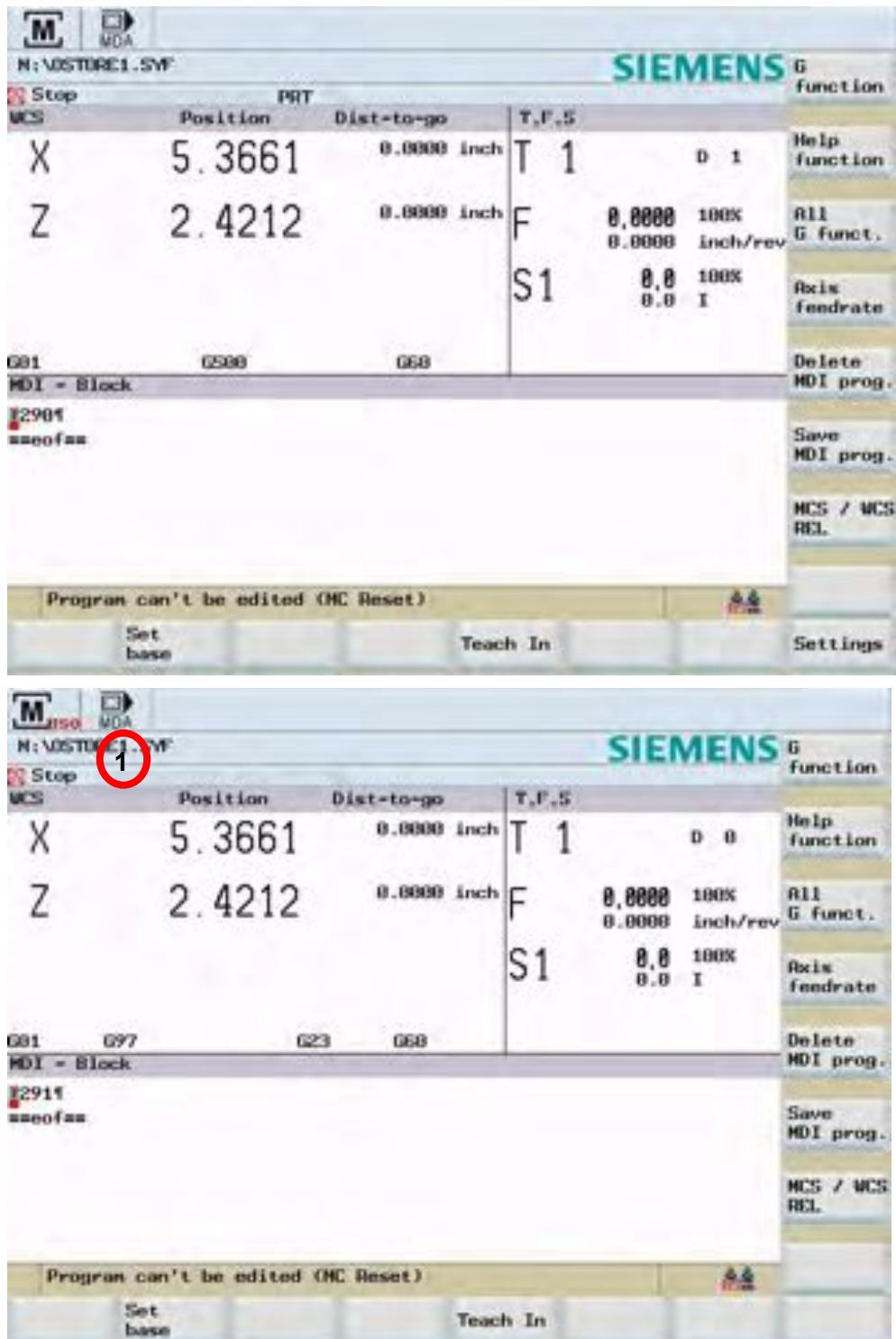
### Switch Over to ISO Dialect

Notes

The following two G functions are used to switch between Siemens Mode and ISO dialect Mode:

G290 - Siemens NC programming language active

G291 - ISO Dialect NC programming language active



- 1 In ISO dialect, you will see ISO in top left hand corner

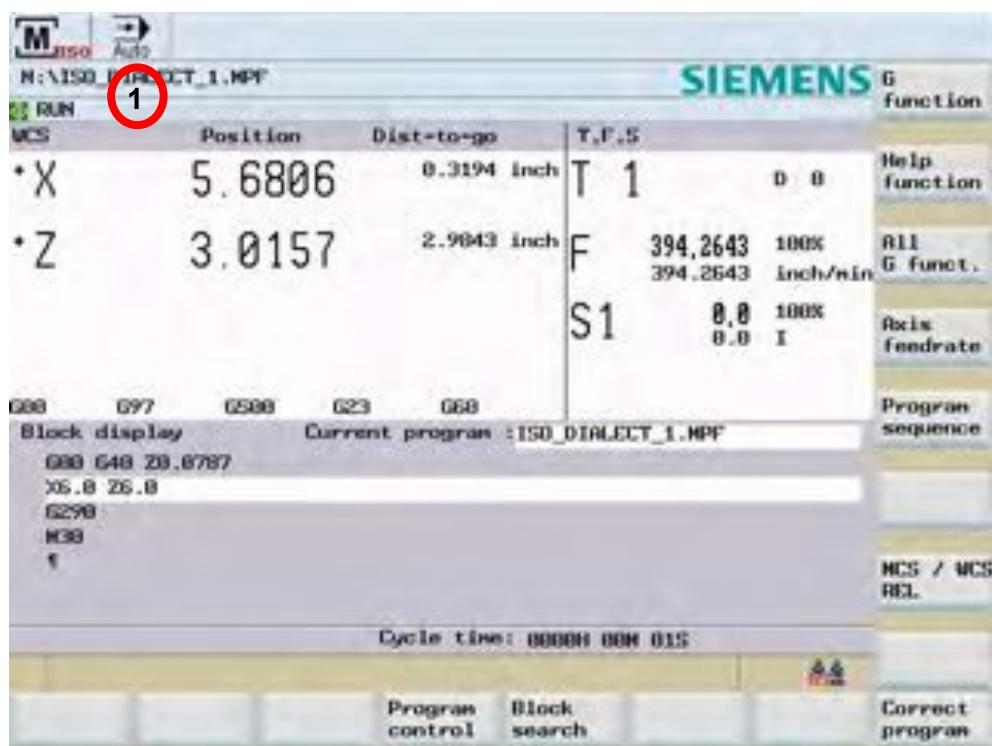
## Section 3

### Switch Over to ISO Dialect

Notes

When writing the NC program the two G functions are placed at the top and bottom of the program

```
G291 ;ISO dialect Mode
G00 G90 G95 G40
G20 G18
G92 S2500
T1
...
...
...
...
G00 G40 Z0.0787
X6.0 Z6.0
G290 ;Siemens mode
M30
```



- 1 In ISO dialect, you will see ISO in top left hand corner

Note: if you press "RESET" button

halfway through running an NC program, the control will revert back to Siemens mode, as G290 is the default code.

## Section 4

### ISO Dialect turning addresses

These are the different addresses that are used in ISO dialect.

Notes

Address	Meaning
F	Feed G94 (mm/inch per min)
F	Feed G95 (mm/inch per rev)
F	Thread pitch
C	Chamfer
R	Radius
Q	
I, K	Interpolation parameters
X	G4 Time unit
A	Contour Angle

## Section 5

### Turning G codes

Notes

Here is a list of ISO dialect G codes that are fundamentally different to G functions in Siemens mode.

#### Circular interpolation, G02/G03

##### Function

This function allows you to program an arc either in clockwise (G02) or counter-clockwise (G03) direction.

##### Programming

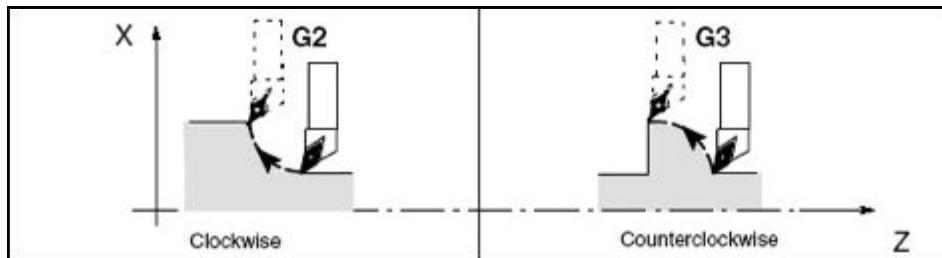
G02/G03 G90 X.. Z.. I.. J.. K.. F.. Absolute end point

Or

G02/G03 G91 X.. Z.. I.. J.. K.. F.. Incremental end point

Or

G02/G03 X.. Z.. R.. F.. Radius of arc



#### Dwell time G4

##### Function

You can use G4 to interrupt workpiece machining between two NC blocks for the programmed length of time, e.g. dwell at bottom of hole

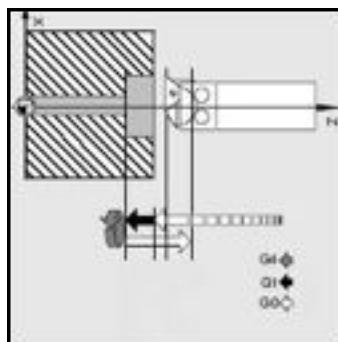
In the feed per minute mode (G94) the dwell time unit is in seconds, while in the feed per revolution (G95) the dwell time unit is in spindle revolutions.

##### Programming

G4 G94 X..      X = Time

Or

G4 G95 X..      X = Rotations



## Section 5

### Turning G codes

Notes

#### Tool length offset (G43, G44, G49)

##### Function

The tool length offset function adds or subtracts the amount stored in the tool offset data memory to or from the Z coordinate values specified in the program to offset the programmed paths according to the length of a cutting tool.

##### Commands

In the execution of the tool length offset function, addition or subtraction of the offset data is determined by the specified G code and the direction of offset by the H code.

##### Programming

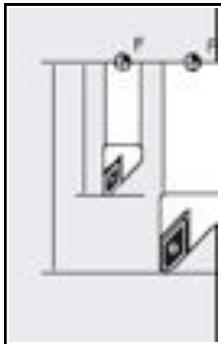
G01 G43 Z... H... ;tool offset is added to Z axis position

Or

G01 G44 Z... H... ;tool offset is subtracted from Z axis position

Or

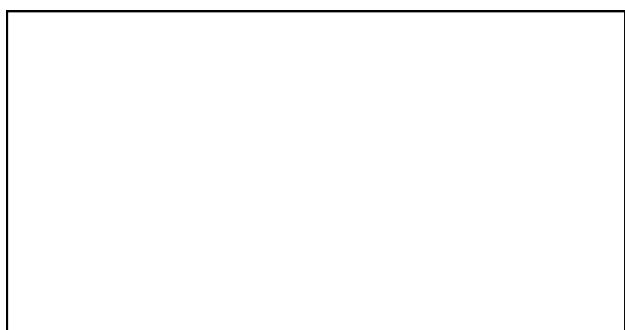
G49 ;cancels the tool offset mode



#### Return value (G98, G99)

##### Function

When using canned cycles, the retraction level for the Z axis is determined through G98/G99. G98/G99 are modal G codes. G98 is usually set as power-on default.



##### Programming

G85 G98 Z... R... F...

Or

G85 G99 Z... R... F...

## Section 5

### Turning G codes

Notes

#### Constant cutting rate G96 G97

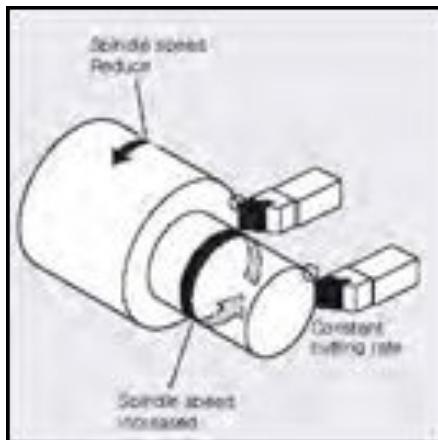
##### Function

The spindle will be programmed in **constant surface speed (m/min)** and the feedrate will be REV/MIN.  
G96 is usually used when turning.

##### Programming

G96 S.. F..

Note: as the tool get closer to the centre line the spindle speed will increase. As the tool gets further away from the centre line the spindle speed will decrease,  
G97 cancels



## Section 6

### Enhanced level commands

Notes

#### Canned cycles (G20 to G94)

note: which G code system is used. (A, B,C)

##### Function

By using canned cycles, it made easier for the programmer to create programs. By means of canned cycles, machining operations frequently used can be determined in a single block through a G function. Normally more than one block is required when operating without canned cycles. Using canned cycles can also shorten the program in order to save memory.

The three canned cycles below are what is called four block operation for one cycle e.g. in-feed, cutting (or threading), retraction, and return.

Description	Code A	Code B	Code C
Straight/taper cutting cycle	G90	G77	G20
Straight/taper thread cutting cycle	G92	G78	G21
Straight/taper facing cycle	G94	G79	G24

Cycles called by G70 to G76 (G code system A and B) perform multi passes

G code	Cycle name	Remark	
G70	Finishing cycle		
G71	Stock removal cycle, longitudinal axis	G70 cycle can be used for finishing	Nose R offset possible
G72	Stock removal cycle transverse axis		
G73	Contour repetition		
G74	Deep hole drilling and recessing in longitudinal axis		
G75	Deep hole drilling and recessing in transverse axis		
G76	Multiple thread cutting cycle		

Cycles called by G72 to G78 (G code system C) perform multi passes.

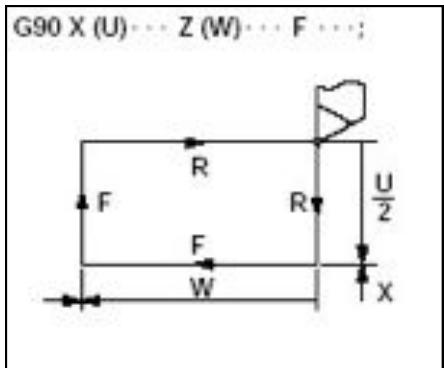
G code	Cycle name	Remark	
G72	Finishing cycle		
G73	Stock removal cycle, longitudinal axis	G72 cycle can be used for finishing	Nose R offset possible
G74	Stock removal cycle transverse axis		
G75	Contour repetition		
G76	Deep hole drilling and recessing in longitudinal axis		
G77	Deep hole drilling and recessing in transverse axis		
G78	Multiple thread cutting cycle		

## Section 6

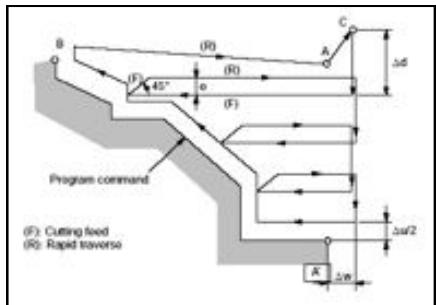
### Enhanced level commands

Notes

Four block operation for one cycle e.g. in-feed, cutting (or threading), retraction, and return.



G72 (G code system B) perform multi passes



## Section 6

### Enhanced level commands

Notes

#### Subprogram call up function (M98, M99)

##### Function

This function can be used when subprograms are stored in the part program memory, subprograms registered to the memory with program numbers assigned can be called up and executed as many times as required.

The created subprograms should be stored in the part program memory before they are called up.

##### Commands

M98	;Subprogram call up
Pxxxx	;program number
Lyyyy	;number of program runs
M99	;end of subprogram

##### Programming

M98 Pyyyyxxxxx ;  
Or  
M98 Pxxxx Lyyyy ;

G291  
:  
M98 P200  
G00 G40 Z1.9685  
:  
:  
:  
M30

200.mpf

#### Chamfering and corner rounding commands (R, C)

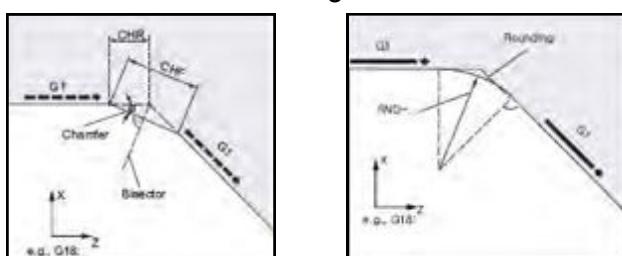
##### Function

It is possible to insert chamfering and corner rounding blocks automatically between:

- Linear interpolation and linear interpolation
- Linear interpolation and circular interpolation
- Circular interpolation and linear interpolation
- Circular interpolation and circular interpolation

##### Programming

C... ;chamfering  
R... ;corner rounding

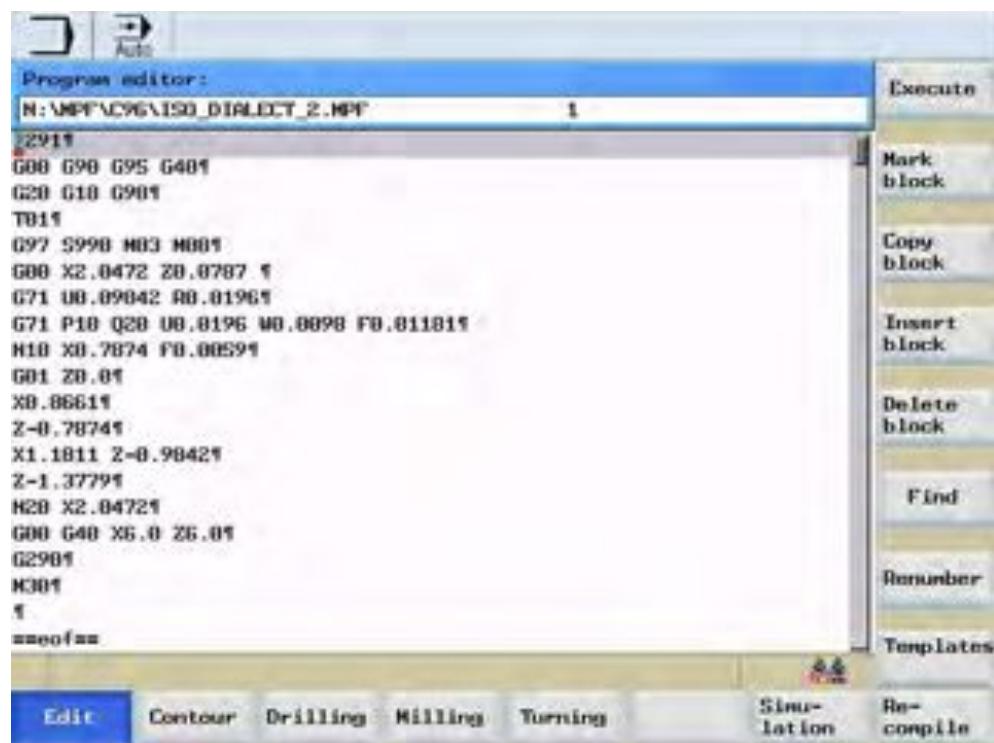


## Section 7

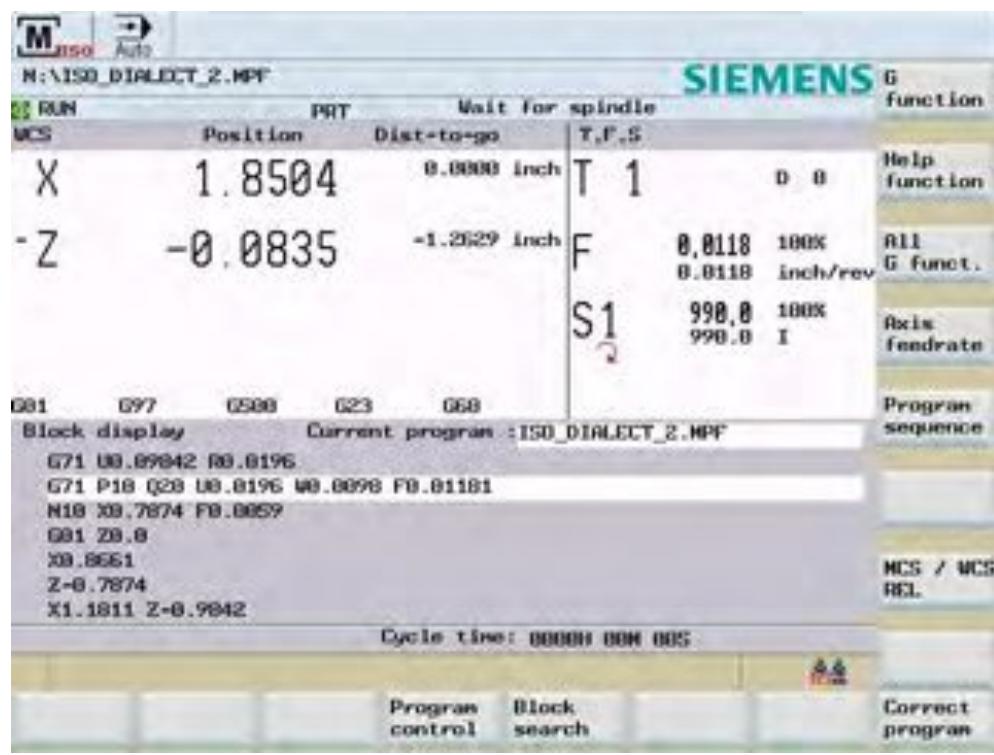
### Basic ISO Dialect program

Notes

An ISO Dialect program using the same basic structure as a Siemens program, but using instead a ISO Dialect stock removal cycle (G71).



The program above shown running in auto mode below: note the ISO in red text.

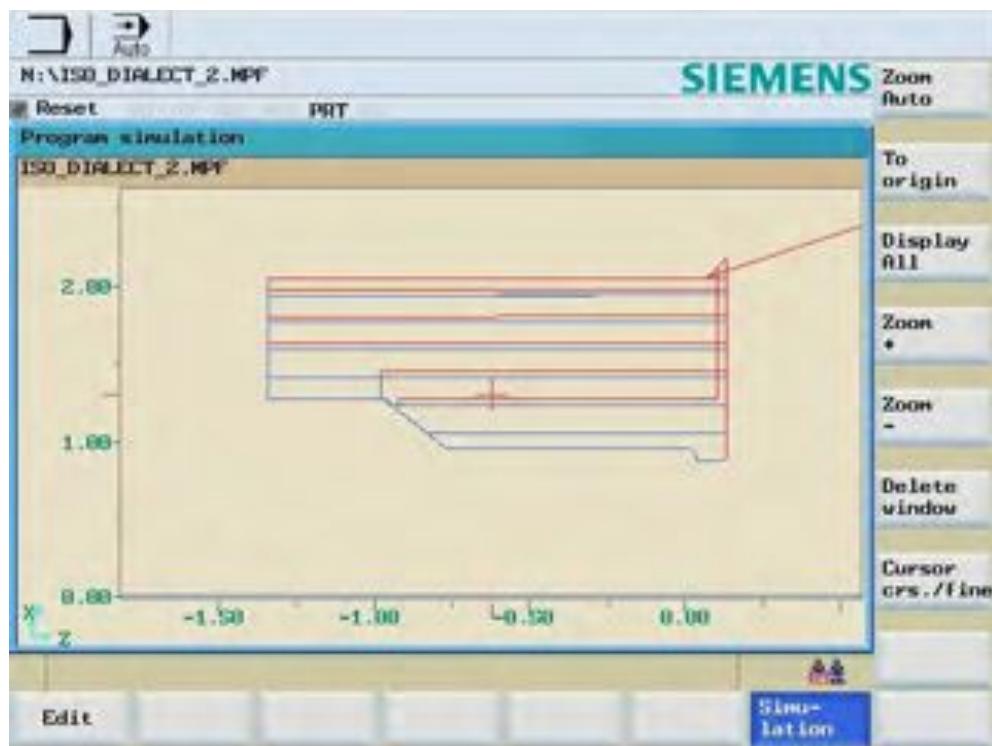


## Section 7

### Basic ISO Dialect program

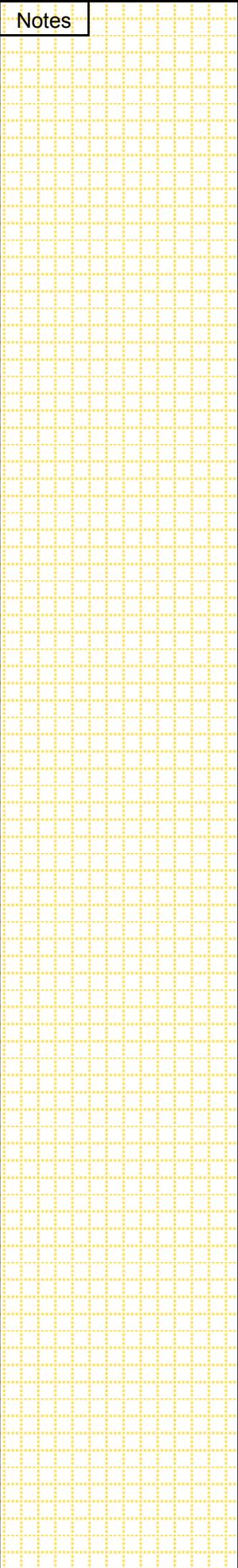
Notes

The same program shown from the simulation page.



---

Notes



## 1 Brief description

**Module objective:**

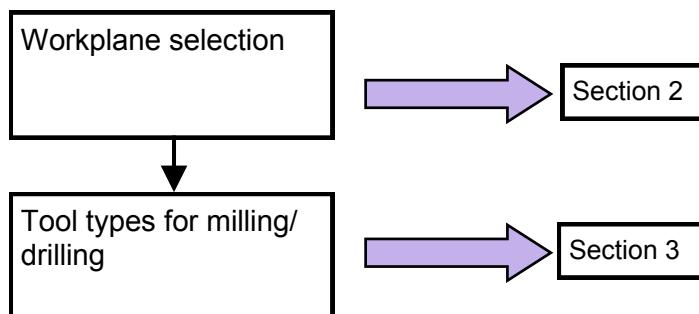
Upon completion of this module you will understand workplane selection, and tool types for driven tools on a turning machine.

**Module description:**

The tool offsets L1, L2 and L3 are assigned to the relevant axis with the workplane preparatory functions (G codes). This module explains this relationship, which is important when using driven tools.

**Module content:**

Workplane selection  
Tool types for milling/drilling



## Section 2

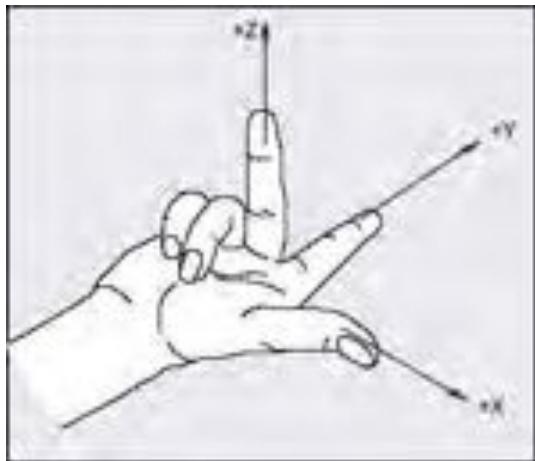
### Working Plane Selection

Notes

#### The right hand rule:

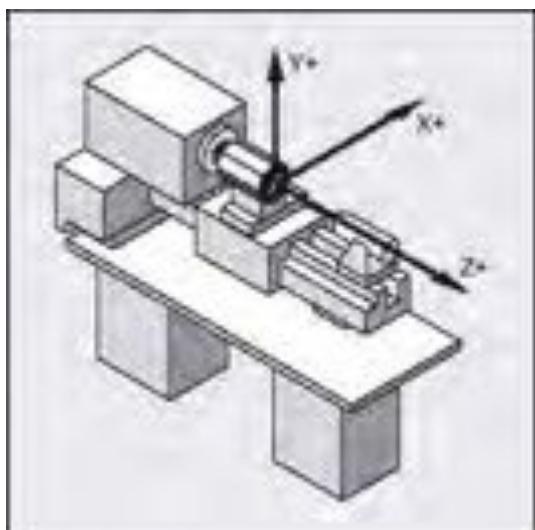
The orientation of the coordinate system relative to the machine depends on the machine type. The axis directions follow the so-called “three-finger-rule” of the right hand.

- The thumb points in the +X direction
- The index finger points in the +Y direction
- The middle finger points in the +Z direction



#### Turning coordinate system.

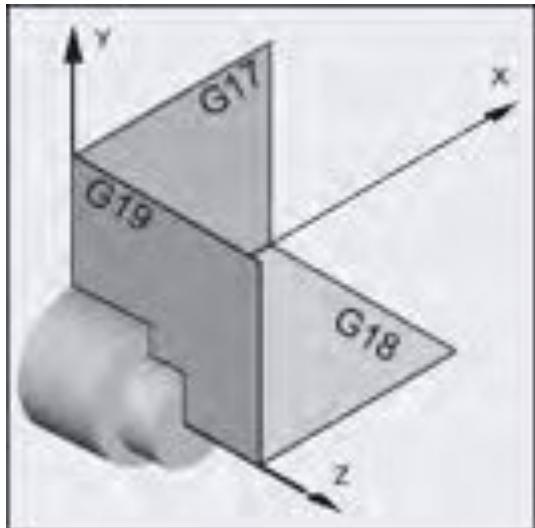
Appling the right hand rule to a turning machine.



## Section 2

### Working Plane Selection

Notes



Turning operations are programmed in the G18 plane.

Drilling & milling operations at the **end face** of the turned part are programmed in the G17 plane that the offset is applied correctly.

Drilling & milling operations at the **peripheral surface** of the turned part are programmed in the G19 plane that the offset is applied correctly

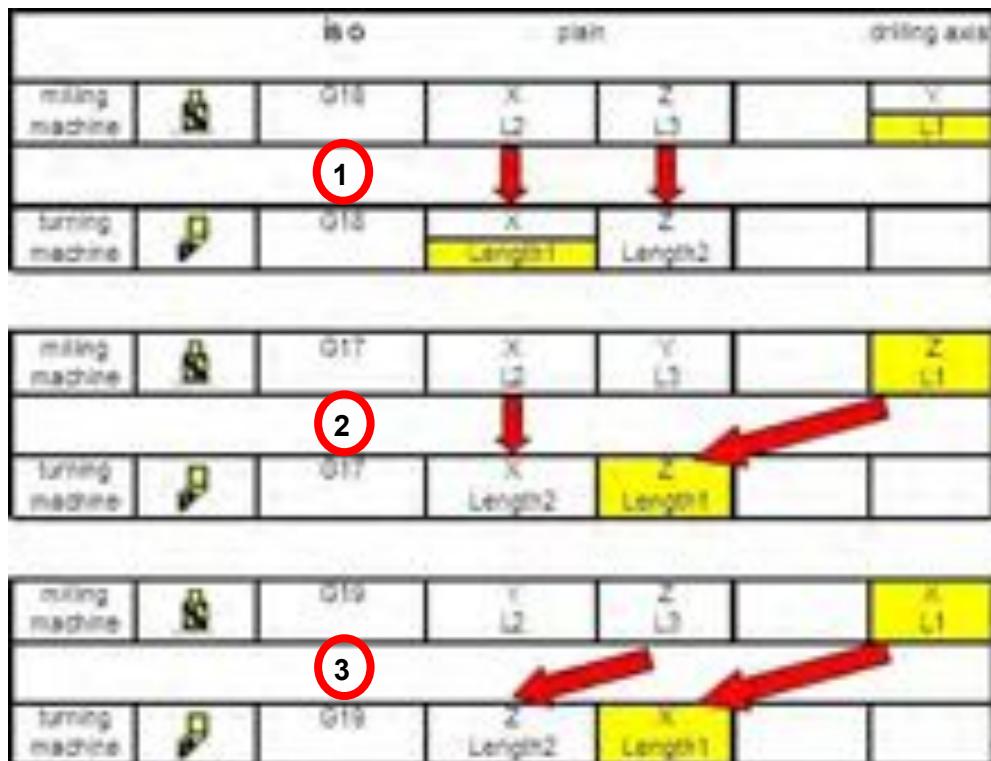
## Section 2

### Working Plane Selection

Notes

This is how the axis are assigned from MILLING to TURNING under ISO STANDARDS.

Make a note of axis assigned to Length1 & Length2 for Turning.



- 1 A turning machine has no Y axis, the default plane is ZX (G18). In the case of turning tools the offset L1 is applied to X
- 2 With G17 on a turning machine, there is no Y axis. Compensation L3 is therefore not required.
- 3 With G19 on a turning machine, there is no Y axis, in the case of a turning tool L2 is then assigned to the Z axis.

## Section 2

### Working Plane Selection

Notes

Typical program using G17 plane drilling in the X/Y axis plane

```
G00 G90 G95 G40 G17
LIMS=2500
G54
T1 D1          ;CALL DRILL TOOL
G94 S2=1000 F150 M2=03 ;START SPINDLE
G00 X0.0 Z2.0   ;RAPID TO SAFE POSITION
G01 Z-10.0      ;DRILL Z-10.0
Z2.0           ;RAPID TO SAFE POSITION
G00 X150.0 Z150.0 ;TOOL CHANGE POSITION
G18            ;SET PLANE TO DEFAULT
M30
```

Typical program using G19 plane drilling in the Y/Z axis plane

```
G00 G90 G95 G40 G19
LIMS=2500
G54
T2 D1          ;CALL DRILL TOOL
G95 S2=1000 F0.15 M2=03 ;START SPINDLE FOR LIVE TOOLING
G00 X20.0 Z2.0   ;RAPID TO SAFE POSITION
Z-30.0          ;RAPID TO SAFE POSITION
G01 X10.0        ;DRILL X10.0
G00 X20.0        ;RAPID TO SAFE POSITION
Z2.0           ;RAPID TO SAFE POSITION
G00 X150.0 Z150.0 ;TOOL CHANGE POSITION
G18            ;SET PLANE TO DEFAULT
```

The same planes apply when milling, the first part of any drilling or milling cycle is the **infeed** direction (or movement).

## Section 3

### Tool types for milling/drilling

When creating a drilling / milling tool you have to be aware of the tool type that you choose, because the tool type goes hand in hand with the workplane.

Use the following sequence to create a milling / drilling tool.

The screenshot shows the SINUMERIK 802D software interface. On the left, there is a toolbar with icons for Offset, Param, and MDA. Below the toolbar is a 'Tool list' tab. The main area displays a table titled 'Tool list' with columns: Type, T, D, Geometry, Wear, and Active tool no. The table contains seven rows of data. A context menu is open on the right side of the screen, listing options: 'D >>', 'Delete tool', 'Extend', 'Edges', 'Find', 'New tool', and 'User data'. At the bottom of the screen, there is a navigation bar with icons for Back, Forward, Home, and Stop.

Type	T	D	Geometry	Wear	Active tool no	I	O	1	0	1
			Length1 Length2 Radius	Length1 Length2 Radius						
1	1	0.0000	0.0000	0.0315	0.0000 0.0000 0.0000	3				
2	1	0.0000	0.0000	0.0157	0.0000 0.0000 0.0000	3				
3	2	0.0000	0.0000	0.0039	0.0000 0.0000 0.0000	3				
4	1	0.0000	0.0000	0.0000	0.0000 0.0000 0.0000	3				
5	1	0.0000	0.0000	0.1298	0.0000 0.0000 0.0000	3				
6	1	0.0000	0.0000	0.1575	0.0000 0.0000 0.0000	3				

## Section 3

### Tool types for milling/drilling

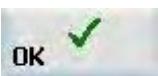
Notes

Type	T	D <sub>x</sub>	Geometry			Wear			I	D	T
			Length1	Length2	Radius	Length1	Length2	Radius			
1	1	1	0.0000	0.0000	0.0315	0.0000	0.0000	0.0000	3		
2	1	2	0.0000	0.0000	0.0157	0.0000	0.0000	0.0000	3		
3	2	2	0.0000	0.0000	0.0039	0.0000	0.0000	0.0000	3		
4	1	4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3		
5	1	6	0.0000	0.0000	0.1298	0.0000	0.0000	0.0000			
6	1	7	0.0000	0.0000	0.1575	0.0000	0.0000	0.0000			

- 1 If you wish to drill in the **End Face** of your turned component, you would choose the top softkey as your tool type

Type	T	D <sub>x</sub>	Geometry			Wear			I	D	T
			Length1	Length2	Radius	Length1	Length2	Radius			
1	1	1	0.0000	0.0000	0.0315	0.0000	0.0000	0.0000	3		
2	1	2	0.0000	0.0000	0.0157	0.0000	0.0000	0.0000	3		
3	2	2	0.0000	0.0000	0.0039	0.0000	0.0000	0.0000	3		
4	1	4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3		
New tool											
Drill											
Tool No.:											

- 2 Enter the tool number here followed by the OK soft key



## Section 3

### Tool types for milling/drilling

Notes

Please make a note of which GEOMETRY columns are which axis.

Type	T	D	X	Length1	Length2	Radius	Length1	Length2	Radius	Wear	
1	1	1	1	0.0000	0.0000	0.0315	0.0000	0.0000	0.0000	3	
2	1	2	1	0.0000	0.0000	0.0157	0.0000	0.0000	0.0000	3	
3	2	3	2	0.0000	0.0000	0.0039	0.0000	0.0000	0.0000	3	
4	1	4	1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3	
cot	6	1	6	0.0000	0.0000	0.1290	0.0000	0.0000	0.0000		
cot	7	1	7	0.0000	0.0000	0.1575	0.0000	0.0000	0.0000		
cot	8	1	8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		

3 This is your X axis offset. Length 2

4 This is your Z axis offset. Length 1

Shown below is a typical program for drilling on the **End Face** of your turned component using G17.

```
N:\MPF\MPF_DRILL_G17.MPF
M00 G90 G25 G40 G17
L0H5=25001
G541
T8 D11
SETH51
SPOS=01
G94 S2=1000 E H2=031
G80 X0.0 Z0.19681
CYCLE82( 0.19680, 0.00000, 0.07870, -0.39370, 0.00000, 0.10000 )
G80 X6.0 Z6.01
SETH51
G181
M301
T
#mcufm
```

## Section 3

### Tool types for milling/drilling

Notes

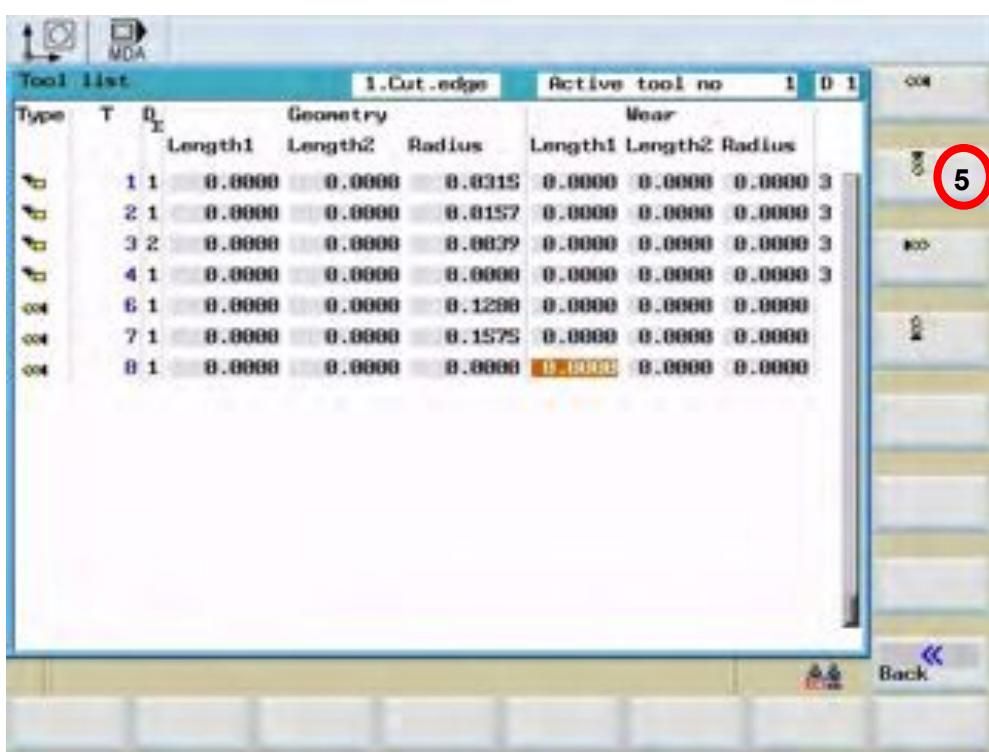
If you wish to drill in the **peripheral surface** of your turned component, you would choose the second softkey as your tool type.

Follow the sequence.

OFFSET  
PARAM

New  
tool

Drill



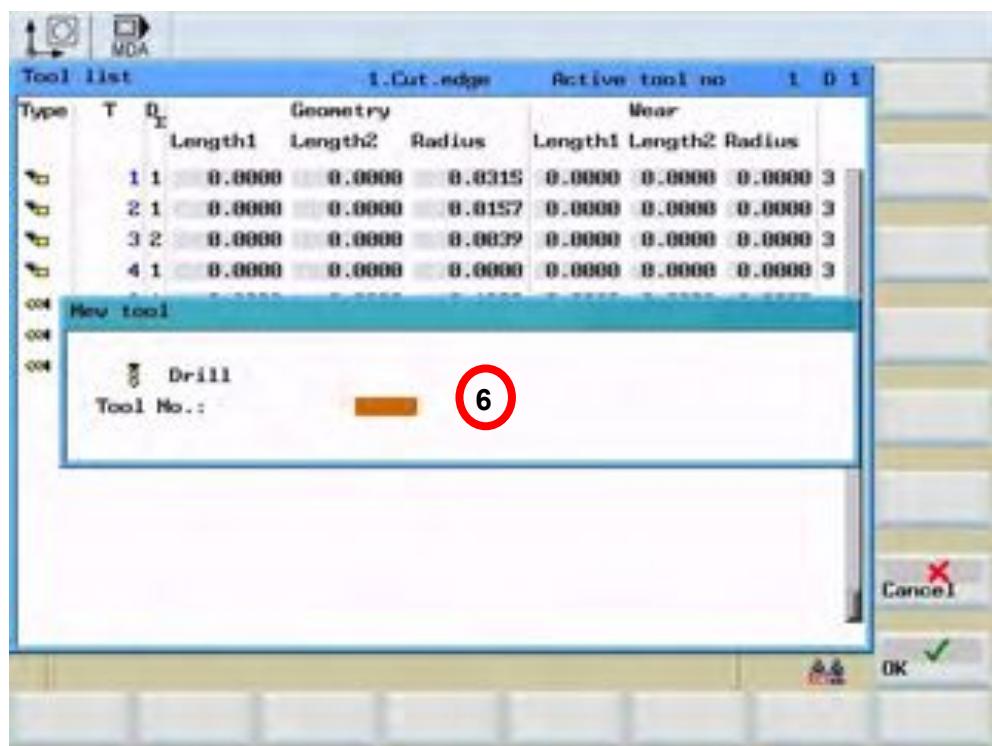
- 5 Soft key to choose milling tools on the **peripheral face**.

Note: this tool type is chosen as long as the configuration of the machine is set with the turret behind the spindle centre line.

## Section 3

### Tool types for milling/drilling

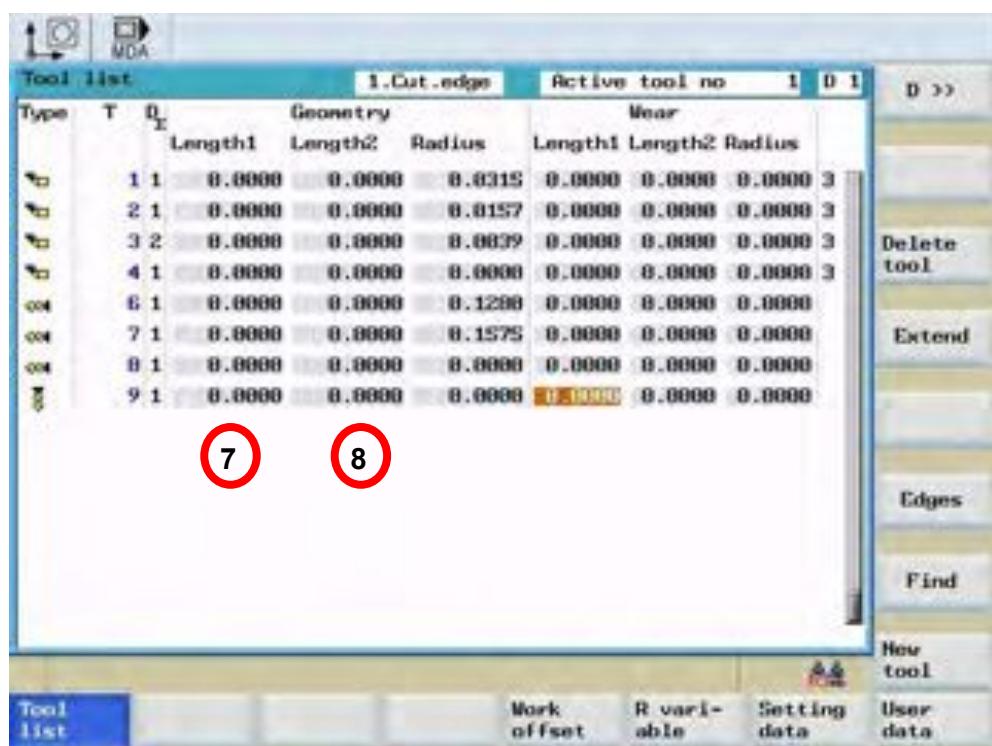
Notes



- 6 Enter the tool number here followed by the OK soft key

When you have selected this tool type, you must remember to use **G19** as your working plane.

When you set the **tool offset** you must also remember to set the correct **Axis to Length 1** and **Length 2** in the **Geometry** column.



- 7 This is your X axis offset. Length 1  
8 This is your Z axis offset. Length 2

## Section 3

### Tool types for milling/drilling

Notes

Shown below is a typical program for drilling on the **peripheral surface** of your turned component using **G19**.

The screenshot shows a CNC program editor interface. The main window displays a G-code program:

```
N:\MPPF\MP1_DRILL_G19.MPF
G00 G90 G25 G40 G19
LIM5=25001
G841
T9 D11
SETM51
SPOS=01
094 S2=1000 F6 M2=031
G00 X1.9685 Z0.19681
Z=1.96851
CYCLE824 1.96850, 1.57480, 0.87870, 0.00000, 0.39370, 0.10000)E
G00 X6.0 Z6.01
SETM51
G181
M301
f
**eof**
```

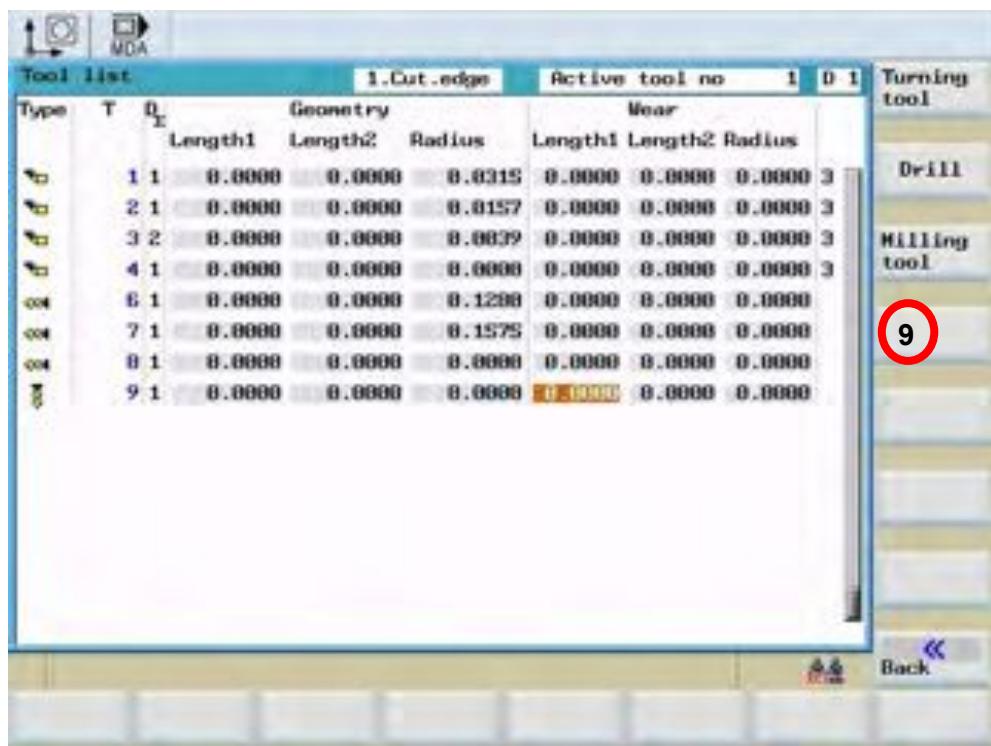
The status bar at the bottom shows tabs for Edit, Contour, Drilling, Milling, Turning, Simulation, and Re-compile. A context menu is open over the first line of code, listing options: Execute, Mark block, Copy block, Insert block, Delete block, Find, Renumber, and Templates.

## Section 3

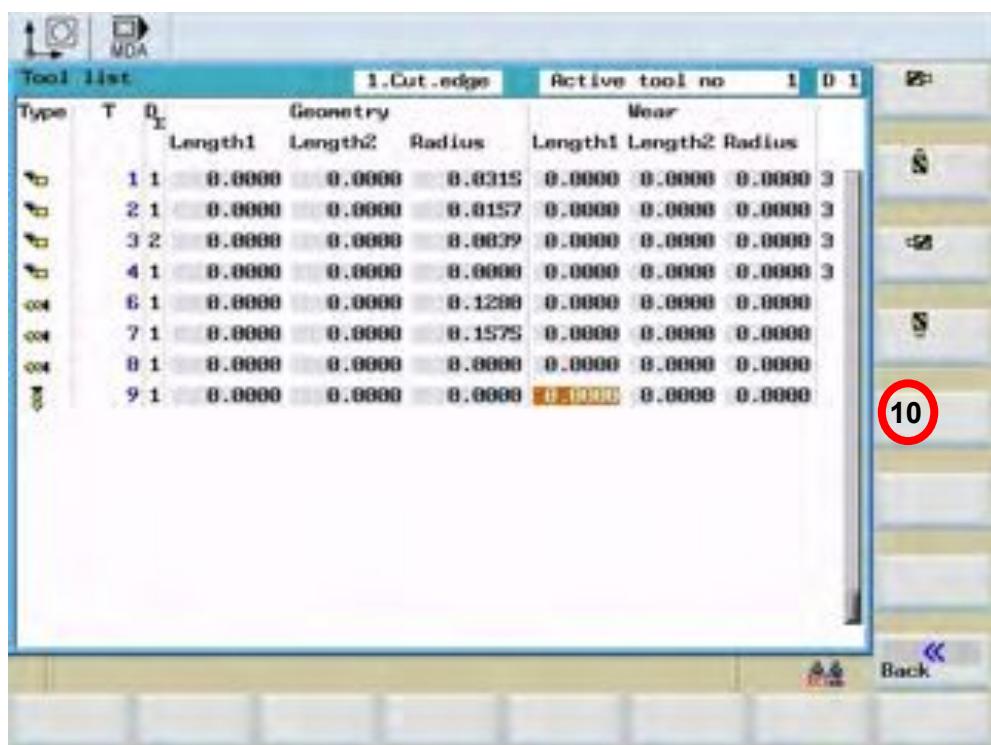
### Tool types for milling/drilling

Notes

When choosing the tool type for **Milling**, the only difference is the graphical representation.



9 Soft key to choose milling tools



10 Soft key to choose the different planes for milling

## 1 Brief description

### Module objective:

Upon completion of this module you can identify which Machine Control Panel (MCP) is installed and perform a pushbutton test

### Module description:

There are two possibilities currently existing to connect a Machine control panel to the 802D SL CNC.

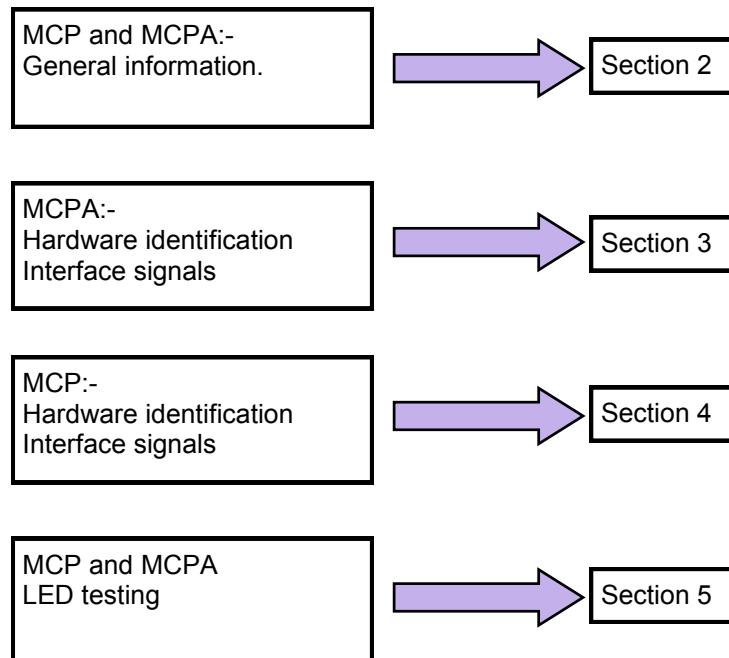
1. The first variant is to connect an MCP to the (additionally required) MCRA board, section 2.2.
2. The second variant is to connect the MCP to the PLC periphery, section 2.2

The hardware differs between the two variants, the part numbers (MLFB) are given in the respective sections.

In certain cases it may become necessary to perform a pushbutton test. The objective of the test, is to determine whether or not a pushbutton / LED on the machine control is defect.

Hardware identification  
Hardware interface

### Module content:



## Section 2

### MCP and MCPA versions

#### 2.1 General information.

The following diagram shows the layout of the 802D machine control panel.

The layout is the same for both of the MCP types.

The CNC functionality is the same for both of the MCP types.

The addressing of the inputs of the MCP differs between MCP types.

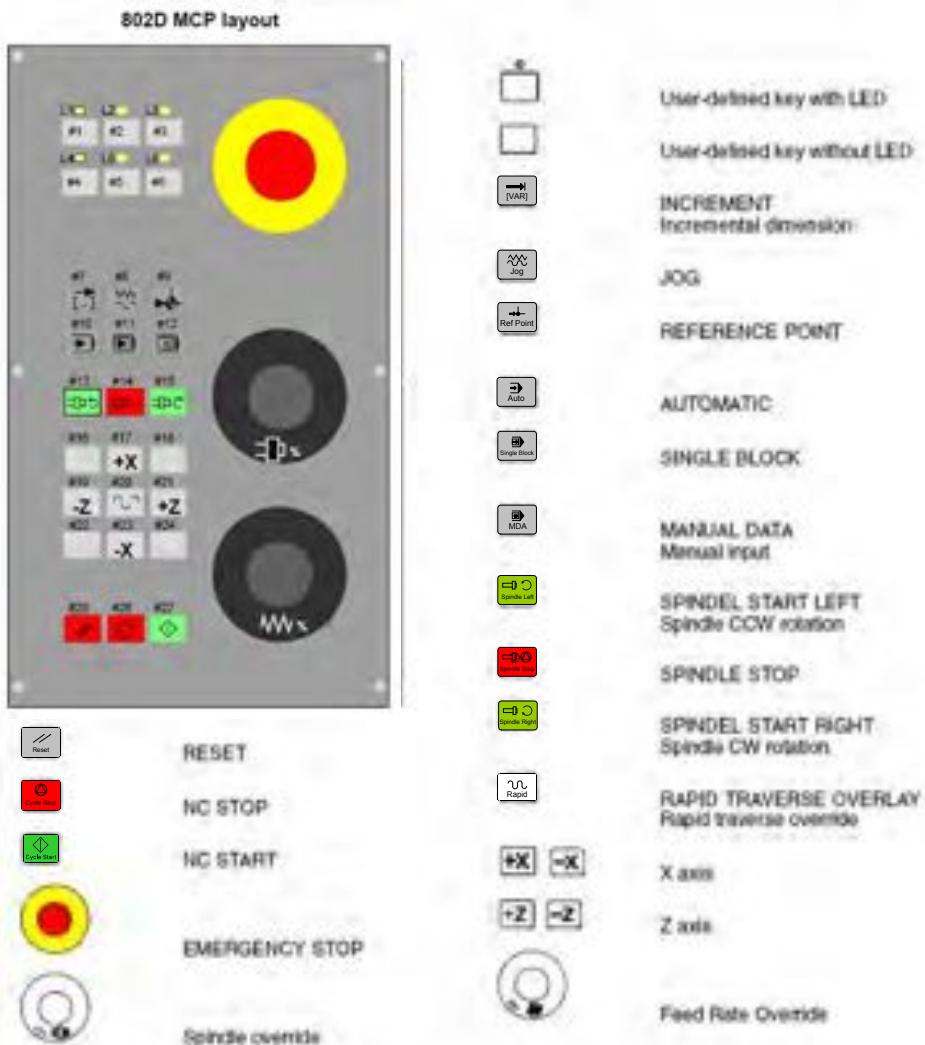
The addressing of the customer LED's differs between MCP types.

The input voltage differs between the MCP types.  
(MCPA = 5V DC, MCP = 24V DC)

Each pushbutton is assigned a mnemonic identifier.

Example: NC START      key is given mnemonic #27  
              +X            key is given mnemonic #17

#### 2.2 MCP/MCPA layout



After performing a pushbutton test, it may be found that all pushbuttons are defective and not just one key or lamp.

In the case where all keys and lamps are not working the MCPA board or the PP72/48 could be faulty and not necessarily the MCP.

Notes

## Section 3

### MCPA - Hardware identification

To carry out a pushbutton test, it is necessary to differentiate between the two MCP variants. When viewed from the front, both MCP's are identical in appearance.

Notes

#### 3.1 Identification of MCP connected to MCPA interface card.

Identification can be done visually by simply examining the hardware of the MCP from the reverse side. The following photograph shows the MCPA (Machine Control Panel Analogue) version. This method comprises of two parts, the MCP and the MCPA interface card.

The MCPA interface card is mounted directly on the PCU210.3.

The MCP connects into the MCPA interface card by way of two 37-way ribbon cables..



The second method of identification is to check the status of PLC Variable V18001000.7 (P\_H\_MCPA)-MCPA exists. The control detects the physical presence of the MCPA and sets this bit=1.

Note:-Plc variable V18001000.7 can be made to change status via the "PLC status" screen, even when the MCPA is present. This will be reset on power Off/On of the control.

Any attempt to write to the variable in the PLC program will result in a "PLC stop" alarm.

Apart from interfacing between the MCP and the PCU210.3, the MCPA also offers extra functionality such as the controlling of a single analogue axis and digital I/O. **Module C11** explains this in more detail.

## Section 3

### MCPA - Hardware identification

The following table contains product information for the MCPA.

Notes

Order No.	6FC5303-0AF30-1AA0
Product name	MCP 802D sl machine control panel
Input voltage	5 V DC ±20%/-15%
Power consumption	5 W
Inputs/outputs	Connector acc. to ML-C-83-603/DIN 41-651
Degree of protection to EN 60529 (IEC 60529)	
• Front	IP 64
• Rear	IP 00
Condensation	not admissible
Ambient temperature	
• Storage	-20 ... +60°C (-4 ... +140°F)
• Transport	-20 ... +60°C (-4 ... +140°F)
• Operation	0 ... +50°C (+32 ... +122°F)
Approx. weight	1.5 kg (3.30 lb)
Dimensions	
• Width	170 mm (6.69 in)
• Height	300 mm (12.99 in)
• Depth	60 mm (2.36 in)

#### 3.2 Interface signals.

Machine control panel



Machine control panel I/F



It is important to correctly connect the ribbon cables between the MCP and the MCPA interface board.

MCP connector X1201 to MCPA Interface board connector X1.

MCP connector X1202 to MCPA Interface board connector X2.

# Section 3

## MCPA - Interface signals

The MCPA uses PLC variables for the interface signals. These are shown in the table below. To carry out a pushbutton test these variables are monitored via the PLC status function of the control or the PLC802 programming tool. Each pushbutton (Key) has been given an mnemonic identifier ranging from #1 to #27

## Notes

#### Status signals from MCPA

10001xxx				Signals from MCPA Interface MCPA → PLC (ReadWrite)					
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
10001000	Key#8	Key#7	Key#6	Key#5	Key#4	Key#3	Key#2	Key#1	
	JOG	INC	User key 8	User key 5	User key 4	User key 3	User key 2	User key 1	
10001001	Key#16	Key#15	Key#14	Key#13	Key#12	Key#11	Key#10	Key#9	
	4" -	Spindle	Spindle	Spindle	MDA mode	Single Block mode	Auto mode	Referencing	in JOG
10001002	Key#24	Key#23	Key#22	Key#21	Key#20	Key#19	Key#18	Key#17	
	4" =	1" -	2" -	3" +	Rapid	3" -	2" +	1" +	
10001003						Key#27	Key#26	Key#25	
						NC start	NC stop	Reset	
10001004				E	D	C	B	A	
10001005				E	D	C	B	A	
						Spindle override			

### Status signals to MCPA

Status Signals to MCPU			Signals to MCPU Interface PLC → MCPU (ReadWrite)					
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
11001000			LED 6	LED 5	LED 4	LED 3	LED 2	LED 1
			User key 5	User key 5	User key 4	User key 3	User key 2	User key 1

**Pushbutton testing examples based on the above diagram:**

#### NC START Pushbutton.

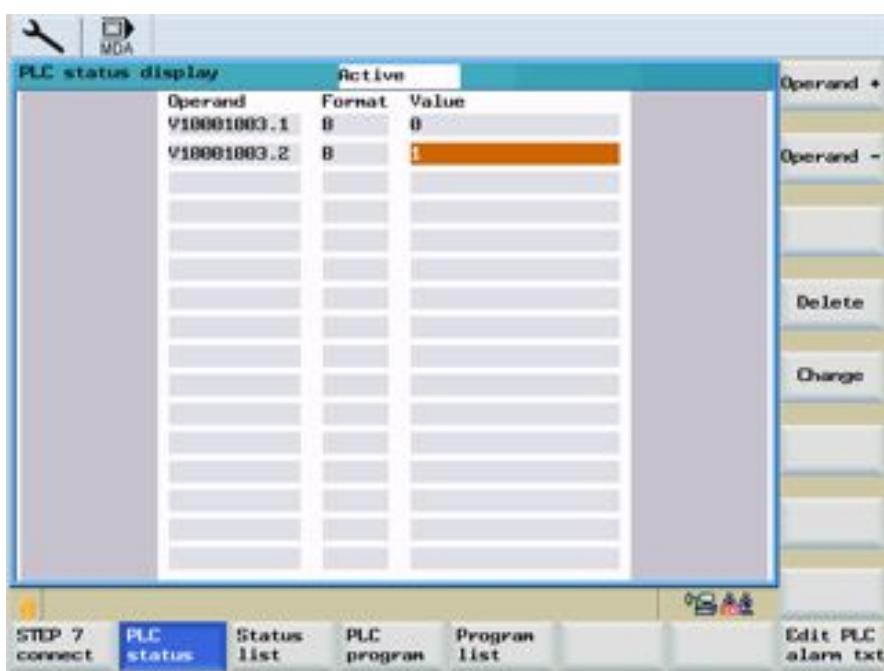
NC START = mnemonic #27  
#27 = V10001003.2

#### NC STOP Pushbutton

NC STOP = mnemonic #26  
#26 = V10001003 1

Due to the update time of the HMI software you should keep the key pressed in order for the change to register on the screen.

Example: How the signals are tested using “PLC Status” on the control. Here the “NC Start” variable V10001003.2=1 indicating that the button is activated



## Section 4

### MCP - Hardware identification

#### 4.1 Identification of MCP connected to a PP72/48 periphery board.

The **MCP** variant uses a PP72/48 periphery board to interface to the control.

Due to both variants of MCP looking identical from the front, Identification is done by visual inspection of the hardware and checking of part numbers. Variable V18001000.7 can be checked in the PLC status area.

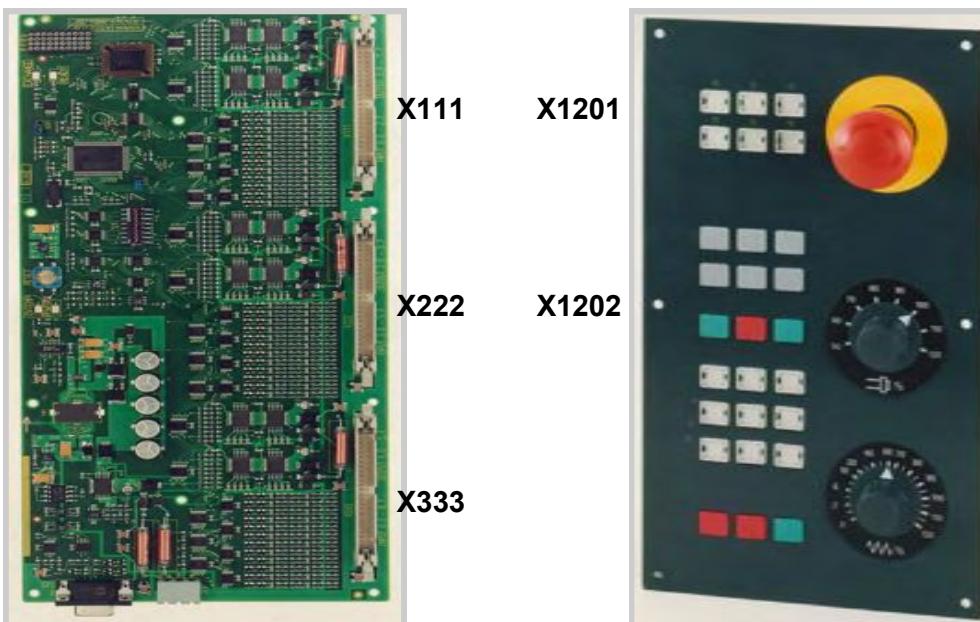
If V18001000.7 = 0, MCP is in use. If V18001000.7 = 1, MCPA is in use.

It must be noted that the variable can be set in the PLC status area regardless of which type of MCP is fitted. This variable reverts to the correct status on power Off/ON of the control.

Attempting to change the state of the variable in the PLC program will Result in a PLC stop alarm.

Notes

#### Periphery board PP72/48



The location of the PP72/48 connectors can be seen above.

The following table contains product information for the MCP.

## Section 4

### MCP - Interface signals

#### 4.2 Interface signals.

A maximum of three PP modules can be fitted to the 802D sl. Compatibility must be taken into account before deciding where to connect the MCP. The following table shows the I/O assignment of an MCP connected to the first PP module. (PP72/48 plug X111 to MCP plug X1201 and PP72/48 plug X222 to MCP plug X1202)

In this case the I/O addressing will start at 0 (Zero).

Notes

Pin assignment of connectors X111/X222/X333 on PP72/48 1 I/O module

Pin	X111	X222	X333	Pin	X111	X222	X333	
1	0V(DC0M)			2	DC24V			
3	I0.0#1	I3.0#05	I6.0	4	I0.1#0	I3.1#05	I6.1	
5	I0.2#0	I3.2#07	I6.2	6	I0.3#4	I3.3	I6.3	
7	I0.4#5	I3.4	I6.4	8	I0.5#6	I3.5	I6.5	
9	I0.6#7	I3.6	I6.6	10	I0.7#8	I3.7	I6.7	
11	I1.0#5	I4.0#04#A	I7.0	12	I1.1#10	I4.1#04#B	I7.1	
13	I1.2#11	I4.2#04#C	I7.2	14	I1.3#12	I4.3#04#D	I7.3	
15	I1.4#13	I4.4#04#E	I7.4	16	I1.5#14	I4.5	I7.5	
17	I1.6#15	I4.6	I7.6	18	I1.7#16	I4.7	I7.7	
19	I2.0#17	I5.0#04#A	I8.0	20	I2.1#18	I5.1#04#B	I8.1	
21	I2.2#19	I5.2#04#C	I8.2	22	I2.3#20	I5.3#04#D	I8.3	
23	I2.4#21	I5.4#04#E	I8.4	24	I2.5#22	I5.5	I8.5	
25	I2.6#23	I5.6	I8.6	26	I2.7#24	I5.7	I8.7	
27,	Not assigned		28	Not assigned		29		
30			31			32		
33	Q0.0#1	Q2.0	Q4.0	34	Q0.1#2	Q2.1	Q4.1	
35	Q0.2#3	Q2.2	Q4.2	36	Q0.3#4	Q2.3	Q4.3	
37	Q0.4#5	Q2.4	Q4.4	38	Q0.5#6	Q2.5	Q4.5	
39	Q0.6#7	Q2.6	Q4.6	40	Q0.7#8	Q2.7	Q4.7	
41	Q1.0	Q3.0	Q5.0	42	Q1.1	Q3.1	Q5.1	
43	Q1.2	Q3.2	Q5.2	44	Q1.3	Q3.3	Q5.3	
45	Q1.4	Q3.4	Q5.4	46	Q1.5	Q3.5	Q5.5	
47	Q1.6	Q3.6	Q5.6	48	Q1.7	Q3.7	Q5.7	
49	DOCOM **		50	DOCOM =				

#### 3.2 Pushbutton testing examples based on the above diagram

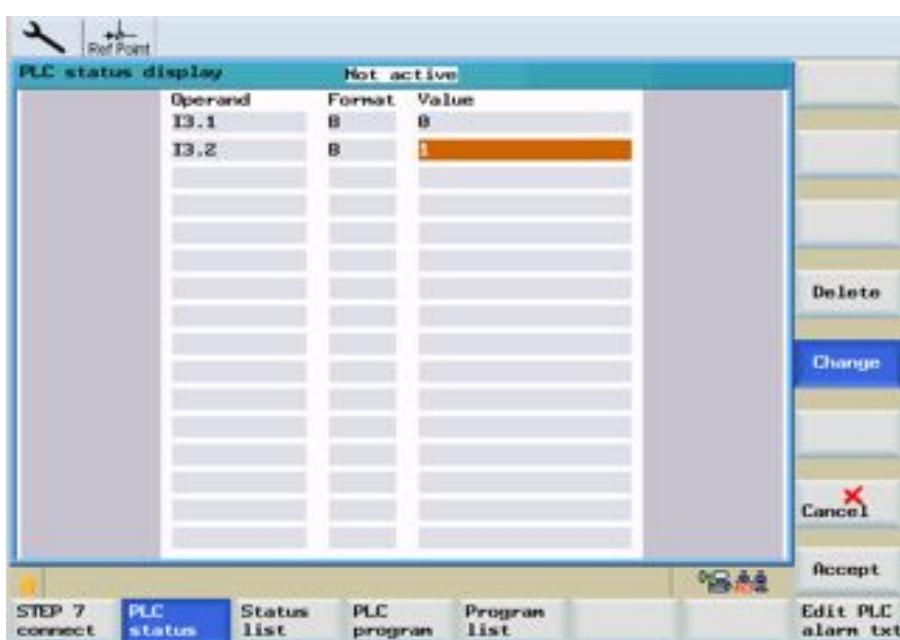
NC START Pushbutton.

NC START = mnemonic #27  
#27 = I3.2

NC STOP Pushbutton

NC STOP = mnemonic #26  
#26 = I3.1

Due to the update time of the HMI software you should keep the key pressed in order for the change to register on the screen.



The state of the inputs can be checked via "The PLC status" area of the control. The above example shows that the "NC start" button is activated.

## Section 5

### MCPA and MCP LED testing

Note:- Care must be taken when testing the 6 LED's.

The user PLC program must be consulted to ensure that artificially changing the state of an output does not result in any unwanted machine movement.

Notes

The "Change" function in the "PLC status" area of the control will not allow the changing of an output that is assigned in the PLC program. The PLC program itself will have to be edited to test the output.

Many machine tool builders have a special test function for the purpose of safely testing the LED's and lamps of a machine.

Other methods of testing the status of a pushbutton are "Status list" on the control and Programming Tool PLC802 software with the functions "Chart status" and "Program status".

**Module C28** offers more information of these methods.

## 1 Brief description

### Module objective:

Upon completion of this module you can locate the status LED's and determine the module status.

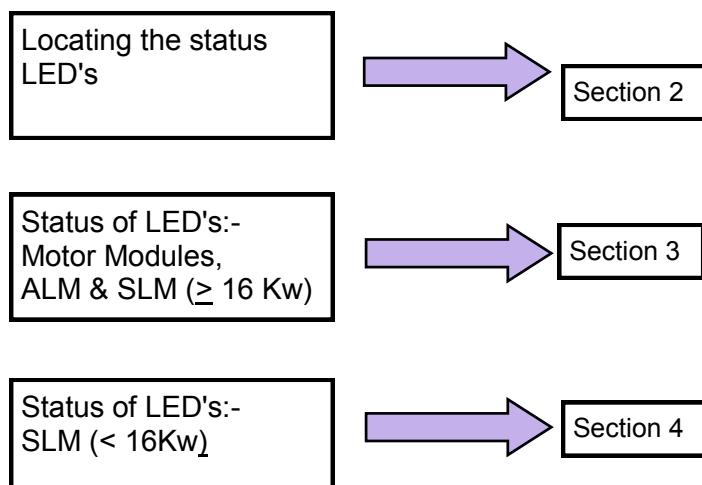
### Module description:

The 802D SL controller has extensive diagnostic possibilities, one of which is the LED diagnostic on the drive modules.

It is possible using the LED status to determine quickly the hardware status of the drive boards. Before investigating faults using external diagnostic tools (HMI alarm, Starter etc) a status should first be taken of the drive LED's.

### Module content:

- Locating the status LED's
- Status of Motor Module LED's
- Status of Active Line Module LED's
- Status of Smart Line Module LED's
- Status of Smart Line Module  $\geq 16\text{ kW}$  LED's



## Section 2

### Locating the status LED's

#### 2.1 Locating the status LED's

As can be seen from the picture below, the status LED's are always located in the same place on the drive modules.



Notes

## Section 3

### Status of LED's:-Motor Modules, ALM & SLM $\geq 16\text{Kw}$ .

#### 3.1 MM LED's, ALM LED's, SLM LED's $\geq 16\text{Kw}$

LED's for:

Motor Module  
Active Line Module  
Smart Line Module  $\geq 16\text{Kw}$

The above modules share identical LED functions, therefore the table below can be used for these module types.

In the case where a fault diagnosis is not possible a status of the LED's should be given to the OEM or to Siemens service personnel.

LEDs on ALM and MM- SLM  $\geq 16\text{Kw}$

LED	Color	Status	Meaning
READY	-	OFF	Power supply out of the permitted tolerance
	Green	ON	Drive ready and DRIVE CLQ communication active
	Orange	ON	DRIVE CLQ communication being established
	Red	ON	The module has at least one fault
	Green/Red	Flash 2Hz	Firmware is being downloaded
	Green/Orange or Red/Orange	Flash 2Hz	Component recognition via LED ( P0124=1 )
DC LINK	-	OFF	Power supply out of the permitted tolerance
	Orange	ON	DC link voltage within the permitted tolerance (only when both modules are ready)
	Red	ON	DC link voltage out of the permitted tolerance (Only when module is ready)

## Section 4

### Status of LEDs:- SLM < 16kW

#### 4.1 SLM LED's < 16kW

LED's for:-

##### Smart Line Module < 16 Kw

The above module does not have DriveCliq capability, therefore the LED functionality is different. The table below describes the meaning of the status LED's.

In the case where a fault diagnosis is not possible a status of the LED's should be given to the OEM or to Siemens service personnel.

LEDs on SLM: 5-10kW

LED	Color	Status	Meaning
READY	Green	ON	Drive ready
	Orange	ON	Pre-charging
	Red	ON	Ovvervoltage, overtemperature or voltage out of the permitted tolerance or DC link voltage out of the permitted tolerance
DC LINK	-	OFF	Power supply out of the permitted tolerance
	Orange	ON	DC link voltage within the permitted tolerance
	Red	ON	DC link voltage out of the permitted tolerance

in the case where a fault diagnosis is not possible a status of the LED's should be given to the OEM or to Siemens service personnel.

Notes



## 1 Brief description

**Module objective:**

Upon completion of this module you can locate the status LED's and determine the status of the CNC controller

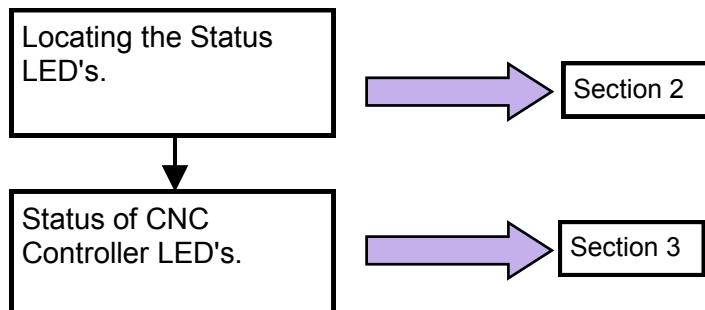
**Module description:**

The 802D SL controller has extensive diagnostic possibilities, one of which is the LED diagnostic on the CNC Controller.

It is possible using the LED status to determine quickly the hardware status of the CNC controller. Before investigating faults using external diagnostic tools (HMI alarm, Starter etc) a status should first be taken of the controller LED's.

**Module content:**

Locating the Status LED's  
Status of CNC Controller LED's



## Section 2

### Locating the Status LED's

#### 2.1 Locating the Status LED's

Notes

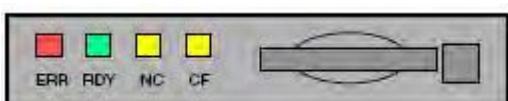
As can bee seen from the picture below, the status LED's are located above the LCD display on the right hand side, behind the plastic cover.



## Section 3

### Status of CNC Controller LED's

#### 3.1 Status of CNC Controller LED's



In normal run condition:-  
The green “RDY” LED is lit constantly.  
The yellow “NC” LED flashes at approx. 1Hz  
The yellow “CF” LED will flash when the CF card (Internal or external) is accessed.

In the case of ERR (red) a further investigation is required, using internal and external diagnostic tools in general.  
Example:- Alarm “400000 PLC stop 2” can be caused by addressing a non-existent input. In this case the “ERR” LED would be permanently lit.  
The cause of the error would need to be investigated with the Programming tool “PLC802”

## 1 Brief description

**Module objective:**

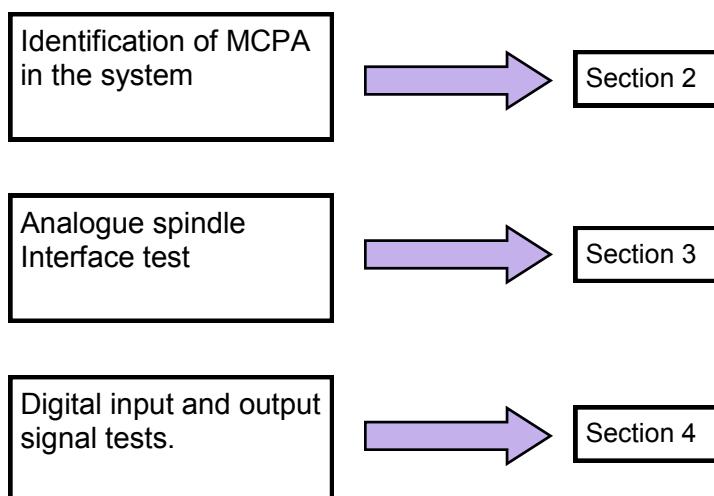
Upon completion of this module you can:-  
Locate the MCPA board in the system  
Perform a hardware diagnosis of the analogue spindle speed setpoint.  
Test the digital inputs and outputs.

**Module description:**

The MCPA board can be used to drive an analogue spindle and also to connect an MCP. Diagnosing MCP signals can be carried out with the help of module C8 (pushbutton test). When hardware is not correctly commissioned it can result in damage to hardware, in this case the MCPA board itself.

**Module content:**

Identification of MCPA in the system.  
Analogue spindle interface test.  
Digital Input and output tests.



## Section 2

### Identification of MCPA in the system

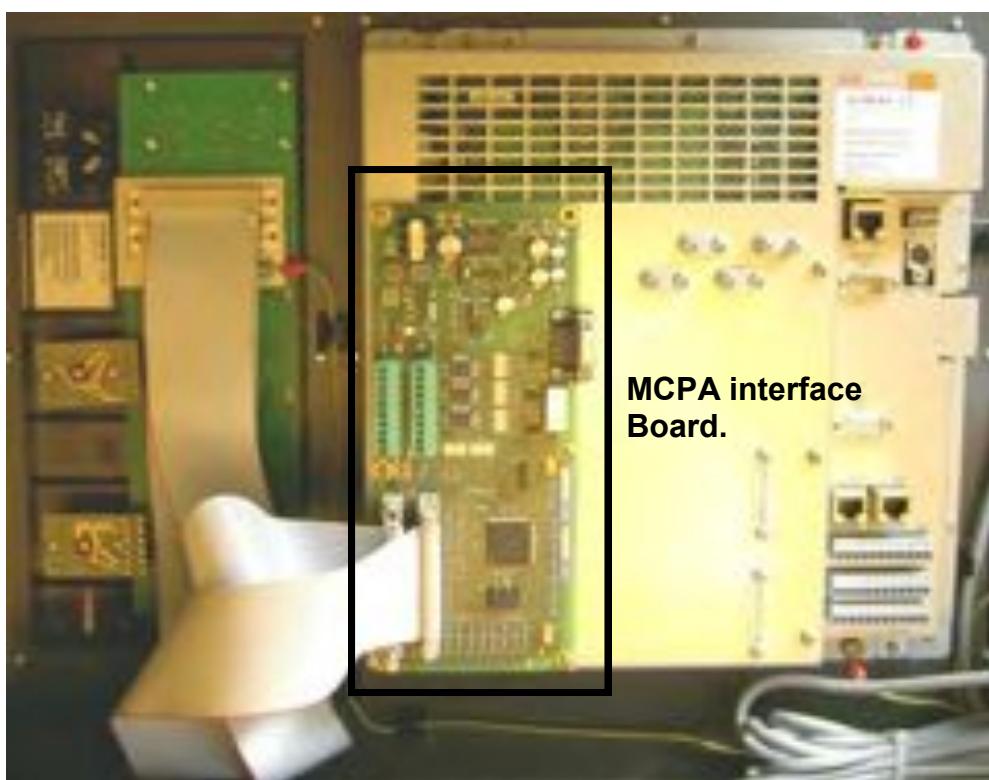
#### 3.1 Identification of MCPA interface card.

Identification can be done visually by simply examining the hardware of the from the reverse side. The following photograph shows the MCPA (Machine Control Panel Analogue) version. This method comprises of two parts, the MCP and the MCPA interface card.

The MCPA interface card is mounted directly on the PCU210.3.

The MCP connects into the MCPA interface card by way of two 37-way ribbon cables.

Notes



#### 3.2 Location and description of connectors.

MCPA interface board brief description of connectors:-

X1-MCP I/O signals (X1201 of MCP)  
X2-MCP I/O signals (X1202 of MCP)

X1020-Digital NC inputs  
X1021-Digital NC outputs

**X701**

X701-Analogue spindle control  
(Speed set-point, enables & direction)

**X1020 X1021**

The MCPA requires a 24V DC supply. This is done via connectors:-  
X1021:- Pin 1 = +24V, Pin 10 = 0V  
X1020:- Pin 10 = 0V

**X2 X1**

## Section 3

### Analogue spindle signal test.

#### 3.1 Machine Data

The operation of the analogue spindle is dependant on various machine data.

One machine data in particular will determine the correct method of proceeding to test the signals on the analogue spindle connector X701.

The machine data in question is Axis Machine Data 30134-  
IS\_UNIPOLAR\_OUTPUT[0].

To find this machine data select the “System” area with the following simultaneous key presses:-



The machine data screen is located with the following softkey selection:-

Machine  
data

Axis  
MD

The correct axis must be selected.

Axis +      Axis -

In this example the Spindle (SP) axis is selected.

Axis-specific machine data		
	SP	
38100 CTRLOUT_SEGMENT_NR[B]	3	po
38118 CTRLOUT_MODULE_NR[B]	1	po
38128 CTRLOUT_NR[B]	1	po
38138 CTRLOUT_TYPE[B]	1	po
38132 IS_VIRTUAL_AX[B]	0	po
38134 IS_UNIPOLAR_OUTPUT[B]	0	po
38238 NUM_EHCS	0	po
38228 ENC_MODULE_NR[B]	1	po
38238 ENC_INPUT_NR[B]	1	po
38248 ENC_TYPE[B]	0	po
38242 ENC_IS_INDEPENDENT[B]	0	cf
38258 ACT_POS_ABS[B]	0.000000	po
38258 ABS_INC_RATIO[B]	4	po
38270 ENC_ABS_BUFFERING[B]	0	po
38300 IS_ROT_AX	1	po
38318 ROT_IS_MODULE	1	po

Axis +  
Axis -  
Refresh  
Find  
Continue  
Find  
Select  
group

The “find” function is available to locate the desired machine data.

Find

Find machine data

Enter number / string

30134

OK ✓

## Section 3

### Analogue spindle signal test.

The value of the machine data 30134 can now be seen.

Notes

Axis-specific machine data		
30134 IS_UNIPOLAR_OUTPUT[0]	1	po
30288 NUM_ENC5	0	po
30228 ENC_MODULE_NR[0]	1	po
30238 ENC_INPUT_NR[0]	1	po
30248 ENC_TYPE[0]	0	po
30242 ENC_IS_INDEPENDENT[0]	0	of
30258 ACT_POS_ABS[0]	0.000000	po
30250 ABS_INC_RATIO[0]	4	po
30270 ENC_ABS_BUFFERING[0]	0	po
30308 IS_ROT_AX	1	po
30318 ROT_IS_MODULO	1	po
30329 DISPLAY_IS_MODULO	1	po
30339 MODULO_RANGE	360.000000	degrees re
30340 MODULO_RANGE_START	0.000000	degrees re
30358 SIMU_RX_VDT_OUTPUT	0	po
30450 IS_CONCURRENT_POS_AX	0	re

There are 3 possible options for Machine Data 30134, these options and the effect they have on the analogue spindle output are described in the following pages.

#### 3.2 Analogue spindle connector (X701).

The MCPA interface board has a 9-pin male D type connector for the outputting of the spindle signals. This connector is labeled as X701.

The following table shows the functionality of the pins of connector X701.

Pin	Name	Function/Description
1	Analogue OUT	+/-10v Analogue output reference signal. Example:-T56 of Simodrive system.
2	Not used	
3	Uni-Dir2	+24v Digital output for Unipolar direction 1
4	Uni-Dir1	+24v Digital output for Unipolar direction 2
5	Enable 1	Normally open, volt free drive enable contact. Example:-T65 of Simodrive system. (Used with Enable 2)
6	Analogue OUT	0v Analogue output reference signal. Example:-T14 of Simodrive system.
7	Not used	
8	Not used	
9	Enable 2	Normally open, volt free drive enable contact. Example:-T9 of Simodrive system. (Used with Enable 1)

## Section 3

### Analogue spindle signal test.

#### 3.3 Testing the signals:-Machine Data 30134=0.

Example:-Select “MDA” mode on the controller and type in a spindle direction and speed.



MDA

In our example M3 S50 (Spindle clockwise at 50 rpm)

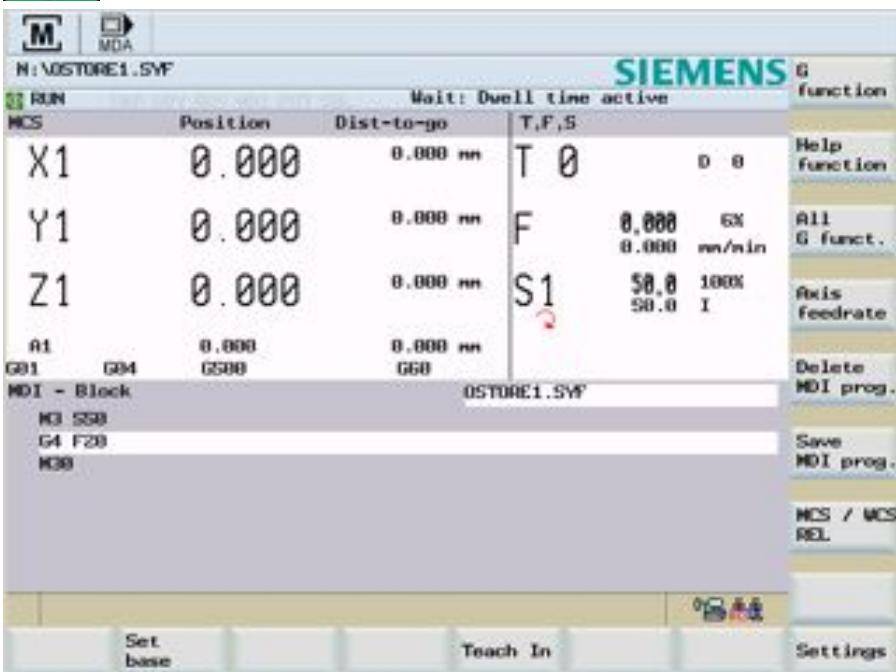
G4 F20 (Dwell for 20 seconds)

M30 (Program end)

Press “NC Start” he spindle will run clockwise at 50 rpm for 20 seconds then halt.



NC Start



Due to the type of connector, it is more convenient to measure the signals at the drive.

The expected signals for this situation will be:-

A positive voltage at X701 pin 1 with respect to pin 6.

The normally open contact between X701 pins 5 & 9 to be closed.

Note 1:- The polarity of the reference voltage between pins 1 & 6 can be reversed by Axis Machine Data 32100 AX\_MOTION\_DIR. If the default value of 1 is entered, the polarity of X701 pin 1 will be positive for a clockwise spindle command (M3). A value of -1 for machine data 32100 would cause X701 pin 1 to be negative for a clockwise spindle command.

Note 2:-The voltage level of pins 1 & 6 is determined by the commanded speed and Axis Machine Data 32250 RATED\_OUTVAL[0] & 32260 RATED\_VELO[0].

Machine Data 32250 is a percentage of the maximum analogue output value of 10V (100%=10V, 50%=5V etc)

Machine Data 32260 is the maximum speed of the spindle. At this speed the reference output on pins 1 & 6 will be at the level set in Machine Data 32250. At other commanded speeds the reference voltage will be Proportionate to this.

Notes

## Section 3

### Analogue spindle signal test.

In our example with the following machine data:-

Notes

30134 = 0

32100 = 1

32250 = 100

32260 = 500

The voltage at X701 pins 1 & 6 would be +1 Volt.

This is because the value of machine data 32250 = 100, this sets the reference voltage to 10V at the speed in machine data 32260 (500 RPM)

The commanded speed (50 RPM) is 10% of the max speed set in machine data 32260 (500 RPM). Therefore 10% of the reference value (1 Volt) is output.

If a spindle counter-clockwise (M04) command was programmed the Polarity of the voltage at X701 pins 1 & 6 would be reversed.

#### 3.4 Testing the signals:-Machine Data 30134=1

This setting only outputs a positive voltage (0 to 10v) on X701 pins 1 & 6 regardless of the programmed direction (M03 or M04).

To facilitate a change in direction:-

X701 pin 3 outputs +24V with the M04 command.

This signal can be utilised to reverse the spindle via relays or contactors which is necessary on certain types of drives.

X701 pin 4 will output +24V for both directions and can be used for enable purposes.

These tests assume Machine Data 32100 AX\_MOTION\_DIR = 1.

Changing 32100 = -1 will invert the behaviour of X701 pin 3

The signals on pins 3 & 4 should be measured to the 0V of MCPA Connector X1021 pin 10.

#### 3.5 Testing the signals:-Machine Data 30134=2

This setting only outputs a positive voltage (0 to 10v) on X701 pins 1 & 6 regardless of the programmed direction (M03 or M04).

To facilitate a change in direction:-

X701 pin 4 outputs +24V with the M03 command.

X701 pin 3 outputs +24V with the M04 command.

These signals can be utilised to reverse the spindle via relays or contactors which is necessary on certain types of drives.

The signals output on pins 3 & 4 should be measured to the 0V of MCPA Connector X1021 pin 10.

These tests assume Machine Data 32100 AX\_MOTION\_DIR = 1.

Changing 32100 = -1 will invert the behaviour of X701 pins 3 & 4

The signals on pins 3 & 4 should be measured to the 0V of MCPA Connector X1021 pin 10.

## Section 4

### Digital input and output signal tests.

#### 4.1 Digital input testing.

Before testing can proceed the physical location of the inputs needs to be known.

These are via connector X1020 on the MCPA board. (See page 2 section 3.2)

Connector X1020 pin assignment

Pin	Description	Designation
1	+24V DC	
2	Digital input 0	\$A_IN[9]
3	Digital input 1	\$A_IN[10]
4	Digital input 2	\$A_IN[11]
5	Digital input 3	\$A_IN[12]
6	Digital input 4	\$A_IN[13]
7	Digital input 5	\$A_IN[14]
8	Digital input 6	\$A_IN[15]
9	Digital input 7	\$A_IN[16]
10	0V DC	

There are eight inputs available, each one has a variable assigned to it. The first variable starts at number 9. The variables 1 to 8 are reserved for the PCU210.3/Sinamics (Connectors X20 and X21)

The operation of the Inputs depends on the setting of certain General Machine Data. Locate the machine data screen by referring to Section 3.1 machine data, page 3. The required machine data are accessed with the "General MD" softkey.

General  
MD

Cursor to or use the "Find" function to locate the correct machine data:-  
10350 FAST\_NUM\_DIG\_INPUTS = 2

The number of bytes available for use with fast digital inputs, this is set to 2 by default. The first byte is used for the PCU210.3/Sinamics.

Attempting to use digital inputs outside of the range set will result in alarm:-  
"017100 Channel 1 block 2 digital input/comparator no. **xx** not activated"  
Where "**xx**" = the digital input number.

10366 HW\_ASSIGN\_DIG\_FASTIN[0] = 00010101H

The hardware (MCPA) has to be assigned to the local bus by entering the above value. (Default value = 00H).

To test an Input, select "MDA" mode and type the following:-  
(Not he text in brackets)

MDA

PG: (Label)

R1=\$A\_IN[9] (Read the state of the digital input on X1020 pin 2 to R1)  
GOTOB PG (Search backwards through the program to the label "PG")



Press the "NC Start" button.

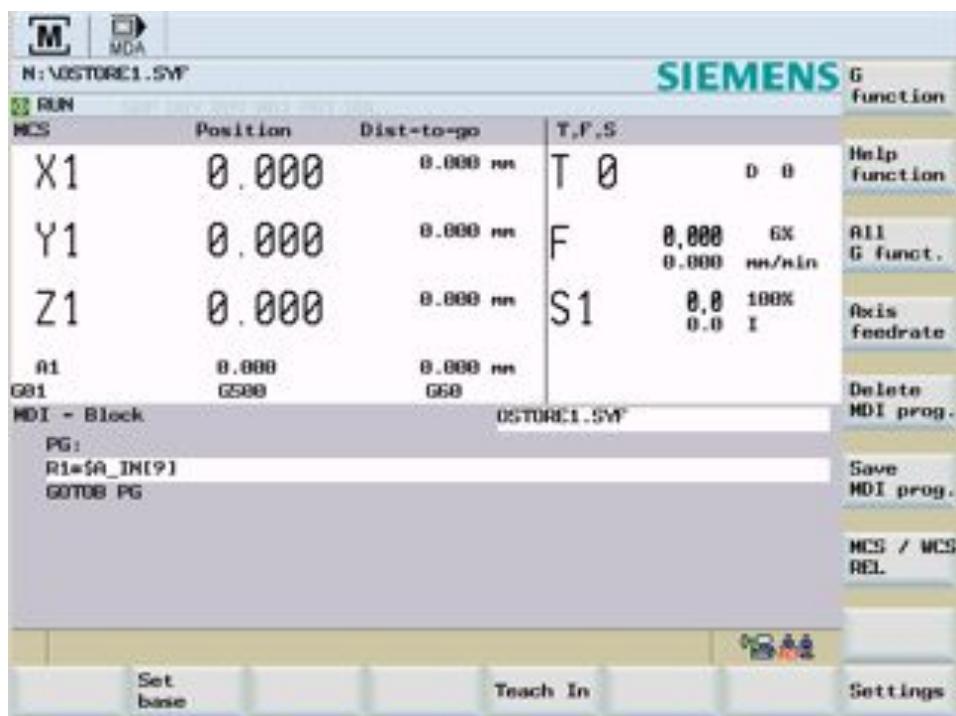
Notes

## Section 4

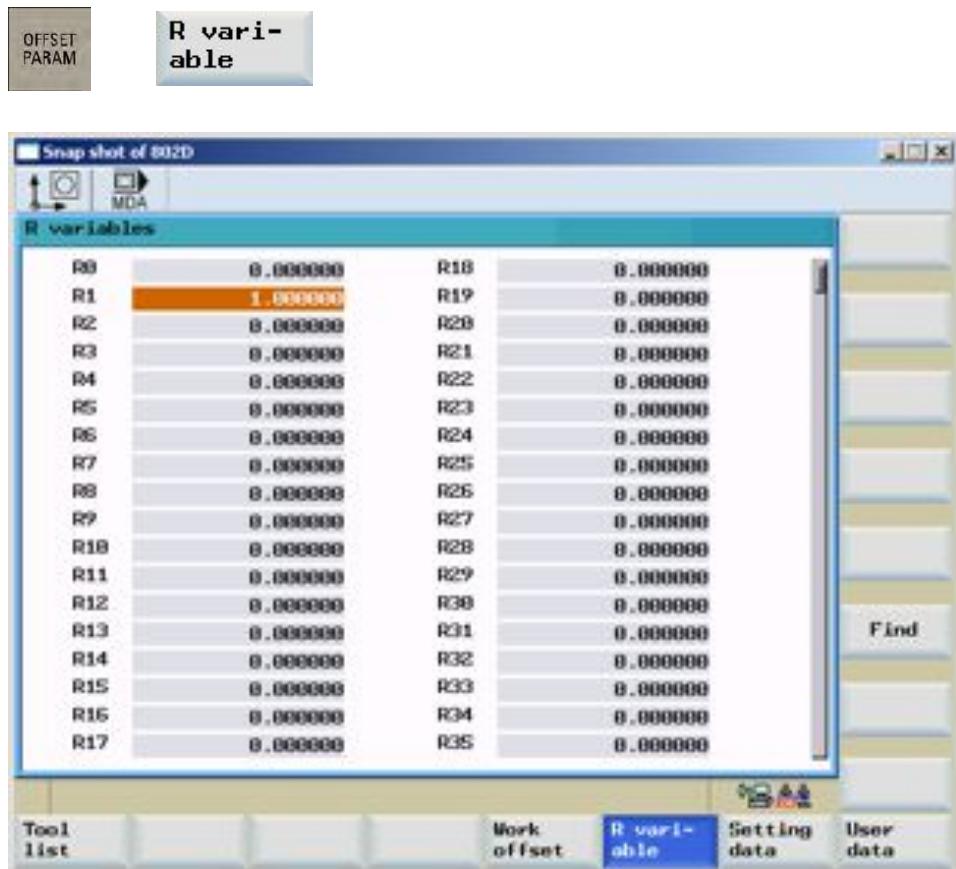
### Digital input and output signal tests.

The program will loop continuously allowing the state of the digital input to be monitored via R parameter R1.

Notes



Checking the state of the input on the “R variable” screen



The value of R1 = 1 which indicates that X1020 pin 2 should be +24V when measured to X1020 pin 10.

If the voltage was not present it would suggest a faulty MCPA board.

## Section 4

### Digital input and output signal tests.

#### 4.2 Digital output testing.

Before testing can proceed the physical location of the outputs needs to be known.

These are via connector X1021 on the MCPA board. (See page 2 section 3.2)

Notes

Connector X1021 pin assignment

Pin	Description	Designation
1		
2	Digital output 0	\$A_OUT[9]
3	Digital output 1	\$A_OUT[10]
4	Digital output 2	\$A_OUT[11]
5	Digital output 3	\$A_OUT[12]
6	Digital output 4	\$A_OUT[13]
7	Digital output 5	\$A_OUT[14]
8	Digital output 6	\$A_OUT[15]
9	Digital output 7	\$A_OUT[16]
10	0V DC	

There are eight outputs available, each one has a variable assigned to it. The first variable starts at number 9. The variables 1 to 8 are reserved for the PCU210.3/Sinamics (Connectors X20 and X21)

The operation of the outputs depends on the setting of certain General Machine Data. Locate the machine data screen by referring to Section 3.1 machine data, page 3. The required machine data are accessed with the "General MD" softkey.

General  
MD

Cursor to or use the "Find" function to locate the correct machine data:-  
10360 FAST\_NUM\_DIG\_OUTPUTS = 2

The number of bytes available for use with fast digital outputs, this is set to 2 by default. The first byte is used for the PCU210.3/Sinamics.

Attempting to use digital outputs outside of the range set will result in alarm:

"017110 Channel 1 block 2 digital output no. **xx** not activated"

Where "**xx**" = the digital output number.

10368 HW\_ASSIGN\_DIG\_FASTOUT[0] = 00010101H

The hardware (MCPA) has to be assigned to the local bus by entering the above value. (Default value = 00H).

To test the state of an output, select "MDA" mode and type the following:- (Not the text in brackets!)

MDA

PG: (Label)

R1=\$A\_OUT[9] (Read the state of the digital output on X1021 pin 2 to R1)  
GOTOB PG (Search backwards through the program to the label "PG")



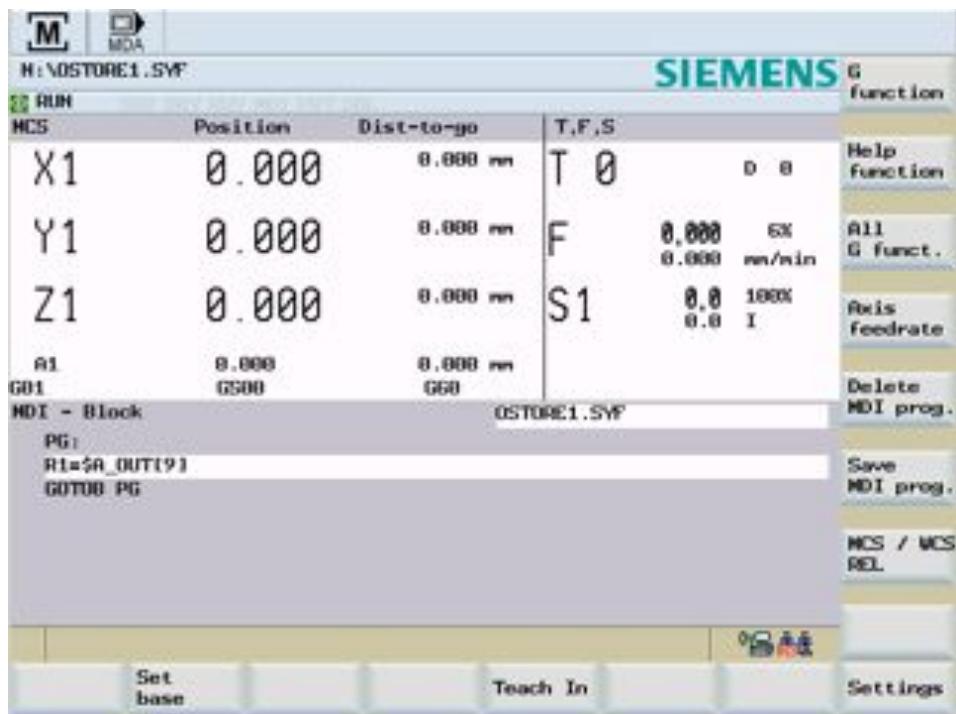
Press the "NC Start" button

## Section 4

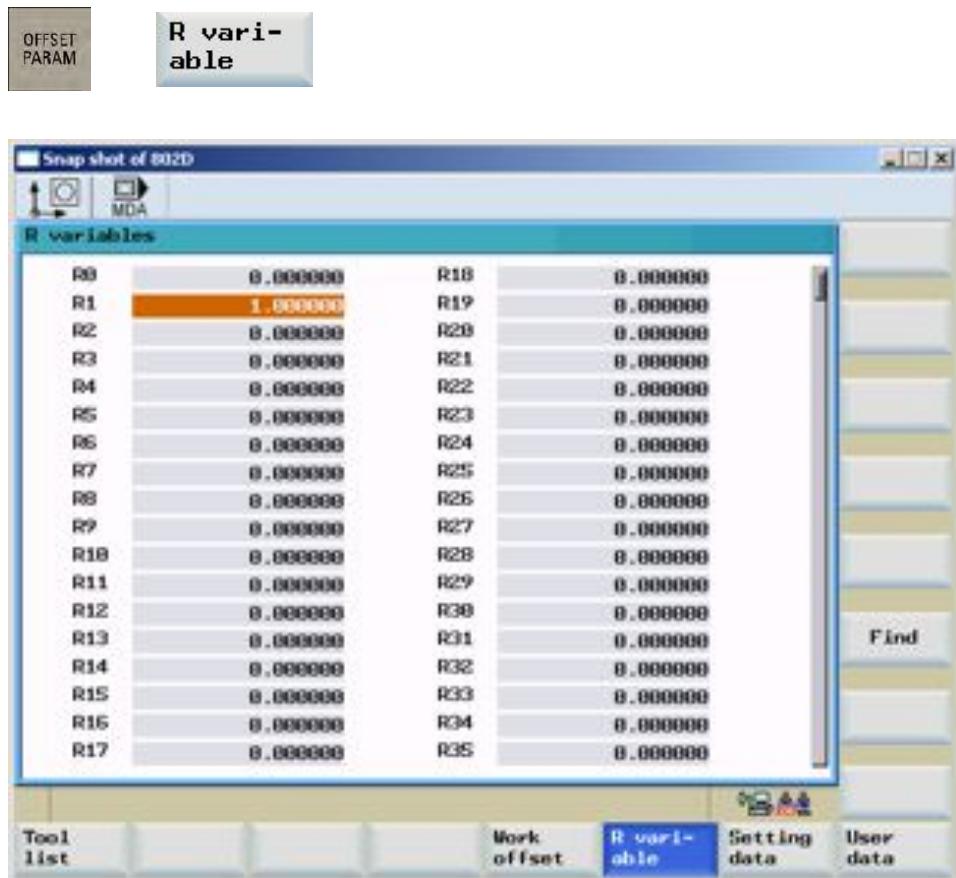
### Digital input and output signal tests.

Notes

The program will loop continuously allowing the state of the digital output to be monitored via R parameter R1.



Checking the state of the output on the "R variable" screen:-



The value of R1 = 1 which indicates that X1021 pin 2 should be +24V when measured to X1021 pin 10.

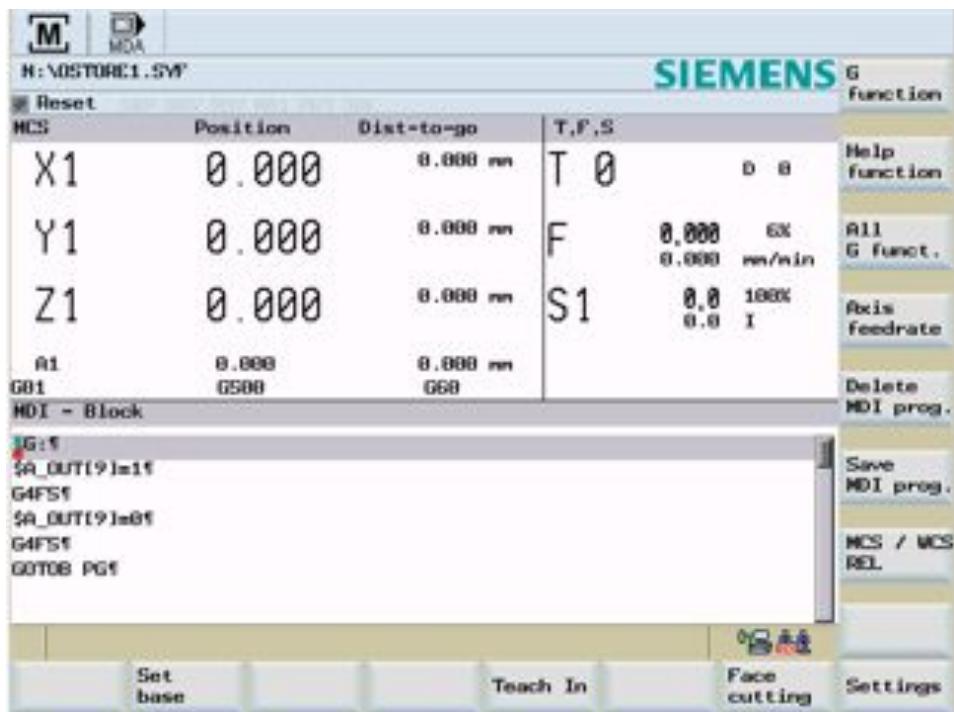
If the voltage was not present it would suggest a faulty MCPA board.

## Section 4

### Digital input and output signal tests.

The state of a digital output can be set with “MDA” mode.

Notes



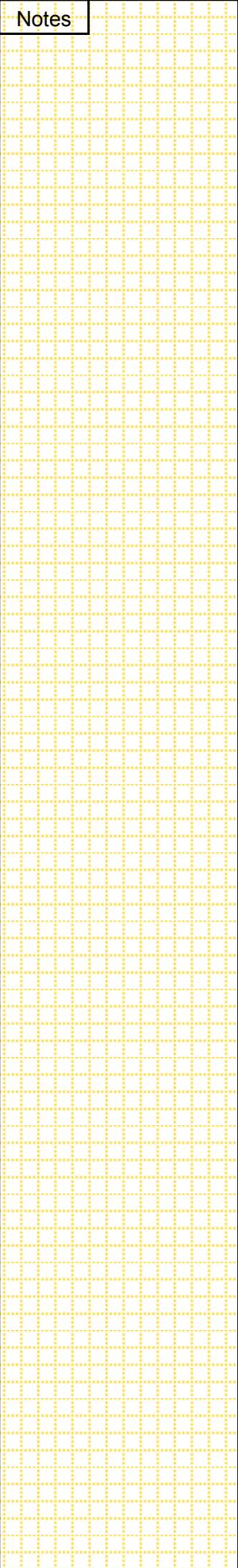
The above program will switch the digital output on X1021 pin 2 on and off every five seconds (G4F5 = 5 second dwell).

This can be checked with a meter.

**Note:-Care should be taken on machinery when switching outputs in this fashion, unwanted movements may occur which can cause damage to the machinery and personnel.**

Blank page

Notes



## 1 Brief description

**Module objective:**

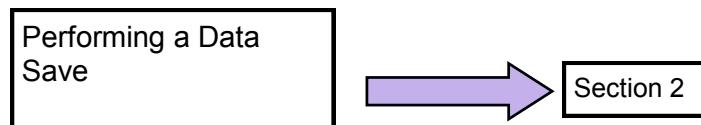
Upon completion of this module you can Save Data which is stored in the volatile data to the internal Flash memory of the CNC Controller.

**Module description:**

The 802D SL controller executes with data which is stored in volatile memory. This volatile memory is retained using a GoldCap capacitor. If the machine is powered off for more than 50 hours, the data which is stored in the volatile memory can be lost. It is therefore necessary to Save this data to Flash memory when the machine is switched off for periods which are longer than 50 hours. When necessary this previously saved data can be reloaded to the volatile memory from the Flash memory. All data excluding the PLC program are saved, the PLC is stored always in non-volatile memory (Flash memory) and will not be lost upon dissipation of the GoldCap.

**Module content:**

Performing a Data Save



## Section 2

### Performing a Save Data

#### 2.1 Performing a Save Data

Notes

To perform a Saving of data, you have to enter the “System Area”. To enter the “System Area” proceed as follows:

On NC Keyboard — Press the key SHIFT + ALARM



To perform a “Save Data” the password “CUSTOMER” has to be set.

SYSTEM

Machine configuration

No.	Axes number	Name	Type
1	1	X1	Linear axis
2	2	Y1	Linear axis
3	3	Z1	Linear axis
4	4	SP	Spindle
5	5	B1	Linear axis

Set password

Change password

Delete password

RCS log-in

Change language

Save data

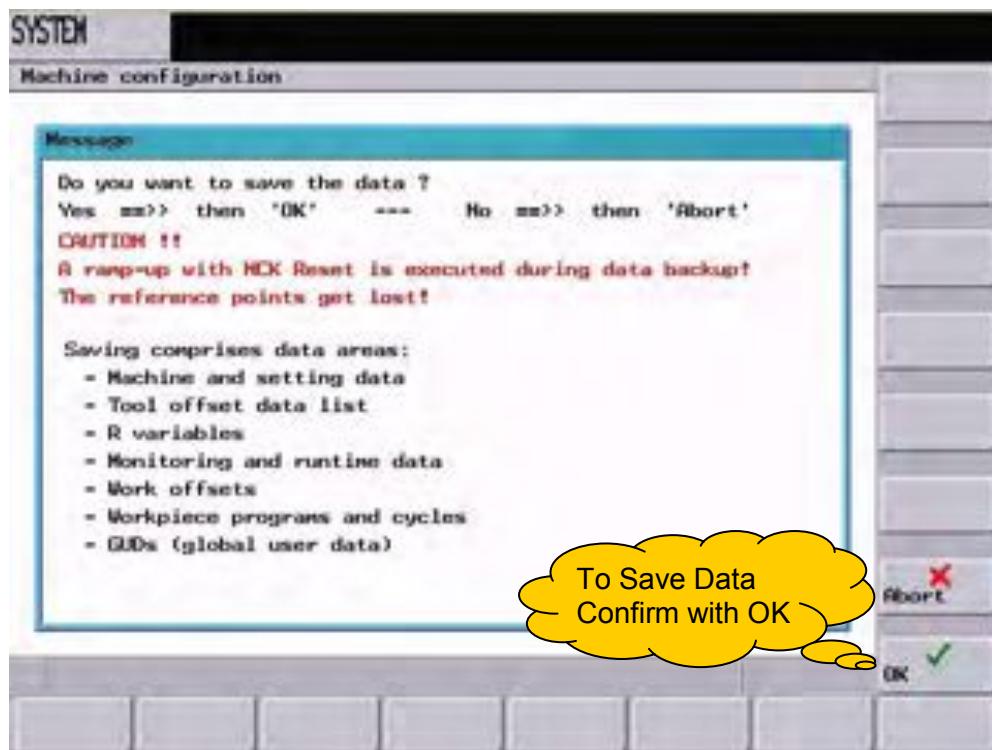
Start-up Machine data Service display PLC Start-up files

## Section 2

### Performing a Save Data

Confirm with OK to save data..

Notes



It is also possible to see in the above picture, which data will be saved.



## 1 Brief description

**Module objective:**

Upon completion of this module you can restore the data which is stored in the flash memory to the internal SRAM (active, volatile) memory of the CNC controller.

**See hand book module C17 Save data**

**Module description:**

The 802D SL controller executes with data which is stored in volatile memory. This volatile memory is retained using a GoldCap capacitor. If the machine is powered off for more than 50 hours, the data which is stored in the volatile memory can be lost. It is therefore necessary to Save this data to Flash memory when the machine is switched off for periods which are longer than 50 hours. When necessary this previously saved data can be reloaded to the volatile memory from the Flash memory. All data excluding the PLC program are saved, the PLC is stored always in non-volatile memory (Flash memory) and will not be lost upon dissipation of the GoldCap.

This module describes the methods of restoring the previously saved data.

**See hand book module C17 Save data**

**Module content:**

Performing a Restore of saved data

Performing a Restore  
of saved data



Section 2

## Section 2

### Restoring Saved Data

#### 2.1 Restoring Saved Data

Notes

The “Saved data ” can be restored in several ways:

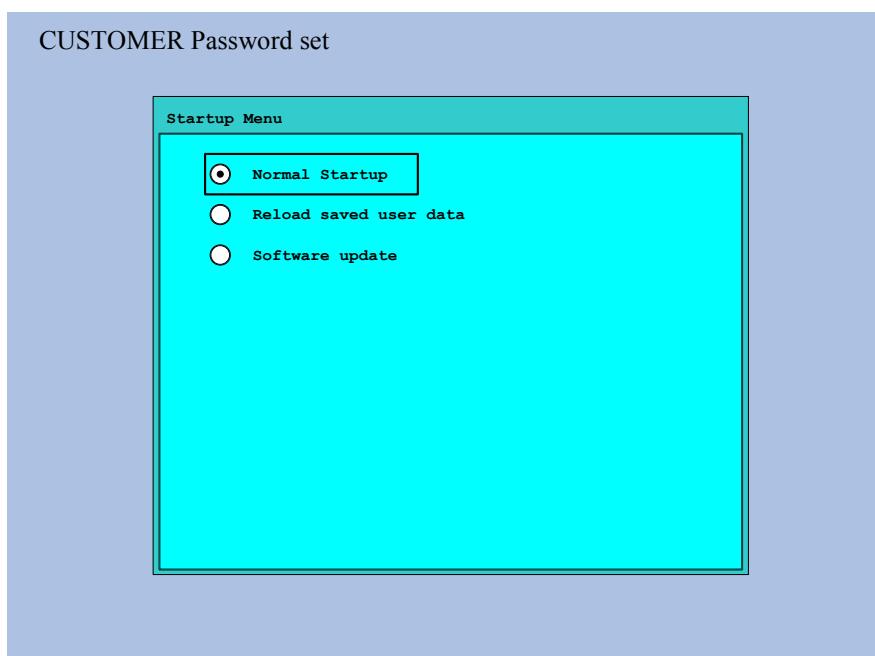
SELECT picture at Startup  
Automatic (in case of data loss)

The system can detect loss of data, upon data loss the saved data will be loaded into the volatile memory automatically.

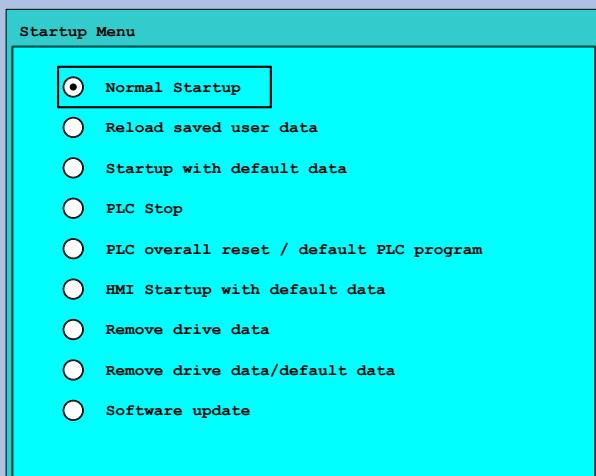
Saved data can be restored by the user at any time by pressing the SELECT key when prompted at Startup time of the control.

The following picture then appears—one of the two will appear dependant upon which password is activated.

The alarm 004062 indicates to the operator that “saved data” has been loaded, this alarm can be acknowledged with the reset key.



#### EVENING—SUNRISE Password set



## 1 Brief description

**Module objective:**

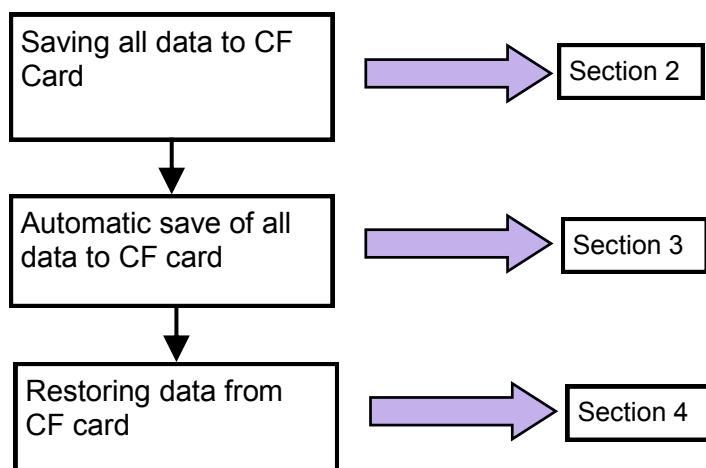
Upon completion of this module you can save to a Compact flash card, a complete backup of the control.

**Module description:**

Due to component defect, it may become necessary to exchange the CNC system. In such a case the “Saved data” function is not of any use. It becomes necessary to previously store a complete backup of the controls machine specific data. Saving the data is described in this module and also loading the data back to a replacement control.

**Module content:**

Saving all data to CF Card  
Automatic save of all data to CF card  
Restoring data from CF card



## Section 2

### Saving All data to Compact Flash card

#### 2.1 Saving All data to Compact Flash card

Notes

All data can be saved to a CF card using Copy and Paste functionality. To use the feature a CF card should be inserted into the controller, as can be seen below:

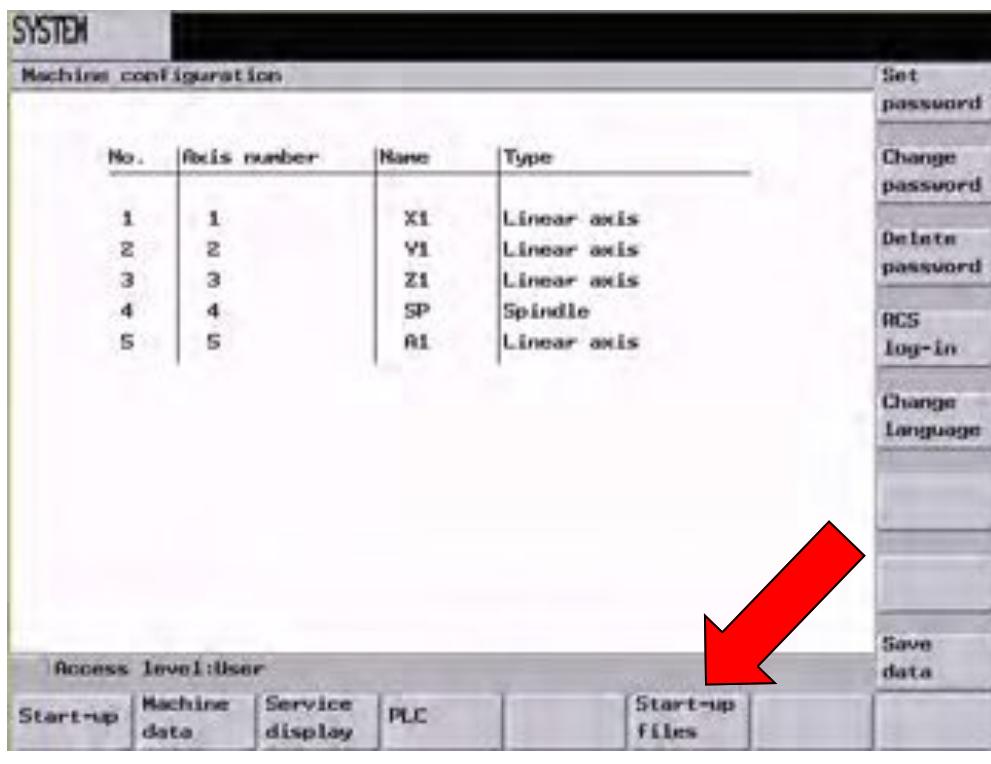


The CF card can be up to 2GB, and should be formatted using the FAT16 or FAT32 file system.

To make a backup of the machine configuration, that the controller can be restored to its original state requires that two archives should be saved to the CF card. These two files can be saved to the card in the SYSTEM area of the control. i.e. NC/PLC archive and HMI archive

The SYSTEM area of the control is reached by pressing the key : SHIFT + ALARM.

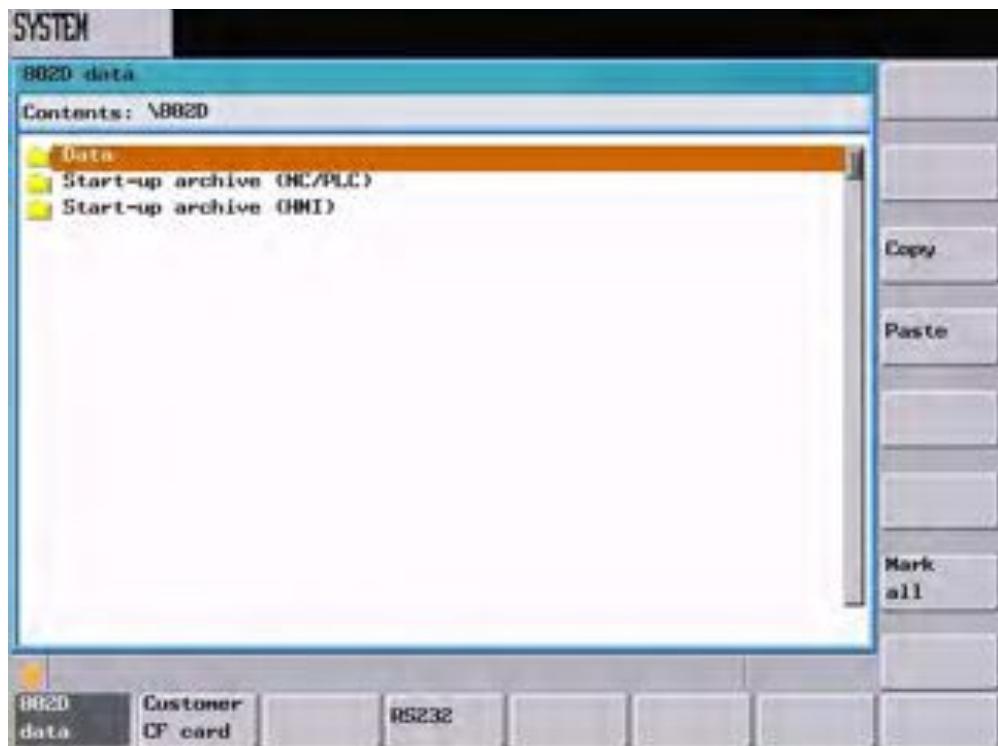
The entry point for saving data is through softkey Start-up files, see picture below:



## Section 2

### Saving all data to compact flash card

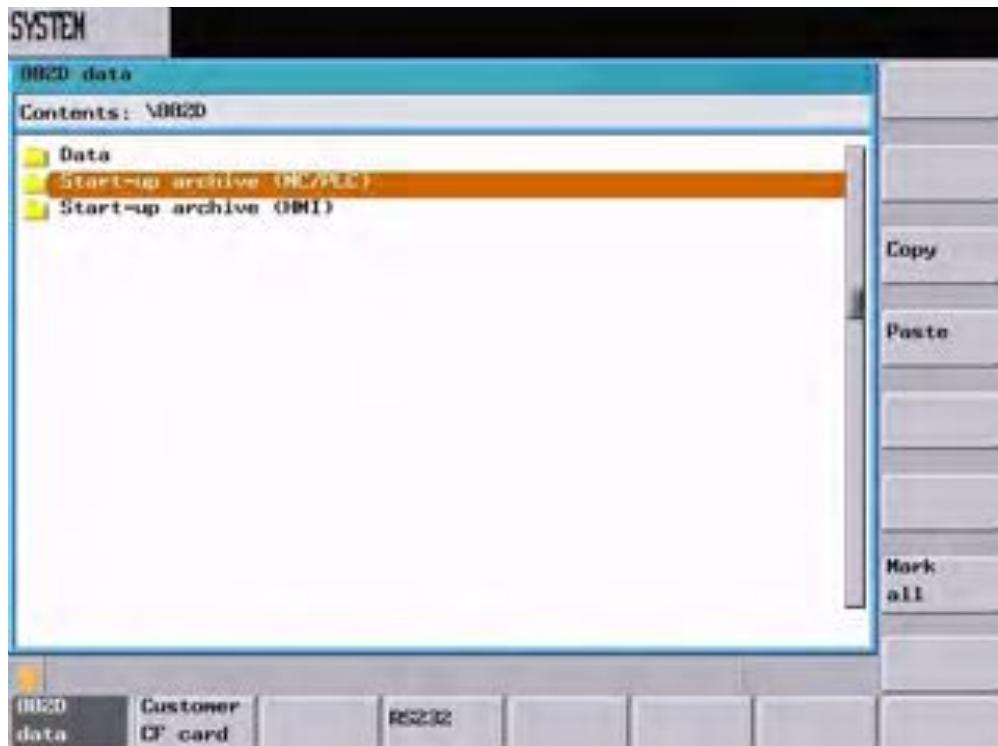
The following display can then be seen:



Notes

The two files previously mentioned can be saved by copying the contents of the HMI and the NC/PLC directories into two archives.

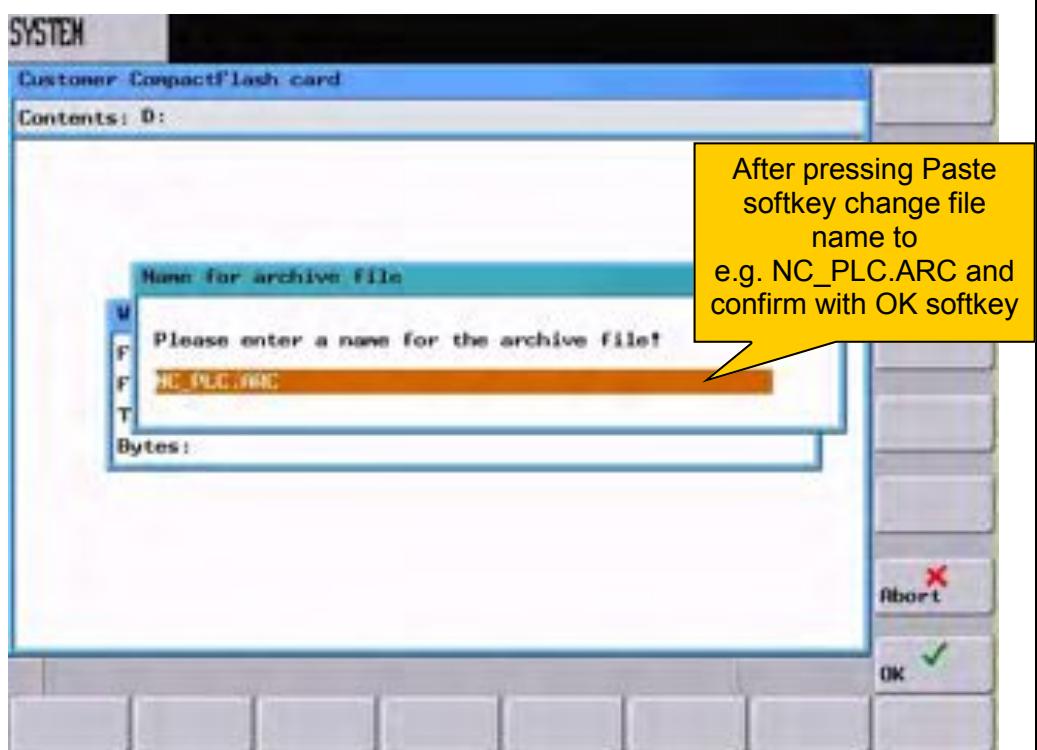
The following picture describes this process:



The destination should now be selected and the Paste function can be used to save the files to the respective card. The destination is the "Customer CF card"

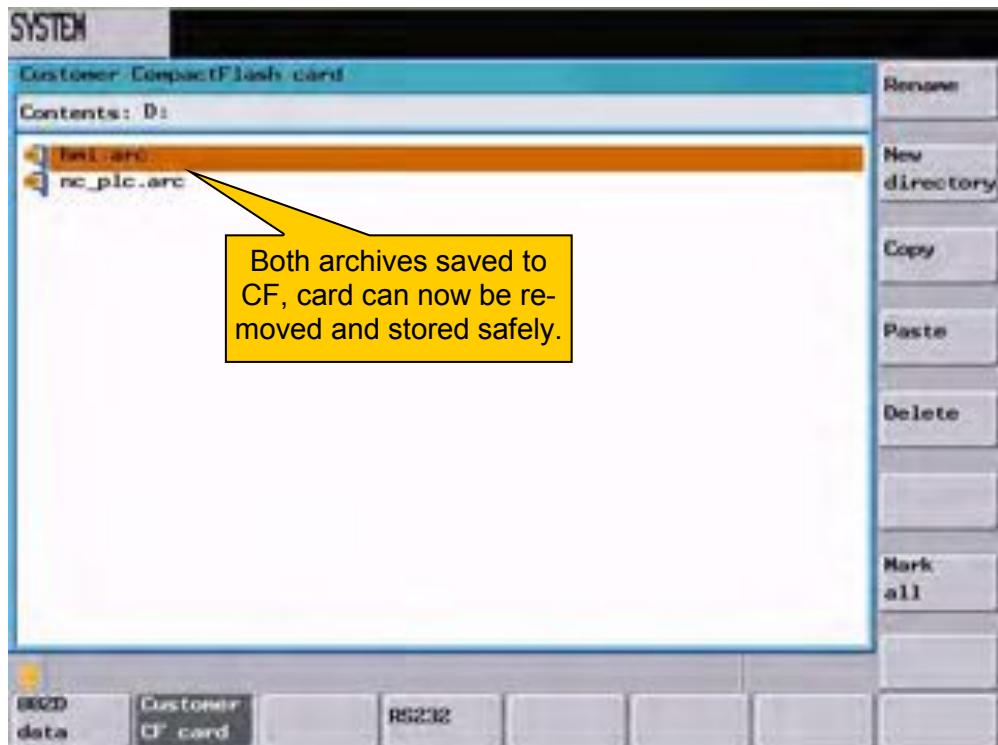
## Section 2

### Saving all data to compact flash card



Notes

The above should be repeated for the HMI archive.  
The two archives should be now visible on the CF card. The CF card  
should now be removed from the system and stored safely.  
To reflect the newest changes this process should be carried out on a  
regular basis.



## Section 3

### Automatic save of all data to CF card

#### 3.1 Automatic save of all data to CF card

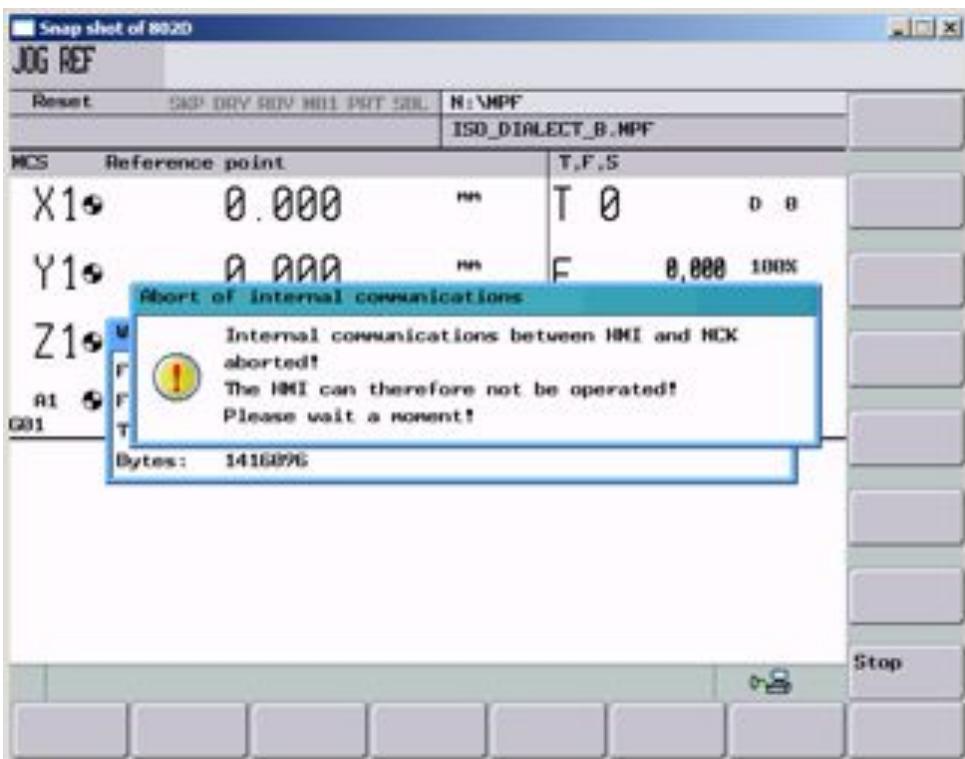
Complete save of all system data can also be implemented by pressing [CTRL] + [S] at any time. This will automatically create a complete archive of all NC & PLC data, and HMI data in two archive files and write these files onto the CF Card. The archives are created with default names.

**802dslibnhmi.arc**  
**802dslibnnc.arc**

The file creation will overwrite any existing files of that name without requiring confirmation.

The only pre-requisite for this procedure is that the protection level password must be set to a minimum of level 3 (User level)

After the file creation procedure is complete, an NCK reset is performed.



DO NOT remove the CF card until this is completed, otherwise system data corruption will occur.

Notes

## Section 4

### Restoring All data from Compact flash card

#### 4.1 Restoring All data from Compact flash card

Notes

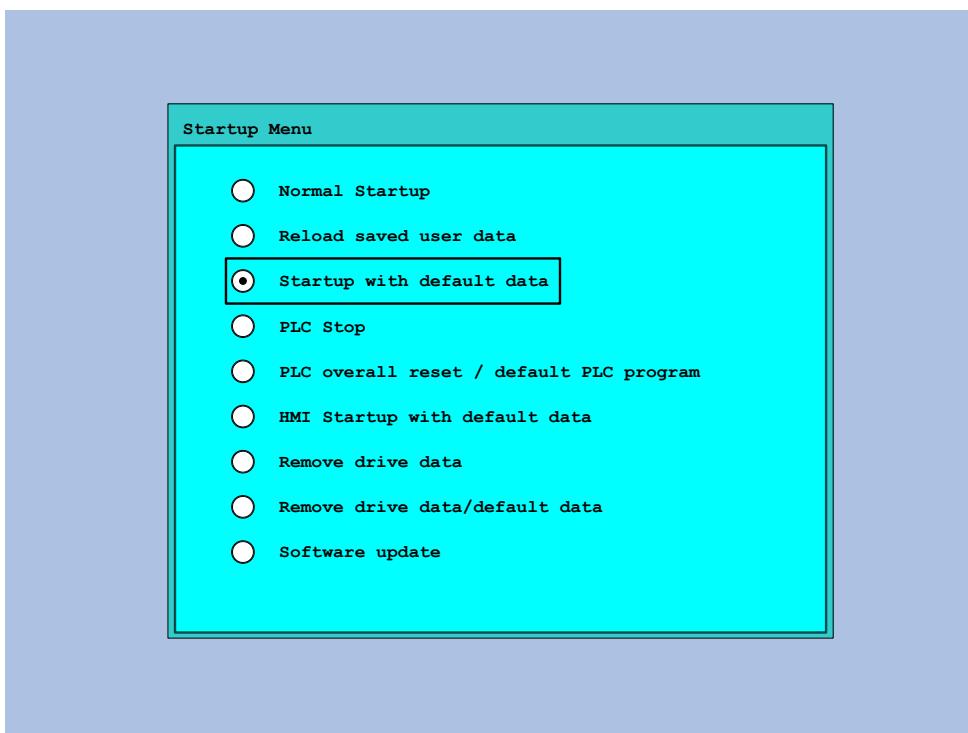
In the case of component exchange it will be necessary to reload the previously saved archives to achieve a functional machine.

The following steps should be carried out in order to reload the data previously stored on the CF card.

The controller should first be switched on and the password for the SERVICE should be set: The password for a new CNC component is always the default “EVENING”, this should be changed after new installation to prevent mishandling of the system in the future.

Once the password is set the system should be switched OFF and ON and the Startup menu entered with the SELECT key when prompted.

See picture below:



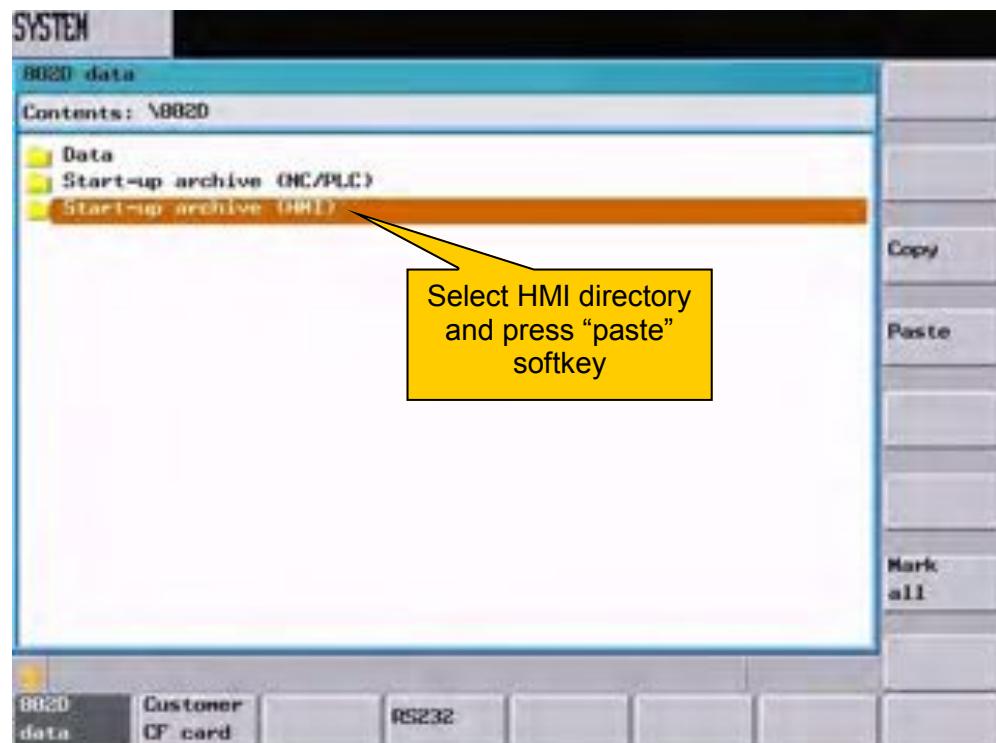
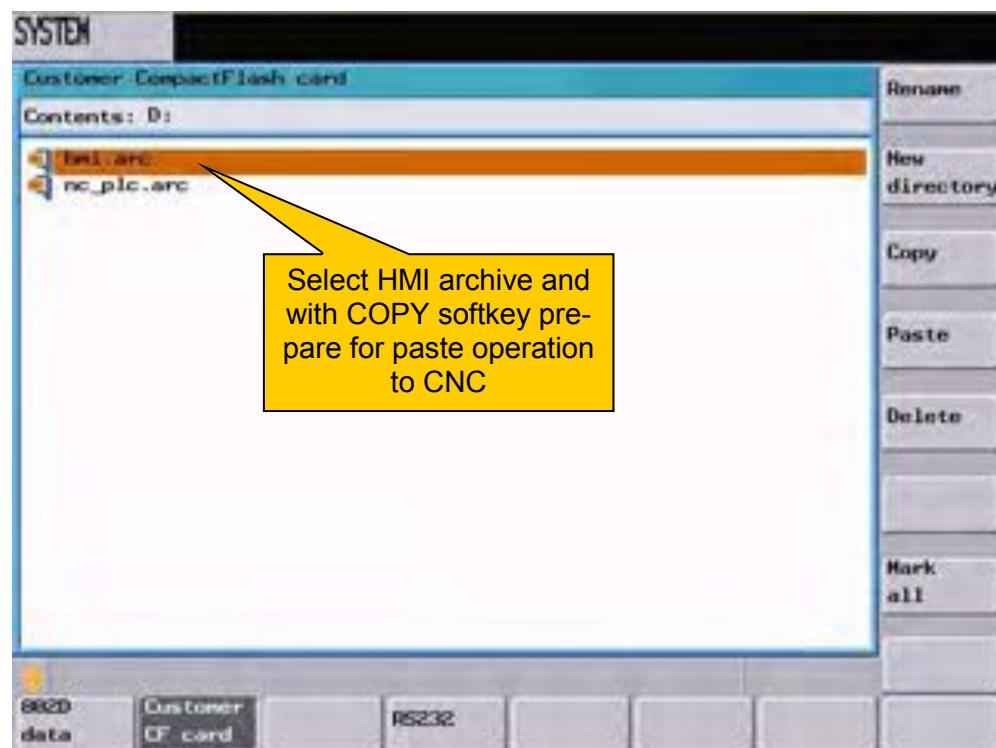
Choose menu option “Startup with default data” and confirm with the ENTER key.

The control will now re-start, wait until the control is finished starting up and once again enter the SERVICE password.

You should see the message “004060 Standard machine data is loaded”. Using copy from CF card and paste to CNC the backup data can be restored to the control. See the following diagrams:

## Section 4

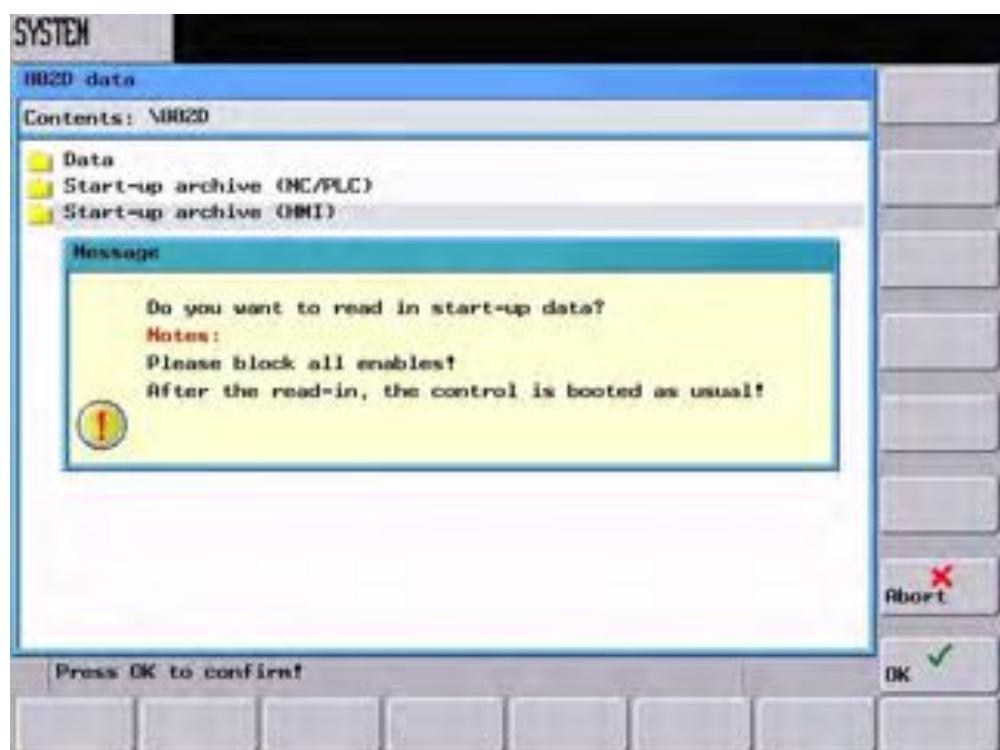
### Restoring All data from Compact flash card



Notes

## Section 4

### Restoring All data from Compact flash card



Confirm the dialog box with the OK softkey, the HMI archive will now be read into the controller.

During this reading in process the controller will reboot once, the process will take approximately 1 minute to complete.

This procedure should be repeated for the NC/PLC archive, the CNC will reboot more times and the process takes approximately 4 minutes to complete. The NC\_PLC archive should be copied to the NC PLC directory and not the HMI.

Upon completion of this process the machine should be functional again.

Note: The HMI archive should be read in first because the specific PLC alarms are in the NC/PLC archive.

## 1 Brief description

### Module objective:

Upon completion of this module you can use the built-in PLC program displays to diagnose faults.

### Module description:

The 802D SL controller has extensive diagnostic possibilities, one of which is the online display of the PLC program.

It is possible using this function to quickly evaluate logic faults or external switching faults on the machine. To use this function effectively previous knowledge of PLC programming is advantageous.

### Module content:

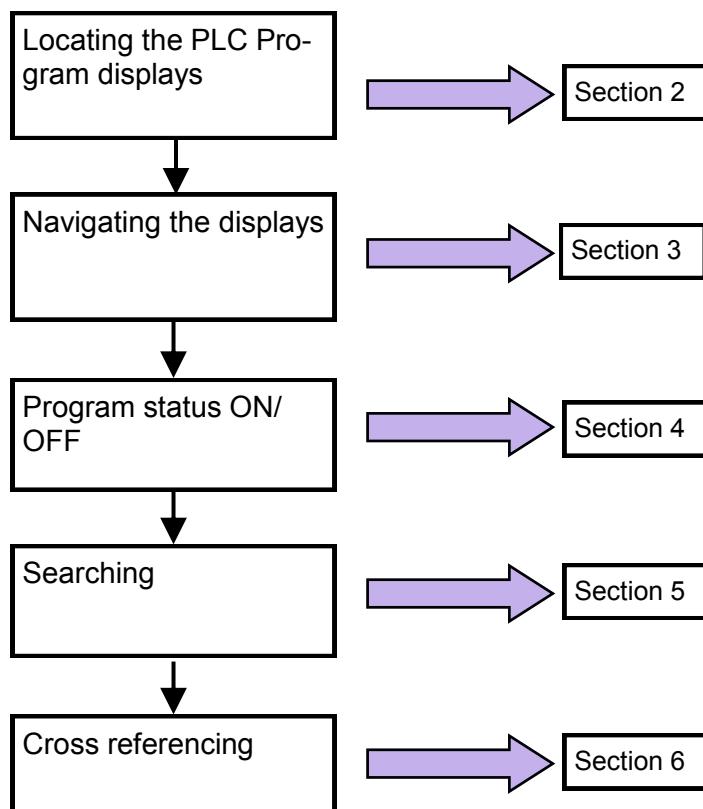
Locating the PLC Program displays

Navigating the displays

Program status ON/OFF

Searching

Cross referencing



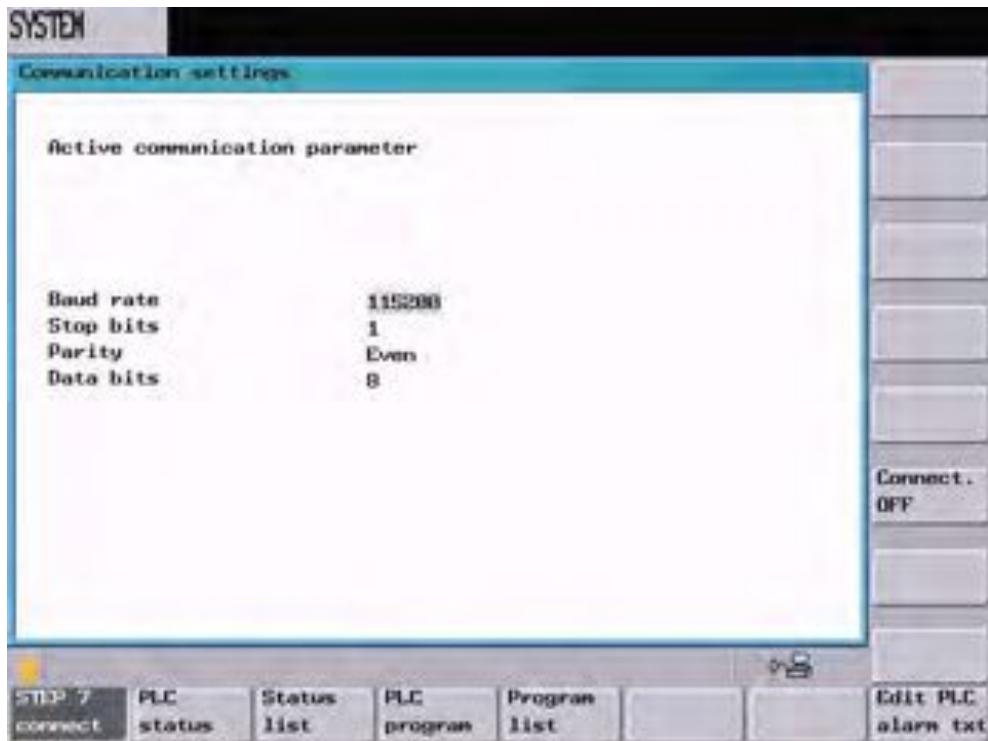
## Section 2

### Locating the PLC Program displays

#### 2.1 Locating the PLC Program displays

The PLC Program display can be found in the “SYSTEM area”. The displays are only available when the “SERVICE” password is set or higher. To enter the “SYSTEM” area press the key “SHIFT + ALARM”.

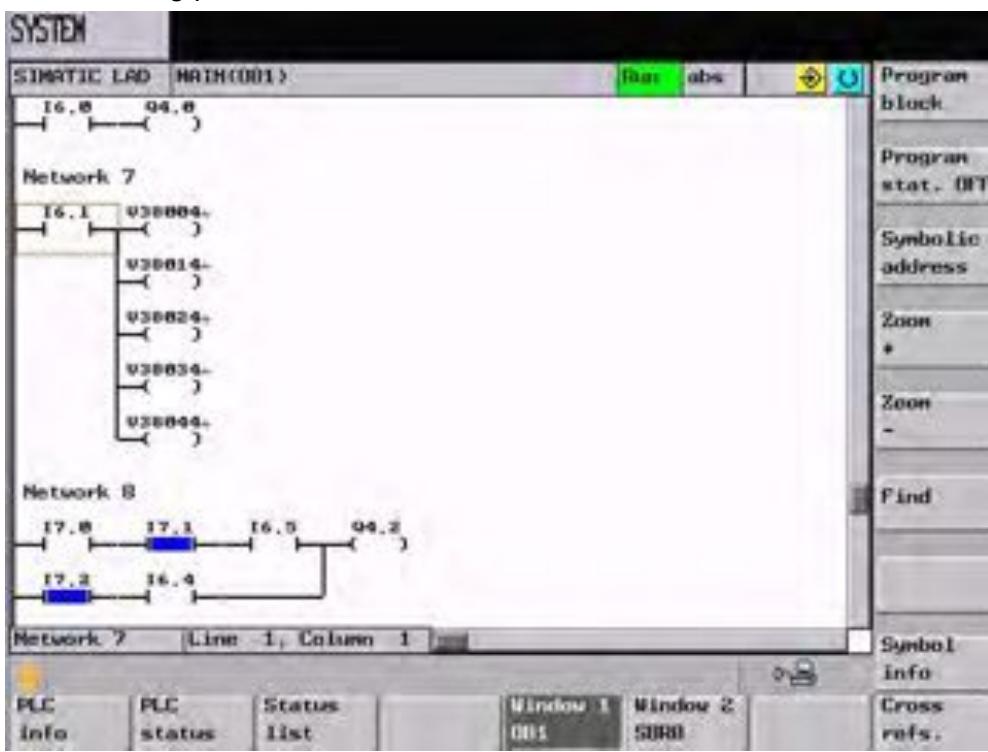
In the system area you will see the softkey PLC, upon pressing the softkey PLC the following picture is displayed:



When entering the PLC area, the default for the control is to open the “Program list” picture. This function is explained in the hand book module C33.

The PLC Program displays can be found by pressing the softkey “PLC Program”

The following picture can be seen:



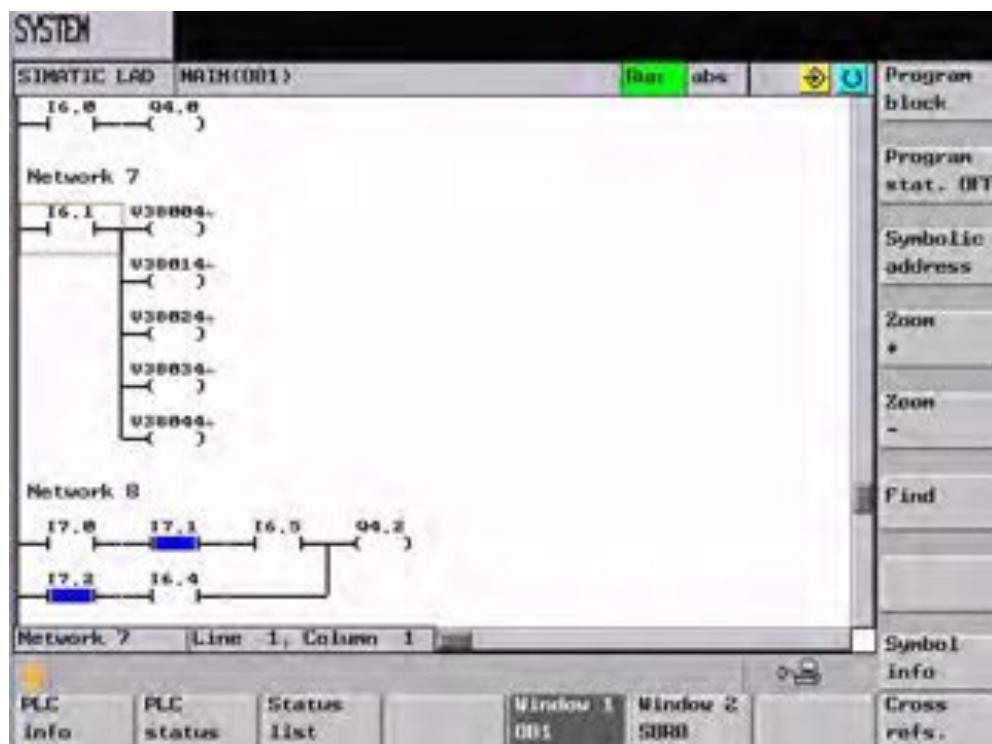
Notes

## Section 3

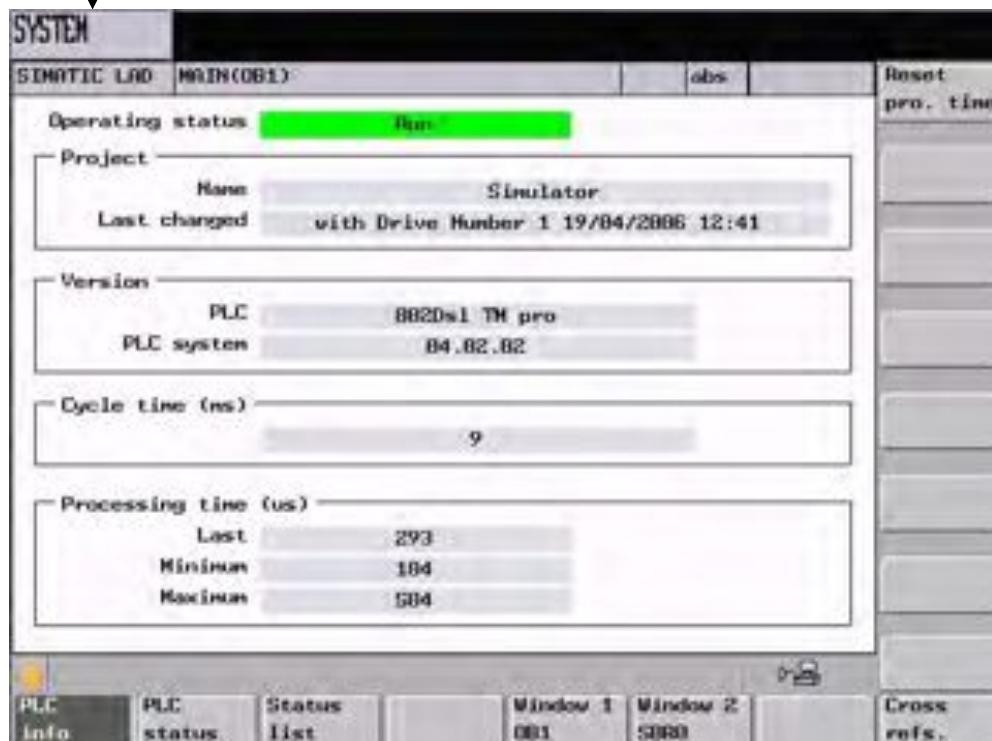
### Navigating the displays

#### 3.1 Navigating the displays

Notes



**PLC info** Project name, Version , Cycle time, Processing time.

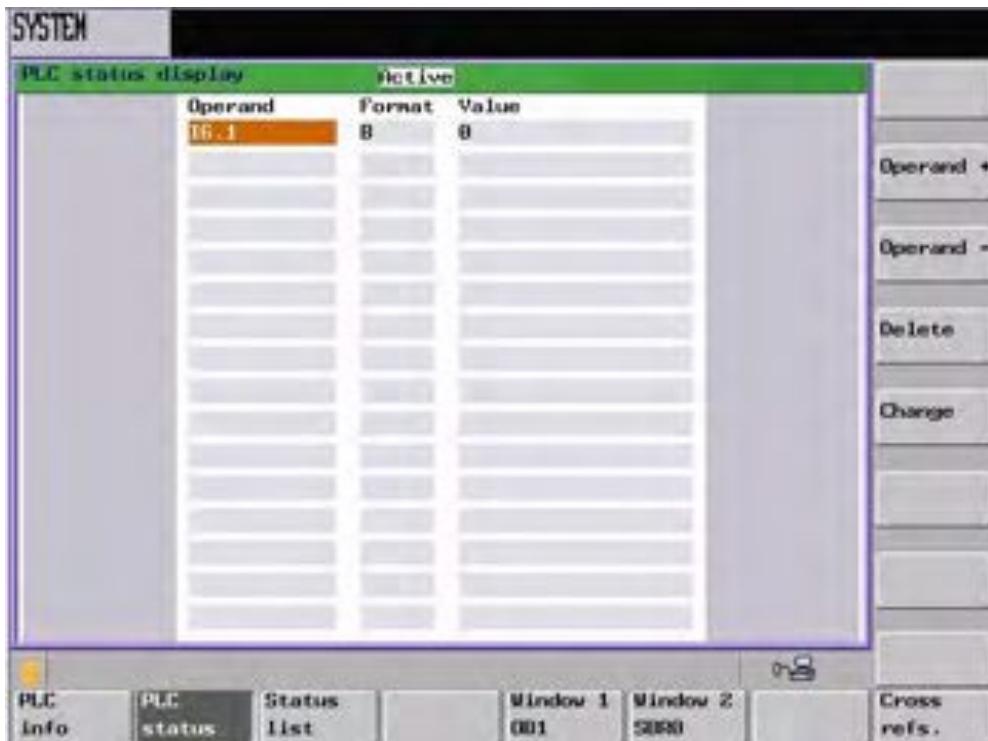


## Section 3

# **Navigating the displays**

**PLC  
status**

Status >> Bits, Bytes, Words, DWords with the possibility to change



## Notes

## Status list

Overview Input, Output, Marker area with the possibility to change



SINUMERIK 802D sl Operating and Service Training Manual

Page 4

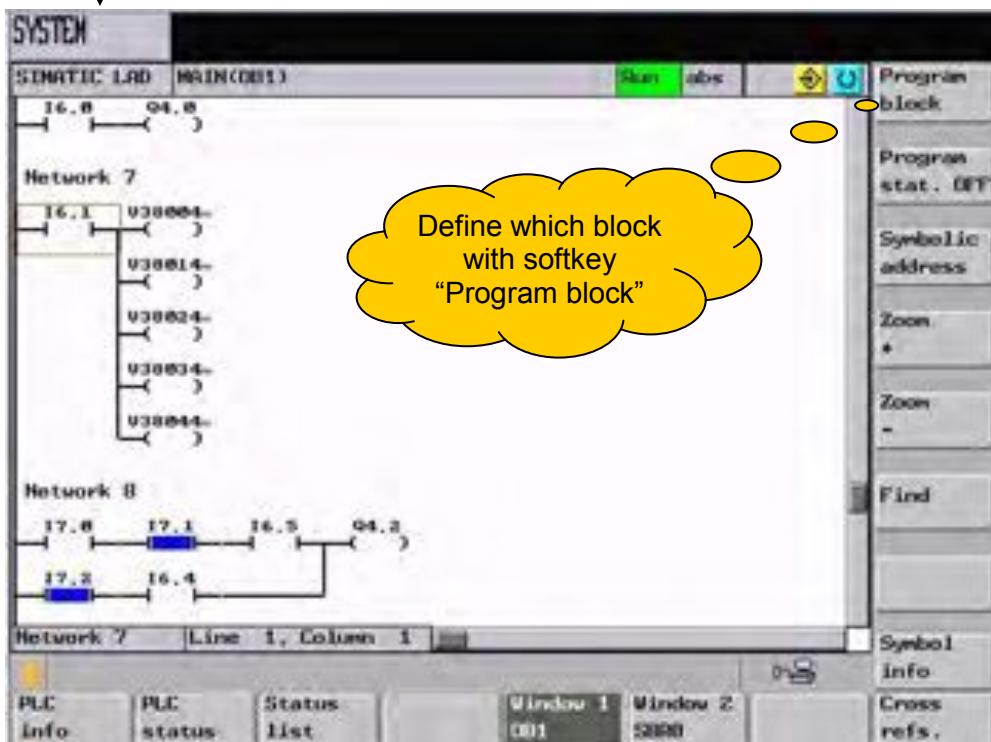
## Section 3

### Navigating the displays

**Window 1**  
OB1

Status >> logic online/offline

↓



Define which block  
with softkey  
"Program block"

Notes

**Window 2**  
SBR0

Status >> logic online/offline as Window 1  
Define which block with softkey "Program block"

↓



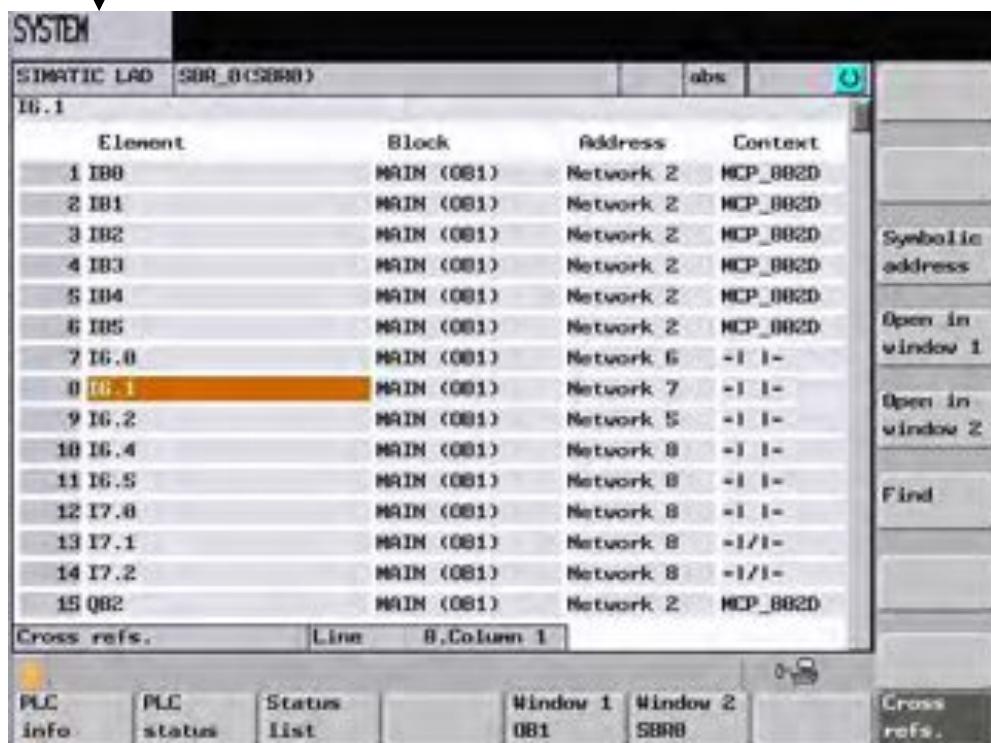
## Section 3

### Navigating the displays

Cross  
refs.

Locator >> Where is it used? And as what?

Notes

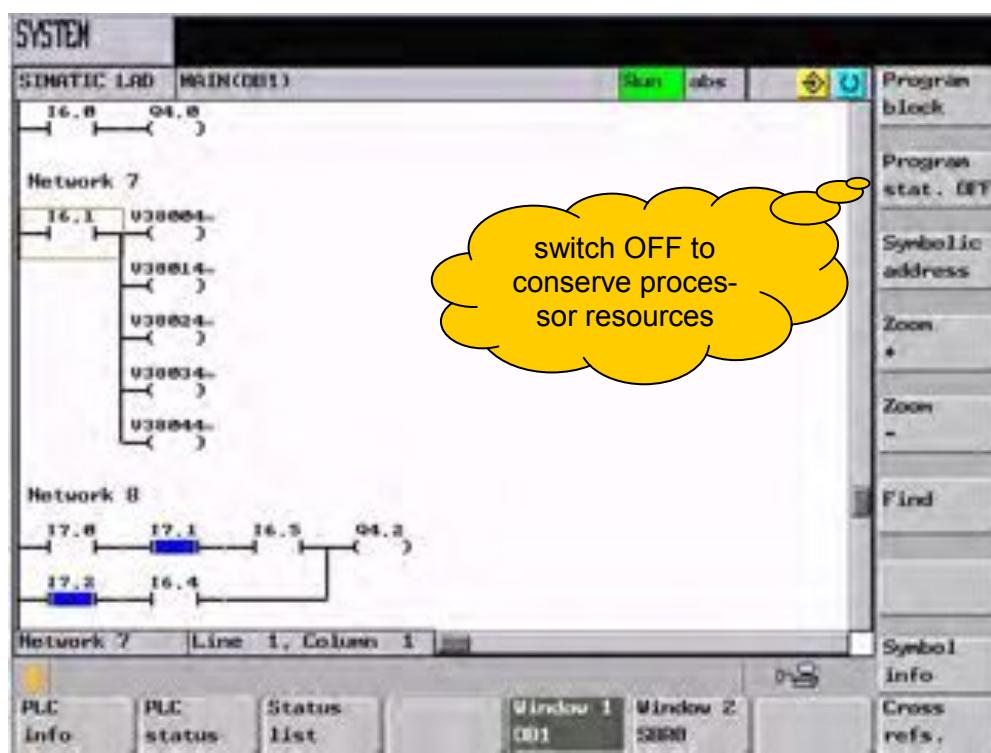


## Section 4

### Program status ON/OFF

#### 4.1 Program status ON/OFF

With Program status ON, a real-time picture can be seen of the logic controlling the machine.



## Section 5

### Searching

#### 5.1 Searching

Search / Find: the function can be used context sensitive to find operands or symbols located in the body of the PLC program or cross reference table.

Notes

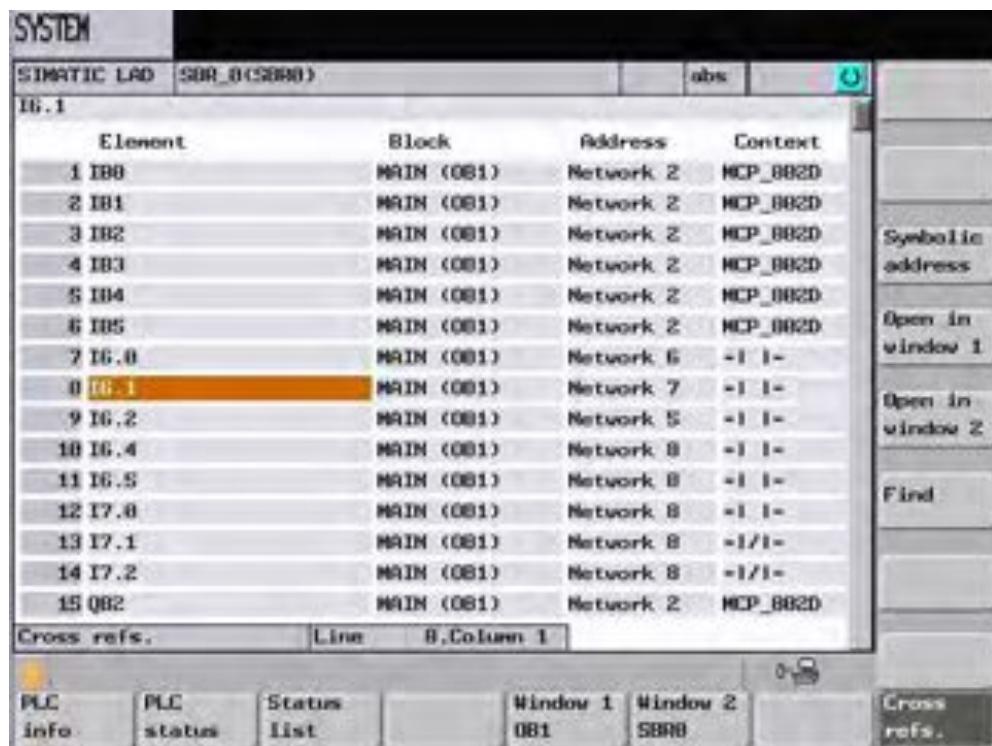


## Section 6

### Cross referencing

#### 6.1 Cross referencing

Find all occurrences of operands in the PLC program





## 1 Brief description

**Module objective:**

Upon completion of this module you can use the built-in PLC program displays to diagnose a PLC generated fault.

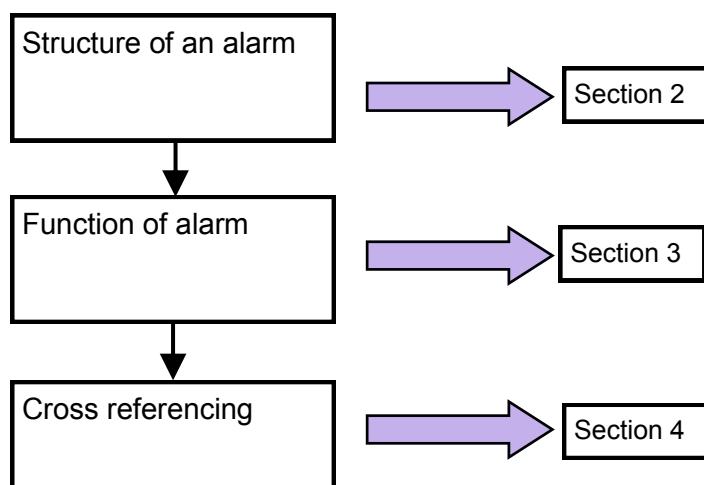
**Module description:**

The 802D SL controller has extensive diagnostic possibilities, one of which is the online display of the PLC program.

It is possible using this function to quickly find the source of a PLC generated fault, and quickly resolve an external fault on the machine. e.g. faulty sensor.

**Module content:**

Structure of an alarm  
Function of alarm  
Cross referencing



## Section 2

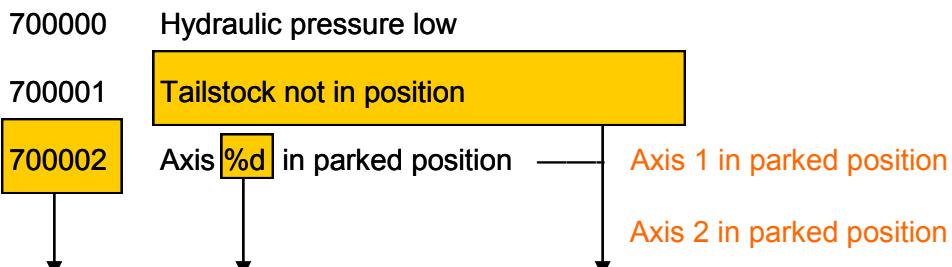
### Structure of an alarm

#### 2.1 Structure of an alarm

To identify external faults from the generated PLC fault, it is necessary to understand the structure of the alarm.

Notes

##### Structure:



Alarm Num- Variable = 1      Alarm Text

Alarm Number	PLC Signal	PLC variable	NC MD alarm configuration	Text File
700000	16000000.0	16001000	14516[0]	Alarm 1
700001	16000000.1	16001004	14516[1]	Alarm 2
700002	16000000.2	16001008	14516[2]	Alarm 3
700003	16000000.3	16001012	14516[3]	Alarm 4
700004	16000000.4	16001016	14516[4]	Alarm 5
700005	16000000.5	16001020	14516[5]	Alarm 6
700006	16000000.6	16001024	14516[6]	Alarm 7
700007	16000000.7	16001028	14516[7]	Alarm 8
700008	16000001.0	16001032	14516[8]	Alarm 9
700009	16000001.1	16001036	14516[9]	Alarm 10
700010	16000001.1	16001040	14516[10]	Alarm 11
700011	16000001.1	16001044	14516[11]	Alarm 12
700012	16000001.1	16001048	14516[12]	Alarm 13
..	..	..	..	..
..	..	..	..	..
..	..	..	..	..
..	..	..	..	..
700063	16000007.7	16001252	14516[63]	Alarm 63

##### Variable definition

- %d Decimal value displayed
- %x Hexadecimal value displayed
- %b Binary value displayed
- %o Octal value is displayed
- %u Unsigned integer is displayed
- %f Floating point number is displayed

##### PLC Signal

The alarm 700000 is generated with the corresponding PLC signal, the reaction from the NC is defined in the NC machine data 14516[index].

## Section 3

### Function of alarm

#### 3.1 Function of alarm

Notes

Alarms are generated for a purpose, the purpose is primarily to protect the machine operator in the case of a dangerous situation and secondly to protect the machine from damage.

To cater for the above two cases we can react in two ways:

- 1) Pure PLC reaction
- 2) PLC >> NC reaction

The first is to use the alarm to warn the operator of a problem, e.g. coolant is low in the machine. In such a case the axis can be prevented from moving by various disable signals in the PLC interface.

The second case is to configure the alarm in such a way that the NC can react itself. e.g. in the case of an alarm NC start can be automatically inhibited.

The possibilities are listed below:

MD 14516[index]	Meaning
Bit 0	NC Start inhibit
Bit 1	Read-in disable
Bit 2	Feed disable for all axis
Bit 3	Emergency stop
Bit 4	PLC Stop
Bit 5	
Bit 6	Cancel with Delete key
Bit 7	Power on

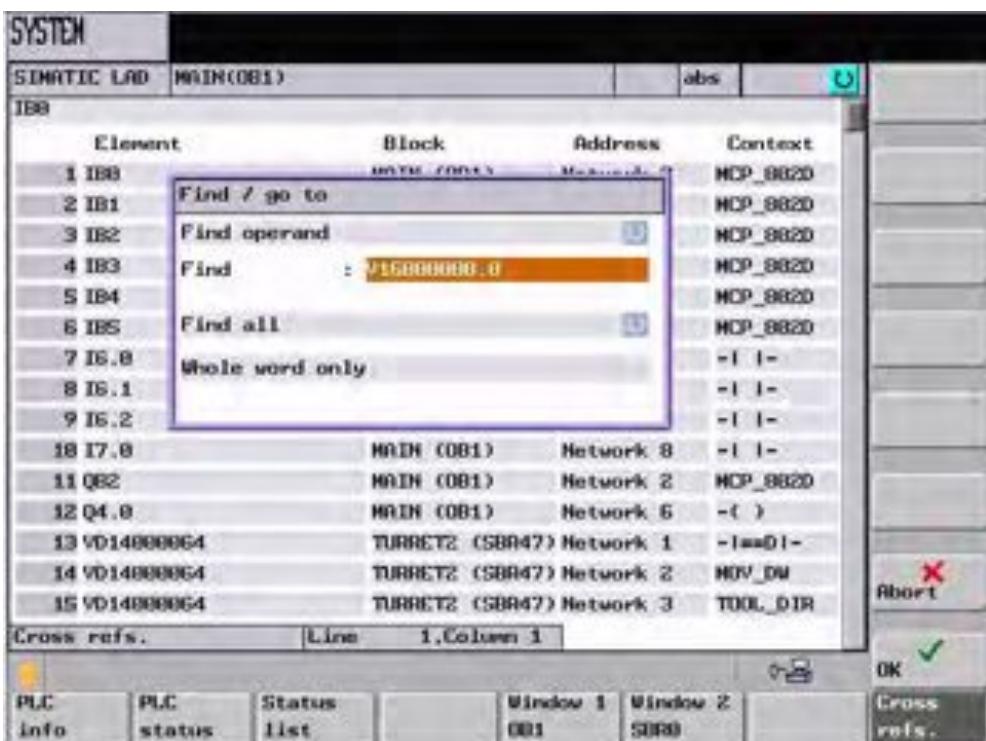
## Section 4

### Cross reference

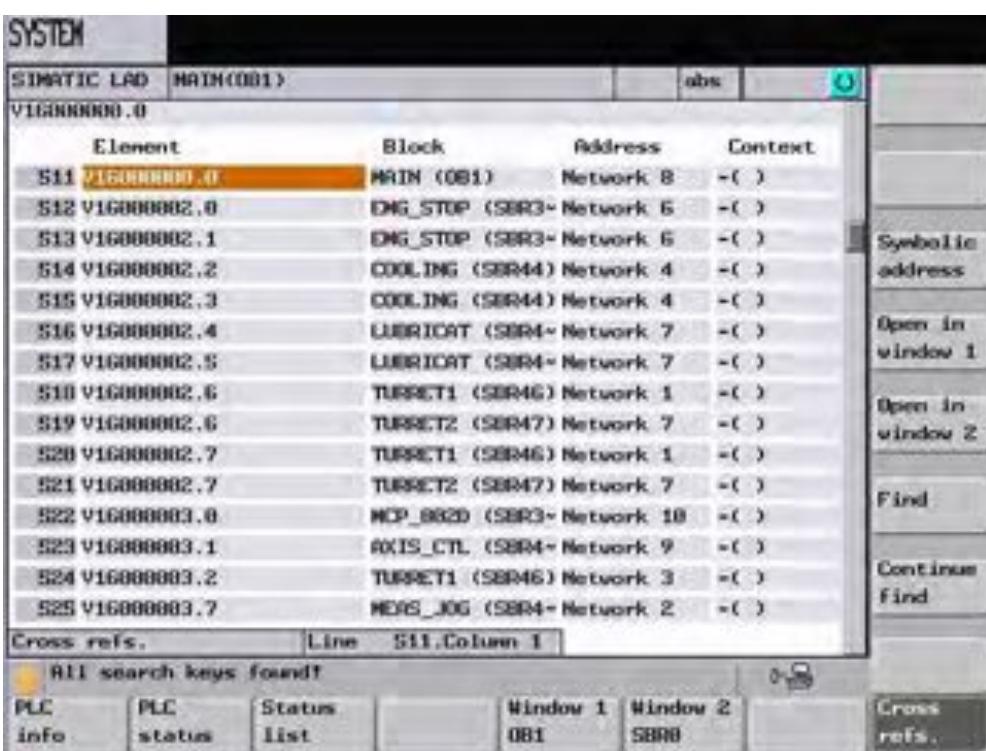
#### 4.1 Cross reference

It may become necessary to track an alarm to its source. A logic condition generates the alarm, the logic condition has to be found in the PLC program and diagnosed using status. To find the logic, the cross reference function should be used.

See the following picture:



#### Result



## 1 Brief description

**Module objective:**

Upon completion of this module you will be able to identify an NC alarm.

**Module description:**

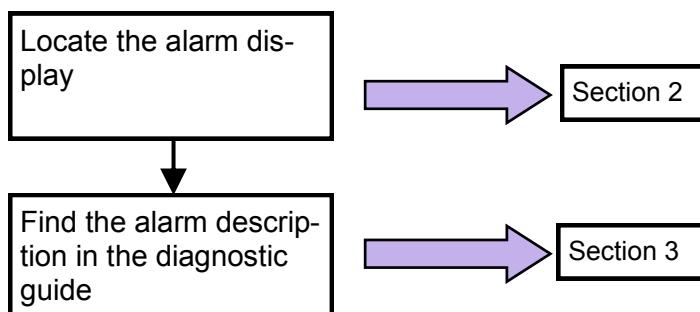
The 802D SL controller has extensive diagnostic possibilities, one of which is the display of the active NC alarm.

With this knowledge you can find in the diagnostic guide, and interpret the fault.

**Module content:**

Locate the alarm display

Find the alarm description in the diagnostic guide

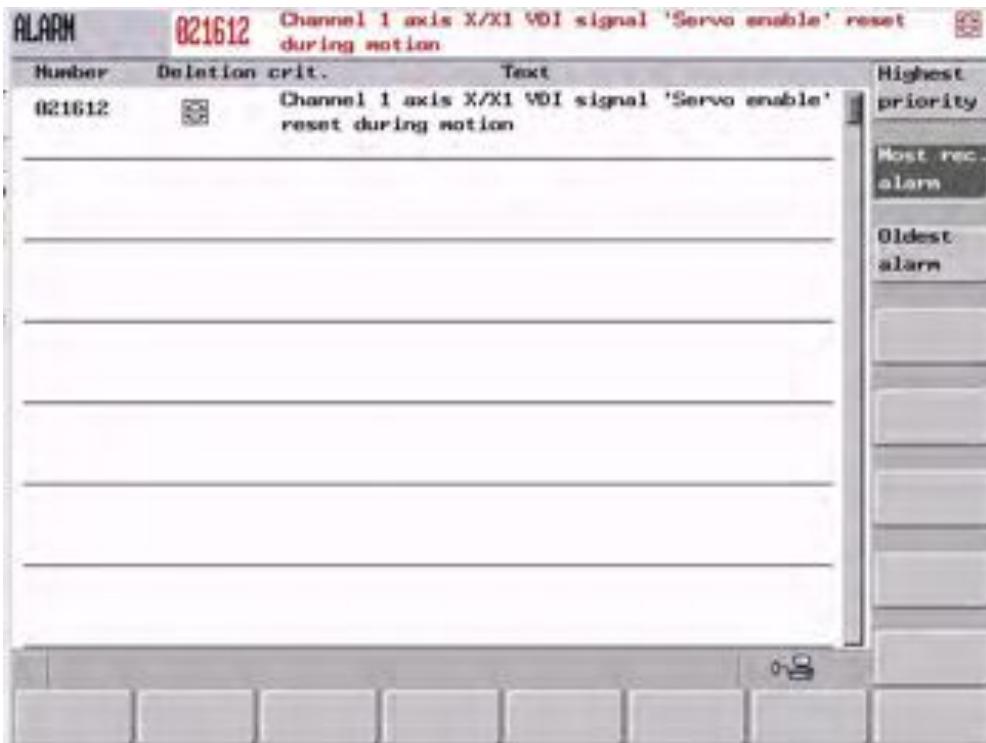


## Section 2

### Locate the alarm display

#### 2.1 Locate the alarm display

The alarm display has to be selected in order to interpret the active alarm. The display can be selected with the ALARM key on the NC keyboard. The following display will be seen :



Notes

The alarm number determines whether an alarm is an NC alarm, the number range can be seen in the following table:

000 000 - 009 999	General alarms	
010 000 - 019 999	Channel alarms	
020 000 - 029 999	Axis/spindle alarms	
030 000 - 099 999	Functional alarms	
060 000 - 064 999	SIEMENS cycle alarms	
065 000 - 069 999	User cycle alarms	

## Section 3

### Find the description in the diagnostic guide

#### 3.1 Find the description in the diagnostic guide

A description of the alarm can be found in the 802D sl Diagnostic Guide.

## 1 Brief description

**Module objective:**

Upon completion of this module you can locate the NC machine data, set the password and make changes.

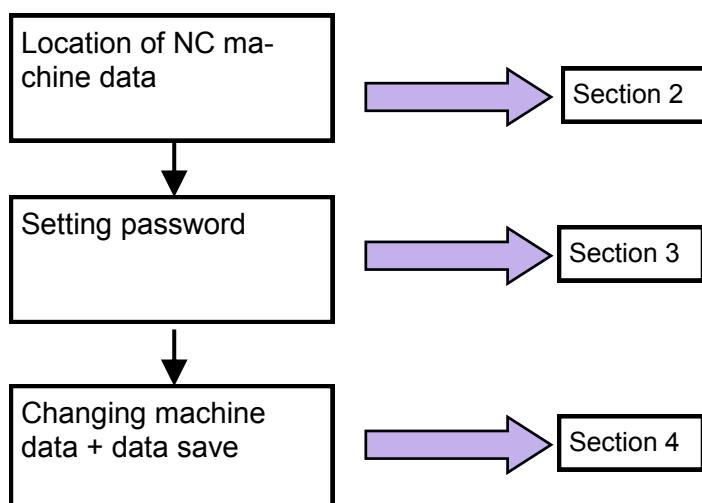
**Module description:**

During the life of a machine it may become necessary to change the values of certain NC machine data. The necessity arises e.g. due to wear on mechanical components, the axis performance can therefore deteriorate. In some cases, further mechanical damage can be prevented with the help of the NC machine data.

A change should only be carried out in conjunction with recommendations from the machine tool builder.

**Module content:**

Location of NC machine data  
Setting password  
Changing machine data + data save

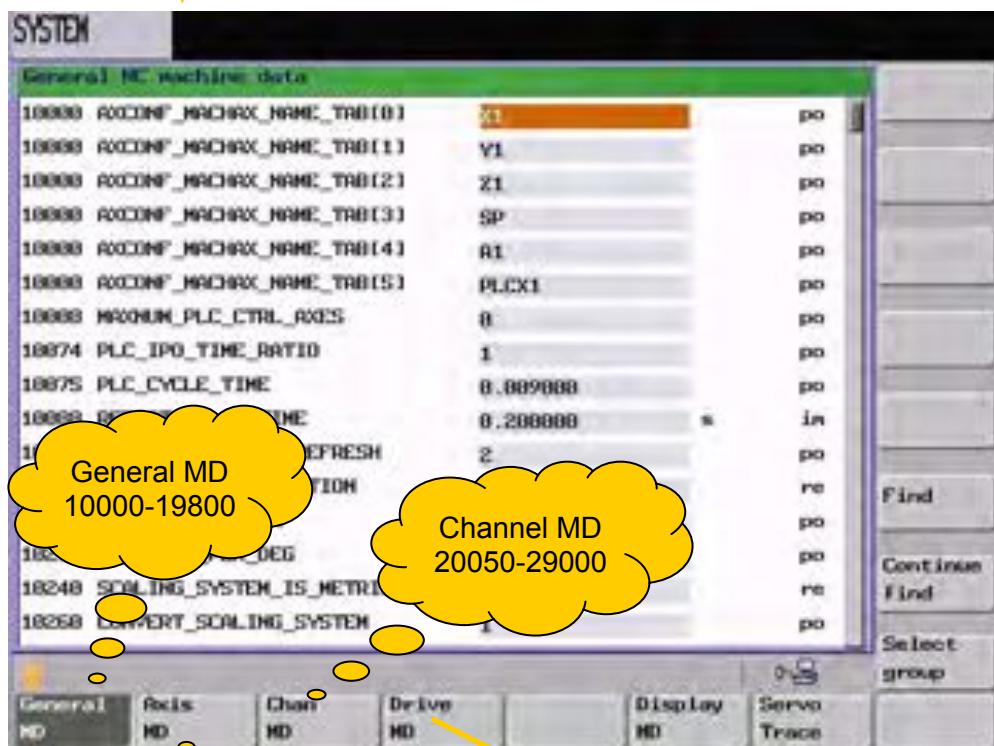
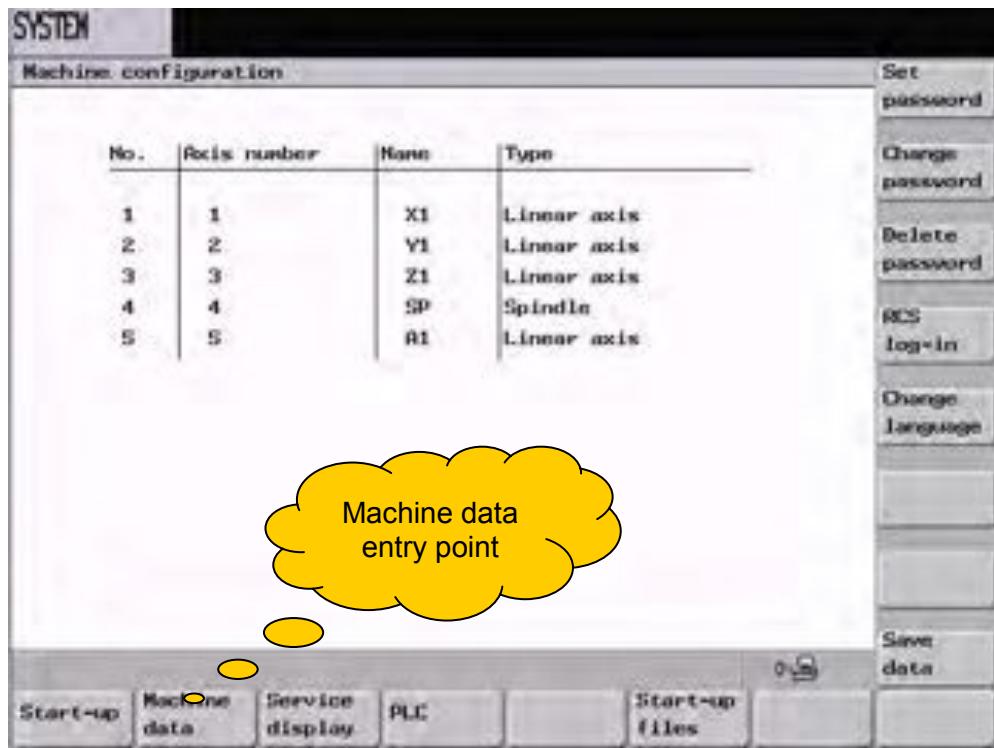


## Section 2

## **Location of NC machine data**

## 2.1 Location of NC machine data

The NC machine data is located in the System area. The system area can be entered with the key combination "SHIFT + ALARM". The following describes graphically how to enter the Machine data area:



General MD  
10000-19800

Channel MD  
20050-29000

Axis MD  
30100-38000

Note:  
Drive machine data should  
only be changed under  
supervision from experts

## Notes

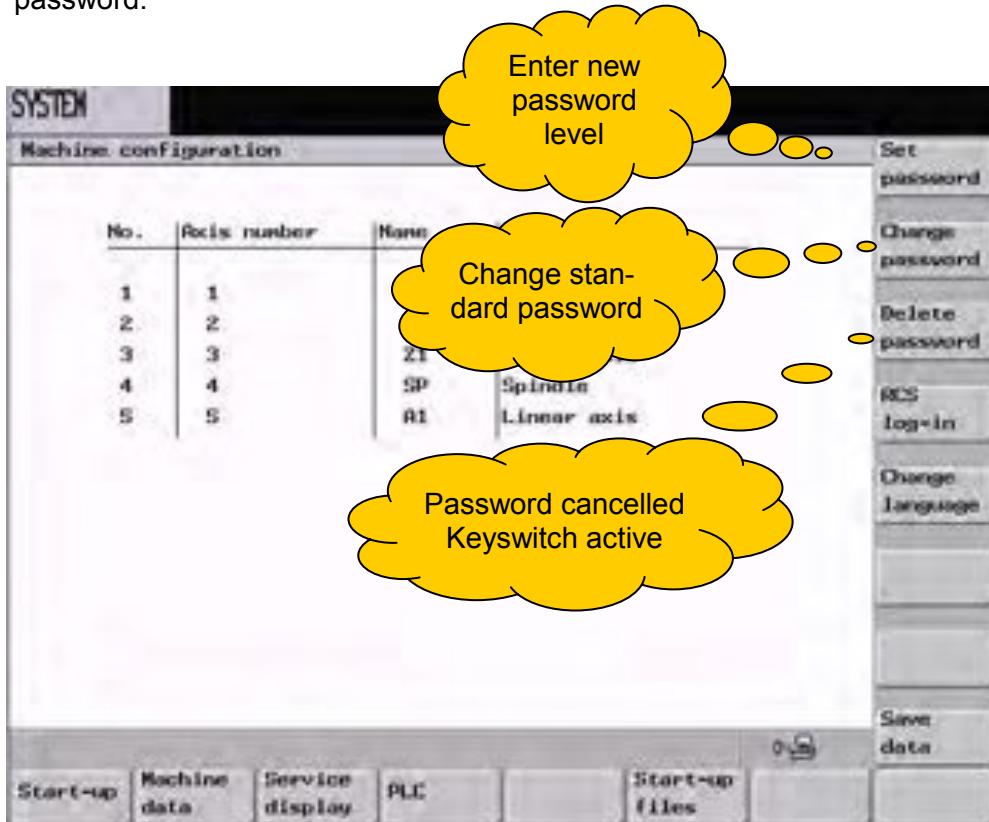
## Section 3

### Setting password

#### 3.1 Setting password

In order to make changes to the NC machine data, the corresponding password must be set. Certain machine data can only be changed with the SERVICE or MANUFACTURER password, in this case, contact the builder of the machine.

The following sequence shows graphically how to change/activate the password.



Notes

## Section 4

### Changing machine data + data save

#### 4.1 Changing machine data + data save

When the necessary password is activated the machine data can be changed.

Machine data's are each having an activation criteria, this criteria determines when the new machine data value will become active. This criteria is written in the machine data list next to the respective data.

- Po = after powering on
- Re = after pressing Reset key on machine control panel
- Cf = after powering on
- Im = immediately

**After making changes to the machine data, it is important to make a new backup see hand book module's C19 and C17**



## 1 Brief description

**Module objective:**

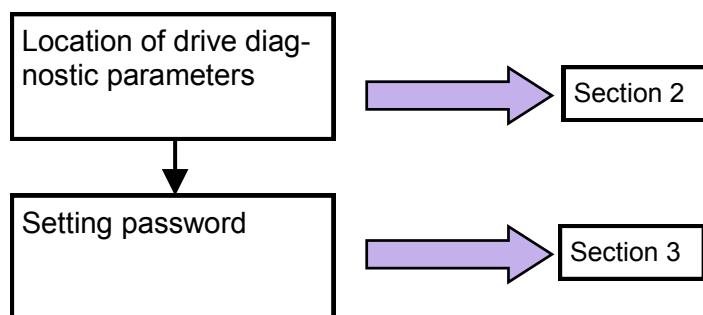
Upon completion of this module you can locate the Drive machine data, set the password and locate the diagnostic parameters.

**Module description:**

In the case of a fault, it is useful to have an overview of the fault conditions which may be existing in other areas of the control. It is often in the case of an axis alarm that a fault in the drive may be the cause.

**Module content:**

Location of drive diagnostic parameters  
Setting password

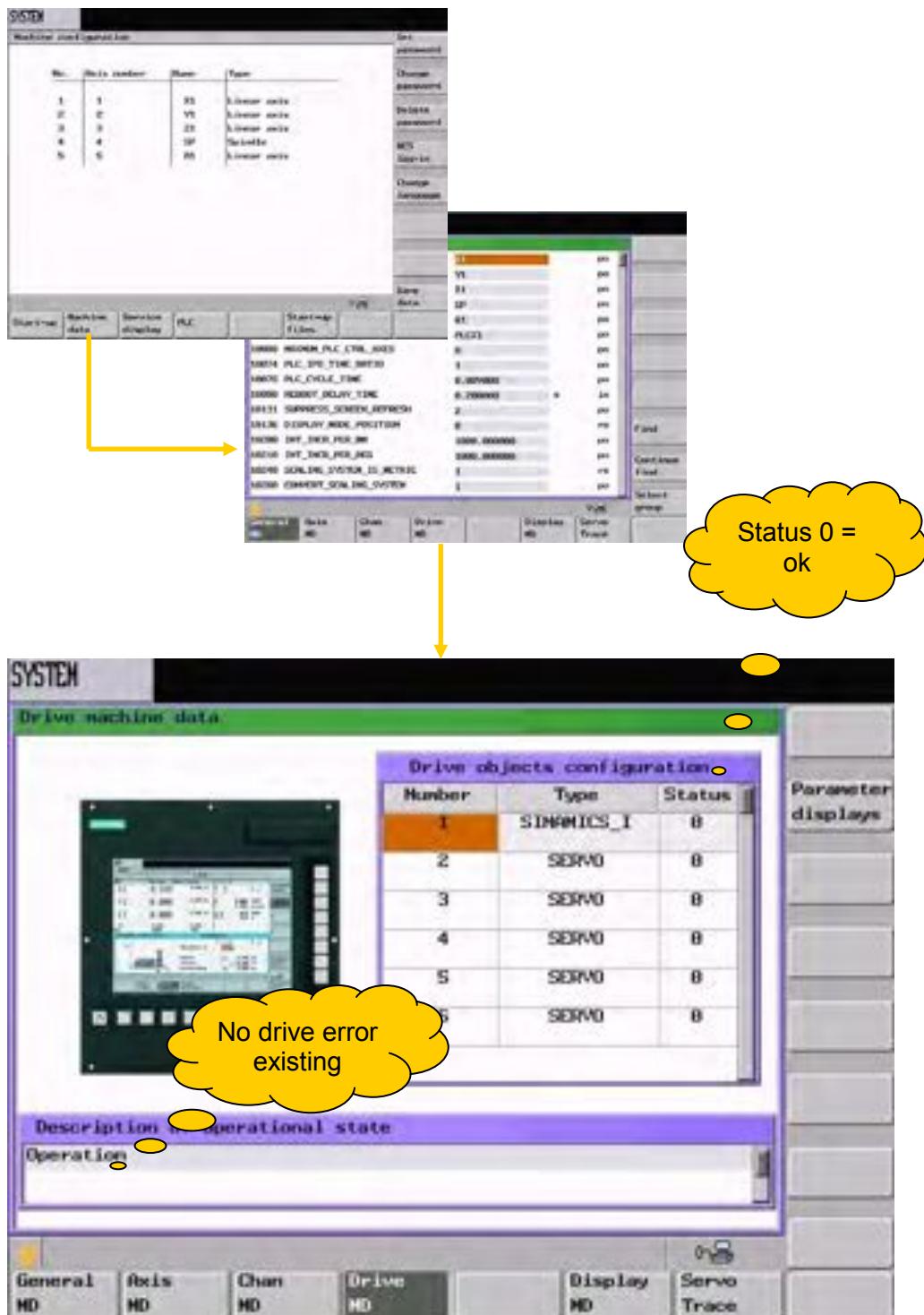


## Section 2

### Location of drive diagnostic parameters

#### 2.1 Location of drive diagnostic parameters

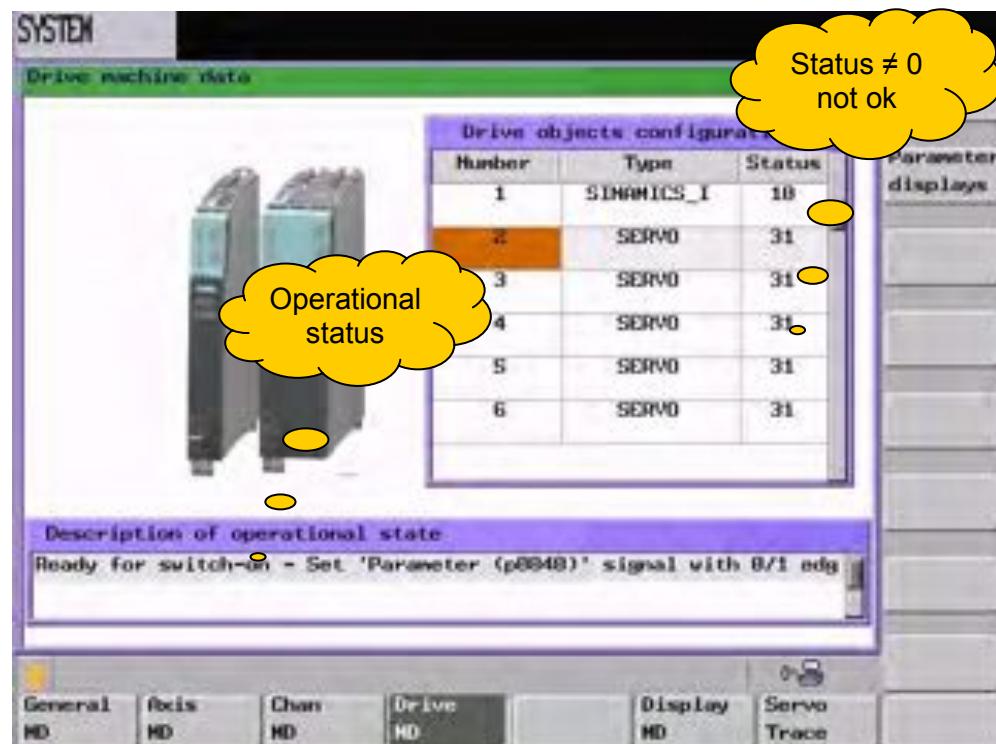
The NC machine data is located in the System area. The system area can be entered with the key combination "SHIFT + ALARM". The following describes graphically how to enter the Machine data area and how to navigate to the drive diagnostic area:



Notes

## Section 2

### Location of drive diagnostic parameters



Notes

Further important diagnostic parameters can be found under the softkey "Parameter displays". The most important parameters can be seen in the following table, and the values can be passed to the relevant service personnel upon request:

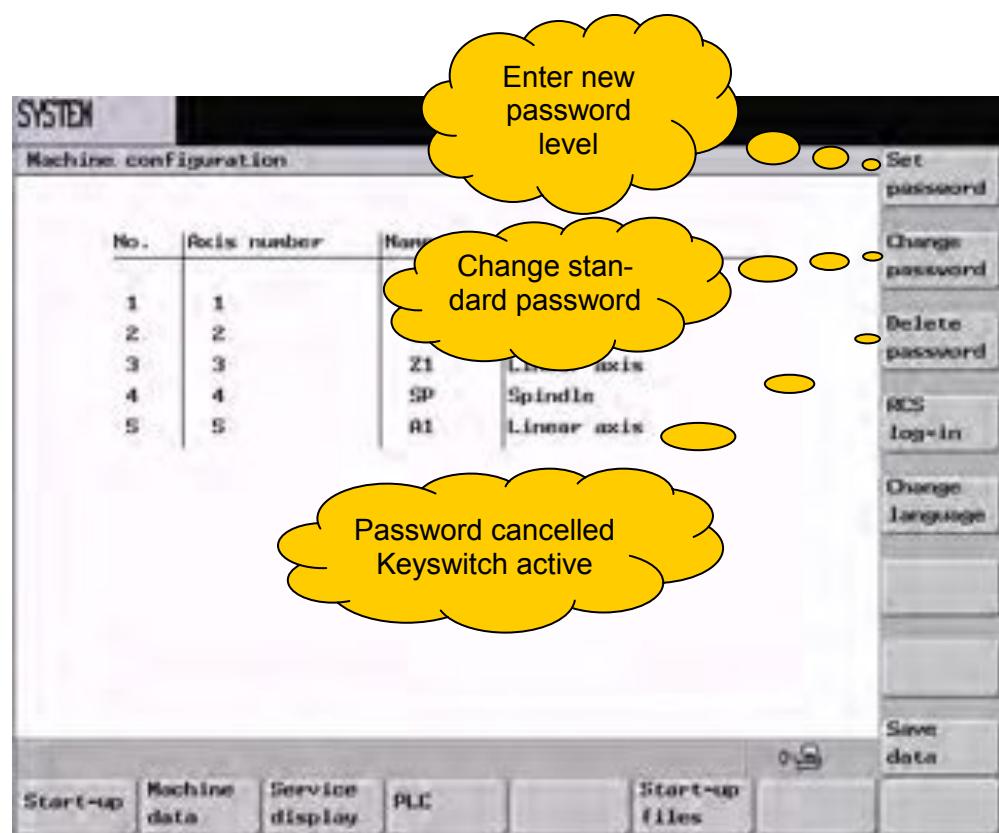
Status Parameter	Definition		
	Application	Value	Description
R2	SINAMICS_I	0	Drive is ready for operation
		10	Drive ready, but lack of drive enable signals or an alarm is output from the drive
		33	Topology error: hardware connection error or the topology comparison class P9906 not set to 3 when changing the spare parts
		35	Initial power on, drive is not commissioned
	ALM	0	Drive is ready for operation
		32	Start ready, waiting for ON/OFF1 signals, corresponding to PCU X20.1
		44	Start disable, EP enable of the infeed module not connected
		45	Start disable, an alarm is output from the infeed module
	SERVO	0	Drive is ready for operation
		23	Start ready, waiting for EP enable signals of the infeed module P854, for SLM, corresponding to PCU X20.1
		31	Start ready, waiting for drive ON/OFF1 enable signals, corresponding to NC/PLC interface enable signals (V380x0002.1 V380x4001.7NC/PLC)
		43	Start disable, ON/OFF3 enable missing, corresponding to PCU X20.2
		45	Start disable, an alarm is output from the module
R20	SERVO		Speed setpoint, smoothed
R21	SERVO		Actual speed, smoothed
R26	ALM/SERVO		DC link voltage, smoothed
R27	ALM/SERVO		Absolute actual current, smoothed
R35	SERVO		Motor temperature
R37	ALM/SERVO		Power module temperature
r722	SINAMICS_I	r722.0	PCU X20.1 status
		r722.1	PCU X20.2 status
P1450[0]	SERVO		Servo speed loop gain
P1452[0]	SERVO		Servo speed integral time

## Section 3

### Setting password

#### 3.1 Setting password

In order to see the data, the correct password should be set.



Notes