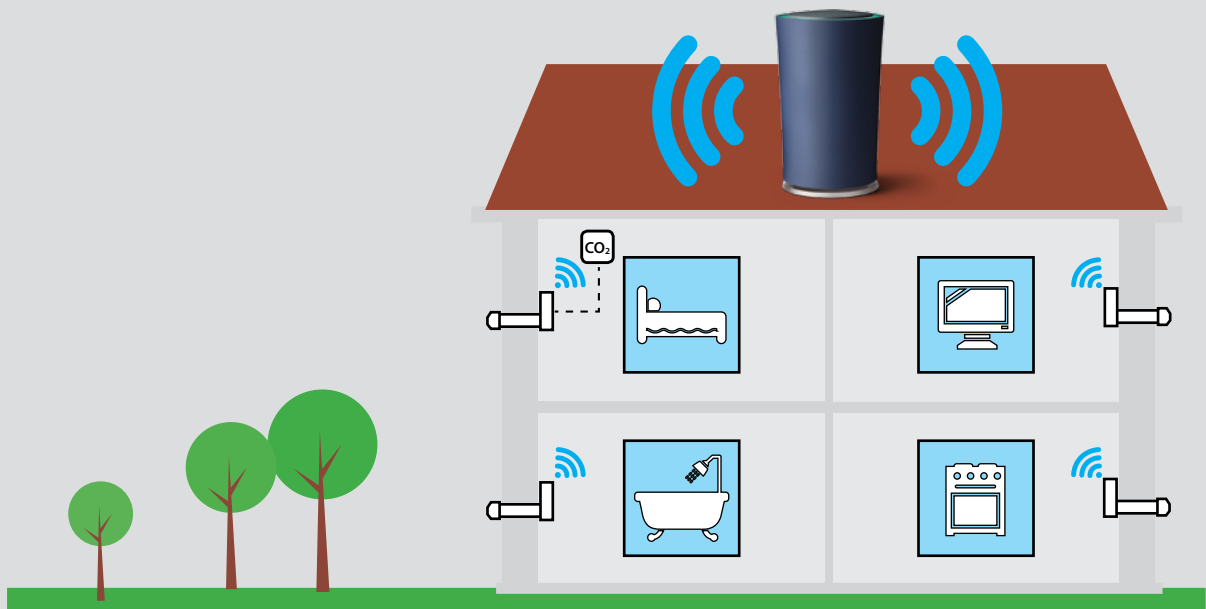


# Smart Home



Anschluss an ein "Smart Home" System

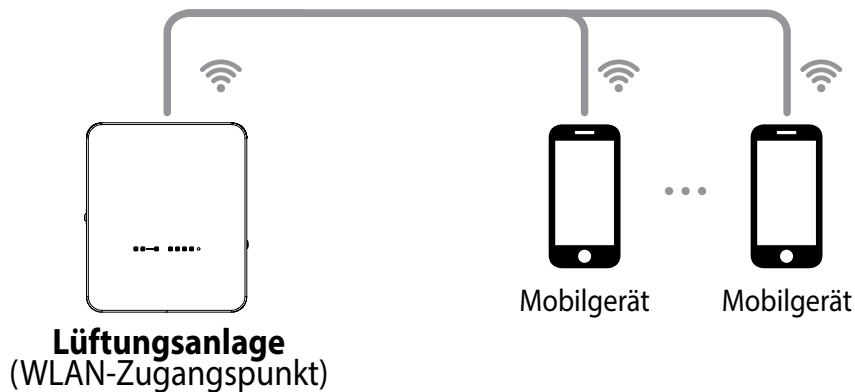
---

## INHALT

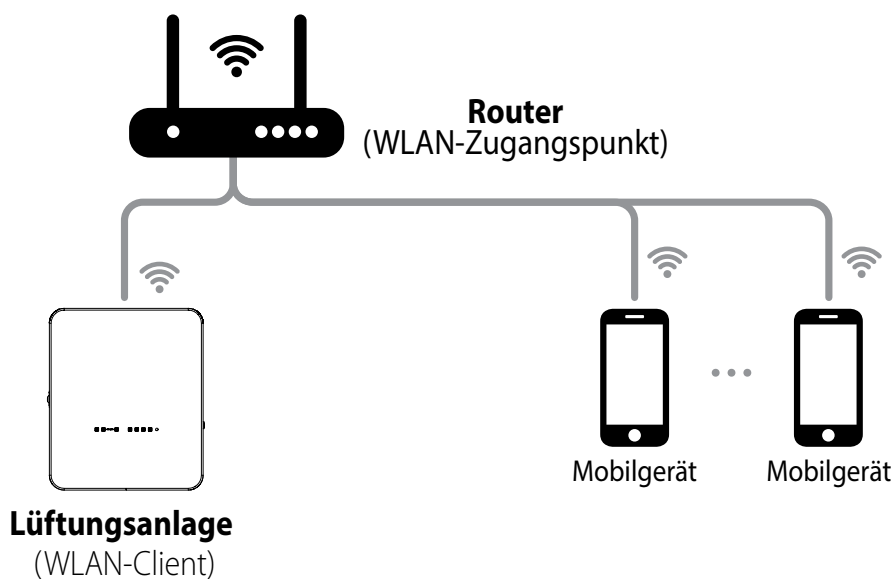
Anschluss und Einstellung.....	2
Netzparameter .....	3
Packetstruktur.....	4
Anwendungsbeispiele der speziellen Befehle im Datenblock.....	5
Beispiele eines kompletten Pakets .....	6
Parametertabelle .....	7
Beispiel der paketverarbeitung, in C geschrieben.....	13

## ANSCHLUSS UND EINSTELLUNG

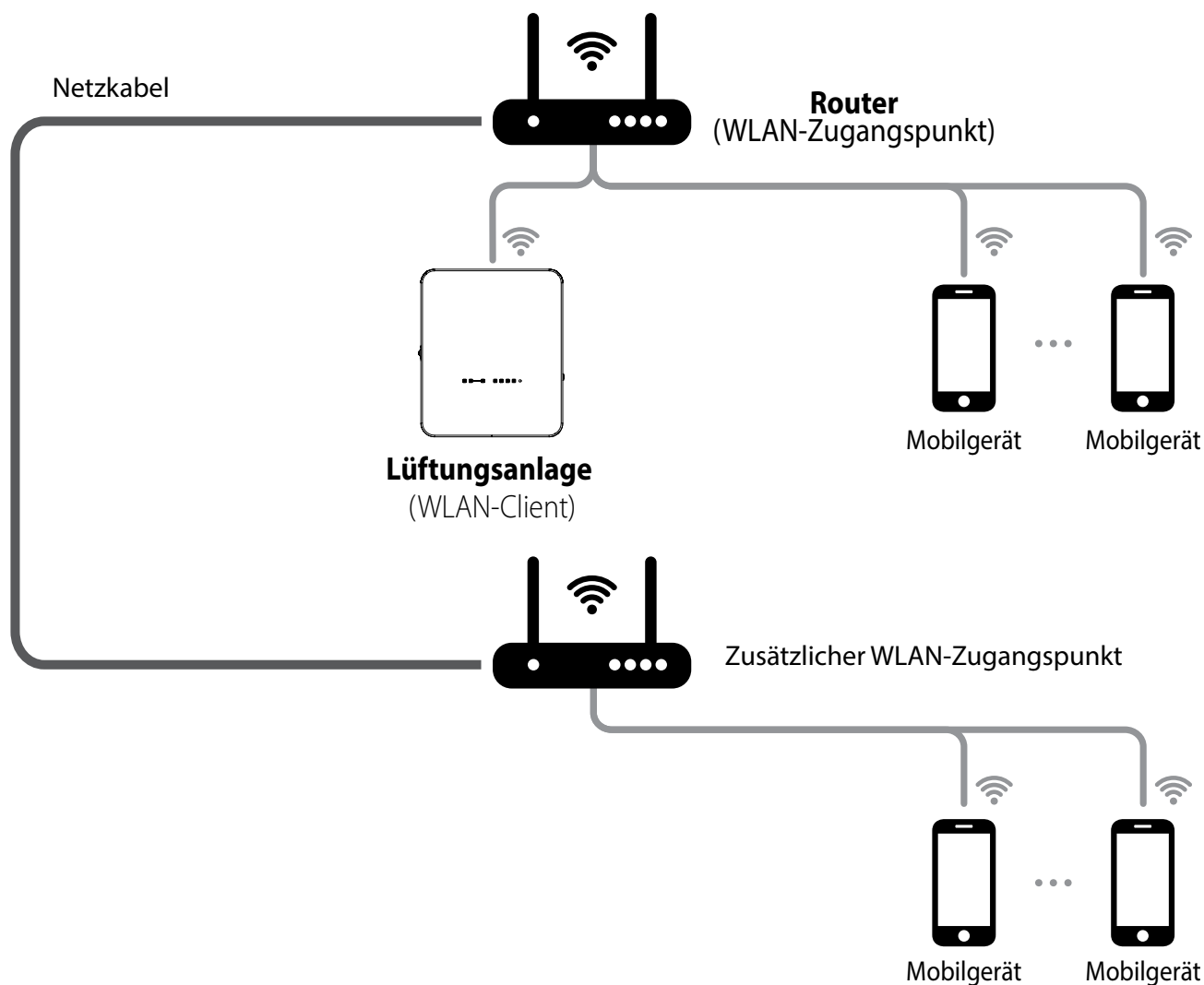
**Beispiel 1:** Schema des direkten Anschlusses der Lüftungsanlage an das Smart Home-System ohne Verwendung eines Routers. Stellen Sie die Lüftungsanlage für den Betrieb von WLAN im Zugangspunktmodus ein (siehe Betriebsanleitung der Lüftungsanlage). Anmerkung: Es können maximal acht Steuergeräte angeschlossen werden.



**Beispiel 2:** Anschlussschema mit einem Router mit einem einzelnen WLAN-Zugangspunkt. Lüftungsanlagen, Mobilgeräte und das Smart Home-System sind mit dem WLAN-Zugangspunkt des Routers verbunden.



**Beispiel 3:** Schema des direkten Anschlusses des Smart Home-Systems mit einem Router, an den mehrere WLAN-Zugangspunkte angeschlossen sind.



## NETZPARAMETER

Der Datenaustausch erfolgt über das UDP-Protokoll (mit Broadcast-Unterstützung).

IP-Adresse der Master-Anlage:

- 192.168.4.1: Wenn die Master-Anlage ohne Router läuft (Anschlussschema Nr. 1).
- Wenn die Master-Anlage über einen Router verbunden ist (Anschlussschema Nr. 2), wird die IP-Adresse über die App eingestellt (siehe Betriebsanleitung des Geräts) und kann statisch oder dynamisch (DHCP) bestimmt werden.

Port der Master-Anlage: 4000

Maximale Paketgröße: 256 Bytes

## PACKETSTRUKTUR

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

**0xFD 0xFD** : Paketanfangszeichen (2 Bytes).

**TYPE** : Protokolltyp (1 Byte). Wert = 0x02.

**SIZE ID** : ID-Blockgröße (1 Byte). Wert = 0x10.

**ID** : ID-Nummer der Steuereinheit. Diese Nummer ist auf dem Etikett (16 Zeichen) auf der Steuerplatine oder dem Gerätegehäuse angeführt.

Sie können die ID-Nummer auch durch das Codewort „DEFAULT\_DEVICEID“ ersetzen. Die ID-Nummer kann verwendet werden:

- Zur Steuerung, wenn die Master-Anlage ohne Router läuft (Anschlussschema Nr. 1).
- Um im Netzwerk nach Master-Anlagen zu suchen, wenn ein Router verwendet wird (Anschlussschema Nr. 2). In diesem Fall reagiert die Anlage nur auf zwei Parameter: 0x007C und 0x00B9 (siehe Parametertabelle).

**SIZE PWD** : PWD-Blockgröße (1 Byte). Mögliche Werte: von 0x00 bis 0x08.

**PWD** : Passwort der Anlage (zulässige Zeichen: „0... 9“, „a... z“ und „A... Z“). Das Standardpasswort lautet „1111“.

Dieses Passwort kann über die App im Menü **Verbindung** -> **Lokal** -> **Einstellungen** geändert werden (siehe Betriebsanleitung des Geräts).

**FUNC** : die Funktionsnummer (1 Byte). Es definiert die Aktion mit den Daten und die **DATA**-Blockstruktur:

0x01: Parameterlesen.

0x02: Parameterschreiben. Die Steuereinheit sendet keine Antwort bezüglich des Status der angegebenen Parameter.

0x03: Parameterschreiben mit anschließender Antwort der Steuereinheit bezüglich des Status der angegebenen Parameter.

0x04: Parameterinkrement mit anschließender Antwort der Steuereinheit bezüglich des Status der angegebenen Parameter.

0x05: Parameterdekrement mit anschließender Antwort der Steuereinheit bezüglich des Status der angegebenen Parameter.

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

**DATA** : Datenblock. Es besteht aus Parameternummern und ihren 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
----	---------	----	---------	----	---------

Parameternummern (siehe Parametertabelle) bestehen aus zwei Bytes (High-Byte ist virtuell). Standardmäßig entspricht das High-Byte jeder Parameternummer in jedem neuen Paket 0x00. Das High-Byte kann innerhalb eines einzelnen Pakets mit dem speziellen Befehl **0xFF** geändert werden (siehe unten).

**P** ist das Low-Byte der Parameternummer. Mögliche Werte: 0x00–0xFB. Die Werte 0xFC–0xFF sind Spezialbefehle:

**0xFC** : Funktionsnummer ändern (**FUNC**). Das folgende Byte muss die neue Funktionsnummer sein, die von 0x01 bis 0x05 reicht. Dieser Befehl wird verwendet, um mehrere Funktionen mit unterschiedlichen Aktionen in einem einzigen Paket zu organisieren.

**0xFD** : Parameter wird von der Steuereinheit nicht unterstützt. Das nachfolgende Byte ist das Low-Byte des nicht unterstützten Parameters. Dieser Befehl wird in der Antwort der Steuereinheit (**FUNC** = 0x06) auf eine nicht unterstützte Lese- oder Schreibanforderung von Parametern verwendet.

**0xFE** : Größe des Parameterwerts **Value** für einen der folgenden Parameter ändern. Das nachfolgende Byte muss die neue Parametergröße sein, gefolgt vom Low-Byte der Parameternummer und dann – vom Wert (**Value**) selbst.

**0xFF** : High-Byte für Parameternummern innerhalb eines einzelnen Pakets ändern. Das nachfolgende Byte muss das neue High-Byte sein.

**Value** : Parameterwert (standardmäßig 1 Byte). Byteanordnung von Low-Byte zu High-Byte.

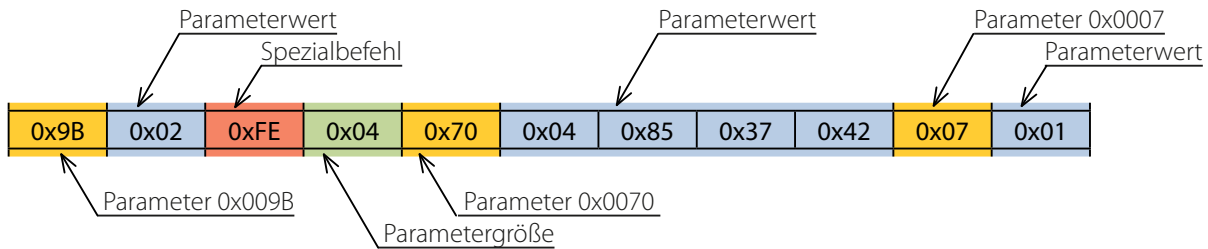
**Chksum L Chksum H** : Prüfsumme (2 Bytes). Dies wird als die Gesamtzahl von Bytes berechnet, die mit dem **TYPE**-Byte beginnen und mit dem letzten Byte des **DATA**-Blocks enden.

**Chksum L**: Low-Byte der Prüfsumme.

**Chksum H**: High-Byte der Prüfsumme.

## ANWENDUNGSBEISPIELE DER SPEZIELLEN BEFEHLE IM DATENBLOCK

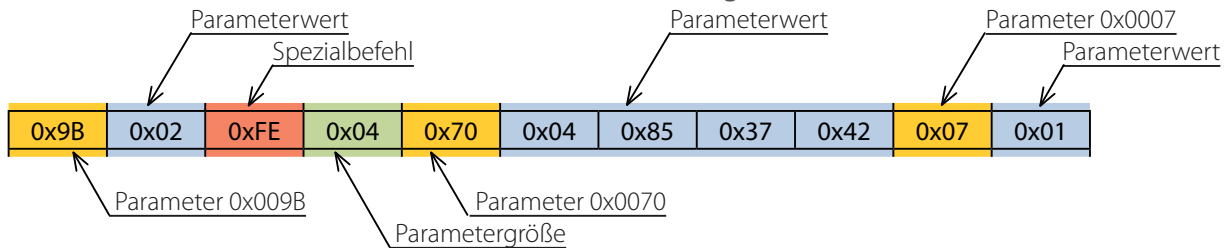
### Schreibanforderung (FUNC = 0x03) für Parameter mit den Nummern 0x009B, 0x0070 und 0x0007



Details der Schreibanforderung:

- Dem Parameter 0x009B muss der Wert 0x02 zugewiesen werden.
- Dem Parameter 0x0070 muss der Wert 0x42378504 zugewiesen werden. Die Wertgröße beträgt 4 Bytes, wie durch den Spezialbefehl 0xFE + 0x04 angegeben.
- Dem Parameter 0x0007 muss der Wert 0x01 zugewiesen werden.

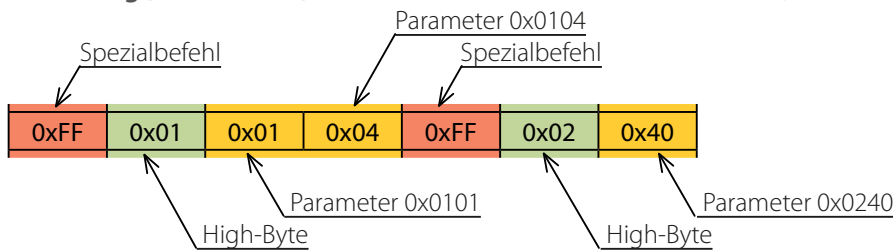
### Antwort der Steuereinheit (FUNC = 0x06) auf die Schreibanforderung



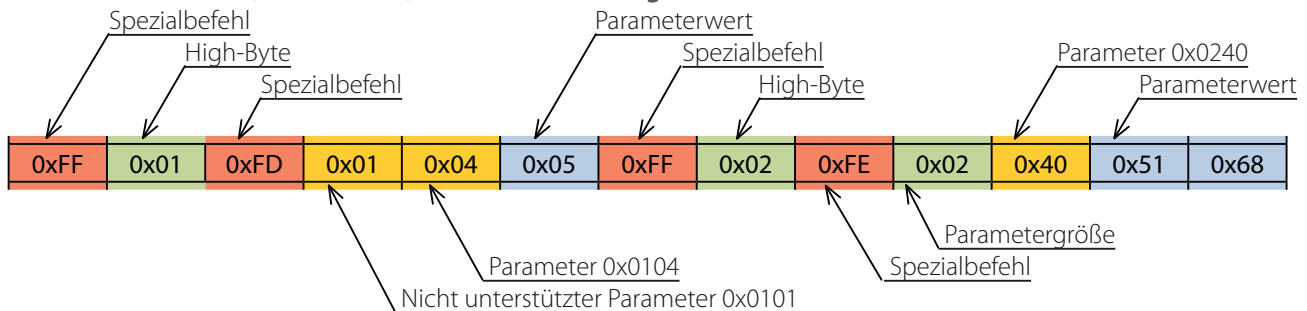
Antwort der Steuereinheit:

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

### Leseanforderung (FUNC = 0x01) für Parameter mit den Nummern 0x0101, 0x0104 und 0x0240



### Antwort der Steuereinheit (FUNC = 0x06) zur Leseanforderung



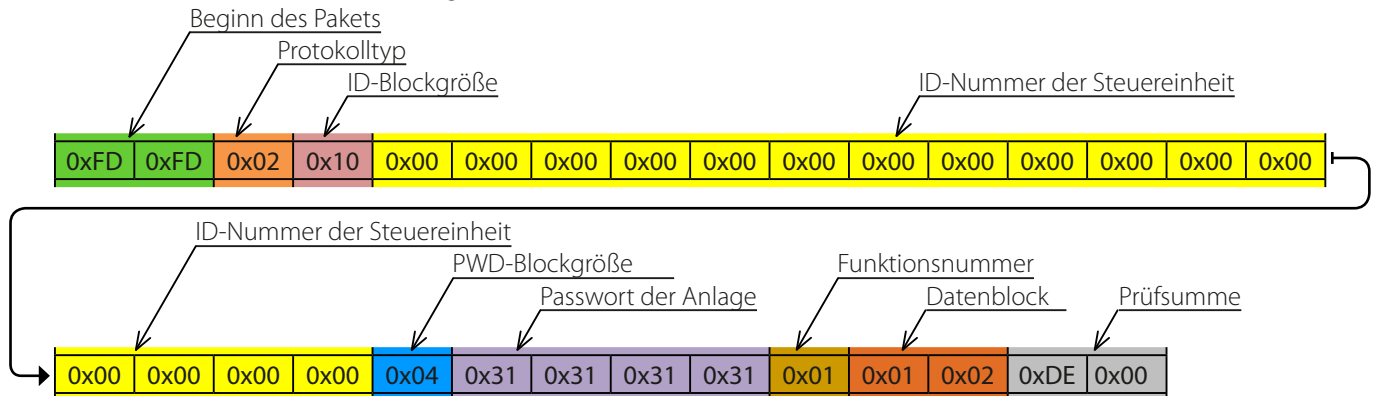
Antwort der Steuereinheit:

- Der Parameter 0x0101 wird von der Steuereinheit nicht unterstützt, wie durch den Spezialbefehl 0xFD angegeben.
- Parameter 0x0104 entspricht 0x05.
- Parameter 0x0240 entspricht 0x6851. Die Wertgröße beträgt 2 Bytes, wie durch den Spezialbefehl 0xFE + 0x02 angegeben.

## BEISPIELE EINES KOMPLETTEN PAKETS

### Senden des Pakets „Smart Home → Steuereinheit“

Dieses Paket enthält eine Leseanforderung (FUNC = 0x01) für Parameter mit den Nummern: 0x0001, 0x0002.

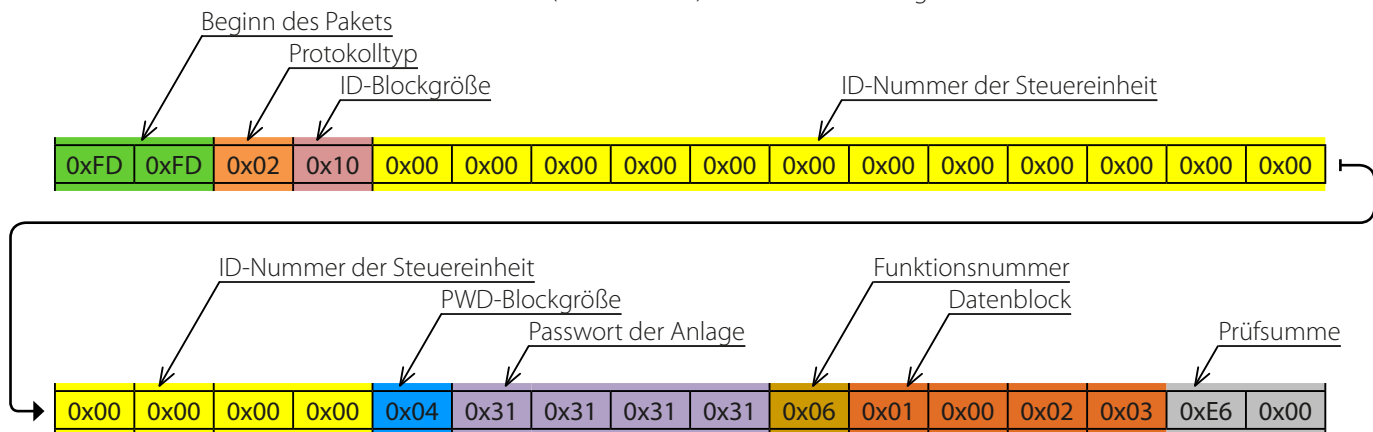


Anfragedetails:

- Prüfsumme: 0x00DE.

### Senden des Pakets „Steuereinheit → Smart Home“

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



Antwort der Steuereinheit:

- Parameter 0x0001 entspricht 0x00.
- Parameter 0x0002 entspricht 0x03.
- Prüfsumme: 0x00E6.

**PARAMETERTABELLE**
**Funktionen:**
**R** – 0x01

**INC** – 0x04

**RW** – 0x03

**W** – 0x02

**DEC** – 0x05

Parameter- Nummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
1/0x0001	R/W/RW	Anlage Ein/Aus	0: Aus 1: Ein 2: invertieren	1
2/0x0002	R/W/RW/INC/DEC	Betrieb der Lüftungsstufe	1: Lüftungsstufe 1 2: Lüftungsstufe 2 3: Lüftungsstufe 3 4: Lüftungsstufe 4 5: Lüftungsstufe 5	1
3/0x0003	R/W/RW/INC/DEC	Maximale Nummer der Lüftungsstufe	3, 5	1
6/0x0006	R	Boost-Betrieb. Die Anlage wechselt in die Boost-Lüftungsstufe für die Dauer der Nachlaufzeit des Boost-Betriebs (siehe Parameter 70, 71, 102).	0: Aus 1: Ein 2: invertieren	1
7/0x0007	R/W/RW	Timer Ein/Aus	0: Aus 1: Ein 2: invertieren	1
8/0x0008	R/W/RW/INC/DEC	Timer-Betrieb	0: standby 1: Lüftungsstufe 1 2: Lüftungsstufe 2 3: Lüftungsstufe 3 4: Lüftungsstufe 4 5: Lüftungsstufe 5	1
9/0x0009	R/W/RW/INC/DEC	Minuten-Sollwert des Timers	0...59 Minuten	1
10/0x000A	R/W/RW/INC/DEC	Stunden-Sollwert des Timers	0...23 Stunden	1
11/0x000B	R	Aktueller Countdown-Zeit des Timers	Byte 1: Sekunden (0...59) Byte 2: Minuten (0...59) Byte 3: Stunden (0...23)	3
13/0x000D	R/W/RW/INC/DEC	Raumtemperatur-Sollwert im Timer-Betrieb	0: nur Lüftung, +15...+30 °C	1
20/0x0014	R/W/RW	Steuerung mit einem BOOST-Schalter	0: Aus 1: Ein 2: invertieren	1
21/0x0015	R/W/RW	Steuerung über Brandmelder	0: Aus 1: Ein 2: invertieren	1
24/0x0018	R/W/RW/INC/DEC	Raumtemperatur-Sollwert im Standardbetrieb	+15...+30 °C	1
29/0x001D	R/W/RW/INC/DEC	Auswahl eines Temperatursensors zur Regelung der Raumtemperatur	0: im Abluftkanal (ExAirIn) 1: externer Sensor im Bedienfeld (Ext) 2: im Zulufkanal (SuAirOut)	1
30/0x001E	R	Aktuelle Temperatur, mit der die Raumtemperatur geregelt wird	-32768: der Sensor fehlt +32767: Kurzschluss	signed 2 (muss durch 10 dividiert werden, ein Zeichen nach dem Komma)

Parameter- Nummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
31/0x001F	R	Aktuelle Zulufttemperatur am Anlageingang	-32768: der Sensor fehlt +32767: Kurzschluss	signed 2 (muss durch 10 dividiert werden, ein Zeichen nach dem Komma)
32/0x0020	R	Aktuelle Zulufttemperatur am Anlageausgang (nach dem Wärmetauscher/nach dem Nachheizregister)	-32768: der Sensor fehlt +32767: Kurzschluss	signed 2 (muss durch 10 dividiert werden, ein Zeichen nach dem Komma)
33/0x0021	R	Aktuelle Ablufttemperatur am Anlageingang	-32768: der Sensor fehlt +32767: Kurzschluss	signed 2 (muss durch 10 dividiert werden, ein Zeichen nach dem Komma)
34/0x0022	R	Aktuelle Ablufttemperatur am Anlageausgang	-32768: der Sensor fehlt +32767: Kurzschluss	signed 2 (muss durch 10 dividiert werden, ein Zeichen nach dem Komma)
50/0x0032	R	Aktueller Status des Boost-Schalters	0: Aus 1: Ein	1
51/0x0033	R	Aktueller Status des Brandmelders	0: Aus 1: Ein	1
54/0x0036	R/W/RW/INC/DEC	Minimale Lüftungsstufe des Ventilators	0...100 %	1
55/0x0037	R/W/RW/INC/DEC	Minimale Lüftungsstufe des Ventilators	0...100 %	1
58/0x003A	R/W/RW/INC/DEC	Lüftungsstufe des Zuluftventilators im Betrieb der ersten Lüftungsstufe	min...max %	1
59/0x003B	R/W/RW/INC/DEC	Lüftungsstufe des Abluftventilators im Betrieb der ersten Lüftungsstufe	min ... max %	1
60/0x003C	R/W/RW/INC/DEC	Lüftungsstufe des Zuluftventilators im Betrieb der zweiten Lüftungsstufe	min ... max %	1
61/0x003D	R/W/RW/INC/DEC	Lüftungsstufe des Abluftventilators im Betrieb der zweiten Lüftungsstufe	min ... max %	1
62/0x003E	R/W/RW/INC/DEC	Lüftungsstufe des Zuluftventilators im Betrieb der dritten Lüftungsstufe	min ... max %	1
63/0x003F	R/W/RW/INC/DEC	Lüftungsstufe des Abluftventilators im Betrieb der dritten Lüftungsstufe	min ... max %	1
64/0x0040	R/W/RW/INC/DEC	Lüftungsstufe des Zuluftventilators im Betrieb der vierten Lüftungsstufe	min ... max %	1
65/0x0041	R/W/RW/INC/DEC	Lüftungsstufe des Abluftventilators im Betrieb der vierten Lüftungsstufe	min ... max %	1
66/0x0042	R/W/RW/INC/DEC	Lüftungsstufe des Zuluftventilators im Betrieb der fünften Lüftungsstufe	min ... max %	1
67/0x0043	R/W/RW/INC/DEC	Lüftungsstufe des Abluftventilators im Betrieb der fünften Lüftungsstufe	min ... max %	1
69/0x0045	R/W/RW/INC/DEC	Lüftungsstufe der Ventilatoren beim Spülen von Elektro-Heizregistern	min ... max %	1



Parameter- Nummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
70/0x0046	R/W/RW/INC/DEC	Lüftungsstufe des Zuluftventilators im BOOST	min ... max %	1
71/0x0047	R/W/RW/INC/DEC	Lüftungsstufe des Abluftventilators im BOOST	min ... max %	1
96/0x0060	R/W/RW/INC/DEC	Typ des Nachheizregisters	0: ausschalten 1: Elektro- (Festwert)	1
99/0x0063	R/W/RW/INC/DEC	Zeiteinstellung des Filterwechsel-Timers	0, 70...365 Tage im Abstand von 5 Tagen	2
100/0x0064	R	Countdown des Timers bis zum Filterwechsel	Byte 1: Minuten (0...59) Byte 2: Stunden (0...23) Byte 3: Tagen (0...365)	4
101/0x0065	W	Countdown des Timers bis zum Filterwechsel zurücksetzen	Jedes Byte	1
102/0x0066	R/W/RW/INC/DEC	Sollwert der Nachlaufzeit des Boostbetriebs	0 ... 60 min	1
103/0x0067	R/W/RW/INC/DEC	Sollwert der Einschaltverzögerung des Boostbetriebs	0 ... 15 min	1
104/0x0068	R/W/RW	Temperaturregelung im Standardbetrieb	0: Aus 1: Ein 2: invertieren	1
106/0x006A	R	Temperatur TE5	-32768: der Sensor fehlt +32767: Kurzschluss	signed 2 (muss durch 10 dividiert werden, ein Zeichen nach dem Komma)
111/0x006F	R/W/RW	RTC-Zeit	Byte 1: RTC-Sekunden Byte 2: RTC-Minuten Byte 3: RTC-Stunden	3
112/0x0070	R/W/RW	RTC-Kalender	Byte 1: RTC-Datum Byte 2: RTC-Wochentag Byte 3: RTC-Monat Byte 4: RTC-Jahr	4
114/0x0072	R/W/RW	Zeitgesteuerter Betrieb	0: Aus 1: Ein 2: invertieren	1
115/0x0073	R	Lüftungsstufe des zeitgesteuerten Betriebs	0: standby 1: Lüftungsstufe 1 2: Lüftungsstufe 2 3: Lüftungsstufe 3 4: Lüftungsstufe 4 5: Lüftungsstufe 5	1
116/0x0074	R	Temperatureinstellung des zeitgesteuerten Betriebs	0: nur Lüftung, +15...+30 °C	1

Parameter- Nummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
119/0x0077	R/W/RW	Zeitplan-Einstellung	Byte 1: 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) Byte 2: Periodennummer: 1...4 Byte 3: Nummer der Lüftungsstufe: 0: standby 1...5 Byte 4: Temperatur 0: nur Lüftung, +15...+30 °C Byte 5: Minuten bis zum Ende des Abstands: 0...59 Byte 6: Stunden bis zum Ende des Abstands: 0...23	6
124/0x007C	R	Suche der Anlagen im lokalen Netzwerk Ethernet	Text ("0...9", "A...F")	16
125/0x007D	R/W/RW	Passwort der Anlage für das Netzwerk Ethernet	Text ("0...9", "a...z", "A...Z")	0-8
126/0x007E	R	Betriebsstunden	Byte 1: Minuten (0...59) Byte 2: Stunden (0...23) Byte 3 und 4: Tage (0...65535)	4
127/0x007F	R	Liste der aktuellen Alarme/ Warnungen	Byte 1: Code Byte 2: Typ: 1: Alarm 2: Warnung	0,2,4...
128/0x0080	W	Alarme zurücksetzen	Jedes Byte	1
129/0x0081	R	Zustand des Heizregisters	0: Aus 1: Ein	1
131/0x0083	R	Alarm-/Warnungsanzeige	0: keine Alarme 1: Alarm (höchste Priorität) 2: Warnung	1
133/0x0085	R/W/RW	Steuerung über Cloud-Server	0: Aus 1: Ein 2: invertieren	1
134/0x0086	R	Version und Datum der Basis- Firmware der Steuereinheit	Byte 1: Firmware-Version (major) Byte 2: Firmware-Version (minor) Byte 3: Tag Byte 4: Monat Byte 5 und 6: Jahr	6

Parameter-Nummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
135/0x0087	W	Werkseinstellungen zurücksetzen	Jedes Byte	1
136/0x0088	R	Filterstatus	0: sauber 3: Auslösen des Filterwechsel-Timers	1
147/0x0093	R	Vorhandensein eines WLAN-Moduls auf der Platine	0: gibt es nicht 1: es gibt	1
148/0x0094	R/W/RW	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	WLAN-Frequenzkanal	1 ... 13	1
155/0x009B	R/W/RW	WLAN-Modul DHCP	0: STATIC 1: DHCP 2: invertieren	1
156/0x009C	R/W/RW	Zugewiesene IP-Adresse des WLAN-Moduls	Byte 1: 0...255, Byte 2: 0...255, Byte 3: 0...255, Byte 4: 0...255	4
157/0x009D	R/W/RW	Subnetzmaske des WLAN-Moduls	Byte 1: 0...255, Byte 2: 0...255, Byte 3: 0...255, Byte 4: 0...255	4
158/0x009E	R/W/RW	Haupt-Gateway des WLAN-Moduls	Byte 1: 0...255, Byte 2: 0...255, Byte 3: 0...255, Byte 4: 0...255	4
159/0x009F	R/W/RW	DNS-Serveradresse für WLAN-Modul	Byte 1: 0...255, Byte 2: 0...255, Byte 3: 0...255, Byte 4: 0...255	4
160/0x00A0	W	Neue WLAN-Parameter übernehmen und Einstellungsmodus des WLAN-Moduls verlassen	Jedes Byte	1
161/0x00A1	R	Verbindungsstatus des WLAN-Moduls zum Zugangspunkt des Routers	0: nicht angeschlossen 1: angeschlossen	1
162/0x00A2	W	Einstellungsmodus des WLAN-Moduls ohne Verwendung der neuen Parameter verlassen	Jedes Byte	1
163/0x00A3	R	Aktuelle IP-Adresse des WLAN-Moduls	Byte 1: 0...255, Byte 2: 0...255, Byte 3: 0...255, Byte 4: 0...255	4
182/0x00B6	R	Spülzustand des Elektro-Heizregisters (Vorheizung, Nachheizung)	0: Aus 1: Ein	1
185/0x00B9	R	Anlagentyp	0x0002	2

Parameter- Nummer, Dez./Hex.	Funktionen	Beschreibung	Mögliche Werte	Größe, Bytes
240/0x00F0	R/W/RW/INC/DEC	Umluftklappe	0: Rückgewinnung ausschalten 1: Rückgewinnung einschalten (nur für Anlagen mit Rückgewinnung)	1
252/0x00FC		Spezialbefehl		
253/0x00FD		Spezialbefehl		
254/0x00FE		Spezialbefehl		
255/0x00FF		Spezialbefehl		
273/0x0111	R	Typ des Steuergeräts		2
274/0x0112	R	Version und Datum der Basis-Firmware des Bedienfeldes	Byte 1: Firmware-Version (major) Byte 2: Firmware-Version (minor) Byte 3: Tag Byte 4: Monat Byte 5 und 6: Jahr	6
1024/0x0400	R/W/RW	Helligkeit-Sollwert der Tastenbeleuchtung	0..80 (20-100%)	1
1025/0x0401	R/W/RW	Ein-/Ausschalten des Schallgebers auf der Platine	0: Aus 1: Ein	1
1026/0x0402	R/W/RW	Auswahl des Beleuchtungsbetriebs	0: statischer Betrieb 1: dynamischer Betrieb	1

## 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-Nummer
            {
                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;
}
//*****//
}
}
}
```



