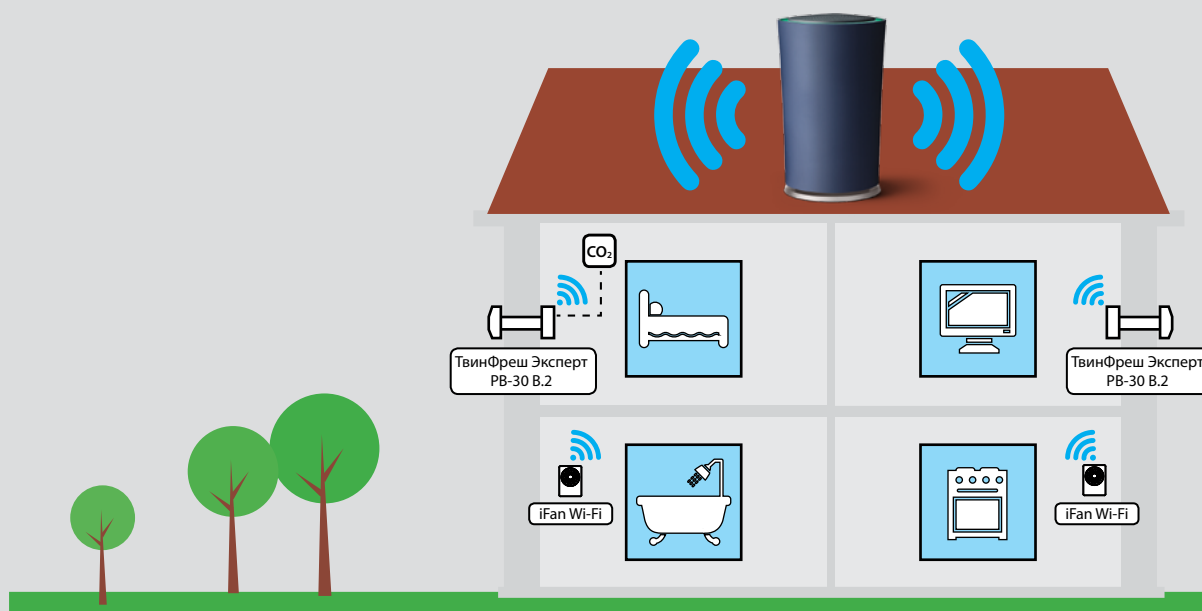


Smart House



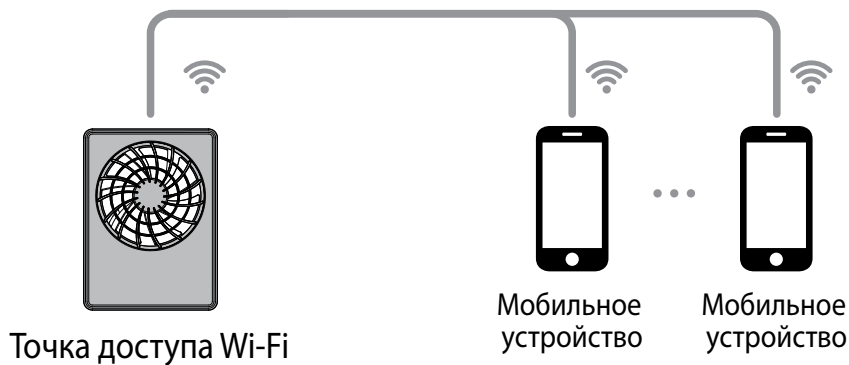
Подключение к системе «Умный дом»

СОДЕРЖАНИЕ

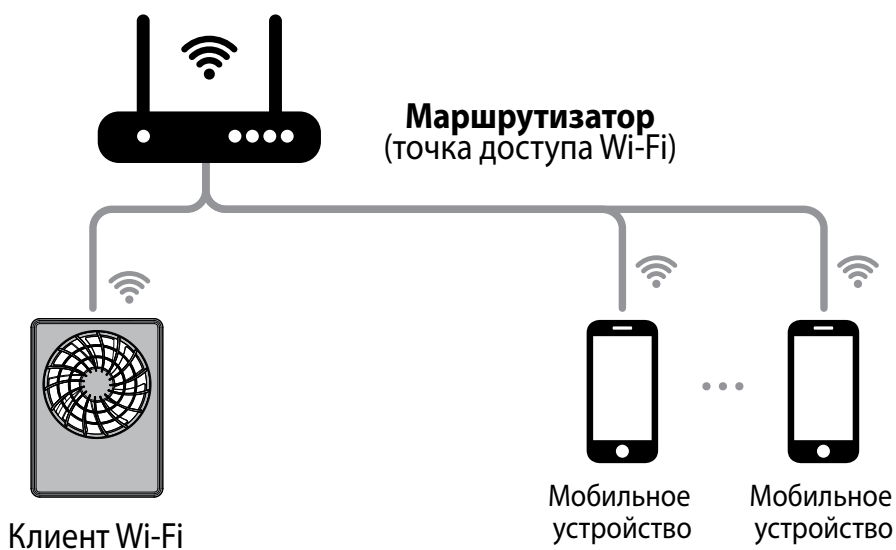
Назначение	2
Параметры сети	3
Структура пакета	4
Примеры использования специальных команд в блоке DATA	5
Примеры полного пакета	6
Таблица параметров	7
Пример обработки пакетов на языке C	10

ПОДКЛЮЧЕНИЕ И НАСТРОЙКА

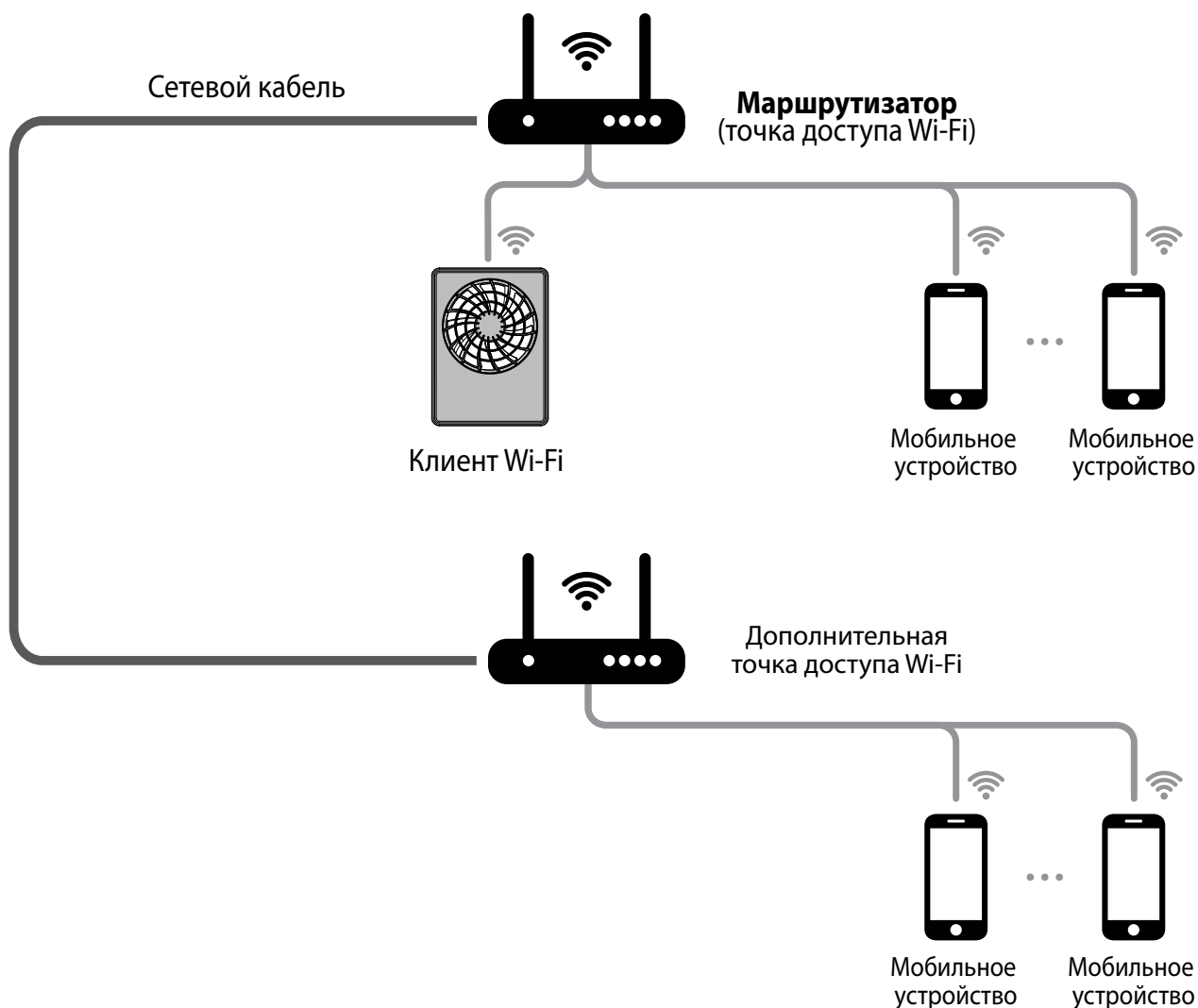
Пример 1: схема прямого подключения вентилятора к системе BMS «Умный дом» без использования маршрутизатора. Настройте вентилятор на работу Wi-Fi в режиме точки доступа (см. руководство пользователя к вентилятору).
Примечание: максимально возможное количество подключаемых устройств управления – восемь.



Пример 2: схема подключения с использованием маршрутизатора с одной точкой доступа Wi-Fi. Вентилятор, телефоны и система BMS «Умный дом» подключаются к Wi-Fi точке доступа маршрутизатора.



Пример 3: схема подключения системы BMS «Умный дом» с использованием маршрутизатора, к которому подключено несколько точек доступа Wi-Fi.



ПАРАМЕТРЫ СЕТИ

Обмен данными производится по транспортному протоколу UDP (поддерживается широковещание).

IP-адрес ведущего устройства:

- 192.168.4.1 – когда ведущее устройство работает без маршрутизатора (схема подключения №1);
- в случае подключения ведущего устройства к маршрутизатору (схема подключения №2) IP-адрес настраивается с помощью мобильного приложения (см. паспорт на изделие) и может быть задан статическим или динамическим (DHCP).

Порт ведущего устройства – 4000.

Максимальный размер пакета – 256 байт.

СТРУКТУРА ПАКЕТА

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

0xFD **0xFD** – признак начала пакета (2 байта).

TYPE – тип протокола (1 байт). Значение = 0x02.

SIZE ID – размер блока **ID** (1 байт). Значение = 0x10.

ID – ID-номер контроллера. Этот номер находится на наклейке (представлен в виде 16 char-символов), которая клеится на плату управления или на корпус изделия.

Также можно использовать в качестве ID-номера кодовое слово DEFAULT_DEVICEID. Его можно применить:

- для управления, если ведущее устройство работает без маршрутизатора (схема подключения №1);
- для поиска ведущих устройств в сети, если используется маршрутизатор (схема подключения №2); при этом устройство будет отвечать только на два параметра: 0x007C и 0x00B9 (см. таблицу параметров).

SIZE PWD – размер блока **PWD** (1 байт). Возможные значения: от 0x00 до 0x08.

PWD – пароль устройства (допустимые символы: "0...9", "a...z", "A...Z"). Пароль по умолчанию – 1111.
Этот пароль можно изменить с помощью мобильного приложения в меню **Подключение → Дома → Настройки** (см. паспорт на изделие).

FUNC – номер функции (1 байт). Определяет действие с данными и структуру блока **DATA**:

- 0x01 – чтение параметров;
- 0x02 – запись параметров. Контроллер не отправляет ответ о состоянии указанных параметров;
- 0x03 – запись параметров с последующим ответом контроллера о состоянии указанных параметров;
- 0x04 – инкремент параметров с последующим ответом контроллера о состоянии указанных параметров;
- 0x05 – декремент параметров с последующим ответом контроллера о состоянии указанных параметров;
- 0x06 – ответ контроллера на запрос (FUNC = 0x01, 0x03, 0x04, 0x05).

DATA – блок данных. Состоит из номеров параметров и их значений:

если FUNC = 0x01 или 0x04 или 0x05:

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

если FUNC = 0x02 или 0x03 или 0x06:

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

Номера параметров (см. таблицу параметров) условно состоят из двух байт (старший байт виртуальный). По умолчанию старший байт каждого номера параметра в каждом новом пакете равен 0x00. Старший байт можно изменить в пределах одного пакета с помощью специальной команды **0xFF** (см. ниже).

P – младший байт номера параметра. Возможные значения: 0x00 – 0xFB. Значения 0xFC – 0xFF являются *специальными командами*:

0xFC – изменить номер функции (**FUNC**). Следующий байт должен быть новым номером функции от 0x01 до 0x05. Используется, чтобы организовать в одном пакете несколько функций с разными действиями;

0xFD – параметр не поддерживается контроллером. Следующий байт – младший байт неподдерживаемого параметра. Используется при ответе контроллера (**FUNC** = 0x06) на запрос чтения или записи несуществующего параметра;

0xFE – изменить размер значения параметра **Value** для одного следующего параметра. Следующим байтом должен быть новый размер параметра, за ним – младший байт номера параметра, а далее – само значение **Value**;

0xFF – изменить старший байт для номеров параметров в пределах одного пакета. Следующим байтом должен быть новый старший байт.

Value – значение параметра (по умолчанию – 1 байт). Следование байт от младшего к старшему.

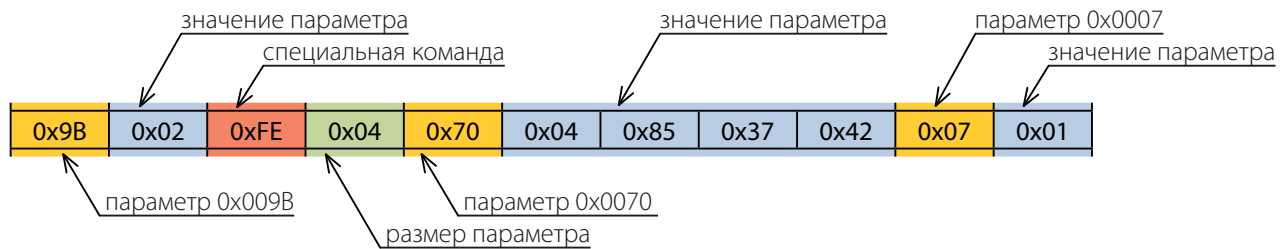
Chksum L **Chksum H** – контрольная сумма (2 байта). Она вычисляется как сумма байт, начиная с байта **TYPE** и заканчивая последним байтом блока **DATA**.

Chksum L – младший байт контрольной суммы.

Chksum H – старший байт контрольной суммы.

ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ СПЕЦИАЛЬНЫХ КОМАНД В БЛОКЕ DATA

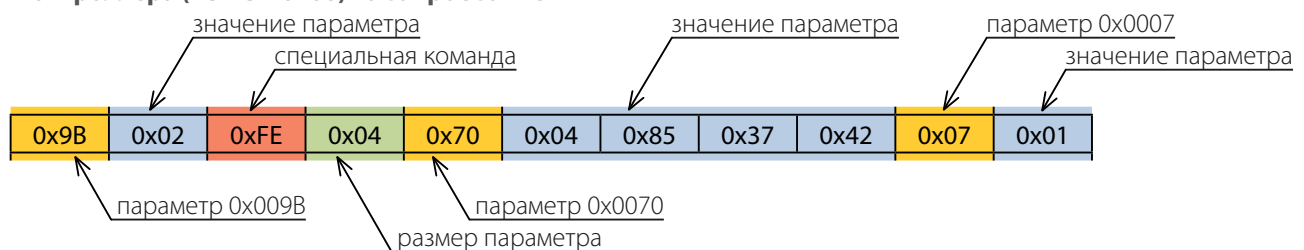
Запрос на запись (FUNC = 0x03) параметров номер 0x009B, 0x0070, 0x0007



В запросе на запись следующее:

- Параметру 0x009B присвоить значение 0x02.
- Параметру 0x0070 присвоить значение 0x42378504. Размер значения – 4 байта, на это указывает специальная команда 0xFE + 0x04.
- Параметру 0x0007 присвоить значение 0x01.

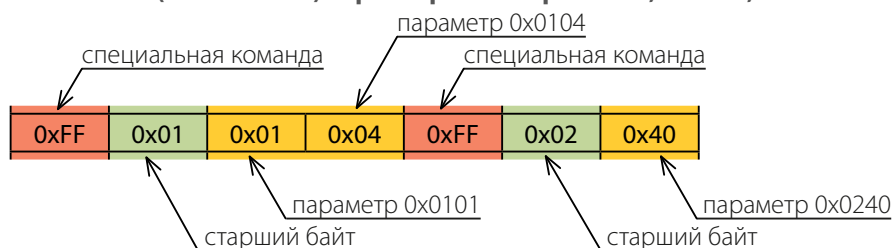
Ответ контроллера (FUNC = 0x06) на запрос записи



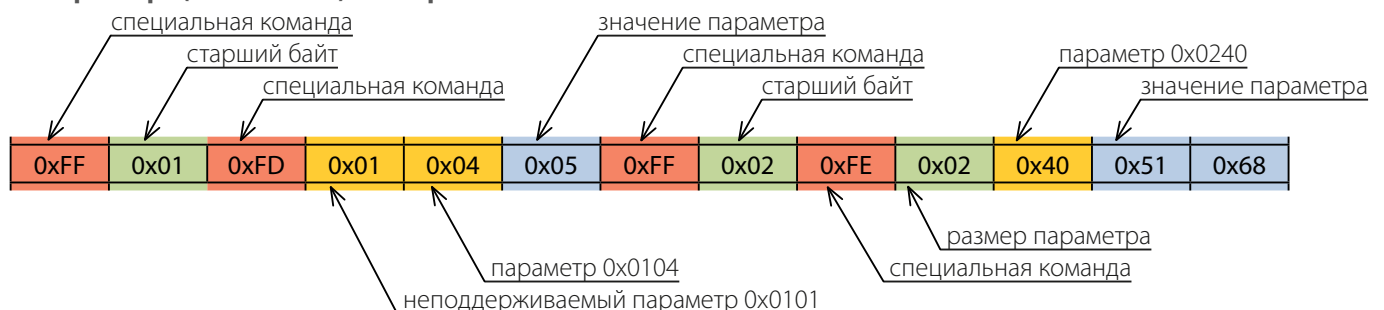
В ответе контроллера следующее:

- Параметр 0x009B имеет значение 0x02.
- Параметр 0x0070 имеет значение 0x42378504. Размер значения – 4 байта, на это указывает специальная команда 0xFE + 0x04.
- Параметр 0x0007 имеет значение 0x01.

Запрос на чтение (FUNC = 0x01) параметров номер 0x0101, 0x0104, 0x0240



Ответ контроллера (FUNC = 0x06) на запрос чтения



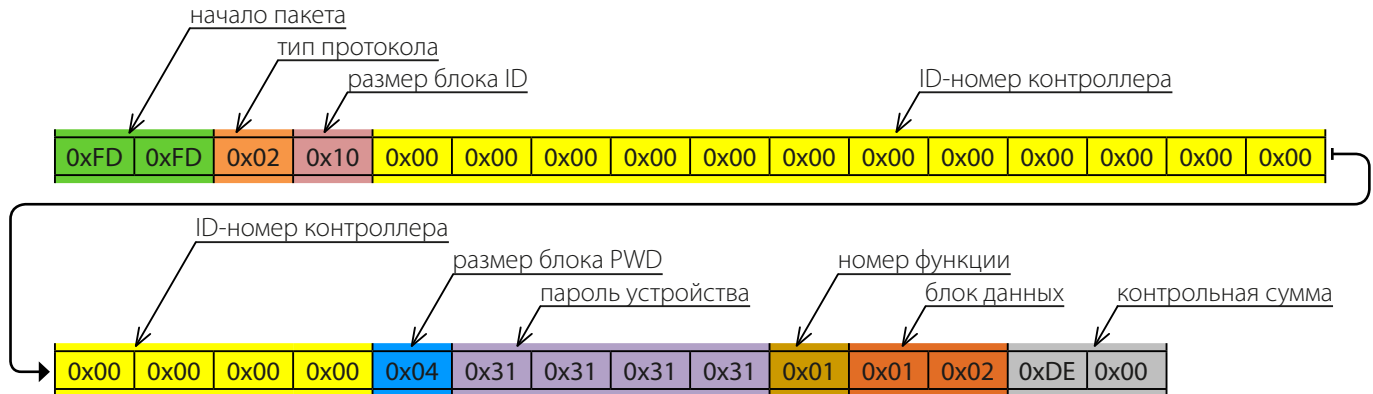
В ответе контроллера следующее:

- Параметр 0x0101 не поддерживается контроллером, на это указывает специальная команда 0xFD.
- Параметр 0x0104 имеет значение 0x05.
- Параметр 0x0240 имеет значение 0x6851. Размер значения – 2 байта, на это указывает специальная команда 0xFE + 0x02.

ПРИМЕРЫ ПОЛНОГО ПАКЕТА

Отправка пакета «Умный дом → Контроллер»

В этом пакете запрос на чтение (FUNC = 0x01) параметров номер: 0x0001, 0x0002.

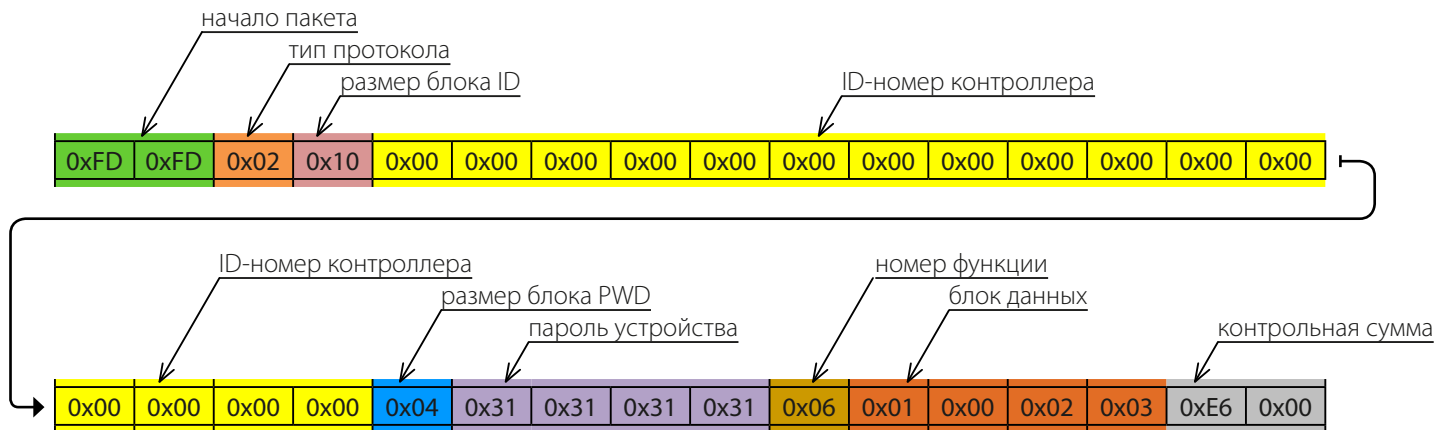


В запросе:

- Контрольная сумма: 0x00DE.

Отправка пакета «Контроллер → Умный дом»

В этом пакете ответ контроллера (FUNC = 0x06) на запрос чтения.



В ответе контроллера:

- Параметр 0x0001 имеет значение 0x00.
- Параметр 0x0002 имеет значение 0x03.
- Контрольная сумма: 0x00E6.

ТАБЛИЦА ПАРАМЕТРОВ
Функции:
R – 0x01

INC – 0x04

RW – 0x03

W – 0x02

DEC – 0x05

Номер параметра, Дес./Hex.	Функции	Описание	Возможные значения	Размер, байт
1/0x0001	R/W/RW	Вкл/выкл вентилятор	0 — выкл 1 — вкл 2 — инвертировать	1
2/0x0002	R	Состояние батарейки	0 — разряжена (отсутствует) 1 — заряд в норме	1
3/0x0003	R/W/RW	Выбор режима 24 часа	0 — выкл 1 — вкл 2 — инвертировать	1
4/0x0004	R	Текущее значение скорости вентилятора (об/мин)	0...6000 RPM	2
5/0x0005	R/W/RW	Вкл/выкл режим BOOST	0 — выкл 1 — вкл 2 — инвертировать	1
6/0x0006	R	Текущее значение времени отсчета таймера BOOST в секундах	0...86400 секунд	3
7/0x0007	R	Текущий статус встроенного таймера	0 — выкл 1 — вкл	1
8/0x0008	R	Текущий статус работы вентилятора по датчику влажности	0 — выкл 1 — вкл	1
10/0x000A	R	Текущий статус работы вентилятора по датчику температуры	0 — выкл 1 — вкл	1
11/0x000B	R	Текущий статус работы вентилятора по датчику движения	0 — выкл 1 — вкл	1
12/0x000C	R	Текущий статус работы вентилятора по сигналу от внешнего выключателя	0 — выкл 1 — вкл	1
13/0x000D	R	Текущий статус работы вентилятора в режиме интервального проветривания	0 — выкл 1 — вкл	1
14/0x000E	R	Текущий статус работы вентилятора в режиме SILENT	0 — выкл 1 — вкл	1
15/0x000F	R/W/RW	Разрешение работы по датчику влажности	0 — выкл 1 — в автоматическом режиме 2 — в ручном режиме	1
17/0x0011	R/W/RW	Разрешение работы по датчику температуры	0 — выкл 1 — вкл 2 — инвертировать	1
18/0x0012	R/W/RW	Разрешение работы по датчику движения	0 — выкл 1 — вкл 2 — инвертировать	1
19/0x0013	R/W/RW	Разрешение работы по сигналу от внешнего выключателя	0 — выкл 1 — вкл 2 — инвертировать	1
24/0x0018	R/W/RW/INC/DEC	Уставка скорости Max	30...100 %	1
26/0x001A	R/W/RW/INC/DEC	Уставка скорости Silent	30...100 %	1

Номер параметра, Дес./Гекс.	Функции	Описание	Возможные значения	Размер, байт
27/0x001B	R/W/RW/INC/DEC	Уставка скорости интервального проветривания	30...100 %	1
29/0x001D	R/W/RW	Активация режима интервального проветривания	0 — выкл 1 — вкл 2 — инвертировать	1
30/0x001E	R/W/RW	Активация режима Silent Mode	0 — выкл 1 — вкл 2 — инвертировать	1
31/0x001F	R/W/RW	Время старта работы режима Silent Mode в секундах	0...86400 секунд	3
32/0x0020	R/W/RW	Время окончания работы режима Silent Mode в секундах	0...86400 секунд	3
33/0x0021	R/W/RW	Текущее время внутренних часов вентилятора в секундах	0...86400 секунд	3
35/0x0023	R/W/RW/INC/DEC	Уставка таймера задержки выключения/BOOST	0 — выкл 2 — 5 минут 3 — 15 минут 4 — 30 минут 6 — 60 минут	1
36/0x0024	R/W/RW/INC/DEC	Уставка таймера задержки включения	0 — выкл 1 — 2 минуты 2 — 5 минут	1
37/0x0025	W	Сброс параметров к заводским значениям	Любой байт	1
124/0x007C	R	Поиск устройств в локальной сети Ethernet	Текст ("0...9", "A...F")	16
134/0x0086	R	Версия и дата основной прошивки контроллера	1-й байт — версия прошивки (major) 2-й байт — версия прошивки (minor) 3-й байт — день 4-й байт — месяц 5-й, 6-й байт — год	6
148/0x0094	R/W/RW	Режим работы Wi-Fi	1 — client 2 — access point	1
149/0x0095	R/W/RW	Имя Wi-Fi в режиме Client	Текст	1 ... 32
150/0x0096	R/W/RW	Пароль Wi-Fi	Текст	8 ... 64
153/0x0099	R/W/RW	Тип шифрования данных Wi-Fi	48 — OPEN 50 — WPA_PSK 51 — WPA2_PSK 52 — WPA_WPA2_PSK	1
154/0x009A	R/W/RW	Частотный канал Wi-Fi	1...13	1
155/0x009B	R/W/RW	DHCP Wi-Fi модуля	0 — STATIC 1 — DHCP 2 — инвертировать	1
156/0x009C	R/W/RW	Заданный IP-адрес Wi-Fi модуля	1-й байт — 0...255 2-й байт — 0...255 3-й байт — 0...255 4-й байт — 0...255	4

Номер параметра, Дес./Hex.	Функции	Описание	Возможные значения	Размер, байт
157/0x009D	R/W/RW	Маска подсети Wi-Fi модуля	1-й байт — 0...255 2-й байт — 0...255 3-й байт — 0...255 4-й байт — 0...255	4
158/0x009E	R/W/RW	Основной шлюз Wi-Fi модуля	1-й байт — 0...255 2-й байт — 0...255 3-й байт — 0...255 4-й байт — 0...255	4
160/0x00A0	W	Применить новые параметры Wi-Fi и выйти из режима настройки Wi-Fi модуля	Любой байт	1
163/0x00A3	R	Текущий IP-адрес Wi-Fi модуля	0...255	4
185/0x00B9	R	Тип устройства		2

ПРИМЕР ОБРАБОТКИ ПАКЕТОВ НА ЯЗЫКЕ C

```
//===== Специальные команды =====//
#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-номер контроллера

//***** Проверка контрольной суммы и начало пакета *****/
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) // Контрольная сумма
    {
        if(receive_data[2] == 0x02) // Тип протокола
        {
            if(memcmp(&receive_data[4], current_id, receive_data[3]) == 0) // ID-номер
            {
                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]; // Место в массиве, где начинается блок FUNC
                //***** FUNC и DATA *****/
                for(; r_pos < receive_data_size - 2; r_pos++)
                {
                    //===== Специальные команды =====//
                    param_size = 1;
                    //=== новый номер функции
                    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) // если номер функции не поддерживается
                            break;
                        continue;
                    }
                    //=== новое значение старшего байта для номеров параметров
                    else if(receive_data[r_pos] == BGCP_CMD_PAGE)
                    {

```

```
        page = receive_data[++r_pos];
        continue;
    }
    //== новое значение размера параметра
    else if(receive_data[r_pos] == BGCP_CMD_SIZE)
    {
        param_size = receive_data[++r_pos];
        r_pos++;
    }
    //== если параметр не поддерживается
    else if(receive_data[r_pos] == BGCP_CMD_NOT_SUP)
    {
        r_pos++;
        //***** обработка неподдерживаемых параметров *****//
        param = (uint16_t)(page << 8) | (uint16_t)(receive_data[r_pos]);
        switch(param)
        {
            case 0x0001:
                break;
            case 0x0002:
                break;
            ...
        }
        //*****//
        continue;
    }
    jump_size = param_size;
    //=====//

    //***** обработка поддерживаемых параметров *****//
    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;
}
//*****//
}
}
}
```

