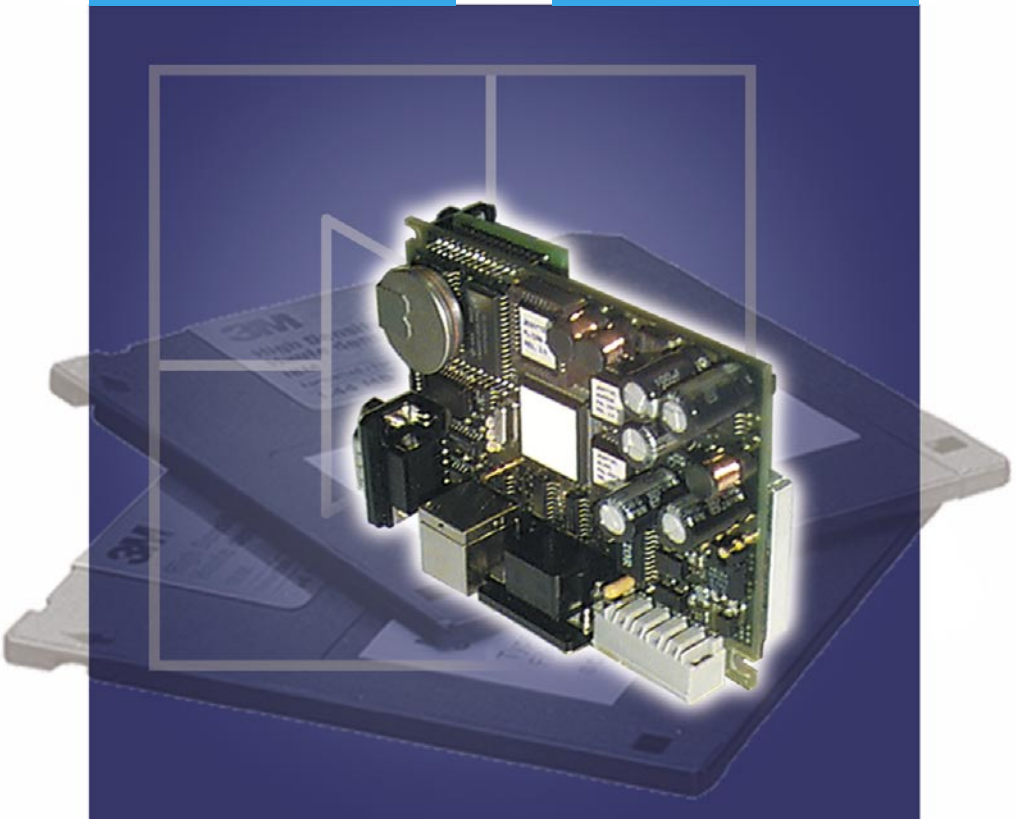
FAULHABER

Motion Controller

4-Quadrant PWM for DC-Micromotors and Brushless DC-Servomotors

Series MCDC 3603 **Series MCBL 3603**



FAULHABER GROUP
We create motion

Operating Instructions

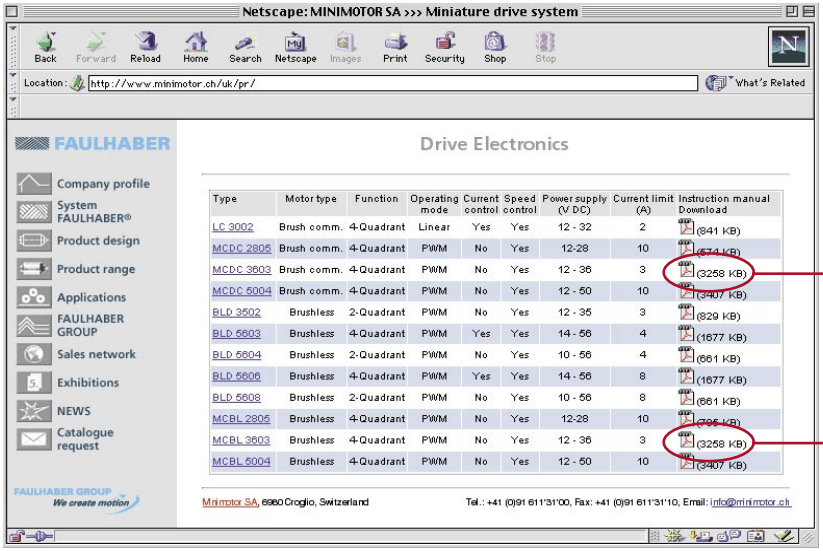
www.faulhaber.com

Miniature Drive Systems

Micro Drives
DC-Micromotors
Precision Gearheads
Servo Components
Drive Electronics

Surf to the following Internet address and you will find the latest edition of the instruction manual on-line:

www.minimotor.ch/uk/pr/



For direct Download:

http://www.minimotor.ch/minicatalog/pdf/DriveCircuits/Manuals/IM_e_MCDC_MCBL_3603.pdf

Index

Description

1. Description

The MCBL 3603 and the MCDC 3603 are very compact motion controllers ideal for our brushless DC-Servomotors and brushed DC-Micromotors.

Each model comprises a PWM servo amplifier.

Technology

Both motion controllers are based on a fast, powerful 16 bit microcomputer system.

This guarantees high dynamics, precise positioning and quiet running, regardless of the motor type used.

The well thought-out design and consistent application of SMD technology ensures a very compact device. The specially developed user software offers high flexibility and simple handling.

Application field

Developed with the use of state-of-the-art technology, the motion controllers are suitable for a wide range of applications: insertion and handling machines, machine tools, robots, X/Y tables, drive and automation systems in medical technology, chemical and food industry, etc.

Programming

One of the most important objectives in the development of these units was to keep its operation as simple as possible. This has been attained with the use of just a few, highly efficient functions.

Manual balancing or potentiometers are no longer required. Menu-guided program and parameter-editing functions are already integrated for operation with an ASCII terminal. In place of internal menu management, the clearly structured command set can be simply integrated into a customer-specific interface, e.g. with Visual Basic, Lab View, Pascal, C++, etc.

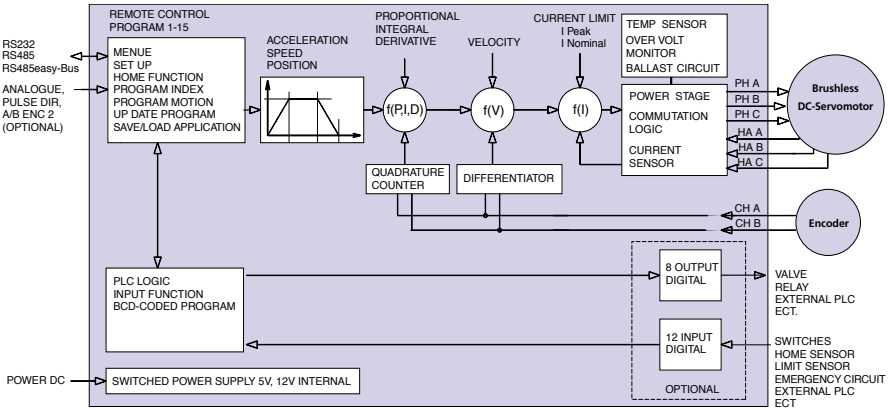
Any PC with Windows operating system can be used as an input terminal. Program updates are made directly via the serial interface without changing the hardware. Communication is made via the serial port RS232 or RS485.

We advise the use of the software WINMOTION® for an easy programming of Motion Controllers provided with Firmware 4.10.

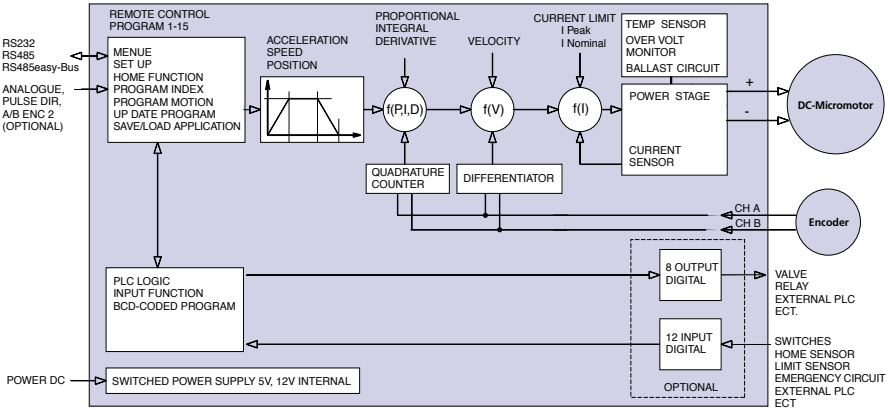
Model overview

2. Model overview

MCBL 3603	Integrated PWM servo amplifier 36V-3A	Brushless DC-Servomotors with encoder



MCDC 3603	Integrated PWM servo amplifier 36V-3A	Brushed DC-Micromotors with encoder



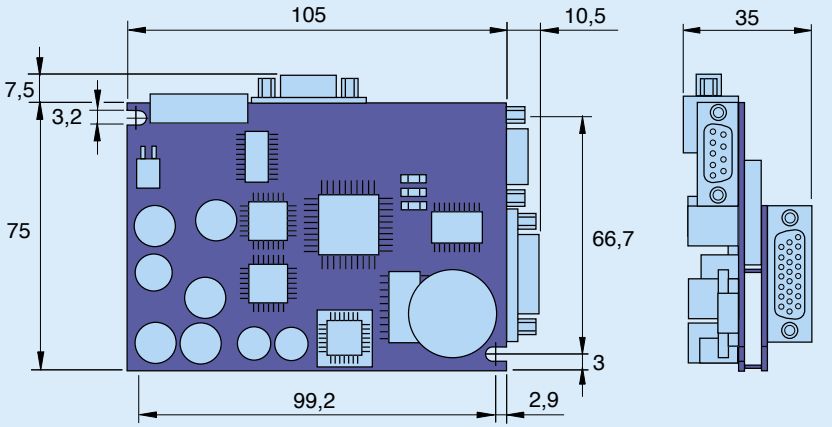
Technical information

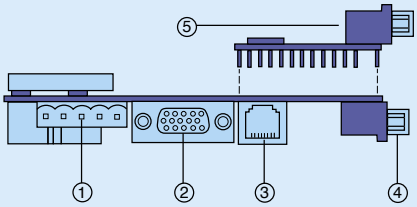
3. Technical data	MCBL 3603	MCDC 3603	
Electrical data			
Supply voltage	12 ÷ 36	12 ÷ 36	V DC
PWM switching frequency	20	20	kHz
Max. continuous output current	3	3	A
Max. peak output current	10	10	A
Max. encoder frequency	200	200	kHz
Software data			
Program memory (16 bit access)	256 x 8	256 x 8	kbyte
Sampling period	500	500	µs
Number of programs	15	15	
Lines per program	50	50	
Number of indexes	50	50	
Communication data			
Interface	RS232 / RS485 / RS485easy-Bus		
Status display	3 LED's		
Optional inputs (5V pull-up standard, 24V pull-down on request)	12 (optional)		
Optional outputs (6 x 50V/500mA open collector, 2 x TTL level)	8 (optional)		
Program and parameter editor	integrated ASCII terminal		
Program up-date	via serial interface		
Application and parameter save / load	via serial interface		
Starting position function	via encoder Z-index / via external sensor		
Temperature rating			
Operating temperature	0 ... + 55		°C
Storage temperature	-20 ... + 80		°C
Weight / Dimensions			
Weight:	130		g
Dimensions: see diagram on page 5			

Dimensions

4. Dimensions for MCBL 3603 and MCDC 3603

Scale reduced





Connection for MCBL 3603

① Power supply motor / motor connection
② Encoder [1] and Hall sensors
③ Input for special function
④ RS232
⑤ Optional digital Input / Output

Connection for MCDC 3603

① Power supply motor / motor connection
② Encoder [1]
③ Input for special function
④ RS232
⑤ Optional digital Input / Output

[1] Line driver encoders for noisy environments or long distances can be used.

Start-up procedure

5. Start-up procedure

Here we list a step-by-step start-up procedure for both the electrics and software.
Also included are several examples in order to allow the user to test the unit and familiarise himself with programming.

We therefore recommend that this sequence is followed for trouble-free installation:

Start-up Procedure for MCBL 3603

- Connect the motor phases to **MOTOR**
- Connect the encoder and the motor Hall sensor leads to **ENCODER HALL**
- Connect the RS232 (or RS485) to the computer port **COM**
- Connect the power supply to **PWR**
- Power the motion controller
- Software start-up

Start-up Procedure for MCDC 3603

- Connect the motor terminals to **MOTOR**
- Connect the encoder to **ENCODER**
- Connect the RS232 (or RS485) to the computer port **COM**
- Connect the power supply to **PWR**
- Power the motion controller
- Software start-up

The computer link is necessary to program the motion controller.
After programming has been completed, the computer link can be disconnected since the programs can be started using the motion controller input functions.

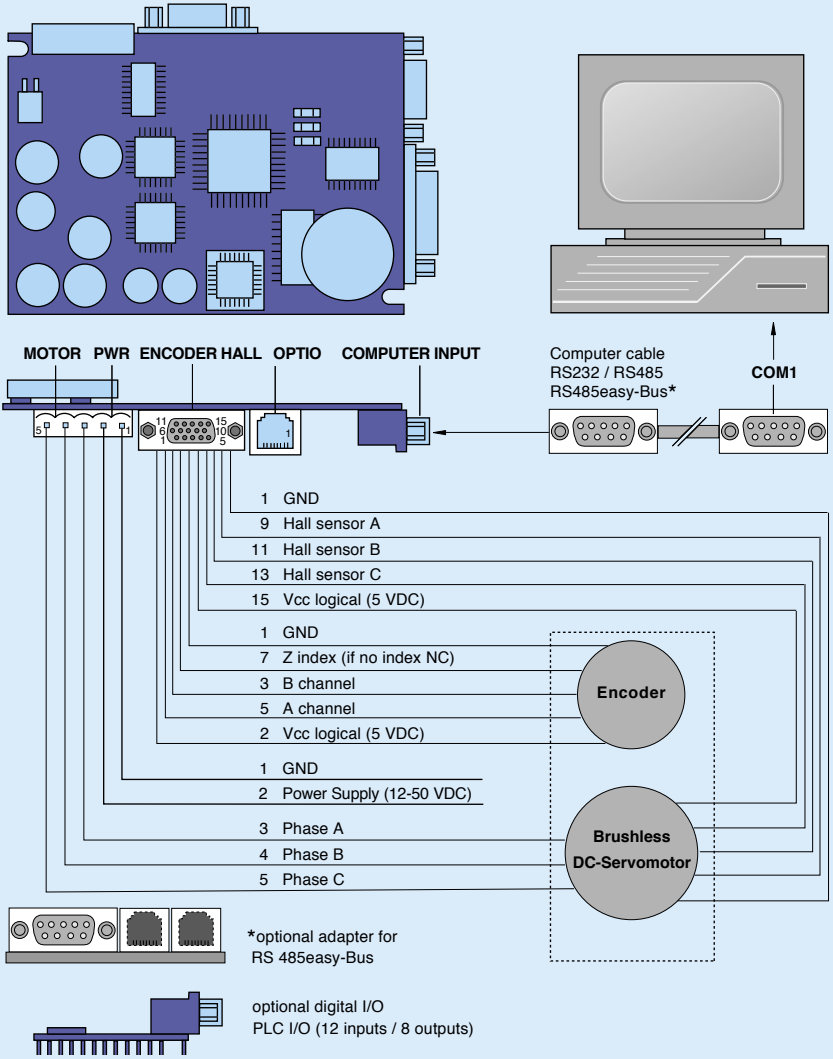
For advanced functions such as:

- Analogue input command
- Stepper motor emulation
- RS 485 serial interface
- Multi-axis operation

please refer to the specifique chapters

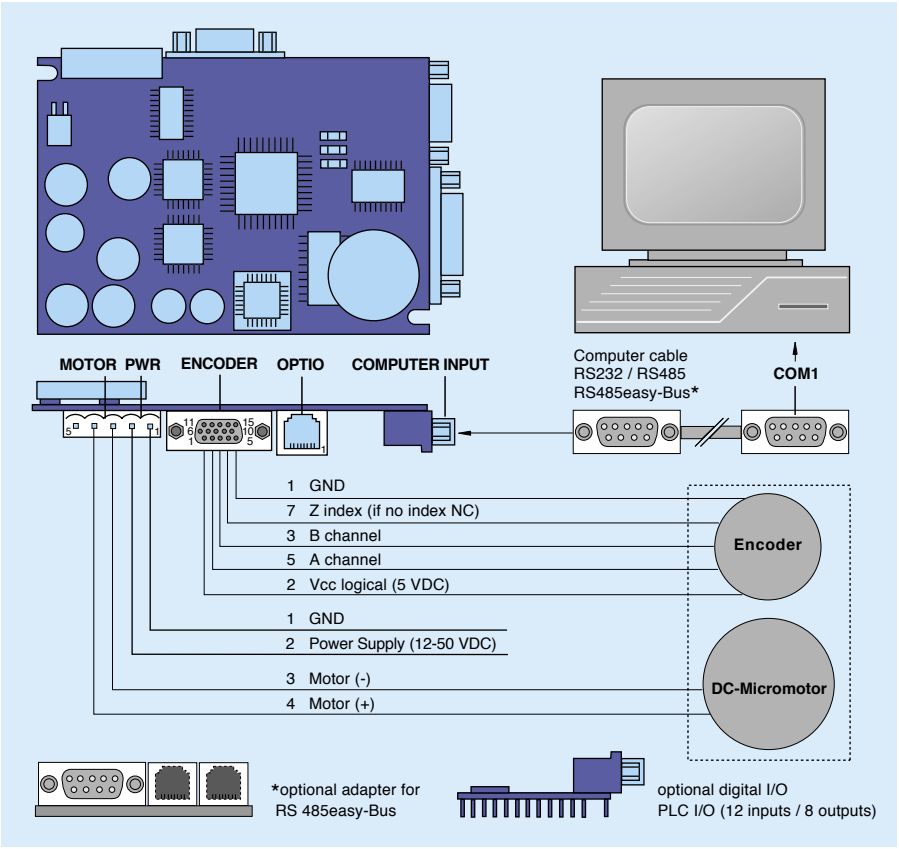
Start-up procedure

6. Connection diagram for MCBL 3603



Start-up procedure

7. Connection diagram for MCDC 3603



PLC I/O description
The PLC I/O port enables direct communication with the position control function without having to use a computer. For example, once a program has been created, it can be executed by simply giving a command to the assigned input. In the initial phases of installation, and to allow the user to better understand the operation of the motion controller, all instructions are given via computer. It is therefore not necessary to connect this port.

RS232 description
This port is the communication link between the motion controller and the external computer via the COM1 connection point. Additional information regarding the connection and set-up is given in chapter 29. The link is make with a standard computer cable which, if necessary, can be optionally supplied by Minimotor.

General software information

8. General software information

Terminal emulator description
The computer is only used as a terminal. The terminal emulator therefore enables communication between the computer and motion controller software. The actual programming is made directly in the motion controller itself.

Motion controller software organisation
The software is constructed on three different levels.

- **Operating system**

  The operating system normally remains invisible to the user and is a background function for:
  – download functions
  – back-up in emergency situations
  For additional information, see chapter 26.

- **Program xx36_yyy.S19**

  Is the basic working program which realises all described functions and programming possibilities. This program is already installed within the unit and automatically goes into operation once the system is started-up. Actualy version 4.10

- **Application user programs**

  Contains the complete set of customer-defined data and parameters (= “application”).

LED status

Display	Description
LED 1	Internal 5V power supply OK
LED 2	Software OK
LED 3	Servo OK, system closed
LED 3 blinking	Error, ask error code, see troubleshooting chapter 27

General programming instructions
General instructions on how to move, insert, delete, etc. within the program:

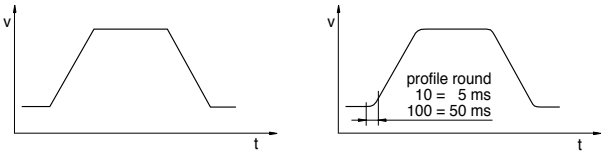
- Close every entry with the command <ENTER>
- Text can be entered using either small or capital letters
- Use the arrows to move up and/or down the menu lines
- To go back to the previous menu always use <ESC>
- Close erroneous entries with <ENTER> and re-enter data

Delete characters	Back Space
Clear line	<CRTL C>
Insert line	<CRTL I>
Page down	<CRTL D>
Page up	<CRTL U>

Set up values

9. Set up values

Description

MODE	0 = Programs and commands operated using the standard inputs (see chapter 19) 2 = Programs and commands operated using the standard inputs. PULSE/DIRECTION input signal for stepper control emulation function active (see chapter 23) 3 = Programs and commands operated using the standard inputs. Analogue input signal for digital speed control function active (see chapter 22) 10 = Programs operated using the 4 binary coded inputs and input 8 as starting trigger (see chapter 20) 12 = Programs operated using the 4 binary coded inputs and input 8 as starting trigger. Stepper control emulation function active (see chapter 23) 13 = Programs operated using the 4 binary coded inputs and input 8 as starting trigger. Analogue input signal for digital speed control function active (see chapter 22)
I NOM	Nominal current
I PEAK	Peak current
PROPORTIONAL	Proportional closed loop parameter (stiffness)
INTEGRAL	Integral closed loop parameter (positioning precision)
DERIVATIVE	Differential closed loop parameter (stability, dynamic)
VELOCITY	Velocity closed loop parameter (oscillation prevention)
INC PER PULSE	Increment (lines) per pulse in MODE 2 or 12 for stepper emulation function (see chapter 23)
DEVIATE POS	Permissible max. position deviation in lines
PROFILE ROUND	To smooths the speed profile out (see below)
	 By setting to profile round, the speed profile is smoothed out, this reducing mechanical stress for better live performance.
INPUT H-ACTIVE	0=Input active low, I=Input active
ANALOG FUNCTION	Analog function active with mode 3 or 13 for digital speed control. 0=CW (+), I=CCW (-), 2=Cw and CCW (+/-)

On line control

10. On line control

Command	Parameter	Description
AC	1 000 - 4 000 000 lines/s^2	Acceleration
AIX	10 - 50 000 (x 1 000) lines/s^2	Overwrite acceleration index at preloaded NIX number by remote control
ANF	0 - 2	Analog function mode 3/13 0 = CW (+) 1 = CCW (-) 2 = CW and CCW (+/-)
CI	0-100	Card identifier for RS485easy-Bus
CO	1 - 8	Clear output
CLO	0 - 1	Clear outputs after HOME function 0 = no, 1 = yes
DIX	± 2 000 000 000 lines	Overwrite distance index at preloaded NIX number by remote control
DRH	1 - 2	Direction of motor rotation, for seeking coarse sensor 1 = CW, 2 = CCW
DRZ	1 - 2	Direction of motor rotation, for seeking Z mark sensor 1 = CW, 2 = CCW
DP	lines	Permitted position deviation in lines
DV	0 - 50	Differential closed loop parameter
EC		Encoder counter on-line diagnosis
ED	1 000 - 5 000 000 lines/s^2	Emergency deceleration with Exit function EE and Limit-switch function LL and LR
GP		Go to position (absolute)
GW		Go way (relative)
GZ		Go to Z-index (encoder)
HO		HOME function according to program
HOF	0 - 100 000 increment	Offset after edge coarse sensor no stop same direction, if HOF is not 0 this value is indicated on HOME menu
ICP	1 - 50	Inc. per pulse, mode 2/12, pulse / direction control
IHA	0 - 1	0 = input low active, 1 = input high active
IN	1 - xx	Nominal current
INH	1 - 8	HOME sensor input number
IP	1 - xx	Peak current
IT	0 - 50	Integral closed loop parameter
IX	1 - 50	Run index # according to program
JP		Jog (run) positive, constant speed
JN		Jog (run) negative, constant speed
JNZ[letter]	1 - 50	Indicate loop reference letter (from A to E). Decrements the loop repeats, whereby if not zero, jump to line xx
NIX	1 - 50	Number index pre-load for changing index parameters by remote control
PG	1 - 15	Run program #
PO	± 2 000 000 000 lines	Position (absolute)
PP	1 - 50	Proportional closed loop parameter
PRF	1 - 100	Rounding of speed profile (should be value), smooth start and smooth stop
PQ		Servo amplifier power OFF
PW		Power ON, reset position counter
PWC		Power ON continue, keep position counter
RI	0 - 100	Required identifier for RS485easy-Bus
*RI	1 - 99	Get back identifier, position, and status complete
RR	1 - 10 000	Repeat way CW/CCW
RW	1 - 10 000	Repeat way (same direction)
SET[letter]	1 - 10 000	Set loop reference letter (five possibilies, from A to E) and number of repeats xxxx

On line control

Command	Parameter	Description
SIX	25 - 1 000 000 lines/s	Overwrite speed index at preloaded NIX number by remote control
SM		Stop motion
SO	1 - 8	Set output
SP	25 - 1 000 000 lines/s	Speed
SR	± 1 - 100 :10	Synchronisation ratio with optional second encoder
TE		Tell error codes 01-99
TGD	± 2 000 000 000	Trigger downward count, absolute, at output x (5 ms) defined in output function
TGU	± 2 000 000 000	Trigger upward count, absolute, at output x (5 ms) defined in output function
TI	? or 1 - 12	Tell status input, 0=Low 1=High
TP		Tell actual position ± 2*10E9
TS		Tell status: 0 = power OFF 1 = power ON 2 = moving 3 = program active 9 = error
VL	1 - 50	Velocity closed loop parameter
WA	± 2 000 000 000 lines	Way (relative)
WT	x 10 ms	Waiting time

Set-up values

11. System parameter set-up

Current limiter set up

The current limits I NOM and I PEAK must be set according the motor used.
The value of I NOM should not exceed the motor's recommended current for continuous operation.
I NOM limit is only active during constant speed operation.
I PEAK limit is only active during acceleration and deceleration.

There is a continuous monitoring of incremental feedback. If the motor is blocked more than 0,5 seconds then the current will be automatic reduced.

Optimising the closed loop parameters

The closed loop system can be optimised by running the motor (including assembled mechanical parts) directly on line and by adjusting the following parameters via the SET UP VALUES menu:

PROPORTIONAL	(1 - 50)
INTEGRAL	(0 - 50)
DERIVATIVE	(0 - 50)
VELOCITY	(0 - 50)

This optimisation is best carried out by running the motor with the RW and/or RR instructions.

When executing these instructions, all parameters (even set-up) can be changed on line, thus enabling the user to see the reaction of the system whilst making changes. One helpful function is the EC (encoder counter) which gives information on the actual motor shaft position.

Improved dynamics

If your application requires more dynamics, this can be obtained by increasing the PROPORTIONAL, DERIVATIVE and VELOCITY LOOP values (e.g. to 10, 20 and 20).
If the motor is noisy or vibrates (indicating system instability), these parameters should be reduced.

Precise positioning

If you need to improve the motor's position holding, an INTEGRAL value should be given (e.g. 5). The INTEGRAL value is only activated when the motor reaches the requested position. In this way the system's dynamic is not influenced by this value.

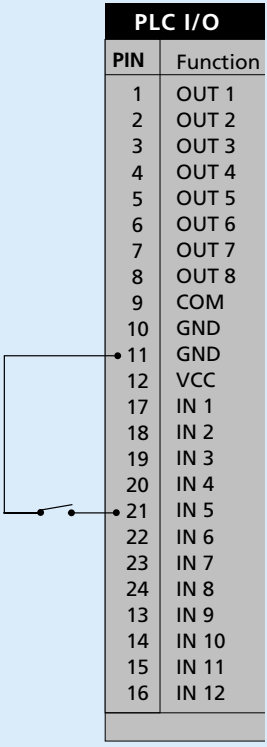
To control the exact position of the motor, the EC (encoder counter) command is used via the ON LINE CONTROL menu.

Call up program

12. Call up program from normal inputs (Optional I/O)

The procedure to execute a program or another instruction via the 8 normal inputs is as follows.

- assign the instruction to the desired input via the INPUT FUNCTION menu
- activate the input via an external circuit (see example below)



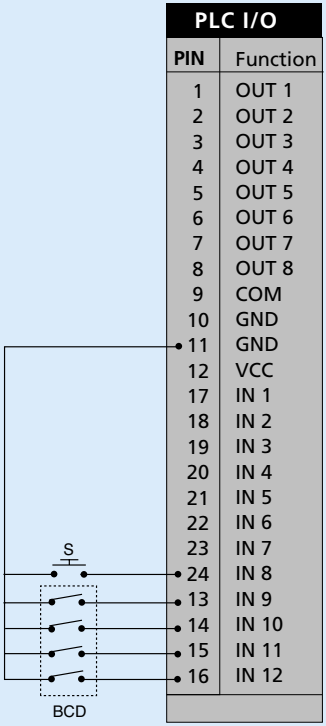
Call up program

13. Call up program from BCD coded digital inputs (Optional I/O)

When the application uses more than 8 digital inputs, the user should call them up via the BCD coded digital inputs. In this case, the MODE in SET-UP VALUES menu should be set to 10 (or 12 or 13).

The input lines 9 - 12 are used as binary coded program numbers. The trigger to start the pre-selected program is input line 8. Program number 0 is not used. Therefore:

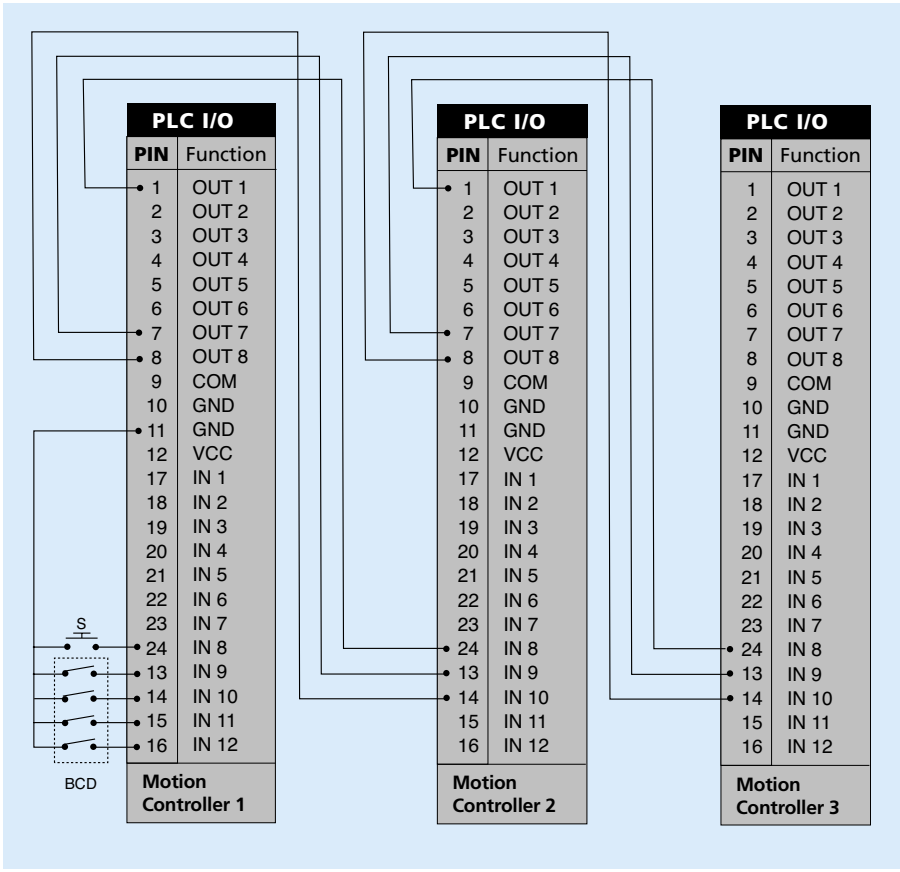
- pre-select program number with binary-switch (numbers 1-15)
- start program with start button S



Call up program

14. Example of sequential multi-axis application (optional I/O)

The program number and program start of the sequential follow-on motion controllers are specified through the output of the lined up motion controller.



Advanced functions

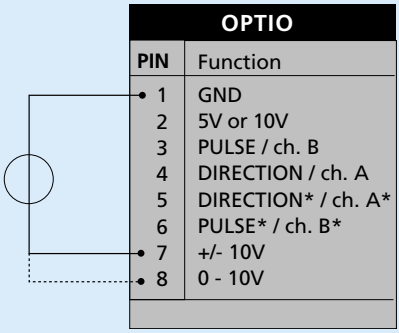
15. Analogue input command

To activate the analogue input command function the MODE parameter in the SET UP VALUES menu must be set to 3 or 13 and the ANALOG FUNCT to:

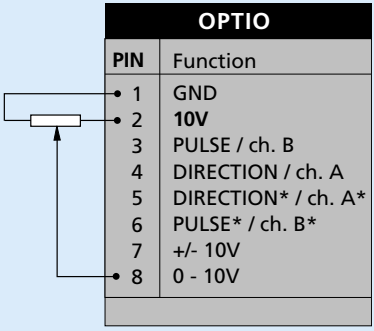
0 for CW operation
1 for CCW operation
2 for CW and CCW operation

The maximum speed is defined with the **SP** command.
For high dynamics we recommend increasing the **AC** value (**AC** > 2 000 000).

External voltage



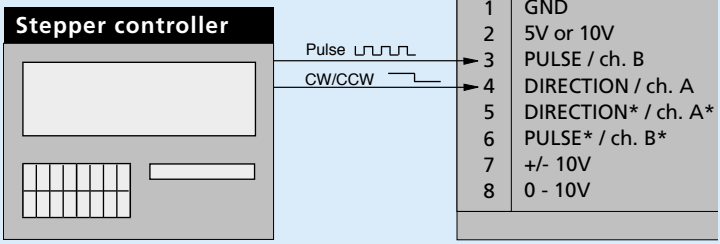
Potentiometer



Advanced functions

16. Stepper motor emulation

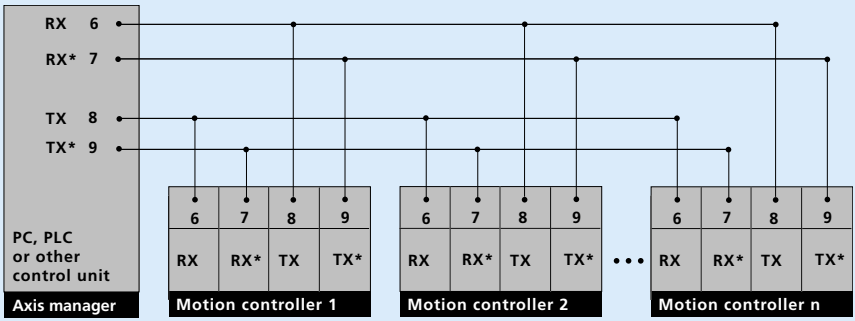
To activate the stepper motor emulation function the MODE parameter in the SET UP VALUES menu must be set to 2 or 12 and the INC PER PULSE according to the application requirements.



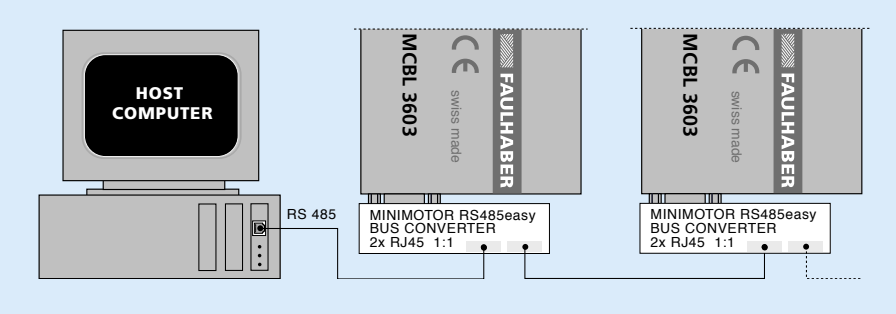
17. RS485easy-Bus

With this feature up to 32 motion controllers can be adressed and controlled by one host computer using a simple RS485 interface. The connection principle is show below.

RS485easy-Bus full duplex



Advanced functions



To simplify the connection Minimotor offers a special RS485 easy-Bus adapter and RJ45 cables.

Cable specification:
Modular RJ45 round shielded cable configuration 1:1. Twisted pairs 1&2, 3&6, 4&5, 7&8

Start-up Procedure
Attention: function only available with software version ≥ 3.00

1) **Assign an adress** (number) to each motion controller (axis).
This operation is made by connecting the single motion controller to the RS232 interface and using the **CI** (card identifier) command.

Example: **CI 5 <CR>** to assign the number **5** to a motion controller.
The number can be checked using the command **CI ? <CR>**

Attention: Each motion controller in the system must have a different number. Once the number is assigned it is memorized even if the power supply is switched off. The **CI** value goes from **0-99.** The number **0** is used as a default value for single axis application.
For multi-axis operation a number from **1-99** should be used, Number **1** must always be used (see below).

2) **Realise an RS485 connections** and set the baud rate in the terminal emulator software to **19 200.** To use the RS485 it is necessary to have a RS232/RS485 converter since PCs usually only offer a RS232 interface.

3) **To operate a motion controller** it is first necessary to adress it using the RI (request identifier) command.

Example: **RI 5 <CR>.** To make the prompt appear.
If it does not appear check the RS485 connection and baud rate.

Using the **RI 0** command the host computer can control all the axis at the same time. In this case the echo from Motion Controller number **1** (in multi-axis the number **1** must always be used) will appear on the computer screen.

Trouble-shooting

18. Trouble-shooting

Error messages are shown with LED 3 blinking. A detailed information on the error type can be obtained using the TE (Tell Error) command.
There are two types of error code: one for input errors (WH wait high or WL wait low) and the other for controller errors (DP deviation position or over-heating).

Error code	Description	Remarks
01 to 12	Waiting for input (low or high)	- Continues if status has been reached or restart with HO, SM or PQ, PW.
50	Deviation position too great	- Difference between the internal calculated position and actual motor position greater than the number of increments defined in DP (deviation position).
60	Power stage over-heating	- > 80° C detected by the temperature sensor.
61	Power supply over voltage	- Power supply voltage or retarding energy on ballast circuit to high.
62	Ballast circuit active too long	- If the ballast circuit is active for more than 5 seconds the power stage is switched off.

19. Operating system error

If the message “user program corrupted” appears on the screen, the operating system has to be turned off and re-started manually. This is done as follows:

- **set the S1 switch from OFF to ON (the SMD multiswitch is located be between the battery a the 9 poles D-sub RS232 connector)**
- **re-load the xx36_yyy.S19 program**
- **turn the S1 switch from ON to OFF position**

The system can now be switched on normally.

Notice of use

20. General usage instruction

Power supply and fuse

Any unstabilised DC power supply voltage within the motion controllers range:

- **MCBL 3603** $12\ V \leq Vm \leq 36V$
- **MCDC 3603** $12\ V \leq Vm \leq 36V$

may be used, although it is advisable to keep this voltage as low as possible in order to minimize the EMI noise. Thus the optimum power supply voltage is given by the following equation:

$$Vm\ [V] = 5V + R\ [\Omega] \cdot I_{max}\ [A] + k_E\ [V/rpm] \cdot n_{max}\ [rpm]$$

Where:
- R = motor terminal resistance
- k_E = motor back-EMF constant
- I_{max} = max. requested motor current for acceleration (= I_{PEAK})
- n_{max} = max. motor speed reach in the application Both motion controllers are provided with an internal fuse.

Braking energy

When decelerating the motor, brake energy is developed. This energy increases the motion controller voltage supply. Therefore the motion controllers supplied with an internal ballast circuit which converts this energy into heat.

Wiring

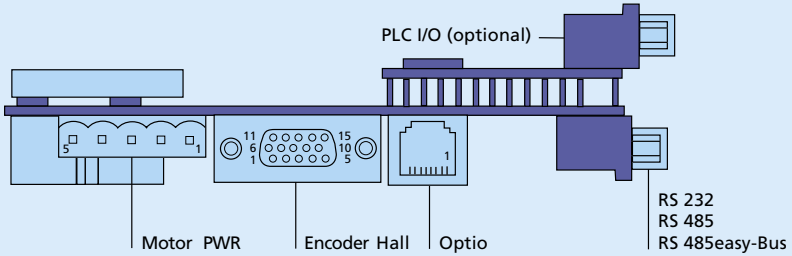
A well known disadvantage of PWM (pulse width modulation), is that it generates a lot of interference. In order to reduce the effect of the interference there are some basic rules to follow:

- **Use wires as short as possible**
- **Avoid running signal wires (logical and analog signal) in close proximity to power lead wires (power supply and motor power leads)**
- **Use shielded wires**

Hardware

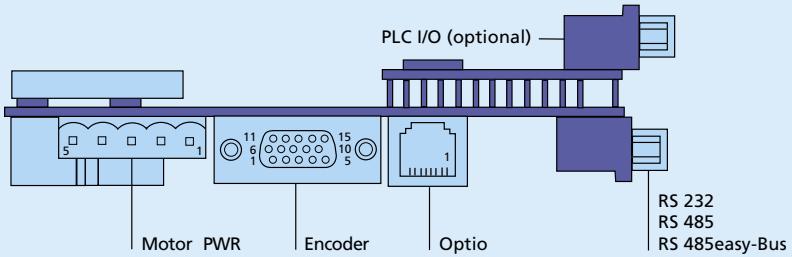
21. Hardware

Connector layout for MCBL 3603



Connect	Nr. Terminal	Function	Type
PWR Motor	2 3	Power supply Motor phases	WAGO Multiconnector 5,0mm
Encoder	15	Encoder input, encoder 1phases, Hall effect sensors	D-SUB High-Density
Optio	8	Pulse/dir, analogue, encoder 2	
PLC I/O	26	12 inputs / 8 outputs	Modular RJ45
RS232/ RS485	9	Serial interface RS232/RS485/ RS485easy-Bus	D-SUB High-Density D-SUB normal

Connector layout for MCDC 3603



Connect	Nr. Terminal	Function	Type
PWR Motor	2 2	Motor terminals Power supply	WAGO Multiconnector 5,0 mm
Encoder	15	Encoder-Input, encoder 1	D-SUB High-Density
Optio	8	Pulse/dir, analogue, encoder 2	Modular RJ45
PLC I/O	26	12 inputs / 8 outputs	D-SUB High-Density
RS232/ RS485	9	Serial interface RS232/RS485/ RS485easy-Bus	D-SUB normal

PIN configuration

22. PIN configuration

Serial interface RS 232 or RS 485, 9 POLE D-SUB

Pin 1	NC	Not connected
Pin 2	RS232	Receiver Rx
Pin 3	RS232	Transmitter Tx
Pin 4	NC	Not connected
Pin 5	RS232	GND
Pin 6	RS485	Receiver R
Pin 7	RS485	Receiver $\overline{R}$
Pin 8	RS485	Transmitter T
Pin 9	RS485	Transmitter $\overline{T}$

RS232 set up

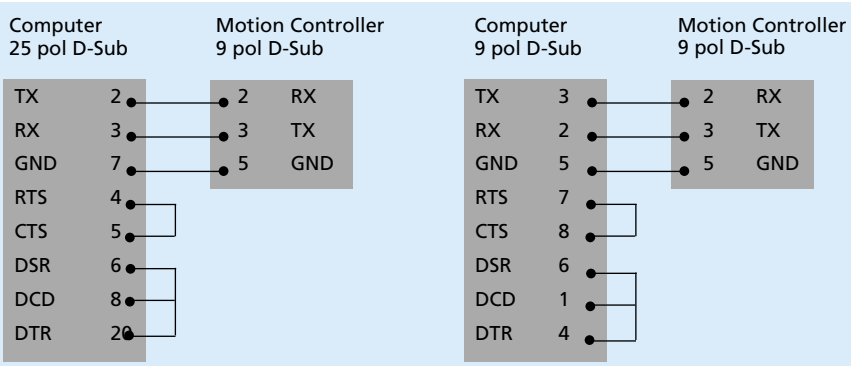
Set the baud rate RS232 via 6-bit CONFIG switch S1 (6 bit SMD multiswitch)

data 8bit
stop bit 1
parity no

bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	Function
x	x	x	OFF	OFF	x	RS232 9 600 baud (default)
x	x	x	OFF	ON	x	RS232 2 400 baud
x	x	x	ON	OFF	x	RS232 4 800 baud
x	x	x	ON	ON	x	RS232 19 200 baud

By turning the system off and back on the new baud rate will be activated.

RS232 electrical connection



PIN configuration

RS485 set up
Setting of the baud rate RS485 over 6-bit CONFIG switch S1: data 8bit, stop bit 1, parity no

bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	Function
x	OFF	OFF	x	x	x	RS485 19 200 baud (default)
x	OFF	ON	x	x	x	RS485 9 600 baud
x	ON	OFF	x	x	x	RS485 38 400 baud
x	ON	ON	x	x	x	RS485 free

By turning system off and back on the new baud rate will be activated.

Bus RS485easy, MODULAR RJ45

Pin 1	NC	
Pin 2	NC	
Pin 3	NC	
Pin 4	RS485	Receiver $\overline{R}$
Pin 5	RS485	Receiver R
Pin 6	NC	
Pin 7	RS485	Transmitter $\overline{T}$
Pin 8	RS485	Transmitter T

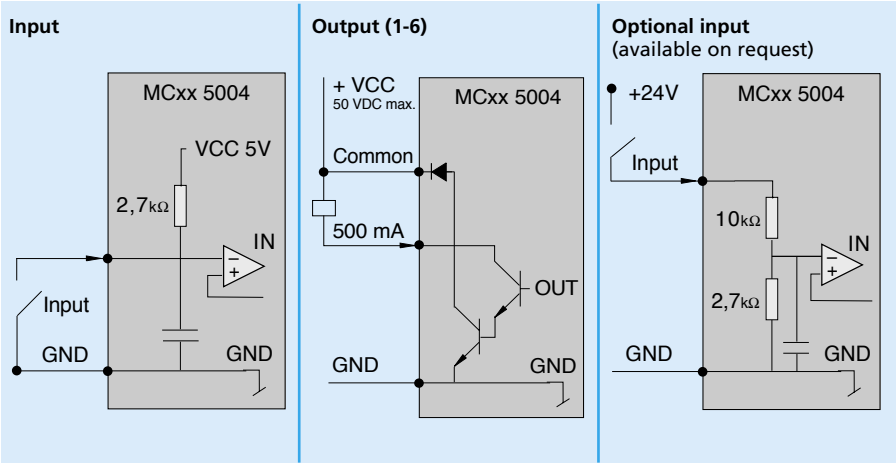
PLC 12 Input / 8 Output available to the user, 26 pole HD-DSUB

Pin 1	Output 1	Active low, open collector 50 V / 500 mA on GND, free-wheeling diode
Pin 2	Output 2	
Pin 3	Output 3	
Pin 4	Output 4	
Pin 5	Output 5	
Pin 6	Output 6	
Pin 7	Output 7	Active low 0 V / 50 mA, high 5 V / 50 mA
Pin 8	Output 8	Active low 0 V / 50 mA, high 5 V / 50 mA
Pin 9	COMMON	Joint cathodes of output free wheeling diodes 1 - 6
Pin 10	GND	2A
Pin 11	GND	2A
Pin 12	VCC	5 V /250 mA
Pin 17	Input 1	Pull up 2,7 k on VCC 5 V standard or optional 24 V (for PNP sensors)
Pin 18	Input 2	
Pin 19	Input 3	
Pin 20	Input 4	
Pin 21	Input 5	
Pin 22	Input 6	
Pin 23	Input 7	
Pin 24	Input 8 [1]	
Pin 25	GND	2A
Pin 26	5V	250 mA
Pin 13	Input 9 [2]	Pull up 2,7 k on VCC 5 V standard or optional 24 V Bit for BCD program decoder.
Pin 14	Input 10 [2]	
Pin 15	Input 11 [2]	
Pin 16	Input 12 [2]	

[1] Program start trigger with BCD coded input (MODE = 10)
[2] BCD coded input for program,1-15, selection (MODE = 10)

PIN configuration

Input and output internal electrical circuit



Encoder Hall, 15 Pole HD DSUB

Pin 1	GND	GND for both, encoder and Hall
Pin 2	5V Encoder	150 mA
Pin 3	Encoder B	Pull up 2,4k to 5V, differential input 26LS32
Pin 4	Encoder $\overline{B}$	middle level:pull up 2,4k to 5V, pull down 2k, differential input 26LS32
Pin 5	Encoder A	Pull up 2,4k to 5V, differential input 26LS32
Pin 6	Encoder $\overline{A}$	middle level:pull up 2,4k to 5V, pull down 2k, differential input 26LS32
Pin 7	Encoder Z	Pull up 2,4k to 5V, differential input 26LS32
Pin 8	Encoder $\overline{Z}$	middle level:pull up 2,4k to 5V, pull down 2k, differential input 26LS32
Pin 9	Hall A	Pull up 2,4k to 5V, differential input 26LS32
Pin 10	Hall $\overline{A}$	middle level:pull up 2,4k to 5V, pull down 2k, differential input 26LS32
Pin 11	Hall B	Pull up 2,4k to 5V, differential input 26LS32
Pin 12	Hall $\overline{B}$	middle level:pull up 2,4k to 5V, pull down 2k, differential input 26LS32
Pin 13	Hall C	Pull up 2,4k to 5V, differential input 26LS32
Pin 14	Hall $\overline{C}$	middle level:pull up 2,4k to 5V, pull down 2k, differential input 26LS32
Pin 15	5V Hall	150 mA

Power supply

Pin 1	GND		
Pin 2	POWER	MCDC 12 - 36V (over voltage limited with protection diode)	
		MCBL 12 - 36V (over voltage limited with protection diode)	
		DC motor	**BL motor**
Pin 3		Motor (-)	Phase A
Pin 4		Motor (+)	Phase B
Pin 5		NC	Phase C

PIN configuration

Optional function, 8 pole modular RJ45

Pin 1	GND	GND internal
Pin 2	10V (5V)	10V default voltage (5V with option second Encoder on request)
Pin 3	Pulse	pull up 2,4k to 5V, differential input 26LS32
Pin 4	Direction	pull up 2,4k to 5V, differential input 26LS32
Pin 5	$\overline{\text{Direction}}$	middle level: Pull up 2,4k to 5V, pull down 2k, differential input 26LS32
Pin 6	$\overline{\text{Pulse}}$	middle level: Pull up 2,4k to 5V, pull down 2k, differential input 26LS32
Pin 7	+/- 10V	analogue input reference, range +/- 10V
Pin 8	0- 10V	analogue input reference, range 0-10V

FAULHABER

The FAULHABER Group:

DR. FRITZ FAULHABER GMBH & CO. KG
Daimlerstraße 23
71101 Schönaich · Germany
Tel.: +49 (0)7031/638-0
Fax: +49 (0)7031/638-100
Email: info@faulhaber.de
www.faulhaber.de

MINIMOTOR SA
6980 Croglio · Switzerland
Tel.: +41 (0)91 611 31 00
Fax: +41 (0)91 611 31 10
Email: info@minimotor.ch
www.minimotor.ch

MicroMo Electronics, Inc.
14881 Evergreen Avenue
Clearwater · FL 33762-3008 · USA
Phone: +1 (727) 572-0131
Fax: +1 (727) 573-5918
Toll-Free: (800) 807-9166
Email: info@micromo.com
www.micromo.com

MA15011, english, 3. issue, 11.05.2005
Version 4.10