



НПФ Техэнергокомплекс

Устройство  
микропроцессорной  
защиты  
**УМПЗ/ТЭК**

Описание протокола связи

# Содержание

Протокол связи.....	2
Общие сведения.....	2
Организация обмена.....	2
Режим работы.....	3
Содержание поля «адрес».....	3
Содержание поля «код функции».....	3
Содержание поля «данные».....	4
Содержание поля «контрольная сумма».....	4
Описание кодов функций.....	6
Код функции 0x03.....	6
Код функции 0x10.....	7
Код функции 0x20.....	8
Код функции 0x41.....	8
Код функции 0x51.....	8
Код функции 0x52.....	9
Карта памяти устройства.....	9
Ограничение на запись данных.....	10
Ограничения протокола на интерфейсах RS232 и RS485.....	11
Тестирование сторонними программными средствами.....	12

# Описание протокола связи

## Протокол связи.

Протокол связи с устройствами серии УМПЗ аналогичен протоколу Modbus RTU.

Параметр	Значение
Скорость обмена, бод	1200, 2400, 4800, 9600, 19200, 38400, 57600
Бит данных	8
Четность	нет
Стоповых бит	1
Управление потоком	нет

## Общие сведения.

При организации связи с устройством используется принцип «главный — подчинённый». Главным всегда является персональный компьютер, подчиненным — устройство. Главный посылает запрос подчиненному, подчинённый выполняет запрос (передает в ответ данные или выполняет какое-либо действие). Если подчинённый обнаружил ошибку в запросе, то запрос не выполняется.

## Организация обмена.

Обмен организуется циклами запрос – ответ:

Формат запроса:

Адрес устройства	Код функции	Данные	Контрольная сумма
1 байт	1 байт	N байт	2 байта

Формат ответа:

Адрес устройства	Код функции	Данные	Контрольная сумма
1 байт	1 байт	N байт	2 байта

Адрес устройства задаётся в диапазоне от 1 до 255.

Коды функций, поддерживаемые устройством (hex):

0x03 – чтение нескольких регистров;

0x10 – запись нескольких регистров;

0x20 – запись ЭНП уставок, очистка аварийного журнала, сброс;

0x41 – чтение нескольких регистров из ЭНП осциллографа;

0x51 – чтение нескольких регистров из ЭНП уставок;

0x52 – запись нескольких регистров в ЭНП уставок.

Данные в поле запроса и ответа зависят от кода функции.

Контрольная сумма вычисляется по алгоритму CRC-16.

### **Режим работы.**

Протокол связи по формату запросов, ответов и режиму аналогичен широко известному протоколу Modbus в режиме RTU. В режиме RTU сообщение начинается и заканчивается интервалом тишины, равным по длительности 3,5 символам на текущей скорости обмена. Длина запроса не должна превышать 255 байт. В целях экономии ресурсов контроль паузы длительностью 1,5 символа не производится.

### **Содержание поля «адрес».**

При формировании запроса в поле «адрес» указывается адрес устройства, которому адресован запрос. Диапазон адресов 1 — 255. При формировании ответа устройством адрес в ответе повторяет адрес в запросе.

### **Содержание поля «код функции».**

При формировании запроса в поле «код функции» указывается код функции, которую необходимо выполнить устройству. Устройство поддерживает коды функций:

- а) общедоступные 0x03, 0x10;
- б) определенные пользователем 0x41, 0x42, 0x51.

При формировании ответа, в случае успешного выполнения запроса, код функции в ответе повторяет код функции в запросе. В случае, если устройство не может выполнить запрос, то в поле «код функции» старший бит устанавливается в единицу.

Запрос успешно выполнено: код функции запроса 0x03, код функции ответа 0x03.

Запрос не выполнен: код функции запроса 0x03, код функции ответа 0x83.

## Содержание поля «данные».

При формировании запроса в поле «данные» указывается дополнительная информация, которая определена для каждого поля функции, например, счётчик байт, адрес регистра, количество регистров.

При формировании ответа, в случае успешного выполнения запроса, в поле «данные» также указывается дополнительная информация, которая определена для каждого поля функции. Пример успешно выполненного запроса (hex):

Запрос - 01 03 00 00 00 01 84 0A

Ответ - 01 03 02 80 00 D9 84

Если запрос не может быть выполнен, то в поле «данные» указывается код исключения, например (hex):

Запрос - 01 05 00 00 00 01 0C 0A

Ответ - 01 85 01 83 50

Запрос не может быть выполнен (hex) 80|05=85.

Код исключения (hex) 01 (код функции (hex) 05 не поддерживается устройством).

## Содержание поля «контрольная сумма».

Поле «контрольная сумма» содержит 16-ти битовое значение контрольной суммы, которое вычисляется по алгоритму CRC16, над содержимым сообщения.

```
/*
Name   : CRC-16
Poly   : 0x8005      x^16 + x^15 + x^2 + 1
Init   : 0xFFFF
Revert : true
XorOut : 0x0000
Check  : 0x4B37 ("123456789")
MaxLen : 4095 байт (32767 бит) – обнаружение одинарных, двойных, тройных и
всех нечетных ошибок
*/
```

```
const unsigned short Crc16Table[256] = {
0x0000, 0xC0C1, 0xC181, 0x0140, 0xC301, 0x03C0, 0x0280, 0xC241,
0xC601, 0x06C0, 0x0780, 0xC741, 0x0500, 0xC5C1, 0xC481, 0x0440,
0xCC01, 0x0CC0, 0x0D80, 0xCD41, 0x0F00, 0xCFC1, 0xCE81, 0x0E40,
0x0A00, 0xCAC1, 0xCB81, 0x0B40, 0xC901, 0x09C0, 0x0880, 0xC841,
0xD801, 0x18C0, 0x1980, 0xD941, 0x1B00, 0xDBC1, 0xDA81, 0x1A40,
0x1E00, 0xDEC1, 0xDF81, 0x1F40, 0xDD01, 0x1DC0, 0x1C80, 0xDC41,
```

## Описание протокола связи

```
0x1400, 0xD4C1, 0xD581, 0x1540, 0xD701, 0x17C0, 0x1680, 0xD641,
0xD201, 0x12C0, 0x1380, 0xD341, 0x1100, 0xD1C1, 0xD081, 0x1040,
0xF001, 0x30C0, 0x3180, 0xF141, 0x3300, 0xF3C1, 0xF281, 0x3240,
0x3600, 0xF6C1, 0xF781, 0x3740, 0xF501, 0x35C0, 0x3480, 0xF441,
0x3C00, 0xFCC1, 0xFD81, 0x3D40, 0xFF01, 0x3FC0, 0x3E80, 0xFE41,
0xFA01, 0x3AC0, 0x3B80, 0xFB41, 0x3900, 0xF9C1, 0xF881, 0x3840,
0x2800, 0xE8C1, 0xE981, 0x2940, 0xEB01, 0x2BC0, 0x2A80, 0xEA41,
0xEE01, 0x2EC0, 0x2F80, 0xEF41, 0x2D00, 0xEDC1, 0xEC81, 0x2C40,
0xE401, 0x24C0, 0x2580, 0xE541, 0x2700, 0xE7C1, 0xE681, 0x2640,
0x2200, 0xE2C1, 0xE381, 0x2340, 0xE101, 0x21C0, 0x2080, 0xE041,
0xA001, 0x60C0, 0x6180, 0xA141, 0x6300, 0xA3C1, 0xA281, 0x6240,
0x6600, 0xA6C1, 0xA781, 0x6740, 0xA501, 0x65C0, 0x6480, 0xA441,
0x6C00, 0xACC1, 0xAD81, 0x6D40, 0xAF01, 0x6FC0, 0x6E80, 0xAE41,
0xAA01, 0x6AC0, 0x6B80, 0xAB41, 0x6900, 0xA9C1, 0xA881, 0x6840,
0x7800, 0xB8C1, 0xB981, 0x7940, 0xBB01, 0x7BC0, 0x7A80, 0xBA41,
0xBE01, 0x7EC0, 0x7F80, 0xBF41, 0x7D00, 0xBDC1, 0xBC81, 0x7C40,
0xB401, 0x74C0, 0x7580, 0xB541, 0x7700, 0xB7C1, 0xB681, 0x7640,
0x7200, 0xB2C1, 0xB381, 0x7340, 0xB101, 0x71C0, 0x7080, 0xB041,
0x5000, 0x90C1, 0x9181, 0x5140, 0x9301, 0x53C0, 0x5280, 0x9241,
0x9601, 0x56C0, 0x5780, 0x9741, 0x5500, 0x95C1, 0x9481, 0x5440,
0x9C01, 0x5CC0, 0x5D80, 0x9D41, 0x5F00, 0x9FC1, 0x9E81, 0x5E40,
0x5A00, 0x9AC1, 0x9B81, 0x5B40, 0x9901, 0x99C0, 0x5880, 0x9841,
0x8801, 0x48C0, 0x4980, 0x8941, 0x4B00, 0x8BC1, 0x8A81, 0x4A40,
0x4E00, 0x8EC1, 0x8F81, 0x4F40, 0x8D01, 0x4DC0, 0x4C80, 0x8C41,
0x4400, 0x84C1, 0x8581, 0x4540, 0x8701, 0x47C0, 0x4680, 0x8641,
0x8201, 0x42C0, 0x4380, 0x8341, 0x4100, 0x81C1, 0x8081, 0x4040
};
```

```
unsigned short Crc16(unsigned char * pcBlock, unsigned short len)
{
    unsigned short crc = 0xFFFF;

    while (len-->0)
        crc = (crc >> 8) ^ Crc16Table[(crc & 0xFF) ^ *pcBlock++];

    return crc;
}
```

Контрольная сумма добавляется к сообщению младшим байтом вперёд.

Например, сообщение: 01 03 00 00 00 01

Контрольная сумма равна (dec) 2692 (hex) **A84**.

Старший байт контрольной суммы (hex) **0A**, младший байт контрольной суммы (hex) **84**.

01 03 00 00 00 01 **84 0A**

**Описание кодов функций.**

## Код функции 0x03.

Функция используется для чтения нескольких регистров, располагающихся в адресном пространстве последовательно один за другим.

Формат запроса:

<b>Адрес устройства</b>	1 байт	0x01 – 0xFF
<b>Код функции</b>	1 байт	0x03
<b>Стартовый адрес</b>	2 байта	0 – 11000 (0x0000 – 0x2AF8)
<b>Количество регистров</b>	2 байта	1 – 125 (0x01 - 0x7D)
<b>Контрольная сумма</b>	2 байта	CRC16

Формат ответа:

<b>Адрес устройства</b>	1 байт	0x01 – 0xFF
<b>Код функции</b>	1 байт	0x03
<b>Количество байт</b>	1 байт	2*N
<b>Значение регистра</b>	N*2 байта	
<b>Контрольная сумма</b>	2 байта	CRC16

N – количество регистров.

Формат ответа при обнаружении ошибки:

<b>Адрес устройства</b>	1 байт	0x01 – 0xFF
<b>Код функции</b>	1 байт	0x83
<b>Код исключения</b>	1 байт	1,2,3,4 (0x01,0x02,0x03,0x04)
<b>Контрольная сумма</b>	2 байта	CRC16

Пример запроса (hex): 01 03 00 00 00 01 84 0A

Пример ответа (hex): 01 03 02 80 00 D9 84

Запрос		Ответ	
Имя поля запроса	Значение	Имя поля ответа	Значение
Адрес устройства	<b>0x01</b>	Адрес устройства	<b>0x01</b>
Код функции	<b>0x03</b>	Код функции	<b>0x03</b>
Стартовый адрес (ст.байт)	<b>0x00</b>	Количество байт	<b>0x02</b>
Стартовый адрес (мл.байт)	<b>0x00</b>	Значение регистра №0 (ст.байт)	<b>0x80</b>
Количество регистров (ст.байт)	<b>0x00</b>	Значение регистра №0 (мл.байт)	<b>0x00</b>
Количество регистров (мл.байт)	<b>0x01</b>	Контрольная сумма (мл.байт)	<b>0xD9</b>

## Описание протокола связи

Контрольная сумма (мл.байт)	<b>0x84</b>	Контрольная сумма (ст.байт)	<b>0x84</b>
Контрольная сумма (ст.байт)	<b>0x0A</b>		

### Код функции 0x10.

Функция используется для записи нескольких регистров, располагающихся в адресном пространстве последовательно один за другим.

Формат запроса:

<b>Адрес устройства</b>	1 байт	0x01 – 0xFF
<b>Код функции</b>	1 байт	0x10
<b>Стартовый адрес</b>	2 байта	0 – 11000 (0x0000 – 0x2AF8)
<b>Количество регистров</b>	2 байта	1 – 123 (0x01 - 0x7B)
<b>Количество байт</b>	1 байт	2*N
<b>Значение регистра</b>	N*2 байта	
<b>Контрольная сумма</b>	2 байта	CRC16

Формат ответа:

<b>Адрес устройства</b>	1 байт	0x01 – 0xFF
<b>Код функции</b>	1 байт	0x10
<b>Стартовый адрес</b>	2 байта	0 – 11000 (0x0000 – 0x2AF8)
<b>Количество регистров</b>	2 байта	1 – 123 (0x01 - 0x7B)
<b>Контрольная сумма</b>	2 байта	CRC16

N – количество регистров.

Формат ответа при обнаружении ошибки:

<b>Адрес устройства</b>	1 байт	0x01 – 0xFF
<b>Код функции</b>	1 байт	0x90
<b>Код исключения</b>	1 байт	1,2,3,4 (0x01,0x02,0x03,0x04)
<b>Контрольная сумма</b>	2 байта	CRC16

Пример запроса (hex): 01 10 00 00 00 02 04 00 01 00 02 23 AE

Пример ответа (hex): 01 10 00 00 00 02 41 C8

Запрос		Ответ	
Имя поля запроса	Значение	Имя поля ответа	Значение



## Описание протокола связи

Адрес устройства	<b>0x01</b>	Адрес устройства	<b>0x01</b>
Код функции	<b>0x10</b>	Код функции	<b>0x03</b>
Стартовый адрес (ст.байт)	<b>0x00</b>	Стартовый адрес (ст.байт)	<b>0x00</b>
Стартовый адрес (мл.байт)	<b>0x00</b>	Стартовый адрес (мл.байт)	<b>0x00</b>
Количество регистров (ст.байт)	<b>0x00</b>	Количество регистров (ст.байт)	<b>0x00</b>
Количество регистров (мл.байт)	<b>0x02</b>	Количество регистров (мл.байт)	<b>0x02</b>
Количество байт	<b>0x04</b>	Контрольная сумма (мл.байт)	<b>0x41</b>
Значение регистра №0 (ст.байт)	<b>0x00</b>	Контрольная сумма (ст.байт)	<b>0xC8</b>
Значение регистра №0 (мл.байт)	<b>0x01</b>		
Значение регистра №1 (ст.байт)	<b>0x00</b>		
Значение регистра №1 (мл.байт)	<b>0x02</b>		
Контрольная сумма (мл.байт)	<b>0x23</b>		
Контрольная сумма (ст.байт)	<b>0xAE</b>		

### Код функции 0x20.

Функция пользователя. Используется для:

- а) записи в ЭНП уставок минимальных значений из ПЗУ;
- б) очистки аварийного журнала;
- в) перезагрузки устройства.

Формат запроса и ответа не стандартный.

### Код функции 0x41.

Функция пользователя. Используется для чтения аварийной осциллограммы. Формат запроса и ответа аналогичен функции 0x03.

Карта памяти аварийного осциллографа не приводится.

### Код функции 0x51.

Функция пользователя. Используется для чтения ЭНП уставок. Формат запроса и ответа аналогичен функции 0x03.

# Описание протокола связи

Стартовый адрес ЭНП уставок 0.

Количество регистров ЭНП уставок 4096.

## Код функции 0x52.

Функция пользователя. Используется для записи ЭНП уставок. Формат запроса и ответа аналогичен функции 0x10.

Стартовый адрес ЭНП уставок 0.

Количество регистров ЭНП уставок 4096.

## Карта памяти устройства.

Стартовый адрес	Количество регистров	Описание	Примечание
0	115	Уставки защит	
115	3	Параметры связи	
118	1	Пароль	Не используется
119	1	Номер индицируемого параметра	
120	9	Зарезервировано	
130	1	Константа (0хАААА)	
131	1	Тип исполнения УМПЗ	505,506,507
132	1	Версия ПО: День.месяц	
133	1	Версия ПО: Часов-минут	
134	1	Серийный номер устройства	
135	1	Ячейка дистанционного управления выключателем	
136	1	Ячейка дистанционного деблокирования УМПЗ	
137	18	Зарезервировано	
155	15	Уставки программируемых светодиодов	
170	64	Уставки входов	
234	64	Уставки выходов	
298	1	Состояние входов устройства	
299	1	Состояние выходов устройства	
300	14	Действующие значения токов и напряжений	

## Описание протокола связи

314	11	Фазы векторов токов и напряжений относительно Ua	
325	5	Зарезервировано	
330	18	Значения параметров на момент отключения	
350	7	Дата и время для чтения из устройства	
357	6	Дата и время для записи в устройство	
363	1	Зарезервировано	
364	35	Зарезервировано	
400	340	Аварийный журнал	
740	10	Дополнительные параметры аварийного журнала	
750	200	Зарезервировано	
950	14	Калибровочные коэффициенты	
963	37	Зарезервировано	

### Ограничения на запись данных.

**Важное замечание: Команда записи нескольких регистров (функция 0x10) будет успешно выполнена только для следующих наборов стартовых адресов и количества регистров:**

Стартовый адрес	Количество регистров	Описание	Примечание
0	115	Запись уставок защит	
115	3	Запись параметров обмена	
135	1	Дистанционное включение/отключение выключателя	
136	1	Дистанционное деблокирование выключателя	
155	15	Запись уставок программируемых светодиодов	
170	64	Запись уставок входов	(*)
234	64	Запись уставок выходов	(**)
170	123	Запись уставок входов-выходов	(***)
293	5	Запись уставок выходов (5 последних регистров)	(****)
357	6	Запись даты и времени.	

(\*) - записать уставки всех входов (8 не программируемых и 8 программируемых). Для корректного отображения уставок 8 не программируемых входов, уставки этих входов нужно предварительно считать из устройства.

## Описание протокола связи

(\*\*) - записать уставки всех выходов (8 не программируемых и 8 программируемых). Для корректного отображения уставок 8 не программируемых выходов, уставки этих выходов нужно предварительно считать из устройства.

(\*\*\*) - записать уставки входов-выходов (123 регистра). Для корректного отображения уставок 8 не программируемых входов и выходов, уставки этих входов и выходов нужно предварительно считать из устройства.

(\*\*\*\*) - записать уставки входов-выходов (5 регистров).

### Ограничения протокола на интерфейсах RS232 и RS485.

Параметры	Операция	RS232	RS485	Примечание
Уставки защит	Чтение			
	Запись			
Параметры связи	Чтение			
	Запись			
Дистанционное управление	Запись			
Программируемые светодиоды	Чтение			
	Запись			
Программируемые входы	Чтение			
	Запись		-	
Программируемые выходы	Чтение			
	Запись		-	
Установка даты и времени	Чтение			
	Запись			
Текущие измерения	Чтение			
Аварийный журнал	Чтение			
Аварийная осциллограмма	Чтение			
Энергонезависимая память уставок	Чтение		-	
	Запись		-	

## Тестирование сторонними программными средствами.

Для тестирования протокола и интерфейса, кроме программы УМПЗ, можно воспользоваться программой ModLink Demo. Программу можно найти по адресу <http://www.ozm.cz/ivobauer/modlink/>

