
РЕАЛИЗАЦИЯ ПРОТОКОЛА МОДБАС для связи с вычислителем ВКТ-7

Содержание

Что нового	2
1 Принцип организации данных, доступных через сеть, в тепловычислителе ВКТ-7	3
2 Обобщенная структура запросов/ответов в сетевом протоколе Modbus	10
2.1 Отступления от требований стандарта Modbus в протоколе в ВКТ-7	11
3 Структура запросов/ответов в сетевом протоколе Modbus	13
3.1 Структура запросов/ответов для операции 0x03 («чтение»).....	13
3.2 Структура запросов/ответов для операции 0x10 («запись»).....	14
4 Возможные варианты запросов	16
4.1 Запрос на чтение перечня активных элементов данных.....	16
4.2 Запрос на запись перечня элементов для чтения	16
4.3 Запрос на запись типа значений	17
4.4 Запрос на запись даты	18
4.5 Запрос на чтение данных.....	18
4.6 Запрос на чтение служебной информации.....	20
4.7 Запрос «Начало сеанса связи»	21
4.8 Запрос «Чтение интервала дат»	21
4.9 Запрос «Чтение номера схемы измерения»	22
4.10 Запрос «Чтение номера активной базы данных»	22
4.11 Запрос «Чтение настройки «Идентификатор абонента»»	23
4.12 Запрос «Чтение состояний дискретных выходов»	24
4.13 Запрос «Запись состояний дискретных выходов»	24
5 Порядок следования запросов	26
5.1 Начало сеанса связи	26
5.2 Получение «свойств»	26
5.3 Порядок получения необходимых данных.....	31
5.4 Получение архивных данных.....	31
5.5 Получение текущих и итоговых текущих данных	31
Приложение А – Расчет контрольной суммы кадра Modbus	33
Приложение Б – Используемые типы данных и символные константы.....	34

Что нового

С появлением версии 1.9 встроенного программного обеспечения вычислителя ВКТ-7 в протокол обмена внесены следующие изменения:

- Изменился анализ ответа на запрос «начало сеанса связи»;
- Изменился анализ ответа при получении «свойств»;
- Изменилось перечисление, описывающее все возможные элементы данных в вычислителе (в связи с добавлением элементов данных «наличие нештатной ситуации по тепловому вводу» и «счетчики времени диагностических ситуаций»);
- Изменился анализ ответа на запрос «чтение данных...»;
- Предусмотрена работа при использовании вычислителей со встроенным адаптером RS-485 (см. Обобщенная структура запросов/ответов в сетевом протоколе Modbus);
- Появилась возможность читать не только текущий номер схемы измерения, но и номер схемы измерения из архива;

С появлением версии 2.0 встроенного программного обеспечения вычислителя ВКТ-7 в протокол обмена внесены следующие изменения:

- Максимальный размер буфера сообщения изменился с 256 на 264 байта;
- Изменилось перечисление, описывающее все возможные элементы данных в вычислителе (в связи с добавлением элементов данных «дополнительное давление» и «дополнительный вход»).
- Появилась возможность дистанционно управлять дискретными выходами.
- Описана возможность чтения номера активной базы данных и Идентификатора Абонента (начиная с версии ПО 1.9).

1 Принцип организации данных, доступных через сеть, в тепловычислителе ВКТ-7

В соответствии с Руководством по эксплуатации тепловычислитель ВКТ–7 предназначен для учета, регистрации и дистанционного мониторинга количества теплоты (тепловой энергии) и параметров теплоносителя в системах водяного теплоснабжения. Под дистанционным мониторингом понимается работа тепловычислителя в сети, удаленный доступ с помощью модемной связи или непосредственная работа с компьютером. В этом случае применяется система верхнего уровня – персональный компьютер с установленным специализированным программным обеспечением.

Система верхнего уровня может получать от тепловычислителя ВКТ-7 следующие данные:

- текущие и архивные результаты измерений (температуры, давления, массы, объемы и т.д.);
- так называемые «свойства» (наименование единиц измерения и данные, необходимые для правильной интерпретации ответов вычислителя).

Каждая из этих величин называется **элементом данных**. В зависимости от выбранной схемы измерения элементы данных могут участвовать или не участвовать в измерениях. Например, в схеме измерения №1 для теплового ввода №1 температура t_2 участвует в расчетах, а в схеме измерения №6 – нет. Элементы данных, участвующие в измерениях, называются **активными**, не участвующие в измерениях – **не активными**.

В зависимости от схемы измерения вычислитель формирует **маску активности элементов**, т.е. для каждого элемента данных определяется, активен ли он при данной схеме измерения. Элементы данных «свойств» **всегда активны**.

В процессе эксплуатации прибора может возникнуть ситуация, когда схема измерения меняется (например, при смене сезона может применяться «летняя» или «зимняя» схема). При изменении схемы, изменяется и маска активности.

Схема измерений сохраняется в каждой архивной записи, поэтому маска активности элементов формируется вычислителем для каждой архивной записи в момент обращения к вычислителю. Если в вычислителе изменилась схема измерения, то для архивной записи, которая сформируется уже после смены схемы измерения, маска активности будет отличаться от маски для той записи, которая была сформирована ранее. В маске активности некоторые элементы могут стать активными, или, наоборот, неактивными. Та маска активности, которая сформировалась при последнем обращении системы верхнего уровня к вычислителю, является **актуальной**.

Из маски активности вычислитель формирует **перечень активных элементов**. Если в маске активности присутствуют все элементы данных (активные и неактивные), то в перечень входят только активные элементы. Перечень состоит из структур, содержащих два поля:

- условный адрес активного элемента, шириной 4 байта;
- размерность этого элемента длиной 2 байта.

При получении данных (архивных или текущих) от вычислителя система верхнего уровня должна сначала запросить перечень активных элементов. Из перечня активных элементов можно выбрать только интересующие клиента элементы, сформиро-

вать **перечень элементов для чтения** и записать его в вычислитель. В дальнейшем вычислитель будет присылать данные в соответствии с записанным перечнем.

Например, если система верхнего уровня из перечня активных элементов, условно состоящего из t_1 , t_2 , t_3 выбрала только t_2 , то вычислитель в ответ на запрос «Чтение данных...» пришлет информацию только по t_2 . При этом перечень активных элементов в вычислителе останется прежним. Система верхнего уровня формирует лишь перечень элементов для чтения данных.

Следует понимать, что перечень активных элементов является функцией настроек вычислителя (в частности схемы измерения) и не зависит от типа данных запрашиваемых системой верхнего уровня. Т.е. при запросе перечня активных элементов в процессе чтения текущих значений и архивных значений перечень активных элементов будет тем же самым (при условии неизменности схемы измерения). Однако, при чтении значений некоторые элементы данных не будут иметь смысла. Поясним на примере. Допустим, система верхнего уровня прочитала такой перечень активных элементов:

- T1 по TB1;
- G1 по TB1;
- M1 по TB1.

Если этот же перечень записать в вычислитель, в качестве перечня элементов для чтения, то при выполнении запроса на чтение данных вычислитель вернет:

- Для текущих значений:
 - значение T1 по TB1 + байт качества + байт HC;
 - значение G1 по TB1 + байт качества + байт HC;
 - бессмысленное значение M1 по TB1 + байт качества равный OPC_QUALITY_BAD | OPC_QUALITY_CONFIG_ERROR (элемент отсутствует) + байт HC.
- Для архивных значений (часовых, суточных, месячных):
 - значение T1 по TB1 + байт качества + байт HC;
 - бессмысленное значение G1 по TB1 + байт качества равный OPC_QUALITY_BAD | OPC_QUALITY_CONFIG_ERROR (элемент отсутствует) + байт HC.
 - значение M1 по TB1+ байт качества + байт HC;
- Для итоговых значений:
 - бессмысленное значение T1 по TB1 + байт качества равный OPC_QUALITY_BAD | OPC_QUALITY_CONFIG_ERROR (элемент отсутствует) + байт HC;
 - бессмысленное значение G1 по TB1 + байт качества равный OPC_QUALITY_BAD | OPC_QUALITY_CONFIG_ERROR (элемент отсутствует) + байт HC.
 - значение M1 по TB1+ байт качества + байт HC;

Т.е. следует учитывать смысл элементов данных при формировании списка элементов для чтения. Например, элемент M (масса) не имеет смысла в текущих значениях, т.к. является параметром, характеризующим накопленную величину. А

G (расход) не имеет смысла в архиве, т.к. суть его - мгновенное значение. Особенно важно это для некоторых схем измерения, когда активны очень многие параметры. Записав полный список для чтения (включая не имеющие смысл элементы) система верхнего уровня «заставляет» вычислитель дать ответ по каждому элементу, даже если он не имеет смысла при данном установленном типе значений. Если активных элементов данных в этой схеме измерения много, то существует риск переполнения буфера передатчика прибора.

Вероятна ситуация, когда система верхнего уровня запрашивает архивные записи за интервал времени, внутри которого произошла смена схемы измерения. Тогда в один из моментов она обратится к записи, в которой схема измерения отличается от схемы измерения из предыдущей записи. Это приводит к тому, что актуальная маска активности не соответствует схеме измерения, для архивной записи, к которой происходит обращение. В этом случае вычислитель переформирует актуальную маску так, чтобы она соответствовала схеме измерения для архивной записи и сигнализирует об изменении схемы измерения в ответе на запрос «Чтение данных...» (подробнее о запросе см. ниже).

Например, система верхнего уровня прочитала данные из архивной записи, сформированной при схеме измерения №4. При этом актуальная маска активности элементов стала соответствовать схеме измерения для этой архивной записи, т.е. схеме №4. Затем система верхнего уровня обратилась к архивной записи сформированной при схеме измерения № 5. Вычислитель обнаруживает несоответствие актуальной маски схеме измерения из архивной записи и переформирует актуальную маску активности элементов в соответствии со схемой измерения №5, а затем сигнализирует об изменении схемы измерения. В результате этого, в актуальной маске активности некоторые элементы, из числа ранее неактивных, могли стать активными; а из числа ранее активных - неактивными. Также изменится и перечень активных элементов. Система верхнего уровня, получив сигнал об изменении схемы измерения, может поступить двумя способами:

1. Повторить запрос на чтение данных. Вычислитель пришлет данные в соответствии с записанным в него перечнем элементов для чтения, но с поправкой на маску активности элементов из архивной записи. Это означает, что если какие-либо элементы данных стали неактивны, в ответе будет содержаться информация об этом. В частности, в байтах качества будут помещены коды OPC_QUALITY_BAD | OPC_QUALITY_CONFIG_ERROR (коды описаны в Приложении Б). Этот способ имеет существенный недостаток: если в маске активности появились новые активные элементы, их нельзя будет получить;
2. Запросить новый перечень активных элементов данных, записать вновь сформированный перечень и затем прочитать данные.

Подробное описание запросов и ответов приведено в разделе «Возможные варианты запросов».

Вычислитель предоставляет данные следующих типов:

- часовой архив;
- суточный архив;
- месячный архив;
- итоговый архив;
- текущие значения;

- итоговые текущие;
- свойства;
- некоторые настроечные параметры;

Текущие значения представляют собой измеренные в данный момент параметры теплотребления. В приборе имеется только одна запись этого типа. Дискретность обновления зависит от установок параметров:

- ПИ – период измерения для температур и давлений (6, 60, 600 с.);
- УО – уставка на отсечку по каждому трубопроводу (от 18 с. до УО).

Список текущих параметров:

- t1 Тв1;
- t2 Тв1;
- t3 Тв1;
- P1 Тв1;
- P2 Тв1;
- dt Тв1;
- tx;
- ta;
- G1 Тв1;
- G2 Тв1;
- G3 Тв1;
- t1 Тв2;
- t2 Тв2;
- t3 Тв2;
- P1 Тв2;
- P2 Тв2;
- dt Тв2;
- G1 Тв2;
- G2 Тв2;
- G3 Тв2;
- Наличие нештатной ситуации по Тв1;
- Наличие нештатной ситуации по Тв2;
- DI;
- P3.

Итоговые текущие значения представляют интегральные значения параметров теплотребления от момента сброса архива вычислителя до конца предыдущего часа. В приборе имеется только одна запись этого типа. Дискретность обновления – один час. Список итоговых текущих параметров:

- V1 Тв1;

- V2 Тв1;
- V3 Тв1;
- M1 Тв1;
- M2 Тв1;
- M3 Тв1;
- Mг Тв1;
- Qo Тв1;
- Qг Тв1;
- ВНР Тв1;
- ВОС Тв1;
- V1 Тв2;
- V2 Тв2;
- V3 Тв2;
- M1 Тв2;
- M2 Тв2;
- M3 Тв2;
- Mг Тв2;
- Qo Тв2;
- Qг Тв2;
- ВНР Тв2;
- ВОС Тв2;
- DI.

Часовой, суточный и месячный архивы представляют собой данные, имеющие смысл значений параметров теплотребления за период времени, определяемый дискретностью архива (за час, сутки или месяц). Т.е. это разница, среднее или длительность, в зависимости от смысла параметра, за период от конца предыдущего часа (суток, месяца) до конца текущего часа (суток, месяца). Конец отчетного месяца определяется настройкой параметра ДО (дата отчета). Архив имеет множество записей с разными хронологическими метками (количество зависит от глубины архива).

Список параметров:

- t1 Тв1;
- t2 Тв1;
- t3 Тв1;
- V1 Тв1;
- V2 Тв1;
- V3 Тв1;
- M1 Тв1;
- M2 Тв1;

- M3 Тв1;
- P1 Тв1;
- P2 Тв1;
- Mг Тв1;
- Qo Тв1;
- Qг Тв1;
- dt Тв1;
- tx;
- ta;
- ВНР Тв1;
- ВОС Тв1;
- t1 Тв2;
- t2 Тв2;
- t3 Тв2;
- V1 Тв2;
- V2 Тв2;
- V3 Тв2;
- M1 Тв2;
- M2 Тв2;
- M3 Тв2;
- P1 Тв2;
- P2 Тв2;
- Mг Тв2;
- Qo Тв2;
- Qг Тв2;
- dt Тв2;
- ВНР Тв2;
- ВОС Тв2;
- Наличие нештатной ситуации по Тв1;
- Наличие нештатной ситуации по Тв2;
- Длительность НС по параметрам Тв1;
- Длительность НС по параметрам Тв2;
- DI;
- P3.

Итоговый архив представляет интегральные параметры теплотребления от момента сброса архива вычислителя до момента формирования очередной записи архива. Дискретность архива – один отчетный месяц. Т.е. при наступлении конца отчетного месяца происходит архивация данных (запись итоговых текущих значений в итоговый архив). Список параметров такой же, как для итоговых текущих

значений. Архив имеет множество записей с разными хронологическими метками (количество зависит от глубины архива). Разница итогового архива и итоговых текущих значений заключается в том, что итоговые текущие значения – это интегралы от сброса до конца предыдущего часа, а итоговый архив, это несколько записей, имеющих смысл интегралов от сброса до конца отчетного месяца.

2 Обобщенная структура запросов/ответов в сетевом протоколе Modbus

Для обмена данными по локальной сети используется протокол Modbus, работающий по принципу ведущий-ведомый (Master-Slave). Это значит, что ведомый (Slave) может послать сетевое сообщение только в ответ на запрос ведущего (Master). В качестве ведущего используется система верхнего уровня. Вычислители ВКТ-7 выполняют только роль ведомого. При взаимодействии ведущего и ведомого используется режим передачи RTU (Remote Terminal Unit). В этом режиме для каждого байта сообщения используется 8-разрядная двоичная система кодирования.

Время, затрачиваемое на передачу одного байта сообщения, называется **периодом символа**. Интервал времени, в течение которого не происходило передачи, называется **интервалом тишины**.

При использовании режима RTU сообщение должно начинаться с интервала тишины длительностью не менее 3,5 периодов символа. Затем начинается передача кадра сообщения. После передачи последнего символа кадра сообщения в качестве признака конца также должен быть выдержан интервал тишины длительностью не менее 3,5 периодов символа. После окончания этого интервала может начинаться следующий кадр сообщения. Всё сообщение должно передаваться как непрерывный поток символов. Интервал тишины между соседними символами не должен превышать 1,5 периода символа.

В отступление от требований стандарта Modbus в вычислителе ВКТ-7 граница кадра определяется интервалом тишины длительностью 62.5 мс или по переполнению входного буфера длиной 264 байт.

Длина кадра сообщения не должна превышать 264 байта.

В качестве интерфейса физического уровня используется RS-232 (RS-485), причем взаимодействие ведущего и ведомого возможно как непосредственно через RS-232 (RS-485), так и с использованием наряду с RS-232 модемов и коммутируемой телефонной линии.

Для обмена по интерфейсу физического уровня RS-232 (RS-485) вычислитель ВКТ-7 использует следующие параметры:

- **формат символа:** 8 бит данных, 2 стоповых бита, без контроля четности;
- **скорость передачи,** бит/с: 1200, 2400, 4800, 9600, 19200;
- управление потоком: отсутствует;
- **сигнал RTS компьютера:** должен иметь уровень не менее +9 V.

Примечание При посылке запроса прибору сначала необходимо передать не менее двух байт 0xFF для гарантированного «вывода» прибора из «спящего» режима. Байты 0xFF могут стоять в начале посылки, то есть необязательно выдерживать паузу между 0xFF и кадром Modbus.

Это требование не распространяется на вычислители со встроенным адаптером RS-485 (для таких вычислителей предварительные байты 0xFF не требуются). При использовании вычислителей со встроенным адаптером RS-485, настроенным на расширенный режим адресации, в начало посылки должен быть добавлен байт

«расширенный адрес» (это поле не участвует в расчете контрольной суммы сообщения).

Кадр Modbus состоит из следующих элементов:

- поля адреса;
- поля функции;
- поля данных;
- контрольной суммы.

Ниже рассмотрено более подробно назначение отдельных полей:

- **адрес** – 8-разрядное число в диапазоне 0-240 (сетевой номер прибора) (для ПО менее 1.9 в диапазоне 0-99), служащее для правильной адресации запроса получателю. На запрос с нулем в качестве сетевого адреса ответит любой прибор (кроме приборов со встроенным адаптером RS-485) – так называемый широковещательный запрос. Его можно применять только при подключении точка-точка, то есть при подключению к компьютеру одного прибора;

Примечание При использовании вычислителей со встроенным адаптером RS-485, настроенным на расширенный режим адресации, первым байтом посылки (до поля «Адрес») должен быть байт «расширенный адрес». Для всех вычислителей со встроенным адаптером RS-485 независимо от режима адресации **не допускается использование адреса 0**.

- **функция** – 8 разрядное значение. При передаче сообщения от ведущего к ведомому поле функции задает последнему действие, которое он должен выполнить. При ответе ведомого значение поля функции указывает, произошел ли обмен нормально или возникло исключение. В нормальном ответе используется тот же код функции, что и в запросе. Возникновение исключения приводит к передаче ответа с кодом функции, в котором старший бит установлен, а остальные биты такие же, как в коде функции запроса. При связи с вычислителем ВКТ-7 используются два кода функций: **0x03** (чтение) и **0x10** (запись);
- **данные** — это последовательность 8-разрядных чисел. Поле данных сообщений, передаваемых ведущим ведомому, содержит предназначенную последнему дополнительную информацию, зависящую от кода функции. Дополнительная информация содержится в двух служебных элементах, каждый из которых представлен двумя 8-разрядными числами. Первый из этих служебных элементов – «**Начальный адрес**», второй – «**Количество регистров**». Более подробная структура кадров запроса и ответа будет рассмотрена для каждой функции отдельно;
- **контрольная сумма** — 16-разрядное поле, необходимое для контроля целостности кадра. Рассчитывается по алгоритму Crc16 (см. Приложение А).

2.1 Отступления от требований стандарта Modbus в протоколе в ВКТ-7

- Все поля в кадрах Modbus, имеющие длину более 1-го байта, должны передаваться в следующем порядке: сначала младший байт, затем старший. Это требование не относится к полям «**Начальный адрес**» и «**Количество регистров**». Они должны передаваться в следующем порядке: сначала старший байт, затем младший байт;

- Поле «**Количество регистров**» не анализируется прибором при обработке запроса и **может быть** установлено в произвольное значение, если иное не оговорено специально при описании запроса;
- Поле «**Количество записываемых байт данных**» может содержать значение, не соответствующее реально записываемому количеству байт данных. Например, в запросах «**Запись состояний дискретных выходов**» и «**Начало сеанса связи**»;
- Граница кадра определяется фиксированным интервалом тишины длительностью 62.5 мс или по переполнению входного буфера длиной 264 байта.

3 Структура запросов/ответов в сетевом протоколе Modbus

В данном разделе поясняется структура запросов/ответов, используемых при получении данных из тепловычислителя ВКТ-7. Конкретные значения полей кадров Modbus приведены в разделе «Возможные варианты запросов».

Корректная работа по протоколу Modbus возможно только с приборами, имеющими версию ПО 1.5 и более.

3.1 Структура запросов/ответов для операции 0x03 («чтение»)

Запрос:

- Адрес;
- Функция – 0x03;
- Начальный адрес (НАст.б) (старший байт);
- Начальный адрес (НАмл.б) (младший байт);
- Количество регистров (КРст.б) (старший байт);
- Количество регистров (КРмл.б) (младший байт);
- Контрольная сумма (КСмл.б) (младший байт);
- Контрольная сумма (КСст.б) (старший байт);

Структура запроса на чтение приведена на рисунке.

A	0x03	НА (ст.б)	НА (мл.б)	КР (ст.б)	КР (мл.б)	КС (мл.б)	КС (ст.б)
---	------	-----------	-----------	-----------	-----------	-----------	-----------

Значения полей «**Начальный адрес**» и «**Количество регистров**» для конкретных запросов будут приведены ниже.

Ответ:

В случае успешного обмена вычислитель присылает ответ следующего формата:

- адрес (значение, скопированное из запроса);
- функция (значение, скопированное из запроса).
- количество байт данных (КБ);
- секция данных длиной «количество байт данных»;
- Контрольная сумма (младший байт);
- Контрольная сумма (старший байт).

Структура ответа на команду чтения приведена на рисунке.

A	0x03	КБ	Данные	КС (мл.б)	КС (ст.б)
---	------	----	--------	-----------	-----------

Если при обработке запроса возникло какое-либо исключение, то вычислитель устанавливает в поле функции старший бит. Это означает, что следует проанализировать код исключения, расположенный в следующем байте (расшифровка кодов ошибок приведена ниже для каждого запроса в отдельности). Ответ вычислителя с кодом ошибки имеет следующий формат:

- адрес (значение, скопированное из запроса);
- функция (значение, скопированное из запроса, с установленным старшим битом);
- код ошибки (КО);

- служебный байт (СБ - анализировать не нужно);
- Контрольная сумма (младший байт);
- Контрольная сумма (старший байт).

Структура ответа на команду чтения при возникновении исключения приведена на рисунке.

A	0x83	КО	СБ	КС (мл.б)	КС (ст.б)
---	------	----	----	-----------	-----------

3.2 Структура запросов/ответов для операции 0x10 («запись»)

Запрос:

- Адрес;
- 0x10;
- Начальный адрес (старший байт);
- Начальный адрес (младший байт);
- Количество регистров (старший байт);
- Количество регистров (младший байт);
- Количество записываемых байт данных;
- Записываемые данные;
- Контрольная сумма (младший байт);
- Контрольная сумма (старший байт).

Структура запроса на запись приведена на рисунке.

A	0x10	НА (ст.б)	НА (мл.б)	КР (ст.б)	КР (мл.б)	КБ	Данные	КС (мл.б)	КС (ст.б)
---	------	-----------	-----------	-----------	-----------	----	--------	-----------	-----------

Значения полей «Начальный адрес» и «Количество регистров» для конкретных запросов будут приведены ниже.

Ответ:

В случае успешного обмена вычислитель присылает ответ следующего формата:

- адрес (значение, скопированное из запроса);
- функция (значение, скопированное из запроса);
- Начальный адрес (старший байт; значение, скопированное из запроса);
- Начальный адрес (младший байт; значение, скопированное из запроса);
- Количество регистров (старший байт; значение, скопированное из запроса);
- Количество регистров (младший байт; значение, скопированное из запроса);
- Контрольная сумма (младший байт);
- Контрольная сумма (старший байт).

Структура ответа на команду записи приведена на рисунке.

A	0x10	НА (ст.б)	НА (мл.б)	КР (ст.б)	КР (мл.б)	КС (мл.б)	КС (ст.б)
---	------	-----------	-----------	-----------	-----------	-----------	-----------

Если при обработке запроса возникло какое-либо исключение, то вычислитель устанавливает в поле функции старший бит. Это означает, что следует проанализировать код исключения, расположенный в следующем байте (расшифровка кодов приведена ниже для каждого запроса в отдельности). Ответ вычислителя с кодом ошибки имеет следующий формат:

- адрес (значение, скопированное из запроса);
- функция (значение, скопированное из запроса, с установленным старшим битом);
- код ошибки;
- служебный байт (СБ - анализировать не нужно);
- Контрольная сумма (младший байт);
- Контрольная сумма (старший байт).

Структура ответа на команду записи при возникновении исключения приведена на рисунке.

А	0x90	КО	СБ	КС (мл.б)	КС (ст.б)
---	------	----	----	-----------	-----------

4 Возможные варианты запросов

Во всех примерах запросов будет использоваться:

- сетевой адрес, равный нулю;
- Frame – массив элементов типа char;
- Checksum l – младший байт контрольной суммы,
- Checksum h – старший байт контрольной суммы.

4.1 Запрос на чтение перечня активных элементов данных

Запрос предназначен для получения перечня активных элементов и определения размера элементов в байтах. Размер элемента нужен для того, чтобы из принятого пакета данных «взять» необходимое количество байт.

Начальный адрес = **0x3FFC**;

Количество регистров = **0x0000**.

Пример:

```
Frame [0] = 0x00;  
Frame [1] = 0x03;  
Frame [2] = 0x3F;  
Frame [3] = 0xFC;  
Frame [4] = 0x00;  
Frame [5] = 0x00;  
Frame [6] = CheckSum l;  
Frame [7] = CheckSum h;
```

Готовый вариант запроса:

```
0x00 0x03 0x3f 0xfc 0x00 0x00 0x88 0x3f
```

Перечень активных элементов является массивом структур. Структура имеет два поля:

- условный адрес элемента (4 байта) (см. ниже перечисление, описывающее все элементы данных);
- размер элемента в байтах (2 байта).

4.2 Запрос на запись перечня элементов для чтения

Запрос предназначен для того, чтобы указать вычислителю значения каких именно элементов данных следует присылать в ответ на запрос «Чтение данных...». После успешного выполнения этого запроса прибор будет присылать значения только для тех элементов данных, логические адреса которых присутствовали в записанном перечне.

Перечень элементов для чтения, также как и перечень активных элементов, является массивом структур. Структуры имеют такие же поля, как и в перечне активных элементов. Перечень элементов для чтения формируется из перечня активных элементов.

Начальный адрес = **0x3FFF**;

Количество регистров = **0x0000**.

Поле условного адреса в элементе массива должно быть сформировано следующим образом:

«условный адрес» | 0x40000000,

Примечание Символ «|» означает операцию побитового «ИЛИ».

Значение поля размера данных в записываемом элементе массива должно быть равно значению поля размера данных из элемента массива, полученного при чтении перечня активных элементов (для соответствующей величины) (о формировании поля размера данных для «свойств» читайте в пункте «Получение «свойств»»).

Пример:

Нужно записать массив из N элементов (размер элемента равен $4+2 = 6$ байт).

```

Frame [0 ] = 0x00;
Frame [1 ] = 0x10;
Frame [2 ] = 0x3F;
Frame [3 ] = 0xFF;
Frame [4 ] = 0x00;
Frame [5 ] = 0x00;
Frame [6 ] = 6*N;
Frame [7 ] = младший байт элемента;
.....
Frame [12] = старший байт элемента;
.....
Frame [6*N+7] = CheckSum l;
Frame [6*N+8] = CheckSum h;

```

Готовый вариант запроса:

(запись массива для чтения из двух элементов: t1(TB1) и V1(TB1))

```

0x00 0x10 0x3f 0xff 0x00 0x00 0x0c 0x00 0x00 0x00 0x40 0x02
0x00 0x03 0x00 0x00 0x40 0x04 0x00 0xa2 0x5c

```

Прибор присылает ответ в виде стандартного подтверждения записи.

Возможные коды исключений:

- 5 – размер массива больше максимально возможного;
- 2 – задан несуществующий элемент.

4.3 Запрос на запись типа значений

Запрос предназначен для того, чтобы указать вычислителю, значения какого типа следует присылать в ответ на запрос «Чтение данных...».

Начальный адрес = **0x3FFD**;

Количество регистров = 0x0000

Возможные типы значений:

- 0-часовой архив;
- 1-суточный архив;
- 2-месячный архив;
- 3-итоговый архив;
- 4-текущие значения;
- 5-итоговые текущие;
- 6-свойства.

Пример:

```

Frame [0 ] = 0x00;
Frame [1 ] = 0x10;
Frame [2 ] = 0x3F;
Frame [3 ] = 0xFD;
Frame [4 ] = 0x00;
Frame [5 ] = 0x00;
Frame [6 ] = 0x02;
Frame [7 ] = тип значений;
Frame [8 ] = 0x00;
Frame [9 ] = CheckSum l;
Frame [10] = CheckSum h;

```

Готовый вариант запроса: (тип значений– суточный архив)

```
0x00 0x10 0x3f 0xfd 0x00 0x00 0x02 0x01 0x00 0x71 0x42
```

Прибор присылает ответ в виде стандартного подтверждения записи.

Возможный код ошибки - 2 (несуществующий тип значений).

4.4 Запрос на запись даты

Запрос предназначен для того, чтобы указать вычислителю, относительно какой хронологической метки следует передавать архивные данные. Запрос содержит хронологическую метку с указанием даты и времени. Вычислитель передает архивные данные в ответ на запрос «Чтение данных...» относительно той хронологической метки, которая была записана в вычислитель в последний раз.

Начальный адрес = **0x3FFB**;

Количество регистров = **0x0000**.

Дата передается в формате VT_DATA_RAP (описание в Приложении Б).

Пример:

```

Frame [0 ] = 0x00;
Frame [1 ] = 0x10;
Frame [2 ] = 0x3F;
Frame [3 ] = 0xFB;
Frame [4 ] = 0x00;
Frame [5 ] = 0x00;
Frame [6 ] = 0x04;
Frame [7 ] = день;
Frame [8 ] = месяц;
Frame [9 ] = год;
Frame [10] = час;
Frame [11] = CheckSum l;
Frame [12] = CheckSum h;

```

Готовый вариант запроса: (30 января 2003г. 0 часов)

```
0x00 0x10 0x3f 0xfb 0x00 0x00 0x04 0x1e 0x01 0x03 0x00 0xfa 0xaf
```

Прибор присылает ответ в виде стандартного подтверждения записи.

Возможный код ошибки

- 3 – в архиве отсутствуют данные за эту дату.

4.5 Запрос на чтение данных

Запрос предназначен для получения информации по тем элементам данных, условные адреса которых присутствовали в перечне элементов для чтения.

Начальный адрес = **0x3FFE**;

Количество регистров = **0x0000**.

Пример:

```
Frame [0] = 0x00;  
Frame [1] = 0x03;  
Frame [2] = 0x3F;  
Frame [3] = 0xFE;  
Frame [4] = 0x00;  
Frame [5] = 0x00;  
Frame [6] = CheckSum l;  
Frame [7] = CheckSum h;
```

Готовый вариант запроса:

```
0x00 0x03 0x3f 0xfe 0x00 0x00 0x29 0xff
```

Прибор присылает ответ в виде последовательности элементов разной длины.

Элемент последовательности имеет следующий вид:

- само значение (размер – в соответствии со значением в поле «размер...», полученным в ответ на запрос «Чтение перечня активных элементов» для соответствующей величины);

Примечание Для величин с «условным адресом» NSPrintTypeM_1, NSPrintTypeM_2 (см. ниже перечисление, описывающее все элементы данных) и имеющих «размер» 1 байт, значение является не числом, а печатным символом «*» или « ».

Примечание Для величин с «условным адресом» QntNS_1, QntNS_2 и имеющих «размер» 10 байт, значение является массивом из пяти чисел типа unsigned short (16 бит).

Примечание Величина с «условным адресом» DopInplmpP_Type имеет тип float (четырёхбайтное число одинарной точности). Для нее поле «размер...» анализировать не обязательно.

- байт качества;
- байт нештатных ситуаций (НС).

Значения байта качества

(Численные значения символьных констант качества приведены в Приложении Б):

- OPC_QUALITY_BAD | OPC_QUALITY_DEVICE_FAILURE – значение элемента находится вне диапазона (в приборе в данном случае на месте числового значения индицируется надпись “нет значения”);
- OPC_QUALITY_UNCERTAIN | OPC_QUALITY_SENSOR_CAL – у данного элемента есть нештатная ситуация;
- OPC_QUALITY_BAD | OPC_QUALITY_CONFIG_ERROR – элемент отсутствует в расчетной схеме.

Байт НС

Содержит код нештатной ситуации для данного элемента.

Если код равен 0 - НС отсутствуют, если 0xff - НС для данного элемента отсутствует, но есть НС у других элементов списка. Любое другое значение означает наличие нештатной ситуации у данного элемента.

Возможный код ошибки: 5 – зафиксировано изменение схемы измерения. В этом случае нужно следовать рекомендациям, описанным в разделе «Принципы организации данных, доступных через сеть, в тепловычислителе ВКТ-7».

4.6 Запрос на чтение служебной информации

Запрос предназначен для получения служебной информации (номер версии ПО, схема измерений, идентификатор абонента, сетевой номер, дата отчета и модель исполнения).

Начальный адрес = **0x3FF9**;

Количество регистров = **0x0000**.

Пример:

```

Frame [0] = 0x00;
Frame [1] = 0x03;
Frame [2] = 0x3F;
Frame [3] = 0xF9;
Frame [4] = 0x00;
Frame [5] = 0x00;
Frame [6] = CheckSum l;
Frame [7] = CheckSum h;

```

Готовый вариант запроса:

0x00 0x03 0x3f 0xf9 0x00 0x00 0x98 0x3e

Сегмент данных ответа состоит из следующих элементов:

Для версии ПО менее 1.5:

- дата отчета целое 8-ми разрядное число;

Для версии ПО 1.5 и более:

- номер версии ПО целое 8-ми разрядное число;
- схема измерения по ТВ1×2 целое 16-ти разрядное число;
- схема измерения по ТВ2×2 целое 16-ти разрядное число;
- идентификатор абонента 8 байт формата ASCII;
- сетевой номер прибора целое 8-ми разрядное число;
- дата отчета целое 8-ми разрядное число;
- модель исполнения целое 8-ми разрядное число.

Примечания:

1. Номер версии ПО отображается в двух тетрадах. Старшая – номер версии, младшая – номер реализации. Например, версия 1.8 имеет вид 0001 1000=0x18.

2. Ответ по схеме измерения по каждому из тепловых вводов содержит информацию непосредственно о схеме измерений, а также о назначении TP3 и t5. Соответствие разрядов ответа необходимой информации представлено на рисунке.

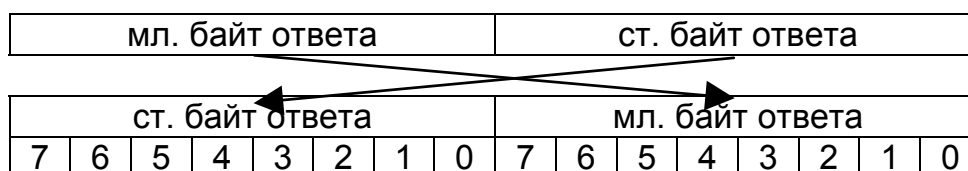


	Схема измерений	Назначение ТРЗ	Назначение t5	
--	--------------------	-------------------	------------------	--

4.7 Запрос “Начало сеанса связи”

Запрос предназначен для подготовки вычислителя к сеансу связи и определения значения «Версия сервера». Должен быть выполнен первым.

Начальный адрес = **0x3FFF**;

Количество регистров = **0x0000**.

Пример:

```

Frame [0 ] = 0x00;
Frame [1 ] = 0x10;
Frame [2 ] = 0x3F;
Frame [3 ] = 0xFF;
Frame [4 ] = 0x00;
Frame [5 ] = 0x00;
Frame [6 ] = 0xCC;
Frame [7 ] = 0x80;
Frame [8 ] = 0x00;
Frame [9 ] = 0x00;
Frame [10] = 0x00;
Frame [11] = CheckSum l;
Frame [12] = CheckSum h;

```

Готовый вариант запроса:

```
0x00 0x10 0x3f 0xff 0x00 0x00 0xcc 0x80 0x00 0x00 0x00 0x64 0x54
```

Ответ на запрос анализировать не нужно. При дальнейшей посылке запроса «Чтение данных» в ответе вычислителя необходимо проанализировать поле «Версия сервера». Это поле располагается в 65-ом по счету байте от начала ответа. Поле «Версия сервера» необходимо при получении «свойств» (см. ниже).

4.8 Запрос “Чтение интервала дат”

Запрос предназначен для получения дат начала/конца архива в приборе и текущей даты (только для версий ПО прибора 1.7 и более).

Начальный адрес = **0x3FF6**;

Количество регистров = **0x0000**.

Пример:

```

Frame [0 ] = 0x00;
Frame [1 ] = 0x03;
Frame [2 ] = 0x3F;
Frame [3 ] = 0xF6;
Frame [4 ] = 0x00;
Frame [5 ] = 0x00;
Frame [6 ] = CheckSum l;
Frame [7 ] = CheckSum h;

```

Готовый вариант запроса:

```
0x00 0x03 0x3f 0xf6 0x00 0x00 0xa8 0x3d
```

Прибор присылает ответ следующего формата:

- Дата начала часового архива;
- Текущая дата;
- Дата начала суточного архива (только для версий ПО прибора 1.6 и более).

Даты передаются в формате VT_DATA_RAP (см. Приложение Б).

4.9 Запрос “Чтение номера схемы измерения”

Запрос предназначен для получения номера схемы измерения для заданного теплового ввода (только для версий ПО прибора 1.9 и более). Для использования этого запроса необходимо не менее одного раза за сеанс связи подать запрос «Запись типа значений». Поле «Количество регистров» должно быть установлено в указанное значение, т.к. при обработке данного запроса вычислитель анализирует это поле.

Использование данного запроса обусловлено тем, что в процессе накопления архива схема измерения могла быть изменена. В ответ на запрос «Служебной информации...» присылается текущая схема измерения, и она может отличаться от схемы измерения, которая использовалась на момент формирования архивной записи. Целесообразно использовать этот запрос после получения первой архивной записи (из нескольких необходимых). Далее, если при последовательном чтении архива обнаруживается смена схемы измерения, нужно следовать рекомендациям, описанным в разделе 1 и потом снова запросить «Номер схемы измерения».

Начальный адрес = **0x3ECD**, для Тв1 и **0x3F5B** для Тв2.

Количество регистров = **0x0001**.

Пример:

```
Frame [0 ] = 0x00;  
Frame [1 ] = 0x03;  
Frame [2 ] = 0x3E (0x3F) ;  
Frame [3 ] = 0xCD 0x5B) ;  
Frame [4 ] = 0x00;  
Frame [5 ] = 0x01;  
Frame [6 ] = CheckSum l;  
Frame [7 ] = CheckSum h;
```

Готовый вариант запроса:

```
0x00 0x03 0x3e 0xcd 0x00 0x01 0xd8 0x0c
```

Прибор присылает ответ следующего формата:

- Номер схемы измерения (1 байт);
- Байт качества;
- Байт НС.

4.10 Запрос “Чтение номера активной базы данных”

Начиная с версии ПО 1.9, в вычислителе ВКТ-7 появилась возможность выбора одной из двух баз данных настроек (могут быть настроены две базы данных (например, летняя и зимняя) и выбрана та, которая используется в данный момент - активная). Запрос предназначен для получения номера активной базы данных (только для версий ПО прибора 1.9 и более). Для использования этого запроса необходимо не менее одного раза за сеанс связи подать запрос «Запись типа значений». Поле «Количество регистров» должно быть установлено в указанное значение, т.к. при обработке данного запроса вычислитель анализирует это поле.

Использование данного запроса обусловлено тем, что в процессе накопления архива в качестве активной могла быть выбрана другая база данных. Целесообразно

использовать этот запрос после получения первой архивной записи (из нескольких необходимых). Далее, если при последовательном чтении архива обнаруживается смена схемы измерения, нужно следовать рекомендациям, описанным в разделе 1 и потом снова запросить «Номер активной базы данных».

Начальный адрес = **0x3FE9**.

Количество регистров = **0x0001**.

Пример:

```
Frame [0 ] = 0x00;  
Frame [1 ] = 0x03;  
Frame [2 ] = 0x3F;  
Frame [3 ] = 0xE9;  
Frame [4 ] = 0x00;  
Frame [5 ] = 0x01;  
Frame [6 ] = CheckSum l;  
Frame [7 ] = CheckSum h;
```

Готовый вариант запроса:

```
0x00 0x03 0x3f 0xed 0x00 0x01 0x58 0x3b
```

Прибор присылает ответ следующего формата:

- Номер активной базы данных (1 байт). 0 – активна БД1, 1 – активна БД2;
- Байт качества;
- Байт НС.

4.11 Запрос “Чтение настройки «Идентификатор абонента»”

Запрос предназначен для получения элемента настроечной базы данных «Идентификатор абонента» (только для версий ПО прибора 1.9 и более). Поле «Количество регистров» должно быть установлено в указанное значение, т.к. при обработке данного запроса вычислитель анализирует это поле.

Начальный адрес = **0x3EA6**.

Количество регистров = **0x0008**.

Пример:

```
Frame [0 ] = 0x00;  
Frame [1 ] = 0x03;  
Frame [2 ] = 0x3E;  
Frame [3 ] = 0xA6;  
Frame [4 ] = 0x00;  
Frame [5 ] = 0x08;  
Frame [6 ] = CheckSum l;  
Frame [7 ] = CheckSum h;
```

Готовый вариант запроса:

```
0x00 0x03 0x3e 0xa6 0x00 0x08 0xa9 0xc7
```

Прибор присылает ответ следующего формата:

- Идентификатор абонента в виде структуры, имеющей два поля. Первое поле (типа unsigned short (16 бит)) определяет длину строки в байтах. Второе поле является массивом символов, имеющим длину, указанную в первом поле;

- Байт качества;
- Байт НС.

4.12 Запрос “Чтение состояний дискретных выходов”

Запрос предназначен для получения состояний дискретных выходов (только для версий ПО прибора 2.0 и более).

Начальный адрес = **0x3FEE**.

Количество регистров = **0x0000**.

Пример:

```
Frame [0 ] = 0x00;  
Frame [1 ] = 0x03;  
Frame [2 ] = 0x3F;  
Frame [3 ] = 0xEE;  
Frame [4 ] = 0x00;  
Frame [5 ] = 0x00;  
Frame [6 ] = CheckSum l;  
Frame [7 ] = CheckSum h;
```

Готовый вариант запроса:

```
0x00 0x03 0x3f 0xee 0x00 0x00 0x28 0x3a
```

Прибор присылает ответ следующего формата:

- Состояние выхода №1 (1 байт);
- Состояние выхода №2 (1 байт).

Возможный код ошибки: 7 – дискретные выходы не являются управляемыми дистанционно.

4.13 Запрос “Запись состояний дискретных выходов”

Запрос предназначен для получения состояний дискретных выходов (только для версий ПО прибора 2.0 и более).

Начальный адрес = **0x3FEE**.

Количество регистров = **0x0000**.

Пример:

```
Frame [0 ] = 0x00;  
Frame [1 ] = 0x10;  
Frame [2 ] = 0x3F;  
Frame [3 ] = 0xEE;  
Frame [4 ] = 0x00;  
Frame [5 ] = 0x00;  
Frame [6 ] = 0x01;  
Frame [7 ] = Состояние выхода №1 (0 или 1);  
Frame [8 ] = Состояние выхода №2 (0 или 1);  
Frame [9 ] = CheckSum l;  
Frame [10] = CheckSum h;
```

Готовый вариант запроса:

```
0x00 0x10 0x3f 0xee 0x00 0x00 0x01 0x00 0x01 0x43 0xb1
```

Прибор присылает ответ в виде стандартного подтверждения записи.

Возможный код ошибки: 7 – дискретные выходы не являются управляемыми дистанционно.

5 Порядок следования запросов

5.1 Начало сеанса связи

Первым запросом должен быть запрос «Начало сеанса связи». Ответ анализировать не нужно. Далее необходимо сделать запрос на «Чтение данных». В ответе вычислителя необходимо проанализировать поле «Версия сервера». После этого необходимо получить «свойства» (см. ниже), после чего можно приступить к получению архивных/текущих данных.

5.2 Получение «свойств»

Для корректной расшифровки ответов прибора необходимо получить так называемые «свойства».

К «свойствам» относятся:

- количество знаков дробной части для различных величин;
- наименование единиц измерения различных величин.

Вычислитель ВКТ-7 передает данные системе верхнего уровня в виде целых чисел (полученных в результате масштабирования вещественных), вещественных (только величины расхода и величина, измеряемая на дополнительном входе), а также просто целых чисел. Для получения верного строкового представления величины нужно выполнить следующие операции:

- преобразовать полученное число в строку. Если величина представлена не вещественным числом, то преобразовывать в строку следует как целое число (при реализации на языке программирования С это может быть вызов функции `sprintf` с параметром `%d`). В противном случае, преобразование должно выполняться как для вещественного числа (`sprintf` с параметром `%f`);
- если величина представлена не вещественным числом, отделить от конца строки заданное в «свойствах» количество знаков и поставить запятую (или точку).

Например, от прибора получено значение (и это какая-либо величина, но не расход и не величина, измеряемая на доп. входе), которое после преобразования в строку дало «12345», и из «свойств» известно, что количество знаков после запятой равно 2. В этом случае окончательное строковое представление величины будет иметь вид «123,45».

Для получения «свойств» следует выполнить приведенную ниже последовательность запросов:

1. Запись типа значение с номером 6 («свойства»);
2. Запись перечня элементов для чтения (формирование перечня для «свойств» будет описано ниже);
3. Чтение данных в соответствии с записанным перечнем. При получении ответа на этот запрос анализировать байты качества и нештатных ситуаций не нужно.

Для определения величин и свойств в тепловычислителе ВКТ-7 использовано следующее перечисление:

Условный адрес	Наименование в перечислении	Текстовое наименование	Тип
0	t1_1Type	t1 Тв1	параметр
1	t2_1Type	t2 Тв1	параметр
2	t3_1Type	t3 Тв1	параметр
3	V1_1Type	V1 Тв1	параметр
4	V2_1Type	V2 Тв1	параметр
5	V3_1Type	V3 Тв1	параметр
6	M1_1Type	M1 Тв1	параметр
7	M2_1Type	M2 Тв1	параметр
8	M3_1Type	M3 Тв1	параметр
9	P1_1Type	P1 Тв1	параметр
10	P2_1Type	P2 Тв1	параметр
11	Mg_1TypeP	Mг Тв1	параметр
12	Qo_1TypeP	Qo Тв1	параметр
13	Qg_1TypeP	Qг Тв1	параметр
14	dt_1TypeP	dt Тв1	параметр
15	tswTypeP	tx	параметр
16	taTypeP	ta	параметр
17	QntType_1HIP	ВНР Тв1	параметр
18	QntType_1P	ВОС Тв1	параметр
19	G1Type	G1 Тв1	параметр
20	G2Type	G2 Тв1	параметр
21	G3Type	G3 Тв1	параметр
22	t1_2Type	t1 Тв2	параметр
23	t2_2Type	t2 Тв2	параметр
24	t3_2Type	t3 Тв2	параметр
25	V1_2Type	V1 Тв2	параметр
26	V2_2Type	V2 Тв2	параметр
27	V3_2Type	V3 Тв2	параметр
28	M1_2Type	M1 Тв2	параметр
29	M2_2Type	M2 Тв2	параметр
30	M3_2Type	M3 Тв2	параметр
31	P1_2Type	P1 Тв2	параметр
32	P2_2Type	P2 Тв2	параметр
33	Mg_2TypeP	Mг Тв2	параметр
34	Qo_2TypeP	Qo Тв2	параметр
35	Qg_2TypeP	Qг Тв2	параметр
36	dt_2TypeP	dt Тв2	параметр
37	tsw_2TypeP	резерв	параметр
38	ta_2TypeP	резерв	параметр
39	Qnt_2TypeHIP	ВНР Тв2	параметр
40	Qnt_2TypeP	ВОС Тв2	параметр
41	G1_2Type	G1 Тв2	параметр
42	G2_2Type	G2 Тв2	параметр
43	G3_2Type	G3 Тв2	параметр
44	tTypeM	ед. измерения по t (температуре)	свойство
45	GTypeM	ед. измерения по G (расходу)	свойство
46	VTypeM	ед. измерения по V (объему)	свойство
47	MTypeM	ед. измерения по M (массе)	свойство
48	PTypeM	ед. измерения по P (давлению)	свойство

Условный адрес	Наименование в перечислении	Текстовое наименование	Тип
49	dtTypeM	ед. измерения по dt (температуре)	свойство
50	tswTypeM	ед. измерения по tx (температуре)	свойство
51	taTypeM	ед. измерения по ta (температуре)	свойство
52	MgTypeM	ед. измерения по M (массе)	свойство
53	QoTypeM	ед. измерения по Q (теплу)	свойство
54	QgTypeM	ед. измерения по Q (теплу)	свойство
55	QntTypeHIM	ед. измерения по ВНР (времени)	свойство
56	QntTypeM	ед. измерения по ВОС (времени)* (см. прим.)	свойство
57	tTypeFractDiNum	кол-во знаков после запятой для t	свойство
58	GTypeFractDigNum1	резерв	свойство
59	VTypeFractDigNum1	кол-во знаков после запятой для V по Тв1	свойство
60	MTypeFractDigNum1	кол-во знаков после запятой для M по Тв1	свойство
61	PTypeFractDigNum1	кол-во знаков после запятой для P	свойство
62	dtTypeFractDigNum1	кол-во знаков после запятой для dt	свойство
63	tswTypeFractDigNum1	кол-во знаков после запятой для tx	свойство
64	taTypeFractDigNum1	кол-во знаков после запятой для ta	свойство
65	MgTypeFractDigNum1	кол-во знаков после запятой для Mg	свойство
66	QoTypeFractDigNum1	кол-во знаков после запятой для Q	свойство
67	tTypeFractDigNum2	кол-во знаков после запятой для t	свойство
68	GTypeFractDigNum2	резерв	свойство
69	VTypeFractDigNum2	кол-во знаков после запятой для V по Тв2	свойство
70	MTypeFractDigNum2	кол-во знаков после запятой для M по Тв2	свойство
71	PTypeFractDigNum2	кол-во знаков после запятой для P	свойство
72	dtTypeFractDigNum2	кол-во знаков после запятой для dt	свойство
73	tswTypeFractDigNum2	кол-во знаков после запятой для tx	свойство
74	taTypeFractDigNum2	кол-во знаков после запятой для ta	свойство
75	MgTypeFractDigNum2	кол-во знаков после запятой для Mg	свойство
76	QoTypeFractDigNum2	кол-во знаков после запятой для Q	свойство
77	NSPrintTypeM_1	Наличие нештатной ситуации по ТВ1	параметр
78	NSPrintTypeM_2	Наличие нештатной ситуации по ТВ2	параметр
79	QntNS_1	Длительность НС по параметрам Тв1	параметр
80	QntNS_2	Длительность НС по параметрам Тв2	параметр
81	DopInpImpP_Type	DI	параметр
82	P3P_Type	P3	параметр

Примечание Наличие элементов QntNS_1 и QntNS_2 возможно только в версиях ПО вычислителя помеченных буквой м. Например ПВ 1.9'м.

Примечание Элемент QntTypeM в зависимости от того задействован или нет дополнительный вход (определяется по активности элемента DopInpImpP_Type) имеет смысл или единиц измерения счетчиков времени, или единиц измерения величины на дополнительном входе.

Примечание Величины с «условным адресом» QntNS_1, QntNS_2 представляют собой массивы длительностей нештатных ситуаций по тепловому вводу: НС по отключению питания, НС расход меньше минимума,

НС расход больше максимума, НС по неисправности датчика температуры, НС dt<2 °С.

При формировании запроса на запись перечня элементов для чтения (для элементов данных «свойств»), массив **должен** состоять из следующих элементов:

tTypeM	(единицы измерения t),
GtypeM	(единицы измерения расхода),
VtypeM	(единицы измерения объема),
MtypeM	(единицы измерения массы),
PtypeM	(единицы измерения давления),
QoTypeM	(единицы измерения тепла),
QntTypeMIM	(единицы измерения счетчиков времени),
QntTypeM	(В зависимости от того задействован или нет дополнительный вход (определяется по активности элемента <code>DopInpImpP_Type</code>) – это или единицы измерения счетчиков времени, или единицы измерения величины на дополнительном входе.)
tTypeFractDigNum	(количество знаков после запятой для t),
VTypeFractDigNum1	(количество знаков после запятой для объема ТВ1),
MTypeFractDigNum1	(количество знаков после запятой для массы ТВ1),
PTypeFractDigNum1	(количество знаков после запятой для давления),
QoTypeFractDigNum1	(количество знаков после запятой для тепла ТВ1),
MTypeFractDigNum2	(количество знаков после запятой для массы ТВ2),
VTypeFractDigNum2	(количество знаков после запятой для объема ТВ2),
QoTypeFractDigNum2	(количество знаков после запятой для тепла ТВ2) .

Для единиц измерения размер в элементе массива равен 7 байт, для количества знаков после запятой – 1 байт.

Пример элемента массива для единиц измерения:

Условный адрес: 0x40000000 | tTypeM;

Размер: 0x0007;

Пример элемента массива для количества знаков после запятой:

Условный адрес: 0x40000000 | QoTypeFractDigNum2;

Размер: 0x0001;

При обработке ответа на чтение свойств следует иметь в виду, что в зависимости от состояния поля «Версия сервера» (см. запрос «Начало сеанса связи») единицы измерения могут передаваться различными способами:

- «Версия сервера» равна 0. Наименование единиц измерения величин передаются в виде массива из семи символов в кодировке OEM;
- «Версия сервера» равна 1. Наименование единиц измерения величин передаются в виде структуры, имеющей два поля. Первое поле (типа `unsigned short` (16 бит)) определяет длину строки в байтах. Второе поле является массивом символов, имеющим длину, указанную в первом поле (символы в кодировке OEM).

Пример запроса на запись перечня элементов для чтения:

```
0xFF 0xFF 0x00 0x10 0x3F 0xFF 0x00 0x00 0x60 0x2C 0x00 0x00 0x40 0x07
0x00 0x2D 0x00 0x00 0x40 0x07 0x00 0x2E 0x00 0x00 0x40 0x07 0x00 0x2F
0x00 0x00 0x40 0x07 0x00 0x30 0x00 0x00 0x40 0x07 0x00 0x35 0x00 0x00
0x40 0x07 0x00 0x37 0x00 0x00 0x40 0x07 0x00 0x38 0x00 0x00 0x40 0x07
0x00 0x39 0x00 0x00 0x40 0x01 0x00 0x3B 0x00 0x00 0x40 0x01 0x00 0x3C
0x00 0x00 0x40 0x01 0x00 0x3D 0x00 0x00 0x40 0x01 0x00 0x42 0x00 0x00
0x40 0x01 0x00 0x46 0x00 0x00 0x40 0x01 0x00 0x45 0x00 0x00 0x40 0x01
0x00 0x4C 0x00 0x00 0x40 0x01 0x00 0x8C 0x75
```

Вычислитель присылает в ответ на запрос стандартное подтверждение на запись:

```
0x00 0x10 0x3F 0xFF 0x00 0x00 0xFD 0xFC
```

Пример запроса на чтение данных в соответствии с записанным перечнем:

```
0xFF 0xFF 0x00 0x03 0x3F 0xFE 0x00 0x00 0x29 0xFF
```

Вычислитель присылает в ответ на запрос запрошенные «свойства»:

(Служебные байты: сетевой адрес, функция, длина всей секции данных)

```
0x00 0x03 0x4F
```

(секция свойств единиц измерения)

```
0x02 0x00 0xF8 0x43 0xC0 0x00 (°C для tTypeM)
```

```
0x04 0x00 0xAC 0x33 0x2F 0xE7 0xC0 0x00 (мЗ/ч для GTypeM)
```

```
0x03 0x00 0x20 0xAC 0x33 0xC0 0x00 (мЗ для VTypeM)
```

```
0x02 0x00 0x20 0xE2 0xC0 0x00 (т для MTypeM)
```

```
0x06 0x00 0xAA 0xA3 0x2F 0xE1 0xAC 0x32 0xC0 0x00 (кг/см2 для PTypeM)
```

```
0x04 0x00 0x83 0xAA 0xA0 0xAB 0xC0 0x00 (Гкал для QoTypeM)
```

```
0x01 0x00 0xE7 0xC0 0x00 (ч для QntTypeNIM)
```

```
0x01 0x00 0xE7 0xC0 0x00 (ч для QntTypeM)
```

(секция свойств количества знаков после запятой)

```
0x02 0xC0 0x00 количество знаков после запятой для tTypeFractDigNum;
```

```
0x02 0xC0 0x00 количество знаков после запятой для VTypeFractDigNum1;
```

```
0x02 0xC0 0x00 количество знаков после запятой для MTypeFractDigNum1;
```

```
0x02 0xC0 0x00 количество знаков после запятой для PTypeFractDigNum1;
```

```
0x03 0xC0 0x00 количество знаков после запятой для QoTypeFractDigNum1;
```

```
0x02 0xC0 0x00 количество знаков после запятой для MTypeFractDigNum2;
```

```
0x02 0xC0 0x00 количество знаков после запятой для VTypeFractDigNum2;
```

```
0x03 0xC0 0x00 количество знаков после запятой для QoTypeFractDigNum2;
```

(контрольная сумма)

```
0xB8 0x33
```

где:

■ длина строки;

■ символы строки, в соответствии с полем «длина»;

■ байт качества (анализировать не нужно);

■ байт нештатной ситуации (анализировать не нужно);

■ количество значков после запятой;

5.3 Порядок получения необходимых данных

За время сеанса связи система верхнего уровня должна не менее одного раза запросить «свойства» для определения наименований единиц измерения и количества знаков в дробной части для элементов данных. Далее нужно запросить перечень активных элементов, сформировать перечень элементов для чтения, и затем записать его в вычислитель. Из ответа на запрос «Чтение перечня активных элементов» для каждого элемента данных определяется его размер, то есть количество байт, которое нужно «взять» из полученного кадра, чтобы правильно интерпретировать значение. Затем выполняются запросы на запись типа значений и запись даты (при чтении архива), а затем и собственно «Чтение данных...». При обработке ответа на запрос «Чтение данных...» используется следующая информация:

- Размер элемента данных (получен при ответе на запрос «Чтение перечня активных элементов»);
- Количество знаков в дробной части и наименование единиц измерения (получены при чтении «свойств»).

5.4 Получение архивных данных

Для получения архивных данных за интересующую клиента дату необходимо выполнить приведенную ниже последовательность запросов:

- запрос на запись типа значений с номером 0-3 в зависимости от желаемого типа данных;
- запрос на чтение перечня активных элементов;
- запрос на запись перечня элементов для чтения;
- запрос на запись даты;
- запрос на чтение данных в соответствии с записанным перечнем элементов для чтения.

Если выполняется циклическое чтение архивных данных, то первые три операции нужно выполнить единожды.

ВНИМАНИЕ! ПРИБОР ПРИСЫЛАЕТ ОТВЕТ С ДАННЫМИ, ОТНОСЯЩИМИСЯ К ТОЙ ДАТЕ, КОТОРАЯ БЫЛА ЗАПИСАНА В ПОСЛЕДНИЙ РАЗ.

Если при записи даты прибор прислал ответ с кодом 3 (отсутствуют данные за указанную дату), то отчет следует формировать из записей архива другого типа. Например, если при чтении суточного архива был получен код 3, то для получения суточной записи за эти сутки нужно использовать данные в часовом разрезе за эти же сутки. Если при получении данных в соответствии с записанным перечнем элементов для чтения получен код 5, это означает, что зафиксировано изменение схемы измерения. В этом случае нужно следовать рекомендациям, изложенным в разделе «Принципы организации данных, доступных через сеть, в тепловычислителе ВКТ-7».

5.5 Получение текущих и итоговых текущих данных

Для получения текущих и итоговых текущих данных необходимо выполнить последовательность запросов:

- запрос на запись типа значений с номером 4 или 5 соответственно;
- запрос на чтение перечня активных элементов;
- запрос на запись перечня элементов для чтения;
- запрос на чтение данных в соответствии с записанным перечнем элементов для чтения.

Приложение А – Расчет контрольной суммы кадра Modbus

Расчет контрольной суммы кадра Modbus RTU может выполняться по следующему алгоритму (текст на языке программирования C):

```
WORD Crc16(BYTE *Data, ULONG size)
{
    union
    {BYTE b[2]; unsigned short w;} Sum;
    char shift_cnt;
    BYTE *ptrByte;
    ULONG byte_cnt = size;

    ptrByte=Data;
    Sum.w=0xffffU;
    for(; byte_cnt>0; byte_cnt--)
    {
        Sum.w=(unsigned short)
        ((Sum.w/256U)*256U+((Sum.w%256U)^(*ptrByte++)));
        for(shift_cnt=0; shift_cnt<8; shift_cnt++)
            /*обработка байта*/
            if((Sum.w&0x1)==1)
                Sum.w=(unsigned short)((Sum.w>>1)^0xa001U);
        else
            Sum.w>>=1;
    }
    return Sum.w;
}
```

Пусть какое-либо сообщение, имеющего длину N, записано в массиве Data[N+2] типа unsigned char. Тогда для этого сообщения контрольную сумму следует формировать следующим образом:

```
WORD CheckSumm = Crc16(Data, N);
Data[N] = CheckSumm;
Data[N+1] = CheckSumm>>8;
```

Приложение Б – Используемые типы данных и символные константы

При реализации протокола обмена возможно использование типа данных VT_DATA_RAP — 4 байта (день, месяц, год, час).

При расшифровке кодов в байте качества элементов данных можно использовать следующие символные константы:

- #define OPC_QUALITY_GOOD 0xC0
- #define OPC_QUALITY_BAD 0x00
- #define OPC_QUALITY_CONFIG_ERROR 0x04
- #define OPC_QUALITY_DEVICE_FAILURE 0x0C
- #define OPC_QUALITY_UNCERTAIN 0x40
- #define OPC_QUALITY_SENSOR_CAL 0x50