

Mjölner control protocol

The Mjölner 200/600 series can be controlled using the Mjölner control protocol. This protocol permits the higher system to control the Mjölner and read out the measuring data. If more than one Mjölner is connected, up to 128 instruments can be on the same bus in parallel.

In order to control the Mjölner the following steps have to be done before the protocol can be used.

1. Buy a Mjölner Win code for your Mjölner 200. Key in this code according to the Mjölner Win description located on your Mjölner Win disk in the \doc\Softcode_1.00.pdf. Start the Mjölner Win and verify, if the Mjölner Win is capable to connect to your instrument.
2. Select an instrument address you want to use with your Mjölner (1 – 127). Turn of your Mjölner, press and hold the keys F1 and F2 (the two right keys underneath the LCD display). Move to the menu item “Enable data comm.”, press ENTER and select “ON” with the + key. Now press ENTER again. The instrument now requests the “Inst. address=”, which is the address you have chosen above. Set the instrument to the desired address with the + and – key and confirm by pressing ENTER again.
3. The instrument is now ready to communicate and can be accessed with the installed address.

NOTE: When this protocol is activated, the Mjölner WIN Software protocol is deactivated. If you wish to use Mjölner Win again, you have to deactivate this protocol (reverse item 2)!

Detailed description of the Mjölner control protocol:

Numbers in this Document:	Hex numbers	0x3B
	Decimal numbers	59
	ASCII Characters	“,”

Calculated check sum: 256 - (Byte 2 + Byte 3 + Byte 4 + Byte 5 + Byte 6 + Byte 7) and 0xFF), then convert result to 2 ASCII bytes and put MSB to Byte 8 and LSB to Byte 9

Example of request below:

256 – ((0x01 + 0x00 + 0x00 + 0x00 + 0x03 + 0xE8) and 0xFF) = 0x14

Byte 8 = “1”, Byte 9 = “4”

Timing: After issuing a command, start a time out of at least 500ms before reissuing another command. Mjölner sends the answer back within this time. If no answer received, the command has not been received.

PC	System that controls the Mjölner
Mjölner	Instrument of the Mjölner series with an individual address
Adr	Actual Mjölner address
AdrPC	System address
CMD	Main command 0x00 - 0x7F
Cmd	Command
Data	Return data
CKH	Check sum higher Byte ASCII
CKL	Check sum lower Byte ASCII
CR	Carriage Return
LF	Line Feed
Start	Start character “,”

Mjölner control protocol

Request from PC to Mjölner

The Mjölner control protocol consists of a 11 byte request. A request is issued by the PC to the Mjölner. This request has the following format:

In this Example we use the following data:

Address 0x01
CMD 0x00
Cmd 1000.0, 1st Byte (LSB) 0x0, 2nd. Byte 0x0, 3rd.Byte 0x03, 4th.Byte (MSB) 0xE8
CKH "1"
CKL "4"
CR 0x0D
LF 0x0A

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10	Byte 11
Start ".," (0x3B)	Adr (0x01)	CMD (0x00)	1 st .Byte Cmd (0x00)	2 nd .Byte Cmd (0x00)	3 rd .Byte Cmd (0x03)	4 th .Byte Cmd (0xE8)	CKH "1" (0x31)	CKL "4" (0x34)	CR 13 (0x0D)	LF 10 (0x0A)

Mjölner control protocol

Answer from Mjölner to PC

The Mjölner has two possibilities to answer the above request if the address (Adr) matched the address previously set in the Mjölner. If the address does not match, Mjölner **will not answer** at all!

1. The address was ok, but the calculated check sum did not match. The answer has 22 Bytes and looks as follows:

".,",	"F"	"E"	"H"	"L"	"E"	"R"	"4"	"A"	CR (0x0D)	LF (0x0A)
".,",	"R"	"E"	"T"	"O"	"R"	"E"	"2"	"F"	CR (0x0D)	LF (0x0A)

2. The address was ok, the calculated check sum did match. The answer has 22 Bytes and looks as follows (IEEE Float Data = 304,6 = 0x43984CCD):

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10	Byte 11
Start ".,", (0x3B)	AdrPC (0x00)	CMD (0x80)	1 st .Byte Data (0xCD)	2 nd .Byte Data (0x4C)	3 rd .Byte Data (0x98)	4 th .Byte Data (0x43)	CKH "8" (0x31)	CKL "C" (0x34)	CR 13 (0x0D)	LF 10 (0x0A)
Byte 12	Byte 13	Byte 14	Byte 15	Byte 16	Byte 17	Byte 18	Byte 19	Byte 20	Byte 21	Byte 22
".,",	"R"	"E"	"T"	"O"	"R"	"E"	"2"	"F"	CR (0x0D)	LF (0x0A)

In this Example we use the following data:

Address PC 0x00
CMD 0x80 = Request CMD with MSBit set
Data 304,6, 1st Byte (LSB) 0xCD, 2nd. Byte 0x4C, 3rd.Byte 0x98, 4th.Byte (MSB) 0x43
CKH "8"
CKL "C"
CR 0x0D
LF 0x0A

Mjölner control protocol

Usable commands from PC to Mjölner

CMD:Cmd	Instrument status
0x00:100.0	<p>Get instrument status</p> <p>Data & 0x01 = 0x01, Continuous Mode on Data & 0x01 = 0x00, Continuous Mode off Data & 0x02 = 0x02, Temperature compensation on Data & 0x02 = 0x00, Temperature compensation off Data & 0x04 = 0x04, Current clamp on Data & 0x04 = 0x00, Current clamp off Data & 0x08 = 0x08, Measurement on Data & 0x08 = 0x00, Measurement off Data & 0x10 = 0x10, LED Ramp up on Data & 0x10 = 0x00, LED Ramp up off Data & 0x20 = 0x20, LED Ramp hold on Data & 0x20 = 0x00, LED Ramp hold off Data & 0x40 = 0x40, LED Ramp down on Data & 0x40 = 0x00, LED Ramp down off Data & 0x80 = 0x80, LED Error on Data & 0x80 = 0x00, LED Error off Data & 0x100 = 0x100, Polarity sense line inverse Data & 0x100 = 0x000, Polarity sense line normal Data & 0x200 = 0x200, Polarity clamp line inverse Data & 0x200 = 0x000, Polarity clamp line normal Data & 0x400 = 0x400, Result ready (this flag can only be read once, then it is reset) Data & 0x400 = 0x000, No new result available</p> <p>Example: Record PC → Mjölner (Hex) 3B 01 00 00 00 00 64 39 42 0D 0A Record Mjölner → PC (Hex) 3B 00 80 00 80 80 44 33 43 0D 0A + (";RETORE2F"CRLF)</p>
0x00:101.0	<p>Get firmware version</p> <p>Firmware version = Data (Example: 5.40)</p> <p>Example: Record PC → Mjölner (Hex) 3B 01 00 00 00 00 65 39 41 0D 0A Record Mjölner → PC (Hex) 3B 00 80 CD CC AC 40 46 42 0D 0A + (";RETORE2F"CRLF)</p>
0x00:102.0	<p>Get internal board temperature in °C</p> <p>Internal board temperature = Data (Example: 27.2)</p> <p>Example: Record PC → Mjölner (Hex) 3B 01 00 00 00 00 66 39 39 0D 0A Record Mjölner → PC (Hex) 3B 00 80 00 70 D9 41 46 36 0D 0A + (";RETORE2F"CRLF)</p>

Mjölner control protocol

CMD:Cmd Get Measurement values

0x00:1000.0

Get measuring value

Measurement value [$\mu\Omega$] = Data

(Example: 304,6)

Example: Record PC → Mjölner (Hex)

Record Mjölner → PC (Hex)

3B 01 00 00 00 03 E8 31 34 0D 0A
3B 00 80 CD 4C D6 43 34 45 0D 0A
+ (";RETORE2F"CRLF)

0x00:1001.0

Get measuring current

Measuring current [A] = Data

(Example: 120,0)

0x00:1002.0

Get temperature

Temperature [$^{\circ}\text{C}$] = Data

(Example: 20,0)

0x00:1003.0

Get sense voltage

Sense voltage, last 25 meas. [$\mu\text{V} / 10$] = Data

(Example: 979.4 → $979.4 * 4 / 100.0 = 39.17\text{mV}$)

0x00:1004.0

Get shunt voltage

Shunt voltage, last 25 meas. [$\mu\text{V} / 10$] = Data

(Example: 979.4 → $979.4 * 4 / 100.0 = 39.17\text{mV}$)

0x00:1005.0

Get clamp voltage

Clamp voltage, last 25 meas. [$\mu\text{V} / 10$] = Data

(Example: 979.4 → $979.4 * 4 / 100.0 = 39.17\text{mV}$)

CMD:Cmd

0x01:100.0

Set instrument status

Set start measurement flag

→ starts the measurement immediately

Example: Record PC → Mjölner (Hex)

Record Mjölner → PC (Hex)

3B 01 01 00 00 00 64 39 41 0D 0A

(";RETORE2F"CRLF)

CMD:Data

0x14:100.0 [A]

Set instrument current

Set the instrument current as a 32 bit float number (5.0A – max measuring current)

→ sets the current of the instrument immediately in the instrument RAM only

Example:

Set the Mjölner to 100.0 A measuring current (IEEE float 42C80000)

Record PC → Mjölner (Hex)

Record Mjölner → PC (Hex)

3B 01 14 00 00 C8 42 45 31 0D 0A

(";RETORE2F"CRLF)