

Dokumentation:

Kommandos für serielle Steuerung

Gültig für folgende Umformer Typen:

**SFU0051**

**SFU0151, SFU0152, SFU0154, SFU0156**

**SFU0200**

**SFU0200DV**

**SFU0300**

**SFU0303**

**SFU0303DV**

RS232 Interface Einstellung: 115kBd, 8data, no parity, 1stopbit

**SFU0102, SFU0202**

**SFU0302**

**SFU0303/2, SFU0200(Bd96), SFU0303(Bd 96)**

RS232 Interface Einstellung: 9600Bd, 8data, no parity, 1stopbit

---

### 1-Byte-Kommandos im HEX Format:

1.

---

Kommando:	<b>Start</b>		Für einen Dauerbetrieb der Spindel nach dem Kommando "Start" 0x24h, sicherstellen, dass permanent eine Kommunikation stattfindet.
Transmit:	<b>0x24</b>		Entweder durch permanentes Senden des Startkommandos oder durch kontinuierliches Auslesen des Statusworts.
Receive:	0xE4	Quittung	Andernfalls wird nach 4sek automatisch ein Spindel-Stop ausgelöst, um zu verhindern, dass die Spindel im Falle eines Kommunikationsverlusts mit dem Umformer unkontrolliert weiterläuft.
	0xll	Low Byte rpm	
	0xhh	High Byte rpm	

2.

---

Kommando:	<b>Stop</b>	
Transmit:	<b>0x25</b>	
Receive:	0xE5	Quittung
	0xll	Low Byte rpm
	0xhh	High Byte rpm

3.

---

Kommando:	<b>Sende Soll Drehzahl (*10)</b>	
Transmit:	<b>0x41</b>	
Receive:	0xC1	Quittung
	0xll	<b>Solldrehzahl</b> Low Byte
	0xhh	<b>Solldrehzahl</b> High Byte

der gesendete Wert kommt im Hex Format und muss mit 10 multipliziert werden

4.

---

Kommando:	<b>Sende aktuelle Drehzahl des Umformers (*10)</b>	
Transmit:	<b>0x42</b>	
Receive:	0xC2	Quittung
	0xll	<b>aktuelle Drehzahl des Umformers</b> Low Byte
	0xhh	<b>aktuelle Drehzahl des Umformers</b> High Byte

der gesendete Wert kommt im Hex Format und muss mit 10 multipliziert werden

5.

---

Kommando:	<b>Sende Drehzahl der Spindel (*10)</b>	
Transmit:	<b>0x43</b>	
Receive:	0xC3	Quittung
	0xll	<b>Drehzahl der Spindel</b> Low Byte
	0xhh	<b>Drehzahl der Spindel</b> High Byte

der gesendete Wert kommt im Hex Format und muss mit 10 multipliziert werden

Wenn ein Drehgeber angeschlossen und in der Kennlinie aktiviert ist, ist das die reale Drehzahl der Spindel, andernfalls ist es die Ausgabedrehzahl des Umformers.

6. \_\_\_\_\_ nur für DressViewLight Versionen **DV** \_\_\_\_\_

Kommando: **Nullung/Kalibrierung des DV\_Load Werts**

Transmit: **0x30**

Receive: 0xF0           Quittung  
          0xll           **( irrelevant )** Low Byte  
          0xhh           **( irrelevant )** High Byte

7. \_\_\_\_\_ nur für DressViewLight Versionen **DV** \_\_\_\_\_

Kommando: **Sende DV\_Load Wert (Wertebereich 0-1023)**

Transmit: **0x31**

Receive: 0xF1           Quittung  
          0xll           **DV\_Load** Low Byte  
          0xhh           **DV\_Load** High Byte

---

8.

---

 Kommando: **Sende Umformer Status**
Transmit: **0x60**
 Receive: 0xE0            Quittung  
           0xll            **Status Wort** Low Byte  
           0xhh            **Status Wort** High Byte
**Bits des Status Words:**


---

Bit	0	<b>reserviert</b>
Bit	1	Status <b>Start/Stop</b>
Bit	2	Status <b>Impulssperre aktiv</b>
Bit	3	Status <b>Fernsteuerung aktiv</b>
Bit	4	Status <b>IST-Drehzahl erreicht</b> (bei angeschlossener Feldplatte: Ist = Soll Drehzahl)
Bit	5	Status <b>Solldrehzahl erreicht</b>
Bit	6	Status <b>Spindel Stop</b>
Bit	7	Status <b>Unterspannung</b>
Bit	8	Status <b>Überspannung</b>
Bit	9	Status <b>Variolast erreicht</b>
Bit	10	Status <b>Error RS232 Interface</b>
Bit	11	Status <b>Spindel nicht bereit</b>
Bit	12	Status <b>Umformer nicht bereit</b>
Bit	13	Status <b>Überlast</b>
Bit	14	Status <b>Übertemperatur Umformer</b>
Bit	15	Status <b>Übertemperatur Spindel</b>

---

**A c h t u n g !**

Nach dem Kommando "Start" 0x24h, bitte sicherstellen, dass permanent eine Kommunikation stattfindet z.B.: durch kontinuierliches Auslesen des Status-Worts. Andernfalls wird nach 4sek automatisch ein Spindel-Stop ausgelöst, um zu verhindern, dass die Spindel im Falle eines Kommunikationsverlusts mit dem Umformer unkontrolliert weiterläuft.

### 3-Byte-Kommandos im HEX Format:

9.

---

 Kommando: **Stelle Solldrehzahl ein (= gewünschter Wert / 10)**

 Transmit: **0x01**  
           0xll           Low Byte **Solldrehzahl**  
           0xhh           High Byte **Solldrehzahl**

 Receive: 0xC1  
           0xll           Low Byte  
           0xhh           High Byte

Der gewünschte Wert muss durch 10 dividiert werden und anschließend zum Umformer gesendet werden

zum Beispiel, setze Drehzahl auf 20.000:  $20.000/10 = 2.000d = 07D0h \Rightarrow$  sende "01 D0 07"

10.

---

 Kommando: **Setze Datenpointer zum Lesen einer Daten Variable**

 Transmit: **0x0C**  
           0xll           Low Byte Pointer  
           0xhh           High Byte Pointer

 Receive: 0xCC  
           0xll           Low Byte Wert  
           0xhh           High Byte Wert

Mit Hilfe dieser Funktion kann jede Umformer Variable ausgelesen werden  
 Eine Liste der verfügbaren Adressen findet sich am Ende des Dokuments.

11.

---

 Kommando: **Stelle Drehrichtung auf Rechts** (von hinten auf die Spindel gesehen)

 Transmit: **0x0A**           **Drehrichtung auf Rechts** (wie beim Bohren)  
           0x00           Low Byte = irrelevant  
           0x00           High Byte = irrelevant

 Receive: 0xCA  
           0xll           Low Byte Wert  
           0xhh           High Byte Wert

12.

---

 Kommando: **Stelle Drehrichtung auf Links** (von hinten auf die Spindel gesehen)

 Transmit: **0x0B**           **Drehrichtung auf Links** (wie beim Lösen einer Schraube)  
           0x00           Low Byte = irrelevant  
           0x00           High Byte = irrelevant

 Receive: 0xCB  
           0xll           Low Byte Wert  
           0xhh           High Byte Wert

## Liste der adressierbaren internen Variablen

Variable	Adresse[Hex]	Multiplikator	weitere Operation
Wirkstrom	0BB6	0,01	in dez wandeln
SpindelSpannung	0BD4	0.1	in dez wandeln
ZK-Spannung	0BCC	0.1	in dez wandeln
LastProzent	08A4	0,1	in dez wandeln
Kühlkörpertemp	0CDA	0,1	in dez wandeln
Minimaldrehzahl	087C	10	
Maximaldrehzahl	087E	10	
Betriebsstunden Umf (H):	0AE2	1	
Betriebsstunden Umf (Min):	0AE4	1	
Verz Überlast	086C	1/256	
Verz Übertemp Umf	086E	1/256	
Verz Übertemp Spin	0870	1/256	
Verz RS232	0872	1/256	
Digital/Relais Ausgang	0908		
Analog Eingang 1	090A	1/1024	=> 10V/1024 proDigit
Analog Eingang 2	090C	1/1024	=> 10V/1024 proDigit
Analog Ausgang 1	090E		
Analog Ausgang 2	0910		
DigitalEingänge	0906		digital abgebildet
	<b>Bit</b>	<b>Funktion</b>	
	0	StartStop	
	1	Verriegelung NotAus	
	2	Verriegelung	
	3	Drehrichtung	
	4	FehlerReset	
	5	-	
	6	Impulssperre (falls vorhanden)	
	7	-	
	8	-	
	9	-	
	10	-	
	11	-	
	12	Übertemperatur Spindel	
	13	-	
	14	-	
	15	-	
FehlerStatus	085A		digital abgebildet
	<b>Bit</b>	<b>Funktion</b>	
	0	Fehler Überlast (Verz.abgelaufen)	
	1	Übertemp.Umformer (Verz.abgelaufen)	
	2	Übertemp.Spindel (Verz.abgelaufen)	
	3	Übertemp.Umf.od.Spn. (Verz.abgelaufen)	
	4	Überspannung aus	
	5	Unterspannung aus	
	6	Unterspannung Stop	
	7	Endstufe abgeschaltet (PDPINT)	
	8	Not-Aus-Eingang verriegelt	
	9	ohne Spindel bzw. Kabelbruch	
	10	Timeout ser. Schnittstelle	
	11	Kennlinie ungültig	
	12	Rückenergie zu groß	
	13	im Speicher....Stop	
	14	Kein SpindelStillstand	
	15	Fehler in der Feldplattenerfassung	

Beispiel: **aktuellen Wirkstrom auslesen**

Adresse: 0x0BB6

Kommando Datenpointer:

Transmit: **0x0C**  
0xB6 Low Byte Pointer  
0x0B High Byte Pointer

→ 0C B6 0B

akt Laststrom = 2,3A →  $2,3 * 100(\text{Multiplikator}) = 230\text{dez} = 0xE6\text{hex}$

Receive: 0xCC  
0xE6 Low Byte Wert  
0x00 High Byte Wert



Walpersdofer Str. 38  
D 91126 Schwabach  
Tel.: +49 (0)9122 63148-0  
Fax.: +49 (0)9122 63148-29  
e-mail: [Info@bmr-gmbh.de](mailto:Info@bmr-gmbh.de)  
Internet: [www.bmr-gmbh.de](http://www.bmr-gmbh.de)

*Quality  
Made in Germany*

Technische Änderungen vorbehalten.  
23. 11. 2016