Системы измерительные вибрационного контроля

L-ViMS и ВИБ-4

Описание программного обеспечения

Ревизия 0.0.5
Март 2025

Автор руководства:
Борисов Алексей

ООО “Л Кард”
117105, г. Москва, Варшавское ш., д. 5, корп. 4, стр. 2

тел.: +7 (495) 785-95-25
факс: +7 (495) 785-95-14

Адреса в Интернет:
http://www.lcard.ru

E-Mail:
Отдел продаж: sale@lcard.ru
Техническая поддержка: support@lcard.ru
Отдел кадров: job@lcard.ru

L-ViMS и ВИБ-4 © Copyrigh 2025, ООО “Л Кард”. Все права защищены.

Таблица 1: Ревизии текущего документа

Ревизия	**Дата**	**Описание**
0.0.1	20.09.2023	Предварительная версия с описанием регистров modbus.
0.0.2	26.07.2024	• В регистры modbus добавлена информация о времени модуля, состоянии дискретных входов и выходов, диагностикая информация, информация о времени и событии сохранения выборки, а также о сохранении идентификторов последних 8 выборок. • Добавлены команды управления дискретными выходами, команды работы с конфигурацией модуля, расширены параметры команды запроса выборки. • Добавлено описание получения непрерывного потока отсчетов по OPC UA. • Добавлена общая информация о возможностях модуля, сетевых настройках, способах организации сети.
0.0.3	15.11.2024	Исправление формулы пересчета кодов отсчетов в электрические и физические величины, исправление названий элементов адресного пространства OPC UA
0.0.4	07.03.2025	Добавлено описание получения измерений и состояния дискретных выходов и входов, а также управления дискретными выходами по протоколу OPC UA. Добавлена информация об обновлении встроенного ПО модуля.
0.0.5	12.03.2025	Добавлена информация о типе канала “Осевое смещение”. В флагах состояния измерения добавлены признаки выхода значения измерения за границы настроенного диапазона.

Оглавление

1 Общие сведения **5**
1.1 Основные возможности модулей системы . . . 5
1.1.1 Общие возможности . . . 5
1.1.2 Измерительные каналы . . . 5
1.1.2.1 Обработка данных и расчет измерений . . . 6
1.1.2.2 Определение признаков тревог . . . 8
1.1.2.3 Порог на изменение значения измерения . . . 10
1.1.3 Дискретные входы . . . 10
1.1.4 События . . . 10
1.1.5 Дискретные выходы . . . 11
1.2 Варианты организации сети системы . . . 13
1.2.1 Использование модулей в сети общего назначения . . . 13
1.2.2 Использование сети на основе модулей SWITCH . . . 14
1.2.3 Использование сети на основе модулей SWITCH с сервером времени 15
1.3 Индикация светодиодов модуля . . . 16
1.4 Первое подключение модуля . . . 16
1.5 Сброс настроек модуля . . . 17
1.6 Обновление встроенного программного обеспечения модуля . . . 18
1.7 Сетевые настройки модуля . . . 19
1.8 Автоматическое обнаружение модулей в локальной сети . . . 20
1.9 Синхронизация времени по протоколу PTP . . . 21
1.10 Организация прямого обмена между модулями . . . 21

2 Описание протокола обмена модулями **22**
2.1 Протокол Modbus TCP . . . 22
2.1.1 Адресное пространство регистров Modbus . . . 22
2.1.1.1 Используемые типы и обозначения . . . 22
2.1.1.2 Регистры адресного пространства Modbus . . . 24
2.1.1.2.1 Блок регистров с информацией о модуле . . . 24
2.1.1.2.2 Блок регистров с состоянием модуля и каналов . . . 25
2.1.1.2.3 Блок регистров для чтения выборок . . . 29
2.1.1.2.4 Блок регистров для передачи команд . . . 31
2.1.1.3 Составные типы, используемые при описании адресного пространства . . . 33
2.1.1.3.1 Значение времени TSVAL . . . 33
2.1.1.3.2 Информация о метке времени TSINFO . . . 34
2.1.1.3.3 Информация о внутреннем локальном времени TSLOC . . . 34
2.1.1.3.4 Информация о измерении диагностического АЦП DGADC . 35
2.1.1.3.5 Регистры состояния измерительного канала CHST . . . 35
2.1.2 Чтение выборок через регистры Modbus . . . 37

2.1.3 Выполнение команд через регистры Modbus . . . 39
2.1.3.1 Команда установки значения признака измерительного канала . 39
2.1.3.2 Запрос сохранения выборки . . . 40
2.1.3.3 Сброс выбранных дискретных выходов . . . 41
2.1.3.4 Ручное изменение состояния выбранных дискретных выходов . . 42
2.1.3.5 Чтение информации о конфигурации режима работы модуля . . 43
2.1.3.6 Чтение блока данных конфигурации режима работы модуля . . 44
2.1.3.7 Начало передачи новой конфигурации режима работы в модуль 44
2.1.3.8 Передача блока новой конфигурации режима работы в модуль . 45
2.1.3.9 Завершение передачи новой конфигурации режима работы в модуль . . . 45
2.1.3.10 Обновление текущей конфигурации режима работы . . . 46
2.1.3.11 Сохранение конфигурации режима работы в энергонезависимую память . . . 46
2.2 Протокол OPC UA . . . 47
2.2.1 Адресное пространство OPC UA . . . 47
2.2.2 Особенности передачи меток времени . . . 53
2.2.3 Чтения потока сырых отсчетов . . . 54
2.3 Формат данных конфигурации режима работы модуля . . . 56
2.3.1 Блок общих настроек измерительного канала . . . 57

3 Предоставляемое ПО **58**
3.1 ViConfigurator . . . 58
3.2 ViView . . . 58

Глава 1

Общие сведения

1.1 Основные возможности модулей системы

Данный раздел тезисно описывает основные возможности модулей системы L-ViMS.

1.1.1 Общие возможности

- Каждый модуль системы может использоваться как отдельное независимое устройство, так и в составе системы (раздел 1.2).
- Каждый модуль является отдельным устройством в сети со своим IP-адресом, который должен быть настроен (раздел 1.4) перед подключением в рабочую сеть. Это относится в том числе и к модулю L-ViMS-SWITCH.
- Каждый модуль имеет контроллер для обработки данных и поддержки интерфейсов обмена, встроенное ПО которого может быть обновлено по сети (раздел 1.6).
- Каждый модуль имеет свою конфигурацию режима работы, которая определяет опрашиваемые каналы, обработку данных, логику генерации событий, управления выходами и т.д. Данная конфигурация хранится в энергонезависимой памяти и после ее записи модуль при старте автоматически загружает ее (в том числе и после сброса питания) и начинает работу в соответствии с ней как автономное устройство.
- Каждый модуль поддерживает на аппаратно-программном уровне синхронизацию времени по протоколу PTP (IEEE 1588v2) в варианте gPTP (IEEE 802.1AS) (раздел 1.9).

1.1.2 Измерительные каналы

Модули L-ViMS-ICP и L-ViMS-NPS имеют по четыре измерительных канала со следующими возможностями:

- Все каналы АЦП в рамках одного модуля работают параллельно и синхронно.
- Все каналы АЦП модуля имеют фиксированную частоту, зависящую от модификации модуля: 128 КГц для модификаций L-ViMS-ICP-10-1, L-ViMS-ICP-4-1 и L-ViMS-NPS-1 и 46,667 КГц для остальных.

- Модули L-ViMS-ICP и L-ViMS-NPS отличаются типами датчиков, которые можно подключить:
  - модуль L-ViMS-ICP поддерживает подключение датчиков с выходом по напряжению с питанием положительной полярности или ICP-датчики. Тип датчика настраивается индивидуально для каждого канала в конфигурации режима работы модуля. Ток питания ICP-датчиков зависит от модификации модуля и не настраивается.
  - модуль L-ViMS-NPS поддерживает подключение датчиков с выходом по напряжению с питанием отрицательной полярности.
- Для каждого канала поддерживается проверка действительности данных по следующими критериям:
  - Проверка, что измерения попадают в настроенный диапазон действительных значений выходного сигнала датчика.
  - Для ICP датчиков проверка на КЗ или обрыв по постоянной составляющей сигнала по настраиваемым порогам.
- Для каждого канала индивидуально настраивается коэффициенты перевода в физические величины.
- Для каждого канала может быть настроена своя обработка данных и вычисление до 8 измерений (раздел 1.1.2.1). Каждому измерению могут быть назначены пороги для определения двух уровней тревог (раздел 1.1.2.2).
- Цикл расчета измерений и обновления состояния канала не превышает 50 мс.
- Для каждого канала может быть назначен канал фазоотметчика, отметки фазы от которого могут использоваться для расчета параметров гармоник, а также могут быть вставлены в потоковые данные и данные выборок.
- Для каждого канала может быть включен по команде через интерфейс признак Bypass для отключения всех проверок на время проведения сервисного обслуживания канала.
- Для каждого канала могут быть установлены по команде через интерфейс или по назначенному событию признаки подавления тревог или умножения уставок.
- Для каждого канала может быть получен непрерывный поток отсчетов с признаками отметки фазы для полной обработки данных на ПК.
- Для каждого канала может быть сохранена в памяти выборка (непрерывный блок отсчетов данных за определенный интервал) по команде или по событию.

1.1.2.1 Обработка данных и расчет измерений

Список доступных измерений определяется настроенным типом канала. Поддерживаются следующие типы каналов и измерения:

- **Виброускорение**. Данный тип канала подразумевает подключение датчиков акселерометров. Доступны следующие измерения:

- **Ускорение**. Вычисление значения (СКЗ, размах или пик) виброускорения в м/с^2 в заданной полосе (верхняя и нижняя границы полосы настраиваемые в пределах от 2 Гц до 10 КГц, верхняя граница не может быть больше нижней более чем в 1000 раз).
- **Ускорение (ВЧ)**. Доступно только для модулей модификаций с частотой преобразования АЦП 128 КГц. Вычисление значения (СКЗ, размах или пик) виброускорения в настраиваемой полосе в границах от 1 КГц до 40 КГц.
- **Скорость**. Вычисление значения (СКЗ, размах или пик) виброскорости в мм/с в заданной полосе (верхняя и нижняя границы полосы настраиваемые в пределах от 2 Гц до 10 КГц, верхняя граница не может быть больше нижней более чем в 1000 раз). Вычисляется путем интегрирования сигнала ускорения.
- **Перемещение**. Вычисление значения (СКЗ, размах или пик) виброперемещения в мкм в заданной полосе (верхняя и нижняя границы полосы настраиваемые в пределах от 2 Гц до 10 КГц, верхняя граница не может быть больше нижней более чем в 1000 раз). Вычисляется путем двойного интегрирования сигнала ускорения.
- **Постоянное напряжение**. Вычисление постоянной составляющей сигнала в Вольтах.
- **nx Амплитуда ускорения**. Вычисление значения (СКЗ, размах или пик) выбранной гармоники сигнала ускорения. Частота гармоники определяется на основе данных фазоотметчика, который должен быть назначен каналу для включения данного измерения.
- **nx Фаза ускорения**. Вычисление значения фазы выбранной гармоники сигнала ускорения. Частота гармоники и точка нулевой фазы определяется на основе данных фазоотметчика, который должен быть назначен каналу для включения данного измерения. Фаза может быть вычислена только если для фазоотметчика с одним событием на оборот.
- **nx Амплитуда скорости**. Аналогично **nx Амплитуда ускорения**, но для сигнала виброскорости.
- **nx Фаза скорости**. Аналогично **nx Фаза ускорения**, но для сигнала виброскорости.
- **nx Амплитуда перемещения**. Аналогично **nx Амплитуда ускорения**, но для сигнала виброперемещения.
- **nx Фаза перемещения**. Аналогично **nx Фаза ускорения**, но для сигнала виброперемещения.

- **Радиальная вибрация**. Данный тип канала подразумевает подключение датчиков перемещения (проксиметров), например вихретоковых, для определения вибрации. Доступны следующие измерения:
  - **Перемещение**. Вычисление значения (СКЗ, размах или пик) виброперемещения в мкм в заданной полосе (верхняя и нижняя границы полосы настраиваемые в пределах от 2 Гц до 10 КГц, верхняя граница не может быть больше нижней более чем в 1000 раз).
  - **Зазор**. Вычисление постоянной составляющей сигнала в Вольтах, для определения значения зазора между датчиком и поверхностью.

  - **nx Амплитуда перемещения**. Вычисление значения (СКЗ, размах или пик) выбранной гармоники сигнала виброперемещения. Частота гармоники определяется на основе данных фазоотметчика, который должен быть назначен каналу для включения данного измерения.
  - **nx Фаза перемещения**. Вычисление значения фазы выбранной гармоники сигнала виброперемещения. Частота гармоники и точка нулевой фазы определяется на основе данных фазоотметчика, который должен быть назначен каналу для включения данного измерения. Фаза может быть вычислена только если для фазоотметчика с одним событием на оборот.
  - **Максимальное смещение от среднего (Smax)**. Вычисление максимального отклонения вала от среднего положения по двум координатам в мкм, соответствует параметры Smax по ГОСТ ИСО 7919-1. Для вычисления требует назначения второго канала этого же типа этого же модуля, для получения второй координаты.

- **Фазоотметчик**. Данный канал предполагает подключение датчиков виброперемещения или датчика оборотов. Канал служит для определения скорости вращения вала по специальным отметкам (выступ или впадина) и определения моментов времени их обнаружения. Полученные отметки фазы могут быть использованы в других каналах как для измерений, так и в выборках и потоке данных для сопоставления данных с отметками фазы. Доступны следующие измерения:
  - **Скорость вращения**. Вычисленная скорость вращения вала в оборотах в минуту.

- **Осевое смещение**. Данный тип канала подразумевает подключение датчиков перемещения (проксиметров), например вихретоковых, для определения изменения положения вала или другого объекта относительно заданной точки. Для канала задается уровень сигнала, который считается нулевым положением, а также какое направление изменения сигнала считается положительным отклонением, а какое отрицательным. Доступны следующие измерения:
  - **Позиция**. Вычисление отклонения измеренного положения поверхности относительно заданной точки с учетом настроенного положительного направления в микрометрах.
  - **Зазор**. Вычисление постоянной составляющей сигнала в Вольтах, для определения значения зазора между датчиком и поверхностью.

1.1.2.2 Определение признаков тревог

Для каждого рассчитываемого измерения могут быть заданы пороги (уставки) для определения двух уровней тревог: “Предупреждение” и “Останов”. Для каждого уровня задается:

- Условие сравнения с порогом
- Верхний порог
- Нижний порог

В зависимости от условия могут использоваться оба порога, только один из двух или не использоваться вообще. Доступны следующие условия:

- **Запрещено** — проверка тревоги не выполняется. Признак тревоги всегда в третьем состоянии (см. ниже).
- **Выше** — тревога, если значение измерения превышает верхний порог
- **Ниже** — тревога, если значение измерения ниже нижнего порога
- **Вне интервала** — тревога, если значение измерения выше верхнего порога или ниже нижнего
- **В интервале** — тревога, если значение измерения ниже верхнего и выше нижнего

При каждом получении измерения модуль проверяет условия для каждого уровня тревог. Эти значения доступны для чтения через интерфейс вместе со значениями измерения, а также используются в логике событий (раздел 1.1.4).

Значение признака для каждого уровня тревоги вычисляется независимо от другого и условия признаков могут быть настроены произвольным образом. Как правило "Останов" используется как более высокий уровень тревоги по сравнению с "Предупреждение". В этом случае для тревоги "Останов" используются более жесткие условия и при увеличении вибрации сперва становится активным признак "Останов", а затем становятся активны оба признака (в этом случае при активном признаке тревоги "Останов" всегда активен признак "Предупреждение").

При этом признак тревоги может принимать в действительности три значения:

- **Ложь** — отсутствие тревоги
- **Истина** — наличие тревоги
- **Z** — Третье состояние, которое условно можно назвать неопределенным. Признаки тревоги принимают это состояние при неисправности канала (КЗ или обрыв), а также если проверка условия тревоги запрещена. Это состояние специальным образом обрабатывается в логике событий (раздел 1.1.4).

Кроме того, индивидуально для каждого канала существуют дополнительные признаки управления определением тревог:

- **Неисправность канала** — при обнаружении неисправности канала, проверка признаков тревог отключается (принимают Z-состояние).
- **Bypass** — специальный признак, устанавливаемый пользователем по команде. Отключает как проверку всех тревог, так и проверку признака неисправности канала (принимают Z-состояние). Как правило используется для сервисных работ с каналом, таких как замена датчика.
- **Подавление тревог** — отключает проверку всех тревог для канала (все тревоги в состоянии Ложь) без отключения проверки признака неисправности. Может устанавливаться как пользователем, так и по настроенному событию.
- **Умножение уставок** — при активном признаке проверка тревог не отключается, но увеличиваются пороги для их определения, в заданное индивидуально для каждого измерения число раз. Может устанавливаться как пользователем, так и по настроенному событию.

1.1.2.3 Порог на изменение значения измерения

Для каждого измерения можно также настроить порог на изменение значения в процентах от заданной пользователем для этого измерения шкалы. Проверка данного порога выполняется следующем образом:

- Как только модулем было получено первое действительное значение для данного измерения, он сохраняет его как базовое.
- При получении последующих действительных значений модуль вычисляет отклонение этого значения от сохраненного базового
- Если отклонение не превышает заданный порог, то признак превышения порога изменения сбрасывается, если он был установлен ранее. Больше никаких действий не выполняется, базовое значение остается прежним.
- Если отклонение превышает порог, то устанавливается признак превышения порога изменения и текущее значение сохраняется в качестве базового. Таким образом последующие измерения уже будут сравниваться с новым базовым значением.
- Если обнаружена неисправность канала, то базовое значение сбрасывается. При неисправности канала отклонение не проверяется, а ожидается первое действительное значение в соответствии с первым пунктом.

Таким образом, если значение измерения постепенно изменяется, то при преодолении каждого интервала заданной величины будет активирован на один цикл сбора признак превышения порога изменения значения. Этот признак вместе с базовым значением и текущим отклонением от базового доступен для чтения по интерфейсу. Признак превышения порога изменения также может быть использован в логике событий (раздел 1.1.4).

1.1.3 Дискретные входы

Модули L-ViMS-ICP, L-ViMS-NPS и L-ViMS-SWITCH имеют по одному дискретному выходу. Состояние дискретного выхода может быть прочитано по интерфейсу, а также использовано в логике событий (раздел 1.1.4), в том числе и как признак подавления тревог или умножения уставок.

1.1.4 События

В модуле может быть настроено до 64 разных событий. Каждое событие может быть либо в активном, либо не в активном состоянии. Состояние определяется с помощью настроенной логики срабатывания события. Входными условиями для построения логики срабатывания могут служить следующие данные:

- Признаки измерений:
  - **Неисправность** — активно, если обнаружена неисправность канала, к которому относится измерение
  - **Предупреждение** — признак выполнения условия тревоги уровня “Предупреждение” для данного измерения

  - **Останов** — признак выполнения условия тревоги уровня “Останов” для данного измерения
  - **Порог отклонения** — признак превышения порога на изменение значения для данного измерения

- Состояние дискретного входа.
- Состояние другого события.

Эти признаки могут объединяться в логическое выражение с помощью стандартных логических элементов “И” (истина, только если оба входных признака истины), “ИЛИ” (истина, если хотя бы один признак истин), “НЕ” (изменение истины на ложь и наоборот). Общее суммарное количество использованных признаков и логических элементов в логике срабатывания одного события ограничено 512 элементами.

Особенным образом обрабатывается Z-состояние для признаков тревог: входы с таким состоянием не учитываются в логике. То есть, если значение на одном входе “И” или “ИЛИ” находится в третьем состоянии, то результат обоих этих элементов соответствует значению второго входа. Если на всех входах логического элемента значение Z, то и выход логического элемента соответствует Z. Если результат всего логического выражения Z, то событие находится в неактивном состоянии.

Состояние событий может быть прочитано по интерфейсу, а также использоваться для:

- Сохранения выборок
- Управления дискретными выходами
- Определения признаков подавления тревог и умножения уставок

1.1.5 Дискретные выходы

Каждый модуль системы L-ViMS имеет дискретные входы. Дискретные входы делаться на следующие типы, отличающиеся только электрическими характеристиками (логика управления идентична):

- **Слаботоковые реле**. Присутствуют на каждом модуле от 1 до 2.
- **Сильнотоковые реле**. Присутствуют на модуле L-ViMS-REL

Управления дискретными выходами выполняется в соответствии с настройками режима работы модуля следующим образом:

- У каждого выхода есть свое логическое состояние (*“активно”* – есть тревога или *“не активно”* — нет тревоги) и электрическое (запитано или обесточено). Может использоваться как прямая логика (запитано при тревоге или Normally De-Energized), так и инверсная (обесточено при тревоге или Normally Energized). Последнее используется в случае, когда отсутствие питания модуля считается тревогой (часто для выходов, использующихся как признак неисправности).
- Для каждого выхода может быть настроено начальное состояние, которое будет установлено после включения питания модуля.

- Каждый выход может быть настроен на ручное управления (по интерфейсу пользователем) или автоматическое по заданному событию.
- При автоматическом управлении могут быть индивидуально настроены задержка включения и выключения, т.е. выход переходит в активное состояние только после того как событие будет активно непрерывно в течение времени включения, и переходит в неактивное только если событие неактивно непрерывно в течение задержки выключения.
- При автоматическом управлении выход может быть настроен на режим с фиксацией. В этом случае при переходе в активное состояние выход остается в нем пока не будет сброшено по команде через интерфейс.
- Каждому выходу может быть назначено событие сброса. При его активации и до тех пор пока это событие активно, выход переходит в неактивное состояние вне зависимости от других условий.

1.2 Варианты организации сети системы

В данном разделе приведены разные варианты организации сети, в которой используются модули системы L-ViMS. На изображениях схемы организации сети будут использованы следующие обозначения:

- `IO` — модуль ввода-вывода (L-ViMS-ICP, L-ViMS-NPS, L-ViMS-REL).
- `SWITCH` — модуль L-ViMS-SWITCH.
- Сплошная синяя линия — физическое соединение по Ethernet.
- Пунктирная линия со стрелками — логический обмен данными:
  - серая — фоновый обмен ПК-модули
  - зеленая — обмен в реальном времени между модулями
  - красная — обмен между модулями и сервером реального времени

1.2.1 Использование модулей в сети общего назначения

Схема организации такой сети показана на рисунке 1.1.

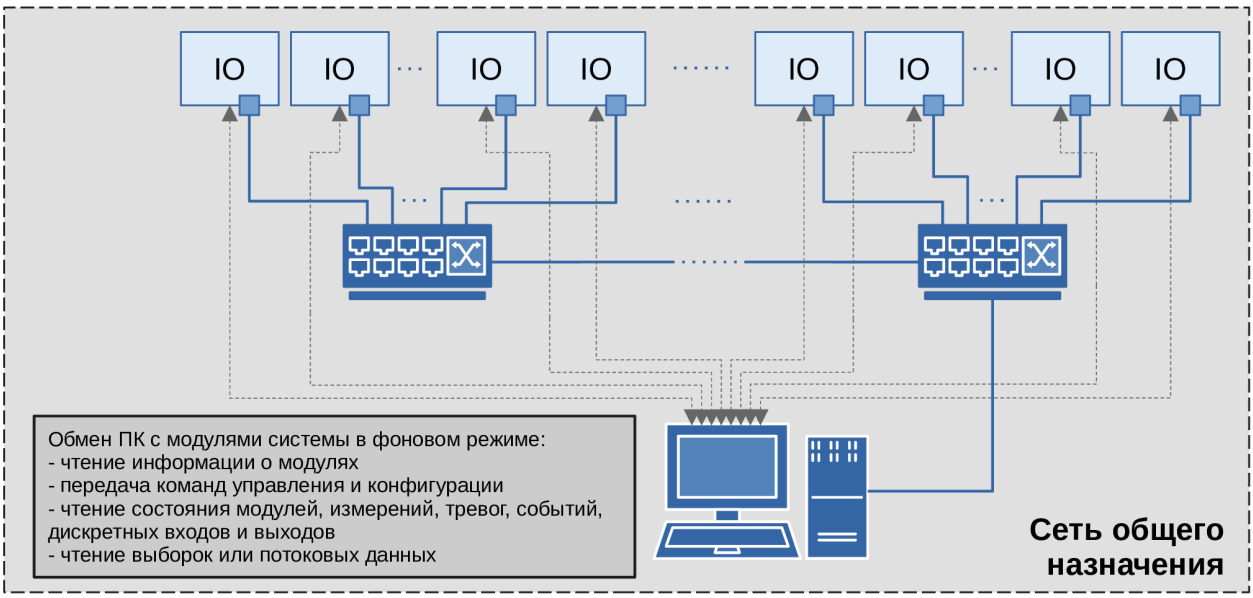

Рис. 1.1: Схема сети общего назначения на базе обычных коммутаторов

Модули системы L-ViMS могут использоваться как отдельные модули ввода вывода, подключенные напрямую к ПК или к локальной сети общего назначения, построенную на безе обычных коммутаторов.

В этом случае все управление случае взаимодействие возможно только между ПК и каждым модулем. Управление выходами модуля может осуществляться также либо с ПК, либо на основе измерений этого же модуля. Обмен данными между модулями напрямую не осуществляется.

Также при таком варианте организации сети не доступна синхронизация времени, привязка данных может осуществляться только на ПК к своему времени, на точность которой будут влиять задержки передачи сети и реакции ОС ПК.

1.2.2 Использование сети на основе модулей SWITCH

Схема организации такой сети показана на рисунке 1.2.

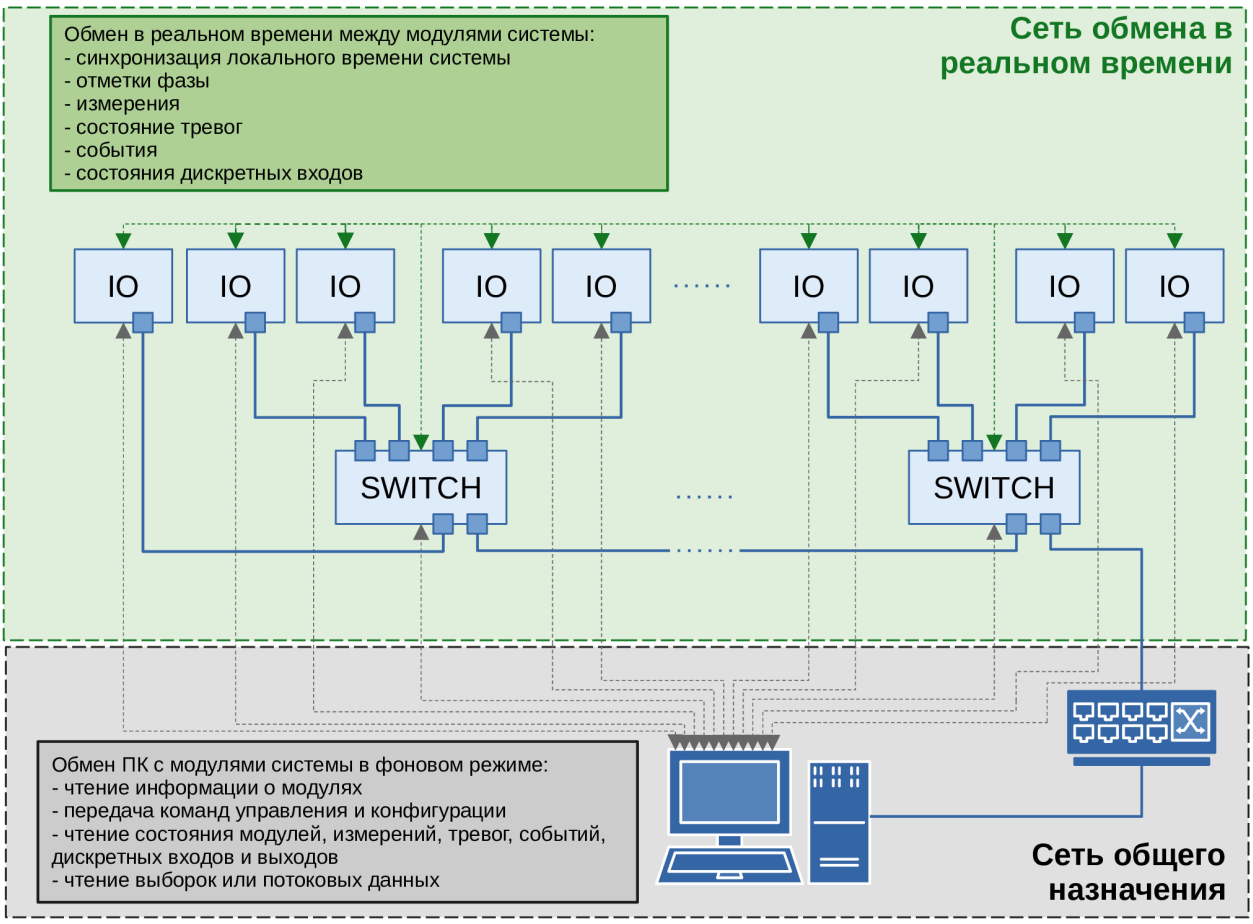

Рис. 1.2: Схема сети реального времени на основе модулей SWITCH

В данном варианте модули системы L-ViMS соединены между собой модулями L-ViMS-SWITCH, образуя сесть реального времени, где помимо обычных пакетов могут передаваться приоритетные передачи.

Пользовательские ПК могут подключаться либо напрямую к данной сети к одному из портов модуля L-ViMS-SWITCH они могут быть включены в сеть общего назначения, один из портов коммутатора которой соединен со сетью на базе модулей L-ViMS-SWITCH.

Для соединения модулей L-ViMS-SWITCH между собой и подключения ПК/общей сети можно использовать гигабитные порты, чтобы увеличить суммарную скорость канала передачи между ПК и модулями.

При организации сети следует избегать замкнутых кольцевых соединений.

В результате сегмент сети, построенный на модулях L-ViMS-SWITCH образует подсеть, в которой помимо передач общего назначения доступны приоритетные передачи напрямую между модулями, а также синхронизация времени между модулями.

Организованная таким образом сеть позволяет:

- использовать признаки тревог, состояния дискретных входов, событий одного модуля в логике событий других модулей системы

- использовать данные фазоотметчика одного модуля в измерительных каналах других модулей системы
- синхронизировать время между всеми модулями системы для сопоставления времени выборок и потоковых данных между модулями
- выполнять одновременное сохранение выборок разных модулей (по событию или запросу).

1.2.3 Использование сети на основе модулей SWITCH с сервером времени

Схема организации такой сети показана на рисунке 1.3.

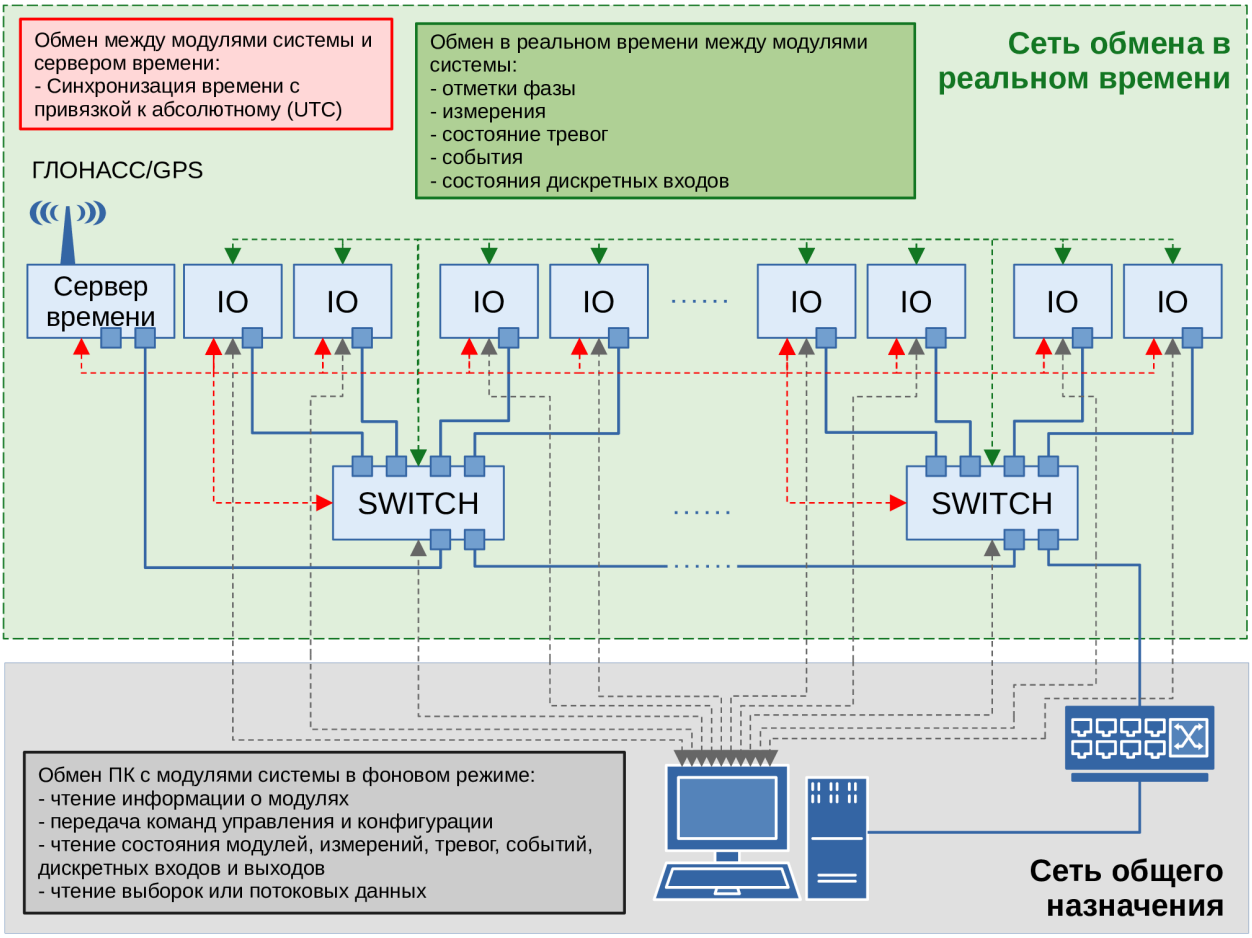

Рис. 1.3: Схема сети на основе модулей SWITCH с сервером времени

Данная схема соответствует предыдущей, но предполагает также подключение сервера времени с поддержкой протокола PTP к одному из модулей L-ViMS-SWITCH. Это позволяет синхронизировать время не только между модулями внутри системы, но и привязать это время к абсолютному.

В качестве сервера времени может использоваться сервер времени ЭНКС-2.

1.3 Индикация светодиодов модуля

Данный раздел будет дополнен в последующих версиях документации...

1.4 Первое подключение модуля

Перед подключением модуля в рабочую сеть, необходимо сконфигурировать корректно сетевые настройки модуля, чтобы модуль имел корректный IP-адрес в рамках используемой подсети, и этот адрес был уникальным в сети.

По умолчанию модулем используется IP-адрес 192.168.0.1. В случае, если IP-адрес у модуля был изменен и неизвестен, настройки модуля всегда могут быть сброшены на значения по умолчанию, как описано в разделе 1.5.

Для начальной настройки сетевых параметров модуль можно подключить напрямую к машине, на которой для сетевого порта настроен адрес из подсети 192.168.0.0/24, т.е. адрес сетевого порта должен быть 192.168.0.x, где x от 2 до 254.

После этого можно воспользоваться программой **ViConfigurator** (см. раздел 3.1). Для подключения необходимо выбрать пункт меню *“Подключение” → “Обзор устройств в сети”* (), в результате чего откроется окно обзора устройств в сети, как показано на рисунке 1.4.

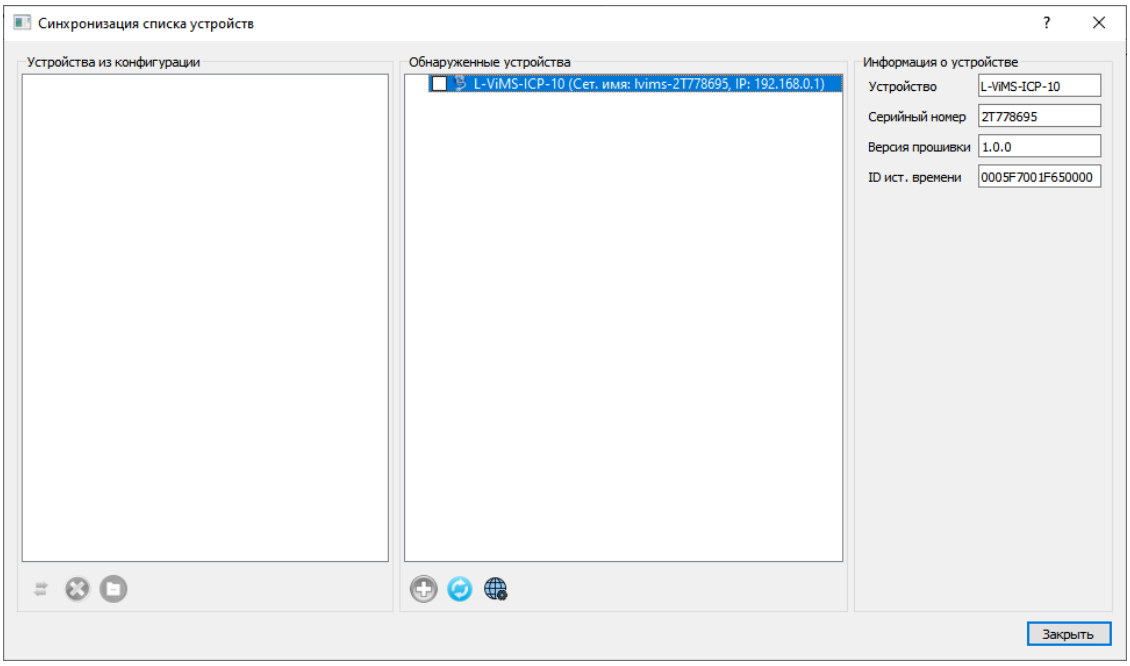

Рис. 1.4: Обзор устройств в сети

Модуль должен появится в списке обнаруженных устройств. Необходимо выбрать модуль в списке, после чего нажать на кнопку *“Настройки интерфейса устройства”* () под списком.

В результате откроется диалог настроек сетевых параметров модуля, как показано на рисунке 1.5.

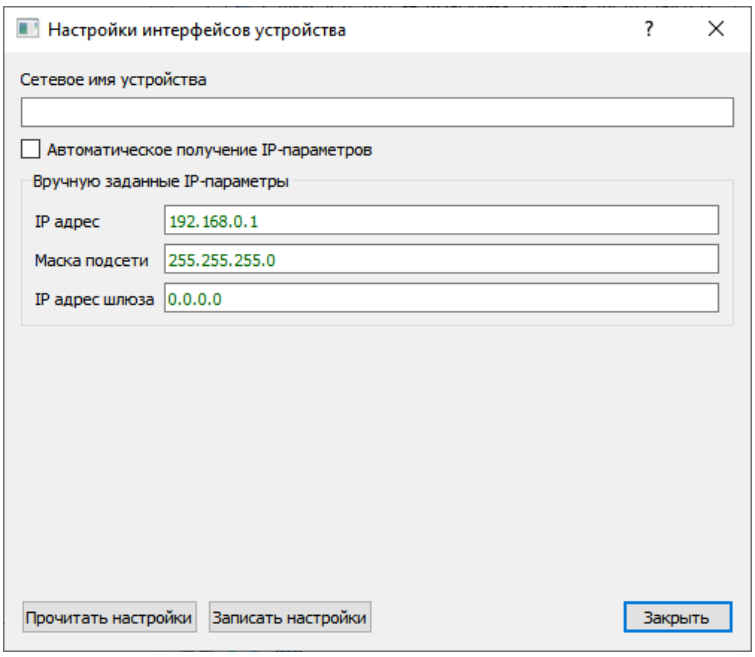

Рис. 1.5: Сетевые настройки модуля

После этого нужно изменить сетевые настройки на требуемые и нажать *“Записать настройки”*. Назначение настроек описано в разделе 1.7.

После успешной записи будет выведено соответствующее сообщение и программа вернется в окно обзора устройств в сети. Т.к. сетевые параметры модуля были изменены, то подключение к нему может быть уже недоступно с данного ПК, и модуль может не отображаться или отображаться с ошибкой подключения, если его адрес больше не принадлежит подсети 192.168.0.0/24.

Теперь модуль можно подключить в рабочую сеть и выполнить заданию его конфигурации режима работы в системе с помощью той же программы **ViConfigurator**.

1.5 Сброс настроек модуля

Модуль позволяет всегда сбросить настройки на значения по умолчанию. Для этого необходимо во время работы модуля зажать потайную кнопку *“Сброс”* на лицевой панели модуля и удерживать ее в течение 5 секунд, пока индикатор *“Раб”* не начнет быстро мигать оранжевым цветом, после чего кнопку *“Сброс”* можно отпустить.

В результате IP-адрес модуля будет сброшен на фиксированный 192.168.0.1, а также стерта рабочая конфигурация модуля. После этого можно подключиться используя известный IP-адрес и заново настроить модуль, как описано в разделе 1.4.

Также для экстренных случаев, если рабочая программа модуля по какой-то причине не доходит до проверки нажатия кнопки сброса настроек, предусмотрен специальный сброс настроек при старте с переходом в режим загрузчика. Для этого необходимо:

- снять питание с модуля
- зажать потайную кнопку *“Сброс”* на передней панели модуля
- подать питание на модуль при зажатой кнопки *“Сброс”*

- продолжать удерживать кнопку *“Сброс”* в течение 5 секунд, пока индикатор *“Раб”* не начнет быстро мигать оранжевым цветом
- отпустить кнопку *“Сброс”*

В результате этих действий модуль не перейдет к выполнению рабочей программы, а выполнит сброс настроек и останется в загрузчике, о чем будет свидетельствовать постоянно горящий оранжевым индикатор *“Раб”* и потухшие индикаторы *“Cm1”* и *“Cm2”*. После этого можно можно снять и подать питание заново для загрузки в рабочий режим со сброшенными настройками, либо выполнить обновление прошивки по сети.

1.6 Обновление встроенного программного обеспечения модуля

Модули системы L-ViMS поддерживают обновление встроенного программного обеспечения модуля по сети.

Последнюю версию встроенного ПО модуля можно скачать с сайта производителя. Файл встроенного ПО модуля имеет специальный формат, содержащий информацию о том, для каких типов модулей он предназначен, о версии встроенного ПО, а также цифровую подпись.

Во время обновления модуль переводится в специальный режим загрузочного устройства. В этом режиме модуль принимает и записывает переданный по сети файл с встроенным ПО в отдельную область внутренней энергонезависимой памяти, после чего проверяет целостность принятого файла, его подпись и соответствие типа модуля, и только после этого выполняет перезапись рабочей области со встроенным ПО. Это позволяет защитить модуль от попытки записать какое-либо стороннее ПО.

Также предварительная запись в отдельную энергонезависимую область позволяет сохранить работоспособность модуля в случае обрыва связи или пропадания питания во время процедуры передачи и обновления встроенного ПО модуля. В случае обрыва связи до завершения полной передачи и записи файла с новой версией встроенного ПО, по истечению времени ожидания передачи файла модуль снова перейдет в рабочий режим, запустив старую нетронутую версию рабочего встроенного ПО, после чего возможно заново повторить процедуру обновления.

В случае пропадания питания во время процедуры обновления возможно два сценария в зависимости от того, в какой момент это произошло. Если питание пропало во время передачи и записи новой версии прошивки до завершения записи, модуль после включения питания загрузится со старой версией рабочего ПО. Если пропажа питания была . Если же питание пропало во время записи уже переданного файла в рабочую область, то модуль после включения питания будет произведена повторная попытка записи нового ПО в рабочую область после чего будет запущена уже новая версия встроенного ПО.

Для обновления встроенного ПО модуля можно воспользоваться штатной программой **ViConfigurator** (раздел 3.1). Для этого необходимо выполнить следующие действия:

- Скачать файл с последней версией встроенного ПО модуля.
- Запустить программу **ViConfigurator**.
- Добавить в программе модуль в текущую конфигурацию.

- Выполнить подключение к модулю.
- Выбрать пункт меню *“Дополнительные действия”* → *“Запись прошивки из файла...”*.
- В появившемся диалоге выбрать скаченный файл с встроенным ПО модуля.
- В появившемся диалоге *“Обновление прошивки”* отметить в столбце *“Выб”* модули из конфигурации, для которых нужно обновить встроенное ПО, и нажать *“ОК”*.
- Дождаться завершения процедуры обновления.

1.7 Сетевые настройки модуля

Основными параметрами модуля для работы по Ethernet являются:

- IP-адрес устройства. Записывается как 4 цифры от 0 до 255 (модуль поддерживает только протокол IPv4), разделенные точками (например, 192.168.0.10). Состоит из адреса подсети, который должен быть одинаковый у всех взаимодействующих устройств, и адреса устройства внутри подсети, который должен быть уникальным в подсети.
- Маска подсети. Формат записи аналогичен IP-адресу. Определяет какая часть IP-адреса относится к адресу подсети, а какая к адресу устройства. Если предствить запись адреса и маски в двоичной записи, то те биты, для которых значение маски равно 1 будет относится к адресу сети, а 0 — к адресу устройства. При этом биты, задающие адрес подсети идут вначале. Например, маска 255.255.255.0 означает, что первые 3 цифры IP-адреса (192.168.0) обозначают адрес подсети, последняя цифра (10) — адрес устройства в подсети. Для записи адреса подсети с маской в данном документе используется также стандартная запись, в которой вместо маски указывается через слеш число бит, отвечающее за задание подсети, т.е. адрес подсети 192.168.12.0 с маской 255.255.255.0 может быть записан как 192.168.12.0/24.
- IP-адрес шлюза. Используется только когда модуль и ПК, с которого выполняется управление модулем, находятся в разных подсетях. Модуль передает пакеты по адресу шлюза, если адрес назначения находится не в той же подсети, что и модуль. При работе в одной локальной подсети не используется.
- Имя экземпляра устройства. Уникальное имя данного экземпляра в виде строки (до 64 английских символов или 32 русских). Используется для возможности автоматического обнаружения модулей в локальной сети (раздел 1.8), а также как идентификатор модуля по умолчанию, при прямом взаимодействии между модулями (раздел 1.10), если не задано имя модуля в конфигурации режима работы модуля.

IP-адрес, маска подсети и адрес шлюза могут быть заданы несколькими способами:

- Статические значения, заданые вручную. Пользователь сам должен позаботиться о том, чтобы адрес принадлежал нужной подсети и был уникальный в ней.

- Получены автоматически от DHCP-сервера. Если задано автоматическое получение адреса и в локальной сети присутствует DHCP-сервер, то модуль делает запрос к нему и использует выделенные DHCP-сервером IP-адреса.

- Может использоваться автоматически получаемый локальный (link-local) адрес (в соответствии с RFC3927). Адрес выбирается случайным образом в диапазоне от 169.254.1.0 до 169.254.254.255 и проверяется, что в сети нет другого устройства с таким адресом (если есть, то идет попытка выбора следующего адреса и т.д.). Это делает возможным подключение к устройству в локальной сети без специальной конфигурации. Следует однако отметить, что т.к. адрес действителен только в одной сети, то два разных устройства в разных сетях могут иметь одинаковый link-local адрес, что приводит к тому, что если у ПК несколько активных сетевых интерфейсов (и на обоих используется link-local адрес или наоборот обычный адрес), то хост не знает на каком интерфейсе искать нужное устройство. Т.е. для подключения по link-local адресу на ПК должен быть либо один активный сетевой интерфейс, либо на нужном интерфейсе должен использоваться link-local адрес, а на другом статический или полученный по DHCP адрес.

При включении автоматического получения адреса модуль выбирает себе link-local адрес (и проверяет его уникальность), параллельно выполняя поиск в сети DHCP-сервера. Если DHCP сервер не обнаружен, то используется link-local адрес. Как только будет обнаружен DHCP-сервер, то предпочтение отдается полученному от него адресу (в частности при старте в сети с DHCP сервером модуль может некоторое время до получения адреса от него использовать link-local адрес). Подобный алгоритм используется при автоматическом получении адреса в частности в ОС Windows, а также и во многих дистрибутивах Linux (иногда предоставляя возможность отдельного разрешения DHCP и link-local адреса). Следует также иметь ввиду, что автоматически получаемый адрес модуль проверяет на уникальность в сети, поэтому существует задержка в несколько секунд от подключения к сети модуля до назначения адреса. Автоматическое получение адреса не требует дополнительных настроек, однако при этом неизвестен адрес устройства со стороны ПК для установления соединения. Для решения этой проблемы возможно использование процедуры поиска устройств в локальной сети, описанной разделе разделе 1.8.

По умолчанию модуль использует статический адрес 192.168.0.1 с маской подсети 255.255.255.0, который может быть переназначен, как описано в разделе 1.4. Сетевые настройки модуля всегда могут быть сброшены на значения по умолчанию, как описано в разделе 1.5.

1.8 Автоматическое обнаружение модулей в локальной сети

Модули системы L-ViMS поддерживают протокол для обнаружения в локальной сети.

В частности, предоставляемые программы **ViConfigurator** (раздел 3.1) и **ViView** (раздел 3.2) поддерживают автоматическое обнаружение и подключение устройств в системы в локальной сети.

Для обнаружения модулей используется протокол mDNS/SD (в соответствии с RFC6762 и RFC6763). В соответствии с ним, каждое устройство при подключении объявляет набор сервисов, который оно поддерживает.

Чтобы различать экземпляры устройств, поддерживающие одинаковый тип сервисов, у каждого экземляра есть свое уникальное имя. По умолчанию используется имя вида *“<тип системы>-<серийный номер модуля>”*, где значение <тип системы> соответствует *“lvims”* или *“vib4”*. Данное имя может быть задано вручную пользователем в сетевых настройках модуля (раздел 1.7), например в соответствии с назначением конкретного модуля в системе, для более наглядной идентификации. Кроме имени экземпляра у каждого сервиса может быть набор текстовых параметров, описывающих данный экземпляр (для модулей системы L-ViMS указываются идентификационные данные типа модуля, серыйнй номер, версия прошивки и т.д.).

При запуске каждый модуль проверяет, что его имя не занято, и если в сети уже присутствует устройство с этим именем, то будет назначено новое с суффиксом *“-<кол-во повторов имени>”*, таким образом имя в любом случае остается уникальным даже в случае ошибки конфигурирования.

Модулями используются следующие имя службы:

- *“_lcard-vims-mb._tcp”* — для модулей системы L-ViMS.
- *“_kd-vib4-mb._tcp”* — для модулей системы ВИБ-4.

Данный раздел будет дополнен в последующих версиях документации...

1.9 Синхронизация времени по протоколу PTP

Данный раздел будет дополнен в последующих версиях документации...

1.10 Организация прямого обмена между модулями

Данный раздел будет дополнен в последующих версиях документации...

Глава 2

Описание протокола обмена модулями

2.1 Протокол Modbus TCP

Модули системы поддерживают стандартный протокол Modbus TCP. Для подключения используется стандартный для данного протокола TCP-порт — 502. Модуль поддерживает несколько одновременных соединений от клиентов по Modbus.

Протокол Modbus-TCP широко распространен и поддерживается в том числе многими SCADA системами, промышленными контроллерами и другими устройствами, что позволяет относительно несложно реализовать получение данных от модуля в подобных системах штатным образом.

Протокол Modbus-TCP не рассчитан на передачу потоков данных, поэтому он не позволяет вычитывать непрерывный поток данных исходных отсчетов АЦП измерительных модулей неограниченной длины, а только сохраненный блок данных за определенное время (выборка). Для получения непрерывного потока данных можно использовать протокол OPC UA .

2.1.1 Адресное пространство регистров Modbus

2.1.1.1 Используемые типы и обозначения

Адреса регистров при описании блоков регистров в адресном пространстве указаны исходя из начала нумерации от 1 в соответствии со спецификацией Modbus. Однако некоторые библиотеки и программы использует нумерацию регистров, начиная от 0. При использовании таких библиотек или программ необходимо вычитать 1 из указанного в данном документе значения адреса.

Для описания регистров используются следующие типы данных:

- **U16** — 16-битное беззнаковое целое значение, занимающее один регистр.
- **U16S** — 16-битное беззнаковое значение, которое используется для передачи значения в процентах от шкалы. Для всех значений этого типа код 0xFFFF соответствующий значению 100%.
- **U32** — 32-битное беззнаковое целое значение, занимающее два регистра. По указанному адресу хранятся младшие 16 бит, по адресу + 1 — старшие 16 бит.
- **I32** — 32-битное знаковое целое значение в дополнительном коде, занимающее два регистра. По указанному адресу хранятся младшие 16 бит, по адресу + 1 — старшие 16 бит.

- **float** — 32-битное значение с плавающей точкой. Занимает два регистра (с указанным адресом и следующим адресом). По указанному адресу хранятся младшие 16 бит, по адресу + 1 — старшие.

- **double** — 64-битное значение с плавающей точкой. Занимает четыре регистра, начиная с указанного адреса. По указанному адресу хранятся младшие 16 бит, по адресу + 3 — старшие.

- **STR[N]** — строка из UTF-8 символов (совпадает с ASCII для латинских символов), максимальной длины в N байт, занимающей подряд N/2 регистров, начиная с указанного адреса. Конец строки обозначается символом с нулевым кодом. Младший байт первого регистра содержит первый символ, старший байт — второй, младший байт второго регистра — 3-ий символ и т. д.

- **U8[N]** — массив байт максимальной длины в N байт, занимающей подряд N/2 регистров, начиная с указанного адреса. Младший байт первого регистра содержит первый байт массива, старший байт — второй, младший байт второго регистра — 3-ий т. д.

- **TYPENAME** — подчеркивание используется для обозначения составного типа — т.е. набора регистров, каждый из которых описан в отдельном разделе в отдельной таблице с указанием адресов относительно начала набора. Используется, когда блок одинаковой структуры может присутствовать в разных местах адресного пространства, для избежания повторения описания.

Значение логических признаков, которые могут иметь третье состояние (tribool), кодируются кодируется 2-мя битами:

- 00 — ложь

- 01 — истина

- 10 — Z (отсутствие действительного значения)

Таким образом, младший бит флага можно также использовать как признак активности флага, а старший как признак действительности значения.

Для адресов, заданных формулой, используются следующие обозначения:

- I — номер измерительного канала модуля (0-3)

- S — номер отсчета выборки относительно заданного отсчета (0-61)

- M — номер измерения (0-9)

- E — номер события (0-63)

Столбец «Аттр» указывает, какие операции возможны с этим регистром (R — чтение, W — запись).

2.1.1.2 Регистры адресного пространства Modbus

2.1.1.2.1 Блок регистров с информацией о модуле

Адрес	Название	Тип	Аттр	Описание
1-16	DEVNAME	STR32	R	Полное название модуля латинскими буквами, включая модификации. Название начинается с обозначения системы, к которой относится модуль: с префикса “L-ViMS-” или “VIB4-”. Примеры названий: “L-ViMS-NPS”, “VIB4-ICP-10-1”.
17-32	SERIAL	STR[32]	R	Серийный номер модуля.
33-40	FWVER	STR[16]	R	Строка с версией прошивки.
41-43	-	-	R	Резерв.
44	SYSTEM_TYPE	U16	R	Кодовое обозначение варианта системы: • 0 — L-ViMS • 1 — ВИБ-4
45	MODULE_TYPE	U16	R	Кодовое обозначение типа модуля: • 2 — ICP-10 • 3 — REL • 4 — NPS • 5 — SWITCH • 6 — ICP-4
46	MODIFICATION	U16	R	Кодовый номер модификации модуля: • 0 — без суффикса модфикации • 1 и выше — число в суффиксе модфикации в названии модуля. Например, значение 1 для ICP-10-1 и NPS-1.
47	REVISION	U16	R	Младшие 8 бит — код ревизии платы. Старшие 8 бит — резерв.
48	PROTOCOL_ID	U16	R	Код протокола обмена. Для описываемых систем всегда равен 0x5649.
49-96	-	-	-	Резерв.
97	DEVMODE	U16	R	Режим работы устройства: • 1 — режим загрузчика • 2 — рабочий режим • 4 — режим обслуживания
98	LAST_WR_ERR	U16	R	Код ошибки для последней операции записи. Значение обновляется в случае, если операция записи регистров была завершена с ошибкой и позволяет узнать причину ошибки.
256-259	MODULE_CLK_ID	U8[8]	R	Идентификатор модуля при синхронизации времени по PTP

2.1.1.2.2 Блок регистров с состоянием модуля и каналов

Адрес	Название	Тип	Аттр	Описание
Состояние модуля				
1024-1025	ST_FLAGS	U32	R	Флаги состояния модуля: • LIVEBIT (бит 0) — Бит проверки корректной работы модуля и интерфейса связи. При корректной работе меняет свое значение раз в секунду на противоположное. • FAULT (бит 1) — Признак наличия глобальной неисправности модуля. Информация о типе обнаруженной ошибки доступна в регистрах DEVFAULT_ERRS. • WARN (бит 2) — Признак наличия обнаруженных ошибок, не связанных с неисправностью аппаратуры самого модуля. Информация о типе обнаруженной ошибки доступна в регистрах DEVFAULT_WARNS.
1026	ST_OPCFG_ID	U16	R	Идентификатор последней записанной конфигурации режима работы модуля. Увеличивается на 1 после записи новой конфигурации. Сохраняет свое значение при сбросе питания модуля
1027	-	-	-	Резерв
1028	ST_WF_RDY_CNT	U16	R	Количество сохраненных в оперативной памяти модуля выборок, доступных для чтения в данный момент времени.

1029-1036	ST_WF_LAST_ID	U16[8]	R	Индентификаторы до восьми последних восьми сохраненных выборок. При сохранении новой выборки ее идентификатор сохраняется в регистр с маладшим адресом, а ранее сохраненные идентификаторы семи выборок смещаются в соответствующие регистр с адресом увеличенным на 1. Таким образом, в регистре с младшим адресом всегда хранится индификатор последней сохраненной выборки, в следующем предпоследней и т.д. Если значение регистра ST_WF_RDY_CNT меньше 8, то в данном массиве действительны только первые ST_WF_RDY_CNT элементов. Следует отметить, что в общем случае порядок завершения сохранения выборок и порядок поступления запросов на сохранение выборок может не совпадать, например, в случае, если был запрос сохранения выборки с явно указынным временем в будущем.
1037-1045	-	-	-	Резерв
1046-1061	CUR_TIME	TSINFO	R	Текущее (на момент обработки запроса чтения) синхронизированное время модуля
1062-1069	LOC_TIME	TSLOC	R	Текущее (на момент обработки запроса чтения) локальное время модуля
1070-1093	-	-	-	Резерв
1094-1101	ST_DIN	U16[8]	R	Состояние дискретных входов. Каждому входу соответствует 2 бита в регистре, каждый регистр содержит состояние 8 входов. Состояние входа кодируется как логический признак с тремя состояниями (tribool).
1102-1109	ST_EVENTS	U16[8]	R	Состояние событий. Каждому событию соответствует 2 бита в регистре, каждый регистр содержит состояние 8 событий. Состояние события кодируется как логический признак с тремя состояниями (tribool).
1110-1141	-	-	-	Резерв

Состояние дискретных выходов				
1280-1343	ST_DOUT_LP	U16[64]	R	Состояние слаботоковых реле. Каждый 16-битовый регистр описывает состояние одного выхода как комбинация следующих флагов: • FAILURE (бит 0-1) — признак неисправности выхода (tribool) • ENERGIZED (бит 2-3) — признак электрического состояния выхода (запитано ли реле) (tribool) • STATE (бит 4-5) — признак логического состояния выхода (наличие тревоги) (tribool) • CTL_EVT (бит 6-7) — состояние управляющего события (tribool) • RST_EVT (бит 8-9) — состояние события сбраса (tribool)
1344-1407	ST_DOUT_HP	U16[64]	R	Состояние сильнотоковых реле. Значение полей аналогично ST_DOUT_LP.
Диагностическая информация о состоянии модуля				
3584-3591	DEVFAULT_ERRS	U16[8]	R	Результат самодиагностики аппаратуры модуля на критические ошибки. Поле представляет собой набор битов, где каждый бит соответствует обнаружению одной из критических ошибок.
3591-3599	DEVFAULT_WARNS	U16[8]	R	Результат самодиагностики некритических ошибок состояния модуля или подключения датчиков.
3600-3615	-	-	-	Резерв
3616-3695	DIAGN_ADC	DGADC [16]	R	Каждый элемент соответствует состоянию измереня одного из диагностических АЦП: • 0 — измерение внутреннего напряжения 3.3 В • 1 — измерение температуры микроконтроллера в °C • 2 — напряжение питания модуля (В) • 8-11 — Для модуля L-ViMS-REL — дигностическое напряжение сильнотоковых реле

Состояние измерительных каналов				
$4096 + 1024*I$	`CH_STATE`	CHST	R	Набор регистров, описывающих текущее состояние соответствующего измерительного канала. Состав регистров описан в таблице ниже.

2.1.1.2.3 Блок регистров для чтения выборок

Адрес	Название	Тип	Аттр	Описание
Выбор читаемой выборки и ее данные				
61452	WF_SEL_ID	U16	RW	Выбор идентификатора выборки для чтения.
61453	WF_SEL_CH	U16	RW	Выбор номера канала в выборке для чтения.
61454-61455	WF_SEL_OFFS	U32	RW	Выбор номера первого отсчета выборки в доступном для чтения блоке данных.
61456 + 2*S	WF_RD_DATA	I32[62]	R	Значение отсчета с номером (S + WF_SEL_OFFS) в выборке с идентификатором WF_SEL_ID для канала WF_SEL_CH. S может быть от 0 до 61. Отсчет представляет собой 32-битное значение в дополнительном коде. Для перевода кода в Вольты и физические величины используются коэффициенты из WF_CHINFO_K_E/WF_CHINFO_B_E и WF_CHINFO_K_P/WF_CHINFO_B_P соответственно. Младшие 4 бита кода используются для передачи дополнительных флагов: • PHASETRIG (бит 0) — признак отметки фазы. Указывает, что для этого отсчета было обнаружено событие отметки фазы в назначенном канале отметчика фазы
Общая информация о выборке				
61696	WF_INFO_ID	U16	R	ID выборки, чтение которой выполняется (значение, записанное в WF_SEL_ID).
61697	WF_INFO_SRC	U16	R	Код, определяющий по какой причине была сохранена выборка. • 0-63 — выборка сохранена по событию с соответствующим номером • 65535 — выборка сохранена по пользовательской команде
61698-61699	WF_INFO_FREQ	float	R	Номинальная частота отсчетов выборки (Гц)
61700-61703	WF_INFO_FREQ_GM	double	R	Частота отсчетов выборки в пересчете на частоту часов источника синхронизации времени

61704-61719	WF_INFO_TS	TSINFO	R	Время сохранения выборки. Соответствует времени отсчета с номером WF_INFO_SIZE - WF_INFO_DELAY - 1, считая от 0. Т.е. если WF_INFO_DELAY равен 0, то последнему отсчету в выборке.
61720-61727	-	-	R	Резерв
61728	WF_INFO_FLAGS	U16	R	Флаги с информацией о выборке (резерв)
61729	WF_INFO_TYPE	U16	R	Тип выборки: • 1 — временная выборка
61730-61731	-	-	R	Резерв
61732-61733	WF_INFO_CH_MASK	U32	R	Битовая маска, указывающая, по каким каналам доступны данные для выборки. Каждому каналу соответствует совой бит (младший бит — 1-ый канал). Если бит установлен, то данные по соответствующему каналу присутствуют.
61734-61735	WF_INFO_SIZE	U32	R	Количество отсчетов для каждого канала в выборке.
61736-61737	WF_INFO_DELAY	U32	R	Количество отсчетов, на которое сохранение выборки задержано относительно момента запроса. Если 0, то последний отсчет соответствует времени выборки.
61738-61743	-	-	R	Резерв
Информация о выбранном канале выборки				
61952	WF_CHINFO_ID	U16	R	ID выборки, чтение которой выполняется (значение, записанное в WF_SEL_ID).
61953	WF_CHINFO_CH	U16	R	Номер канала, которому соответствует информация (значение, записанное в WF_SEL_CH).

61954-61955	WF_CHINFO_FLAGS	U32	R	Битовая маска флагов информации о состоянии канала: • CH_ENABLED (бит 0) — Бит указывает, что данный канал разрешен. Должен соответствовать значению соответствующего бита регистра WF_INFO_CH_MASK в информации о выборке. Если 0, то вся дальнейшая информация не действительна.
61956-61957	WF_CHINFO_K_E	float	R	Коэффициент шкалы для перевода из 32-битного дополнительного кода отсчета в электрическую величину (В) для считываемой выборки. Перевод осуществляется по формуле: $E = SAMPLE * K_E + B_E$, где $SAMPLE$ — исходный код отсчета выборки, а E — полученное значение электрической величины.
61958-61959	WF_CHINFO_B_E	float	R	Коэффициент смещения для перевода из 32-битного дополнительного кода отсчета в электрическую величину (В) для считываемой выборки по формуле из описания регистра WF_CHINFO_K_E.
61960-61961	WF_CHINFO_K_P	float	R	Коэффициент шкалы для перевода из электрической величины в физическую (м/с^2 для канала виброускорения, мкм для канала радиальной вибрации). Перевод осуществляется по формуле: $P = E*K_P+B_P$, где E — значение электрической величины, полученной из отсчета, по формуле, приведенной в описании регистра WF_CHINFO_K_E, а P — результирующая физическая величина.
61962-61963	WF_CHINFO_B_P	float	R	Коэффициент смещения для перевода из электрической величины в физическую по формуле из описания регистра WF_CHINFO_K_P.
61964-61975	-	-	R	Резерв
61976	WF_CHINFO_CHST	CHST	R	Состояние канала, которому соответствует выборка, на момент сохранения выборки.

2.1.1.2.4 Блок регистров для передачи команд

Адрес	Название	Тип	Аттр	Описание
Параметры передаваемой команды				
65024-65025	CMD_REQ_SIGN	U32	W	Сигнатура действительной команды. В этот регистр всегда должно записываться значение 0x5649434D при записи новой команды
65026	CMD_REQ_CODE	U16	W	Код запрашиваемой команды. Определяет выполняемые действия, формат данных и результатов команды. Значения приведены при описании конкретных команд соответствующей главе.
65027	CMD_REQ_ID	U16	W	Пользовательский идентификатор. Записанное значение доступно в результатах в регистре CMD_RESP_ID. Может служить для контроля соответствия ответа и запроса
65028-65031	-	-	W	Резерв. Должны записываться нулевые значения.
65032-65141	CMD_REQ_DATA	U8[220]	W	Данные команды. Формат определяется кодом команды.
Результат выполнения команды				
65152	CMD_RESP_STATUS	U16	R	Статус выполнения команды: • 0 — Нет выполненной команды. Дальнейшие данные не действительны. Означает, что с момента последнего чтения регистра не было записано параметров новой команды. • 2 — Команда успешно выполнена. • 3 — Команда выполнена с ошибкой. Код ошибки доступен в регистре CMD_RESP_ERRCODE.
65153	CMD_RESP_ERRCODE	U16	R	В случае, если команда завершилась с ошибкой, содержит код ошибки.
65154	CMD_RESP_REQCODE	U16	R	Код команды. Значение, записанное в CMD_REQ_CODE. Определяет формат данных в ответе.
65155	CMD_RESP_ID	U16	R	Пользовательский идентификатор команды. Содержит значение, записанное в CMD_REQ_ID.

65156	CMD_RESP_SIZE	U16	R	Размер данных результата выполнения команды в регистрах CMD_RESP_DATA в байтах. Если значение не кратно 2, то в последнем регистре старший байт не действителен.
65157-65159	-	-	W	Резерв.
65160	CMD_RESP_DATA	U8[220]	W	Данные результата выполнения команды. Формат определяется кодом команды. Действительный размер в байтах определяется значением CMD_RESP_SIZE.

2.1.1.3 Составные типы, используемые при описании адресного пространства

2.1.1.3.1 Значение времени TSVAL

Данный набор регистров описывает значение метки времени. Многие поля имеют прямое соответствие с полями метки времени протокола PTP. Метка может содержать либо локальное время в системе, либо абсолютное время, в зависимости от значения флага PTP_TIMESCALE В первом случае метка содержит время с момента включения модуля, относительно которого синхронизировано время. Во втором случае, метка содержит прошедшее время с 1 января 1970 года в соответствии с международном атомным временем (TAI), которое опережает время UTC на фиксированное число секунд, которое доступно в поле UTC_OFFS.

Адрес	Название	Тип	Описание
0	FLAGS	U16	Набор флагов с информацией о метке времени: • UTC_OFFS_VALID (бит 2) — признак, что поле UTC_OFFS содержит действительное значение. • PTP_TIMESCALE (бит 3) — признак, что метка содержит абсолютное время, а не локальное для системы • TIME_TRACEABLE (бит 4) — определяет, привязано ли указанное время к международному • VALID (бит 15) — признак, что метка времени имеет действительное значение
1	SEC_H	U16	Старшие 16 бит 48-битного значения количества секунд
2-3	SEC_L	U32	Младшие 32 бита 48-битного значения количества секунд
4-5	NSEC	U32	Количество наносекунд (от 0 до 999999999)

6	NSEC_FRACT	U16	Количество дробных долей наносекунд. Единица соответствует $1/2^{16}$ нс
7	UTC_OFFS	U16	Количество секунд, на которое указанное время опережает время UTC. Действительно, если установлен флаг UTC_OFFS_VALID Используется, если метка представляет собой время TAI, которое опережает UTC за счет наличия високосных секунд в UTC. На текущий момент TAI опережает UTC на 37 секунд.

2.1.1.3.2 Информация о метке времени TSINFO

Данный набор регистров содержит информацию о значении метки времени и ее источнике.

Адрес	Название	Тип	Описание
0-7	TS_VAL	TSVAL	Значение времени
8-11	CLK_ID	U8[8]	Идентификатор источника времени, в пересчете на время которого была получена данная метка.
12-15	-	-	резерв

2.1.1.3.3 Информация о внутреннем локальном времени TSLOC

Данный набор регистров описывает несинхронизированное время модуля в виде интервала времени с момента запуска модуля и его соотношение с синхронизированным временем.

Адрес	Название	Тип	Описание
0-1	DFREQ	float	Отклонение локальной частоты модуля от частоты, с которой синхронизированно время в ppm
2-3	SEC	U32	Количество секунд
4-5	NSEC	U32	Количество наносекунд
6-7	-	-	резерв

2.1.1.3.4 Информация о измерении диагностического АЦП DGADC

Адрес	Название	Тип	Описание
0	STATUS	U16	Состояние измерения: • PRESENT (бит 0) — Признак наличия данного диагностического АЦП в модуле. • VALID (бит 1) — Признак наличия действительного измерения • RENGE_ERR (бит 2) — Признак наличия ошибки выхода за допустимый диапазон значения диагностического АЦП
1	-	-	Резерв
2-3	VALUE	float	Значение измерения диагностического параметра. Единицы измерения зависят от номера диагностического АЦП.

2.1.1.3.5 Регистры состояния измерительного канала CHST

Адрес	Название	Тип	Описание
0	CH_TYPE	U16	Тип канала: • 0 — Канал не используется в конфигурации • 1 — Канал виброускорения • 2 — Канал радиальной вибрации • 3 — Канал фазоотметчика
1	CHST_STATUS	U16	Статус канала. Определяет корректность измерений для данного канала. Возможные значения: • 0xFFFF — Измерения еще не готовы (начальное значение при старте или после обновления настроек, пока не пройдет начальный переходный интервал) • 0 — данные действительны, канал в исправном состоянии • 1 — обнаружено короткое замыкание линий подключения датчика • 2 — обнаружен обрыв линий подключения датчика • 3 — данные вне диапазона действительных значений датчика

2	CHST_FLAGS	U16	Значение специальных признаков для данного канала: • BYPASS (бит 0) — состояние установленного признака “Bypass” • INH_CMD (бит 2) — состояние установленного вручную признака подавления тревог • INH_EVT (бит 3) — состояние установленного по событиям признака подавления тревог • TM_CMD (бит 4) — состояние установленного вручную признака умножения уставок • TM_EVT (бит 5) — состояние установленного по событиям признака умножения уставок
3	CHST_MEAS_CNT	U16	Количество настроенных измерений для данного канала
4-19	-	-	Резерв.
$20 + 10 * M$	MEAS_TYPE	U16	Тип измерения с номером M: • 0 — измерение не используется • 1 — виброускорение • 2 — виброскорость • 3 — виброперемещение • 4 — зазор • 5 — скорость вращения • 6 — амплитуда гармоники виброускорения • 7 — фаза гармоники виброускорения • 8 — амплитуда гармоники виброскорости • 9 — фаза гармоники виброскорости • 10 — амплитуда гармоники виброперемещения • 11 — фаза гармоники виброперемещения • 12 — постоянная составляющая • 16 — Максимальное смещение от среднего (Smax) • 17 — виброускорение (ВЧ)

21 + 10 * M	MEAS_ST_FLAGS	U16	Флаги состояния измерения. Значения битов поля: • FAILURE (бит 0-1) — признак неисправности (tribool) • ALERT (бит 2-3) — признак выхода за уставку "предупреждение" (tribool) • DANGER (бит 4-5) — признак выхода за уставку "останов" (tribool) • BASE_VALID (бит 8) — признак действительности базового значения • BASE_EXC (бит 9) — признак, что отклонение значения от базового превышает заданный порог на отклонение значения • MIN_LIMIT (бит 10) — признак, что значение измерения меньше нижней границы настроенного диапазона и возвращено ограниченное нижней границей значение • MAX_LIMIT (бит 11) — признак, что значение измерения выше верхней границы настроенного диапазона и возвращено ограниченное верхней границей значение
22-23 + 10 * M	MEAS_VALUE	float	Значение измерения в физических величинах.
24 + 10 * M	MEAS_FS_PERC	U16S	Значение измерения в процентах от заданной пользовательской шкалы.
25 + 10 * M	BASELINE_PERC	U16S	Базовое значение измерения (с которым сравнивается текущее для определения превышения порога отклонения). Задается в процентах от пользовательской шкалы. Действительно только если установлен соответствующий флаг в MEAS_ST_FLAGS.
26-29 + 10 * M	MEAS_RESVD	U16[4]	Резервные поля (читаются 0).

2.1.2 Чтение выборок через регистры Modbus

Выборка сохраняется одновременно по выбранной группе каналов. Помимо сохраненных данных по каждому выбранному каналу, выборка содержит как общую для всех каналов информацию, так и информацию, относящуюся отдельно к каждому каналу.

Каждой выборке соответствует свой идентификатор (ID) выборки — 16-битный счетчик, соответствующий номеру сохраняемой выборки с начала работы модуля. Номер каждой последующей выборки увеличивается на 1 (переходя от 0xFFFF в 0).

С помощью регистра ST_WF_RDY_CNT и набора регистров ST_WF_LAST_ID[] можно

определить, выборки с какими идентификаторами сохранены в оперативной памяти модуля и доступны для чтения. Последняя сохраненная выборка имеет идентификатор, равный значению ST_WF_LAST_ID[0], предыдущая — ST_WF_LAST_ID[1] и т.д.. Количество действительных индентификторов в ST_WF_LAST_ID[] соответствует минимальному значению из количества готовых выборок (ST_WF_RDY_CNT) и количества доступных регистров ST_WF_LAST_ID[].

Следует отметить, что хотя как правило идентификатор в каждом следующем элементе массива ST_WF_LAST_ID[] меньше на 1 значения предыдущего, но в общем случае это условие может не выполняться, т.к. идентификаторы выборки выделяются при поступнлении запросов на их сохранение, а порядок поступления запросов не всегда соответствует порядку выполнения сохранения выборок. Например, если запрос указывает, что выборка должна быть сохренена в определенное время в будущем и между наступлением этого момента придет новый запрос на сохранение выборки немедленно, то ему будет присвоен следующий идентификатор, но сохранение новой выборки завершиться раньше.

Для чтения общей информации о выборке необходимо записать идентификатор выборки в регистр WF_SEL_ID, после чего прочитать информацию из блока регистров с информацией о выборке. Регистр WF_INFO_CH_MASK определяет, отсчеты каких каналов сохранены в выборке. После этого возможно последовательно выбрать каждый доступный канал, путем записи его номера в регистр WF_SEL_CH, и прочитать по нему информацию из блока регистров с информацией о выбранном канале выборки, а также собственно сами отсчеты.

Для чтения отсчетов помимо идентификатора выборки и номера канала, необходимо также записать в регистр WF_SEL_OFFS номер первого читаемого отсчета. Это необходимо, т.к. количество отсчетов в выборке превышает размер области, выделенной для чтения отсчетов, и отсчеты считываются блоками, изменяя каждый раз номер первого отсчета в блоке. Значения отсчетов считываются из регистров WF_RD_DATA.

Для чтения информации и данных выборки рекомендуется использовать функцию Modbus “Read/Write Multiple Registers (23)”, для того, чтобы за одну операцию выполнять запись в регистры WF_SEL_ID/WF_SEL_CH/WF_SEL_OFFS и чтение информации/отсчетов выборки. Это позволяет не только сократить количество операций (т. к. запись и чтение выполняется за одно обращение), но и гарантирует, что при попытке одновременного чтения выборок разными клиентами, другой клиент не нарушит процесс чтения выборки, изменив значение WF_SEL_ID/WF_SEL_CH/WF_SEL_OFFS в момент между записью этих регистров и чтением собственно данных/информации о выборках. При этом рекомендуется всегда записывать все значимые значения регистров выбора (WF_SEL_ID для информации о выборке, WF_SEL_ID и WF_SEL_CH для информации о канале, WF_SEL_ID, WF_SEL_CH и WF_SEL_OFFS для чтения отсчетов) в каждой операции чтения-записи, чтобы гарантировать корректную совместную работу нескольких клиентов. Если используются раздельные операции чтения и записи, то при чтении следует также считывать значения WF_SEL_CH/WF_SEL_ID/WF_SEL_OFFS (или WF_INFO_ID при чтении информации о выборке и WF_CHINFO_ID/WF_CHINFO_CH при чтении информации о канале) и сверять, что они соответствуют записанным значениям, повторяя попытку, если другой клиент изменил значения.

Следует отметить, что размер выборки и смещение при чтении отсчетов задаются в отсчетах (а не в регистрах), каждый из которых занимает два регистра.

За одну операцию по Modbus можно прочитать 125 регистров, или 62 отсчета (60, если также считывать обратно значения регистров WF_RD_ID/WF_SEL_CH/WF_SEL_OFFS. Соответственно, всю выборку необходимо вычитывать по 62 (60) отсчета, изменяя зна-

чения WF_SEL_OFFS. При попытке чтения за пределами выборки (т. е. если WF_SEL_OFFS + S >= WF_INFO_SIZE), чтение завершится с ошибкой. Также чтение может быть завершено с ошибкой, если выборка будет удалена во время чтения (если читается самая старая выборка и во время чтения происходит сохранение новой выборки, для которой недостаточно свободного места).

2.1.3 Выполнение команд через регистры Modbus

Модули поддерживают выполнение пользовательских команд через регистры адресного пространства Modbus. Для выполнения команды необходимо все ее параметры записать в блок регистров параметров команды. Запись всех параметров должна осуществляться за один запрос по интерфейсу modbus и в качестве адреса всегда использоваться начальный адрес блока. Формат и размер поля данных (CMD_REQ_DATA) зависит от кода команды (CMD_REQ_CODE). Размер реально записанных данных команды определяется количеством записываемых регистров за операцию записи. Как правило с командой необходимо записать все указанные параметры, иначе команда завершиться с ошибкой и не будет выполнена. Для некоторых команд может быть обязательные только часть параметров, а остальные могут принимать значения по умолчанию, если размер записи меньше, чем позиция данного параметра. В этом случае в описании данных команды явно указано, какой первый параметр является опциональным.

Получение результата выполнения команды осуществляется с помощью чтения блока регистров результатов выполнения команды. Данный блок также должен быть прочитан за один раз с первого адреса после записи параметров команды. После выполнения чтения результат выполнения последней команды стирается и повторное чтение завершится с признаком отсутствия результатов. Размер чтения может быть выбран исходя из максимального размера данных в результате для записанного кода команды, либо всегда может быть прочитан весь блок с результатами выполнения команды (его размер выбран так, чтобы его можно было считать за одно чтение), а уже далее размер полезных данных в ответе может быть определен по значению регистра CMD_RESP_SIZE.

Для выполнения команд рекомендуется использовать функцию Modbus “Read/Write Multiple Registers (23)”, которая позволяет за одну операцию записать параметры команды и получить результаты ее выполнения. Это позволяет в случае нескольких клиентов, работающих с одним модулем, гарантировать, что ответ соответствует записанной команде, и делает невозможной ситуацию, что между записью параметров команды и чтением результатов будут записаны параметры новой команды другим клиентом. При использовании отдельных функций записи и чтения рекомендуется изменять для каждой команды параметр пользовательского идентификатора CMD_REQ_ID и сверять полученный идентификатор в результатах (CMD_RESP_ID) для подтверждения соответствия ответа команде.

2.1.3.1 Команда установки значения признака измерительного канала

Команда позволяет изменить состояние одного из специальных признаков для выбранного набора измерительных каналов.

Код команды: 0x0100

Данные команды:

Смещение	Название	Тип	Описание
0	FLAG_TYPE	U16	Признак канала, значение которого должно быть изменено: • 1 — Bypass • 2 — Подавление тревог • 3 — Умножение уставок
1	FLAG_STATE	U16	Значение признака (0 — не активен, 1 — активен), которое должно быть установлено
2-3	CH_MASK	U32	Маска, определяющая, для каких каналов должно быть изменено значение признака.

Данные результата: нет.

2.1.3.2 Запрос сохранения выборки

Код команды: 0x0200
Данные команды (опциональные параметры с WF_REQ_DELAY):

Смещение	Название	Тип	Описание
0	WF_REQ_FLAGS	U16	Флаги запроса. Резерв. Должен быть записан 0
1	WF_REQ_TYPE	U16	Тип запрашиваемой выборки. Возможные значение аналогичны значениям регистра WF_INFO_TYPE
2-3	-	-	Резерв. Должны быть записаны нулевые значения
4-5	WF_REQ_CH_MASK	U32	Маска, определяющая, по каким каналам запрашивается сохранение выборки. Должны быть указаны только каналы, разрешенные в конфигурации модуля.
6-7	WF_REQ_SIZE	U32	Количество отсчетов, которое должно быть сохранено для каждого канала. Число не должно превышать 720896. Если заданное число не кратно 4, то значение отсчетов в результирующей выборке будет увеличено до ближайшего кратного 4 значения.

8-9	WF_REQ_DELAY	U32	По умолчанию по приходу запроса или при наступлении указанного времени в качестве выборки сохраняются последние WF_REQ_SIZE отсчетов, т.е. момент запроса соответствует последнему отсчету выборки. При указании данного поля сохранение выборки будет задержано на указанное число отсчетов, т.е. время запроса будет отсчету с номером WF_REQ_SIZE - WF_REQ_DELAY - 1, если считать от нуля.
10-15	-	-	Резерв. Если эти поля записываются, то необходимо записывать 0.
16-23	WF_REQ_TIME	TSVAL	Время, когда необходимо сохранить выборку. Если запись в данное поле не выполняется или в него записывается недействительное время, то выборка сохраняется немедленно, если модулем уже было получено указанное число отсчетов АЦП с момента записи последней конфигурации. Если же указано действительное время и оно превышает текущее время модуля, то запрос будет отложен и выборка будет сохранена в указанное время (с задержкой на WF_REQ_DELAY отсчетов)

Данные результата: нет.

2.1.3.3 Сброс выбранных дискретных выходов

Команда предназначена для сброса активного состояния (тревоги) дискретных выходов, для которых был настроен режим работы с фиксацией, после деактивации управляющего события. При этом сама команда действует на все указанные дискретные выходы, если они находятся в состоянии тревоги и разрешены в конфигурации. В случае если событие управления реле (тревоги) активно во время команды сброса, то состояние выхода будет немедленно переведено в неактивное состояние, но если условие тревоги будет сохраняться в течение настроенной задержки включения, то выход снова перейдет в активное состояние по истечении данного интервала.

Код команды: 0x0300

Данные команды:

Смещение	Название	Тип	Описание
0-3	DOUT_LP_MASK	U64	Маска, указывающая, какие слаботоковые реле должны быть сброшены. Каждому реле соответствует один бит (младщий бит первому выходу). Если бит установлен, то состояние выхода будет сброшено, в противном случае команда не оказывает влияния на этот выход.
4-7	DOUT_HP_MASK	U64	Маска, указывающая, какие сильнотоковые реле должны быть сброшены. Каждому реле соответствует один бит (младщий бит первому выходу). Если бит установлен, то состояние выхода будет сброшено, в противном случае команда не оказывает влияния на этот выход.

Данные результата: нет.

2.1.3.4 Ручное изменение состояния выбранных дискретных выходов

Команда предназначена для изменения состояния дискретных выходов, для которых было настроено ручное управление. Команда никак не влияет на дисретные выходы, для которых было настроено управление состоянием по событию, даже если они указаны в параметрах команды.

Код команды: 0x0301

Данные команды:

Смещение	Название	Тип	Описание
0-3	DOUT_LP_MASK	U64	Маска, указывающая, состояние каких слаботоковых реле должно быть изменено. Каждому реле соответствует один бит (младщий бит первому выходу). Если бит установлен, то состояние выхода будет изменено, в противном случае команда не оказывает влияния на этот выход.
4-7	DOUT_LP_VALS	U64	Устанавливаемые состояния слаботоковых реле. Каждому реле соответствует один бит (младщий бит первому выходу). Бит имеет значение, только если соответствующий бит в DOUT_LP_MASK установлен. Если бит равен 1, то реле переходит в активное состояние, если 0 — в неактивное

8-11	DOUT_HP_MASK	U64	Маска, указывающая, состояние каких сильнотоковых реле должно быть изменено. Каждому реле соответствует один бит (младщий бит первому выходу). Если бит установлен, то состояние выхода будет изменено, в противном случае команда не оказывает влияния на этот выход.
12-15	DOUT_HP_VALS	U64	Устанавливаемые состояния сильнотоковых реле. Каждому реле соответствует один бит (младщий бит первому выходу). Бит имеет значение, только если соответствующий бит в DOUT_HP_MASK установлен. Если бит равен 1, то реле переходит в активное состояние, если 0 — в неактивное

Данные результата: нет.

2.1.3.5 Чтение информации о конфигурации режима работы модуля

Команда позволяет прочитать информацию о текущей конфигурации режима работоы модуля. Данная информация необходима для дальнейшего чтения собственно данных конфигурации.

Код команды: 0x1000

Данные команды: нет.

Данные результата:

Нет, если в модуле отсутствует текущая конфигурация режима работы. Если конфигурация была записана, то возвращается информация о ней, следующего формата:

Смещение	Название	Тип	Описание
0-1	OPCFG_SIZE	U32	Полный размер записанной конфигурации в байтах. Размер может быть не кратен 2, в этом случае при чтении конфигурации для регистра, соответствующего последнему байту, старший байт содержит недействительные данные.
2-3	OPCFG_CRC32	U32	Контрольная сумма всех данных конфигурации.
4	OPCFG_ID	U16	Идентификатор конфигурации.
5	MODULE_TYPE	U16	Идентификатор типа модуля, для которого предназначена конфигурация. Для корректной конфигурации должен совпадать со значением регистра MODULE_TYPE.

6	MODIFICATION	U16	Идентификатор модификации модуля, для которого предназначена конфигурация. Для корректной конфигурации должен совпадать со значением регистра MODIFICATION.

2.1.3.6 Чтение блока данных конфигурации режима работы модуля

Команда позволяет прочитать блок из массива данных, содержащих текущую конфигурацию режима работы модуля, заданного размера и с заданным смещением от начала конфигурации. За один раз возможно считать блок до 220 байт.

Если в модуле отсутствует текущая рабочая конфигурация, то выполнение команды завершится с ошибкой. Если будет запрошен блок, выходящий за размер массива данных, представляющих текущую конфигурацию, то в ответе будет возвращено меньше данных, чем запрошено (0, если смещение превышает размер данных конфигурации).

Для чтения конфигурации необходимо сперва выполнить чтение информации о текущей конфигурации, чтобы проверить наличие текущей конфигурации и ее размер, после чего выполнить чтение всех данных конфигурации блоками по 220 байт от начала до конца данных конфигурации, увеличивая смещение.

Формат данных конфигурации описан в разделе 2.3.

Код команды: 0x1001

Данные команды:

Смещение	Название	Тип	Описание
0-1	BLOCK_OFFSET	U32	Смещение в байтах от начала конфигурации блока данных, который требуется прочитать.
2-3	BLOCK_SIZE	U32	Размер блока, который требуется прочитать. Максимальное допустимое значение — 220 байт.

Данные результата: При успешном выполнении команды, возвращенные данные соответствуют прочитанному блоку текущей конфигруации режима работы модуля. Возвращенные данные всегда относятся к действительной части конфигурации и их размер может быть меньше запрашиваемого, если запрашиваемых блок выходит за границы действительных данных конфигурации.

2.1.3.7 Начало передачи новой конфигурации режима работы в модуль

Данная команда должна выполняться перед началом записи новой конфигурации режима работы модуля. Команда выполняет необходимую подготовку к приему новой конфигурации, также очищая результаты предыдущей передачи (если уже была до этого начата передача конфигурации). Данная команда возвращает идентификатор новой записываемой конфигурации, который должен использоваться в последующих командах записи.

Код команды: 0x1010

Данные команды: нет.

Данные результата:

Смещение	Название	Тип	Описание
0	OPCFG_ID	U16	Идентификатор новой записываемой конфигурации.

2.1.3.8 Передача блока новой конфигурации режима работы в модуль

Команда выполняет запись блока данных, который содержит часть данных новой конфигурации модуля, начиная с указанного смещения от начала конфигурации.

Код команды: 0x1011

Данные команды:

Смещение	Название	Тип	Описание
0	OPCFG_ID	U16	Идентификатор новой записываемой конфигурации, полученный в результатах выполнения команды начала передачи конфигурации.
1	Reserved	U16	Резерв.
2-3	BLOCK_OFFSET	U32	Смещение в байтах от начала конфигурации записываемого блока данных.
4-(4 + (N-1)/2)	BLOCK_DATA	U8[N]	Данные записываемого блока. Размер блока в байтах не должен превышать 212 байт.

Данные результата: Нет

2.1.3.9 Завершение передачи новой конфигурации режима работы в модуль

Команда должна выполняться после того, как были переданы все блоки данных, соответствующие новой передаваемой конфигурации. По данной команде выполняется проверка переданных ранее данных конфигурации с помощью контрольной суммы. После данной команды модуль не будет принимать новые блоки данных конфигурации до выполнения команды начала записи новой конфигурации. При этом сама команда не влияет на текущую конфигурацию модуля или на записанную в энергонезависимую память конфигурацию, но должна предшествовать соответствующим командам.

Код команды: 0x1012

Данные команды:

Смещение	Название	Тип	Описание
0	OPCFG_ID	U16	Идентификатор новой записываемой конфигурации, полученный в результатах выполнения команды начала передачи конфигурации.
1	Reserved	U16	Резерв.
2-3	BLOCK_SIZE	U32	Общий размер данных записываемой конфигурации.

4-5	CRC	U32	Контрольная сумма CRC32 по данным конфигурации для проверки целостности переданных данных новой конфигурации.

Данные результата: Нет

2.1.3.10 Обновление текущей конфигурации режима работы

По данной команде выполняется обновление текущей рабочей конфигурации модуля. Для выполнения данной операции должна быть предварительно завершена передача в модуль конфигурации, на которую требуется обновить текущую. При выполнении данной операции должен быть указан корректный идентификатор последней переданной конфигурации, иначе выполнение команды завершится с ошибкой. При успешном выполнении данной команды работа модуля перезапускается в соответствии с новой конфигурацией, обновляются данные текущей конфигурации модуля, а также обновляется идентификатор текущей конфигурации модуля (регистр ST_OPCFG_ID).

Код команды: 0x1020

Данные команды:

Смещение	Название	Тип	Описание
0	OPCFG_ID	U16	Идентификатор ранее переданной конфигурации, на которую необходимо обновить текущую.

Данные результата: Нет

2.1.3.11 Сохранение конфигурации режима работы в энергонезависимую память

По данной команде выполняется запись ранее переданной конфигурации в энергонезависимую память. При выполнении данной операции должен быть указан корректный идентификатор последней переданной конфигурации, иначе выполнение команды завершится с ошибкой. В случае успешного завершения операции, переданная конфигурация будет сохранена в энергонезависимую память и будет загружена при следующем включении питания или перезагрузке модуля.

Код команды: 0x1020

Данные команды:

Смещение	Название	Тип	Описание
0	OPCFG_ID	U16	Идентификатор ранее переданной конфигурации, которую необходимо сохранить в энергонезависимую память.

Данные результата: Нет

2.2 Протокол OPC UA

Модули системы L-ViMS поддерживают также работу по протоколу OPC UA TCP (Binary data encoding), выступая в качестве сервера. Для подключения используется стандартный для данного протокола TCP-порт — 4840.

В данном протоколе доступны следующие функции:

- Получение непрерывного потока отсчетов каналов АЦП (для версии прошивки 1.0.0 и выше). Это позволяет выполнять прием и обработку всего потока сыры данных от АЦП на ПК, аналогично АЦП общего назначения. Данная возможность доступна только по протоколу OPC UA в связи с ограничением скорости протокола Modbus TCP.

- Получение значений измерений, состояния дискретных входов и выходов, а также управление состоянием дискретных выходов (для версии прошивки 1.0.3 и выше). Данная возможность позволяет получать данные и управлять выходами с помощью стандартного клиента OPC UA общего назначения, в том числе из различных SCADA-систем.

Остальные возможности, в том числе выполнение конфигурации модуля, доступны на данный момент только по протоколу Modbus TCP.

Оба протокола могут работать одновременно, что например, позволяет работать с модулем по протоколу Modbus TCP, открывая дополнительное соединение по OPC UA только в случае, когда необходимо запустить непрерывный сбор данных или выполнять настройку по Modbus TCP с одновременным получением данных в SCADA системе по OPC UA.

Для получения данных из модуля через OPC UA требуется предварительно записать конфигурацию режима работы в модуль, в которой разрешены требуемые каналы ввода-вывода и настроены требуемые измерения. Только для разрешенных каналов и измерений будут доступны соответствующие элементы в адресном пространстве OPC UA.

Протокол OPC UA позволяет выполнять одновременное подключение нескольким клиентом, однако при использовании данного протокола для получения потока сырых данных АЦП, в связи с тем, что операция чтения потока данных требует значительных ресурсов модуля, рекомендуется не использовать больше одной подписки на сырые данные на канал, иначе если модуль не будет успевать передавать данные, это может привести к ошибкам переполнения буфера, приводящим к разрывам в потоке данных.

2.2.1 Адресное пространство OPC UA

В данном разделе описываются все элементы (Node) адресного пространства OPC UA, специфичные для модулей системы L-ViMS.

Элементы могут быть доступны по явно указанному идентификатору элемента (Node ID), так и найдены в адресном пространстве по иерархическому имени элемента (Browse Name).

Все идентификаторы элементов адресного пространства (Node ID) имеют целочисленный тип и располагаются в адресном пространстве с индексом 1 (индекс 0 используется для стандартных элементов, определенных спецификациями OPC UA).

Для идентификаторов узла, заданных формулой, используются следующие обозначения:

- I — номер измерительного канала модуля (0-3)
- M — номер измерения (0-9)
- D — номер дискретного входа или выхода определенного типа

При указании имен элементов в адресном пространстве используется символ “/” для указания ссылки типа `Organizes` (аналог директории в файловой системе), а символ “.” для указания ссылки типа `HasProperty` или `HasComponent`, которые используются для задания свойств или компонентов исходного элемента соответственно. Все элементы располагаются в поддиректориях директории `/Objects/Vi/` (где первый символ “/” означает корневую директорию, стандартный элемент “Root Folder” адресного пространства OPC UA). Если имя зависит от номера канала, измерения и т.д., то место подстановки указывается в треугольных скобках, а обозначения аналогичны обозначениям в задании идентификатора узла, но в именах они считаются от 1, поэтому в строке подстановки добавляется $+1$, например имя элемента `AI_CH<`I`+1>_M<`M`+1>` для второго измерения ($M == 1$) первого канала ($I == 0$) будет `AI_CH1_M2`. Далее в таблице для каждой группы элементов указывается общая родительская директория, в которой располагаются все элементы данной группы, а в описании элемента уже его имя внутри данной директории.

Столбец «Аттр» указывает, какие операции возможны с этим регистром (R — чтение, W — запись, M — мониторинг состояния значения).

Node id	Название	Тип	Аттр	Описание
Значения измерений (/Objects/Vi/Data/AI/Meas)				
4352 + 1024 * I + 32 *M	ACQ_CH_MEAS_VAL	double	RM	AI_CH<I+1>_M<M+1> Значение соответствующего измерения в физических единицах. Элемент присутствует только в случае, если измерение с соответствующим номером добавлено в записанной конфигурации режима работы модуля. При отслеживании значения измерения допустимо установить Sampling Interval равным нулю, в этом случае значение элемента будет обновляться при каждом завершении пересчета или изменении состояния измерения (отправление значения клиенту осуществляется в соответствии с установленным интервалом публикации). Если в конфигурации режима работы модуля для измерения задано пользовательское имя, то отображаемое имя данного элемента будет соответствовать пользовательскому имени измерения.
4353 + 1024 * I + 32 *M	ACQ_CH_MEAS_RANGE	Range	R	AI_CH<I+1>_M<M+1>.EURange Диапазон возможных значений данного измерения. Соответствует заданным границам в параметрах измерения в конфигурации режима работы модуля. Данный тип представляет собой структуру из двух полей типа double — минимального и максимального значений.

4354 + 1024 * I + 32 *M	ACQ_CH_MEAS_UNITS	EUInfo	R	AI_CH<I+1>_M<M+1>. EngineeringUnits Описание единиц соответствующего измерения. Зависит от типа измерения в конфигурации режима работы модуля. Тип данных соответствует структуре **EUInformation**, значения полей установлены в соответствии с процедурой отображения кодов величин UN/CEFACT, описанной в спецификации OPC 10000-008.
Состояние дискретных входов (/Objects/Vi/Data/DI)				
49152 + 16 * D	DIN_CH_VALUE	boolean	RM	DI<D+1> Текущее состояние соответствующего дискретного входа. Элемент присутствует только в случае, если дискретный вход разрешен в конфигурации режима работы модуля. При отслеживании значения измерения допустимо установить Sampling Interval равным нулю, в этом случае значение элемента будет обновляться при каждом цикле опроса состояния входа модулем (отправление значения клиенту осуществляется в соответствии с установленным интервалом публикации).. Если в конфигурации режима работы модуля для дискретного входа задано пользовательское имя, то отображаемое имя данного элемента будет соответствовать пользовательскому имени входа.

Состояние дискретных выходов (/Objects/Vi/Data/DO)				
51200 + 16 * D	DOUT_LP_CH_VALUE	boolean	RWM	DO_LP<D+1> Текущее логическое состояние соответствующего слаботокового реле. Элемент присутствует только в случае, если выход разрешен в конфигурации режима работы модуля. Данный элемент всегда доступен для чтения и отслеживания значения выхода. Возможность записи значения зависит от настроек выхода в конфигурации режима работы модуля: • Для режима управления состоянием выхода по команде допустима запись как истинного, так и ложного значения для перевода выхода в соответствующее логическое состояние. • Для режима управления состоянием выхода по событию с фиксацией допустима запись только значения “ложь” для сброса состояния активной тревоги выхода. • Для режима управления состоянием выхода по событию без фиксации операция записи недоступна. Если в конфигурации режима работы модуля для дискретного выхода задано пользовательское имя, то отображаемое имя данного элемента будет соответствовать пользовательскому имени выхода.
52224 + 16 * D	DOUT_HP_CH_VALUE	boolean	RWM	DO_HP<D+1> Текущее логическое состояние соответствующего сильнотокового реле. Логика работы аналогична элементу DOUT_LP_CH_VALUE.

Чтение сырых потоквых данных (/Objects/Vi/Data/AI/Raw)				
5104 + 1024 * I	ACQ_CH_RAW_DATA	I32[]	M	AI_CH<I+1>_RAW Данный элемент позволяет получать сырые отсчеты соответствующего канала АЦП. Элемент присутствует только в случае, если соответствующий вход АЦП разрешен в конфигурации режима работы модуля. Формат отсчетов соответствует формату данных регистра WF_RD_DATA. Поток данных получается путем подписки на обновление данных элемента (при явной операции чтения всегда возвращается пустой массив). Размер массива определяется в соответствии с номинальной частотой отсчетов и установленным интервалом опроса, но не превышает 12800 отсчетов. Если в конфигурации режима работы модуля для данного канала АЦП задано пользовательское имя, то отображаемое имя данного элемента будет соответствовать пользовательскому имени канала.
4992 + 1024 * I	ACQ_CH_RAW_EN	boolean	R	AI_CH<I+1>_RAW.Enabled Признак, доступен ли данный канал для чтения сырых данных АЦП. В отличие от других элементов данный элемент может быть прочитан явно по Node ID даже если канал не разрешен в рабочей конфигурации модуля чтобы проверить разрешения канала.
4994 + 1024 * I	ACQ_CH_RAW_FREQ	double	R	AI_CH<I+1>_RAW.FreqLocal Номинальная частота отсчетов данного канала в Гц (для существующих модулей одинакова для всех каналов).

4995 + 1024 * *I*	ACQ_CH_RAW_FREQ_GM	double	R	AI_CH<*I*+1>_RAW.FreqGM Частота отсчетов данного канала в Гц, пересчитанная на частоту источника синхронизации времени (для существующих модулей одинакова для всех каналов).
4996 + 1024 * *I*	ACQ_CH_RAW_K_E	double	R	AI_CH<*I*+1>_RAW.ConvElecK Коэффициент шкалы для перевода из 32-битного дополнительного кода отсчета в электрическую величину. Формула перевода соответствует приведенной при описании регистра WF_CHINFO_K_E
4997 + 1024 * *I*	ACQ_CH_RAW_B_E	double	R	AI_CH<*I*+1>_RAW.ConvElecB Коэффициент смещения для перевода из 32-битного дополнительного кода отсчета в электрическую величину. Формула перевода соответствует приведенной при описании регистра WF_CHINFO_K_E
4998 + 1024 * *I*	ACQ_CH_RAW_K_P	double	R	AI_CH<*I*+1>_RAW.ConvPhysK Коэффициент шкалы для перевода из электрической величины в физическую. Формула перевода соответствует приведенной при описании регистра WF_CHINFO_K_P
4999 + 1024 * *I*	ACQ_CH_RAW_B_P	double	R	AI_CH<*I*+1>_RAW.ConvPhysB Коэффициент смещения для перевода из электрической величины в физическую. Формула перевода соответствует приведенной при описании регистра WF_CHINFO_K_P

2.2.2 Особенности передачи меток времени

В протоколе OPC UA все данные сопровождаются метками времени, определяющими момент времени, когда эти данные были получены.

В связи с тем, что модули системы L-ViMS не имеют встроенных часов, то привязка к астрономическому времени возможно только в случае синхронизации времени модуля по протоколу PTP с сервером реального времени, который находится в той же сети.

В противном случае модуль использует свое внутреннее время или внутреннее время синхронизированное с другими модулями системы L-ViMS без привязки к астрономическому времени. В отличие от PTP в OPC UA нет возможности явной передачи локаль-

ного времени системы без связи с астрономическим. При использовании локального времени время старта модуля, относительно которого синхронизировано локальное, соответствует нулевому значению кода времени в OPC UA, которое соответствует началу 1 января 1601 года UTC. Таким образом по году можно определить, используется ли астрономическое время или внутреннее. Следует отметить, что локальное время не может быть напрямую представлено в виде времени формата Unixtime, т.к. в Unixtime отсчет идет от 1970 года.

2.2.3 Чтения потока сырых отсчетов

Модули системы L-ViMS позволяют выполнять чтение непрерывного потока отсчетов измерительных каналов, что позволяет использовать их как универсальные модули АЦП, с полным приемом и обработкой всех данных на ПК. Потоковое чтение может осуществляться параллельно с внутренней обработкой данных модулем в соответствии с записанной конфигурацией. При этом для получения потока сырых отсчетов АЦП необходимо предварительно записать в модуль рабочую конфигурацию, в которой будут разрешены каналы, для которых требуется прием сырых отсчетов, а также настроен как минимум способ подключения датчика (для модуля L-ViMS-ICP влияет на включение источника тока).

Пример реализации потокового чтения на языке *C* может быть скачан на сайте производителя.

Перед началом приема потоковых данных возможно получить информацию о том, для каких каналов модуля разрешен потоковый сбор данных (ACQ_CH_RAW_EN), частоте отсчетов (ACQ_CH_RAW_FREQ) и коэффициенты для перевода отсчетов в электрические (ACQ_CH_RAW_K_E и ACQ_CH_RAW_B_E) и физические величины (ACQ_CH_RAW_K_P и ACQ_CH_RAW_B_P), путем чтения значений соответствующих элементов адресного пространства.

Само же чтение потоковых данных выполняется с помощью подписки на обновление значения элемента ACQ_CH_RAW_DATA. Для этого необходимо создать подписку (запрос `CreateSubscription`), которая может быть общая на все каналы, после чего добавить в нее элементы ACQ_CH_RAW_DATA нужных каналов (запрос `CreateMonitoredItems`).

Для того, чтобы передача данных всех каналов началась с одной точки необходимо либо добавить элементы для всех каналов за один запрос `CreateMonitoredItems` сразу с разрешенным мониторингом состояния (`MonitoringMode` = REPORTING (2)), либо сперва добавить элементы с запрещенным мониторингом состояния (`MonitoringMode` = DISABLED (0)), а затем разрешить для всех элементов мониторинг состояния с помощью запроса `SetMonitoringMode`.

При создании подписки особое значение имеют следующие параметры запросов:

- `samplingInterval` из `MonitoringParameters` элемента запроса `CreateMonitoredItems` — определяет время в мс, за которое собираются отсчеты, для формирования каждого оповещения. Этот параметр определяет количество отсчетов в каждой порции передаваемых данных, как ACQ_CH_RAW_FREQ * `samplingInterval` / 1000. При этом размер блока не может быть больше 12800 отсчетов. Если по указанной формуле будет получено большее значение, то параметр `samplingInterval` будет уменьшен модулем, и в результатах выполнения запроса будет возвращено скорректированное значение.

- `requestedPublishing` запроса `CreateSubscription` — определяет интервал передачи сохраненных данных. Если равен `samplingInterval`, то как только будет

принят очередной блок данных по каждому каналу, он будет сразу передан. Если же `requestedPublishing` задан в N раз больше, чем `samplingInterval`, то модуль будет передавать данные по N блоков.

- `queueSize` из `MonitoringParameters` элемента запроса `CreateMonitoredItems` — определяет максимальное количество блоков данных (размером, определяемым `samplingInterval`), которое может быть сохранено в очереди на отправку в модуле до публикации. Значение параметра должно быть больше 1 (т.к. 1 или 0 в OPC UA соответствует работе без использования очереди с сохранением только последнего значения, что не подходит для получения непрерывных данных без пропусков), а также больше чем `publishingInterval`/`samplingInterval`, чтобы сохранить все блоки за одну публикацию. Рекомендуется использовать значение в интервале 20-30, чтобы с одной обеспечить надежную передачу без потерь в случае временного падения скорости и при этом не использовать слишком много ресурсов модуля.

- `timestampsToReturn` запроса `CreateMonitoredItems` — если требуются метки времени (которое может быть синхронизировано по PTP), соответствующие передаваемым блокам, то необходимо указать равным SOURCE (0), иначе можно указать NEITHER (3).

После разрешения отслеживания состояния элементов с отсчетами, модуль начнет передавать данные блоками в определенное количество отсчетов. Каждый блок отсчетов для каждого канала передается как одно оповещение об изменении значения элемента и содержит в себе:

- Значение элемента, которое представляет собой массив, каждый элемент которого —32-битное целое число со знаком, содержащее как код АЦП, так и дополнительные флаги (формат аналогичен формату данных регистра `WF_RD_DATA`). Преобразование кода в вещественные электрические или физические величины необходимо выполнять вручную, используя прочитанные коэффициенты.

- Статус элемента. Позволяет проверить наличие ошибок при подписке а также наличие переполнения внутренней очереди. Если перед принятым блоком был потерян один или несколько блоков данных и данные между принятым последним блоком и текущем не являются непрерывны, то в статусе будет установлен флаг `OVERFLOW`. Это может произойти из-за недостаточной скорости передачи в канале связи.

- Отметка времени. Присутствует, если была запрошена в периметре `timestampsToReturn` при создании подписки. Соответствует времени последнего отсчета блока. Представляет собой 64-битное значение в количестве 100 нс интервалов. Если используется внутреннее время системы, то соответствует времени, которое прошло с момента включения модуля, относительно которого синхронизировано время. Если абсолютное, то метка содержит время UTC, начиная с 1 января 1601 года. Следует обратить внимание, что начало отсчета времени в OPC UA отличается от того, что используется в Unixtime или PTP (1 января 1970), т.е. для перевода времени OPC UA в Unixtime нужно вычесть 11644473600 секунд.

Для завершения потокового сбора данных необходимо завершить отслежвать состояние соответствующих элементов (запрос `DeleteMonitoredItems`) и удалить подписку (запрос `DeleteSubscriptions`).

2.3 Формат данных конфигурации режима работы модуля

Конфигурация режима работы модуля передается в модуль и читается из модуля как массив байт, закодированных описанным в данном разделе образом.

Формат данного массива байт не привязан к протоколу и в отличие от разделов, посвященных описанию регистров modbus, в данном разделе смещение и размер полей будет указан в байтах, начиная с нулевого смещения.

Данные конфигурации представляют собой набор блоков, идущих друг за другом. Каждый блок начинается с заголовка, фиксированного формата:

Смещение	**Название**	**Тип**	**Описание**
0	`OPCFG_BLOCK_TYPE`	U16	Тип блока. Определяет способ интерпретации данных блока. Значения типа блоков и формат соответствующих данных описаны в последующих разделах.
2	`OPCFG_BLOCK_SIZE`	U16	Размер данных блока в байтах, следующих за заголовком.

Размер данных блока может быть произвольным значением от 0 до 65535. При этом заголовок нового блока всегда выравнен на 4 байта, т.е. его смещение от начала конфигурации всегда кратно 4. Если размер предыдущего блока не кратен 4, то перед новым блоком добавляется соответствующее количество (от 1 до 3) незначащих байт.

Таким образом при разборе конфигурации по заголовку блока всегда можно определить положение следующего блока, даже если тип текущего блока неизвестен. Алгоритм разбора конфигурации всегда должен корректно обрабатывать ситуацию, когда в конфигурации обнаружен блок неизвестного типа, т.к. в новых версиях ПО всегда могут появится новые возможности, которые потребуют ввести новые типы блоков конфигурации. Также алгоритм разбора должен корректно обрабатывать случай, когда размер данных блока больше, чем размер, определяемый форматом данных, описанным в данном документе, т.е. в последующих версиях ПО информация в блоке может расширяться путем добавления дополнительных полей в конец данных блока. Если для блока указан минимальный размер, то алгоритм также должен учитывать, что данных может быть меньше вплоть до указанного размера. Если минимальный размер не указан, то блок всегда должен содержать все описанные в данном документе поля. Если размер данных блока меньше минимально, то конфигурация считается недействительной.

Последним блоком должен всегда идти блок с типом 0xFFFF и нулевым размером, обозначающий конец конфигурации. Если в конфигурации отсутствует блок с типом 0xFFFF, то такая конфигурация считается недействительной.

Все остальные типы блоков являются опциональными и могут отсутствовать в конфигурации. В этом случае возможность, описанная данным блоком, не используется в конфигурации.

Некоторые типы блоков могут требовать определенного взаимного расположения в последовательности блоков, образующей конфигурацию. Подобные требования приведены в последующем описании типов блоков.

2.3.1 Блок общих настроек измерительного канала

Данный блок содержит общие параметры для одного указанного измерительного канала. Блок данного типа должен располагаться раньше в конфигурации, чем блоки другого типа, задающие параметры для данного канала или параметры измерений данного канала.

Данный раздел будет дополнен в последующих версиях документации...

Глава 3

Предоставляемое ПО

Данный раздел описывает штатное предоставляемое ПО для работы с модулями системы L-ViMS.

3.1 ViConfigurator

ViConfigurator представляет собой программу с графическим интерфейсом для создания и редактирования рабочей конфигурации как отдельных модулей системы L-ViMS, так всей системы в целом, а также для выполнения дополнительных сервисных операций, как изменение сетевых настроек, обновления встроенного программного обеспечения модулей и др.

Данный раздел будет дополнен в последующих версиях документации...

3.2 ViView

ViView представляет собой программу с графическим интерфейсом для мониторинга состояния и данных модулей системы L-ViMS, а также для ручного управления дискретными выходами, признакам умножения уставок и bypass для каналов, просмотра выборок и непрерывных данных.

Данный раздел будет дополнен в последующих версиях документации...