

**I N D E X**

<b>INTRODUCTION</b>	<b>PAGE</b>
● General .....	1
● Operating panel .....	2
● External operating and positioning panel .....	3
● Operating panel keys .....	4
● Switches on external panel .....	7
<b>OPERATING SEQUENCES</b>	
● Traverse to reference point .....	9
● Manual data input (MDI) .....	10
● Input of sub-program call-up via keyboard .....	11
● Input of dwell via keyboard (G4) .....	11
● "M30" input via keyboard .....	12
● Input of jump address (G20,G21) via keyboard .....	12
● Clearing complete part program memory .....	13
● Securing part program memory .....	13
● Manual input of part programs into memory .....	14
● Inserting complete blocks into programs .....	15
● Deleting complete blocks in program memory ..	16
● Modifying, deleting or inserting words .....	17
● Changing the output speed .....	18
● Actioning programs from the part program store ..	19
● Loading part programs via input devices .....	20
● Output of part programs via output devices ..	21
● Block search, sub-program and main program search .....	22
● Input of compensation values into the compensation table .....	23
● Teach - In operation .....	24
● Example for Teach - In operation .....	25
● Manual operation via jog buttons .....	26
● Manual operation via handwheel .....	27
<b>ERROR CODES</b>	
● Fault displays - general .....	28
● Error codes .....	29
● Error messages in clear text .....	35
<b>OPERATION DIALOGUES</b>	
● Questions requiring decisions .....	36
● Messages to the operator .....	36

## GENERAL

The BOSCH CNC ALPHA has been designed for direct programming at the machine.

The housing dimensions were chosen to suit this usage, making it possible to build the control directly into the machine tool or the machine panel.

The operating procedures have been designed to be as simple as possible in order to suit the usage of the control. The operator is being guided through questions and instructions in clear text. Incorrect input of data results in error messages either in clear text or as error codes.

Even though the operating procedures are simple it is advisable to familiarise yourself with the control before the first integration.

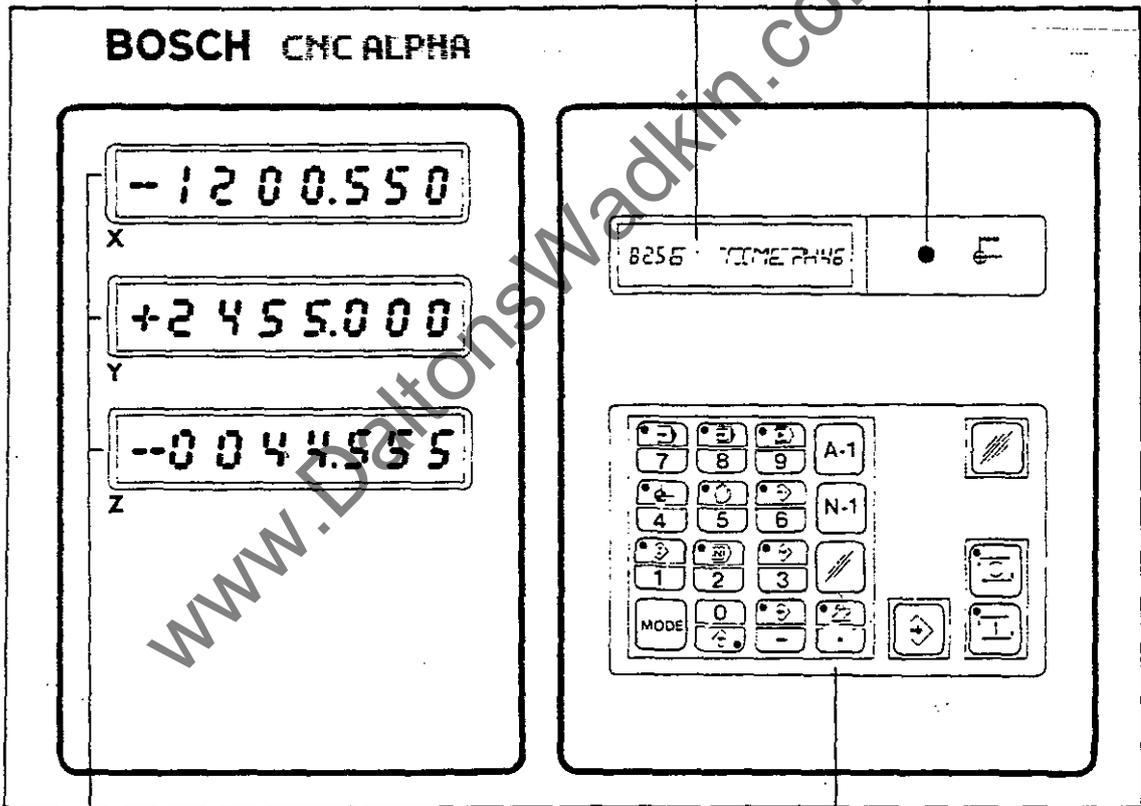
The operating instructions serve for this purpose. They are also intended to increase the operator's efficiency in using the control and to offer readily accessible information on the various operating input functions. At the end of the manual you will find a listing of the dialogues and error messages used by the control.

We hope that these instructions will be an effective help in introducing you to operating the ALPHA.

## OPERATING PANEL

LED display  
incremental programming mode

alpha-numeric display



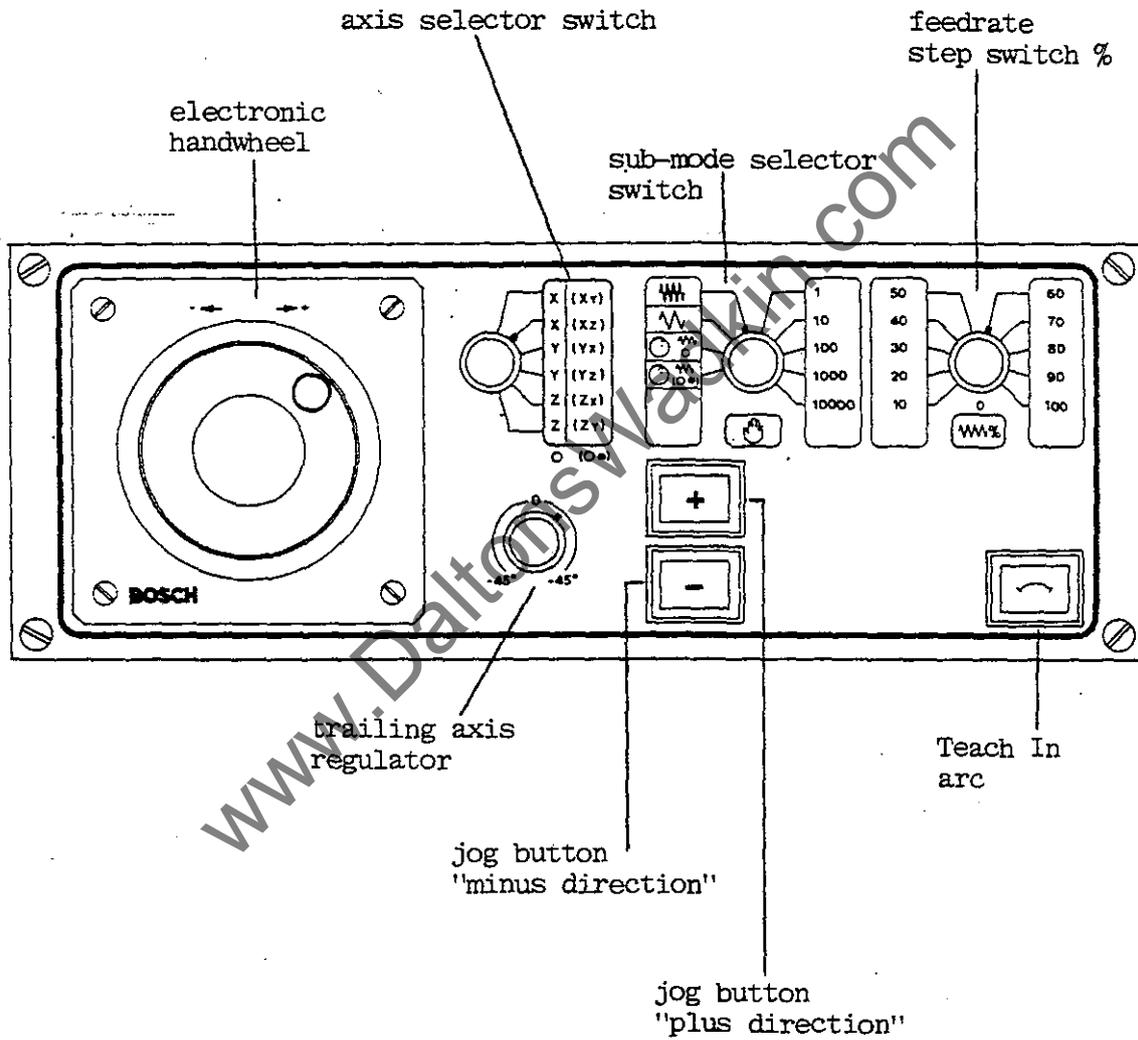
axis position  
displays

operating panel - keyboard

# EXTERNAL POSITIONING AND OPERATING PANEL (OPTION)

# BOSCH

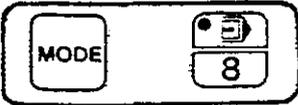
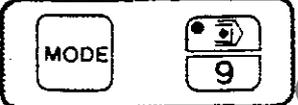
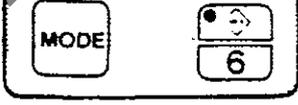
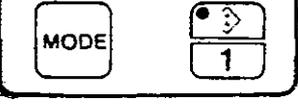
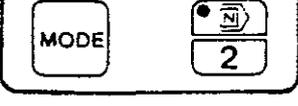
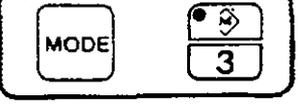
Industrierausrüstung



**OPERATING PANEL KEYS**

- before starting to operate the control it is advisable to become familiar with the individual keys and their functions
- most keys have two functions and the selection is made with the "MODE" key

**MODES OF OPERATION**

KEY	MODE	SELECTION	EXPLANATIONS
	"Automatic"		<ul style="list-style-type: none"> <li>• modes of operation are selected through actuation of the "MODE" key and the key for the particular mode at the same time</li> <li>NOTE: the "MODE" key must always be pressed first and held</li> <li>• the input of the various modes of operation is shown in the operating sequences as depicted on the left (keys on grey background)</li> <li>• if the "MODE" key is not actuated the mode is not activated but the lower key function is effective</li> <li>• if a mode is selected the lamp in the respective key lights up</li> </ul>
	"Single Block"		
	"Manual Data Input"		
	"Reference Point"		
	"Manual"		
	"Teach-in"		
	"Insert"		
	"Block Search"		
	"Modification"		

## MODES OF OPERATION

KEY	MODE	SELECTION
	"Part program input"	
	"Part program output"	
	"Tool compensations"	

## SPECIAL FUNCTIONS

### KEY



### FUNCTION

- this key is used to increment the address by 1
- the address sequence is determined as follows:  
- G, X, Y, Z, F, S, M, T
- in addition, the following addresses are effective in conjunction with the "G" address:  
- A, B, C, H
- this key is active during the program cycle, i.e. the address information can be checked while "Cycle Start" is active



- this key serves to increment the block number by 1 in the modes "Block Search" and "Modification"

## REMAINING KEYS (numerical inputs, checking functions)

### KEY

### FUNCTION



- keys 0 - 9 are used for input of figures ("MODE" key not actuated)



- negative sign ("MODE" key not actuated)



- decimal point ("MODE" key not actuated)

**BOSCH****Industrieausrüstung**

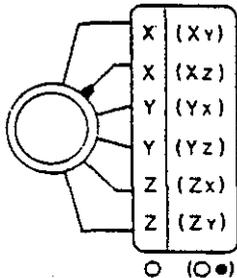
KEY	FUNCTION	EXPLANATIONS
	Clear key	<ul style="list-style-type: none"> <li>• this key serves to clear incorrect inputs in all modes of operation</li> </ul>
	Enter key	<ul style="list-style-type: none"> <li>• the entered, modified or cleared values are taken into store, or dialogue questions are acknowledged (key = end of block character)</li> </ul>
	Control Reset	<ul style="list-style-type: none"> <li>• this key serves to reset the control to a defined state</li> <li>• while a program is running this key is only effective if the  key has been actuated first</li> </ul>
	Feed Hold	<ul style="list-style-type: none"> <li>• interrupts the active program and stops all axis movement</li> <li>• enables the function of the  key while a program is running</li> </ul>
	Program Start (Cycle Start)	<ul style="list-style-type: none"> <li>• starts the execution of a program in the modes "Automatic" and "Single Block"</li> <li>• this key must be actuated twice</li> <li>• it starts the actioning of information entered in "MDI" mode</li> </ul>

## SWITCHES ON EXTERNAL PANEL Industrierausrüstung

**SWITCH**

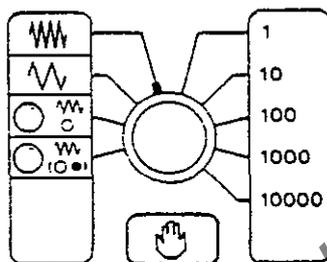
**FUNCTION**

**EXPLANATIONS**



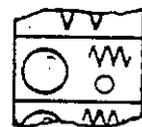
**Axis Selector Switch**

- selects the axes which are to be driven via the handwheel or the jog buttons
  - if only one axis is to travel the axis addresses above the "O" are applicable
  - if two axes are driven simultaneously via the handwheel the axis addresses above the "(O●)" are applicable
- in this case the first axis address represents the main axis and the second one represents the trailed axis  
 Example: selection "(Xy)"  
 X = main axis  
 y = trailed axis

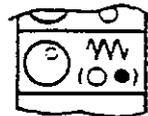


**Sub-mode Selector Switch**

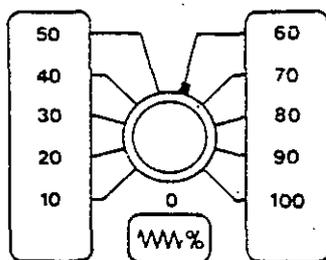
- this switch is used to select either the feedrates or the incremental step functions for jogging operation
- for handwheel operation only the following positions are effective



feedrate for handwheel operation with only one axis

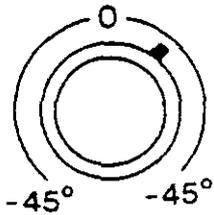


feedrate for handwheel operation with two axes



**Feedrate Override Step Switch**

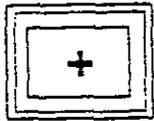
- this switch serves to reduce the programmed or manually set feedrate to a particular % value
- it is effective for all movements except when "RAPID" (GO) is active
- it can also be effective with "RAPID" depending on the specification

**BOSCH****Industrierausrüstung****SWITCH****FUNCTION**

**Trailing Axis Regulation**  
(trailing angle)

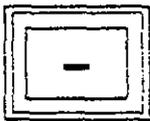
**EXPLANATIONS**

- this potentiometer is only effective in handwheel operation with two axes
- it determines the direction and the speed of the trailed axis in relation to the main axis
- 0 to +45° = both axes move in the same direction (both "+" or both "-")
- 0 to -45° = the axes move in opposite directions (1st axis in "+", 2nd axis in "-" or 1st axis in "-", 2nd axis in "+")



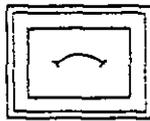
Jog Key "+"

- the axis selected with the axis selector switch travels in positive direction at the feedrate or by the incremental value set with the sub-mode selector switch



Jog Key "-"

- same as above, but in minus direction



Arc  
"G05"

- this key is only used in conjunction with the "Teach In" mode
- it is only effective if pressed together with the  key (operating panel)
- in Teach-In operation this key is used to connect start and end point with an arc

## TRAVERSE TO REFERENCE POINT

- **EXAMPLE 1:** X, Y axis are to drive to the reference point

### INPUT



select reference point mode



select X reference point



select Y reference point



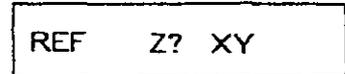
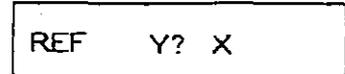
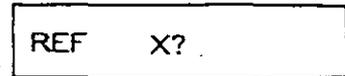
enable traverse to reference point X, Y,



start traverse to reference point X, Y,

reference point is reached, position value is set in X, Y, display

### DISPLAY



**NOT APPLICABLE TO WADKIN UX ENC.**

- **EXAMPLE 2:** only X and Z axes are to drive to the reference point

beginning of procedure as above, then



select X reference point



Y-value is skipped i.e. no Y-value entered



select Z reference point

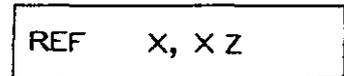
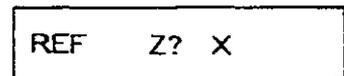
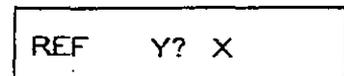


enable traverse to reference point X, Z



start traverse to reference point X, Z

reference point is reached, position value is set in X, Z display



## MANUAL DATA INPUT (MDI)

EXAMPLE: G1 X1.2 Z2 F100 M3

### INPUT



select MDI mode



A-1 Y-value is skipped, i.e. no value entered



A-1 S-value is skipped, i.e. no value entered

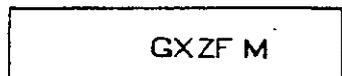
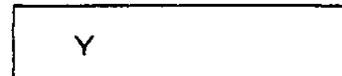


A-1 display of the addresses, for which values have been entered



program start

### DISPLAY



- NOTE! Old values must be cleared before new values can be entered, as shown in the example below:

EXAMPLE: G0 X20



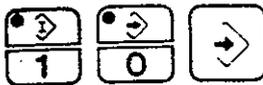
etc.



## INPUT OF SUBPROGRAM CALL-UP VIA THE KEYBOARD

- operating sequence for 'subprogram call-up' in the modes "INSERT" and "MODIFICATION"
- EXAMPLE: CALL-UP OF A SUBPROGRAM DURING THE INPUT OF A MAIN PROGRAM VIA THE KEYBOARD**

### INPUT



input of the last word (Z10) in block 17, transfer of the block to memory and incrementing block no. by 1

### DISPLAY

18G



enter subprogram call-up (G22)

18G 22



enable subprogram number input

18G SBP=C



input (1-99) of the subprogram number (e.g. subprogram no. 2 is to be called up) and enable input of the repetitions

18G REP=B



input (1-99) determining how often the selected subprogram is to be repeated (for instance 4 = 5 times)

18G REP=B4



incrementing the address

18X

## INPUT OF DWELL VIA KEYBOARD

- operating sequence for the input of a dwell in the modes "MDI", "INSERT" and "MODIFICATION"
- EXAMPLE: ENTERING A DWELL VIA THE KEYBOARD**

### INPUT



select "G4" dwell

### DISPLAY

G 4



enable input of time

11G DWELL



enter dwell time (e.g. 50 = 5 sec)

11G DWELL 50



transfer block to memory and proceed with next block

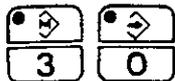
12G

**BOSCH**

Industrieausrüstung

**M 30 INPUT VIA KEYBOARD**

- operating sequence for keyboard input of "M30" (program end) with backwards jump address in the modes "INSERT" and "MODIFICATION"
- together with "M30" a backwards jump address (block number) is programmed to determine to which block the backwards jump is to be made at the end of the program

**INPUT**

enter "M30" program end  
(for instance in block N8)



enable jump address input



enter backwards jump address  
(block N2=2) and complete  
block

**DISPLAY**

- when the control recognises "M30" program end it will action a jump back to the entered block number once the last program block has been carried out

**INPUT OF JUMP ADDRESSES G20, G21 VIA KEYBOARD**

- operating sequence for keyboard input of jump addresses under "G20" and "G21" in the modes "INSERT" and "MODIFICATION"

**INPUT**

enter G-function "G20" - uncon-  
ditional jump (e.g. in block N11)



enable jump address input



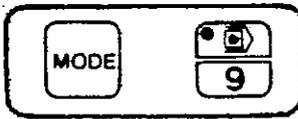
enter the jump address (e.g. jump  
to block N6) and increment address

**DISPLAY**

- jumps under "G21" are entered in the same way

## CLEARING COMPLETE PART PROGRAM MEMORY (G39)

### INPUT



select MDI mode



enable program memory (G37)



select MDI mode

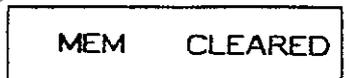
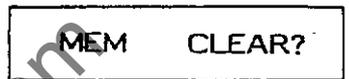
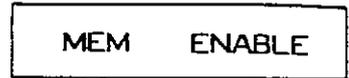
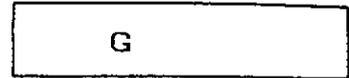


enter 39  
control requests confirmation  
of the instruction



complete part program memory  
will be cleared

### DISPLAY

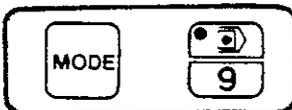


- subsequently "INSERT" mode can be selected and new part programs can be entered

## SECURING PART PROGRAM MEMORY (G38)

- once the part program memory has been secured input in the modes "INSERT", "MODIFICATION" and "TOOL COMPENSATION" is inhibited

### INPUT

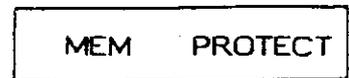


select MDI mode



secure part program store  
(G38)

### DISPLAY

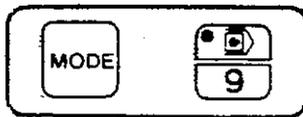


- the part program store is now secured and protected from accidental clearing **AND UNAUTHORISED TAMPERING.**

## INPUT OF PROGRAMS INTO THE PROGRAM MEMORY

- when entering complete part programs into the part program memory it is not necessary to enter numbers for main programs
- the control recognises the program marking with G78 = main program and increments the main program number by 1 for each programmed "G78"
- for subprograms, however, the program number must be entered after "G98" = subprogram start
- 9 main programs (9 x G78) with the program numbers 1 - 9 can be stored
- 99 subprograms (99 x G98) with the program numbers 1 - 99 can be stored

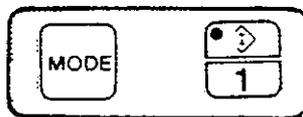
### INPUT



select MDI mode



enable part program memory



select "INSERT" mode



this input will select the first free block number, into which the first block with G78 or G98 can be entered directly



enter main program start (G78)

OR



enter subprogram start (G98)



enter subprogram number (e.g. no. 1)



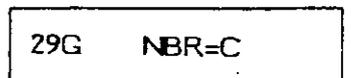
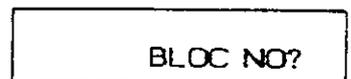
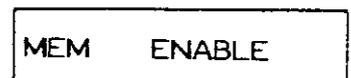
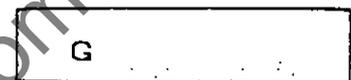
this key is used to increment from address to address within a block; the values for the addresses can be entered at the same time



with the actuation of this key the entered block is transferred and stored, block number is incremented by 1

**NOTE!** this key must be actuated at the end of each block, otherwise the block will not be transferred to memory (key = end of block)

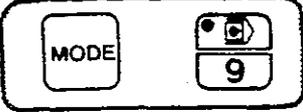
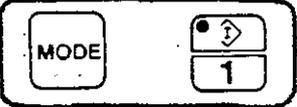
### DISPLAY



## INSERTING COMPLETE BLOCKS INTO STORED PART PROGRAMS

**NOTE:** if additional blocks are inserted in part programs already in the memory, the control automatically changes the jump addresses programmed under G20 or G21 into the new values

- **EXAMPLE:** a block G1 Y20 is inserted between blocks N10 and N11

INPUT		DISPLAY
	select MDI mode	G
	enable part program memory	MEM ENABLE
	select "INSERT" mode	BLOC NO?
	enter number of block in front of which a new block is to be inserted (e.g.N11)	11G
	enter G-function (G1) of the new block and increment address (X)	11X
	X-value is skipped	11Y
	enter Y-value (Y20) and transfer the new block into the program memory	

- the block numbers of the subsequent blocks are increased by 1

● **Example:**

block sequence before insertion	block sequence after insertion
N9 G0 X20 Z10	N9 G0 X20 Z10
N10 G2 Y10 F30	N10 G2 Y10 F30
N11 G0 Z5	N11 G1 Y20 (new inserted block)
N12 X40	N12 G0 Z5 (old block N11 before insertion)
N13 M2	N13 X40
	N14 M2

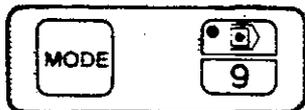
- switching over to another mode of operation is possible as soon as the input is completed

## DELETING COMPLETE BLOCKS IN THE PROGRAM MEMORY

- **NOTE:** when blocks are deleted from stored part programs the control automatically changes the jump addresses programmed under "G20" to the new values
- **EXAMPLE:** BLOCK 12 IS TO BE DELETED

**INPUT**

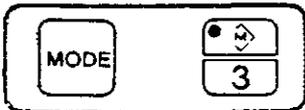
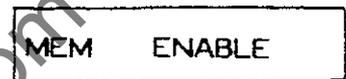
**DISPLAY**



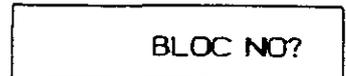
select MDI mode



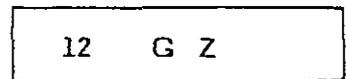
enable part program memory



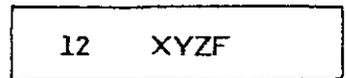
select "MODIFICATION" mode



enter number of block to be deleted (e.g. N12)



block N12 is deleted, content of former block 13 is displayed



**NOTE:** if the delete key is pressed once more after deleting one block, this will delete the block before the one, which was first deleted

- once the input is completed a different mode of operation can be selected (MODE-key + respective mode key).
- the block numbers of the subsequent blocks are reduced by 1
- **EXAMPLE:**

block sequence before deletion

block sequence after deletion

```
N9  G0  X20  Z10
N10 G2  Y10  F30
N11      X40
N12  G1  Z50
N13      X55  Y40  Z60  F20
N14      M2
```

```
N9  G0  X20  Z10
N10 G2  Y10  F30
N11      X40
N12      X55  Y40  Z60  F20
N13      M2
```

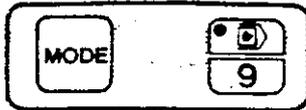
(old block N13 before delete)

## MODIFYING, DELETING OR INSERTING WORDS

- in the "MODIFICATION" mode words in blocks of stored part programs can be modified, deleted or inserted
- **EXAMPLE:** IN BLOCK N10 THE Z-VALUE IS CHANGED FROM Z12 to Z12.5

INPUT

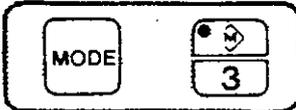
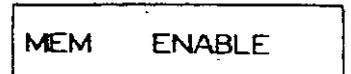
DISPLAY



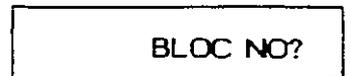
select MDI mode



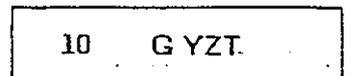
enable part program memory (G37)



select "MODIFICATION" mode



select block containing word to be modified (e.g. N10)

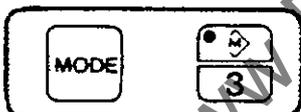


search for the address (e.g. Z), delete old and enter new value (e.g. 12.5)

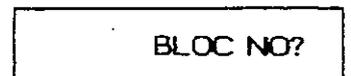


- if further values in other blocks are to be modified, either increment the block number with N+1 until the next required block is reached, or select the required block number direct by reselecting the "MODIFICATION" mode (keys  +  )

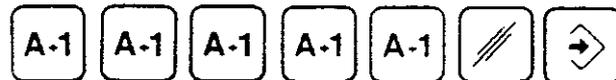
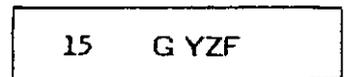
- **EXAMPLE:** IN BLOCK 15 THE F-WORD IS TO BE DELETED



select "MODIFICATION" mode



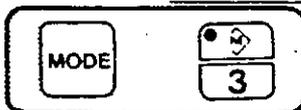
select block (e.g. N15) with word to be deleted



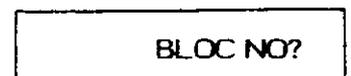
search for and delete the word (e.g. F-word)



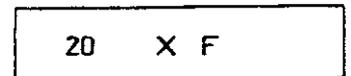
- **EXAMPLE:** IN BLOCK N20 Y-1.6 IS TO BE INSERTED



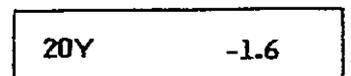
select "MODIFICATION" mode



select block (e.g. N20) into which the word is to be inserted



search for the address (e.g. Y) and enter the value (e.g. -1.6)

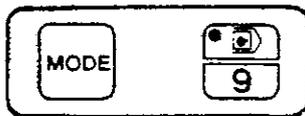


## CHANGING THE OUTPUT SPEED (BAUD RATE)

- for the connection of various input and output devices the required output speed (Baud Rate) can be entered over the keyboard
- the output speed is entered as a 1-digit figure (0 - 7)
- the following output speeds can be selected:

0 =	50	Baud	8 =	1800	Baud	} included in extension package
1 =	75	Baud	9 =	2000	Baud	
2 =	110	Baud (ASR33)	10 =	2400	Baud	
3 =	134.5	Baud	11 =	3600	Baud	
4 =	150	Baud	12 =	4800	Baud	
5 =	300	Baud (ASR43)	13 =	7200	Baud	
6 =	600	Baud	14 =	9600	Baud	
7 =	1200	Baud				

### INPUT



select MDI mode



enable program memory



enable output speed



display of active output speed

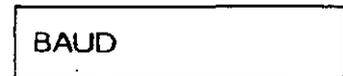
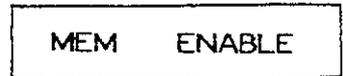
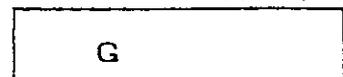


delete old output speed



enter required output speed (e.g. 2 = 110 Baud)

### DISPLAY



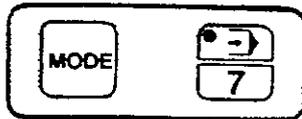
- once the input is completed another mode of operation can be selected

## ACTIONING PROGRAMS FROM THE PART PROGRAM STORE

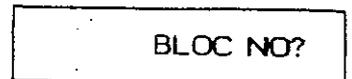
- programs, which are stored in the part program memory, are called up with their program number and then actioned in AUTOMATIC or SINGLE BLOCK mode
- the selected program can be actioned complete or from a specific block onwards
- **EXAMPLE 1:** ACTIONING MAIN PROGRAM NO. 2 FROM THE BEGINNING IN 'AUTOMATIC' MODE

**INPUT**

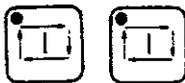
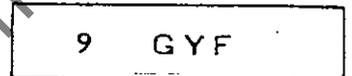
**DISPLAY**



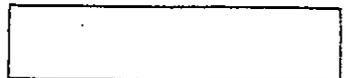
select "AUTOMATIC" mode



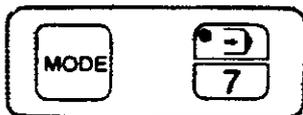
enable program no. selection and enter program number (e.g.2), control goes to 1st block of the selected program (= "G78" main program identif.)



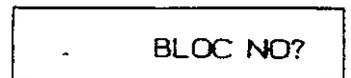
program start: program is automatically carried out block by block



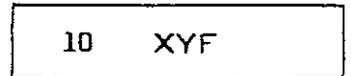
- **EXAMPLE 2:** ACTIONING MAIN PROGRAM NO. 2 FROM BLOCK N10 ONWARDS IN "SINGLE BLOCK" MODE



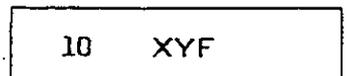
select "AUTOMATIC" mode



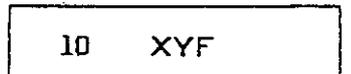
select block number (e.g. N10)



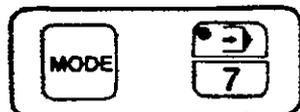
select "SINGLE BLOCK" mode



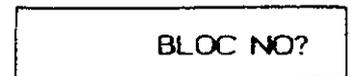
start (program must be restarted at the end of each block)



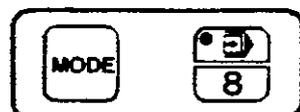
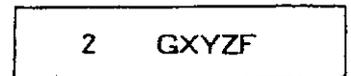
- **EXAMPLE 3:** ACTIONING MAIN PROGRAM NO. 2 FROM THE BEGINNING IN "SINGLE BLOCK" MODE



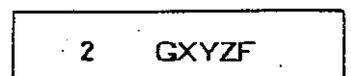
select "AUTOMATIC" mode



enable program number selection and enter program number



select "SINGLE BLOCK" mode

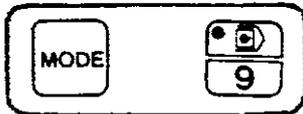


start (program must be restarted at the end of each block)

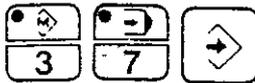
## LOADING PART PROGRAMS INTO THE PROGRAM STORE

- part programs can be loaded from cassette unit or tape reader
- paper tapes must be written in ISO code
- whichever block number is entered from tape, the memory stores the 1st block of the input part program in the lowest free block number

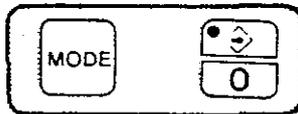
### INPUT



select MDI mode



enable program memory



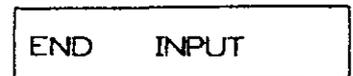
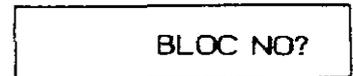
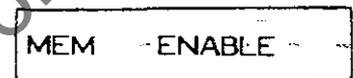
select program load



let the control find the first free block number (e.g. 15), under which the first block of the program will be stored and start loading action

the completion of the loading will be indicated by "END INPUT"

### DISPLAY



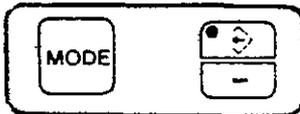
- **NOTE:** cassettes must be loaded at the same Baud-rate as they were output (e.g. output at 1200 Baud - input also at 1200 Baud)
- **NOTE:** if, when loading a program, a block is selected, which has already been used, the new program will be entered from this block number onwards and the previous contents of this and the subsequent blocks are shifted upwards in the program store by the number of blocks making up the new program, i.e. in this way a previously entered program might be split.  
it is therefore safest to key in block number 999, thereby letting the control search for the lowest free block number

**BOSCH**

Industrierausrüstung

## OUTPUT OF PART PROGRAMS FROM THE PART PROGRAM STORE

- part programs can be output on to cassette or paper tape
- paper tapes are punched in ISO code
- the output part program generally begins with block number N1, no matter which block number has been selected
- the control always outputs the complete contents up to the end of the program store  
i.e. it is not possible to output individual part programs on their own
- the stored part program is retained in the program store

**INPUT**

select program output mode



enter the block number from which onwards the output is required

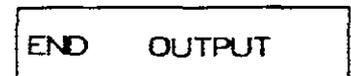
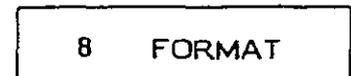
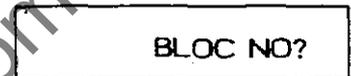


select tabulated output output will start

OR



select not tabulated output output will start

**DISPLAY**

the completion of the output operation is indicated by the display "END OUTPUT"

# BOSCH

Industrierausrüstung

## BLOCK SEARCH, SUB-PROGRAM AND MAIN PROGRAM SEARCH

- in the "BLOCK SEARCH" mode blocks, subprograms and main programs can be searched for
- when searching for sub and main programs the control jumps to the first block of the searched program, which serves for the program identification  
(G98 = subprogram, G78 = main program)

### EXAMPLE 1: BLOCK SEARCH, E.G. BLOCK NO. N15

#### INPUT



select "Block Search" mode



enter required block number.  
when block is found the display shows the used addresses

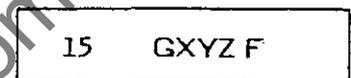
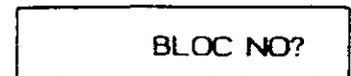


this key is used to display the addresses in the block with their content one by one

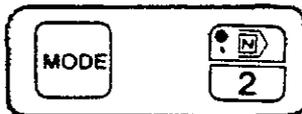


this key is used to pass on to the next block, the addresses of which can then be displayed again with 'A+1'

#### DISPLAY



### EXAMPLE 2: MAIN PROGRAM SEARCH, E.G. No. 2



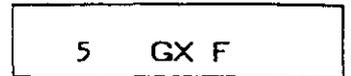
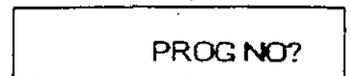
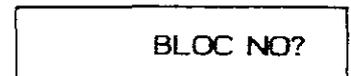
select "Block Search" mode



enable program number

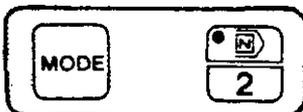


enter main program no. (e.g. 2)

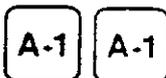


- key 'N+1' can be used to pass on to the next block and "A+1" to display the individual addresses of the block

### EXAMPLE 3: SUBPROGRAM SEARCH, E.G. No. 1



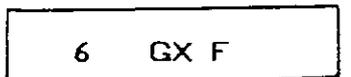
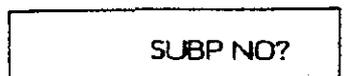
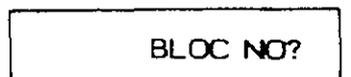
select "Block Search" mode



enable subprogram number



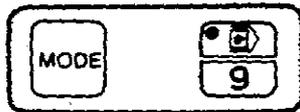
enter subprogram no.(e.g.1)



- key 'N+1' can be used to pass on to the next block and 'A+1' to display the individual addresses of the block

**INPUT OF COMPENSATION VALUES INTO THE COMPENSATION TABLE**

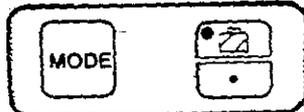
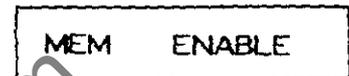
- the tool compensation table is laid out for the storage of 16 tool length compensation values under "L" and 16 cutter radius compensation values under "R"
- EXAMPLE:** INPUT IN T1 - R = 2 mm L = 3 mm, T2 - NO INPUT, T3 - R = 1.2 mm L = NO INPUT, ETC.

**INPUT****DISPLAY**

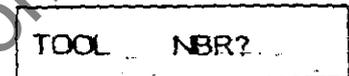
select "MDI" mode



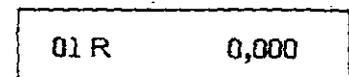
enable program memory



select tool compensation table



select compensation number (e.g. 1 = T1)



enter the cutter radius compensation value (e.g. 2 = R2mm) the control will go on to the tool length comp. value of the selected comp. number



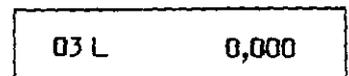
enter tool length compensation number (e.g. 3 = L3mm) the control will go on to the next compensation number



increment the compensation number if there are no values to be entered in e.g. T2



enter the radius compensation value in T3 (e.g. 1.2 = R1.2mm) the control will go on to the tool length compensation value of T3



etc

- the 'A+1' key can be used to skip complete compensation numbers
- once the input into the compensation table is completed the contents can be brought up step by step with the 'A+1' key for checking

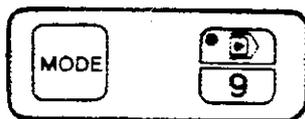
**NB. L LENGTH COMPENSATION IS NOT APPLICABLE TO UXCAR**

**BOSCH**

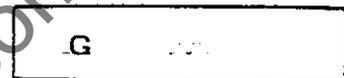
Industrieausrüstung

**TEACH-IN OPERATION**

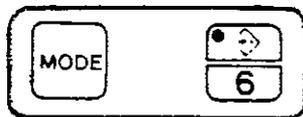
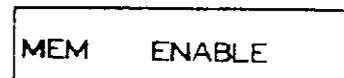
- **NOTE:** "Teach In" can not be used to insert blocks into existing programs
- in "Teach In" mode the first work-piece can be machined as in "Manual" operation and the relevant information stored block by block
- no matter how the axes travel to the different points of the workpiece, the control always links one point to the next with one movement (which corresponds to one block)
- the type of start input determines whether two points of the work-piece are to be linked with a straight line (linear) or with an arc (circular)

**INPUT****DISPLAY**

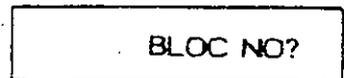
select "MDI" mode



enable program store



select "Teach In" mode



enter 999 to call up the first free block number (e.g. N8) in the memory (input can start from this block)

**A-1**

this key is used to call up the functions required in the block, so that values can be entered:

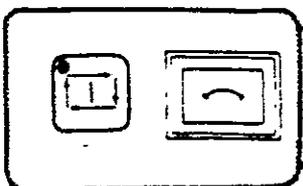
- F-value (feedrate)
- S-value (spindle speed)
- M-function
- T-number (tool number)

- these addresses represent supplementary information and must be entered

- the axes can be moved to the various points of the work-piece in manual operation (see also page 26) and the details of these points can be stored



- if this input key is pressed on its own the control will store the movement from start to end point as a linear movement and automatically increment the block number



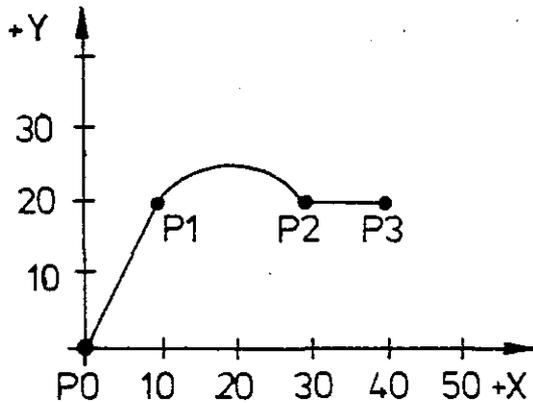
- if these two keys are pressed simultaneously the start and end point will be linked with an arc and the information stored as one block in the memory

- **NOTE:** the  key must be pressed and held before the start key is pressed, otherwise a linear movement will result

- **NOTE:** Once the input is completed "G78" or "G98" (main or subprogram identification) must be entered in the 1st block (see page 15)
- in the last block of the program either "M2" (main program) or "G99" must be entered (subprogram)

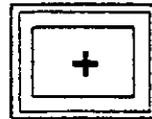
### EXAMPLE FOR 'TEACH IN'

- first the "Teach In" mode must be selected (see page 24)



#### Input: P0 → P1 Y-axis

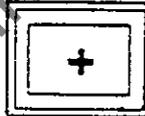
- axis selector switch to Y-axis
- sub-mode selector switch to 10000
- jogging the Y-axis to Y-value of P1 by pressing



twice = Y20

#### Input: P0 → P1 X-axis

- axis selector switch to X-axis
- sub-mode selector switch remains as before
- jogging the X-axis to X-value of P1 by pressing



once = X10

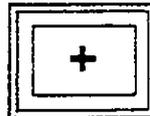
#### Storing P0 → P1 as a straight line



- P0 — P1 is stored as one block

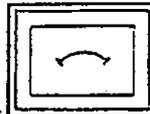
#### Input: P1 → P2 X-axis (input for Y-axis not required, since the value remains the same)

- axis selector switch to X-axis
- sub-mode selector switch to 10 000
- jogging axis by pressing



twice = X20 (P1 → P2)

#### Storing P1 → P2 as an arc



- P1 → P2 is stored as one block

#### Input: P2 — P3 X-axis (only)

- axis selector switch to X-axis
- sub-mode selector switch to 10 000
- jogging the axis

#### Storing P2 — P3 as a straight line



## MANUAL OPERATION

- in the "MANUAL" mode the axes are either moved via the jog buttons or with the handwheel
- the traversing speed is set with the sub-mode selector switch
- the same procedure is used to move the axes to the various points of a work-piece in "Teach In" mode

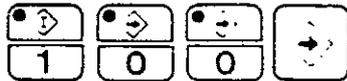
### MOVING THE AXES WITH THE JOG BUTTONS

#### INPUT

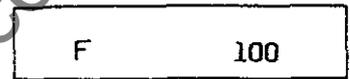


select "MANUAL" mode  
(NOTE: this is not to be input for "Teach In" operation)

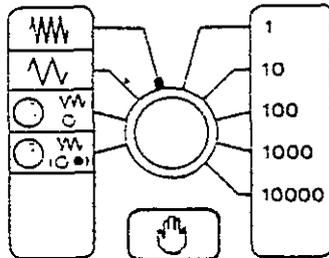
#### DISPLAY



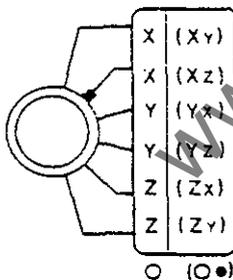
enter the feedrate at which the traversing is to take place (e.g. F100)



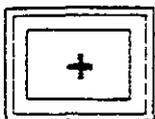
OR



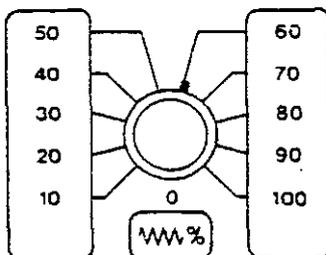
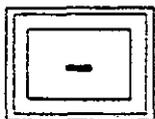
select the feedrate or an incremental jog with the sub-mode selector switch



select the axis, which is to be moved via the jog button  
(for jogging the axes the addresses above the "O" are valid)



actuate one of these two buttons to action axis traverse at the feedrate entered manually or selected with the sub-mode selector switch



the feedrate override potentiometer can be used to reduce the set or entered feedrate to a specific percentage value

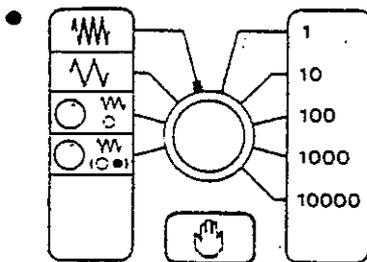
## MOVING THE AXES WITH THE HANDWHEEL

- the handwheel can be used to move one axis on its own or two axes simultaneously
- if two axes are moved, one of them represents the main axis, the second axis is being trailed
- this operating procedure is also possible in "Teach In" mode

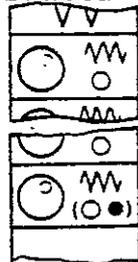
### INPUT



select "MANUAL" mode  
(NOTE: this is not to be input in "Teach In" operation)

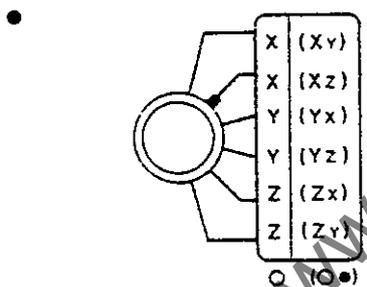


set the sub-mode selector switch to



if only one axis is to be moved

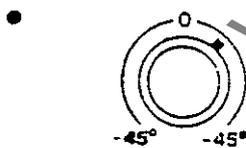
if two axes are to be moved.



select the axis (axes)

- if only one axis is moved the addresses above "0" are valid

- if two axes are moved the addresses above "O●" are valid

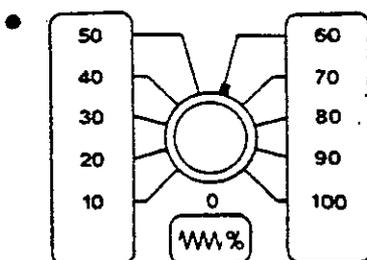


NOTE: this is only active for handwheel operation with two axes

selection of direct and feedrate of the trailed axis (secondary axis) in relation to the main axis

0 to +45° : both axes move in the same direction

0 to -45° : the two axes move in opposite directions

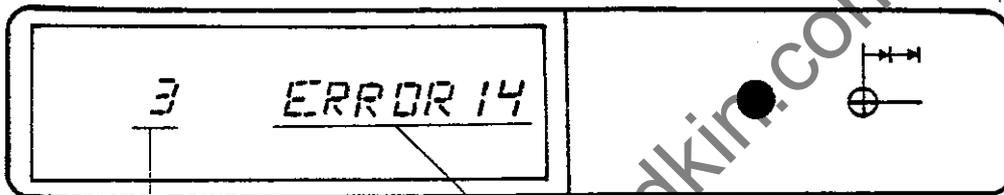


NOTE: the feedrate override potentiometer can be used to reduce the feedrate to a specific percentage value

## ERROR CODES

### FAULT DISPLAYS - GENERAL

- all error messages are displayed on the alpha-numeric display
- there are two types of error message:
  - error messages in clear text
  - error messages in coded form
- the coded error messages are displayed as follows:



block containing  
the error

error code (type of error)

- coded error messages can only be cleared by actuating the **A-1** key or switching to a different mode of operation
- the block shown to contain an error can be selected in "Modification" mode and the addresses of this block can then be checked through with the **A-1** key.
- the incorrect address can be deleted or modified at the same time

## ERROR CODES

Error Message	Cause	Error elimination
ERROR 1	<ul style="list-style-type: none"> <li>G5 has been programmed with radius</li> </ul>	<ul style="list-style-type: none"> <li>change G-code or cancel radius</li> </ul>
ERROR 3	<ul style="list-style-type: none"> <li>G1, G2, G3, G61, G62, or G63 has been programmed with axis addresses but without F-value</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and enter feedrate (F-value)</li> </ul>
ERROR 4	<ul style="list-style-type: none"> <li>G92, G25, G26, G27 or G74 has been programmed without axis addresses</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and enter axes addresses</li> </ul>
ERROR 5	<ul style="list-style-type: none"> <li>a sub-program has been called up which is not contained in the memory</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and enter the correct sub-program number</li> </ul>
ERROR 6	<ul style="list-style-type: none"> <li>more than 5-fold sub-program nesting has been programmed</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and delete the 6th sub-program call-up, then modify program</li> </ul>
ERROR 7	<ul style="list-style-type: none"> <li>after a jump back from a sub-program into the active main program no M2 has been found at the end of the main program</li> </ul>	<ul style="list-style-type: none"> <li>select the last block of the active main program in "Modification" mode and enter M2</li> </ul>
ERROR 8	<ul style="list-style-type: none"> <li>a non-defined G-code has been programmed</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and delete or modify G-code</li> </ul>
ERROR 9	<ul style="list-style-type: none"> <li>the programmed circular movement exceeds software limits or the maximum interpolation path is being exceeded.</li> </ul>	<ul style="list-style-type: none"> <li>modify the part program since the programmed path exceeds the field of operation</li> </ul>

Error Message	Cause	Error Elimination
<b>ERROR 10</b>	<ul style="list-style-type: none"> <li>a change of radius compensation (G40, G41, G42) has been programmed, although the next movement is a circular one</li> </ul>	<ul style="list-style-type: none"> <li>program the change of radius compensation (G40, G41, G42) either one block earlier or one block later</li> </ul>

**EXAMPLE 1:**

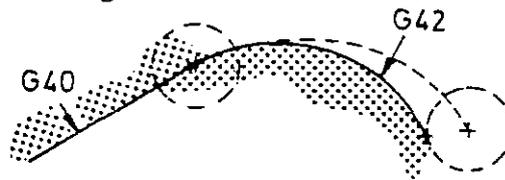
change from G40 to G41

**EXAMPLE 2:**

change from G41 to G42

**EXAMPLE 3:**

change from G40 to G42

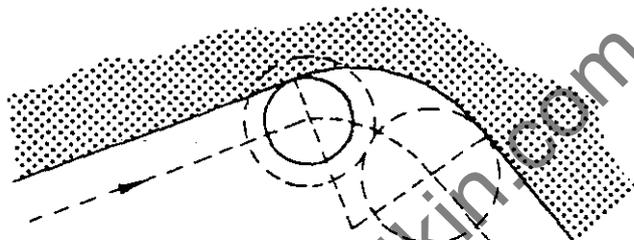
**EXAMPLE 4:**

change from G42 to G41

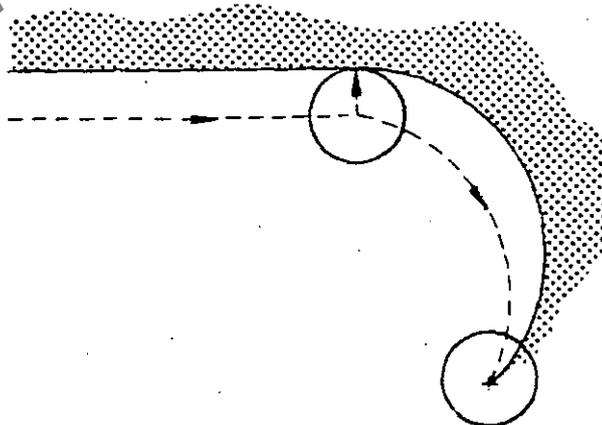


**BOSCH****Industrierausrüstung**

Error Message	Cause	Error Elimination
<b>ERROR 11</b>	<ul style="list-style-type: none"> <li>while cutter radius compensation is active (G41, G42) a new radius compensation value has been called up during a circular movement</li> </ul> <p>Example:</p>	<ul style="list-style-type: none"> <li>call-up of the new radius compensation value (T) is to be programmed before or after the circular movement</li> </ul>



<b>ERROR 12</b>	<ul style="list-style-type: none"> <li>G40 "Clearing cutter radius compensation" has been programmed although the next movement is a circular one</li> </ul> <p>Example:</p>	<ul style="list-style-type: none"> <li>clear the cutter radius compensation before or after the circular movement</li> </ul>
-----------------	--	--



<b>ERROR 13</b>	<ul style="list-style-type: none"> <li>tool number "T0" has been programmed or T26 - T99</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and delete or modify tool number "T0", T26 - T99</li> </ul>
-----------------	---	--

**BOSCH****Industrierausrüstung**

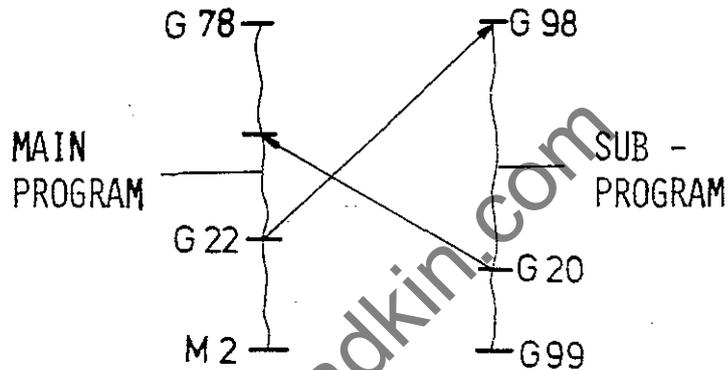
Error Messages	Cause	Error Elimination
ERROR 14	<ul style="list-style-type: none"> <li>"F0" has been programmed as the feed-rate (F-value)</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and enter F-value</li> </ul>
ERROR 15	<ul style="list-style-type: none"> <li>sub-program call-up (G22, G23) has been programmed with the number "0" or without any number</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and enter number to go with G22 or G23</li> </ul>
ERROR 16	<ul style="list-style-type: none"> <li>jump command (G20, G21) has been programmed with address "0" or without any address</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and enter address to go with G20 or G22</li> </ul>
ERROR 17	<ul style="list-style-type: none"> <li>the first movement after a change of plane (G17, G18, G19) is a circular one</li> </ul> <p>EXAMPLE: change of plane at "B"</p> 	<ul style="list-style-type: none"> <li>program the plane change during a linear movement</li> </ul>
ERROR 18	<ul style="list-style-type: none"> <li>a tool compensation (G41, G42, G43) has been called up without tool number "T"</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and enter tool number (T)</li> </ul>

**BOSCH****Industrierausrüstung**

Error Message	Cause	Error Elimination
ERROR 21	<ul style="list-style-type: none"> <li>with G41 or G42 being active, one of the axes of the active plane (G17, G18, G19) has been programmed together with G74 (automatic traverse to reference point)</li> </ul>	<ul style="list-style-type: none"> <li>cancel the tool compensation with "G40" and only then carry out "G74" traverse to the reference point</li> </ul>
ERROR 22	<ul style="list-style-type: none"> <li>with G41 or G42 being active, a conditional sub-program call-up has been programmed (G23)</li> </ul>	<ul style="list-style-type: none"> <li>cancel the tool compensation with "G40", then program the conditional sub-program call-up with "G23"</li> </ul>
ERROR 23	<ul style="list-style-type: none"> <li>with G41 or G42 being active, a change of plane has been programmed with "G17", "G18" or "G19"</li> </ul>	<ul style="list-style-type: none"> <li>cancel the tool compensation with "G40", then change the plane with "G17", "G18" or "G19"</li> </ul>
ERROR 24	<ul style="list-style-type: none"> <li>with G41 or G42 being active, M2 or M30 (program end) has been programmed</li> </ul>	<ul style="list-style-type: none"> <li>cancel the tool compensation with "G40", then program M2 or M30</li> </ul>
ERROR 25	<ul style="list-style-type: none"> <li>with G41 or G42 being active, a conditional jump (G21) has been programmed</li> </ul>	<ul style="list-style-type: none"> <li>cancel the tool compensation with "G40", then program the conditional jump with "G21"</li> </ul>
ERROR 26	<ul style="list-style-type: none"> <li>the next block number would be N1000 (only up to 999 blocks can be stored)</li> </ul>	<ul style="list-style-type: none"> <li>output all stored part programs on to paper tape or cassette, then clear store with "G39"</li> <li>another 999 blocks can then be entered into the memory</li> </ul>
ERROR 27	<ul style="list-style-type: none"> <li>with "G41" being active, the control has no programmed axis movements available to calculate the cutter path</li> </ul>	<ul style="list-style-type: none"> <li>select incorrect block in "Modification" mode and program some axis information</li> </ul>

**BOSCH****Industrieausrüstung**

Error Message	Cause	Error Elimination
<b>ERROR 28</b>	<ul style="list-style-type: none"> <li>in a selected sub-program a jump has been programmed with "G20" or "G21", which would lead to an endless repetition of this sub-program</li> </ul>	<ul style="list-style-type: none"> <li>check the jump address programmed under "G20" or "G21"</li> <li>no jump must be programmed which would go back into the main program before the sub-program call-up</li> </ul>



<b>ERROR 29</b>	<ul style="list-style-type: none"> <li>no "M2" or "M30" has been programmed at the end of the active main program</li> </ul>	<ul style="list-style-type: none"> <li>select the last block of the active main program in "Modification" mode and enter "M2" or "M30"</li> </ul>
<b>ERROR 30</b>	<ul style="list-style-type: none"> <li>the tool radius is too large for the programmed contour angle</li> </ul>	<ul style="list-style-type: none"> <li>use tool with smaller radius</li> </ul>
<b>ERROR 31</b>	<ul style="list-style-type: none"> <li>the programmed radius is too small</li> </ul>	<ul style="list-style-type: none"> <li>radius must be increased</li> </ul>
<b>ERROR 40</b>	<ul style="list-style-type: none"> <li>in "MDI" mode G"41" or "G42" has been programmed</li> </ul>	<ul style="list-style-type: none"> <li>cutter radius compensation can not be used in "MDI" mode</li> </ul>
<b>ERROR 41</b>	<ul style="list-style-type: none"> <li>in "MDI" mode "G2", "G3", "G62" or "G63" have been entered without radius</li> </ul>	<ul style="list-style-type: none"> <li>circular interpolation is not possible in "MDI" mode without specifying the radius</li> </ul>
<b>ERROR 42</b>	<ul style="list-style-type: none"> <li>in "Teach In" mode an arc has been programmed as the first movement</li> </ul>	<ul style="list-style-type: none"> <li>insert a linear movement as the first movement</li> </ul>
<b>ERROR 45</b>	<ul style="list-style-type: none"> <li>the used parameter is zero</li> </ul>	<ul style="list-style-type: none"> <li>change the value</li> </ul>
<b>ERROR 46</b>	<ul style="list-style-type: none"> <li>the used parameter is negative</li> </ul>	<ul style="list-style-type: none"> <li>change the value</li> </ul>

**ERROR MESSAGES - CLEAR TEXT**

- ERR N-ADDR** ● N-address is incorrect
- PARITY** ● parity error in the program store
- ERR +X +Y +Z** ● hardware limit switches have been triggered in the displayed axis and the displayed direction  
*JOG OFF IN LOW JOG*
- SERVO XYZ** ● servo error in the displayed axis (axes)
- TRANSMIT** ● transfer error (sending) between sequencer and axis card (hardware fault)
- RECEIPT** ● transfer error (receiving) between sequencer and axis card (hardware fault)
- BREAK CY** ● axes are stopped at the end of the active block; in order to continue operation the interface signal "transfer allow" must be set to +24 V
- FEED HOLD** ● program sequence and all axis movements are stopped immediately. *DUE TO NO AIR SUPPLY TO MACHINE. OR INSUFFICIENT VACUUM OR DRILLING CYCLE IS IN OPERATION.*
- ERR S/M** ● programmed S- or M-output is already being used (this occurs when M- or S-functions are programmed in two consecutive blocks, and the M- or S-function programmed first has not been completely output yet)
- TRAVEL** ● software limit switches have been triggered, or the reference point was not set after power ON, or 3D has been programmed on a 2½ D version
- ERR MEM** ● fault with the part program memory (i.e. battery fault)
- AXES ALLOW** ● signal "Axis Allow" is not present at the interface
- TRAN ALLOW** ● signal "Transfer Allow" is not present at the interface
- FRAMING** ● bit format is wrong (during data in)
- UND. BLOC** ● the character is not allowed (during data in)

## OPERATION DIALOGUES

### QUESTIONS

- these questions are parts of dialogues which require decisions from the operator

### DISPLAY

DISPLAY	MEANING
MEM CLEAR?	● is the complete part program store to be cleared?
BLOC NO?	● enter block no.
PROG NO?	● enter main program number
SUBP = C	● enter sub-program number
REP = B	● enter repetition factor for the sub-program
ADR = A	● enter jump address (block number)
TOOL NO	● enter tool no.
REF X? (Y, Z)	● are the axes to travel to the reference point?
DWELL =	● enter dwell time (H1 = 0.1sec)
DEL M FUNCT?	● is the table with the M-codes to be cleared? (machine parameter)
READ MACH PAR	● are the parameters to be read in via the serial interface?
WRIT MACH PAR	● are the parameters to be output via the serial interface?
EDIT M-FUNCT	● "EDIT" mode for M-parameters?
LIST M-FUNCT	● the M-functions are listed one by one

### MESSAGES

- these messages provide important information to the operator

### DISPLAY

DISPLAY	MEANING
MEM CLEARED	● complete part program store has been cleared
MEM FULL ...	● the maximum storage capacity of the part program store has been reached
	● program data were not read in with the Baud rate with which they were output
MEM PROTECT	● access to the part program store is inhibited
MEM ENABLE	● access to the part program store is enabled
XYZ IN REF	● X, Y, Z axis have reached the reference point
HALT M RETURN	● an M-code has been output, but not yet acknowledged
NOT IN POS	● one or several axes have not reached the programmed position yet (no feed)
HALT AXIS REQ	● position request put to the axis control card has not been answered by this card
END OUTPUT	● data output is completed (reader, cassette)
END INPUT	● data input is completed (reader, cassette)

**NOTE:** the displays "NOT IN POS" and "HALT M RETURN" are only displayed after the actuation of the  key.