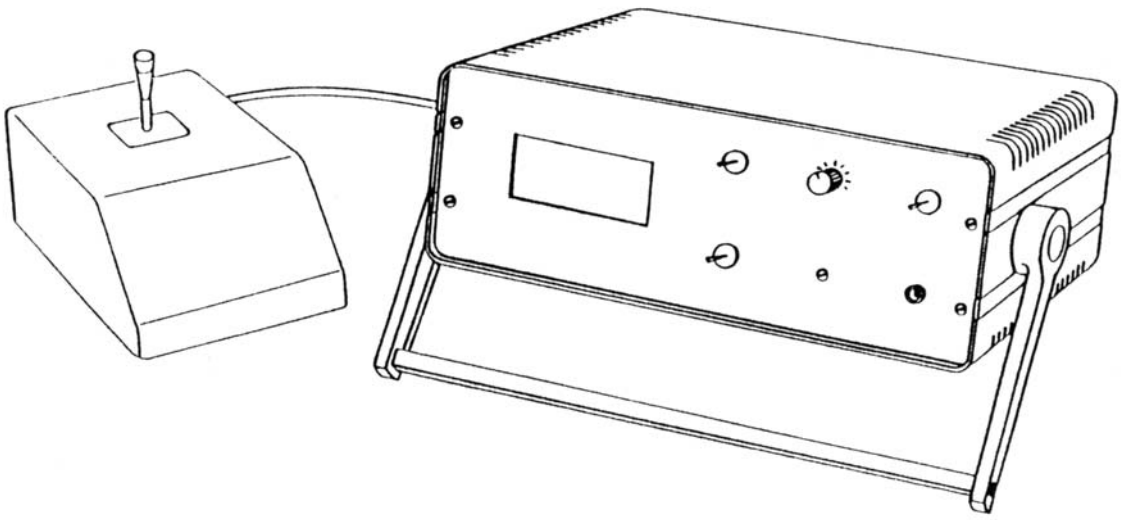
# Precision positioning system

# MCL



LANG GMBH & CO. KG
Dillstraße 4
D-35625 Hüttenberg
Tel. 06403/7009-0
Telefax 06403/7009-40
e-mail lang.Entwicklung@t-online.de

Please read before
beginning operation!!

### Interfacing the MCL with LANG Commander

The LANG Commander is used as a console device for MCL/MCC positioning systems. Supply voltage for the commander is 19V DC, not regulated. This voltage is applied to Pin 1 of the RS232 interface connector by Jumper J7 inside the MCL.
This voltage is applied by a jumper to pin 1, if a terminal was delivered with the controller.
MCL - 2 jumper (bridge) 19 from board no. 070192
MCL - 3 jumper (bridge) 12 from board no. 070592
If using LANG Commander, please be sure to use genuine interface cables supplied by LANG only.

### Power supply of MCL

The MCL can be supplied with either 100V - 120V or 200V - 240V. Please ensure that the voltage selector is set to the voltage appropriate to your mains supply voltage.
The voltage selector is located on the rear panel of the controller next to the mains inlet and power switch. The required voltage should align with the arrow on the rear panel. If it is not make the setting as follows:
1. remove the fuse-drawer from the rear panel
2. position the drawer so that your voltage aligns with arrow on panel
3. replace fuse-drawer into rear panel

Misalignment of voltage selector and mains supply voltage will result in damage to the controller and the mains fuse will blow.

### Ventilation openings

In order to ventilate the power amplifier stages ventilation apertures are built into the housing of the MCL. To prevent the system from damage, keep liquid, chips und conducting parts away

### Cleaning

For cleaning use a soft cloth dampened with a mild detergent and water solution. Please avoid using chemicals that contain benzine, acetone or similar solvents.

### Developing the controller

Lang does improve the systems and keeps it up to date, therefor Lang reserves the right for developing and modification.

**Changes regarding Eprom version 8**

**1. new operating element stop-Switch**

3 pole cable-box; by connecting pins 1 and 3 (for example via a so called STOP-switch) all motormovements are interrupted. RS232 interface is still active, in order to read out position data. Deconnecting pins 1 and 3 deactivates stop ( see chapter 2)

**2. improved motorspeed**

The motor speeds op position control system MCL are increased as follows:
MCL-2: $14s^{-1}$
MCL-3: $11s^{-1}$

**3. optionales CTS signal for serial interface RS232**

For safe reception, some PCs require an interval between two characters on the serial interface following each other. This interval can be adjusted ba register 12 according to table 7 ( see chapter 12).
Another possibility to avoid loss of characters is, to evaluate CTS Signal of the PC. Using this signal, the PC can prevent the MCL from transmitting characters. For compatibility reasons evaluation of CTS after power-on is deactivated. It can be activated though, by programming a dedicated register (see chapter 6).

**4. new command 'W'**

If there is a need for executing many relative vectors very fast, the new command 'w' may be helpful (see chapter 7.4)

TÜV Südwest
Fachbereich Gerätetechnik und Elektronik
Prüfzentrum Elektronik Sensorik Umweltsimulation
Dudenstraße 28, 68167 Mannheim

TÜV. SÜDWEST

# Prüfbescheinigung
Certificate of conformity / Certificat de conformité

## Nr. 70/007/10.D00107/95
Certificate No. / No. du certificat

| Auftragsdatum<br>Date of order/<br>Date de commande | Prüfbericht<br>test report / protocole d'essais | Ausstellungsdatum<br>Date of certificate / Date de certificat |
|---|---|---|
| 21.07.1994 | GEL3-EV-7.940028444 | 29.11.1995 |

Hiermit wird bescheinigt, daß nachfolgend genanntes Produkt die zu prüfenden Anforderungen gemäß aufgeführter Prüfgrundlagen erfüllt.
A sample of product has been tested and found to be in conformity with the below mentioned standards.
Un échantillon du produit a été essayé et trouvé conforme aux conditions d'essais ci-dessous.

| | | | |
|---|---|---|---|
| **Firma**<br>Client/Mandant | 3827879<br>Lang GmbH & Co. KG<br>Dillstraße 4<br>35625 Hüttenberg | **Fertigungsstätte**<br>Manufacturer/Constructeur | 3827879<br>Lang GmbH & Co. KG<br>Dillstraße 4<br>35625 Hüttenberg |
| **Erzeugnis**<br>Product/Produit | 3-Achsen-Mikroskoptisch-Steuerung | | |
| **Typ(en)**<br>Model, Type/<br>Modèle, Type | MCL-3 | | |
| **Beschreibung**<br>Description | Die 3-Achsen-Mikroskoptisch-Steuerung "MCL-3" erfüllt alle unten aufgeführten Prüfanforderungen. | | |
| **Prüfgrundlagen**<br>standards/<br>conditions d'essais | siehe unten | | |
| **Prüfungen** | EMV-Prüfung an Mikroskoptisch-Steuerungen | | |

| **Prüfergebnis(se)** | | | | |
|---|---|---|---|---|
| | ENV 50140 | 08.93 | Anforderung: | 10 V/m |
| | IEC 801-4 | 1988 | | 5 kV |
| | IEC 801-2 | 1991 | | 8 kV |
| | IEC 801-5/DIN VDE 0843 T.5 | | | 2 kV |
| | IEC 1000-4-11 | | | 100 % / 0,5 Perioden |
| | | | | 60 % / 8 Perioden |
| | | | | 30 % / 28 Perioden |
| | EN 55011 | 1991 | | Gruppe 1, Klasse B |
| | ENV 50141 | 08.93 | | 10 V |

**Mannheim, den 29.11.1995**

Prüfzentrum Elektronik Sensorik Umweltsimulation

H. Koch

GE-18 3/94 Ht, Stand 03/94

## Index

1 Functional Description
2 Operating Elements
3 Operation without PC
4 RS 232 Interface
5 Initial start of operation
- 5.1 Connections
- 5.2 Functional Test
- 5.3 Trouble shooting regarding RS232 interface

6 Table of Registers
- 6.1 Registers MCL2
- 6.2 Registers MCL3

7 Commands
- 7.1 Calibration
- 7.2 Measurement of maximum axes travel
- 7.3 Absolute positioning of axes
- 7.4 Relative positioning of axes
- 7.5 Activation of joystick
- 7.6 Break a RUN command

8 Adjusting the MCL
- 8.1 Number of revolutions/sec (speed)
- 8.2 Rampfunction of speed (acceleration)
- 8.3 Adaptation of resolution
- 8.4 Adaptation of leadscrew pitch
- 8.5 Setting of position counter
- 8.6 Disconnection of separate axes
- 8.7 Delaying echos
- 8.8 Adjustment of current reduction
- 8.9 Change of echo termination characters

9 Error messages
10 Inputs and outputs
11 Technical data
12 Appendix
- Sample program in GWBASIC for IBM-Pc
- Description of OPEN- command in GWBASIC
- Motor connection
- Test and adjustment
- Error recovery
- Transformer wiring
- Schematics

# 1. FUNCTIONAL DESCRIPTION

The stepping motor control MCL is able to drive the coordinate measuring instruments especially in automated manufacturing at a resolution of up to 0.0001mm. A unique feature of the control is a minimal motor noise. Due to the „dynamic micro stepping principle“ a resolution of 40,000 micro steps per motor revolution is achieved. Despite of this high resolution speeds of up to 9 rev/sec are possible using stepping motors with 200 steps.
The position control unit uses linear interpolation technique, so all axes reach the destination position at the same time. Limitation of acceleration is done by individually programmable ramp functions.
The MCL can either operate standalone using LANG commander and/or joystick or can be commanded by a PC-program. The LCD-display on the front panel is optional.
The instruction set of the MCL has been provided with new commands but is still compatible to its successfull predeccessor MCCxx1.
The commands to the MCL are given via registers. Each register can be written, rewritten and read. After initiating the „START“-command, the MCL executes all commands stored in the registers. At the end of execution MCL is ready to accept new data. This is signalled by the status display.
The synchronizing of the MCL with the PC is as follows:
- Command input by PC to the MCL
- „START“ command initiates execution
- Status display signals end of execution
- MCL is ready to accept new data

To avoid faults caused by malfunctions of the RS 232 interface, every character string sent from the PC to the MCL is preceeded by "U" (ASCII 0x55) and terminated by „carriage return“ (0xD). All characters before the „U" are ignored.

## Writing to registers

Registers are written to by sending their address2 and the desired register contents, both preceeded by „U“.
Example3

```
100 'Example in BASIC: "Write the register 0 with 12345"
110 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
120 PRINT#1,"U";CHR$(0);"12345"             'Write adr. register 0
```

## Reading registers

To read a register, the appropriate register number has to be sent. As response, the MCL transmits the register contents as an ASCII-string terminated by „CR“.
Example

```
100 'Example in BASIC: "Read the register 0"
110 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
120 PRINT#1,"U";CHR$(64)                    'Read adr. register 0
130 INPUT#1,MSG$                                'Receive data
140 PRINT MSG$                                  ' and indicate
```

## Initiating a START command

Starting the MCL is done by reading the start register (register 16). Having performed all commands which were stored in its registers, the MCL responds with a status message to the PC. This message indicates, that new commands can be transmitted by the PC.
Example

```
100 'Example in BASIC: "Start the MCL"
110 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
120 PRINT#1,"U";CHR$(80)                    'Start
```

The MCL can process both 7-bit and 8-bit ASCII-codes and is compatible with MCCxx without any special adaptations. The register numbers are accepted no matter if bit 7 is equal to 0 or 1.

1 MCCxx is available with up to 4 axes and output current ratings of 1.5A, 2.5A, 4.0A and 6.0A.
2 register addresses are summarized in chapter 6
3 all examples in this manual are written in GWBASIC. They are designed to run on a PC.

## Default values after RESET

The basic adjustment of the MCL after switching-on or reset is described in chapter 6. Positions are displayed in the dimension of 0.1micrometer in case of a leadscrew pitch of 4.0mm.
For a quiet operation and a safe positioning, the motors with a step angular error of ≤ 3% are to be used. For safe operation LANG recommends the following motor type: ZSS 42.200.1,2 (PHYTRON)
Other motor types can be used as well, however some specification (e.g. maximum speed) may not be achieved.

**Reduction of motor loss**

To avoid unnecessary heating of the motorwindings, the MCL decreases phase currents down to 50% of the nominal value at every intermission (see chapter 8.8).

## 2 OPERATING ELEMENTS

The display and all the operating elements except the power switch are positioned on the front panel.

| Operating element | Signification |
|---|---|
| CLEAR X/Y/Z | Switch to zero the position registers and the display; x, y and z axis individually |
| SPEED 1..10 | Potentiometer to adjust the motor speed if operating with external clock. The value which was written to register 9 (motor speed), can be adjusted from 0 to 100%. |
| JOYSTICK H/A | Joystick selector switch<br>H = manual operation (PC is inactive)<br>A = automatic operation with commands according to chapter 7 |
| RESET | press upward initiate RESET |
| ON | equipment-on indicator lamp |
| LCD DISPLAY (optional) | LCD display with 4*16 signs to indicate operating mode and absolute position values P: $-99{,}999{,}999.9 \leq P \leq +99{,}999{,}999.9$. |
| STOP (optional) | 3 pole cable-box; by connecting pins 1 and 3 (for example via a so called STOP-switch) all motormovements are interrupted. RS232 interface is still active, in order to read out position data.<br>**Attention:** Initiating the command „Calibrate“ in this situation will reset all apsolute position registers to zero. |

table 1: operating elements located on the frontpanel of MCL

| | | |
|---|---|---|
| Joystick switch in position A | | |
| | BAUD RATE LESEN<br>(Read the baud rate) | MCL adjusts the baud rate automatically, using SPACE-character (0x20); see chapter 4, RS232 interface |
| | EMPFANGSBEREIT<br>(Ready to receive) | MCL is waiting for commands via RS232 interface |
| | POSITION FAHREN<br>(go to position) | MCL moves the axes to the desired absolute position |
| | RELATIVE GERADE<br>(relative straight line) | starting at the current position, MCL executes a relative positioning cycle according to the desired distance |
| | CALIBRIEREN<br>(Calibration) | MCL moves all axes to zero position |
| | TISCHSCHLAENGE | MCL moves all axes to their maximum position |
| | JOYSTICK AUTO | MCL operates under joystick control |
| Joystick switch in position H | | |
| | JOYSTICK HAND | MCL operates under joystick control without PC |

table 2: operation modes of MCL

# 3 OPERATION WITHOUT PC

With the MCL, the execution of simple movements is feasible without PC. For that purpose the joystick switch has to be set on "H". Then any position can be reached using the joystick. The current absolute position is displayed permanently. All axes can be zeroed individually pressing the CLEAR switch.

# 4 RS 232 INTERFACE

Serial interface RS 232, (300 to 19,200bd for MCL2 and 300 to 9600bd for MCL3) Default values: 2400bd, 11 bit frame (1 startbit, 8 databits, no parity, 2 stop bits)
For safe operation a RS 232 interface consisting of following signals is required:

| | |
|---|---|
| RxD | receiver line of the MCL (transmission line of the PC) |
| TxD | transmission line of the MCL (receiver line of the PC) |
| RTS | Request to send from MCL |
| GND | Signal ground |
| CTS | Clear to send from PC4 |

A restricted operation without RTS is possible, please see chapter 5.3.

## Automatic baud rate tuning

In operation mode "AUTOBAUD", an automatic baud rate tuning is performed by MCL. To that purpose, the PC is required to transmit a SPACE character (0x20) to MCL, following every power-on. In this case, the display shows the message "BAUDRATE LESEN". This message is cleared, as soon as the baud rate has been tuned.

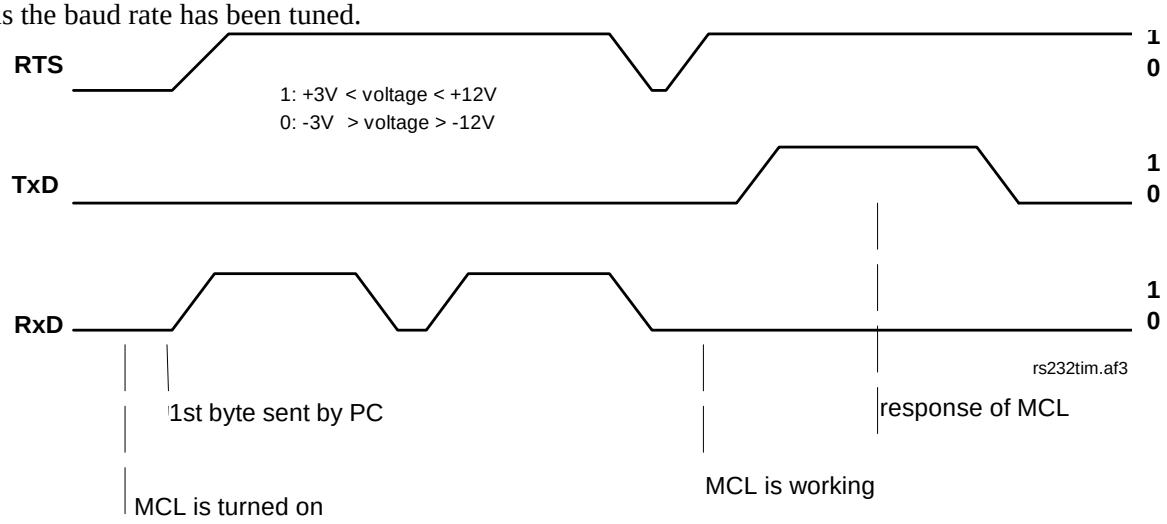



figure: signal level of RS 232 on the conduction side

4 is to be activated via content of register 17: 0 = CTS is deactivated; 1 = CTS is active; content of register 17 defaults to 0

# 5 Initial start of operation

**ATTENTION**: The ventilation apertures on the back panel of control unit **must not be** covered!

## 5.1 CONNECTIONS

- connect motors using delivered cables
- connect the joystick
- connect the computer via interface cable
- connect the mains

## 5.2 FUNCTIONAL TEST

- switch-on the MCL
- joystick switch in position "H"
- move joystick in every direction:

The motor should turn depending on steering of the joystick. If there is no reaction, then please check the connections of the motors and joystick. If all connections seem to be OK, then the unit should be checked for hidden damages during transport.

- Joystick switch in position "A"
- Function call according to chapter 7 and given program examples

Example

```
100 'Example in BASIC
110 'Calibrate the connected table
120 ‘The command "C" is placed into the command register (register 7)
130 ' automatically after switch-on or after reset
140 OPEN "com2:2400,n,8,2,ds0" AS #1          'Open channel
150 PRINT#1,"U";CHR$(80)                      'Start
160 'waiting for the end of calibration and MCL status message
170 INPUT#1,MSG$ : PRINT MSG$                 'Status: "AAA."
```

## 5.3 Trouble shooting regarding RS232 interface

• MCL does not respond via RS 232
- Test all pin connections and interface cable; see voltage level given in chapter 4
- Test interface conditions (OPEN command) in PC
•Some bytes of messages of the MCL are lost:
- Delay the echo of MCL (see chapter 8.7)
- there is no CTS line available at the PC side of the interface (RTS line of MCL is not evaluated by PC)
Any time the MCL works and it does not receive any data, the interface will be locked using RTS. In case the PC waits for status message of MCL, a synchronisation of the PC and MCL is done without checking the RTS line (see examples in this manual)
However, problems can arise, if following commands are used:

Adaptation of resolution, see chapter 8.3

Adaptation of leadscrew pitch, see chapter 8.4

because the MCL does not transmit status messages after adjustments are executed. In this case, the PC must be delayed, for example, using loops, in order to avoid the loss of data and commands.

# 6 Table of Registers

## 6.1 Registers MCL2

| register nr. | read address | write address | basic description | default value |
|---|---|---|---|---|
| 0 | 64 | 0 or 255 | preselection X | 0 |
| 1 | 65 | 1 | preselection Y | 0 |
| 2 | - | - | unused | 0 |
| 3 | 67 | 3 | absolute position X | 0 |
| 4 | 68 | 4 | absolute position Y | 0 |
| 5 | - | - | unused | 0 |
| 6 | 70 | 6 | status | OK... |
| 7 | 71 | 7 | command | c |
| 8 | 72 | 8 | ramp 1...99 | 50 |
| 9 | 73 | 9 | motor speed 0...150 | 50 |
| 10 | 74 | 10 | current reduction 0...10 | 5 |
| 11 | 75 | 11 | mask 0...3 | 3 |
| 12 | 76 | 12 | delay time for echos from MCL | 2 |
| 13 | 77 | 13 | leadscrew pitch X | 40,000 |
| 14 | 78 | 14 | leadscrew pitch Y | 40,000 |
| 15 | 79 | 15 | resolution | 10 |
| 16 | 80 | - | start | - |
| 17 | 81 | 17 | activate CTS: „1“<br>deactivate CTS: 0 | 0 |

## 6.2 REGISTER-MCL3

| register Nr. | read adress | write address | basic description | default value |
|---|---|---|---|---|
| 0 | 64 | 0 or 255 | preselection X | 0 |
| 1 | 65 | 1 | preselection Y | 0 |
| 2 | 66 | 2 | preselection Z | 0 |
| 3 | 67 | 3 | absolute position X | 0 |
| 4 | 68 | 4 | absolute position Y | 0 |
| 5 | 69 | 5 | absolute position Z | 0 |
| 6 | 70 | 6 | status | OK... |
| 7 | 71 | 7 | command | c |
| 8 | 72 | 8 | ramp 1...99 | 50 |
| 9 | 73 | 9 | motor speed 0...110 | 50 |
| 10 | 74 | 10 | current reduction 0...10 | 5 |
| 11 | 75 | 11 | mask 0...7 | 7 |
| 12 | 76 | 12 | delay time for replies | 2 |
| 13 | - | - | unused | ERR 2 |
| 14 | - | - | unused | ERR 2 |
| 15 | - | - | unused | ERR 2 |
| 16 | 80 | - | start | - |
| 17 | 81 | 17- | activate CTS: „1“<br>deactivate CTS: 0 | 0 |
| 18 | - | - | unused | ERR 2 |
| 19 | - | - | unused | ERR 2 |
| 20 | - | - | unused | ERR 2 |
| 21 | 85 | 21 | leadscrew pitch (pitch) X | 40.000 |
| 22 | 86 | 22 | leadscrew pitch (pitch) Y | 40.000 |
| 23 | 87 | 23 | leadscrew pitch (pitch) Z | 40.000 |
| 24 | - | - | unused | ERR 2 |
| 25 | 89 | 25 | resolution | 10 |

# 7 COMMANDS

Movements of the MCL are initiated through the transfer of position data, writing of commands in register 7 and START command subsequently. Following commands are available:
a: terminate a RUN command
c: start calibration, move the axes, until limit switches are reached
e: move axes to an absolute position with external clock
g: execute a relative vector with external clock starting at the current position
j: activate the joystick in combination with position counting
l: measure distance from current position up to the limit switches
m: set mask (compatible to MCCxx)
p: set position (compatible to MCCxx)
r: move axes to an absolute position with internal clock
s: activate the joystick without position counting
v: execute a relative vector with internal clock starting at the current position
Having executed these commands, except „i“ and „s“, MCL transmits a status message. This message consists of 5 bytes:

byte 1: status of limit switches for the X axis
byte 2: status of limit switches for the Y axis
byte 3: status of limit switches for the Z axis
byte 4: „--"
byte 5: „--“

The status of the limit switches is coded according to the following table:

| binary | ASCII | description |
|---|---|---|
| 01000000 | @ | no limit switch is touched |
| 01000001 | A | limit switch zero position |
| 01000100 | D | limit switch end position |

table 3: status of the limit switches, coding of byte 1 to 3

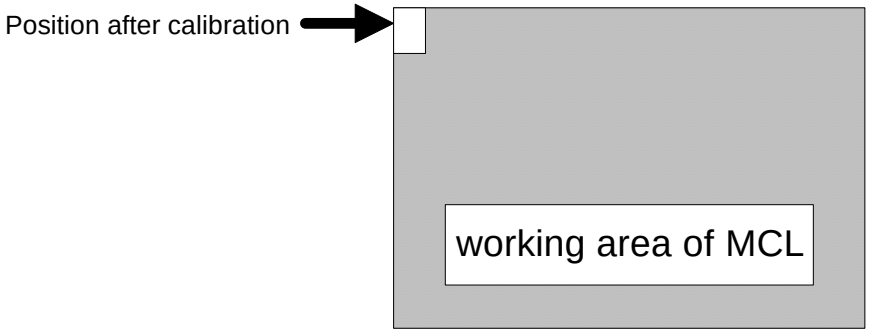
## 7.1 CALIBRATION

The command "calibrate" leads to a movement of all axes to the zero position until the limit switches are reached. At this moment, the absolute position registers (registers3 to 5) are cleared.

Example

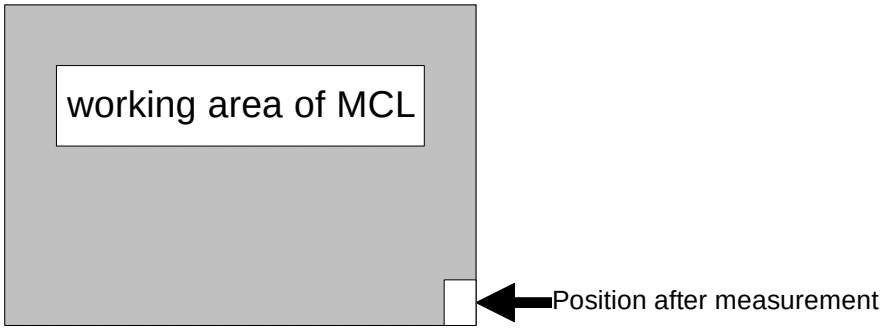
```
100 'Example in BASIC: "Calibrate the table"
110 OPEN "com2:2400,n,8,2,ds0" AS #1          'Open channel
120 PRINT#1,"U";CHR$(7);"c"                   'Command "calibrate"
130 PRINT#1,"U";CHR$(80)                      'Start
140 INPUT#1,MSG$ : PRINT MSG$                 'Wait of status "AA--."
```

If a table is connected to the motors of axes x and y, after calibration it has reached position shown below.



## 7.2 Measurement of maximum axes travel

The table is moved to the limit switch position which is opposite the zero position. The distance can be read from registers 3 to 5 (absolute position). This command is to be used after a calibration process only.




After completion of the measurement, the table has reached the position shown above.

Example

```
100 'Example in BASIC: "Measure the movement of the table"
110 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
120 PRINT#1,"U";CHR$(7)";l"                 'Command "measure length"
130 PRINT#1,"U";CHR$(80)                    'Start
140 INPUT#1,MSG$                            'Waiting of READY
150 PRINT MSG$                                    'Status "DD--."
160 'Display the absolute position
170 PRINT#1,"U";CHR$(67)                    'Display the X position
180 INPUT#1,MSG$ : PRINT "X = ";MSG$
190 PRINT#1,"U";CHR$(68)                    'Display the Y position
200 INPUT#1,MSG$ : PRINT "Y = ";MSG$
210 PRINT#1,"U";CHR$(69)                    'Display the Z position
220 INPUT#1,MSG$ : PRINT "Z = ";MSG$
```

## 7.3 Absolute positioning of axes

After loading the destination position into the preselection registers (register 0 to 2) and transmitting the START command, the MCL moves along a line between the current and the desired position. The absolute value of the position must not exceed the range of

$$\left|positionvalue\right| < \frac{9.9 \cdot 10^{12}}{A \cdot S}$$

where is A: resolution (register 15, see chapter 8.3)
S: value for leadscrew pitch (register 13 and 14, see chapter 8.4)

Position data are represented as signed integers. After the adaptation of resolution (chapter 8.3), the position values can be entered in micrometer, millimetre or other dimensions. The position value transmitted by the MCL is always rounded off.

If the table reachs a limit switch before the desired position, all axes stop and the current position will be stored in registers 3 to 5 (absolute position).
The absolute position can be reached using internal clock (command "r") or external clock (command "e") according to the following examples. At any time, the movement can be broken off with an "a" (see chapter 7.6).

Example for MCL2

```
100 'Example in BASIC: "Moving to an absolute position"
110 'The velocity of the table is evaluated using
120 'internal clock which is adjusted in register 9
130 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
140 PRINT#1,"U";CHR$(7)"r"                  'Running with int.clock
150 PRINT#1,"U";CHR$(0);"10000"             'Position X=10000
160 PRINT#1,"U";CHR$(1);"20000"             'Position Y=20000
170 PRINT#1,"U";CHR$(80)                    'Start
180 INPUT#1,MSG$ : PRINT MSG$               'Waiting for READY
```

Example

```
100 'Example in BASIC: "Starting-up an absolute position"
110 'The velocity of the table is evaluated using
120 'both internal clock which is set in register 9
130 'and "SPEED" selector (Potentiometer)
140 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
150 PRINT#1,"U";CHR$(7)"e"                  'Running with ext.clock"
160 PRINT#1,"U";CHR$(0);"10000"             'Position X=10000
170 PRINT#1,"U";CHR$(1);"20000"             'Position Y=20000
180 PRINT#1,"U";CHR$(80)                    'Start
190 INPUT#1,MSG$ : PRINT MSG$               'Waiting for READY
```

## 7.4 Relative positioning of axes

The vector is loaded to the preselection registers (register 0 to 2). After the START command, the MCL executes the movement and calculates the value of the endposition. This value is stored as an absolute position (register 3 and 4). The absolute value of the position must not exceed the range of

$$|positionvalue| < \frac{9.9 \cdot 10^{12}}{A \cdot S}$$

where is A: resolution (register 15, see chapter 8.3)
S: value for leadscrew pitch (register 13 and 14, see chapter 8.4)

Position data are represented as signed integers. After the adaptation of resolution (chapter 8.3), the position values can be entered in micrometer, millimetre or other dimensions. The position value transmitted by the MCL is always rounded off.
If the table reachs a limit switch before the desired position, all axes stop and the current position will be stored in registers 3 to 5 (absolute position).
The movement can be executed using internal clock (command "v") or external clock (command "g") according to the following examples. At any time, the movement can be broken off with an "a" (see chapter 7.6).

Example for MCL2

```
100 'Example in BASIC: "Running a relative vector"
110 'The velocity of the table is evaluated with
120 'internal clock which is adjusted in register 9
130 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
140 PRINT#1,"U";CHR$(7)"v"                  'Rel. vector int.clock"
150 PRINT#1,"U";CHR$(0);"-5000"             'Vector X=-5000
160 PRINT#1,"U";CHR$(1);"2000"              'Vector Y=2000
170 PRINT#1,"U";CHR$(80)                    'Start
180 INPUT#1,MSG$                            'Waiting for READY
190 PRINT MSG$                                    'Status
```

Example

```
100 'Example in BASIC: "Running a relative vector"
110 'The velocity of the table is evaluated with
120 'intern clock which is adjusted in regigter 9
130 'and with using of the "SPEED"selector
140 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
150 PRINT#1,"U";CHR$(7)"e"                  'Rel. vector ext.clock"
160 PRINT#1,"U";CHR$(0);"-5000"             'Vector X=-5000
170 PRINT#1,"U";CHR$(1);"2000"              'Vector Y=2000
180 PRINT#1,"U";CHR$(80)                    'Start
190 INPUT#1,MSG$                            'Waiting of READY
200 PRINT MSG$                                    'Status
```

## 7.5 Activation of joystick

The joystick can be used in two operating modes:

Joystick selector switch on "H"

In this case, the PC switched-off. That means:
- the table can be moved until the limit switches are reached
- the position is incremented/decremented and displayed
- the interface is locked (RTS is set), all the data from the PC will be lost

Joystick selector switch on "A"

In this case, the joystick can be switched-on a command in two different ways.
a) activation of the joystick with position counting (command "j"):
- The table can be moved inside of the predefined working area only
- The position is incremented/decremented and displayed

- No other commands are executed until the joystick is turned off

b) activation of the joystick without position counting (command "s"):

- the table can be moved inside of the predefined working area only
- the position is not incremented/decremented and not displayed
- no other commands are executed until the joystick is turned off

The PC can turn off the joystick by sending a "j" without preceding "U" (cf. following examples).

Example

```
100 'Example in BASIC: "activation of the joystick with pos. counting"
110 OPEN "com2:2400,n,8,2,ds0" AS #1          'Open channel
120 PRINT#1,"U";CHR$(7);"j"                        'Joystick on"
130 PRINT#1,"U";CHR$(80)
140 PRINT "Joystick is active"
150 PRINT "Switch-off with <j>"
160 I$=INPUT$(1) : IF I$<>"j" THEN 160'
170 PRINT#1,"j"                                    'Joystick off"
180 INPUT#1,MSG$ : PRINT MSG$              'waiting for status
```

Example

```
100 'Example in BASIC: "activation of  joystick without pos. counting"
110 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
120 PRINT#1,"U";CHR$(7);"s"                 'Joystick on"
130 PRINT#1,"U";CHR$(80)
140 PRINT "Joystick is active"
150 PRINT "Switch-off with <j>"
160 I$=INPUT$(1) : IF I$<>"j" THEN 160'
170 PRINT#1,"j"                             'Joystick off"
180 INPUT#1,MSG$ : PRINT MSG$               'waiting for status
```

### Reading the position registers during joystick operation

This function can be used if the selector switch is in position "A", that is, if the joystick has been activated by the PC with position incrementing/decrementing. The readout of position registers is done analogous to the operation without joystick.

Example

```
100 'Example in BASIC: "Reading absolute position X"
110 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
120 PRINT#1,"U";CHR$(7);"j"                 'Joystick on"
130 PRINT#1,"U";CHR$(80)
140 PRINT#1,"U";CHR$(67)                    'Read-out absolute pos. X
150 INPUT#1,MSG$    '"
160 PRINT MSG$      '"
```

For a faster readcycle, a shortened command-form is allowed in joystick mode. The read command is accepted without the single-phased "U" and the terminating „CR“. This way is used in the following example.

Example for the MCL2

```
100 'Example in BASIC:
110 'switchs on the joystick and indicates the
120 'X and Y positions constantly on the display
130 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
140 PRINT#1," ";                            'Space to baud rate-recogn.
150 PRINT#1,"U";CHR$(12);"0"                'no delay
160 PRINT#1,"U";CHR$(7);"j"                 'Joystick on"
170 PRINT#1,"U";CHR$(80)
180 PRINT "Joystick is active"
190 PRINT "Switch-off with <j>"
200 PRINT " Positions X,Y :"
210 PRINT#1,CHR$(67);                       'Absolute position X
220 INPUT#1,MSGX$
230 PRINT#1,CHR$(68);                       'Absolute position Y
240 INPUT#1,MSGY$ '"
250 LOCATE 3,17                             'Indicate position
260 PRINT RIGHT$(""+MSGX$,8);
270 PRINT RIGHT$(""+MSGY$,8);
280 IF INKEY$ <> "q" THEN GOTO 210
290 PRINT#1,"j";                            'Switch-off joystick
300 INPUT#1,MSG$    'get state signal
```

## 7.6 Terminate a RUN command

The run commands "move axes to an absolute position" and "execute a relative vector" can be broken off any time with command "a" without preceding "U". After the command "a" the motors are stopped, the position registers (register 3 to 5) are loaded with current position and the MCL is ready to receive new commands. The position is displayed.

Example for MCL2

```
100 '*******move to an absolute position ********
110 OPEN "com2:2400,n,8,2,ds0" AS #1                    'Open channel
120 PRINT : PRINT "Move to an absolute position "
130 PRINT "Position X,Y "; : INPUT X$,Y$ : K$="r"
140 PRINT#1,"U";CHR$(0);X$;CHR$(13);"U";CHR$(1);Y$
150 PRINT#1,"U";CHR$(7);K$;CHR$(13);"U";CHR$(80)
160 PRINT "break off with "A"
170 I$=INKEY$
180 IF IN$="a" OR IN$="A" THEN PRINT#1,"a"
190 IF LOC(1) = 0 THEN 170
200 INPUT#1,MSG$ : PRINT MSG$
```

Example

```
100 '******* Run vector ******************************
110 OPEN "com2:2400,n,8,2,ds0" AS #1                    'Open channel
120 PRINT : PRINT "RUN VECTOR"
130 PRINT "Straight X,Y "; : INPUT X$,Y$ : K$="v"
140 PRINT#1,"U";CHR$(0);X$;CHR$(13);"U";CHR$(1);Y$
150 PRINT#1,"U";CHR$(7);K$;CHR$(13);"U";CHR$(80)
160 PRINT "break off with "A"
170 I$=INKEY$
180 IF IN$="a" OR IN$="A" THEN PRINT#1,"a"
190 IF LOC(1) = 0 THEN 170
200 INPUT#1,MSG$ : PRINT MSG$
```

# 8 ADJUSTMENT OF MCL

## 8.1 Number of revolutions/sec (speed)

The speed of the motors is adjustable in stages (st) from 0.01 rev/sec (stage0) up to 9 rev/sec (stage 90, MCL3), and 12 rev/sec (stage 120, MCL2). The upper speed-ranges can be reached with optimal tuning of motors and mechanics only. Except stage 0, the number of revolution (n) can be calculated according to the given equation: $n = st * 0.1$

| stage | speed [rev/sec] | stage | speed [rev/sec] | stage | speed [rev/sec] |
|---|---|---|---|---|---|
| 0 | 0.01 | 20 | 2.0 | 120 | 12 |
| 1 | 0.1 | 30 | 3.0 | 130 | 13 |
| 2 | 0.2 | 90 | 9.0 | 140 | 14 |
| 9 | 0.9 | 100 | 10 | 150 | 15 |
| 10 | 1.0 | 110 | 11 | | |

table 4: stage versus speed

Example
```
100 'Example in BASIC: "Adjust the number of revolution (speed)"
110 OPEN "com2:2400,n,8,2,ds0" AS #1                'Open channel
120 PRINT#1;"U";CHR$(9);"90"                        'Speed stage 90
```
As shown in table 5, the speed in joystick mode can be adjusted changing the speed register (register 9) also. A lower value allows a finer positioning in lower speed ranges.

| Register value | speed [rev/sec] |
|---|---|
| 2 | 0.00016 |
| 10 | 0.00033 |
| 30 | 0.041 |
| 50 | 0.61 |
| 70 | 3.6 |
| 90 | 5.5 |

table 5: speed register value versus actual speed in joystick mode

## 8.2 Rampfunction of speed (acceleration)

The acceleration ramps of motors are adjustable in stages 1 to 99. Stage 1 will result with the lowest acceleration, that is the flattest ramp. Accordingly, stage 99 corresponds to steepest ramp.
Example
```
100 'Example in BASIC: "Adjust the ramp"
110 OPEN "com2:2400,n,8,2,ds0" AS #1                'Open channel
120 PRINT#1;"U";CHR$(8);"90"                        'Ramp stage 90
```

## 8.3 Adaptation of resolution

The position data have to be transmitted as signed integers to the MCL. Setting the appropriate resolution, the integers are adapted to the respective needs. The resolution (A) is set in multiple of 0.0001mm. A resolution of 10, for example, corresponds to an input in micrometers.
range of possible input values: all integer numbers from 1 up to (0.1*S), where S stands for leadscrew pitch (see chapter 8.4).
Adjusting the resolution changes the value of absolute position immediately. However, the zero position remains unchanged.
Example for MCL2
```
300 'Example in BASIC: "adaptation of resolution"
310 'The resolution changes the value of absolute position immediately
320 OPEN "com2:2400,n,8,2,ds0" AS #1                'Open channel
330 PRINT#1;"U";CHR$(15);"5"                        'Resolution 0.0005 mm
```

## 8.4 Adaptation of leadscrew pitch

The leadscrew pitch can be adapted for any axis individually. The input of leadscrew pitch (S) occurs in multiples of 0.0001 mm. The value S=40,000 thereby yields a leadscrew pitch of 4.0mm.
range of possible input values: $1000 \leq S \leq 100000$
Adjustment of leadscrew pitch changes the value of absolute position immediately. However, the zero position remains unchanged.
Example for MCL2

```
400 'Example in BASIC: "adaptation of  leadscrew pitch"
410 'The leadscrew pitch changes the value of the absolute position immediately
420 OPEN "com2:2400,n,8,2,ds0" AS #1          'Open channel
430 PRINT#1;"U";CHR$(13);"10000"              'Spindle lead X 1 mm
440 PRINT#1;"U";CHR$(14);"10000"              'Spindle lead Y 1 mm
```

## 8.5 Setting of position counter

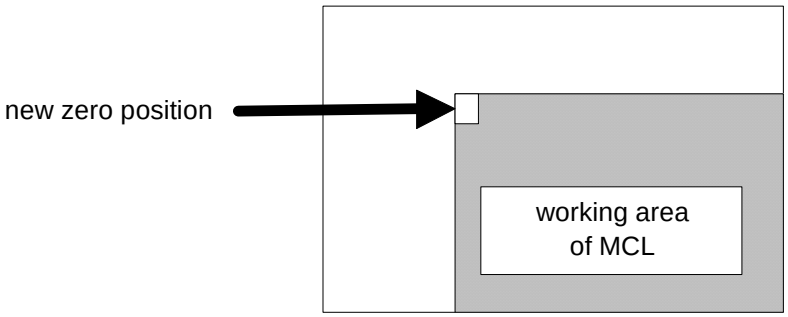
The travel length of the table is limited by setting the absolute position register (register 3 to 5) on an initial value. The absolute value of the entered position must not exceed the range of

$$\left|positionvalue\right| < \frac{9.9 \cdot 10^{12}}{A \cdot S}$$

where is A: resolution (register 15, see chapter 8.3)
S: value for leadscrew pitch (register 13 and 14, see chapter 8.4)

The absolute position can be set in two ways:

- A write operation to the registers 3 to 5 leads to a new counter reading. This new absolute position is displayed by the MCL at once. START commands, that are initiated thereafter, refer to this position.
- By entering new position values into preselection registers and accepting with command "p", the absolute position is set with new value as mentioned above. This command is implemented for compatibility to MCCxx only. It shall therefore not be used for new program developments.




Example for MCL2

```
100 'Example in BASIC: "Set a position counter"
110 OPEN "com2:2400,n,8,2,ds0" AS #1        'Open channel
120 PRINT#1,"U";CHR$(3);"-5000"             'New position X
130 PRINT#1,"U";CHR$(4);"2000"              'New position Y
```

## 8.6 Disconnection of separate axes

By the help of a binary mask, the axes can be switched-off individually. This can be helpful, for example, if an axis does not have a limit switch, but shall be calibrated with another axis. The mask is adjustable in two ways:

- Writing the mask to register 11
- By entering new mask value into register 6 and accepting with command "m", the mask is set with new value as mentioned above. This command is implemented for compatibility to MCCxx only. It shall therefore not be used for new program developments.

**ATTENTION: It is strictly forbidden, to switch off all axes at the same time!**

| content of mask register | effect on X axis | effect on Y axis | effect on Z axis |
|---|---|---|---|
| 0 | forbidden | forbidden | forbidden |
| 1 | 1 | 0 | 0 |
| 2 | 0 | 1 | 0 |
| 3 | 1 | 1 | 0 |
| 4 | 0 | 0 | 1 |
| 5 | 1 | 0 | 1 |

| | | | |
|---|---|---|---|
| 6 | 0 | 1 | 1 |
| 7 | 1 | 1 | 1 |

table 6: masking of axes; 0: Axis is switched-off; 1: Axis is switched-on; values > 3 are ignored from the MCL2; values > 7 are ignored from the MCL3

Example

```
500 'Example in BASIC: "Activate the axes"
510 OPEN "com2:2400,n,8,2,ds0" AS #1          'Open channel
520 PRINT#1,"U";CHR$(11);"3"                  'X and Y axes active
```

## 8.7 Delaying feedback messages

For safe reception, some PCs require an interval between two characters on the serial interface following each other. The period of this delay is adjusted by register 12 according to table 7. The default value after power-on and reset is 4ms.

| content of register 12: | interval period [ms] |
|---|---|
| 0 | 0 |
| 1 | 2 |
| 2 | 4 |
| 3 | 6 |
| 9 | 18 |

table 7: adjusting the delay of characters following each other. possible range of input values: 0 to 9

Example

```
600 'Example in BASIC: "Delay for echos"
610 OPEN "com2:2400,n,8,2,ds0" AS #1          'Open channel
620 PRINT#1,"U";CHR$(12);"0"                  'Delay = 0
```

## 8.8 Adjustment of current reduction

If the MCL is non-operative, motor currents are reduced automatically. This current reduction prevents an unnecessary heating of the motor windings. As a disadvantage of a current reduction, slight position deviations caused by current change in the motors can occur. In order to adjust the current reduction to individual demand, register 10 must be set. The value, the current is to be reduced to, is calculated as follows:

$$I_{idle} = \frac{I_{normal} \cdot [content\ of\ register10]}{10}$$

where $I_{normal}$ = nominal (rated) current

Register 10 can be adjusted with unsigned integer in the range of 0 to 10. If register10 is set to 0, the motors currentless in non-operative state. If register10 is set to 10, the current remains unchanged. The default value of register10 after power-on is 5; e.g the current is cut by half in non-operative state.

## 8.9 Change of echo termination characters

As a default setting after power-on and reset, all the commands are terminated by „CR“ (0xD). This terminating sign can be changed by writing to register 16.
Change of terminating sign must occur, **before** the first echo from the MCL is demanded. If the first echo of the MCL has been transmitted, write cycles to register 16 are not executed and the error message "ERR 4" will occur.

Example

```
700 'Example in BASIC: " change termination character to line feed <LF>"
710 OPEN "com2:2400,n,8,2,ds0" AS #1                'Open channel
720 PRINT#1,"U";CHR$(16);CHR$(10)                        '<LF>
```

# 9 ERROR MESSAGES

In case of errors, following error messages occur:

| | |
|---|---|
| ERR 1 | wrong command |
| ERR 2 | wrong register shall be read |
| ERR 3 | wrong data |
| ERR 4 | wrong register shall be written to |
| ERR 5 | statusmessage instead of mask in status register |
| ERR 6 | wrong value (0 or >3) in mask register of MCL2<br>wrong value (0 or >7) in mask register of MCL3 |

# 10 INPUTS AND OUTPUTS

## 10.1 MOTOR CONNECTION X/Y/Z

| Pin Nr. | Line colour | 12-pole cable box, motor | pin assignment: |
|---|---|---|---|
| 1 + 9 | blue | K | phase 1R |
| 2 + 10 | pink | J | phase 1T |
| 3 + 11 | white | B | phase 2T |
| 4 + 12 | brown | C | phase 2R |
| 5 | yellow | G | limit switch end pos. |
| 6 | grey | H | limit switch zero pos. |
| 7 | red | A | +5V |
| 8 | black | F | GND |
| 13 | green | E | limit switch end pos. |
| 14 | violet | D | limit switch zero pos. |
| 15 | | | |

Pin assignment of connecting socket (15-pole D-Sub-connector, MCL)

## 10.2 RS 232 CONNECTION

| Pin Nr. | signal name | comment |
|---|---|---|
| 1 | either n.c. or 19V DC, non regulated achieved by jumper J7 | |
| 2 | RxD | Receiver line of MCL |
| 3 | TxD | Tranmission line of MCL |
| 4 | n.c. | |
| 5 | GND | Signal ground (frame) |
| 6 | n.c. | |
| 7 | RTS | Request to send, from MCL |
| 8 | CTS (optional) | Clear to send, from PC |
| 9 | n.c. | |

Pin assignment of connecting socket (9 pole)

## 10.3 Interfacecable MCL to PC

| MCL | | IBM PC | |
|---|---|---|---|
| 9 pole connector | pin assignment | 9 pole connector | pin assignment |
| 1 | n.c. | - | - |
| 2 | RxD | TxD | 3 |
| 3 | TxD | RxD | 2 |
| 4 | n.c. | - | - |
| 5 | GND | GND | 5 |
| 6 | n.c. | - | - |
| 7 | RTS | CTS | 8 |
| 8 | CTS | RTS | 7 |
| 9 | n.c. | - | - |

## 10.4 Joystick connection

| | | | |
|---|---|---|---|
| | → | 8,9 | VAref (+5V) |
| X ----- | → | 3 | X-axis |
| Y---------- | → | 4 | Y-axis |
| Z---------- | → | 5 | Z-axis |
| | → | 1,2 | GND (0V) |

## 11 Technical data

| | |
|---|---|
| power supply: | 100V - 120V / 200V - 240V +/-10%;<br>50/60 Hz, 70 VA |
| fuses: | |
| primary winding (fuse socket): | 0.8A time-lag/1.6A time-lag |
| secondary (on the circuit board) | |
| fu1 (Logic voltages): | 0.5A, time-lag |
| fu2 (Logic voltages): | 0.5A, time-lag |
| fu3 (Motor voltage): | 5.0A, time-lag |
| fu4 (Motor voltage): | 5.0A, time-lag |
| maximum power failure period: | <50 ms; power failure (< 0.77 nominal voltage) results in RESET condition for MCL |
| maximum speed: | MCL3: 11 rev/sec<br>MCL2: 15 rev/sec<br>using motors with 200steps/rev |
| maximum motor current: | 1.2 A per motor phase |
| maximum motor voltage: | +/-16V |
| resolution: | max. 40,000 steps/rev using motors with 200steps/rev |
| baud rate (selectable): | 300 to 19.200 / AUTOBAUD MCL2<br>300 to 9,600 / AUTOBAUD - MCL3 |
| Environmental conditions: | |
| Temperature range | Operating: 5 to 40°C<br>Nonoperating: 0 to 43°C |
| Humidity | Operating: 8 to 80%<br>Nonoperating: 0 to 80% |
| Size X x Y x Z: | 250 mm x 230 mm x 95 mm |
| Weight: | 3.5 kg |

# 12 Appendix

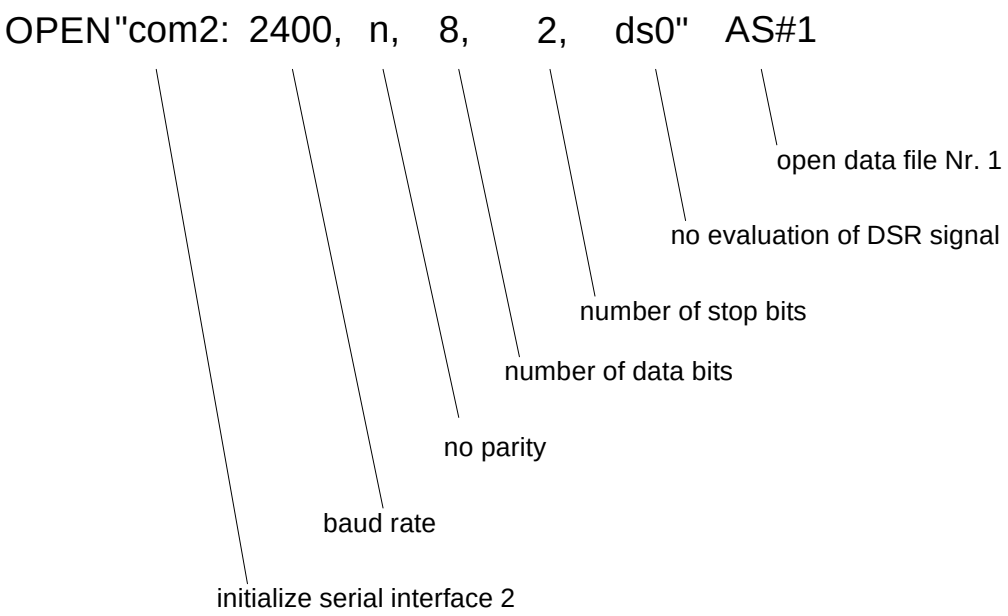
## 12.1 Sample program in GWBASIC for PC

```
100 'Steuerprogramm für MCL2 mit IBM Personal Computer
110 '
120 'Fa. LANG Graviermaschinen
121 'Abteilung Elektronik
122 'Bereich Positioniersteuerungen
129 'Dillstrasse 4, 35625 Hüttenberg
130 'R. Schmidt1994
140 'F.Moos     1994
150 'Tel.: 06403 / 7009 - 30
160 ON ERROR GOTO 190
170 '******** Initialisierung ******************************************
180 CLS : PRINT "*** PRUEFE SERIELLE VERBINDUNG / TISCH CALIBRIEREN ***"
190 CLOSE : OPEN "com2:2400,n,8,2,ds0" AS #1       'open chanel
200 ADR = 7 : MSG$ = "c" : GOSUB 370               'comand calibration
210 ADR = 16 : GOSUB 420                           'table calibration
220 '
230 '******** Ausgangsmenue ********************************************
240 PRINT : PRINT "Lesen = L : Ausgabe = A : Joystick = J/S"
250 PRINT "Position = P : Gerade = G"
260 IN$ = INPUT$(1)
270 IF IN$ = "a" OR IN$ = "A" THEN GOSUB 350 : GOTO 240
280 IF IN$ = "l" OR IN$ = "L" THEN GOSUB 400 : GOTO 240
290 IF IN$ = "j" OR IN$ = "J" THEN K$="j" : GOSUB 460 : GOTO 240
300 IF IN$ = "s" OR IN$ = "S" THEN K$="s" : GOSUB 460 : GOTO 240
310 IF IN$ = "p" OR IN$ = "P" THEN GOSUB 610 : GOTO 240
320 IF IN$ = "g" OR IN$ = "G" THEN GOSUB 730 : GOTO 240
330 GOTO 260
340 '
350 '******** Register der MCC13 - JS beschreiben **********************
360 PRINT "AUSGABE: <ADR>,<DATEN><RET>"; : INPUT ADR,MSG$
370 PRINT#1,"U";CHR$(ADR);MSG$
380 RETURN
390 '
400 '******** Register der MCC13 - JS lesen *****************************
410 PRINT "REGISTER LESEN: <ADR><RET>"; : INPUT ADR
420 PRINT#1,"U";CHR$(ADR+64)
430 INPUT#1,MSG$ : PRINT MSG$
440 RETURN
450 '
460 '******** Joy - Stick ein-/ausschalten ******************************
470 PRINT "JOY - STICK ist aktiv : mit <J> ausschalten"
480 PRINT#1,"U";CHR$(7);K$
490 PRINT#1,"U";CHR$(64+16)
500 IN$ = INPUT$(1) : IF IN$ = "j" OR IN$ = "J" THEN 510 ELSE 500
510 PRINT#1,"j"
520 INPUT#1,MSG$
530 PRINT#1,"U";CHR$(7);"r"
540 PRINT "JOY - STICK ist inaktiv"
550 PRINT " X = "; : ADR = 3 : GOSUB 420 'Position anzeigen
560 PRINT " Y = "; : ADR = 4 : GOSUB 420 '"
570 PRINT " Z = "; : ADR = 5 : GOSUB 420 '"
```

```
580 RETURN
590 '
600 '******** Absolutposition anfahren **********************************
610 PRINT : PRINT "ABSOLUTPOSITION ANFAHREN"
620 PRINT "Position X,Y "; : INPUT X$,Y$ : K$="r"
630 PRINT#1,"U";CHR$(0);X$;CHR$(13);"U";CHR$(1);Y$
640 PRINT#1,"U";CHR$(7);K$;CHR$(13);"U";CHR$(80)
650 PRINT "Abbruch mitA"
660 IN$=INKEY$
670 IF IN$="a" OR IN$="A" THEN PRINT#1,"a"
680 IF LOC(1) = 0 THEN 660
690 INPUT#1,MSG$ : PRINT MSG$
700 RETURN
710 '
720 '******** Gerade fahren ******************************************
730 PRINT : PRINT "GERADE FAHREN"
740 PRINT "Vektor X,Y "; : INPUT X$,Y$ : K$="v" : GOTO 630
```

## 12.2 Description of OPEN-instruction in GWBASIC for PC

Example:



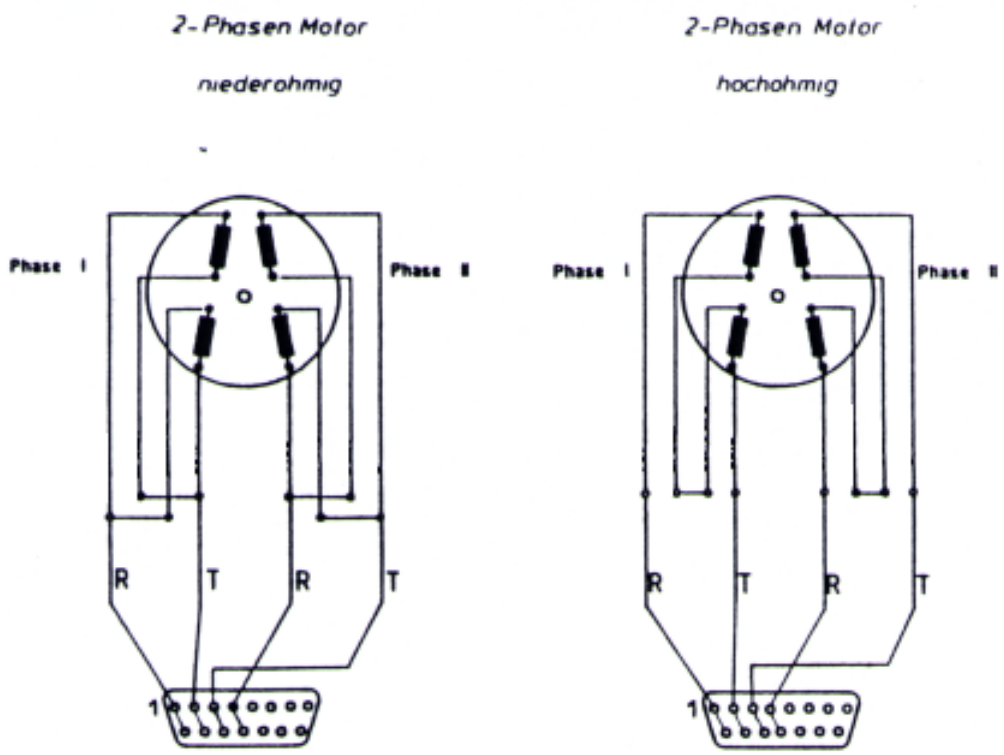

## 12.3 Motor connection

The MCL is designed for the operation with light coordinate tables operated by 2 phases stepping motors up to 1.2 A. Due to the high resolution driving capability and adjustable acceleration via ramps smooth operation is ensured in all operating modes.
In order to guarantee safe operation, please note the following recommendations:

- Select motors with low-resistance and low-inductance
- Make sure, motors with 8 terminals (ZSS 42.200.1,2) are connected according to fig. below, in order to achieve a low-resistance
- motor current is approved up to 1.2 A per phase. However, to avoid unnecessary heating, the motor current should be adjusted as low as possible.
- rated current leads to saturation of magnetic material. In this case, step angular errors will increase.
- a motor current of 0.8A is sufficient to operate microscope table EK8, if a low-resistance is achieved

**Connection of motor**

## 12.4 Testing and adjusting




2 phase motor, low-resistance 2 phase motor, high-resistance

***15 pin connector***

The MCL can be tested and adjusted according to following instructions:
**ATTENTION: qualified technical personnel only!**
**pull mains before opening the unit!**
Test the following voltages with digital multimeter:
MP15: controlled logic voltage (+12V +/-5%)
MP14: controlled logic voltage (-12V +/-5%)
MP10: controlled logic voltage (+4.8V to 5.25V
MP9: reference voltage (+5V +/-5V%)
MP16: motor voltage +UBM (circ. +15V <+18V)
MP17 : motor voltage -UBM (circ. -15V >-18V)

### Adjustment of motor current with oscilloscope (X/Y display):

- X motor current:
oscilloscope connections: GND with MP8, channel1 with MP1, channel2 with MP2. Adjust the motor current with P7.
- Y motor current:
oscilloscope connections: GND with MP8, channel1 with MP4, channel2 with MP5. Adjust the motor current with P8.
- Z motor current:
oscilloscope connections: GND with MP8, channel1 with MP6, channel2 with MP7. Adjust the motor current with P9.

**ATTENTION**: The measured voltage $U_{peak}$ (radius) is equal to the motor current.

Joy-Stick – adjustment with digital multimeter: GND to MP3

· X-axis:
Adjust by P1 (symmetry) and P4 (amplification) in that way, that the adjustable voltage range at end-scale deflection of Joystick on MP11 reaches from <0,15 V ... >4,85 V and the symmetry is in central position of Joystick exactly 2,5V.

· Y-axis:
Adjust by P2 (symmetry) and P5 (amplification) in that way, that the adjustable voltage range at end-scale deflection of Joystick on MP12 reaches from <0,15 V ... >4,85 V and the symmetry is in central position of Joystick exactly 2,5V.

· Z-axis:
Adjust by P3 (symmetry) and P6 (amplification) in that way, that the adjustable voltage range at end-scale deflection of Joystick on MP13 reaches from <0,15 V ... >4,85 V and the symmetry is in central position of Joystick exactly 2,5V.

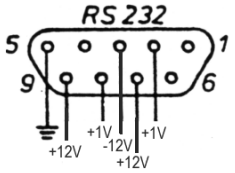
### Limit switch polarity:

| Jumper: | Function: |
|---|---|
| J1.1 | limit switch X -endposition active low |
| J1.2 | limit switch X -endposition active high |
| J2.1 | limit switch X -zero position active low |
| J2.2 | limit switch X -zero position active high |
| J3.1 | limit switch Y -endposition active low |
| J3.2 | limit switch Y -endposition active high |
| J4.1 | limit switch Y -zero position active low |
| J4.2 | limit switch Y -zero position active high |
| J5.1 | limit switch Z -endposition active low |
| J5.2 | limit switch Z - endposition active high |
| J6.1 | limit switch Z - zero position active low |
| J6.2 | limit switch Z - zero position active high |

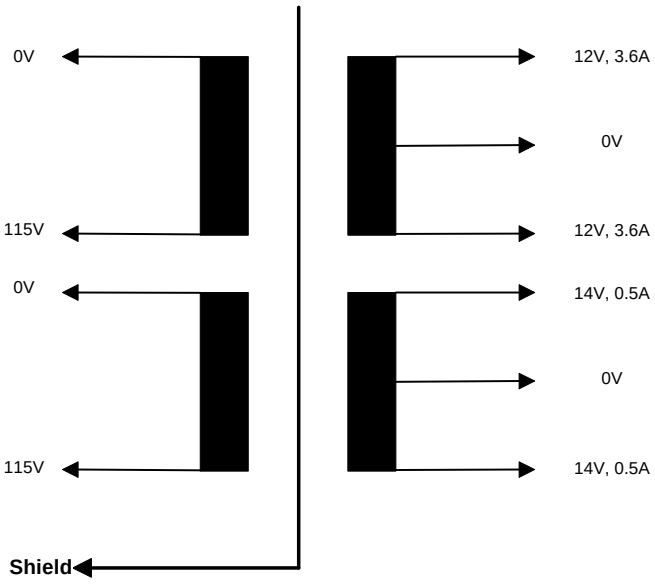
### Adjustment of baud rate:

| Baud rate | S 1.1 | S 1.2 | S 2.1 | S 2.2 |
|---|---|---|---|---|
| 9.600 | OFF | ON | OFF | OFF |
| 4.800 | ON | OFF | OFF | OFF |
| 2.400 | ON | ON | OFF | OFF |
| 1.200 | OFF | OFF | OFF | ON |
| 600 | OFF | ON | OFF | ON |
| 300 | ON | OFF | OFF | ON |
| AUTOBAUD | ON | ON | OFF | ON |

### Error recovery instruction

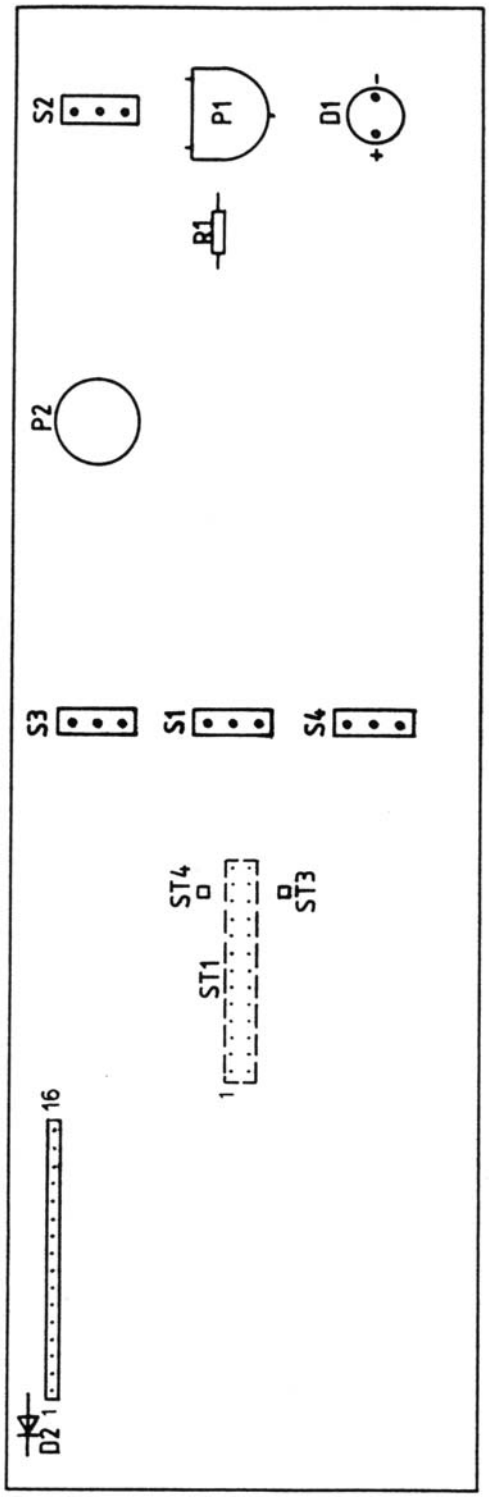
| error description | fault localization / fault clearing |
|---|---|
| 1 blackout [failure] | test power supply and mains fuse in fusesocket on the back side of unit |
| 2 motor becomes too hot | check wiring of the motor (see:motor connection) |
| 3 motor doesn't rotate with high speed | motor is high-resistive (see: motor connection) |
| 4 the individual motor hums, and does not rotate in spite of low adjusted speed | change the motor cables with each other.<br>If the fault remains with the same axis, check cable and motor<br>If the fault moves to another axis, check MCL |
| 5 single axis does not turn, no humming noise can be heard | a) check the limit switch<br>b) test according to 4 |
| 6 no data transmission over RS 232 | a) test the voltages on the mcl by pulled interface cable  <br>b) check computer and interface cable |
| 7 echos of the MCL are displaced, the correct message appears after several read cycles | a message of MCL hasn't been read out from the receive buffer. Test the application program, after a START or READ command the reply of MCL was ignored |

## 12.6 Transformer wiring

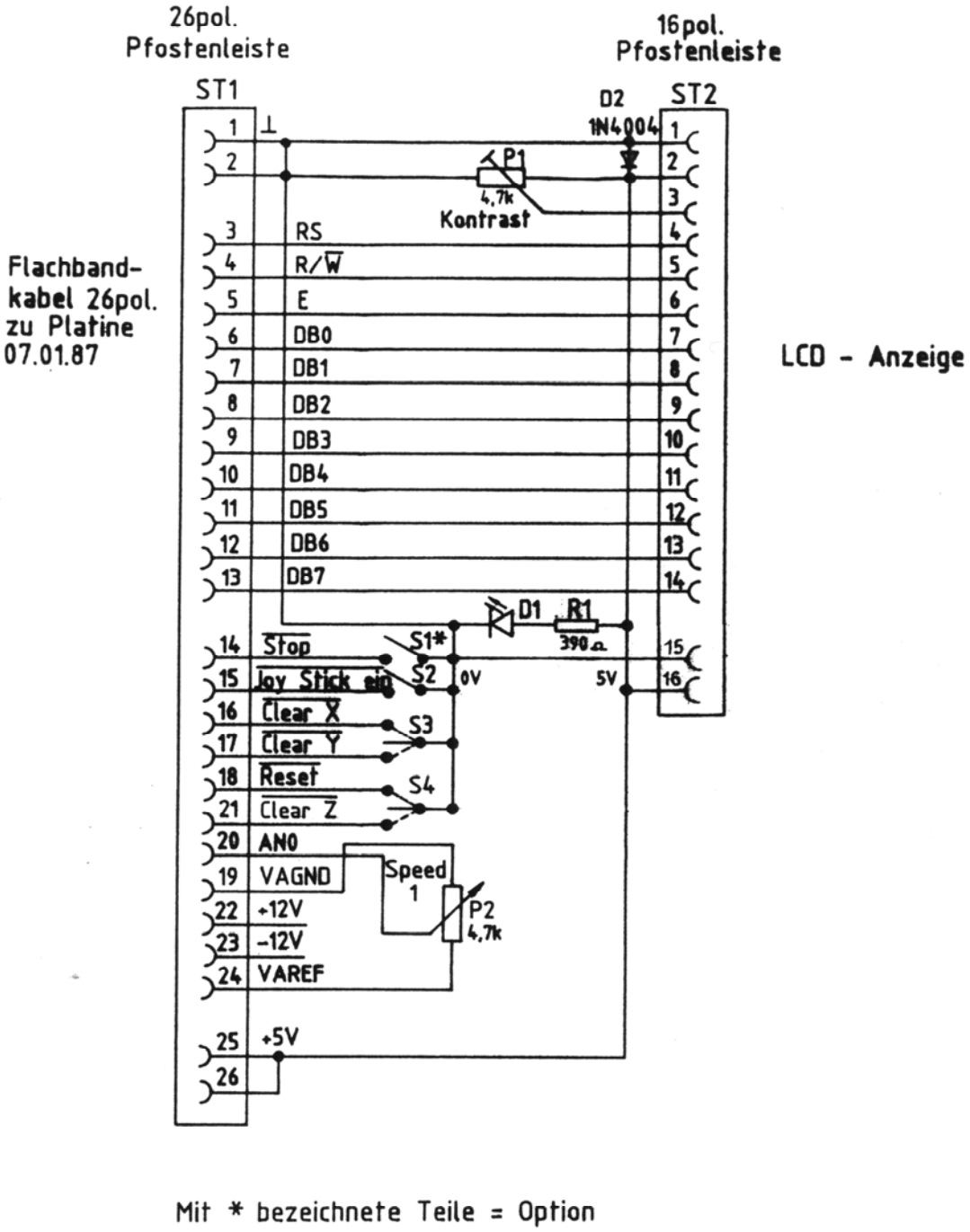


**SELECTION OF THE POWER SUPPLY:**
Selection of the power supply voltage at the incoming socket

**ATTENTION !! ATTENTION !! ATTENTION !! ATTENTION !! ATTENTION !!**
While the power selections is set to 110V, but the device is being connected to 200V - 240V, the electronics could be damaged. The fuse blows in any case of wrong connection.

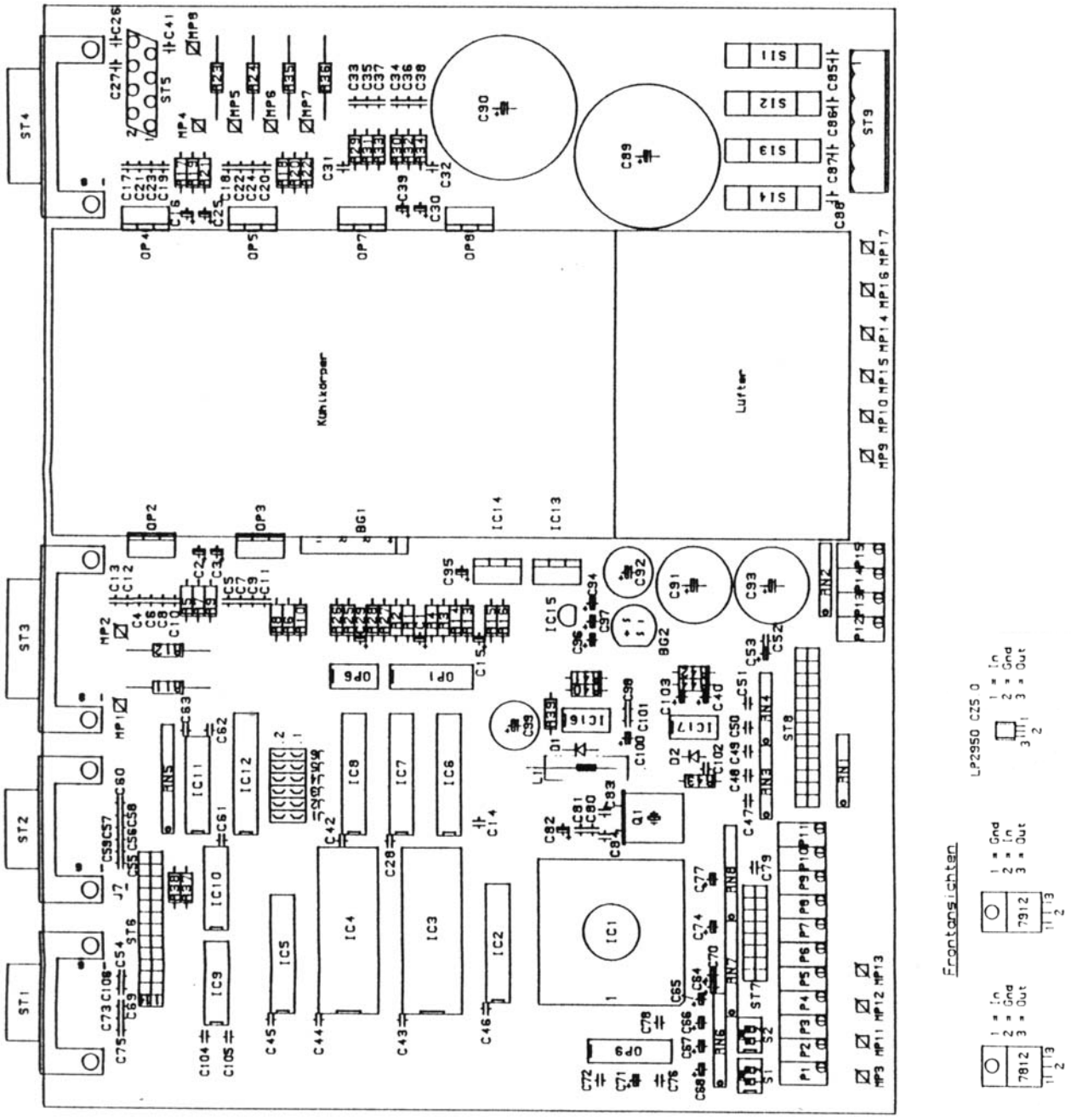



*Frontplattenverbinder MCL* / **Front panel connector MCL**
07.02.97




Mit * bezeichnete Teile = Option

***Frontplattenverbinder MCL* / Front panel connector MCL**
07.02.97




**MCL- 2 oder 3 Achsensteuerung 07 05 94**
*MCL- 2 or 3 axis control 07 05 94*