

# Serielle Schnittstelle MVC-1

## Programmieranleitung zur Benutzung der seriellen Schnittstelle an der MVC Steuerung.

**Rev. 2.00**

**Gültig ab Firmware Version:**

MC 3.01.00

HC 2.04.01

VC 2.00.01



**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](mailto:info@fgyger.ch)  
Web: [www.fgyger.ch](http://www.fgyger.ch)

Alle Rechte ausdrücklich vorbehalten. Vervielfältigung oder Mitteilung an Dritte, gleichgültig in welcher Form, ist ohne unsere schriftliche Genehmigung nicht gestattet.

## **Inhaltsverzeichnis**

<b>1</b>	<b>SCHNITTSTELLENKONZEPT UND HARDWAREINFORMATIONEN.....</b>	<b>3</b>
1.1	Allgemeine Angaben zur RS 232 Schnittstelle.....	3
1.1.1	Pinbelegung RS-232 Schnittstelle.....	3
<b>2</b>	<b>KOMMUNIKATION ÜBER DIE SCHNITTSTELLE .....</b>	<b>4</b>
2.1	Modultypen und Adressierung.....	4
2.1.1	Adressierung der Module bzw. Modulhälften.....	5
2.2	Grundlagen der Kommunikation.....	6
2.2.1	Befehlstypen.....	6
2.2.2	Verbindungsaufnahme mit Terminalprogramm .....	7
<b>3</b>	<b>BEFEHLSSATZ.....</b>	<b>9</b>
3.1	Befehlslaufzeiten.....	9
3.2	Globale Befehle .....	9
3.3	Befehlssatz Funktionstyp „Master“ .....	10
3.3.1	Globale Dosierbefehle .....	10
3.3.2	Setzen von Parameterwerten.....	11
3.3.3	Auslesen von Parameterwerten .....	12
3.4	Befehlssatz Funktionstyp „Ventil“ .....	13
3.4.1	Schusskontrolle über die externen Hardware-Triggeringänge.....	13
3.4.2	Schussauslösung über Software.....	14
3.4.3	Setzen von Parameterwerten.....	15
3.4.4	Auslesen von Parameterwerten .....	17
3.5	Befehlssatz Funktionstyp „Heizung“ .....	18
3.5.1	Steuerbefehle .....	18
3.5.2	Setzen von Parameterwerten.....	18
3.5.3	Auslesen von Parameterwerten .....	19
<b>4</b>	<b>PROGRAMMIERBEISPIELE .....</b>	<b>20</b>
4.1	Heizungsmodul.....	20
4.2	Ventilmodul: Dosieren mit nichtflüchtig gespeicherten Werten .....	21
4.3	Ventilmodul: Parameterwerte abfragen .....	22
4.4	Ventilmodul: Dosieren mit temporär gespeicherten Werten .....	23
4.4.1	Einzelschüsse dosieren mit temporären Werten .....	24
4.4.2	Schussserien dosieren mit temporären Werten.....	25

# 1 Schnittstellenkonzept und Hardwareinformationen

Die MVC-Steuerung verfügt über eine serielle Schnittstelle, über welche die Steuerung vollständig parametrisiert und kontrolliert werden kann.

## 1.1 Allgemeine Angaben zur RS 232 Schnittstelle

Parameter	Wert
Schnittstellenart	Serielle Punkt-zu-Punkt-Verbindung (UART) via RS232 oder USB-RS232 Konverter
Schnittstellenbetrieb	asynchron voll duplex
Baudrate	38400 Andere Baudraten können nach erfolgreicher Verbindung via Parameter eingestellt werden, dies ist der Default-Wert. Siehe Kommandoliste für Details
Datenbits	8
Parität	keine
Stoppbits	1
Flusssteuerung	keine

Der Anschluss für die Schnittstelle befindet sich in der Mitte der Anschlussklemmen (9pol D-Sub Female).

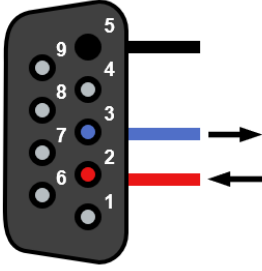


RS-232-Interface



Bitte lesen sie vor der Programmierung über die serielle Schnittstelle die Bedienungsanleitung für die MVC Software durch (separates Dokument). Dort sind wichtige Grundzusammenhänge erklärt zum korrekten Betrieb der Steuerung.

### 1.1.1 Pinbelegung RS-232 Schnittstelle

Funktion	Pin	D-Sub 9Pol Buchse
GND	5	
TXD	3	
RXD	2	

Mit einem 9-poligen 1:1 Verbindungskabel kann die Schnittstelle direkt mit einer RS-232 Gegenstelle verbunden werden.

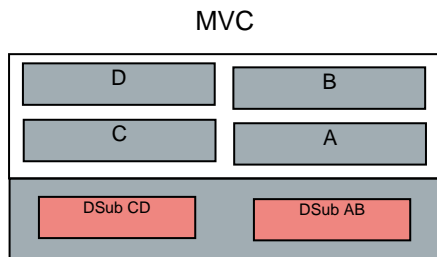
Alternativ kann unser USB-Konverter eingesetzt werden, die Schnittstelle erscheint dann als virtueller COM-Port im PC.

## 2 Kommunikation über die Schnittstelle

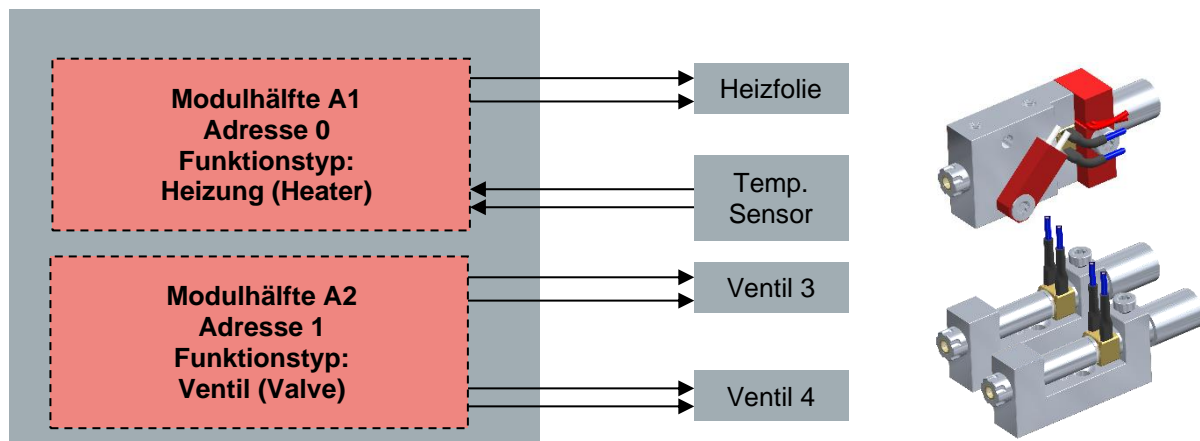
### 2.1 Modultypen und Adressierung

Jede Steuerung besteht aus einem Master-Modul und aus einem, zwei, drei oder vier Leistungsmodulen.

Die Leistungsmodule werden benannt mit den Buchstaben A, B, C, D. Entsprechend sind die Klemmenbeschriftungen zuzuordnen. Siehe dazu näheres im Manual zur MVC-Steuerung.



Jedes Modul ist technisch noch unterteilt in 2 Modulhälften. Jede Modulhälfte besitzt einen eigenständigen Microcontroller.



Die Modulhälften können verschiedene Funktionen erfüllen, je nach installierter Firmware (muss im Werk erfolgen).

Möglicher Funktionstyp einer Modulhälfte:

- Ventiltreiber =>2 Ventile können angesteuert werden
- Heizung =>1 Heizung kann angesteuert werden.
- Schrittmotor =>1 Schrittmotor kann angesteuert werden (Derzeit nur für Kundenspezifische Anwendungen)

Je nach installierter Anzahl Module und definiertem Funktionstyp lassen sich so vielfältige Kombinationen zusammenstellen, z.B. mit 4 Ventilmodulen kann eine Steuerung für 16 Ventile realisiert werden, etc.

Obiges Beispiel zeigt Modul A, welches als Ventil-Heiz-Kombination definiert ist. Es wäre auch denkbar, dass beide Modulhälften mit Funktionstyp „Ventil“ definiert sind, dann wäre es ein reines Ventilmodul.

Jede Modulhälfte hat eine Adresse (hier gezeigt 0 und 1) mit welcher das Modul angesprochen werden kann.

Es steht für jeden Funktionstyp ein entsprechender Befehlssatz zur Verfügung.

### 2.1.1 Adressierung der Module bzw. Modulhälften

			Beispielkonfiguration 1 Bestückte Module: A, B, C, D Konfiguration als reiner Ventilcontroller für 16 Ventile		Beispielkonfiguration 2 Bestückte Module: A, B, C, D Konfiguration als Ventil-Heiz- Kombination für 2 Heizungen, 12 Ventile		Beispielkonfiguration 3 Bestückte Module: A, C Konfiguration als Ventil-Heiz- Kombination für 2 Heizungen, 4 Ventile			
Stecker	Pin	Modul	Funktion	Funktionstyp	Funktion	Funktionstyp	Funktion	Funktionstyp	Adresse	
1 25-Pol D-Sub- Stecker	1+14	A	Ventil A1	Ventil	Heizfolie	Heizung	Heizfolie	Heizung	0	
	2+15		Ventil A2		-		-			
	3+16		-		Temperatursensor		Temperatursensor			
	4+17		-	Ventil	-	Ventil	-	Ventil		1
	5+18		Ventil A3		Ventil A3		Ventil A3			
	6+19		Ventil A4		Ventil A4		Ventil A4			
	8+20	B	Ventil B1	Ventil	Ventil B1	Ventil		-	2	
	9+21		Ventil B2		Ventil B2					
	10+22		-		-					
	11+23		-	Ventil	-	Ventil	-	-		3
	12+24		Ventil B3		Ventil B3					
	13+25		Ventil B4		Ventil B4					
2 25-Pol D-Sub- Stecker	1+14	C	Ventil C1	Ventil	Heizfolie	Heizung	Heizfolie	Heizung	4	
	2+15		Ventil C2		-		-			
	3+16		-		Temperatursensor		Temperatursensor			
	4+17		-	Ventil	-	Ventil	-	Ventil		5
	5+18		Ventil C3		Ventil C3		Ventil C3			
	6+19		Ventil C4		Ventil C4		Ventil C4			
	8+20	D	Ventil D1	Ventil	Ventil D1	Ventil		-	6	
	9+21		Ventil D2		Ventil D2					
	10+22		-		-					
	11+23		-	Ventil	-	Ventil		-		7
	12+24		Ventil D3		Ventil D3					
	13+25		Ventil D4		Ventil D4					
		Master	Master	Master	Master	Master	Master	Master	8	

## 2.2 Grundlagen der Kommunikation

### 2.2.1 Befehlstypen

Der MVC-Befehlssatz unterscheidet 3 verschiedene Befehlstypen

- **Typ A = Ausführungsbefehl**

Dies sind einstellige ASCII-Zeichen, welche eine Aktion in der Steuerung auslösen.

Beispiel: Y (Gross Y) löst einen Einzelschuss von Ventil 1 aus.

Senden	Y				
Empfangen		Y	LF	CR	>

=> Der gesendete Befehl wird zur Quittierung zurückgesandt, zusätzlich wird ein Line Feed / Carriage Return gesandt. Das Prompt-Zeichen > bildet den Abschluss der Quittierung, und signalisiert, dass die Steuerung wieder eingabebereit ist.

- **Typ B = Parametrierbefehl**

Diese Befehle bestehen aus einer Zeichenkette, welche z.B. einen Wert übermittelt an die Steuerung.

Beispiel: 0\* wählt die Moduladresse 0 aus als aktive Adresse für die Kommunikation.

Beispiel: 500T übermittelt den Wert 500 zum Parameter T an die aktive Adresse.

Senden	5		0		0		T				
Empfangen		5		0		0		T	LF	CR	>

=> Jedes übermittelte Zeichen wird zur Quittierung zurückgesandt.

Wiederum bilden Line-Feed, Carriage Return und das Prompt-Zeichen den Abschluss.

=>Die zu sendenden Zeichen pro Befehl können auch an einem Stück an die MVC übermittelt werden, die Quittierung jedes einzelnen Zeichens muss nicht abgewartet werden. Wichtig ist jedoch, keine neuen Befehle zu übermitteln, solange das Abschlusszeichen (Prompt >) noch nicht empfangen wurde vom Sender.

- **Typ C = Parameter-Abfrage-Befehl**

Dies sind einstellige ASCII-Zeichen, welche einen Parameterwert auslesen, den die MVC-Steuerung dann als Zeichenkette zurückgibt.

Beispiel: „i“ fragt den Wert von Parameter i ab. Als Antwort kommt .i00000233 zurück.

Senden	i													
Empfangen		.	i	0	0	0	0	0	2	3	3	LF	CR	>

=>Die Steuerung quittiert die Anfrage durch Rückgabe des Werts im obigen Format mit führenden Nullen. Wiederum bilden Line-Feed, Carriage Return und das Prompt-Zeichen den Abschluss.

### Übermittlungstiming



Der nächste Befehl kann gesandt werden, wenn die Quittierung vollständig empfangen wurde. (Prompt-Zeichen > empfangen)

Die Befehlslaufzeiten einzelner Befehle entnehmen sie bitte im Kapitel „Befehlssatz“.

### 2.2.2 Verbindungsaufnahme mit Terminalprogramm

Für erste Lernschritte in der seriellen Kommunikation mit der MVC eignet sich die Verbindung via einfachem Terminalprogramm, z.B. Windows Hyperterminal.

Bei neueren Betriebssystemen ist dieses nicht mehr enthalten im Betriebssystem. Es gibt entsprechende Downloadquellen im Internet mit Alternativen

Alternativ kann z.B. das Opensource Programm „Tera Term pro“ eingesetzt werden.

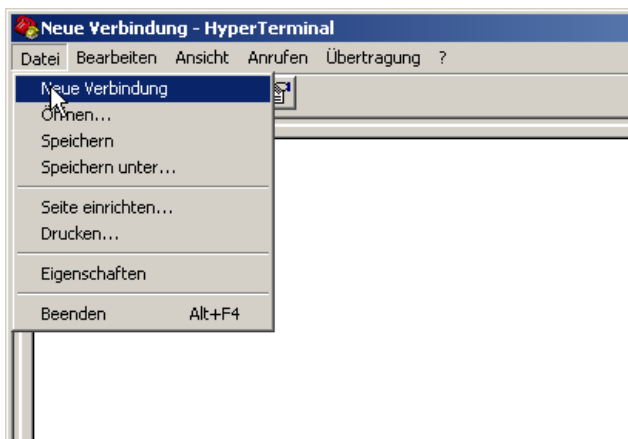
Das Vorgehen ist analog dem hier beschriebenen auf Basis vom Windows Hyperterminal.

Voraussetzungen:

Die MVC-Steuerung ist entweder an der RS232 oder über USB verbunden, für USB ist der entsprechende Konvertertreiber installiert. MVC ist eingeschaltet.

Starten sie ihr Terminalprogramm

Eröffnen sie eine neue Verbindung (falls sie Hyperterminal zum ersten Mal ausführen, werden sie direkt zum nächsten Schritt geleitet).



Geben sie der Verbindung einen beliebigen Namen, wählen sie das Com-Port aus, an welchem die Steuerung angeschlossen ist, und stellen sie die Anschlusseinstellungen wie gezeigt ein. Bestätigen sie jeweils mit ok.

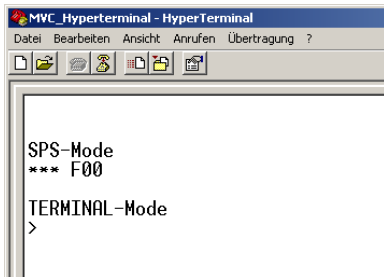
Die MVC-Steuerung wird nun verbunden, der Übertragungsbildschirm ist jedoch noch leer.

## Erste Befehle

Betätigen sie die Taste [Esc] => dieses Kommando schaltet die Steuerung zwischen dem SPS – und Terminalmodus hin und her.

Die Steuerung quittiert dies mit den jeweiligen Anzeigen:

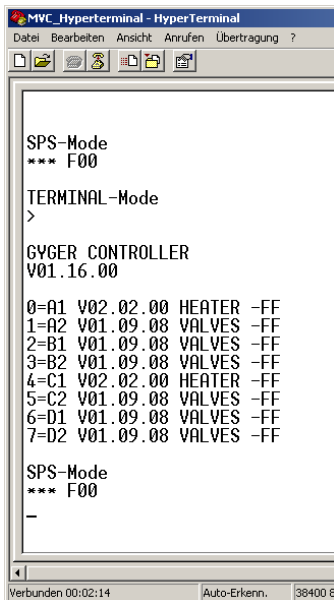
Beim Umschalten in den SPS-Mode gibt die Steuerung einen kleinen Bericht aus, welcher den Steuerungstyp, die installierte Firmware, sowie den eingestellten Funktionsmodus (hier F00) ausgibt.



Wenn sie die Anzeige „Terminal-Mode“ mit dem Prompt-Zeichen sehen, ist die Steuerung zum Datenempfang bereit.

Geben sie nun [ctrl] + [r] ein (Drücken der ctrl-Taste zusammen mit der r-Taste).

=>Die Steuerung löst eine Reinitialisierung aus, und gibt einen Bericht über die verfügbaren Modul-Adressen und ihren Funktionen. Zusätzlich sehen sie hier auch alle Firmware-Versionen. Die Steuerung schaltet mit diesem Befehl auch in den SPS-Mode und zeigt den SPS-Mode an.



Schalten sie wieder mit [esc] zurück in den Terminal-Modus.

Nun können sie durch direkte Eingaben via Tastatur arbeiten mit dem System.

Sie können zum Beispiel die Programmierbeispiele von Kapitel 0

**Befehlssatz** Funktionstyp „Heizung“

Wählen sie vor der Ausführung der Befehle eine Moduladresse mit Funktionstyp „Heizung“ an.



### 2.2.3 Steuerbefehle

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
Regler einschalten (Heizung Ein)	X		UND-Verknüpfung mit externem Hardware-Enable-Signal d.h. die Heizung wird erst aktiviert, wenn Hardware-Enable UND Software-Befehl aktiv sind.	A
Regler ausschalten (Heizung Aus)	S			A

#### Power-on-Verhalten:

Das Verhalten ist abhängig vom gesetzten SPS-Funktionsmode auf dem Mastermodul:

- F00 (Standard-Modus): Heizung ist bei Power-on immer deaktiviert.
- F01 (letzter Zustand wiederherstellen): Heizung ist bei Power-on im selben Zustand, wie er bei Power-off war.

### 2.2.4 Setzen von Parameterwerten

Eingabeformat: [Wert][Befehlszeichen]

z.B. „400T“ übermittelt eine Soll-Temperatur von 40.0°C an die angewählte Heizung.

Funktion	Befehls-zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
SOLL-Temperatur eingeben	T	1-1100 (=0.1-110°C)	Eingabe in Zehntel Grad	B
Proportionalteil Kp eingeben	A	1-10000	Dimensionslose Zahl * Default-Wert: 400	B
Integralteil Ki eingeben	B	1-1000	Dimensionslose Zahl * Default-Wert: 127	B
Differentialteil Kd eingeben	C	1-10000	Dimensionslose Zahl * Default-Wert: 616	B
Obere Abweichung für Ready-Signal eingeben	H	0-500 (=0-50°C)	Eingabe in Zehntel Grad Default-Wert: 20 (2.0°C)	B
Untere Abweichung für Ready-Signal eingeben	L	0-500 (=0-50°C)	Eingabe in Zehntel Grad Default-Wert: 20 (2.0°C)	B

Diese Eingaben werden nichtflüchtig gespeichert im EEPROM.

\*Achtung! Verstellung der PID-Regelparameter wird nicht empfohlen. Die Default-Werte sind optimal für die Gyger-Ventilheizung.

### 2.2.5 Auslesen von Parameterwerten

Eingabeformat: [Befehlszeichen]

z.B. „i“ fragt die momentane Ist-Temperatur der angewählten Heizung ab.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
SOLL-Temperatur auslesen [0.1°C]	t	1-1000 (=0.1-100°C)	Ausgabe in Zehntel Grad gültiger Bereich: 1 bis 1000 (=0.1 bis 100°C)	C
Regler-Status auslesen	q		Bit0: Status des Reglers (0=AUS; 1=EIN) Bit1: Externer Enable-Eingang (0=Disabled) Bit2: Status des READY-Ausgangssignals (1=READY)	C
IST-Temperatur auslesen [0.1°C]	i	0-1000 (=0-100°C)	Ausgabe in Zehntel Grad Wert 0 wird interpretiert als Drahtbruch oder Sensor nicht vorhanden = Heizfolienausgänge werden deaktiviert aus Sicherheitsgründen, ungeachtet vom Schaltzustand oder Status.	C
Proportionalteil Kp auslesen	a	1-10000	Dimensionslose Zahl	C
Integralteil Ki auslesen	b	1-1000	Dimensionslose Zahl	C
Differentialteil Kd auslesen	c	1-10000	Dimensionslose Zahl	C
Obere Abweichung für Ready-Signal auslesen [0.1°C]	h	0-500 (=0-50°C)	Ausgabe in Zehntel Grad	C
Untere Abweichung für Ready-Signal auslesen [0.1°C]	l	0-500 (=0-50°C)	Ausgabe in Zehntel Grad	C

durcharbeiten.

## 3 Befehlssatz

### 3.1 Befehlslaufzeiten

Die Befehlslaufzeiten für das Arbeiten über die serielle Schnittstelle bewegen sich im Millisekundenbereich. Für die meisten Befehle gilt hier ein Bereich von 2-15ms. Die Zeit gilt ab Empfang des letzten Zeichens bis die Rückgabe des Prompt-Zeichens erfolgt, d.h. bis die Steuerung wieder Empfangsbereit für einen neuen Befehl ist.

Die Schnittstelle sollte darum bei zeitkritischen Anwendungen nur zum Parametrieren verwendet werden, die Auslösung von Dosierschüssen sollte dann über die externen Triggereingänge erfolgen, hier können praktisch in Echtzeit Schussfolgen ausgelöst werden.

Brauchen sie zu den Timings noch genauere Auskunft, kontaktieren sie uns.

### 3.2 Globale Befehle

Diese Befehle können ausgeführt werden, ungeachtet welche Moduladresse, Modultyp, etc. gerade aktiv ist.

Eingabeformat: [Befehlszeichen]

z.B. „ESC“ (Druck der Taste Escape) schaltet die Steuerung zwischen Terminal- und SPS-Mode um.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
Umschalten Terminal-Mode, SPS-Mode	ESC		nur im Terminal-Mode kann mit der Steuerung kommuniziert werden.  Es ist normalerweise nicht nötig, diesen Modus zu verlassen, das Umschalten in den SPS-Modus wird für spezielle Funktionen verwendet (wie die Parametersatzwahl über externe Eingänge).	A
Modul-Adresse wählen	*	0-8	Moduladresse kann jederzeit umgewählt werden	B
Momentan angewählte Moduladresse abfragen	=		gibt die gerade angewählte Moduladresse und Modultyp aus.	C
Neuinitialisierung des Systems, Funktionstyp und Version aller Module anzeigen.	CTRL[R]		Nach Ausgabe der Information wird die Steuerung automatisch in den SPS-Mode geschaltet.	C

### 3.3 Befehlssatz Funktionstyp „Master“

Wählen sie vor der Ausführung dieser Befehle die Moduladresse 8 an (Mastermodul).

#### 3.3.1 Globale Dosierbefehle

Mit diesen Befehlen können direkt auf dem Mastermodul Dosierbefehle ausgelöst werden. Die Dosierbefehle werden dann auf allen installierten Ventilmodulen gleichzeitig ausgelöst.

Eingabeformat: [Befehlszeichen]

z.B. „V“ löst einen Einzelschuss aller angeschlossenen Ventile aus, mit Berücksichtigung des jeweils aktiven Parametersatzes pro Ventil.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
Einzelschuss alle Ventile	V		Verwendet Ventilparameter: A, B, C, D	A
Schuss-Serie alle Ventile	Q		kann mit Stopp-Befehl vorzeitig beendet werden Verwendet Ventilparameter: A, B, C, D, G	A
Endlos Schuss-Serie-Mode alle Ventile	U		läuft endlos bis zum Stopp-Befehl Verwendet Ventilparameter: A, B, C, D	A
Endlos Schuss-Serie-Mode alle Ventile, Hardware-Enable	L		läuft endlos bis zum Stopp-Befehl Verwendet Ventilparameter A, B, C, D	A
Alle Ventile stoppen	S		wird verwendet zum Beenden der Modi Q, U, L	A

Für Detailerklärungen siehe die gleichnamigen Kommandos beim Befehlssatz Funktionstyp „Ventil“.

### 3.3.2 Setzen von Parameterwerten

Eingabeformat: [Wert][Befehlszeichen]

z.B. „01F“ wählt die SPS-Funktionsnummer 01 an.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
SPS-Funktionsnummer eingeben	F	00-01	<p>00=Standard-Modus Aktivierte Trigger-Modi oder Heizsysteme sind bei jedem Power-on Vorgang wieder deaktiviert. Es wird für jedes Ventil der erste Parametersatz aktiviert bei Power-on.</p> <p>01=Letzten Zustand wiederherstellen Steuerung geht bei Power-on in den selben Zustand wie dies bei Power-Off der Fall war. d.h. aktivierte Trigger-Modi, Heizungen, und geladene Parametersets werden wieder aktiviert.</p>	B
Parametersatzwahl-Art selektieren	G	0-2	<p>0= Parametersatzwahl über Software / RS232</p> <p>1 = Wahl von 2 Parametersätzen pro Modul (A, B, C, D) über den externen Hardwareeingang E5 E5 low = Parametersatz 0n E5 high = Parametersatz 1n</p> <p>2 = Wahl von 4 Parametersätzen pro 2 Module (AB, CD). Jeweils E5 von zwei Modulen wird für die Parametersatzwahl verwendet, gültig für beide Module A-E5 low, B-E5 low = Parametersatz 0n A-E5 low, B-E5 high = Parametersatz 1n A-E5 high, B-E5 low = Parametersatz 2n A-E5 high, B-E5 high = Parametersatz 3n (Analog für C und D-Modul)</p> <p><b>Bei Pegeländerung dauert die Umschaltung in der Elektronik max. 0.15s! In dieser Zeit werden keine Schüsse ausgelöst.</b></p> <p>Die Parametersatzwahl über externe Eingänge ist nur aktiv wenn: -SPS-Funktionsnummer = F01 -Steuerung im SPS-Modus (Kommando esc bewirkt Umschaltung)</p>	B
Baud Rate selektieren	%	0-5	0=9600, 1=19200, 2=38400, 3=57600, 4=115200, 5=230400	B

Diese Eingaben werden nichtflüchtig gespeichert im EEPROM.

### 3.3.3 Auslesen von Parameterwerten

Eingabeformat: [Befehlszeichen]

z.B. „g“ liest die im Controller gespeicherte Parametersatzwahl-Art aus

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
Parametersatzwahl-Art auslesen	g	0-1		C

=>Die eingestellte Baud-Rate kann nicht ausgelesen werden.

=>Die eingestellte SPS-Funktionsnummer kann indirekt ausgelesen werden indem in den SPS-Modus gewechselt wird (Kommando esc). Bei der Umschaltung zeigt die MVC unter anderem diese Information an (siehe Kapitel 0 Abschnitt „Erste Befehle“)

### 3.4 Befehlssatz Funktionstyp „Ventil“

Wählen sie vor der Ausführung der Befehle eine Moduladresse mit Funktionstyp „Ventil“ an.

#### 3.4.1 Schusskontrolle über die externen Hardware-Triggereingänge

Nach Aktivierung des gewünschten Auslösemodus erfolgt die Schussauslösung praktisch in Echtzeit über die Triggereingänge. Die Verzögerungen sind lediglich im Bereich von wenigen Mikrosekunden.



Während ein ext. Auslösemodus aktiv ist, können keine Parameter geändert werden. Nur die Befehle „S“ (Stopp des externen Auslösemodus) und „esc“ (Umschaltung Terminal- SPS-Modus) sind anwendbar.

Eingabeformat: [Befehlszeichen]

z.B. „X“ aktiviert die Triggereingänge für Auslösung von Einzelschüssen auf Ventil 1 und 2.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
Einzelschuss-Mode V1/V2 EIN, Auslösung über ext. Hardware-Trigger	X		gilt für V1 und V2  Mit einer positiven Flanke am entsprechenden Signaleingang wird ein Einzelschuss des zugehörigen Ventils erzeugt. z.B. Eingang X2.1 für Ventil 1. Verwendet Ventilparameter: A, B, C, D  Parameter C wird insofern verwendet, dass kein neuer Taktimpuls verarbeitet wird, bevor die Zeit C abgelaufen ist.	A
Einzelschuss-Mode V1/V2 einschalten, Kontrolle über ext. Hardware-Puls	T		gilt für V1 und V2 Pulsdauer=Öffnungszeit  Solange der zugehörige Signaleingang High ist, wird das Ventil geöffnet. =>Positive Flanke: öffnet das Ventil. Negative Flanke: schliesst das Ventil wieder. Verwendet Ventilparameter: A, D	A
Schuss-Serie-Mode V1/V2 EIN, Auslösung über ext. HW-Trigger	P		gilt für V1 und V2  Mit einer positiven Flanke am entsprechenden Signaleingang wird eine Schuss-Serie des entsprechenden Ventils erzeugt.  Verwendet Ventilparameter: A, B, C, D, G  Achtung: Wenn G=0=unendlich gesetzt ist, wird keine Dosierung ausgelöst.	A
Endlos Schuss-Serie-Mode V1/V2 EIN, Kontrolle über HW-Enable (Pegelgesteuert).	L		gilt für V1 und V2  Solange der entsprechende Signaleingang High ist, dosiert das zugehörige Ventil mit der eingestellten Frequenz und Öffnungszeit.  Verwendet Ventilparameter: A, B, C, D	A
Stopp Trigger-Mode	S		Beendet einen aktiven Trigger-Mode (X, T, P, L)	A

#### • Power-on-Verhalten

Das Verhalten ist abhängig vom gesetzten SPS-Funktionsmode auf dem Mastermodul:

- F00 (Standard-Modus): Bei Power-on sind keine Hardware-Trigger-Modi aktiv.
- F01 (Zuletzt gewählter Modus wiederherstellen): Wenn ein Trigger-Mode bei Power-off aktiv war, wird dieser automatisch wieder aktiviert nach Power-on.



### 3.4.2 Schussauslösung über Software

Die Befehlsverzögerung nach Empfang des Befehlszeichens bis zur ersten Schussauslösung beträgt ca. 2ms. Für zeitkritische Anwendungen verwenden sie die Funktionen zur Schussauslösung mit Hardware-Trigger.

Eingabeformat: [Befehlszeichen]

z.B. „Y“ löst einen Einzelschuss von Ventil1 aus, wenn Moduladresse 0 gewählt ist.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
Einzelschuss V1	Y		Statusabfrage über Befehl „q“  Verwendet Ventilparameter: A, B, C*, D *=C wird nur für die Statusabfrage verwendet. Status q ist nur korrekt, wenn C>B	A
Einzelschuss V2	Z		Statusabfrage über Befehl „q“  Verwendet Ventilparameter: A, B, C*, D *=C wird nur für die Statusabfrage verwendet. Status q ist nur korrekt, wenn C>B	A
Einzelschuss V1/V2 gemeinsam	V		Statusabfrage über Befehl „q“  Verwendet Ventilparameter: A, B, C*, D *=C wird nur für die Statusabfrage verwendet. Status q ist nur korrekt, wenn C>B	A
Schuss-Serie V1	Q		kann mit Stopp-Befehl jederzeit beendet werden Statusabfrage über Befehl „q“  Verwendet Ventilparameter: A, B, C, D, G	A
Schuss-Serie V2	R		kann mit Stopp-Befehl jederzeit beendet werden Statusabfrage über Befehl „q“  Verwendet Ventilparameter: A, B, C, D, G	A
Endlos Schuss-Serie-Mode V1/V2 EIN	U		gilt für V1 bis V4 (beide Modulhälften) läuft endlos bis zum Stopp-Befehl  Verwendet Ventilparameter: A, B, C, D	A
Stopp Schuss-Serie	S		Stoppen eines aktiven Schuss-Serie-Modes, (Q, R, U)	A

=>wird unmittelbar nach einem Befehl Y oder Z (Einzelschuss), eine Schuss-Serie gestartet mit Q oder R, lässt sich diese erst starten, nachdem Parameter C (Taktzeit) abgelaufen ist vom vorhergehenden Einzelschuss.

### 3.4.3 Setzen von Parameterwerten

#### Parametersätze

Pro Ventil können 4 komplette Parametersets direkt im EEPROM des Mikrocontrollers hinterlegt werden. Jeder Parametersatz enthält die Parameter A, B, C, D und G. Durch den Parameter n bzw N kann der gewünschte Parametersatz aktiviert/angewählt oder beschrieben werden.

Eine Aktivierung ist gültig, bis ein anderer Parametersatz gewählt wird.

0n= 1. Parametersatz von Ventil1	4n= 1. Parametersatz von Ventil2
1n= 2. Parametersatz von Ventil1	5n= 2. Parametersatz von Ventil2
2n= 3. Parametersatz von Ventil1	6n= 3. Parametersatz von Ventil2
3n= 4. Parametersatz von Ventil1	7n= 4. Parametersatz von Ventil2

Die Aktivierung erfolgt Paarweise, d.h. wenn 0n gewählt wird, wird automatisch 4n auch aktiviert (n+4). Bei Schussauslösung werden dann für beide Ventile die Parameter des ersten Parametersatzes verwendet (0n, 4n)

Ebenso wenn 5n aktiviert wird, wird automatisch 2n auch aktiviert (n-4).



Die Hinterlegung von mehreren Parametersätzen macht vor allem Sinn, wenn die externe Parametersatzwahl (Parameter G auf Masteradresse) verwendet wird.

#### Power-on-Verhalten

Das Verhalten ist abhängig vom gesetzten SPS-Funktionsmode auf dem Mastermodul:

- F00 (Standard-Modus): Bei Power-on wird immer das erste Parametersatz-Paar automatisch geladen (0n, 4n).
- F01 (letzter Zustand wiederherstellen): Es wird das Parametersatz-Paar geladen, das bei Power-off aktiv war.

Eingabeformat: [Wert][Befehlszeichen]

z.B. „400A“ übermittelt eine Peakzeit von 400µs auf den aktiven Parametersatz im Arbeitsspeicher des Mikrocontrollers.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
PEAK Time eingeben	A	10-65'535	Eingabe in Mikrosekunden, wenn Open time < Peak time ist, wird die Peakzeit beschnitten, d.h. Peak time = Open time. Praktikabler Wert: SMLD 300: 150µs SMLD 300G: 400µs	B
OPEN Time eingeben	B	10-9'999'999	Eingabe in Mikrosekunden Praktikable Werte: 400 – 9'999'999µs	B
CYCLE Time eingeben	C	10-9'999'999	Eingabe in Mikrosekunden Schussfrequenz=(1/CYCLE Time) Bedingung: CYCLE Time>OPEN Time  Ein neuer Schuss kann erst ausgelöst werden, wenn die Cycle-time abgelaufen ist. Im Zusammenhang mit den Befehlen Q, R und U dient die Cycletime zur Vorgabe der Dosierfrequenz.	B
PEAK Strom eingeben	D	0-15	der Peakstrom beträgt ungefähr: $I_p=450mA + (D \cdot 50mA)$ Der Peakstrom kann zwar pro Ventil eingegeben sein, muss aber pro Modulhälfte (V1,V2) gleich sein! Praktikabler Wert: 11 = 1000mA	B
Schusszahl eingeben	G	0-65535	0= unendlich. Wirksam mit den Schuss-Serie-Befehlen Q und R, sowie den Hardware-Triggerbefehlen	B
Parametersatz aktivieren / wählen	n	0-7	Ladet den entsprechenden Parametersatz aus dem EEPROM in den Arbeitsspeicher des Mikrocontrollers. Ebenso den Parametersatz n+4, bzw n-4 (paarweise Aktivierung für beide Ventile).	C
Parameter-Satz im EEPROM abspeichern	N	0-7	0-3 = Parametersätze für Ventil 1 4-7 = Parametersätze für Ventil 2	B
Schusszähler Ventil 1 nullen	I		Nullsetzung des Schusszählers Ventil 1 (siehe Näheres im Abschnitt „Auslesen von Parameterwerten“)	B
Schusszähler Ventil 2 nullen	J		Nullsetzung des Schusszählers Ventil 2	B



Wichtig: Die Eingaben auf diesen Parametern werden nicht automatisch nichtflüchtig gespeichert, sie werden nur in den Arbeitsspeicher des Mikrocontrollers übertragen.  
Verwenden sie darum immer den Befehl „N“ um Werte nichtflüchtig im EEPROM abzulegen.

Dieses Konzept hat den Vorteil, wenn sie z.B. mit einem Ventil dosieren, und über die Schnittstelle für jeden Schuss eine andere Öffnungszeit definieren wollen, dann wird das EEPROM nicht dauernd neu beschrieben (begrenzte Anzahl Schreibzyklen).



Achtung: Bei falschen Eingaben können die Ventilsulen verbrannt werden. Konsultieren sie unsere Anleitung zur MVC-Software für weitere Hinweise!

### 3.4.4 Auslesen von Parameterwerten

Eingabeformat: [Befehlszeichen]

z.B. „a“ liest die im Controller gespeicherte Peakzeit aus für den angewählten Parametersatz.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
PEAK Time auslesen	a	10-65'535	Ausgabe in Mikrosekunden	C
OPEN Time auslesen	b	10-9'999'999	Ausgabe in Mikrosekunden	C
CYCLE Time auslesen	c	10-9'999'999	Ausgabe in Mikrosekunden	C
PEAK Strom auslesen	d	0-15	der Peakstrom beträgt ungefähr: $I_p=450\text{mA} + (D \cdot 50\text{mA})$	C
Schusszahl auslesen	g	0-65535		C
Momentan aktiven Parametersatz abfragen	p	0-7		C
Ventil-Status auslesen	q		Dient zur Ventil-Statusabfrage in den Modi Y, Z, V, Q und R Ausgabe von 2 Statusbits als ASCII-codierte Dezimalzahl Binäre Gewichtung: Bit0=1(dezimal 1): V1=aktiv, Bit4=1(dezimal 16): V2=aktiv Beispiel: Bit0 und Bit4 gesetzt ergibt >.q00000017 (16+1)	C
Schusszähler Ventil 1 auslesen	y	0-9'999'999	Die total abgegebenen Schüsse werden pro Ventil automatisch gezählt. Der Zählerstand kann ausgelesen werden mit diesem Befehl. Der Schusszähler ist flüchtig, bei Power-on ist er also jedes Mal wieder auf 0 gesetzt.	C
Schusszähler Ventil 2 auslesen	z	0-9'999'999	dito für Ventil 2	C
Schusstotal Ventil1 (Bits 31-24) auslesen	u	0-255		C
Schusstotal Ventil1 (Bits 23-0) auslesen	v	0-16777215	Schusstotal V1 = (Wert [u] * 16'777'216) + (Wert [v]) Wertebereich V1 ist 0 bis 4'294'967'295 (32-Bit) Vor Abfrage Befehl I ausführen (Schusszähler nullen und speichern)	C
Schusstotal Ventil2 (Bits 31-24) auslesen	w	0-255		C
Schusstotal Ventil2 (Bits 23-0) auslesen	x	0-16777215	Schusstotal V2 = (Wert [w] * 16'777'216) + (Wert [x]) Wertebereich V2 ist 0 bis 4'294'967'295 (32-Bit) Vor Abfrage Befehl J ausführen (Schusszähler nullen und speichern)	C

### 3.5 Befehlssatz Funktionstyp „Heizung“

Wählen sie vor der Ausführung der Befehle eine Moduladresse mit Funktionstyp „Heizung“ an.

#### 3.5.1 Steuerbefehle

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
Regler einschalten (Heizung Ein)	X		UND-Verknüpfung mit externem Hardware-Enable-Signal d.h. die Heizung wird erst aktiviert, wenn Hardware-Enable UND Software-Befehl aktiv sind.	A
Regler ausschalten (Heizung Aus)	S			A

#### Power-on-Verhalten:

Das Verhalten ist abhängig vom gesetzten SPS-Funktionsmode auf dem Mastermodul:

- F00 (Standard-Modus): Heizung ist bei Power-on immer deaktiviert.
- F01 (letzter Zustand wiederherstellen): Heizung ist bei Power-on im selben Zustand, wie er bei Power-off war.

#### 3.5.2 Setzen von Parameterwerten

Eingabeformat: [Wert][Befehlszeichen]

z.B. „400T“ übermittelt eine Soll-Temperatur von 40.0°C an die angewählte Heizung.

Funktion	Befehls-zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
SOLL-Temperatur eingeben	T	1-1100 (=0.1-110°C)	Eingabe in Zehntel Grad	B
Proportionalteil Kp eingeben	A	1-10000	Dimensionslose Zahl * Default-Wert: 400	B
Integralteil Ki eingeben	B	1-1000	Dimensionslose Zahl * Default-Wert: 127	B
Differentialteil Kd eingeben	C	1-10000	Dimensionslose Zahl * Default-Wert: 616	B
Obere Abweichung für Ready-Signal eingeben	H	0-500 (=0-50°C)	Eingabe in Zehntel Grad Default-Wert: 20 (2.0°C)	B
Untere Abweichung für Ready-Signal eingeben	L	0-500 (=0-50°C)	Eingabe in Zehntel Grad Default-Wert: 20 (2.0°C)	B

Diese Eingaben werden nichtflüchtig gespeichert im EEPROM.

\*Achtung! Verstellung der PID-Regelparameter wird nicht empfohlen. Die Default-Werte sind optimal für die Gyger-Ventilheizung.

### 3.5.3 Auslesen von Parameterwerten

Eingabeformat: [Befehlszeichen]

z.B. „i“ fragt die momentane Ist-Temperatur der angewählten Heizung ab.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich	Bemerkung	Befehls-Typ
SOLL-Temperatur auslesen [0.1°C]	t	1-1000 (=0.1-100°C)	Ausgabe in Zehntel Grad gültiger Bereich: 1 bis 1000 (=0.1 bis 100°C)	C
Regler-Status auslesen	q		Bit0: Status des Reglers (0=AUS; 1=EIN) Bit1: Externer Enable-Eingang (0=Disabled) Bit2: Status des READY-Ausgangssignals (1=READY)	C
IST-Temperatur auslesen [0.1°C]	i	0-1000 (=0-100°C)	Ausgabe in Zehntel Grad Wert 0 wird interpretiert als Drahtbruch oder Sensor nicht vorhanden = Heizfolienausgänge werden deaktiviert aus Sicherheitsgründen, ungeachtet vom Schaltzustand oder Status.	C
Proportionalteil Kp auslesen	a	1-10000	Dimensionslose Zahl	C
Integralteil Ki auslesen	b	1-1000	Dimensionslose Zahl	C
Differentialteil Kd auslesen	c	1-10000	Dimensionslose Zahl	C
Obere Abweichung für Ready-Signal auslesen [0.1°C]	h	0-500 (=0-50°C)	Ausgabe in Zehntel Grad	C
Untere Abweichung für Ready-Signal auslesen [0.1°C]	l	0-500 (=0-50°C)	Ausgabe in Zehntel Grad	C

## 4 Programmierbeispiele

### Grundsatz

1. Wahl der Moduladresse (falls nicht schon aktiv)
2. Ausführung des Befehls.

Eine einmal gewählte Moduladresse bleibt aktiv, bis eine andere gewählt wird.

### 4.1 Heizungsmodul

#### Konfiguration der Steuerung

Beispielkonfiguration 3 aus der Tabelle bei Kapitel 2.1.1

MVC-Steuerung mit Modul A + C bestückt als Mischmodule für Ventil/Heiz-Kombination.

Modul	Funktionstyp	Adresse
A	Heizung	0
	Ventil	1
C	Heizung	4
	Ventil	5
Master	Master	8

Den installierten Heizungen auf Modul A und C wird eine Solltemperatur übermittelt und die Heizungen werden aktiviert. Dann wird der Status der Heizungen periodisch abgefragt bis zur Erreichung der Solltemperatur.

Funktion	Befehls-Zeichen (ASCII)	Wertebereich
<b>0*</b>	<b>0*&gt;</b>	<b>Adresse 0 wurde gewählt (Heizung auf Modul A)</b>
500T	500T>	Solltemperatur 50°C einstellen (Wert in Zehntelgrad)
X	X>	Heizung einschalten
i	.i00000233>	Ist-Temperatur auslesen (Wert wird in Zehntelgrad mit führenden Nullen gesendet = 23.3°C)
q	.q00000003>	Status wird zurückgelesen (Heizung eingeschalten aber noch nicht innerhalb der Regeltoleranz)
<b>4*</b>	<b>4*&gt;</b>	<b>Adresse 4 wurde gewählt (Heizung auf Modul C)</b>
430T	430T>	Solltemperatur 43°C einstellen (Wert in Zehntelgrad)
X	X>	Heizung einschalten
i	.i00000253>	Ist-Temperatur auslesen (Wert wird in Zehntelgrad mit führenden Nullen gesendet = 25.3°C)
q	.q00000003>	Status zurückgelesen
<b>0*</b>	<b>0*&gt;</b>	<b>Heizung Modul A wählen</b>
i	.i00000384>	Ist-Temperatur auslesen = 38.4°C
q	.q00000003>	Status zurücklesen
<b>4*</b>	<b>4*&gt;</b>	<b>Heizung Modul C wählen</b>
i	.i00000406>	Ist-Temperatur auslesen = 40.6°C
q	.q00000003>	Status zurücklesen
<b>0*</b>	<b>0*&gt;</b>	<b>Heizung Modul A wählen</b>
i	.i00000496>	Ist-Temperatur auslesen = 49.6°C
q	.q00000007>	Status zurücklesen (7=Heizung ein und innerhalb der Regeltoleranz)
<b>4*</b>	<b>4*&gt;</b>	<b>Heizung Modul C wählen</b>
i	.i00000433>	Ist-Temperatur auslesen = 43.3°C
q	.q00000007>	Status zurücklesen (7=Heizung ein und innerhalb der Regeltoleranz)

## 4.2 Ventilmodul: Dosieren mit nichtflüchtig gespeicherten Werten

### Konfiguration der Steuerung

Beispielkonfiguration 1 aus der Tabelle bei Kapitel 2.1.1

Vollbestückte MVC-Steuerung mit Modul A, B, C, D ausgelegt als Ventiltreiber.

Modul	Funktionstyp	Adresse
A	Ventil	0
	Ventil	1
B	Ventil	2
	Ventil	3
C	Ventil	4
	Ventil	5
D	Ventil	6
	Ventil	7
Master	Master	8

Dem Modul A = 4 Ventile, werden Werte übermittelt (für alle Ventile dieselben), dann werden zum Test Schussfolgen ausgelöst bei jedem Ventil, zum Schluss werden die Module in den externen Trigger-Modus versetzt, zum Triggern von Schuss-Serien über die Hardware-Eingänge.

Gesendete Befehle	Empfangene Zeichen	Bemerkung
<b>0*</b>	<b>0*&gt;</b>	<b>Adresse 0 wird gewählt</b>
0n	0.n>	Aktivierung des 1. Parameterpaars (0n, 4n)
400A	400A>	Peak-Zeit von 400µs wird übermittelt in Arbeitsspeicher
1000B	1000B>	Öffnungszeit von 1000µs wird übermittelt in Arbeitsspeicher
100000C	100000C>	Cycle-Time von 100'000µs = 0.1s wird übermittelt
10G	10G>	Schusszahl 10 wird übermittelt
0N	0N>	Die Werte werden auf Parametersatz 1 nichtflüchtig gespeichert für Ventil A1
4N	4N>	Die Werte werden auf Parametersatz 1 nichtflüchtig gespeichert für Ventil A2
Q	Q>	Auslösung Schussserie Ventil A1= 10 Schuss im Abstand von 0.1s
R	R>	Auslösung Schussserie Ventil A2= 10 Schuss im Abstand von 0.1s
<b>1*</b>	<b>1*&gt;</b>	<b>Adresse 1 wird gewählt (zweite Modulhälfte von Modul A).</b>
0n	0.n>	
400A	400A>	Wiederholung des Vorgangs wie oben
1000B	1000B>	
100000C	100000C>	
10G	10G>	
0N	0N>	
4N	4N>	
Q	Q>	Auslösung Schussserie Ventil A3= 10 Schuss im Abstand von 0.1s
R	R>	Auslösung Schussserie Ventil A4= 10 Schuss im Abstand von 0.1s
P	P>	Das ganze Modul A wird in den ext. Trigger-Mode versetzt für Auslösung Schuss-Serien über die Hardware-Eingänge.



### 4.3 Ventilmodul: Parameterwerte abfragen

#### Konfiguration der Steuerung:

Beispielkonfiguration 1 aus der Tabelle bei Kapitel 2.1.1

Vollbestückte MVC-Steuerung mit Modul A, B, C, D ausgelegt als Ventiltreiber.

Modul	Funktionstyp	Adresse
A	Ventil	0
	Ventil	1
B	Ventil	2
	Ventil	3
C	Ventil	4
	Ventil	5
D	Ventil	6
	Ventil	7
Master	Master	8

Die Parameterwerte von Modul A = 4 Ventile sollen abgefragt werden. Dabei soll jeweils der 1. Parametersatz pro Ventil abgefragt werden.

Gesendete Befehle	Empfangene Zeichen	Bemerkung
<b>0*</b>	<b>0*&gt;</b>	<b>Adresse 0 wird gewählt</b>
0n	0.n>	Anwahl 1. Parametersatz von Ventil A1
a	.a00000400>	Abfrage Peakzeit
b	.b00001000>	Abfrage Öffnungszeit
c	.c00100000>	Abfrage Cycle-Time
g	.g00000010>	Abfrage Schusszahl
4n	4.n>	Anwahl 1. Parametersatz von Ventil A2
a	.a00000400>	Abfrage Peakzeit
b	.b00001000>	Abfrage Öffnungszeit
c	.c00100000>	Abfrage Cycle-Time
g	.g00000010>	Abfrage Schusszahl
<b>1*</b>	<b>1*&gt;</b>	<b>Adresse 1 wird gewählt (zweite Modulhälfte von Modul A).</b>
0n	0.n>	Anwahl 1. Parametersatz von Ventil A3
a	.a00000400>	Abfrage Peakzeit
b	.b00001000>	Abfrage Öffnungszeit
c	.c00100000>	Abfrage Cycle-Time
g	.g00000010>	Abfrage Schusszahl
4n	4.n>	Anwahl 1. Parametersatz von Ventil A4
a	.a00000400>	Abfrage Peakzeit
b	.b00001000>	Abfrage Öffnungszeit
c	.c00100000>	Abfrage Cycle-Time
g	.g00000010>	Abfrage Schusszahl

## 4.4 Ventilmodul: Dosieren mit temporär gespeicherten Werten

### Konfiguration der Steuerung

Beispielkonfiguration 1 aus der Tabelle bei Kapitel 2.1.1

Vollbestückte MVC-Steuerung mit Modul A, B, C, D ausgelegt als Ventiltreiber.

Modul	Funktionstyp	Adresse
A	Ventil	0
	Ventil	1
B	Ventil	2
	Ventil	3
C	Ventil	4
	Ventil	5
D	Ventil	6
	Ventil	7
Master	Master	8

Für spezielle Zwecke (z.B. Schussauslösung über Software mit dauernd ändernden Parametern) ist es hilfreich, die Parameter nur temporär in den Arbeitsspeicher der Steuerung zu schreiben.

So werden nicht unnötig Speichervorgänge auf das EEPROM gemacht. Zu diesem Zweck wird einfach der Speicherbefehl weggelassen (z.B. 0N speichert den 1. Parametersatzes von Ventil 1 im EEPROM).

#### Der Arbeitsspeicher wird geleert wenn:

- Ein anderer Parametersatz geladen wird (0n, 4n)
- Eine andere Adresse selektiert wird (0\*, 8\*)
- Ein externer Auslösemodus aktiviert wird

#### 4.4.1 Einzelschüsse dosieren mit temporären Werten

Im Beispiel werden auf Ventil 1 und 2 je zwei Schüsse ausgelöst mit unterschiedlicher Öffnungszeit

Gesendete Befehle	Empfangene Zeichen	Bemerkung
0*	0*>	Adresse 0 wird gewählt
0n	0.n>	<b>Aktivierung des 1. Parameterpaars (0n, 4n)</b> =>Damit werden die nichtflüchtig gespeicherten Parameter in den Arbeitsspeicher geladen. Alle Parameter die nun nachfolgend gesetzt werden, überschreiben den entsprechenden geladenen Wert im Arbeitsspeicher.  Fokus für temporäre Schussauslösung liegt auf Ventil 1
2000B	2000B>	Neue Öffnungszeit wird in den Arbeitsspeicher übermittelt
Y	Y>	Auslösung Einzelschuss Ventil A1 mit temporärer Öffnungszeit
q	.q00000000>	Abfrage Status (optional)=> Bit 0 wieder auf 0 nachdem die Dosierung fertig ist.
3000B	3000B>	Neue Öffnungszeit wird in den Arbeitsspeicher übermittelt
Y	Y>	Auslösung Einzelschuss Ventil A1 mit temporärer Öffnungszeit
q	.q00000000>	Abfrage Status (optional)=> Bit 0 wieder auf 0 nachdem die Dosierung fertig ist.
4n	4.n>	<b>Aktivierung des 1. Parameterpaars (0n, 4n)</b> =>Damit werden die nichtflüchtig gespeicherten Parameter in den Arbeitsspeicher geladen. Alle Parameter die nun nachfolgend gesetzt werden, überschreiben den entsprechenden geladenen Wert im Arbeitsspeicher.  Fokus für temporäre Schussauslösung liegt auf Ventil 2
2000B	2000B>	Neue Öffnungszeit wird in den Arbeitsspeicher übermittelt
Z	Z>	Auslösung Einzelschuss Ventil A2 mit temporärer Öffnungszeit
q	.q00000000>	Abfrage Status (optional)=> Bit 0 wieder auf 0 nachdem die Dosierung fertig ist.
3000B	3000B>	Neue Öffnungszeit wird in den Arbeitsspeicher übermittelt
Z	Z>	Auslösung Einzelschuss Ventil A2 mit temporärer Öffnungszeit
q	.q00000000>	Abfrage Status (optional)=> Bit 0 wieder auf 0 nachdem die Dosierung fertig ist.

=>Bei Bedarf können noch weitere Parameter als die Öffnungszeit temporär in den Arbeitsspeicher geschrieben werden (z.B. Peakzeit, Peakstrom, etc)

#### 4.4.2 Schussserien dosieren mit temporären Werten

Im folgenden Beispiel wird eine Schussserie nacheinander bei 2 Ventilen gemacht mit Öffnungszeiten von 1000µs, Frequenz 100 Hz, 1000 Schuss = 10s totale Dosierdauer pro Ventil.

Gesendete Befehle	Empfangene Zeichen	Bemerkung
0*	0*>	Adresse 0 wird gewählt
0n	0.n>	<b>Aktivierung des 1. Parameterpaars (0n, 4n)</b> =>Damit werden die nichtflüchtig gespeicherten Parameter in den Arbeitsspeicher geladen. Alle Parameter die nun nachfolgend gesetzt werden, überschreiben den entsprechenden geladenen Wert im Arbeitsspeicher.  Fokus für temporäre Schussauslösung liegt auf Ventil 1
1000B	1000B>	Öffnungszeit von 1000µs wird übermittelt in Arbeitsspeicher
Y	Y>	Auslösung Einzelschuss Ventil A1 mit temporärer Öffnungszeit
10000C	10000C>	Taktzeit von 10'000µs wird übermittelt in Arbeitsspeicher = 100 Hz Dosierfrequenz
1000G	1000G	Schusszahl von 1000 wird übermittelt in Arbeitsspeicher
Q	Q>	Auslösung Schussfolge Ventil 1
q	.q00000001>	Abfrage Status (optional)=> Bit 0 auf 1 während der Dosierung = 10 Sekunden lang
q	.q00000000>	Abfrage Status (optional) => Bit 0 wieder auf 0 nachdem die Dosierung fertig ist.
4n	4.n>	<b>Aktivierung des 1. Parameterpaars (0n, 4n)</b> =>Damit werden die nichtflüchtig gespeicherten Parameter in den Arbeitsspeicher geladen. Alle Parameter die nun nachfolgend gesetzt werden, überschreiben den entsprechenden geladenen Wert im Arbeitsspeicher.  Fokus für temporäre Schussauslösung liegt auf Ventil 2
1000B	1000B>	Öffnungszeit von 1000µs wird übermittelt in Arbeitsspeicher
10000C	10000C>	Taktzeit von 10'000µs wird übermittelt in Arbeitsspeicher = 100 Hz Dosierfrequenz
1000G	1000G	Schusszahl von 1000 wird übermittelt in Arbeitsspeicher
R	R>	Auslösung Schussfolge Ventil 2
q	.q00000001>	Abfrage Status (optional)=> Bit 0 auf 1 während der Dosierung = 10 Sekunden lang
q	.q00000000>	Abfrage Status (optional) => Bit 0 wieder auf 0 nachdem die Dosierung fertig ist.

=>Bei Bedarf können noch weitere Parameter temporär in den Arbeitsspeicher geschrieben werden (z.B. Peakzeit, Peakstrom, etc)