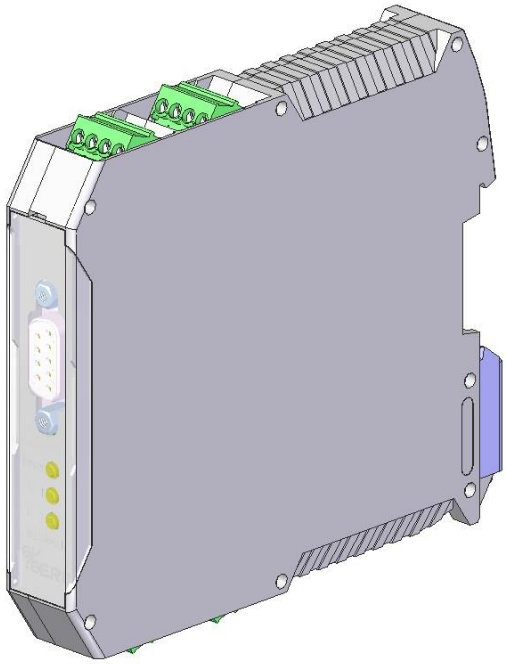
Serial Interface VC Mini

Programming instructions for using the serial interface at the VC Mini control

Rev. 2.00

Valid for Firmware Version:
VCM 1.03.03



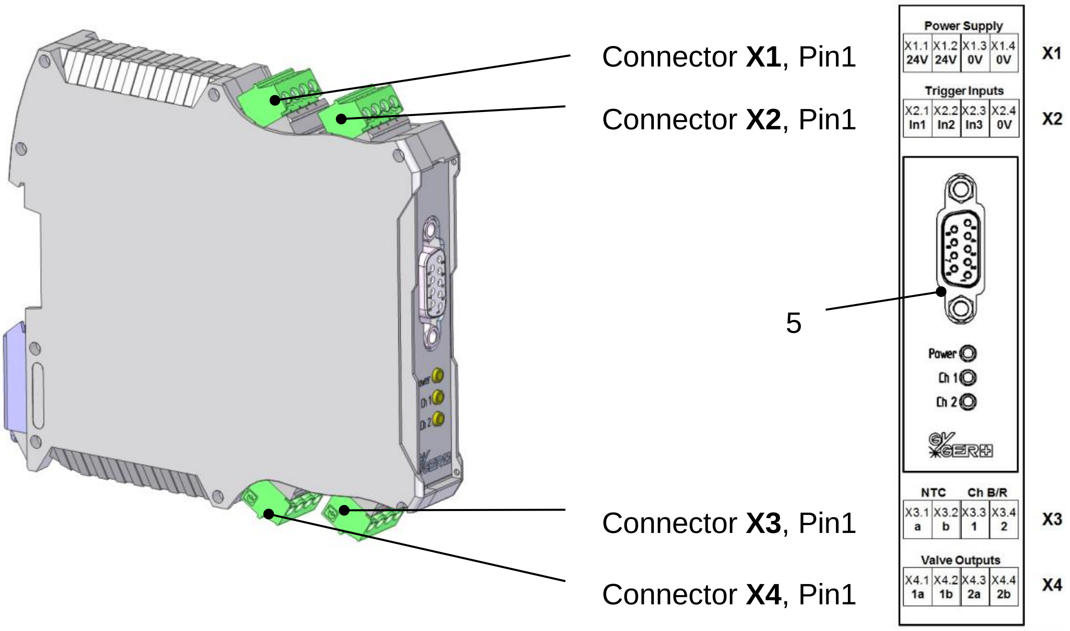
Fritz Gyger AG
Bodmerstr.12
CH-3645 Gwatt (Thun)

Tel: +41 (0)33 336 22 77
Fax: +41 (0)33 336 64 34
Email: info@fgyger.ch
Web: www.fgyger.ch

Table of content

1 Interface concept and hardware information

1.1 Electrical connections / Pin assignment



Connector	Pin	Function	Electrical data / comments
X1 Power supply	1	+24 V DC	Power supply 24 V DC, max 1A
	2	+24 V DC	
	3	0 V (GND)	
	4	0 V (GND)	
X2 Trigger inputs	1	Trigger valve 1*	12 V / 24V Signal inputs Vin (low): max. 5V Vin (high): min. 8V Power consumption at 24V = 4-5 mA
	2	Trigger valve 2*	
	3	Trigger special functions	
	4	0V (GND)	
X3 Signal outputs	1	Not used	
	2	Not used	
	3	Busy out valve 1**	24 V DC Signal outputs Load current: max. 0.5 A Protection: Short circuit, overload
	4	Busy out valve 2**	
X4 Valve outputs	1	Valve 1a	Max. nominal current 0.85A (100 % ED) Max. peak current 1.3 A (<10 % ED) Protection: Overload, fast cutoff for inductive loads, limited short circuit protection.
	2	Valve 1b	
	3	Valve 2a	
	4	Valve 2b	
5 RS-232	2	RXD	
	3	TXD	
	5	GND	

*Trigger-inputs are only active, when the corresponding triggermode is set via software.

**Busy-Out generates a high-signal, as long as the valve is opened.
Parallel the corresponding LED lights (B/R1=valve 1, B/R2=valve 2)

Connector types:
X1 – X4: Phoenix Contact MC 1,5/ 4-ST-3,5
RS232: D-Sub 9 pol female

1.2 General information about the RS 232 interface

The VC MINI- controller has a serial interface via which the control can be fully configured and controlled.

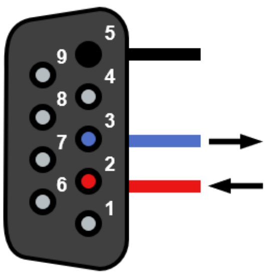
The connection for the interface is integrated in the frond plate (9pole D-Sub Female)

Parameter	Value
Interface Type	Serial point-to-point connection (UART) via RS232 or USB to RS232 converter
Interface Operation	asynchronous full-duplex
Baud rate	38400 Other baud rates can be set via parameters after a successful connection, this is the default value. See command list for details
Data bits	8
Parity	none
Stop bits	1
Flow control	none

Please read before programming via the serial interface the user manual for the MVC/VC software (separate document). There, important fundamental relationships are explained for the correct operation of the controller.

1.2.1 Pin assignment RS-232 interface

Function	Pin	D-Sub 9Pin female
GND	5	
TXD	3	
RXD	2	



With a 9-pin 1: 1 connection cable, the interface can be directly connected to an RS-232 remote site.

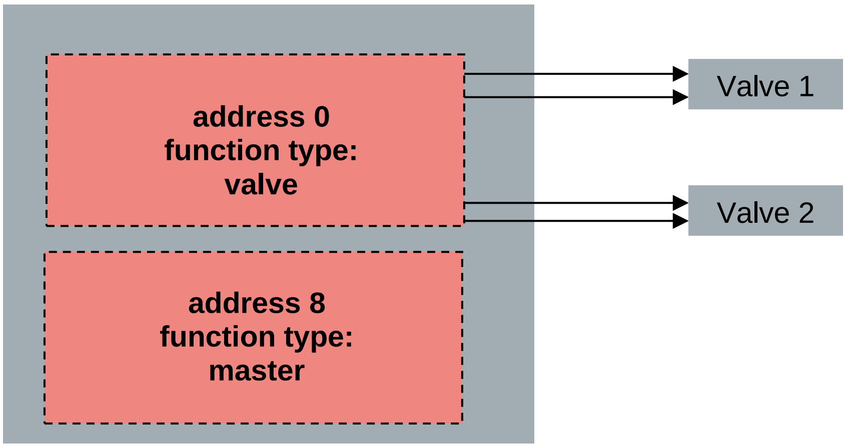
Alternatively, our USB converter can be used, the interface will appear as a virtual COM port on the PC.

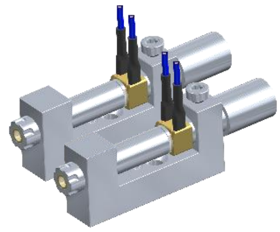
2 Communication via the serial interface

2.1 Modultypes and addressing

In contrast to the MVC 1 controller which consists of a master module and up to 4 power modules with independent microcontrollers, the VC Mini contains only a microcontroller in which all functions are united.

For greatest compatibility with the MVC 1 command logic, the addressing structure is maintained even in the VC Mini. This also helps the logical division of the functions.





For each function type, a corresponding command set is available.

2.2 Basics of communication

2.2.1 Command types

The VC Mini instruction set distinguishes 3 different types of commands

- **Type A = execution command**
  These are single-digit ASCII characters, which trigger an action in the controller.
  Example: Y (Y Large) triggers a single shot of valve 1.

Send	Y				
Receive		Y	LF	CR	>

=> The transmitted command is sent back for acknowledgment; in addition, a line feed / carriage return is sent. The prompt character > is the final acknowledgment, and signals that the control is ready for input again.

- **Type B = parameterization command**
  These commands are composed of a string, which transmits a value to the controller.
  Example: 0 * selects the module address 0 as the active address for communication.
  Example: 500T transmits the value 500 for parameter T to the active address.

Send	5		0		0		T				
Receive		5		0		0		T	LF	CR	>

=> Each transmitted character is sent back for acknowledgment.
Again, line feed, carriage return, and the prompt character mark the end.
=> The characters to be sent per command can also be sent in one piece to the VC Mini, the acknowledgment of each individual character must not be awaited. But it is important to convey any new commands, as long as the termination character (prompt>) has not yet been received.

- **Type C = query command parameters**
  These are single-digit ASCII characters, which read a parameter value that the VC Mini controller then returns as a string.
  Example: "i" queries the value of parameter i. In response comes back .i00000233.

Send	i													
Receive		.	i	0	0	0	0	0	2	3	3	LF	CR	>

=> The controller acknowledges the request by returning the value in the above format with leading zeros. Again, line feed, carriage return and the prompt character mark the end.

Transmission timing

The next command can be sent when the acknowledgment has been received in full.
(received prompt character >)

For the instruction execution times of individual commands, refer to the section "instruction set".

2.2.2 *Connection setup with a simple terminal program*

For first learning steps in the serial communication with the VC Mini, the connection via a simple terminal program, (for example Windows Hyper Terminal) is suitable.

In newer operating systems, Hyper Terminal is no longer included in the operating system. There are appropriate download Internet sources with alternatives.

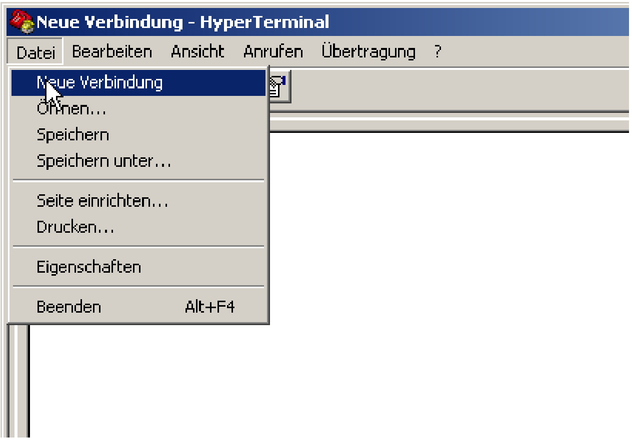
Alternatively, for example, the open source program "Tera Term Pro" can be used.

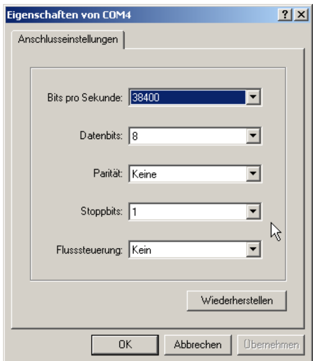
The procedure is analogous to that described here based on the Windows Hyper Terminal.

Prerequisites:
The VC Mini controller must be connected to either the RS232 or USB. When using USB, the converter driver must be installed. VC Mini must be turned on.

Start your terminal program

Create a new connection (if you are running Hyper Terminal for the first time, you are routed directly to the next step).



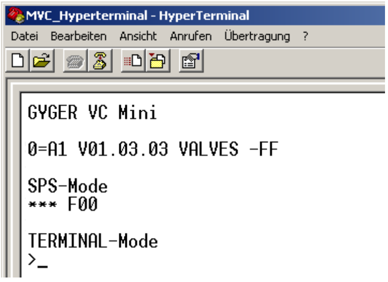


Give the connection a name, select the COM port to which the controller is connected, and make the port settings as shown. Confirm by clicking OK.

The VC Mini controller is now connected, but the transmission screen is still empty.

First commands
Press the [Esc] key. This command switches the control back and forth between the PLC (shown as SPS) - and Terminal Mode.
The controller acknowledges this with the respective Listings.

When switching to PLC mode, the controller outputs a small report which shows the control type, the installed firmware, and the set operating mode (here F00).



When you see „Terminal mode“ and the prompt character, the control is ready to receive data.

Now you can direct input via keyboard and work with the system.
You can, for example, practice the programming examples from Chapter *0 Programming* examples*.*

3 Instruction set

3.1 Instruction execution times

The instruction execution times for the work via the serial interface are in the millisecond range. Here applies a range of 2-15 ms for most commands. The time applies from receipt of the last character to the return of the prompt character. That means until the controller is ready to receive a new command.

The interface should therefore be used in time-critical applications only for parameter setting, the release of dispensing shots should then take place via the external trigger inputs, here shot sequences can be triggered in real time (µs delay).

If you still need more detailed information about the timing, contact us.

3.2 Global commands

These commands can be executed regardless of which module address or module type is currently active.

Input format: [command character]
e.g. „ESC“ (pressing the Escape key) switches the control between the terminal and the PLC mode.

Function	Command sign (ASCII)	Value range	Remark	Command type
Switching Terminal-Mode, PLC-Mode	ESC		Only in terminal mode, you can communicate with the controller It is usually not necessary to leave this mode, switching to the PLC mode is used for special customer-specific applications.	A
Select Module adress	*	0-8	Module address can be selected at any time.	B
Query currently selected module address	=		Outputs the currently selected module address and module type.	C
Re-initialization of the system, view function type and version of all modules.	CTRL[R]		After output of information, control is automatically switched to the PLC mode.	C

3.3 *Instruction set function type „Master“*

Select the module address 8 (master module) before the execution of these commands.

3.3.1 Setting of parameter values

Input format: [value][command character]
e.g. „01F“ selects the PLC function number 01

Function	Command sign (ASCII)	Value range	Remark	Command type
PLC funtion number	F	00-01	00=Standard mode Activated trigger modes or heating systems are turned off at every power-on process again. For each valve, the first parameter set is activated at power-on. 01=Last State Restore Control transfers at power-on in the same condition as was the case before power-off. That means, activated trigger modes, heaters, and loaded parameter sets are reactivated.	B
Selecting parameter set selection type	G	0-1	0= Parameter set selection via Software / RS232 1 = Selection of 2 parameter sets via the external hardware input X2.3 X2.3 low = parameter set 0n X2.3 high = parameter set 1n **At level change switching takes max 0.15s in the electronics! During this time no shots are fired.** The parameter set selection via external inputs is only active if: -PLC function number = F01 -Control is in the PLC mode (command esc causes switching)	B
Selecting Baud Rate	%	0-5	0=9600, 1=19200, 2=38400, 3=57600, 4=115200, 5=230400	B

These entries are stored non-volatile in EEPROM.

3.3.2 Reading out parameter values

Input format: [command character]
e.g. "g" reads the stored in controller parameter set selection-type

Function	Command sign (ASCII)	Value range	Remark	Command type
Read parameter set selection-type	g	0-1		C

=>The set baud rate cannot be read.

=> The selected PLC function number can be read indirectly by changing in the PLC mode (command ESC). When switching the VC Mini shows, among other things, this information (see Section “First commands” in Chapter *0)*

3.4 Instruction set function type "valve"

Select the module address (0) with function type “valve” before the execution of these commands.

3.4.1 Shot control via the external hardware trigger inputs

After activating the desired trigger mode, the triggering is nearly in real time via the trigger inputs. The delays are in the region of a few microseconds.

While a trigger mode is active, no further entries are possible on the corresponding module. Only the command "stop" (stop of a external trigger mode) or “esc” (switching terminal-SPS mode) can then be performed.

Input format: [command character]
e.g. "X" activates the external trigger for triggering of single shots on valve 1 and 2.

Function	Command sign (ASCII)	Value range	Remark	Command type
Single shot mode V1/V2 ON, triggering via ext. Hardware-Trigger	X		applies to V1 and V2 With a positive edge at the appropriate input signal, a single shot of the associated valve is generated. e.g. Input X2.1 for valve 1. Uses valve parameters: A, B, C, D Parameter C is used in so far that no new pulse is processed before the time C has expired.	A
Switch on single shot mode V1/V2, control via ext. hardware pulse	T		applies to V1 and V2 Pulse duration = opening time As long as the corresponding input signal is high, the valve is opened. => Positive edge: the valve opens. Negative edge: closes the valve again. Uses valve parameters: A, D	A
Shot series mode V1/V2 ON, triggering via ext. HW-Trigger	P		applies to V1 and V2 With a positive edge at the appropriate input signal a shot series of the corresponding valve is generated. Uses valve parameters: A, B, C, D, G Caution: If G = 0 = infinite set, triggering does not work (no dispensing)	A
Endless shot series mode V1/V2 ON, control via HW Enable (level-controlled).	L		applies to V1 and V2 As long as the corresponding input signal is high, the associated valve dispenses with the set frequency and opening time. Uses valve parameters: A, B, C, D	A
Stop Trigger Mode	S		Stops a running trigger mode (X, T, P, L)	A

- **Power-on-behavior**
  The behavior depends on the set PLC function mode on the master module:
  • F00 (standard mode): At power-on no hardware trigger modes are active.
  • F01 (last selected mode restore): If a trigger mode at power-off was active, it is automatically reactivated after power-on.

3.4.2 Shot release via software

The command delay after receiving the command character until the first shot is fired is approximately 2ms. For time-critical applications you should use the capabilities of shot with hardware trigger.

Input format: [command character]
e.g. "Y" triggers a single shot of Valve1 if module address 0 is selected.

Function	Command sign (ASCII)	Value range	Remark	Command type
Single shot V1	Y		Status query via command „q" Uses valve parameters: A, B, C *, D * = C is used only for the status inquiry. Status q is only correct shown if C> B	A
Single shot V2	Z		Status query via command „q" Uses valve parameters: A, B, C *, D * = C is used only for the status inquiry. Status q is only correct shown if C> B	A
Single shot V1/V2 together	V		Status query via command „q" Uses valve parameters: A, B, C *, D * = C is used only for the status inquiry. Status q is only correct shown if C> B	A
Shot series V1	Q		Can be terminated with stop command at any time Status query over command "q" Uses valve parameters: A, B, C, D, G	A
Shot series V2	R		Can be terminated with stop command at any time Status query over command "q" Uses valve parameters: A, B, C, D, G	A
Endless shot series mode V1/V2 ON	U		applies to V1 to V4 (both module halves) running endlessly until the stop command Uses valve parameters: A, B, C, D	A
Stop shot series	S		Stop an active shot series modes, (Q, R, U)	A

=>If immediately after a Y or Z command (single shot), a shot-series is started with Q or R, this can start only after parameter C (cycle time) has elapsed from the previous single shot.

3.4.3 Setting of parameter values

Parameter sets

4 complete parameter sets can be stored directly in the EEPROM of the microcontroller per valve. Each parameter set contains the parameters A, B, C, D and G. Via the n respectively N parameter the desired parameter set are activated / selected or written.
Activation is valid until another parameter set is selected.

0n= 1. Parameter set of valve1	4n= 1. Parameter set of valve2
1n= 2. Parameter set of valve1	5n= 2. Parameter set of valve2
2n= 3. Parameter set of valve1	6n= 3. Parameter set of valve2
3n= 4. Parameter set of valve1	7n= 4. Parameter set of valve2

The activation is in pairs, that is, when 0n is selected, automatically activates also 4n (n + 4). During firing, the parameters of the first parameter set is then used for both valves (0n, 4n)
Likewise, if 5n is activated, automatically activated also 2n (n-4).

> The deposit of multiple parameter sets makes sense especially when using external parameter set selection (parameter G on master module).

Power-on-behavior

The behavior depends on the set PLC function mode on the master module:
- F00 (standard mode): At power-on is always the first parameter set-pair automatically loaded (0n, 4n).
- F01 (last state restore): It is the parameter set-pair loaded that was active at power-off.

Input format: [value] [command character]
e.g. "400A" transmits a peak time of 400µs on the active parameter set in the memory of the microcontroller.

Function	Command sign (ASCII)	Value range	Remark	Command type
Set PEAK Time	A	10-65'535	Input in microseconds If OpenTime < peak time the peak time is cutted down, ie Peak time = Open time. Reasonable values: SMLD 300: 150µs SMLD 300G: 400µs	B
Set OPEN Time	B	10-9'999'999	Input in microseconds Reasonable values: 400 – 9'999'999µs	B
Set CYCLE Time	C	10-9'999'999	Input in microseconds Firing frequency = (1 / Cycle Time) Condition: CYCLETIME > OPEN Time A new shot can be triggered only when the cycle time has expired. In connection with the commands Q, R and U the Cycle Time is used to set the dispensing frequency.	B
Set PEAK current	D	0-15	The peak current is about: Ip = 450mA + (D * 50mA) Although the peak power can be set per valve, it must be the same for each module half (V1, V2)! Reasonable value: 11 = 1000mA	B
Set Nr. of shots	G	0-65535	0= endless. To use with the shot-series commands Q and R, as well as with the hardware trigger commands	B
Activate / choose parameter set	n	0-7	Load the corresponding parameter set from the EEPROM into the main memory of the microcontroller. Likewise, the parameter set n + 4, or n-4 (pairs activation for both valves).	C
Safe parameter set into EEPROM	N	0-7	0-3 = parameter set for valve 1 4-7 = parameter set for valve 2	B
Zeroing Shotcounter valve 1	I		Zeroing the shot counter of valve 1 (See details in the "reading of parameter values"	B
Zeroing Shotcounter valve 2	J		Zeroing the shot counter of valve 2	B

Important: The entries on these parameters are not automatically saved non-volatile, they are transmitted only in the RAM memory of the microcontroller.
Use therefore always the command "N" to store values in the EEPROM non-volatile.

This concept has the advantage, for example, if you dispense with a valve, and want to define a different opening time via the interface for each shot, then the EEPROM is not constantly rewritten (limited number of write cycles).

Attention: With incorrect entries the valve spool can be burned. Consult our guide to MVC/VC software for further instructions!

3.4.4 *Reading out parameter values*

Input format: [command character]
e.g. "a" reads the peak time stored in the controller for the selected parameter set.

Function	Command sign (ASCII)	Value range	Remark	Command type
Read PEAK Time	a	0-65'535	Output in microseconds	C
Read OPEN Time	b	0-9'999'999	Output in microseconds	C
Read CYCLE Time	c	0-9'999'999	Output in microseconds	C
Read PEAK current	d	0-15	The peak current is about: Ip=450mA + (D*50mA)	C
Read number of shots	g	0-65535		C
Query currently active parameter set	p	0-7		C
Read valve status	q		Used for valve status query in the modes Y, Z, V, Q and R Output of 2 statusbits as ASCII coded decimal number. Binary coding: Bit0 = 1 (decimal 1): V1 = active Bit4 = 1 (decimal 16): V2 = active Example: If Bit0 and Bit4 are set = result: .q00000017 (16 + 1)	C
Read shot counter valve 1	y	0-9'999'999	The total shots fired per valve are automatically counted. The count can be read out with this command. The shot counter is volatile, at power-on it is set to 0.	C
Read shot counter valve 2	z	0-9'999'999	ditto for valve 2	C
Read total shot counter valve1 (Bits 31-24)	u	0-255		C
Read total shot counter valve1 (Bits 23-0)	v	0-16777215	Total shots V1 = (value [u] * 16'777'216) + (value [v]) Value range V1 is 0 to 4'294'967'295 (32-Bit) Before query run command I (add volatile shotcounter to total counter and clear the volatile counter)	C
Read total shot counter valve2 (Bits 31-24)	w	0-255		C
Read total shot counter valve2 (Bits 23-0)	x	0-16777215	Total shots V2 = (value [w] * 16'777'216) + (value [x]) Value range V2 is 0 to 4'294'967'295 (32-Bit) Before query run command J (add volatile shotcounter to total counter and clear the volatile counter)	C

4 Programming examples

Principle
1. Choice of module address (if not already active)
2. execution of the command.

A once selected module address will remain active until another is selected.

4.1 Valve module: Dispensing with non-volatile saved (EEPROM) values

The valve module gets values transmitted (for all valves the same), then test shot sequences are triggered at every valve, at the end, the module is set to the external trigger mode to trigger shot series via the hardware inputs.

Sent commands	Received characters	Comment
0*	**0*>**	**Address 0 is selected**
0n	0.n>	Activation of the 1st parameter set pair (0n, 4n)
400A	400A>	Peak time of 400µs is transmitted (into controller RAM)
1000B	1000B>	Opening time of 1000µs is transmitted
100000C	100000C>	Cycle Time of 100'000µs = 0.1s is transmitted
10G	10G>	Number of shots 10 is transmitted
0N	0N>	The values are stored in non-volatile EEPROM at parameter set 1 for Valve A1
4N	4N>	The values are stored in non-volatile EEPROM at parameter set 1 for Valve A2
Q	Q>	Triggering shot series Valve A1 = 10 shots at intervals of 0.1s
R	R>	Triggering shot series Valve A2 = 10 shots at intervals of 0.1s
P	P>	The whole module A is set in the ext. trigger mode for triggering shot series via the hardware inputs.

4.2 Valve module: Read out parameter values

The parameter values of the valves shall be queried. Here, the 1st parameter set per valve is queried.

Sent commands	Received characters	Comment
0*	**0*>**	**Address 0 is selected**
0n	0.n>	Selecting parameter set 1 of valve 1
a	.a00000400>	Read peak time
b	.b00001000>	Read opening time
c	.c00100000>	Read cycle time
g	.g00000010>	Read number of shots
4n	4.n>	Selecting parameter set 1 of valve 2
a	.a00000400>	Read peak time
b	.b00001000>	Read opening time
c	.c00100000>	Read cycle time
g	.g00000010>	Read number of shots

4.3 Valve module: Dispensing with temporary saved (RAM) values

For special purposes (e.g. shot release via software with constantly changing parameters), it is helpful to write the parameters only temporarily in the RAM memory of the controller.

So stores are not unnecessarily made on the EEPROM. For this purpose, just leave the store instruction away (e.g. 0N stores the 1st parameter set of valve 1 in the EEPROM).

The RAM memory is cleared when:
-Another parameter set is loaded (0n, 4n)
-Another address is selected (0*, 8*)
-An external trigger mode is activated

4.3.1 Dispensing single shots with temporary saved (RAM) values

In the example, two shots are released on valve 1 and 2 with different opening times.

Sent commands	Received characters	Comment
0*	0*>	Address 0 is selected
0n	**0.n>**	**Activation of the 1st parameter pair (0n, 4n)** =>The non-volatile stored values are loaded into the controllers RAM. All parameters that are set afterwards will overwrite the values in the RAM. Focus for temporary shot release is on Valve 1
2000B	2000B>	New opening time is transmitted to the controller RAM.
Y	Y>	Triggering one shot Valve A1 with temporary opening time
q	.q00000000>	Query status (optional) => Bit 0 goes 0 after the dispensing is done
3000B	3000B>	New opening time is transmitted to the controller RAM.
Y	Y>	Triggering one shot Valve A1 with temporary opening time
q	.q00000000>	Query status (optional) => Bit 0 goes 0 after the dispensing is done
4n	**4.n>**	**Activation of the 1st parameter pair (0n, 4n)** =>The non-volatile stored values are loaded into the controllers RAM. All parameters that are set afterwards will overwrite the values in the RAM. Focus for temporary shot release is on Valve 2
2000B	2000B>	New opening time is transmitted to the controller RAM.
Z	Z>	Triggering one shot Valve A2 with temporary opening time
q	.q00000000>	Query status (optional) => Bit 0 goes 0 after the dispensing is done
3000B	3000B>	New opening time is transmitted to the controller RAM.
Z	Z>	Triggering one shot Valve A2 with temporary opening time
q	.q00000000>	Query status (optional) => Bit 0 goes 0 after the dispensing is done

=>If necessary, additional parameters than the opening time can be temporarily stored in the controller RAM (e.g. peak time, peak current, etc.)

4.3.2 Dispensing shot series with temporary saved (RAM) values

In the following example a shot series is made successively at 2 valves. With opening times of 1000µs, frequency 100 Hz, 1000 shots = 10s total dispensing time per valve.

Sent commands	Received characters	Comment
0*	0*>	Address 0 is selected
0n	**0.n>**	**Activation of the 1st parameter pair (0n, 4n)** =>The non-volatile stored values are loaded into the controllers RAM. All parameters that are set afterwards will overwrite the values in the RAM. Focus for temporary shot release is on Valve 1
1000B	1000B>	New opening time is transmitted to the controller RAM.
Y	Y>	Triggering one shot Valve A1 with temporary opening time
10000C	10000C>	Cycle time of 10'000µs is transmitted to controller RAM = 100 Hz dispensing frequency
1000G	1000G	Number of shots is transmitted to controller RAM
Q	Q>	Triggering shot sequence valve 1
q	.q00000001>	Query status (optional) => Bit 0 is 1 while dispensing = 10 seconds
q	.q00000000>	Query status (optional) => Bit 0 goes 0 after the dispensing is done
4n	**4.n>**	**Activation of the 1st parameter pair (0n, 4n)** =>The non-volatile stored values are loaded into the controllers RAM. All parameters that are set afterwards will overwrite the values in the RAM. Focus for temporary shot release is on Valve 2
1000B	1000B>	New opening time is transmitted to the controller RAM.
10000C	10000C>	Cycle time of 10'000µs is transmitted to controller RAM = 100 Hz dispensing frequency
1000G	1000G	Number of shots is transmitted to controller RAM
R	R>	Triggering shot sequence valve 2
q	.q00000001>	Query status (optional) => Bit 0 is 1 while dispensing = 10 seconds
q	.q00000000>	Query status (optional) => Bit 0 goes 0 after the dispensing is done

=>If necessary, additional parameters can be temporarily stored in the controller RAM (e.g. peak time, peak current, etc.)