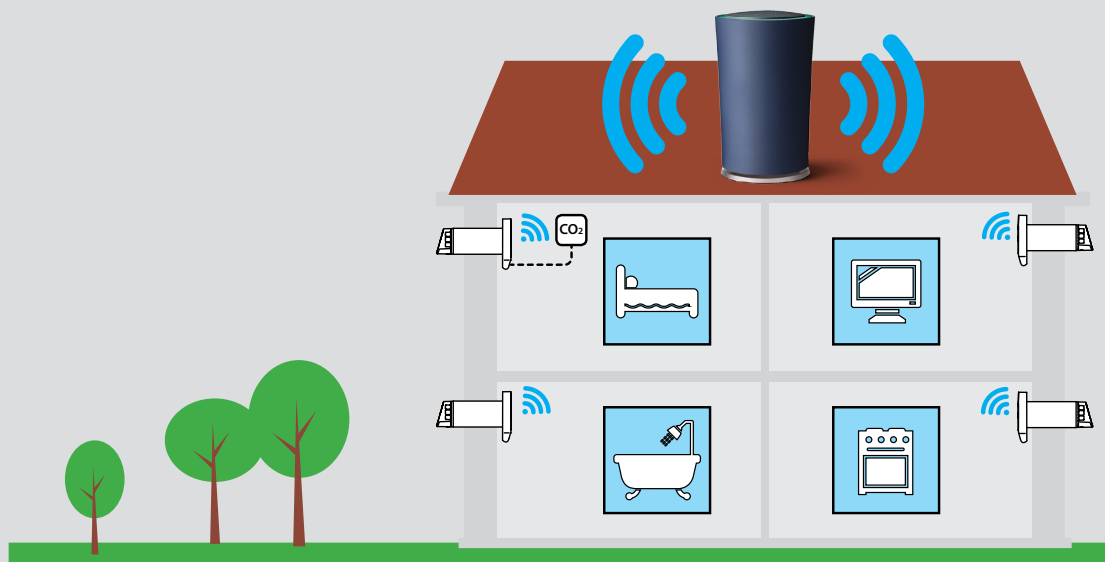


# ANSCHLUSSANLEITUNG

---

**Breezy**  
**Breezy Eco**

## Smart House



Verbindung mit dem Smart Home-System

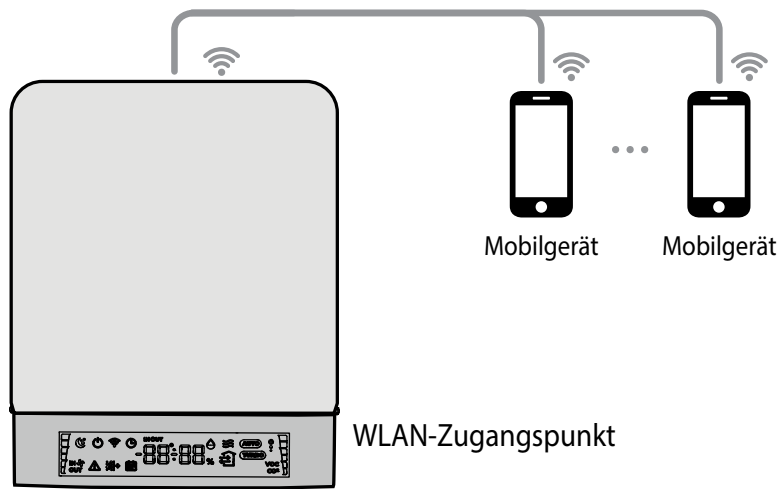
---

**INHALT**

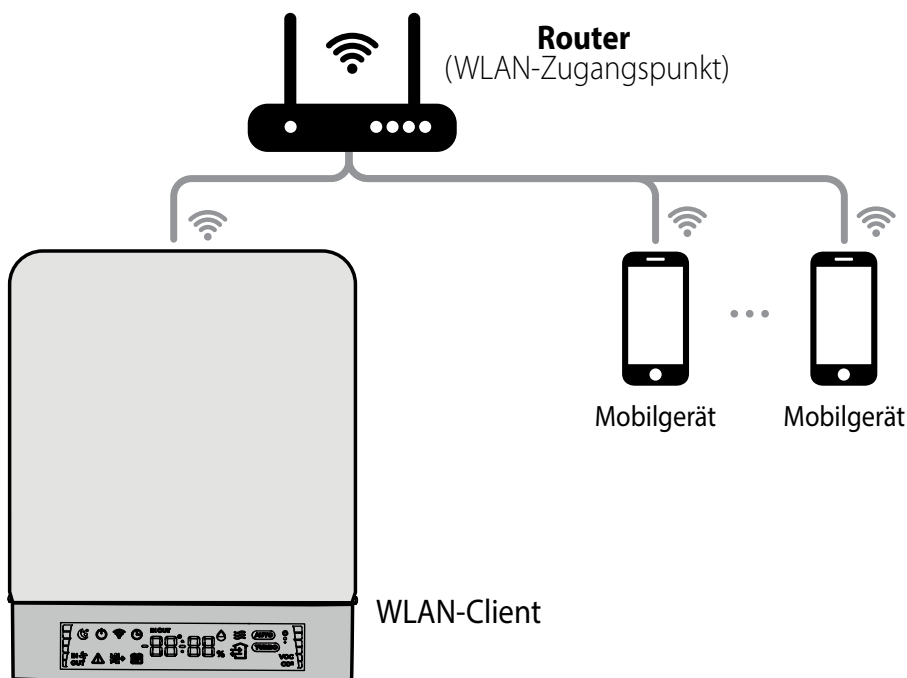
Verbindung und Einstellung ..... 2  
 Netzwerkparameter ..... 3  
 Paketstruktur ..... 4  
 Beispiele für die Verwendung spezieller Befehle im Datenblock ..... 5  
 Beispiele für ein vollständiges Paket ..... 6  
 Parametertabelle ..... 7  
 Beispiel der Paketverarbeitung, in C geschrieben ..... 12

**VERBINDUNG UND EINSTELLUNG**

**Beispiel 1:** Schema für die direkte Verbindung des Ventilators mit dem BMS Smart Home-System ohne Router. Konfigurieren Sie den Ventilator für den Betrieb über WLAN als Zugangspunkt (siehe Betriebsanleitung des Ventilators). Hinweis: Es können maximal acht Steuergeräte angeschlossen werden.



**Beispiel 2:** Schema für die Verbindung mit einem Router mit einem WLAN-Zugangspunkt. Der Ventilator, die Smartphones und das BMS Smart Home-System werden mit dem WLAN-Zugangspunkt des Routers verbunden.



## NETZWERKPARAMETER

Der Datenaustausch erfolgt über das Transportprotokoll UDP (Broadcasting wird unterstützt).

IP-Adresse des Geräts:

- 192.168.4.1 – wenn das Gerät ohne Router betrieben wird (Verbindungsschema Nr. 1);
- wenn das Gerät mit einem Router verbunden ist (Verbindungsschema Nr. 2), wird die IP-Adresse über die Smartphone-App konfiguriert (siehe Betriebsanleitung des Geräts) und kann als statisch oder dynamisch (DHCP) eingestellt werden.

Der Geräteport ist 4000.

Die maximale Paketgröße beträgt 256 Byte.

## PAKETSTRUKTUR

0xFD	0xFD	TYPE	SIZE ID	ID	SIZE PWD	PWD	FUNC	DATA	Chksum L	Chksum H
------	------	------	---------	----	----------	-----	------	------	----------	----------

**0xFD** **0xFD** – Paketanfang (2 Bytes).

**TYPE** – Protokolltyp (1 Byte). Der Wert ist 0x02.

**SIZE ID** – ID-Blockgröße (1 Byte). Der Wert ist 0x10.

**ID** – Steuereinheit-ID. Diese Nummer befindet sich auf einem Aufkleber (dargestellt als 16 Zeichen), der auf der Steuerplatine oder dem Gehäuse des Geräts angebracht ist.

Es ist auch möglich, das Codewort „DEFAULT\_DEVICEID“ als ID-Nummer zu verwenden. Es kann verwendet werden:

- zur Steuerung des Geräts, wenn es ohne Router betrieben wird (Verbindungsbeispiel Nr. 1);
  - zur Suche nach Geräten im Netzwerk, wenn ein Router verwendet wird (Verbindungsbeispiel Nr. 2);
- Gleichzeitig reagiert das Gerät nur auf zwei Parameter: 0x007C und 0x00B9 (siehe Parametertabelle).

**SIZE PWD** – **PWD**-Blockgröße (1 Byte). Möglicher Wertebereich: von 0x00 bis 0x08.

**PWD** – Gerätepasswort (gültige Zeichen sind: „0...9“, „a...z“, „A...Z“). Das Standardpasswort ist 1111.

Dieses Passwort kann im **Einstellungsmenü** der Smartphone-App geändert werden.

**FUNC** – Funktionsnummer (1 Byte). Bestimmt die mit den Daten auszuführende Aktion und die Struktur des **DATA**-Blocks:

0x01 – Lesen von Parametern;

0x02 – Schreiben von Parametern. Die Steuereinheit antwortet nicht auf die Anfrage nach dem Status dieser Parameter;

0x03 – Schreiben von Parametern mit der anschließenden Antwort der Steuereinheit über den Status der angegebenen Parameter;

0x04 – Inkrementieren von Parametern mit der anschließenden Antwort der Steuereinheit über den Status der angegebenen Parameter;

0x05 – Dekrementieren von Parametern mit der anschließenden Antwort der Steuereinheit über den Status der angegebenen Parameter;

0x06 – Antwort der Steuereinheit auf die Anfrage (FUNC = 0x01,0x03,0x04,0x05).

**DATA** – Datenblock. Es besteht aus Parameternummern und deren Werten:

wenn FUNC = 0x01 oder 0x04 oder 0x05:

P1	P2	Pn
----	----	----

wenn FUNC = 0x02 oder 0x03 oder 0x06:

P1	Value 1	P2	Value 2	Pn	Value n
----	---------	----	---------	----	---------

Die Parameternummern (siehe Parametertabelle) bestehen grob aus zwei Bytes (das High-Byte ist virtuell).

Standardmäßig ist das High-Byte jeder Parameternummer in jedem neuen Paket 0x00.

Das High-Byte kann innerhalb desselben Pakets mit dem Spezialbefehl 0xFF (siehe unten) geändert werden.

**P** – das Low-Byte der Parameternummer. Der mögliche Wertebereich: 0x00 – 0xFB. Die Werte 0xFC – 0xFF sind spezielle Befehle:

**0xFC** – Änderung der Funktionsnummer (**FUNC**). Das nächste Byte muss eine neue Funktionsnummer von 0x01 bis 0x05 sein. Es wird verwendet, um mehrere Funktionen mit verschiedenen Aktionen in einem einzigen Paket zu organisieren;

**0xFD** – der Parameter wird von der Steuereinheit nicht unterstützt. Das nächste Byte ist das Low-Byte des nicht unterstützten Parameters. Es wird verwendet, wenn die Steuereinheit auf eine Anforderung zum Lesen oder Schreiben eines nicht vorhandenen Parameters antwortet (**FUNC** = 0x06);

**0xFE** – die Größe des Parameterwerts Value für einen nachfolgenden Parameter zu ändern. Das nächste Byte muss die neue Parametergröße sein, gefolgt von dem niederwertigen Byte der Parameternummer und dann dem Wert Value selbst;

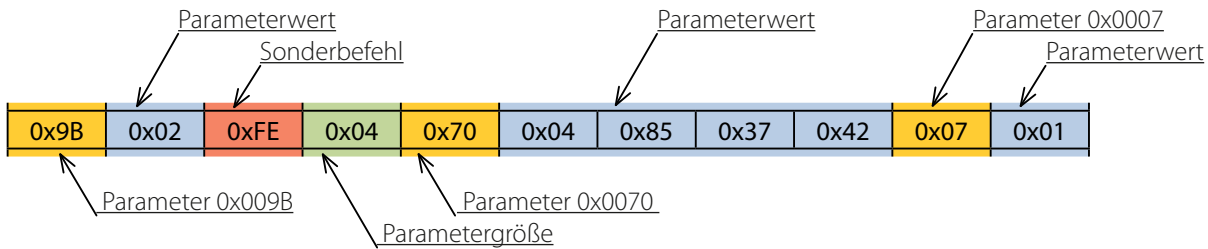
**0xFF** – das High-Byte für Parameternummern innerhalb desselben Pakets zu ändern. Das nächste Byte muss das neue High-Byte sein.

**Value** – den Wert des Parameters (standardmäßig 1 Byte). Die Byte-Reihenfolge ist von niedrig nach hoch.

**Chksum L** **Chksum H** – Prüfsumme (2 Bytes). Sie wird als Summe der Bytes berechnet, beginnend mit dem **TYPE**-Byte und endend mit dem letzten Byte des **DATA**-Blocks.  
**Prüfsumme L** ist das niedrigste Byte der Prüfsumme.  
**Prüfsumme H** ist das höchste Byte der Prüfsumme.

## BEISPIELE FÜR DIE VERWENDUNG SPEZIELLER BEFEHLE IM DATENBLOCK

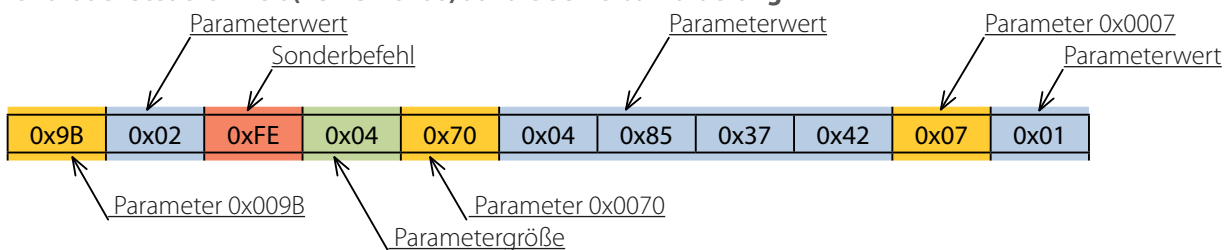
### Anforderung zum Schreiben (FUNC = 0x03) Parameter 0x009B, 0x0070, 0x0007



In der Schreibenanforderung:

- Dem Parameter 0x02 den Wert 0x009B zuweisen.
- Dem Parameter 0x42378504 den Wert 0x0070 zuweisen. Die Wertgröße beträgt 4 Bytes, wie durch den Sonderbefehl 0xFE + 0x04 angegeben.
- Dem Parameter 0x01 den Wert 0x0007 zuweisen.

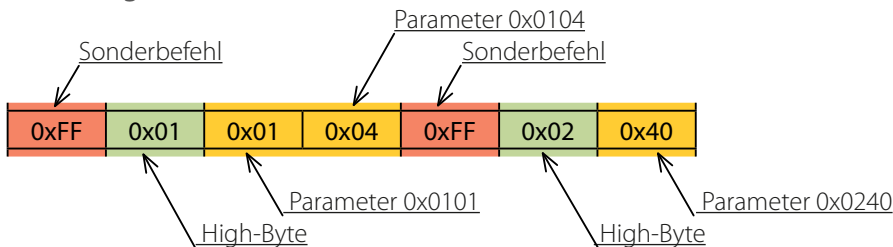
### Die Antwort der Steuereinheit (FUNC = 0x06) auf die Schreibenanforderung



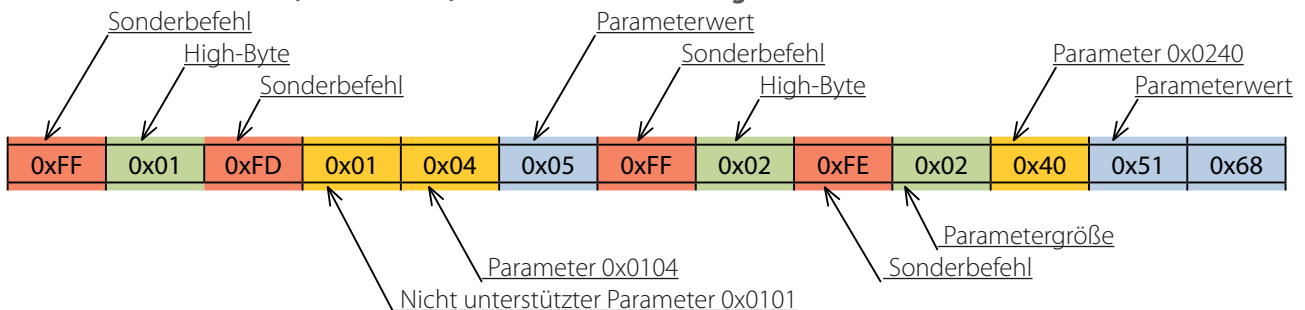
In der Antwort der Steuereinheit:

- Parameter 0x009B hat den Wert 0x02.
- Parameter 0x0070 hat den Wert 0x42378504. Die Wertgröße beträgt 4 Bytes, wie durch den Sonderbefehl 0xFE + 0x04 angegeben.
- Parameter 0x0007 hat den Wert 0x01.

### Leseanforderung (FUNC = 0x01) der Parameter 0x0101, 0x0104, 0x0240



### Die Antwort der Steuereinheit (FUNC = 0x06) auf die Leseanforderung



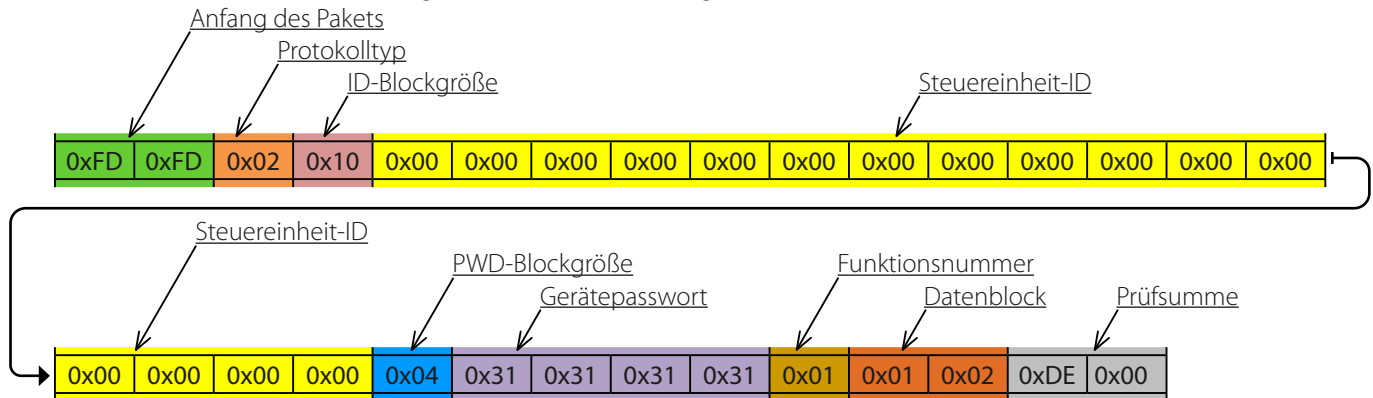
In der Antwort der Steuereinheit:

- Der Parameter 0x0101 wird von der Steuereinheit nicht unterstützt. Dies wird durch den Sonderbefehl 0xFD angezeigt.
- Parameter 0x0104 hat den Wert 0x05.
- Parameter 0x0240 hat den Wert 0x6851. Die Wertgröße beträgt 2 Bytes, wie durch den Sonderbefehl 0xFE + 0x02 angezeigt.

## BEISPIELE FÜR EIN VOLLSTÄNDIGES PAKET

### Senden des Pakets „Smart Home -> Steuereinheit“

Dieses Paket stellt eine Leseanforderung (FUNC = 0x01) für die folgenden Parameter dar: 0x0001, 0x0002.

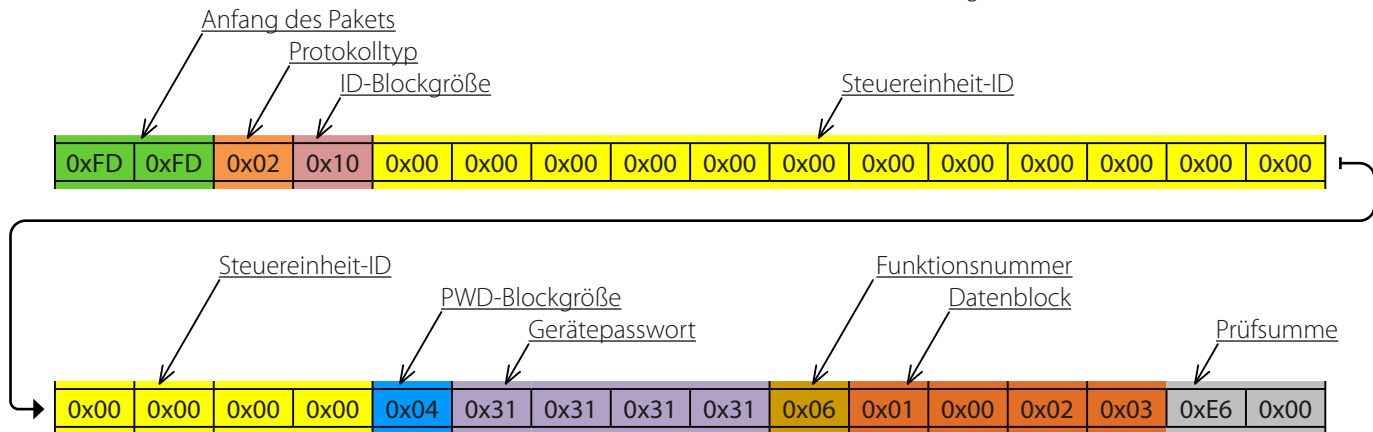


Die Anforderung umfasst:

- Prüfsumme: 0x00DE.

### Senden des Pakets „Steuereinheit -> Smart Home“

Dieses Paket enthält die Antwort der Steuereinheit (FUNC = 0x06) auf eine Leseanforderung.



In der Antwort der Steuereinheit:

- Parameter 0x0001 hat den Wert 0x00.
- Parameter 0x0002 hat den Wert 0x03.
- Prüfsumme: 0x00E6.

## PARAMETERTABELLE

Funktionen:		R – 0x01	INC – 0x04	RW – 0x03	W – 0x02	DEC – 0x05
Parameternummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes		
1/0x0001	R/W/RW	Ein-/Ausschalten der Anlage	0: AUS 1: EIN 2: invertieren	1		
2/0x0002	R/W/RW/INC/DEC	Drehzahlbetrieb	1: erste Lüftungsstufe 2: zweite Lüftungsstufe 3: dritte Lüftungsstufe 4: vierte Lüftungsstufe 5: fünfte Lüftungsstufe 255: manuelle Einstellung der Lüftungsstufe (siehe Parameter 68)	1		
7/0x0007	R/W/RW/INC/DEC	Timer	0: AUS 1: Nacht 2: Turbo	1		
11/0x000B	R	Aktueller Countdown-Zeit des Timers	1. Byte: Sekunden (0...59) 2. Byte: Minuten (0...59) 3. Byte: Stunden (0...23)	3		
15/0x000F	R/W/RW	Steuerung mit Feuchtigkeitssensor	0: AUS 1: EIN 2: invertieren	1		
17/0x0011	R/W/RW	Steuerung mit CO <sub>2</sub> -Sensor	0: AUS 1: EIN 2: invertieren	1		
25/0x0019	R/W/RW/INC/DEC	Einstellung des Feuchtesollwerts	40 .. 80 RH%	1		
26/0x001A	R/W/RW/INC/DEC	Einstellung des CO <sub>2</sub> -Sollwerts	400...2000 ppm	2		
31/0x001F	R	Aktuelle Außenlufttemperatur	-32768: kein Sensor +32767: Kurzschluss	signed 2 (muss durch 10 geteilt werden, eine Dezimalstelle)		
32/0x0020	R	Aktuelle Zulufttemperatur nach dem Nachheizregister	-32768: kein Sensor +32767: Kurzschluss	signed 2 (muss durch 10 geteilt werden, eine Dezimalstelle)		
33/0x0021	R	Aktuelle Zulufttemperatur am Anlageneingang	-32768: kein Sensor +32767: Kurzschluss	signed 2 (muss durch 10 geteilt werden, eine Dezimalstelle)		
34/0x0022	R	Aktuelle Zulufttemperatur am Anlagenausgang	-32768: kein Sensor +32767: Kurzschluss	signed 2 (muss durch 10 geteilt werden, eine Dezimalstelle)		
36/0x0024	R	Aktuelle RTC-Batteriespannung	0...5000 mV	2		
37/0x0025	R	Aktuelle Raumluftfeuchtigkeit	0 – 100 RH%	1		
39/0x0027	R	Aktueller CO <sub>2</sub> -Gehalt im Innenraum	0...2000 ppm	2		
58/0x003A	R/W/RW/INC/DEC	Zuluftventilator-Drehzahl bei 1. Lüftungsstufe	10...100 %	1		
59/0x003B	R/W/RW/INC/DEC	Abluftventilator-Drehzahl bei 1. Lüftungsstufe	10...100 %	1		

Parameternummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
60/0x003C	R/W/RW/INC/DEC	Zuluftventilator-Drehzahl bei 2. Lüftungsstufe	10...100 %	1
61/0x003D	R/W/RW/INC/DEC	Abluftventilator-Drehzahl bei 2. Lüftungsstufe	10...100 %	1
62/0x003E	R/W/RW/INC/DEC	Zuluftventilator-Drehzahl bei 3. Lüftungsstufe	10...100 %	1
63/0x003F	R/W/RW/INC/DEC	Abluftventilator-Drehzahl bei 3. Lüftungsstufe	10...100 %	1
68/0x0044	R/W/RW/INC/DEC	Ventilator-drehzahl im manuellen Drehzahlmodus	10...100 %	1
74/0x004A	R	Zuluft-Ventilator-drehzahl	0 ... 5000 U/min	2
75/0x004B	R	Abluft-Ventilator-drehzahl	0 ... 5000 U/min	2
99/0x0063	R/W/RW/INC/DEC	Zeiteinstellung des Filterwechsel-Timers	0, 70 ... 365 Tage	2
100/0x0064	R	Countdown des Timers bis zum Filterwechsel	1. Byte: Minuten (0...59) 2. Byte: Stunden (0...23) 3., 4. Byte: Tage (0...365)	4
101/0x0065	W	Countdown des Timers bis zum Filterwechsel zurücksetzen	1: ausführen	1
104/0x0068	R/W/RW	Steuerung des Heizregisters	0: AUS 1: EIN 2: invertieren	1
111/0x006F	R/W/RW	RTC-Zeit	1. Byte: RTC-Sekunden (0...59) 2. Byte: RTC-Minuten (0...59) 3. Byte: RTC-Stunden (0...23)	3
112/0x0070	R/W/RW	RTC-Kalender	1. Byte: RTC-Tag (1...31) 2. Byte: RTC-Wochentag (1...7) 3. Byte: RTC-Monat (1...12) 4. Byte: RTC-Jahr (0...99)	4
114/0x0072	R/W/RW	Wochenplan-Modus	0: AUS 1: EIN 2: invertieren	1
119/0x0077	R/W/RW	<p>Zeitplan-Einstellung</p> <p>Bei der Leseanforderung muss der spezielle Befehl 0xFE verwendet und der Wert des Parameters 0x02 angegeben werden, um den gewünschten Wochentag und die Nummer des Zeitraums auszuwählen.</p> <p>Bei der Schreibanforderung werden alle 6 Bytes verwendet, ebenso bei der Antwort der Steuereinheit.</p> <p>Der Beginn des ersten Zeitraums ist immer 00:00, und der Beginn jedes folgenden Zeitraums ist das Ende des vorherigen.</p> <p>Das Ende des letzten Zeitraums ist immer 24:00.</p>	<p>1. Byte: Wochentag: 0: Alle Tage (write only) 1: Montag 2: Dienstag 3: Mittwoch 4: Donnerstag 5: Freitag 6: Samstag 7: Sonntag 8: Mo...Fr (write only) 9: Sa...So (write only)</p> <p>2. Byte: Periodennummer: 1...4</p> <p>3. Byte: Drehzahl 0: Standby 1...5</p> <p>4. Byte: reserviert</p> <p>5. Byte: Minuten bis zum Ende des Abstands: 0...59</p> <p>6. Byte: Stunden bis zum Ende des Abstands: 0...23</p>	6

Parameternummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
124/0x007C	R	Suche nach Geräten im lokalen Ethernet-Netzwerk. Die Antwort enthält die Geräte-ID.	Text („0...9“, „A...F“)	16
125/0x007D	R/W/RW	Gerätepasswort für das Ethernet-Netzwerk	Text („0...9“, „a...z“, „A...Z“)	0-8
126/0x007E	R	Motorbetriebsstunden	1. Byte: Minuten (0...59) 2. Byte: Stunden (0...23) 3. und 4. Byte: Tage (0...65535)	4
127/0x007F	R	Liste der aktuellen Alarme/ Warnungen	1. Byte: Code 2. Byte: Typ: 1: Alarm 2: Warnung	0,2,4...
128/0x0080	W	Alarme zurücksetzen	1: ausführen	1
129/0x0081	R	Zustand des Heizregisters	0: AUS 1: EIN	1
131/0x0083	R	Alarm-/Warnungsanzeige	0: keine 1: Alarm (hat höhere Priorität) 2: Warnung	1
132/0x0084	R	Luftqualitätsstatus	1. Byte: RH: 0: normal 1: über Sollwert 2. Byte: CO <sub>2</sub> : 0: normal 1: über Sollwert 3. Byte: reserviert 4. Byte: reserviert 5. Byte: VOC (AQI): 0: normal 1: über Sollwert	5
133/0x0085	R/W/RW	Betrieb über Cloud-Server aktivieren	0: AUS 1: EIN 2: invertieren	1
134/0x0086	R	Firmware-Version und Datum der Firmware-Version der Steuereinheit	1. Byte: Firmwareversion (major) 2. Byte: Firmwareversion (minor) 3. Byte: Tag 4. Byte: Monat 5., 6. Byte: Jahr	6
135/0x0087	W	Alle Parameter auf Werkseinstellungen zurücksetzen	1: ausführen	1
136/0x0088	R	Zustand des Filters	0: sauber 1: verschmutzt	1
148/0x0094	R/W/RW/INC/DEC	WLAN-Betrieb	1: client 2: access point	1
149/0x0095	R/W/RW	WLAN-Name im Clientbetrieb	Text	1...32
150/0x0096	R/W/RW	WLAN-Passwort	Text	8...64
153/0x0099	R/W/RW	WLAN-Verschlüsselungstechnologie	48: OPEN, 50: WPA_PSK, 51: WPA2_PSK, 52: WPA_WPA2_PSK	1
154/0x009A	R/W/RW/INC/DEC	WLAN-Frequenzkanal	1...13	1
155/0x009B	R/W/RW	WLAN-Modul DHCP	0: STATIC 1: DHCP 2: invertieren	1

Parameternummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
156/0x009C	R/W/RW	Zugewiesene IP-Adresse des WLAN-Moduls	1. Byte: 0...255 2. Byte: 0...255 3. Byte: 0...255 4. Byte: 0...255	4
157/0x009D	R/W/RW	Subnetzmaske des WLAN-Moduls	1. Byte: 0...255 2. Byte: 0...255 3. Byte: 0...255 4. Byte: 0...255	4
158/0x009E	R/W/RW	Haupt-Gateway des WLAN-Moduls	1. Byte: 0...255 2. Byte: 0...255 3. Byte: 0...255 4. Byte: 0...255	4
160/0x00A0	W	Neue WLAN-Parameter übernehmen und Einstellungsmodus des WLAN-Moduls verlassen	1: ausführen	1
162/0x00A2	W	Einstellungsmodus des WLAN-Moduls ohne Verwendung der neuen WLAN-Parameter verlassen	1: ausführen	1
163/0x00A3	R	Aktuelle IP-Adresse des WLAN-Moduls	1. Byte: 0...255 2. Byte: 0...255 3. Byte: 0...255 4. Byte: 0...255	4
183/0x00B7	R/W/RW/INC/DEC	Drehrichtung des Ventilators	0: Lüftung 1: Rückgewinnung 2: Zuluft 3: Abluft	1
185/0x00B9	R	Anlagentyp	0: 65535, 17: Breezy 160, 20: Breezy Eco 160, 22: Breezy 200, 24: Breezy Eco 200	2
252/0x00FC	<b>Spezialbefehle</b>			
253/0x00FD				
254/0x00FE				
255/0x00FF				
297/0x0129	R	Wärmerückgewinnungseffizienz	0...100 %	1
298/0x012A	R/W/RW	Die Einstellungen für die Ventilator Drehzahl für alle Betriebe auf die Werkseinstellungen zurücksetzen	1: ausführen	1
770/0x0302	R/W/RW	Einstellung der Zeit für den Nacht-Timer	0. Byte: Minuten 1. Byte: Stunden	2
771/0x0303	R/W/RW	Einstellung der Zeit für den Turbo-Timer	0. Byte: Minuten 1. Byte: Stunden	2
774/0x0306	R	Aktuelle Drehzahl im Zeitplanbetrieb	0...3	1
779/0x030B	R	Frostschutzstatus	0: inaktiv 1: aktiv	1
789/0x0315	R/W/RW	Luftqualitätssensor-basierte (VOC) Steuerung	0: AUS 1: EIN 2: invertieren	1
799/0x031F	R/W/RW/INC/DEC	Luftqualitätssensor (VOC)-Sollwert	50-250 index	2
800/0x0320	R	Aktuelle Luftqualitätsstufe (VOC)	0 – 500 index	2

Parameternummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
1024/0x0400	R/W/RW/INC/DEC	Helligkeit der Bildschirmbeleuchtung (manuell)	1...100	1
1025/0x0401	R/W/RW	Aktivierung des Tonsenders (Reaktion auf Steuerbefehle)	0: AUS 1: EIN 2: invertieren	1
1026/0x0402	R/W/RW	Betrieb der Bildschirmbeleuchtung	0: Auto 1: Manual 2: invertieren	1
1027/0x0403	R/W/RW/INC/DEC	Auswahl des Temperatursensors, der auf dem Bildschirm des Geräts angezeigt werden soll	0: alternierend 1: Zuluft 2: Abluft	1
1028/0x0404	R/W/RW/INC/DEC	Auswahl des Luftqualitätssensors, der auf dem Bildschirm des Geräts angezeigt werden soll	0: alternierend 1: CO <sub>2</sub> 2: VOC	1
1029/0x0405	R/W/RW/INC/DEC	Anzeige der Uhrzeit oder der Temperatur/Luftfeuchtigkeit auf dem Bildschirm	0: alternierend 1: Zeit 2: Temperatur und Luftfeuchtigkeit	1
1030/0x0406	R/W/RW	Anzeige der Zeit im Standby-Betrieb auf dem Bildschirm	0: AUS 1: EIN	1
1031/0x0407	R/W/RW	Aktivieren/Deaktivieren der gesamten Bildschirmanzeige	0: AUS 1: EIN 2: Intervall, während dem die Anzeige ausgeschaltet ist	1
1032/0x0408	R/W/RW	Startzeit für das Ausschalten der Bildschirmanzeige	0. Byte: Minuten (0...59) 1. Byte: Stunden (0...23)	2
1033/0x0409	R/W/RW	Endzeit für das Ausschalten der Bildschirmanzeige	0. Byte: Minuten (0...59) 1. Byte: Stunden (0...23)	2

## BEISPIEL DER PAKETVERARBEITUNG, IN C GESCHRIEBEN

```
//===== Spezialbefehle =====//
#define BGCP_CMD_PAGE                0xFF
#define BGCP_CMD_FUNC                0xFC
#define BGCP_CMD_SIZE                0xFE
#define BGCP_CMD_NOT_SUP             0xFD
//=====//

#define BGCP_FUNC_RESP                0x06

uint8_t receive_data[256];
uint16_t receive_data_size;
uint8_t State_Power;
uint8_t State_Speed_mode;
char current_id[17] = "002D6E1B34565815"; // ID-Nummer der Steuereinheit

//***** Anfang der Prüfsumme und Beginn des Pakets *****//
uint8_t check_protocol(uint8_t *data, uint16_t size)
{
    uint16_t i, chksum1 = 0, chksum2 = 0;
    if((data[0] == 0xFD) && (data[1] == 0xFD))
    {
        for(i = 2; i <= size-3; i++)
            chksum1 += data[i];
        chksum2 = (uint16_t)(data[size-1] << 8) | (uint16_t)(data[size-2]);
        if(chksum1 == chksum2)
            return 1;
        else
            return 0;
    }
    else
        return 0;
}
//*****//

int main(void)
{
    ...

    if(check_protocol(receive_data, receive_data_size) == 1) // Prüfsumme
    {
        if(receive_data[2] == 0x02) // Protokolltyp
        {
            if(memcmp(&receive_data[4], current_id, receive_data[3]) == 0) // ID
            {
                uint16_t jump_size = 0, page = 0, param, param_size, r_pos;
                uint8_t flag_check_func = 1, BGCP_func;

                r_pos = 4 + receive_data[3];
                r_pos += 1 + receive_data[r_pos]; // Position im Array, wo FUNC-Block startet
                //***** FUNC und DATA *****//
                for(; r_pos < receive_data_size - 2; r_pos++)
                {
                    //===== Spezialbefehle =====//
                    param_size = 1;
                    //=== Neue Funktionsnummer
                    if((flag_check_func == 1) || (receive_data[r_pos] == BGCP_CMD_FUNC))
                    {
                        if(receive_data[r_pos] == BGCP_CMD_FUNC)
                            r_pos++;
                        flag_check_func = 0;
                        BGCP_func = receive_data[r_pos];
                        if(BGCP_func != BGCP_FUNC_RESP) // Wenn die Funktionsnummer nicht unterstützt wird
                            break;
                        continue;
                    }
                    //=== Neuer High-Byte-Wert für Parameternummer
                    else if(receive_data[r_pos] == BGCP_CMD_PAGE)
                    {

```

```
    page = receive_data[++r_pos];
    continue;
}
//=== neuer Wert der Parametergröße
else if(receive_data[r_pos] == BGCP_CMD_SIZE)
{
    param_size = receive_data[++r_pos];
    r_pos++;
}
//=== falls der Parameter nicht unterstützt wird
else if(receive_data[r_pos] == BGCP_CMD_NOT_SUP)
{
    r_pos++;
    //***** Verarbeitung der nicht unterstützten Parameter *****/
    param = (uint16_t)(page << 8) | (uint16_t)(receive_data[r_pos]);
    switch(param)
    {
        case 0x0001:
            break;
        case 0x0002:
            break;
        ...
    }
    //*****//
    continue;
}
jump_size = param_size;
//=====//

//***** Verarbeitung der unterstützten Parameter *****/
param = (uint16_t)(page << 8) | (uint16_t)(receive_data[r_pos]);
switch(param)
{
    case 0x0001:
        State_Power = receive_data[r_pos+1];
        break;
    case 0x0002:
        State_Speed_mode = receive_data[r_pos+1];
        break;
    ...
}
//*****//
r_pos += jump_size;
}
//*****//
}
}
}
```





