

### 1. Введение.

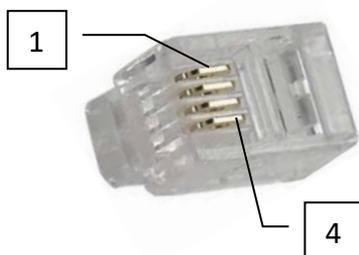
Настоящий документ описывает протокол обмена дополнительных внешних устройств с системой EctoControl. Протокол основан на основе стандарте Modbus RTU (MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1b), <http://www.modbus.org>.

### 2. Описание физического и канального уровней

На физическом уровне используется интерфейс RS-485, полудуплексный режим. В качестве канального уровня используется протокол UART, скорость 19200 бит/с, 8 бит данных, 1 стартовый и 1 стоповый бит, без контроля четности.

Для подключения устройств к системе EctoControl или к разветвителю используется четырехжильный кабель. Каждое устройство имеет в своем составе кабель длиной 1...10м, оснащенный соединителем типа 4P4C (аналогичный устанавливаемому на кабеле трубок проводных телефонных аппаратов). В зависимости от производителя кабеля возможны 2 варианта соответствий цвета жилы сигналу интерфейса:

Номер контакта	Наименование сигнала	Цвет (вариант кабеля №1)	Цвет (вариант кабеля №2)
1	+12V	Желтый	Красный
2	GND	Белый	Белый
3	A	Зеленый	Зеленый
4	B	Коричневый	Синий



По линиям +12V и GND на устройство поступает питание (7...14В постоянного тока). Линии «А» и «В» представляют собой сигнальные линии интерфейса RS-485. Уровень напряжения при формировании сигнала устройством лежит в диапазоне 0...5В, при формировании сигнала внешним устройством – 0...12В.

### 3. Описание транспортного уровня.

Для реализации обмена данными с устройствами используются стандартные функции протокола Modbus:

Read Holding Registers (0x03)

Write Holding Registers (0x10)

Read input registers (0x04)

Каждое устройство на шине является ведомым, для опроса устройств и управления ими используется единое ведущее устройство (система EctoControl).

Условные обозначения:

RSVD – зарезервировано для последующего использования

R – доступ только на чтение (input registers)

RW – доступ на чтение и запись (holding registers)

Каждое устройство содержит однотипную информационную структуру, содержащую общую информацию об устройстве.

Начальный адрес структуры (регистры): 0x0000, holding registers

Длина структуры (регистры): 4

Адрес байта	0	1	2	3	4	5	6	7
Поле	RSVD	UID			RSVD	ADDR	TYPE	CHN_CNT
Регистр	0		1		2		3	
Доступ	R		R		RW		R	
Адрес регистра	0x0000		0x0001		0x0002		0x0003	

**UID** – уникальный идентификатор устройства (длина 3 байта), диапазон значений: [ 0x800000; 0xFFFFF ].

**ADDR** – логический адрес устройства на шине ModBus, по умолчанию 0xFO (может быть принудительно записан мастером с помощью широковещательного пакета). Имеет диапазон 0x01...0x20.

**TYPE** – тип устройства, определяющий его физические свойства:

Измерительные датчики:

0x22 – датчик температуры

0x23 – датчик влажности воздуха

Контактные датчики:

0x50 – универсальный контактный датчик

Устройства управления:

0xC0 – блок управления реле двухканальный

0xC1 – блок управления реле десятиканальный

0x11 – адаптер OpenTherm

**CHN\_CNT** – число каналов устройства (1...10).

Для расчета CRC используется полином 0xA001 при начальном значении CRC равном 0xFFFF. Примеры программ для расчёта см. здесь: [https://www.modbustools.com/modbus\\_crc16.html](https://www.modbustools.com/modbus_crc16.html)

#### 4. Структура данных измерительных датчиков

Начальный адрес структуры (регистры): 0x0020, input registers

Длина структуры (регистры): CHN\_CNT (1...10)

Байт	0:MSB	1:LSB	2:MSB	3:LSB	...	CHN_CNT-2:MSB	CHN_CNT-1:LSB
Поле	CHN 0		CHN 1		...	CHN 9	
Регистр	0		1		...	CHN_CNT-1	
Доступ	R		R		R	R	
Адрес	0x0020		0x0021		...	0x0020+CHN_CNT-1	

**CHN0...CHN9** – данные датчика для каналов 1...10. Данные действительны только для того количества доступных каналов, которое указано в поле CHN\_CNT информационной структуры.

CHN N – данные измерения. Для датчиков температуры (TYPE=0x22) представляет собой знаковое целое шестнадцатиразрядное число, MSB – старший байт, LSB – младший байт. Диапазон возможных значений: -32768...+32767. Показания температуры указываются как число -400...+990, что соответствует -40...+99 град. С в десятых долях градуса. Например, значение 0x123 будет соответствовать 29,1C (в десятичной системе 0x123 = 291, т.е. 29,1C). Для датчика влажности воздуха (TYPE=0x23) возможные значения будут в диапазоне 0...1000, что отражает показание относительной влажности в десятых долях процента. Например, данные 0x381 будут соответствовать 89,7% (0x381 = 897, т.е. 89,7%).

#### 5. Структура данных контактных датчиков

Начальный адрес структуры (регистры): 0x0010, input registers

Длина структуры (регистры): CHN\_CNT (1...10)/8, но не менее 1 регистра.

Байт	0:MSB	1:LSB	2:MSB	3:LSB	...	CHN_CNT-2:MSB	CHN_CNT-1:LSB
Поле	CHN 0...7		CHN 8...15		...		
Регистр	0		1		...	CHN_CNT-1	
Доступ	R		R		R	R	
Адрес	0x0020		0x0021		...	0x0020+CHN_CNT-1	

Номер регистра соответствует целочисленному делению номера канала на 16:

**REG\_NO = CHN\_NO/16**

Номер байта соответствует целочисленному делению номера канала на 8:

**BYTE\_NO = CHN\_NO/8**

Номер бита в каждом байте соответствует номеру канала по остатку от деления на 8:

**BIT\_NO = CHN\_NO%8**

Установленный бит означает тревогу датчика, сброшенный – норму.

#### 6. Устройства управления

Устройства такого рода имеют 2 типа структур: битовые маски каналов и массив таймеров.

Битовая маска каналов:

Начальный адрес структуры (регистры): 0x0010, input registers

Длина структуры (регистры): CHN\_CNT (1...10)/8, но не менее 1 регистра.

Байт	0	1	2	3	...
Поле	CHN 0-7	CHN 8-15	CHN 16-23	CHN 24-31	...
Регистр	0		1		...
Доступ	RW		RW		RW
Адрес	0x0010		0x0011		...

Номер регистра соответствует целочисленному делению номера канала на 16:

**REG\_NO = CHN\_NO/16**

Номер байта соответствует целочисленному делению номера канала на 8:

**BYTE\_NO = CHN\_NO/8**

Номер бита в каждом байте соответствует номеру канала по остатку от деления на 8:

**BIT\_NO = CHN\_NO%8**

Установленный бит означает включенное устройство, сброшенный – отключенное.

Массив таймеров:

Начальный адрес структуры (регистры): 0x0020, holding registers

Длина структуры (регистры): CHN\_CNT.

Байт	0:MSB	1:LSB	2:MSB	3:LSB	...
Поле	CHN 0		CHN 1		...
Регистр	0		1		...
RW access	RW		RW		RW
Адрес	0x0020		0x0021		...

CHN *n* – данные времени, в течение которого должно удерживаться заданное состояние устройства (биты 14...0) и начальное состояние устройства (бит 15). Состояние реле применяется только в пакете записи в регистры таймеров, после получения устройством команды значение старшего бита мгновенно применяется к соответствующему выходу и удаляется из регистра, остаток соответствует таймауту, по истечении которого состояние реле будет изменено на противоположное.

Нормальное состояние таймера равно нулю, (таймер считает сверху вниз). Состояние, отличное от нуля, свидетельствует о том, что таймер запущен. Когда значение таймера достигает нуля, выход соответствующего канала инвертируется. Значение соответствующего регистра определяет задержку кратно 500мс. Например, для задержки в 5 секунд необходимо передать значение регистра 0x000A (10 dec). Таким образом в диапазоне 0x0001-0x7FFF возможно формирование интервалов времени от 500мс до 16383.5с (4 часа, 33мин., 3.5с).

## 6. Адаптер OpenTherm

Содержит 2 вида структур: команды (доступна только на запись, кроме регистра 0x0013, который доступен также и для чтения) и состояния (доступна только на чтение).

Структура команд:

Байт	Поле	Номер регистра	Тип доступа	Адрес регистра
0	Биты управления режимами котла: bit 0: Первый контур отопления (1-вкл, 0-откл) bit 1: Контур ГВС (1-вкл, 0-откл) bit 2: Второй контур отопления (1-вкл, 0-откл) bit 3: Режим регулирования температуры (1-ПЗА, 0 - Термостат) bit 4: Сброс блокировки котла, возникшей в результате ошибки его работы. (1 – выполнить сброс, 0 – нет действия). bit 5: Релейный режим (горелка на максимуме) (1-вкл, 0-откл) bit 6: Режим ПЗА с двумя термодатчиками (1-вкл, 0-откл)	0	W	0x0010

	bit 7: Перезагрузка адаптера (1- выполнить перезагрузку, 0-нет действия)			
1	Максимальный уровень модуляции горелки (0...100%)			
2	Уставка температуры ГВС (0...80С, шаг 1С)	1	W	0x0011
3				
4	Номер кривой ПЗА для режима ПЗА (0...15)	2	W	0x0012
5				
6	Уставка температуры контура отопления (0...80С, шаг 1С)	3	W/R	0x0013
7				
8	Уставка максимальной температуры первого контура (0...80С, шаг 1С)	4	W	0x0014
9				
10	Уставка минимальной температуры первого контура (0...80С, шаг 1С)	5	W	0x0015
11				
12	Уставка максимальной температуры ГВС (0...80С, шаг 1С)	6	W	0x0016
13				
14	Уставка минимальной температуры ГВС (0...80С, шаг 1С)	7	W	0x0017
15				
16	Уставка температуры контура отопления для аварийного режима (недоступны термодатчики) (0...80С, шаг 1С). Используется для случая, когда термодатчики воздуха отсутствуют и котлу необходимо ориентироваться на собственный встроенный датчик температуры на подающем трубопроводе.	8	W	0x0018
17				
18	Температура воздуха в отпливаемом помещении (в десятых долях С, значения -400...+1000, что соответствует -40,0...+100,0С)	9	W	0x0019
19				
20	Температура воздуха окружающей среды для режима ПЗА (в десятых долях С, значения -400...+1000, что соответствует -40,0...+100,0С)	10	W	0x001A
21				

Структура данных состояния:

Байт	Поле	Регистр	RW access	Адрес
0	Уникальный код производителя котла	0	R	0x0020
1				
2	bit 0: Состояние горелки (1-вкл, 0 -откл) bit 1: Первый контур отопления (1-вкл, 0 -откл) bit 2: ГВС (1-вкл, 0 -откл) bit 3: Второй контур отопления (1-вкл, 0 -откл) bit 4: Наличие ошибок котла (1-есть, 0 -нет) bit 5: Связь между котлом и адаптером (1-есть, 0 нет).	1	R	0x0021
3				
4	Температура первого контура отопления (0...80С, шаг 0,1С)	2	R	0x0022
5				
6	Температура ГВС (0...80С, шаг 0,1С)	3	R	0x0023
7				

8	Расход ГВС (л/мин или дл/мин, зависит от модели котла, в десятых долях единицы измерения)	4	R	0x0024
9	Давление в контуре отопления (0...5бар, шаг 0,1 бар)			
10	Код ошибки, характерный для всех котлов. 0: Необходимо обслуживание 1: Котел заблокирован 2: Низкое давление в отопительном контуре 3: Ошибка розжига 4: Низкое давление воздуха 5: Перегрев теплоносителя в контуре	5	R	0x0025
11	Код ошибки, характерный для данной модели котла.			
12	Код самодиагностики котла. Характере для конкретной модели и конкретного производителя.	6	R	
13				
14	Текущая уставка температуры контура отопления (0...80С, шаг 0,1С)	7	R	0x0026
15				

## 7. Адаптер LoRa

Адаптер LoRa хранит в себе все данные подключенных радиодатчиков и позволяет системе EctoControl увидеть эти датчики как обычные проводные датчики интерфейса ДОП (RS-485). Таким образом система может видеть до 32 датчика собственной радиосети и еще 32 датчика порта ДОП (под видом которых работают радиодатчики LoRa). Поэтому система видит радиодатчики LoRa только как проводные с портом ДОП и это нисколько не отражается на программах и оповещениях системы.

Программирование датчиков происходит по одному. Происходит это так:

1. Система отправляет широковещательный пакет с функцией **PROG\_READ (0x46)**, ожидая получить данные датчика на шине. Адаптер LoRa использует этот пакет для активации поиска радиодатчиков и не отвечает системе.
2. Адаптер LoRa ожидает программирования одного радиодатчика в течение 60с. Если датчик был запрограммирован, адаптер запоминает его данные и ждет повторного запроса от системы.
3. Система повторно отправляет широковещательный пакет с функцией **PROG\_READ (0x46)**, при этом адаптер, храня данные ранее запрограммированного радиодатчика, ответит функцией **PROG\_READ (0x46)** с адреса **0x7F** (с этого адреса будут отвечать только вновь добавленные датчики). Если новые радиодатчики не были запрограммированы, адаптер не ответит системе.
4. Если адаптер ответил данными нового датчика, система обязана сменить адрес датчику, так как он изначально лежит вне допустимого диапазона (0x00...0x1F). Для этого она отправляет адаптеру пакет с функцией **PROG\_WRITE (0x47)** на адрес **0x7F** с указанием нового адреса датчика в диапазоне 0x00...0x1F. Новый адрес выбирается исходя из уже имеющихся в памяти системы датчиков. Адаптер подтверждает, что новый датчик заменил свой адрес на заданный пакетом с функцией **PROG\_WRITE (0x47)**, высланным с нового адреса.
5. Система обязана запомнить данные датчика с новым адресом и периодически опрашивать его, равно как и опрашивать ранее запрограммированные датчики функцией **Read input registers (0x03)**.
6. Если нужно запрограммировать более одного радиодатчика, п.1 надо повторить столько раз, сколько датчиков необходимо добавить (до 32 шт).

Удаление радиодатчик LoRa из адаптера.

Адаптер хранит в себе данные радиодатчиков, и если в него запрограммированы все 32 датчика и каждый из них периодически опрашивается системой, запрограммировать в адаптер новый датчик будет невозможно. Однако система может перестать опрашивать один или несколько датчиков, и если адаптер не увидит запроса данных какого-либо датчика за период более 150с, он будет считать, что система удалила у себя этот датчик и позволит запрограммировать на его место новый.

Как происходит эмуляция проводных датчиков адаптером LoRa?

1. Если радиодатчик LoRa не на связи, адаптер перестает отвечать на запросы от системы по адресу «пропавшего» датчика через 150с после его ухода со связи. Отсутствующий ответ на запрос система интерпретирует как признак «нет связи» с указанным датчиком.
2. Датчик температуры LoRa при неисправности сенсора и при первом включении может в качестве значения температуры прислать код 0x7E7E. Система может интерпретировать это значение как отсутствие связи с датчиком или его частичное повреждение.

## 8. Функции для программирования устройств в систему

Для задания устройству уникального адреса на шине Modbus используется процедура программирования адреса. Процедура основана на отправке специальной функции PROG\_READ (0x46) и PROG\_WRITE (0x47).

Функция PROG\_READ (0x46) (высылается ведущим устройством единственному устройству на шине с неизвестным адресом).

**Запрос** от ведущего устройства:

Широковещательный адрес	Код функции	CRC_HI	CRC_LO
0x00	0x46	0xXX	0xXX

**Ответ** от ведомого устройства:

Адрес ведомого устройства	Код функции	Адрес устройства	CRC_HI	CRC_LO
0x00	0x46	ADDR	0xXX	0xXX

Устройство возвращает в ответе свой текущий адрес ADDR.

Функция PROG\_WRITE (0x47) (высылается ведущим устройством ведомому с указанием сменить свой имеющийся адрес на заданный).

**Запрос** от ведущего устройства:

Широковещательный адрес	Код функции	Новый адрес	CRC_HI	CRC_LO
0x00	0x47	ADDR	0xXX	0xXX

**Ответ** от ведомого устройства:

SLAVE_ADDR	Код функции	Новый адрес	CRC_HI	CRC_LO
ADDR	0x47	ADDR	0xXX	0xXX

Устройство обновляет свой адрес на ADDR и высылает ответ уже с нового адреса.

ПРИЛОЖЕНИЕ. ПРИМЕРЫ ОБМЕНА ДАННЫМИ С УСТРОЙСТВАМИ

Все данные представлены в шестнадцатеричном виде.

1. Программирование устройства с адресом 01h нового адреса 05h.

Запрос мастера:	00h	46h		
	адрес	функция	CRC старш.	CRC младш.

Ответ устройства:	00h	46h	01h		
	адрес	функция	Адрес устр-ва	CRC старш.	CRC младш.

Запрос мастера:	00h	47h	05h		
	адрес	функция	Адрес устр-ва	CRC старш.	CRC младш.

Ответ устройства:	00h	47h	05h		
	адрес	функция	Адрес устр-ва	CRC старш.	CRC младш.

3. Чтение типа устройства с адресом 01h

Запрос мастера:	01h			
	адрес	функция		

Ответ устройства:	01h				
	адрес	функция			

4. Чтение ID устройства с адресом 10h

Запрос мастера:	10h								
	адрес	функция							

Ответ устройства:	10h								
	адрес	функция							

5. Чтение данных датчика температуры (влажности) с адресом 07h

Запрос мастера:	07h								
	адрес	функция							

Ответ устройства:	07h								
	адрес	функция							

Полученные данные температуры (123h) в десятичном виде выглядит как 291, т.е. 29,1С для датчика влажности и 29,1% для датчика влажности.

1. Включение второго канала реле с адресом 18h

Запрос мастера:	18h								
	адрес	функция							

Ответ устройства:	18h								
	адрес	функция							

1. Включение четвертого канала блока реле на 100 секунд

Запрос мастера:	00h								
	адрес	функция							

Ответ устройства:	00h								
	адрес	функция							