



VSM PANEL APP:
программное обеспечение
для разработки кода прикладных
программ автоматки VSM
руководство пользователя

СОДЕРЖАНИЕ

1 ОБЩИЕ СВЕДЕНИЯ	3
2 УСТАНОВКА	3
2.1 Подготовка рабочей среды	3
2.2 Установка дополнительных пакетов	3
2.3 Установка VSM PANEL APP	3
2.4 Запуск VSM PANEL APP	3
3 ПОДКЛЮЧЕНИЕ	4
4 ЗАВЕРШЕНИЕ РАБОТЫ	6
5 УДАЛЕНИЕ	6
6 ОПИСАНИЕ ИНТЕРФЕЙСА	7
7 ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ БЛОКОВ	22
7.1 Логические и математические функции	22
7.2 Блок ПИД-регулирования	38
Экземпляр ПИД-регулятора	38
Подчинённые блоки ПИД-регулятора	39
ПИД-регулятор: последовательности (sequences)	39
ПИД-регулятор: управление группой последовательностей (control sequences)	40
ПИД-регулятор: элемент группы управления последовательностями	40
ПИД-регулятор: гистерезисы (hysteresises)	40
PID: Интервалы гистерезиса	41
7.3 Блоки сохранения данных	41
Блок сохранения данных типа self	41
Блок сохранения данных типа by_name	42
Блок сохранения данных типа by_name_ext	42
Хранение файлов и контрольная сумма	42
7.4 Блоки журналирования и контроля аварий (логгер)	43
Общий параметр для блоков журналирования и контроля аварий	43
Блок журналирования и контроля аварий	43
Параметры элементов блока журналирования и контроля аварий	44

1 ОБЩИЕ СВЕДЕНИЯ

VSM PANEL APP — редактор для создания кода.

Редактор предназначен для программирования автоматики VSM путём создания логических блоков и объединения их в программу VSM прикладного уровня.

Область применения — автоматика инженерных систем зданий, ЖКХ.

2 УСТАНОВКА

2.1 Подготовка рабочей среды

Для работы редактора VSM PANEL APP необходим персональный компьютер с операционной системой Linux Ubuntu Mate версии 22.04.3 LTS для 64-битной аппаратной архитектуры с графическим интерфейсом.

Загрузите установочный образ с официального сайта по ссылке: <https://ubuntu-mate.org>

Системные требования

• Процессор	Core i3
• Архитектура	64 бит
• ОЗУ	4 Гб
• Накопитель HDD или SSD	16 Гб

ВНИМАНИЕ!

Рекомендуется иметь характеристики аппаратного обеспечения не ниже приведённых.

2.2 Установка дополнительных пакетов

Перед запуском редактора VSM PANEL APP установите необходимые для его работы библиотеки.

Для этого откройте терминал и введите команду:

```
sudo apt-get install libboost-all-dev net-tools
```

Подтвердите установку, после чего набор необходимых библиотек будет загружен.

2.3 Установка VSM PANEL APP

1. Загрузите архив panel_app.tar.gz по [ссылке](#)
2. Скопируйте архив в домашнюю папку пользователя операционной системы;

```
cp panel_app.tar.gz ~/
```

3. Перейдите в домашнюю папку;

```
cd ~
```

4. Распакуйте архив;

```
tar -zxvf panel_app.tar.gz
```



2.4 Запуск VSM PANEL APP

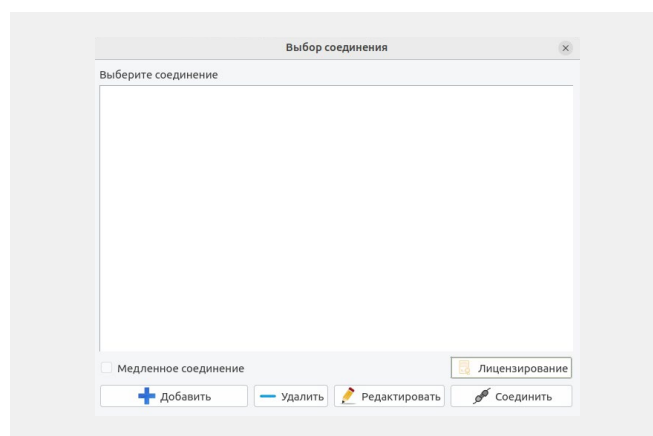
1. Перейдите в домашнюю папку;

```
cd ~
```

2. Запустите VSM PANEL APP

```
./panel_app
```

В результате успешного запуска откроется окно выбора соединения.



3 ПОДКЛЮЧЕНИЕ

Для начала работы в редакторе VSM PANEL APP необходимо выбрать соединение с компьютером VSM, на котором установлено требуемое прикладное программное обеспечение (см. рисунок 1).

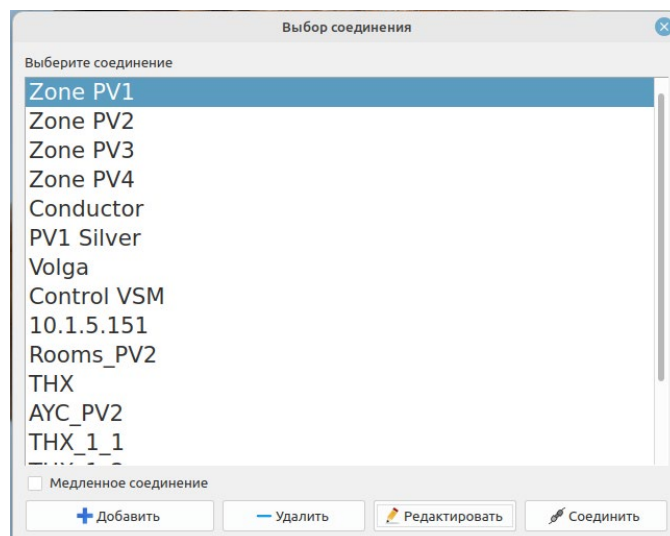


Рисунок 1. Выбор соединения

Если список соединений пуст или в списке нет необходимого соединения, то следует создать его с помощью кнопки «Добавить». При нажатии этой кнопки откроется следующее диалоговое окно (см. рисунок 2).

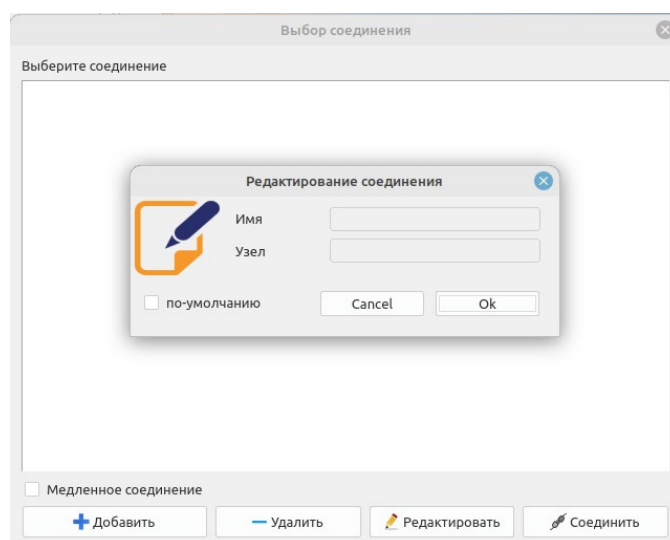


Рисунок 2. Создание соединения

В поле «Имя» необходимо ввести наименование соединения (см. рисунок 3).

В поле «Узел» следует ввести IP-адрес компьютера VSM, на котором будет создаваться или редактироваться прикладное программное обеспечение VSM.

После заполнения полей требуется нажать кнопку «Ok».

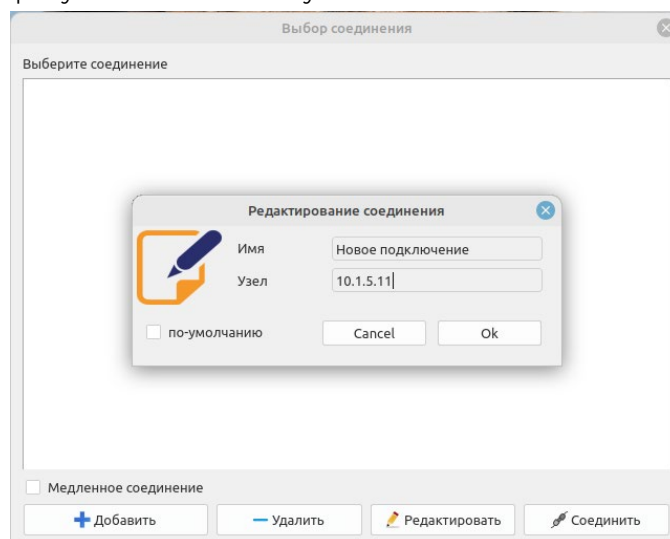


Рисунок 3. Ввод параметров для создания соединения

Созданное соединение появится в списке соединений (см. рисунок 4).

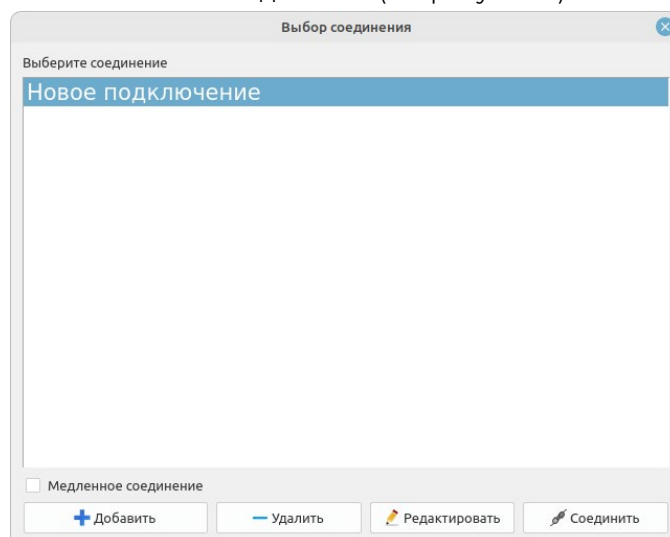


Рисунок 4. Список соединений

После выбора из списка соединений необходимого следует нажать кнопку «Соединить».

На следующем этапе появится всплывающее окно с отображением процесса соединения с компьютером VSM (см. рисунок 5).

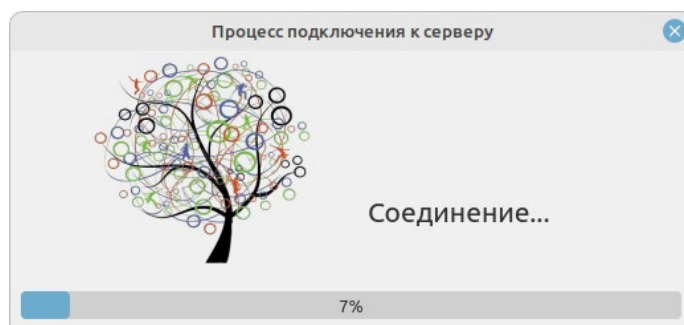


Рисунок 5. Всплывающее окно с отображением процесса подключения

Процесс успешного подключения завершается выводом главного экрана VSM PANEL APP (см. рисунок 6).

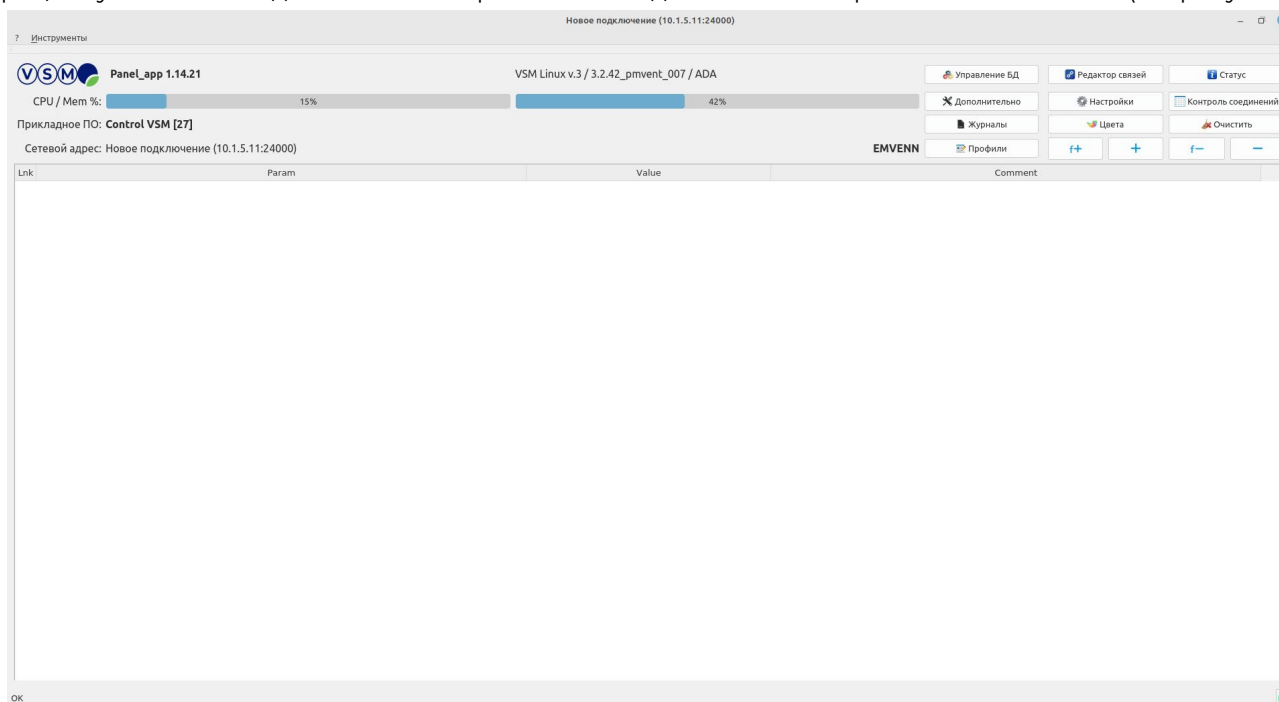



Рисунок 6. Главный экран VSM PANEL APP

4 ЗАВЕРШЕНИЕ РАБОТЫ

Завершение работы в редакторе VSM PANEL APP осуществляется нажатием кнопки  в правом верхнем углу главного экрана.

Примечание: перед завершением работы VSM PANEL APP для сохранения изменений в прикладной программе, сделанных в редакторе, необходимо сохранить базу данных (прикладную программу) на сервере или на локальном компьютере (подробное описание представлено в следующей главе).

5 УДАЛЕНИЕ

1. Перейдите в домашнюю папку

```
cd ~
```

2. Удалите редактор VSM PANEL APP

```
rm -rf panel_app
```

6 ОПИСАНИЕ ИНТЕРФЕЙСА

Внешний вид редактора VSM PANEL APP представлен на рисунке 7.

Числами отмечены текстовые поля, обозначающие:

1. Наименование соединения и (IP-адрес компьютера VSM, к которому выполнено подключение).
2. Текущая версия редактора VSM PANEL APP.
3. Версии прошивки/ ядра /NAND памяти компьютера VSM.
4. Загруженность процессора / дисковой памяти компьютера VSM.
5. Наименование и [код] прикладного программного обеспечения VSM, установленного на компьютер VSM.
6. Наименование соединения и (IP-адрес компьютера VSM, к которому выполнено подключение).
7. Номер процессорного модуля компьютера VSM.
8. Наименование полей таблицы области программирования.
9. Статус корректного функционирования прикладной программы VSM.

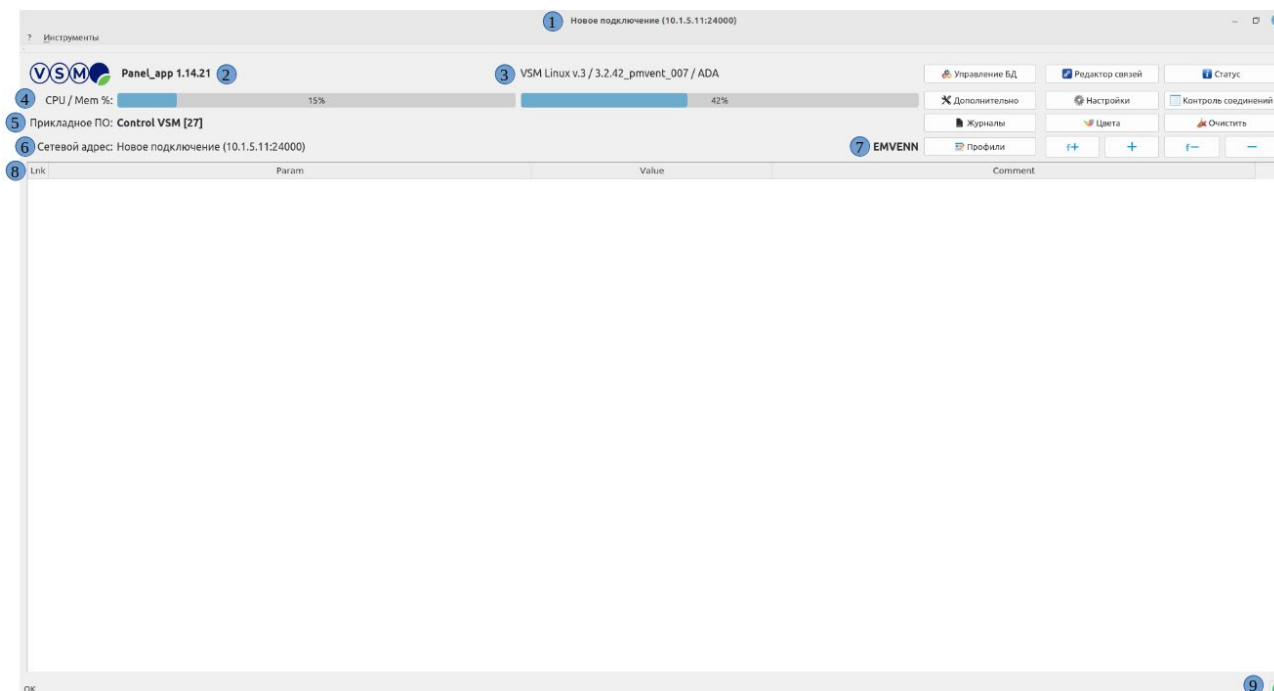


Рисунок 7. Главный экран VSM PANEL APP

На рисунке 8 числами отмечены кнопки редактора VSM PANEL APP, обозначающее:

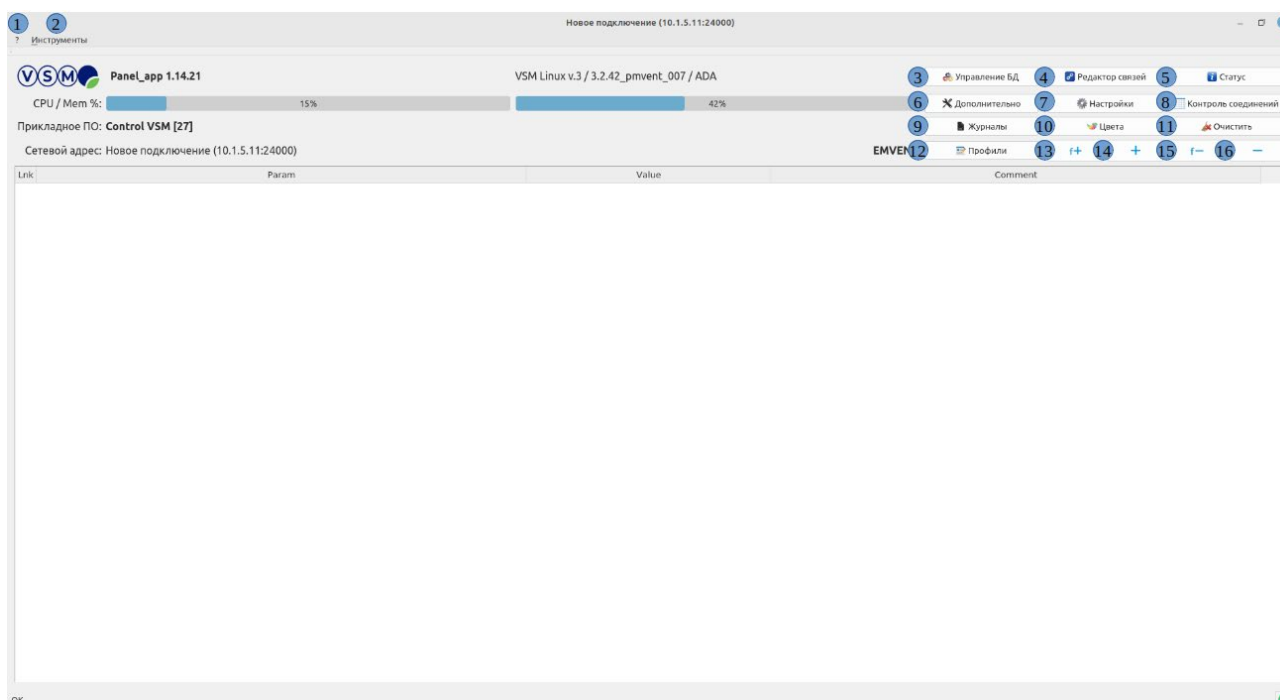


Рисунок 8. Главный экран VSM PANEL APP

1. Подробнее о программе (см. рисунок 9)

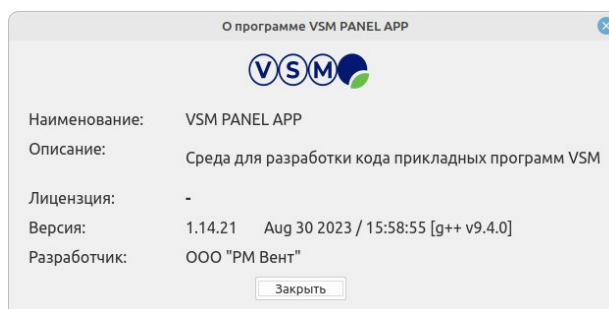


Рисунок 9. Окно о программе

2. При нажатии кнопки «Инструменты» можно выбрать следующие действия:
 - 2.1. Поиск текста в прикладной программе VSM
 - 2.2. Выравнивание столбцов в области программирования
3. При нажатии кнопки «Управление БД» можно выбрать следующие действия (см. рисунок 10):

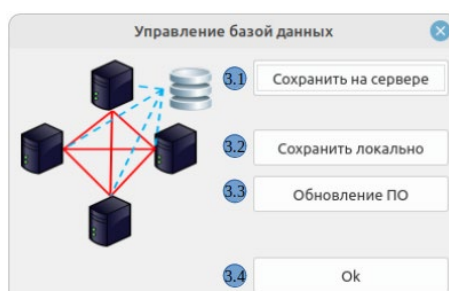


Рисунок 10. Управление БД. Диалоговое окно

- 3.1. Сохранить базу данных (прикладную программу VSM) на сервере.
 - 3.2. Сохранить базу данных (прикладную программу VSM) на локальном компьютере.
 - 3.3. Обновить версию автоматики прикладного программного обеспечения (прикладную программу VSM).
 - 3.4. Сохранение изменений и выход из диалогового окна.
4. При нажатии кнопки «Редактор связей» открывается диалоговое окно, представленное на рисунке 11.

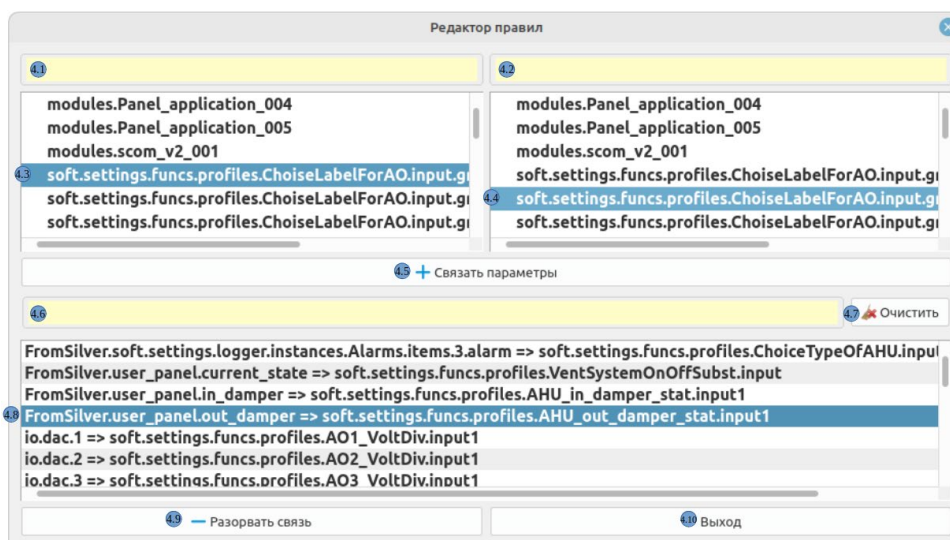


Рисунок 11. Редактор связей. Диалоговое окно

В открывшемся диалоговом окне можно выбрать следующие действия:

- 4.1. Поиск параметра в левом столбце, содержащегося в прикладной программе VSM.
- 4.2. Поиск параметра в правом столбце, содержащегося в прикладной программе VSM.
- 4.3. Список всех параметров, содержащихся в прикладной программе VSM. Здесь выбирается первый параметр для создания связи.

- 4.4. Список всех параметров, содержащихся в прикладной программе VSM. Здесь выбирается второй параметр для создания связи.
 - 4.5. Для образования связи между выбранными ранее параметрами необходимо нажать кнопку «Связать параметры».
 - 4.6. Поиск связи, содержащейся в прикладной программе VSM.
 - 4.7. Очищение поля для поиска связи (4.6).
 - 4.8. Список связей между, содержащихся в прикладной программе VSM. Здесь выбирается связь между двумя параметрами, которую необходимо разорвать.
 - 4.9. Кнопка для разрыва связи параметров.
 - 4.10. Выход из диалогового окна.
5. При нажатии кнопки «Статус» открывается диалоговое окно, представленное на рисунке 12. Здесь находится основная информация о компьютере VSM, системном и прикладном программном обеспечении VSM.

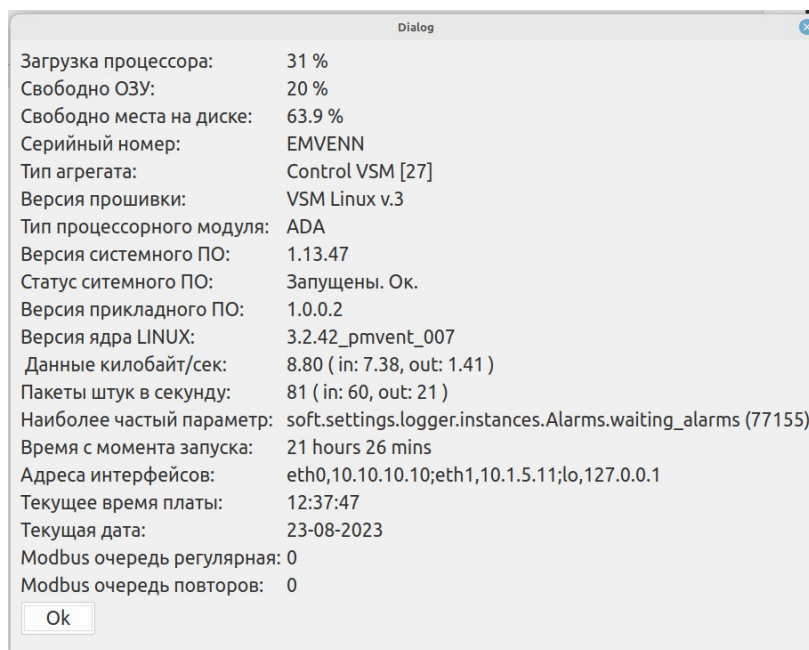


Рисунок 12. Статус. Диалоговое окно

6. При нажатии кнопки «Дополнительно» открывается диалоговое окно, представленное на рисунке 13.
- Кнопка «Удалить дубликаты в списке» позволяет удалить в буфере обмена дублирующиеся имена из скопированный строки.

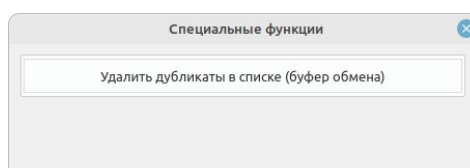


Рисунок 13. Дополнительно. Диалоговое окно

Пример:

На рисунке 14 представлены 7 параметров, записанных в одной строке (см. рисунок 14). В данной строке параметров имеется два с одинаковым обозначением.

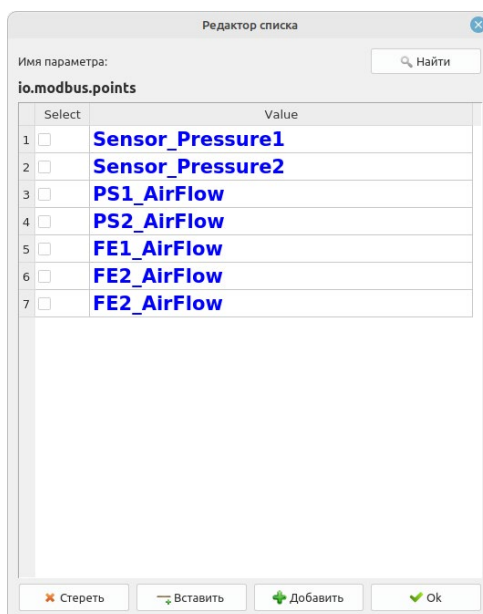


Рисунок 14. Представление строки параметров в виде таблицы

Примечание: для просмотра строки в табличном виде (рисунок 14) необходимо 2 раза кликнуть компьютерной мышью на строку на главном экране (рисунок 15).

На рисунке 15 представлен внешний вид строки параметров.

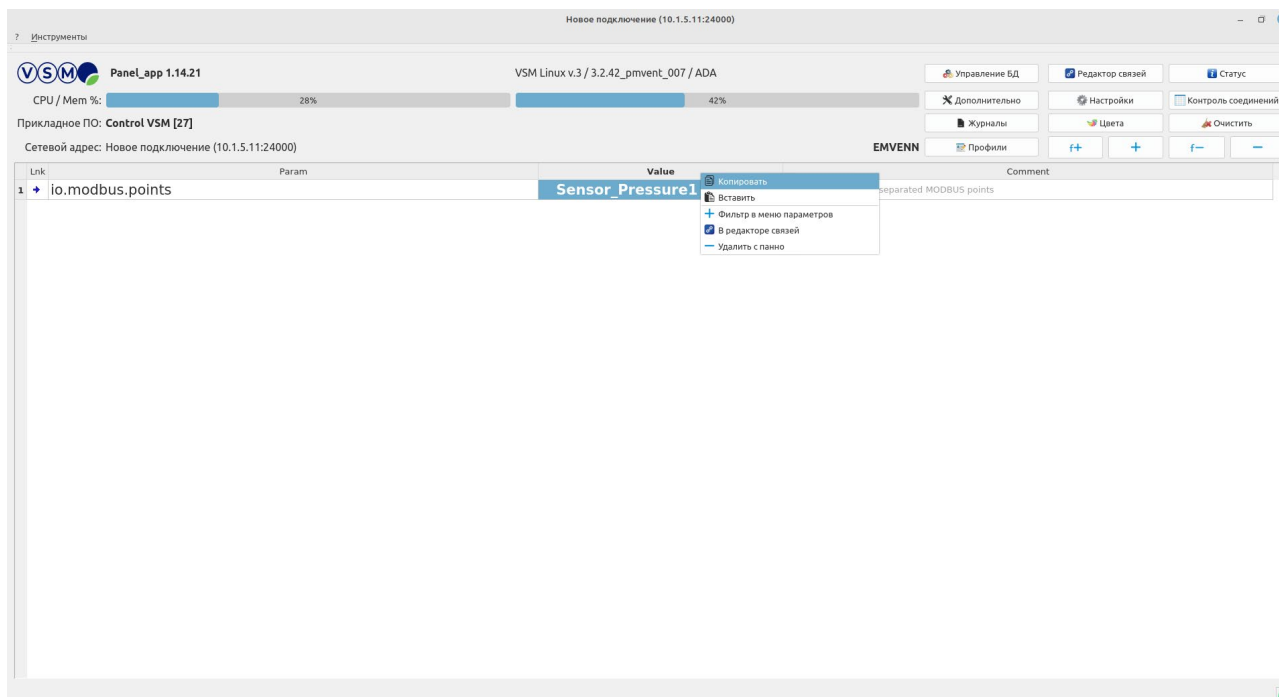


Рисунок 15. Вид строки. Процесс копирования строки

При нажатии на правую кнопку компьютерной мыши появляется диалоговое окно. Для того, чтобы скопировать строку в буфер обмена, необходимо нажать на «Копировать».

На следующем этапе следует нажать на кнопку «Дополнительно» на главном экране VSM PANEL APP и в диалогом окне нажать на кнопку «Удалить дубликаты в списке» (см. рисунок 16) и закрыть диалоговое окно.

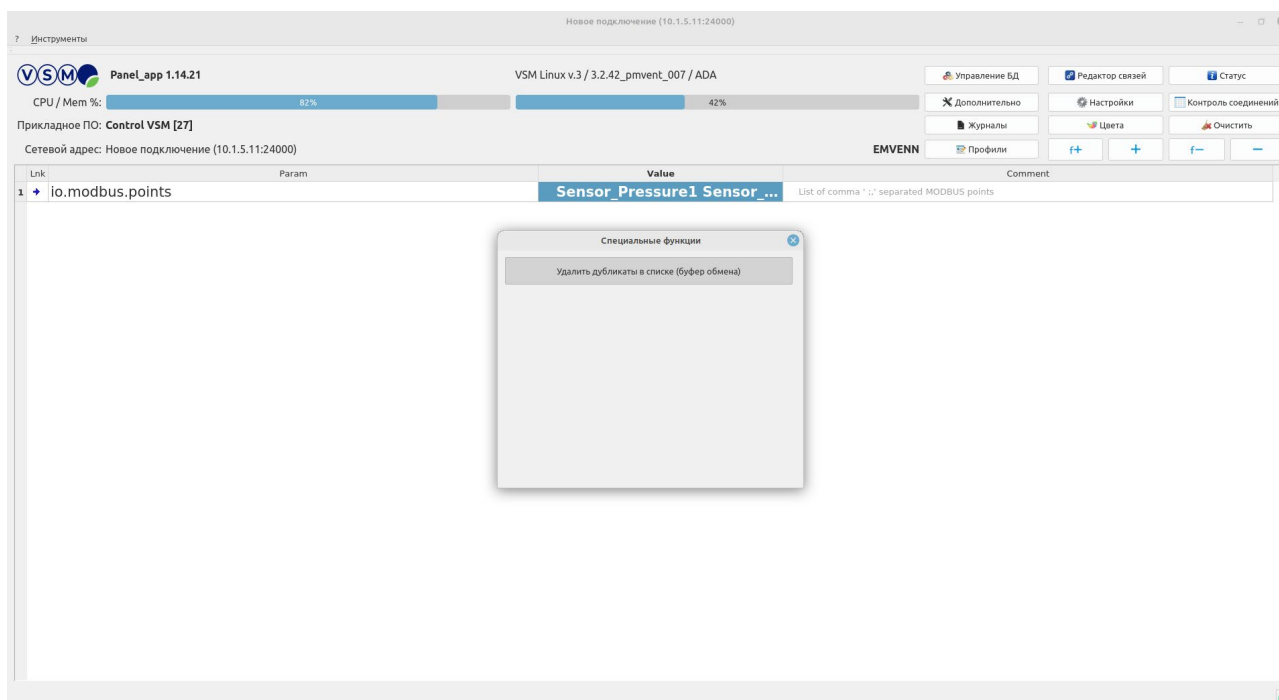


Рисунок 16. Нажатие кнопки «Удалить дубликаты в списке» в диалоговом окне

Далее необходимо выбрать строку параметров, нажать на правую кнопку компьютерной мыши и в появившемся диалоговом окне выбрать «Вставить» (см. рисунок 17).

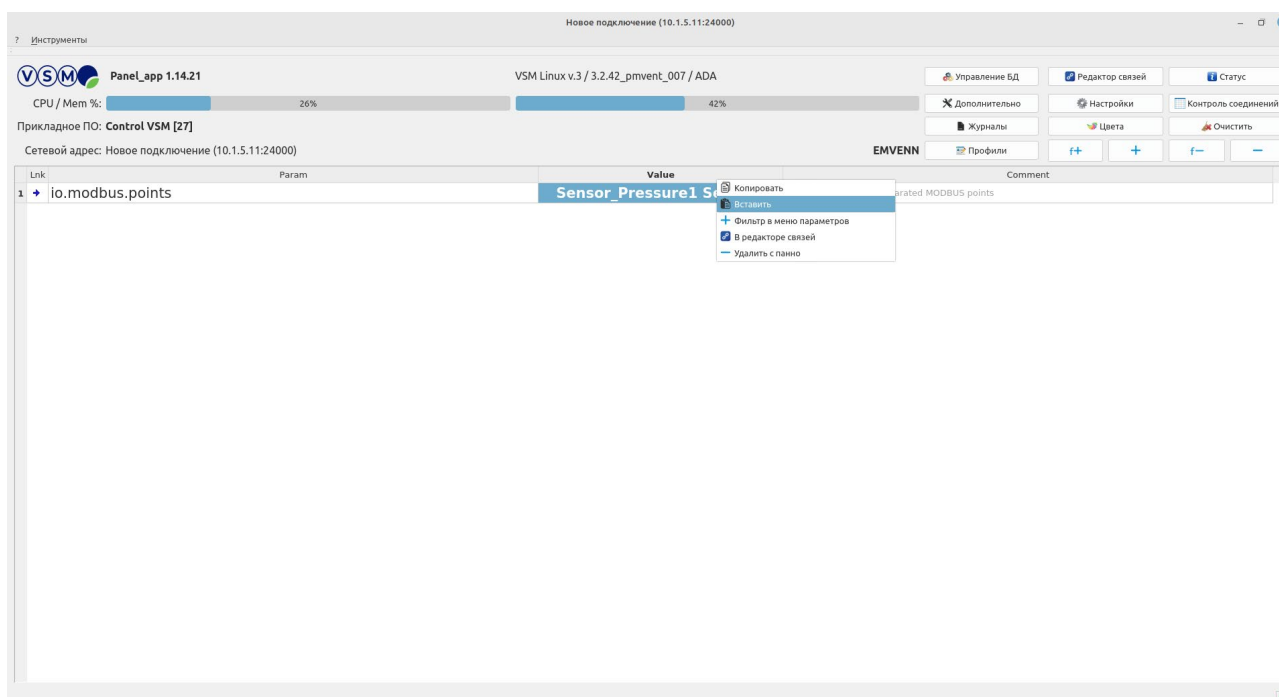


Рисунок 17. Вид строки. Процесс вставки строки

При правильном выполнении всех шагов дубликат будет удалён из строки (см. рисунок 18).

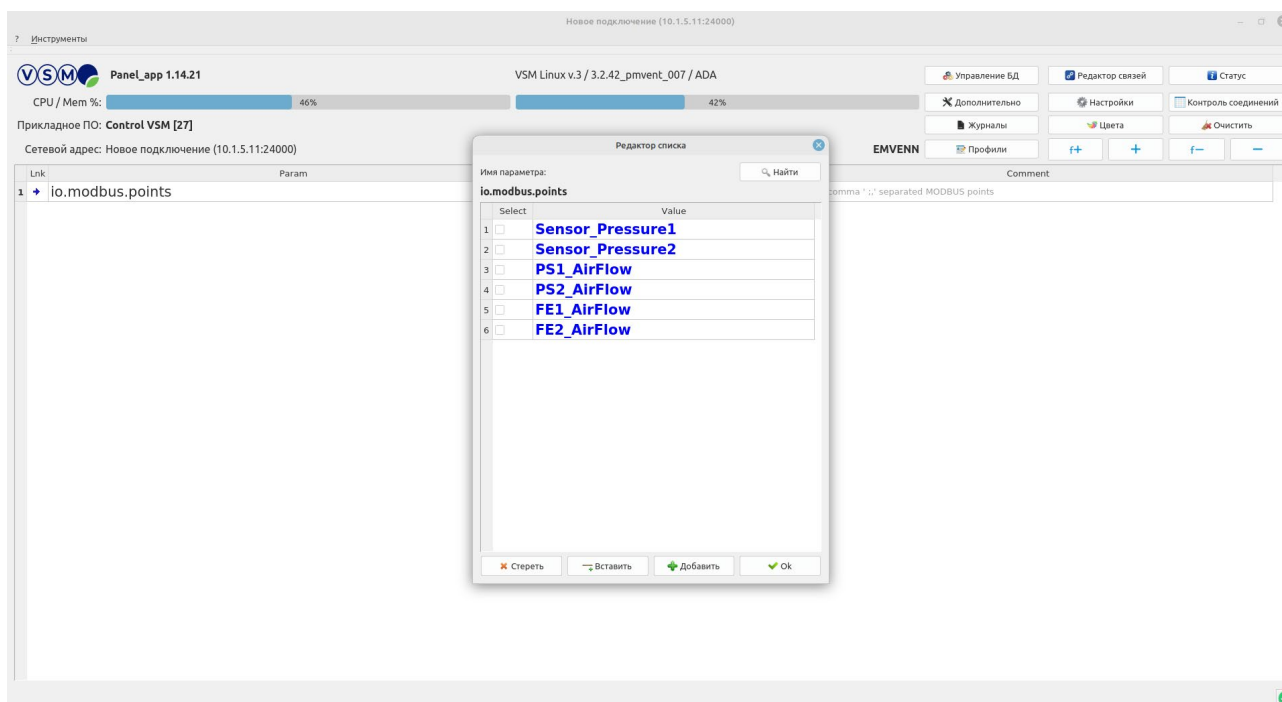


Рисунок 18. Представление строки параметров в виде таблицы

7. Нажатие кнопки «Настройки» приводит к открытию диалогового окна, представленного на рисунке 19.
 - 7.1. Функция для отображения строк в табличной форме.
 - 7.2. Функция загрузки последнего профиля пользователя.
 - 7.3. Функция доступна только для разработчиков редактора VSM PANEL APP.

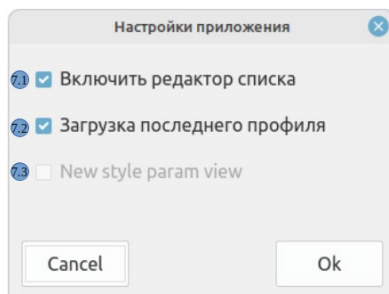


Рисунок 19. Настройки. Диалоговое окно

8. При нажатии кнопки «Контроль соединений» открывается диалоговое окно, представленное на рисунке 20.
 - 8.1. В таблице представлена информация (наименование, IP-адрес, версия) о всех клиентах, подключенных к компьютеру VSM:
 - 8.1.1. Компонент системы диспетчеризации SCOM VSM.
 - 8.1.2. Редактор VSM PANEL APP (новая версия).
 - 8.1.3. Компонент системы диспетчеризации SCOM VSM.
 - 8.1.4. Компонент системы диспетчеризации SCOM VSM.
 - 8.1.5. CONTROL SERVICE обеспечивает обновление ПО VSM, выгрузку данных и журналов функционирования.
 - 8.1.6. Компонент системы диспетчеризации SCOM VSM.
 - 8.1.7. SENSE MONITOR обеспечивает обмен данными по протоколу Modbus TCP/RTU (Master/Slave), опрос цифровых температурных датчиков 1wire.
 - 8.1.8. Редактор VSM PANEL APP (старая версия).
 - 8.1.9. DEVICE CONTROL обеспечивает доступ к цифровым и аналоговым входам/выходам компьютера VSM, DIP-переключателям, управление светодиодами
 - 8.1.10. LOGIC SERVICE обеспечивает интерпретацию и выполнение FBD блоков прикладной программы.

- 8.2. Время работы сервисов VSM с момента запуска компьютера VSM.
- 8.3. Количество параметров, содержащихся в прикладной программе VSM.
- 8.4. Количество активных FBD блоков, содержащихся в прикладной программе VSM.
- 8.5. Количество активных ПИД-регуляторов, содержащихся в прикладной программе VSM.

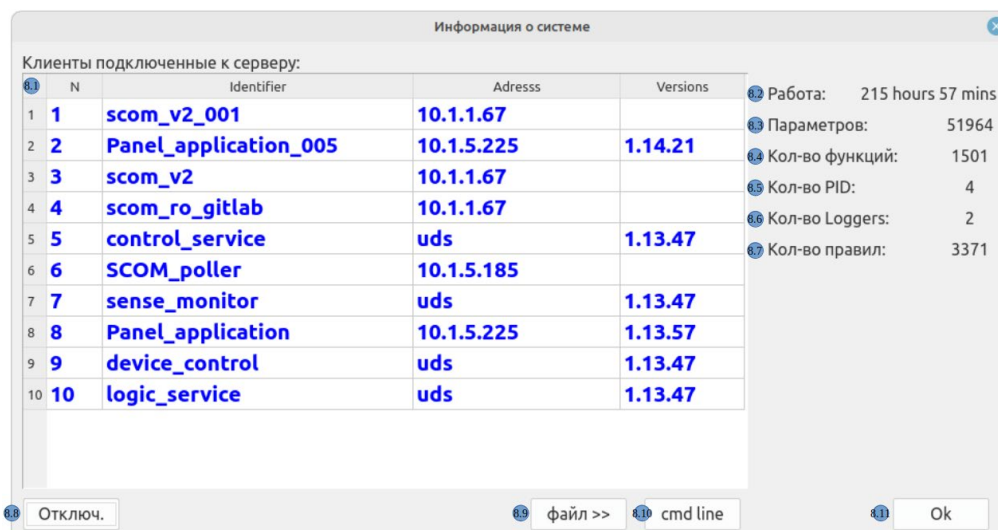


Рисунок 20. Контроль соединений. Диалоговое окно

- 8.6. Количество журналов аварий и протоколирования параметров.
 - 8.7. Количество связей, содержащихся в прикладной программе VSM.
 - 8.8. Принудительное отключение выбранного клиента (см. пункт 8.1) от компьютера VSM.
 - 8.9. Отправка файла обновлений выбранному клиенту (см. пункт 8.1).
 - 8.10. Кнопка доступна только разработчикам редактора VSM PANEL APP.
 - 8.11. Подтверждение действий и выход из текущего диалогового окна.
9. При нажатии кнопки «Журналы» открывается диалоговое окно, представленное на рисунке 21.

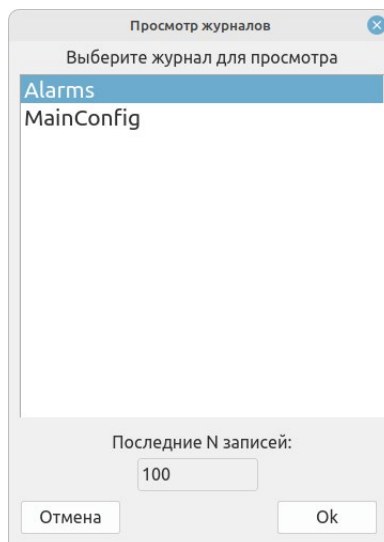


Рисунок 21. Журналы. Диалоговое окно

Здесь перечислены журналы аварий и протоколирования параметров, содержащиеся в прикладной программе VSM.

Для просмотра содержимого журнала необходимо выбрать его из списка и нажать кнопку «Ок». Откроется диалоговое окно, представленное на рисунке 22.

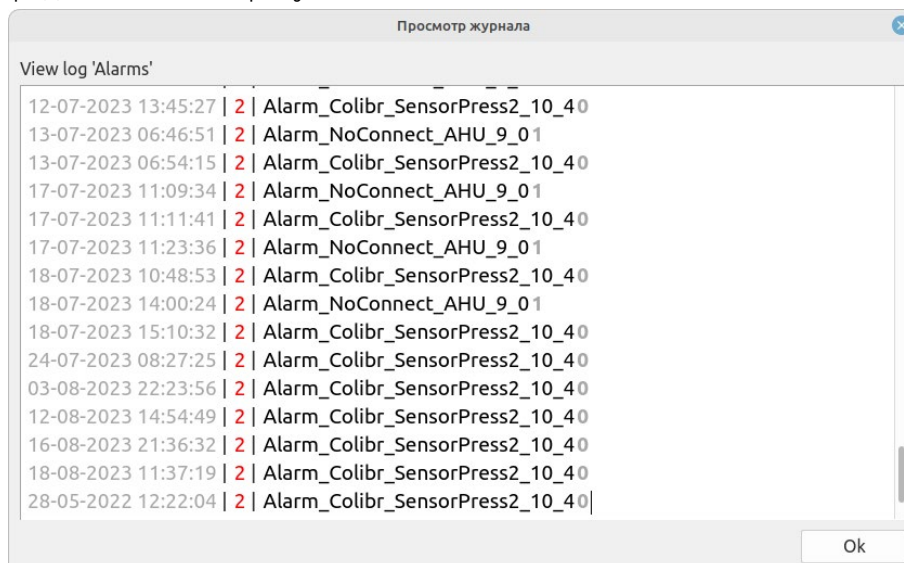


Рисунок 22. Содержимое журнала

10. При нажатии кнопки «Цвета» открывается диалоговое окно, представленное на рисунке 23.

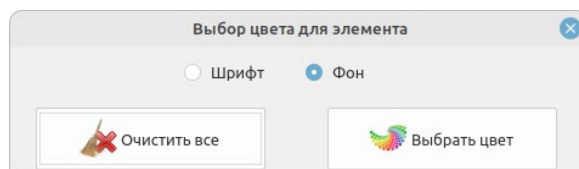


Рисунок 23. Цвета. Диалоговое окно

Здесь можно выбрать изменение цвета для шрифта или фона.

Нажатие кнопки «Выбрать цвет» открывает диалоговое окно с настройками цвета (см. рисунок 24).

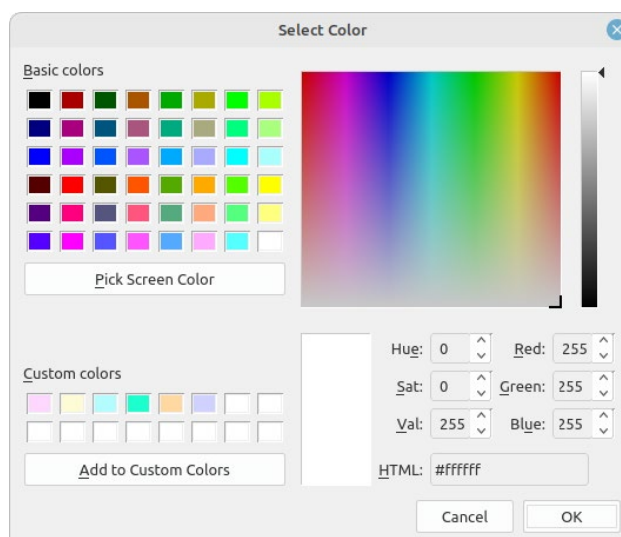


Рисунок 24. Настройки цвета

Здесь можно выбрать цвет из основной палитры, определить цвет на экране и установить его или задать собственный.

Пример: Изменение цвета фона строки пользовательским цветом.

1 шаг: Выделение строки целиком, нажатием на номер строки.

2 шаг: Нажатие кнопки «Цвет» и выбор параметра «Фон» (рисунок 25).

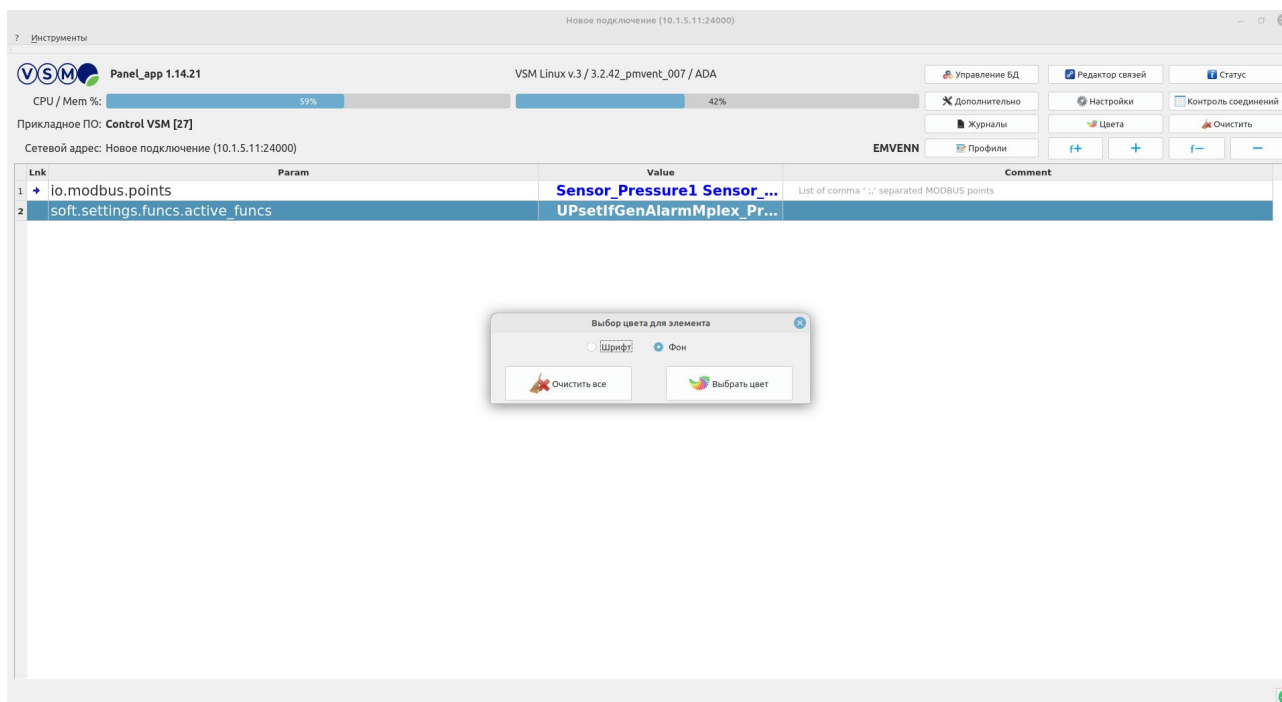


Рисунок 25. Изменение фона у выбранной строки. Шаг 1 и 2

3 шаг: После нажатия кнопки «Выбрать цвет» необходимо выбрать пустое (белое) поле в таблице «Custom colors» и установить настройки цвета по желанию с помощью спектра или указывая интенсивность красного, зеленого и синего цвета (рисунок 26).

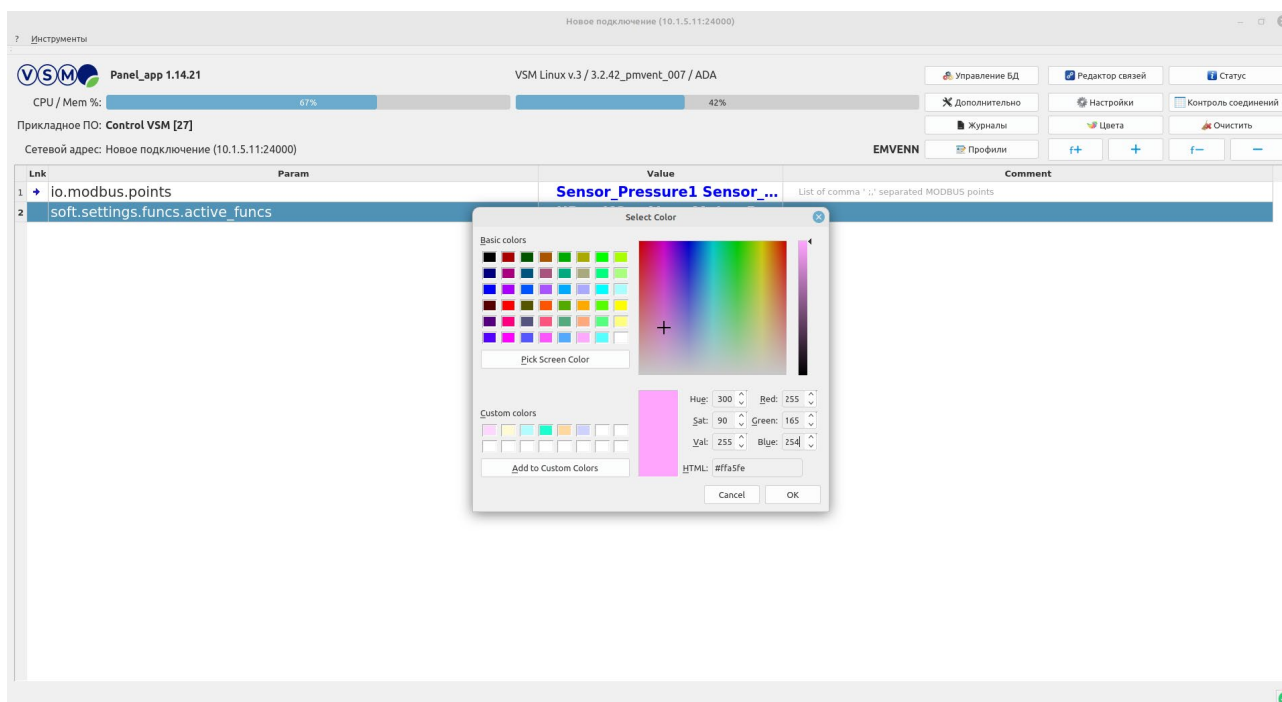


Рисунок 26. Изменение фона у выбранной строки. Шаг 3

4 шаг: После завершения настройки цвета необходимо нажать кнопку «Add to Custom Colors» для его добавления в палитру цветов пользователя (рисунок 27) и нажать кнопку «OK».

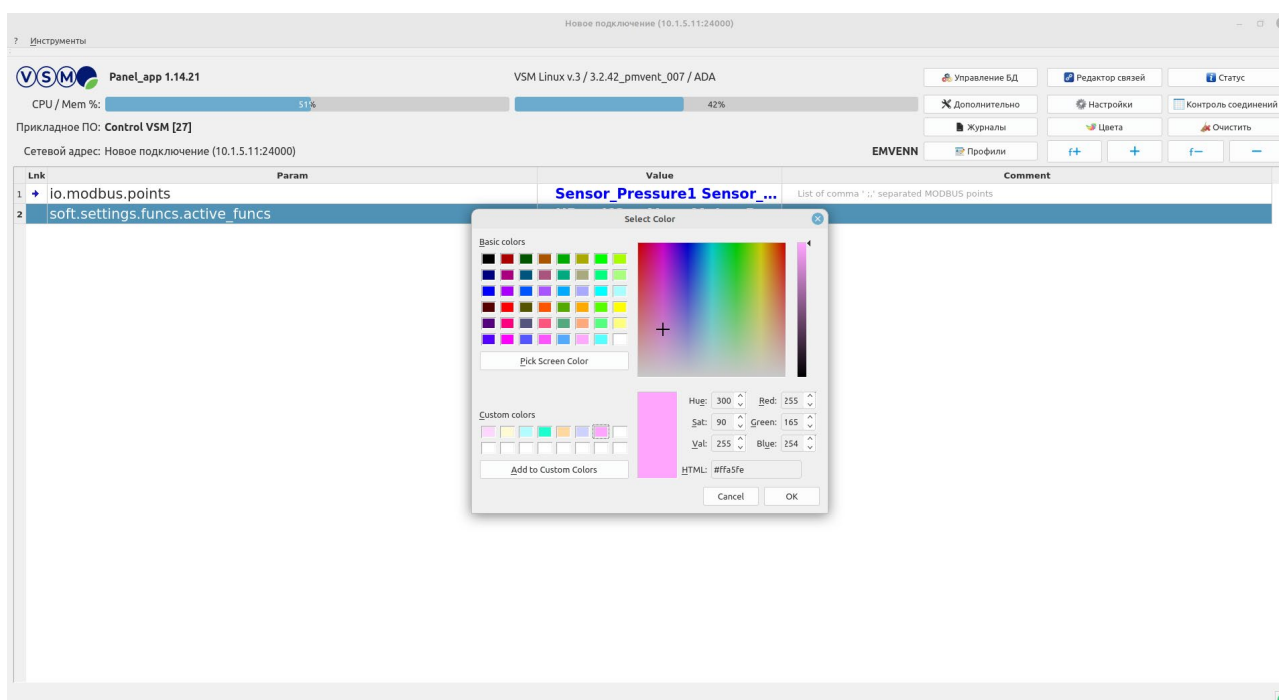


Рисунок 27. Изменение фона у выбранной строки. Шаг 4

5 шаг: Демонстрация изменённого цвета фона строки (рисунок 28).

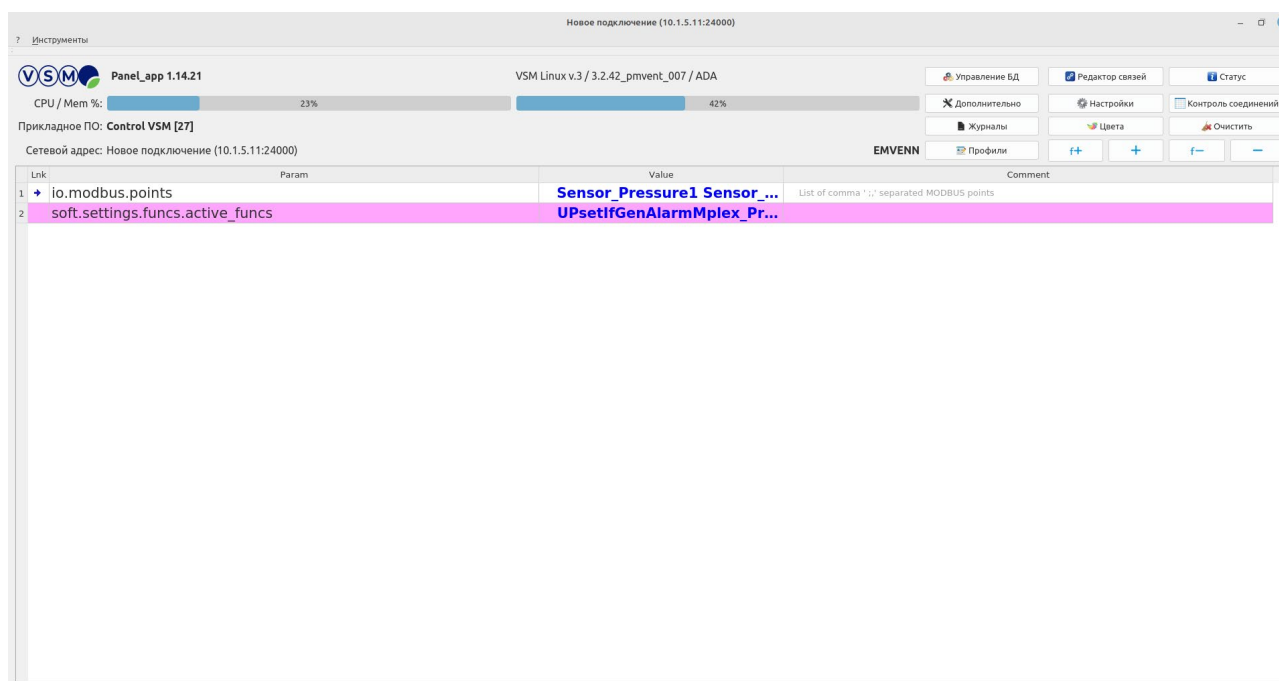


Рисунок 28. Изменение фона у выбранной строки. Шаг 5

Нажатие кнопки «Очистить всё» приводит к сбросу цветового выделения шрифта текста в строках и/или фона строк таблицы области программирования.

11. Нажатие кнопки «Очистить» приводит к удалению выведенных в редакторе VSM PANEL APP строк в области программирования, но не удаляет эти строки из прикладной программы VSM (см. рисунок 29).

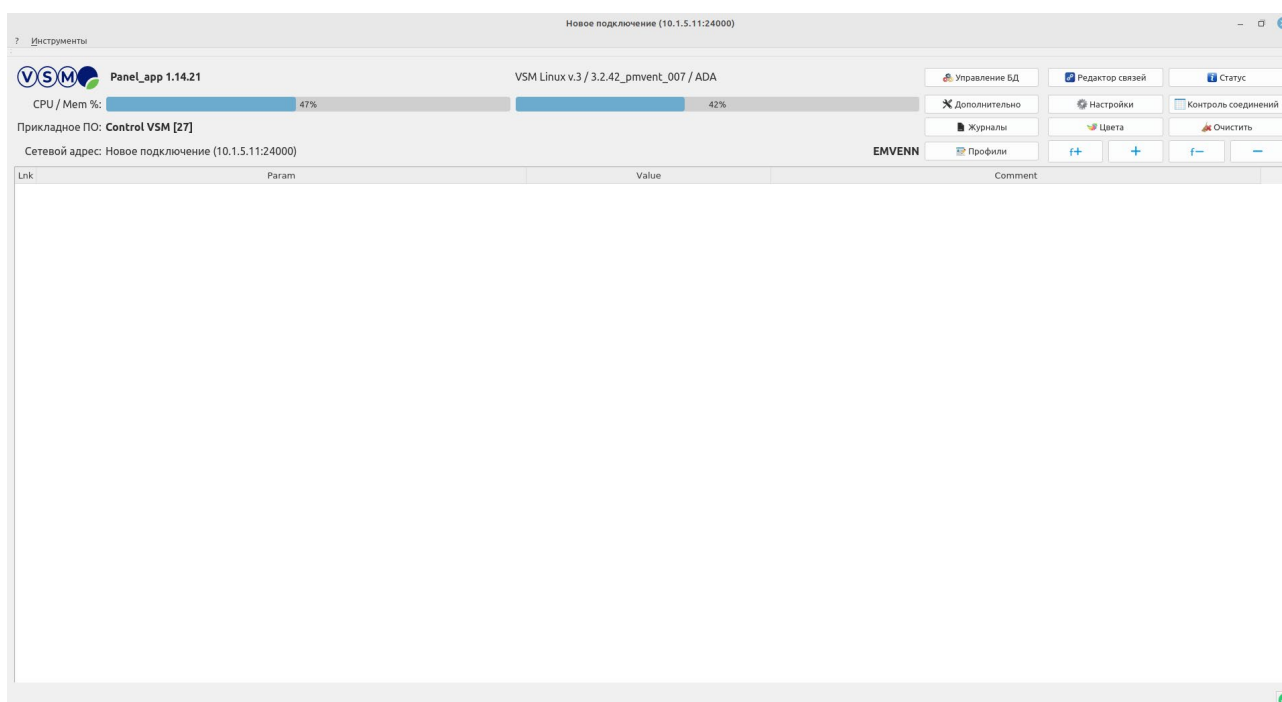


Рисунок 29. Внешний вид главного экрана редактора VSM после нажатия кнопки «Очистить»

12. При нажатии кнопки «Профили» открывается диалоговое окно, представленное на рисунке 30.

Профиль — сохранённая последовательность выведенных в редакторе строк с учётом цветовых настроек фона и шрифта.

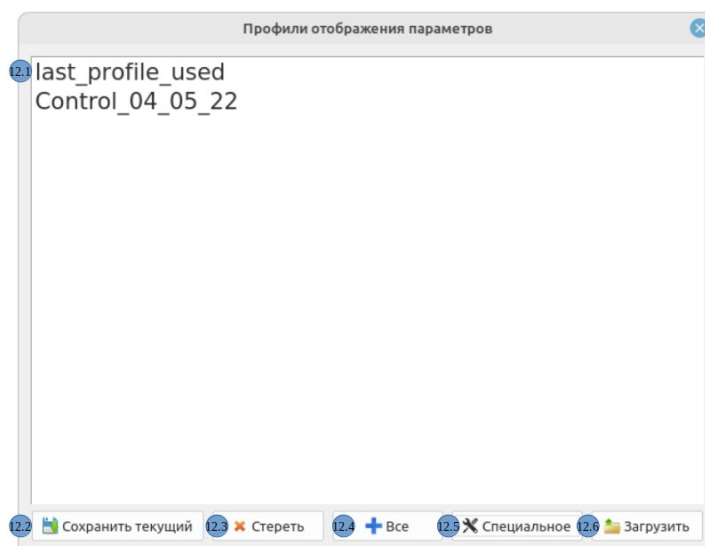


Рисунок 30. Профили. Диалоговое окно

В открывшемся диалоговом окне:

- 12.1. Список сохранённых профилей пользователя.
- 12.2. Кнопка для сохранения текущего профиля.
- 12.3. Удаление профиля из списка сохранённых.
- 12.4. Выделение всех профилей в списке.
- 12.5. Нажатие кнопки «Специальное» приводит к открытию диалогового окна, см. рисунок 31.
- Выбрав профиль из списка, его можно сохранить на компьютере пользователя, нажав кнопку «Выгрузить в файл XML».

- Для добавления профиля в список профилей редактора с компьютера пользователя следует нажать кнопку «Загрузить из файла XML».

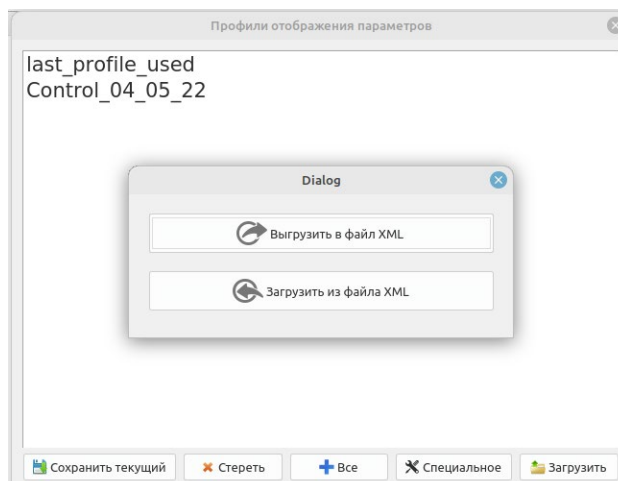


Рисунок 31. Профили. Диалоговое окно при нажатии кнопки «Специальное»

12.6. Загрузка в редактор выбранного профиля.

13. Нажатие кнопки «f+» приводит к открытию диалогового окна, представленного на рисунке 32.

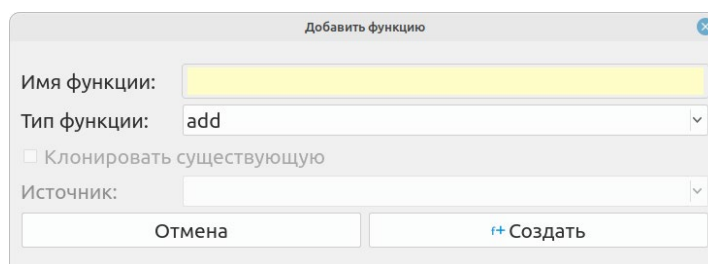


Рисунок 32. Создание функции. Диалоговое окно

Здесь создаются новые функции (FBD блоки) прикладной программы VSM (подробнее о функциях см. в следующей главе).

Для создания функции необходимо присвоить ей имя, выбрать тип из раскрывающегося списка и нажать кнопку «Создать».

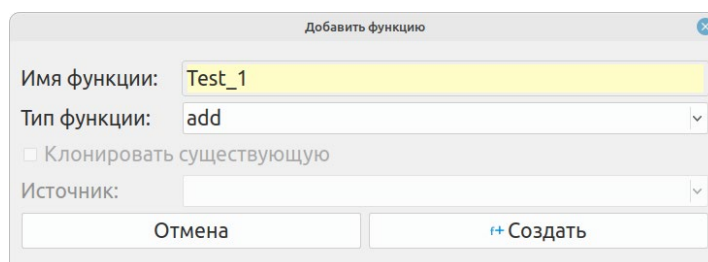


Рисунок 33. Создание функции. Диалоговое окно

После выполненных действий диалоговое окно закрывается, а на главном экране в области программирования появляются строки для настройки функции (см. рисунок 34).

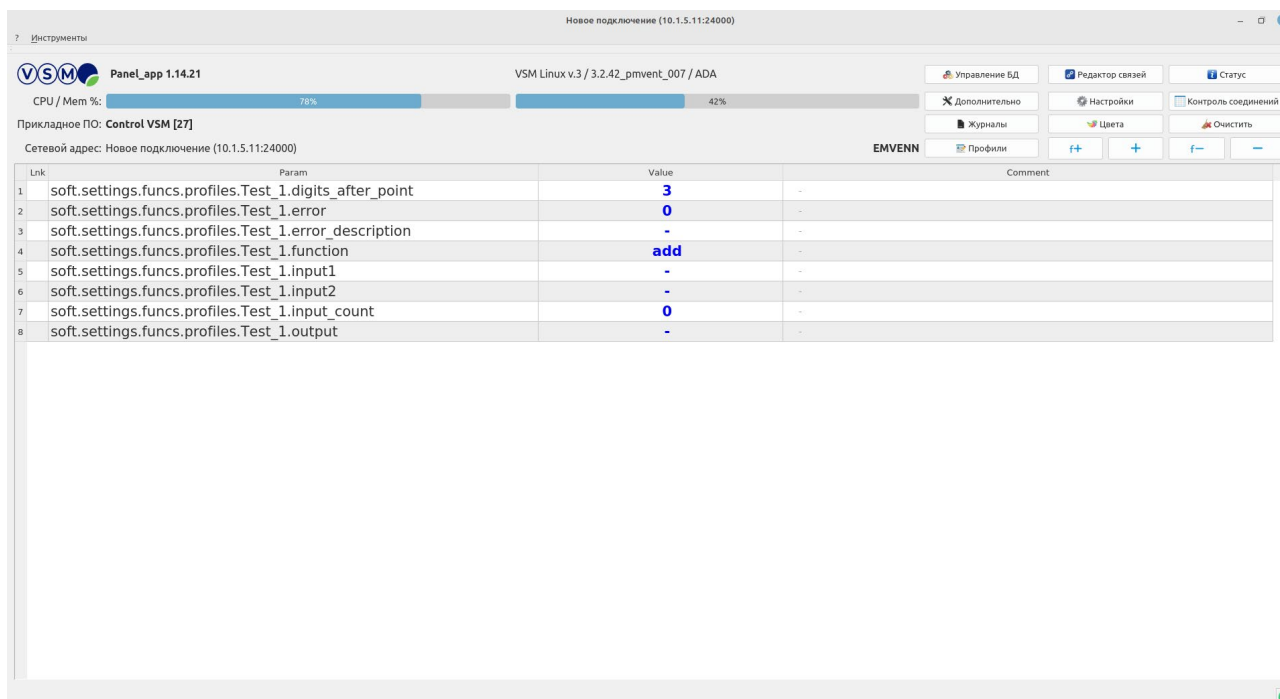


Рисунок 34. Вид главного экрана VSM PANEL APP после создания функции

14. Нажатие кнопки «+» приводит к открытию диалогового окна, представленного на рисунке 35.

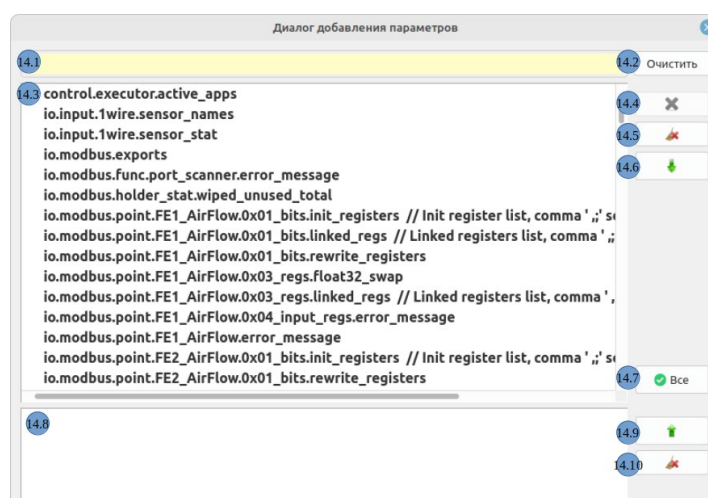


Рисунок 35. Добавление. Диалоговое окно

В открывшемся диалоговом окне находятся следующие элементы:

- 14.1 Поле ввода для поиска параметров в прикладной программе VSM.
- 14.2 Кнопка для сброса текста в поле ввода 14.1.
- 14.3 Поле со списком параметров, содержащихся в прикладной программе VSM.

Для того, чтобы выделить параметр/параметры необходимо кликнуть на него/на каждый из них левой кнопкой компьютерной мыши один раз. При двойном клике параметр добавляется в поле 14.8 (см. рисунок 36).

- 14.4 Кнопка доступна только разработчикам редактора VSM PANEL APP.
- 14.5 Кнопка для отмены выделения параметров в списке 14.3
- 14.6 Кнопка для переноса выделенного параметра/параметров в поле 14.8.
- 14.7 Кнопка для выделения и переноса в поле 14.8 всех параметров из поля 14.3.

14.8 Поле с параметрами из прикладной программы VSM, которые выбраны для добавления в редактор VSM PANEL APP.

- 14.9 Кнопка для удаления выбранного параметра из поля 14.8.

14.10 Кнопка для удаления всех параметров из поля 14.8.

14.11 Поле для ввода имени нового параметра.

14.12 Кнопка для добавления параметров из поля 14.8 или 14.11 в редактор VSM PANEL APP.

Примечание: нажатие кнопки 14.12 добавляет параметр из поля 14.11 не только в редактор, но и в прикладную программу VSM.

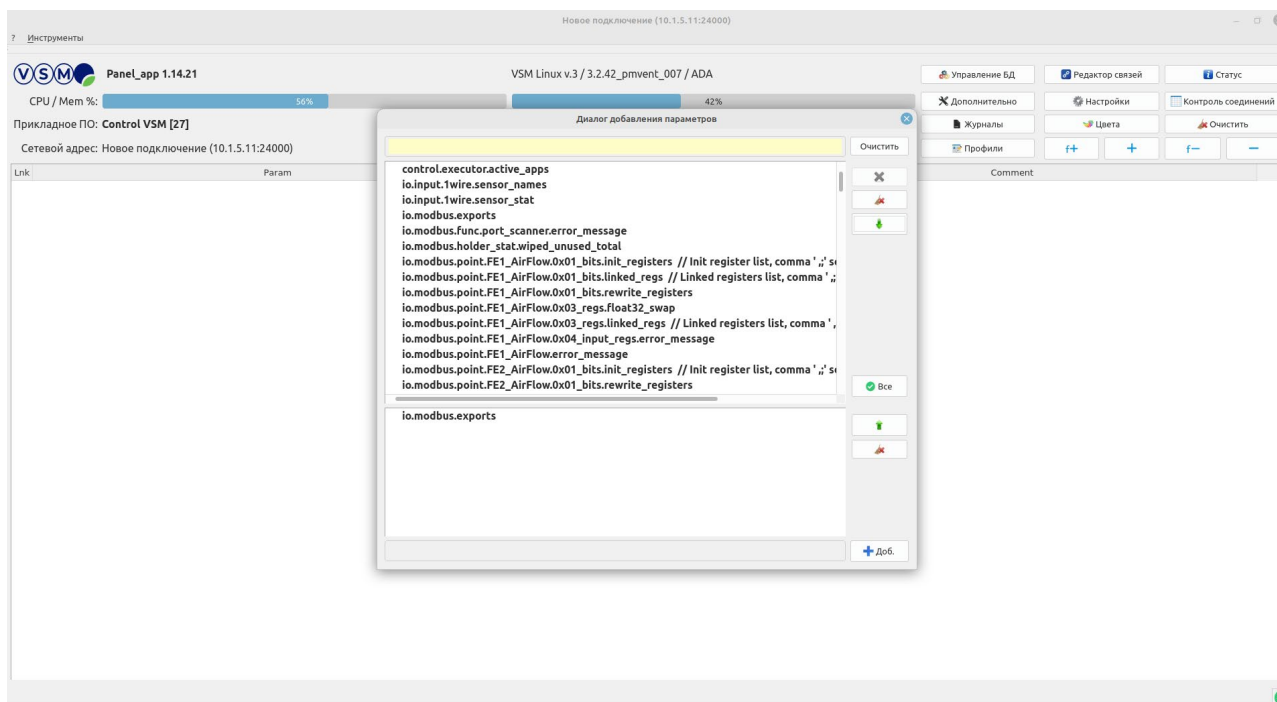


Рисунок 36. Выделение и добавление параметра в поле 14.8

15. Нажатие кнопки «f-» приводит к открытию диалогового окна, представленного на рисунке 37.

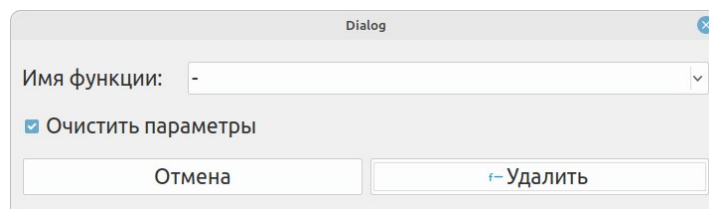


Рисунок 37. Удаление функции. Диалоговое окно

В раскрывающемся списке «Имя функции» перечислены все названия функций, содержащиеся в прикладной программе VSM (см. рисунок 38).

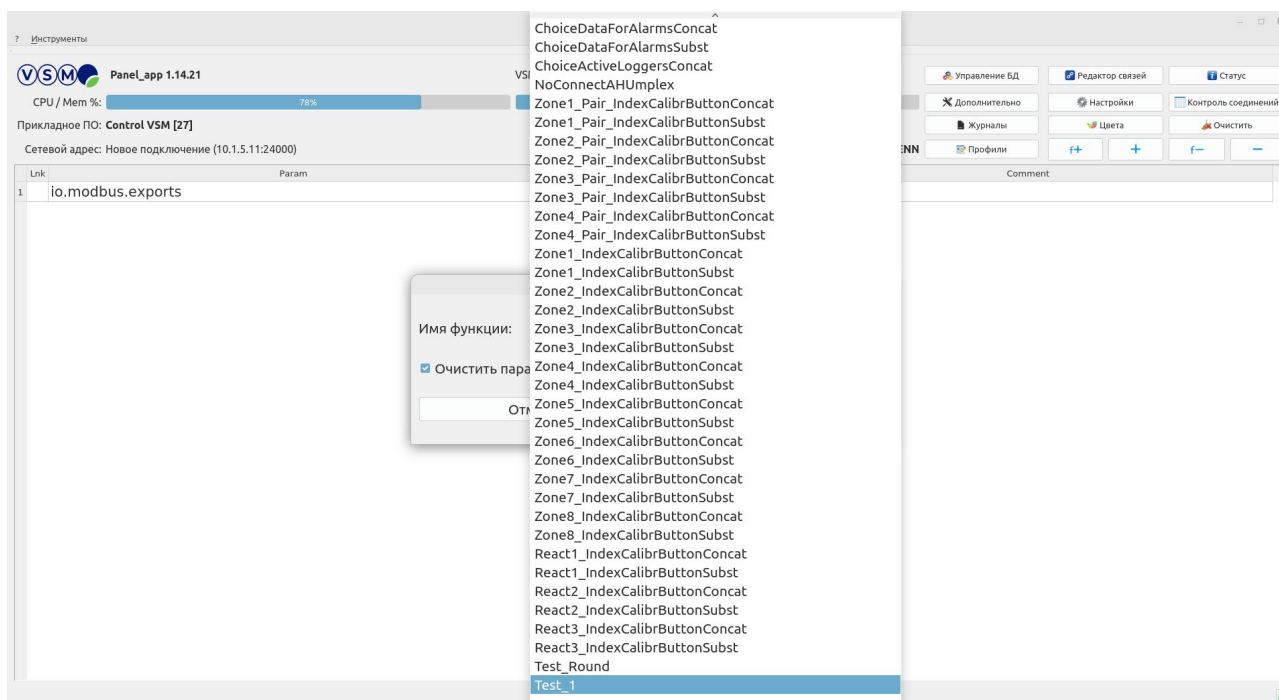


Рисунок 38. Раскрывающийся список «Имя функции»

Для быстрого поиска необходимой функций можно начать печатать её название на клавиатуре.

Для удаления выбранной функции из прикладной программы VSM следует нажать кнопку «Удалить» (см. рисунок 39).



Рисунок 39. Удаление функции. Диалоговое окно

Флаг «Очистить параметры» предназначен для полного удаления функции из прикладной программы VSM. Если флаг выключен, то функция будет удалена из списка активных функций, но её параметры останутся в прикладной программе.

16. Кнопка «←» предназначена для удаления выбранных строк из области программирования на главном экране редактора VSM PANEL APP.

7 ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ БЛОКОВ

7.1 Логические и математические функции

Описание функций представлено в таблице 1.

Таблица 1. Логические и математические функции

Имя функции	Количество входных параметров	Описание
and	1 и более	Конъюнкция или логическое умножение «И» (AND)
or	1 и более	Дизъюнкция или логическое сложение «ИЛИ» (OR)
xor	1 и более	Функция «ИСКЛЮЧАЮЩЕЕ ИЛИ» (XOR)
min	1 и более	Определение минимального значения среди входных значений
max	1 и более	Определение максимального значения среди входных значений
max_sequence	Входные значения: · input_value — входное значение Настройки: · enable — вкл./выкл. работы блока. При включении найденное максимальное число инициализируется текущим значением входа. · digits_after_point — количество знаков после запятой на выходе блока Выходные значения: · output_max — максимальное значение	Определение максимального значения в ряде чисел, поступающих на один вход
mod	1 и более	Модуль числа
sub	2 и более	Вычитание входных значений
add	2 и более	Сложение входных значений
mult	2 и более	Умножение входных значений
div	2 и более	Деление входных значений
equal	2 и более	Определение равенства значений На выходе булево значение: 0 или 1
less	2 и более	Оператор сравнения «Меньше» (<) На выходе булево значение: 0 или 1
less_equal	2 и более	Оператор сравнения «Меньше или равно» (≤) На выходе булево значение: 0 или 1
more	2 и более	«Больше» (>) На выходе булево значение: 0 или 1
more_equal	2 и более	Оператор сравнения «Больше или равно» (≥) На выходе булево значение: 0 или 1
concat	2 и более · use_spaces — использование пробелов в качестве разделителя для склеивания · use_delimeter — использование значения параметра delimeter в качестве разделителя · delimeter — значение разделителя, если use_delimeter = 1	Конкатенация текстовых строк. Пробел в качестве разделителя при use_spaces = 1, без пробелов use_spaces = 0
exp	1	Экспонента e^X , где X — степень (входное значение)
log	1	Натуральный логарифм $\ln(X)$, где X — входное значение

Имя функции	Количество входных параметров	Описание
sqrt	1 или 2	Корень n-ой степени $\sqrt[n]{X}$, где X — подкоренное выражение (первый вход), n — показатель корня (второй вход)
pow	1 или 2	Возведение в степень X^n , где X — основание степени n — показатель степени Если блок имеет один вход, то выходное значение=X ² Если блок имеет два входа: X — первый вход, n — второй
range	3 · Первый вход — текущее значение, · Второй вход — минимальное значение, · Третий вход — максимальное значение	Ограничение диапазона числа
boolean_inverse	· count — Количество входов выходов	Инверсия или логическое отрицание «НЕ»
random	· begin_range — начало диапазона, целое число	Генерация случайного числа в заданном диапазоне с заданной частотой
	· end_range — конец диапазона, целое число	
	· timeout — интервал генерации значений в миллисекундах	
round	· count — количество входов и выходов · base — основание для округления	Округление чисел: основание = 0 — округление до целого числа, основание = 10 — округление до десятков, основание =100 — округление до сотен и т.д.
numeric_compare	Входные значения: · value_a — входное значение A · value_b — входное значение B Выходные значения: · is_equal = 1, если значения равны, 0 — нет · is_less = 1 если значение A меньше B, 0 — нет · is_more = 1 если значение A больше B, 0 — нет · result = 1 (значения равны), = 2 (A меньше B), = 3 (A больше B)	Сравнение двух чисел. Выходы — результат сравнения
enthalpy	Входные значения: · set_temp — заданное значение температуры · set_h — заданное значение относительной влажности Выходные значения: · d_set — абсолютная влажность · i_set — энтальпия	Расчёт энтальпии и абсолютной влажности
range_alarm	· input — входное значение · min — минимальный порог входного значения · max — максимальный порог входного значения · timer — время отложенной реакции в секунда, если 0 = выключено · out_of_min — флаг выхода за пределы минимума (1 - да, 0 - нет) · out_of_max — флаг выхода за пределы максимума (1 - да, 0 - нет) · out_of_range — флаг выхода за пределы диапазона (1 - да, 0 - нет) · output — выходное значение	Контроль диапазона числа с сигнальными выходами и отложенной реакцией по таймеру. Если таймер выключен (timer=0) флаги выставляются сразу в соответствии с входящей величиной. Если таймер включен, то флаги выставляются по истечении времени таймера, при условии что за время работы таймера входящее значение постоянно находилось за пределами диапазона.

Имя функции	Количество входных параметров	Описание
pwm	<ul style="list-style-type: none"> a_value — коэффициент A interval — интервал в секундах T 	ШИМ Выходное значение меняется от 0 к 1 по заданному интервалу. Интервал считается по формуле $A \cdot T$
pwm_milli	<ul style="list-style-type: none"> a_value — коэффициент A interval — интервал в миллисекундах T 	ШИМ Выходное значение меняется от 0 к 1 по заданному интервалу. Интервал считается по формуле $A \cdot T$ Интервал задается в миллисекундах
unique	<ul style="list-style-type: none"> input_count — количество входов non_unique_value — значение, для которого игнорируется проверка на уникальность 	Вывод уникальных значений параметров. На выход подаются только те параметры, значения которых уникальны.
unique_inputs	<ul style="list-style-type: none"> input_count — количество входов input.N — входы блока (подаются на выход, если соблюдается условие уникальности значений) default.input.N — значения по умолчанию (подаются на выход, если на входе есть повторяющиеся значения) error_flag = 0 (входные значения уникальны), = 1 (есть повторяющиеся значения) — на выход блока поступают значения по умолчанию 	Блок для контроля повторяющихся значений. Входная группа подаётся на выход, если отвечает критерию уникальности, если нет — на выход подаётся группа по-умолчанию.
delay_value	<ul style="list-style-type: none"> input_count — количество входов/выходов delay_seconds — время задержки в секундах input.N — входное значение output.N — выходное значение 	Блок для вывода значения на выход с указанной задержкой
delay	<ul style="list-style-type: none"> input — вход функции (булевый тип: 0 или 1) delay_sec — интервал задержки в миллисекундах falling — задержка при нисходящем фронте rising — задержка при нарастающем фронте output — выход функции (булевый тип: 0 или 1) 	Блок для вывода значения на выход с указанной задержкой по нарастающему и/или нисходящему фронту
break	<ul style="list-style-type: none"> input_count — число входов/выходов timeout — интервал задержки в миллисекундах input.N — вход функции output.M — выход функции 	Блок задержки изменения значений между входом и выходом. Значение на выходе не меняется чаще, чем один раз в указанное количество миллисекунд. ВНИМАНИЕ! Блок задерживает значение, только если оно меняется чаще чем интервал задержки timeout. В ином случае выход будет меняться без задержки. На момент окончания задержки на выход подаётся последнее значение, поступившее на вход.
state_diagram	<ul style="list-style-type: none"> input — входное значение output — выходное значение rules — правила подстановки (список правил) Пример: 1=>2=3; 2=>3=4; 0=>1=2 default_value — значение на выходе при отсутствии правила для входного значения 	Диаграмма состояний Значение выхода изменяется в соответствии с последовательностью входных значений, описанных правилами. Если правило применить не удалось, выход принимает значение, равное default_value

Имя функции	Количество входных параметров	Описание
hysteresis	<ul style="list-style-type: none"> input — входное значение output — выходное значение input_count — количество интервалов Интервалы: <ul style="list-style-type: none"> N.x1 — начало интервала input >= x1 N.x2 — конец интервала input < x2 N.state — выходное значение при достижении интервала 	Автономный гистерезис (см. описание PID-регулятора). Интервалы описывают только явно определенные состояния. Если при создании блока входное значение находится в неопределённой зоне, то значение выхода устанавливается в 0.
io	<ul style="list-style-type: none"> input_count — количество входов в одной группе group_count — количество групп входов group_index — индекс назначенного входа 	Групповой переключатель входов. Коммутирует одну группу из множества входных групп с одной выходной (M:1).
io_zero	<ul style="list-style-type: none"> input_count — количество входов в одной группе group_count — количество групп входов group_index — индекс назначенного входа 	Групповой переключатель входов. Коммутирует одну группу из множества входных групп с одной выходной (M:1). Индекс группы начинается с 0.
ref_io	<ul style="list-style-type: none"> input_count — количество входов в одной группе group_count — количество групп входов group_index — индекс назначенного входа input.groupN.base_param — имя параметра, по которому осуществляется индексация входной группы 	Ссылочный групповой переключатель входов. Коммутирует одну группу из множества входных групп с одной выходной (M:1).
rio	<ul style="list-style-type: none"> input_count — количество входов в одной группе group_count — количество групп выходов group_index — индекс назначенного выхода 	Обратный групповой переключатель входов. Коммутирует входную группу с одной из множества выходных групп (1:M).
rio_zero	<ul style="list-style-type: none"> input_count — количество входов в одной группе group_count — количество групп выходов group_index — индекс назначенного выхода 	Обратный групповой переключатель входов. Коммутирует входную группу с одной из множества выходных групп (1:M). Индекс группы начинается с 0.
rio	<ul style="list-style-type: none"> input_count — количество входов в одной группе group_count — количество групп выходов group_index — индекс назначенного выхода 	Обратный групповой переключатель входов. Коммутирует входную группу с одной из множества выходных групп (1:M).
setter	<ul style="list-style-type: none"> items_count — количество входов/выходов activate — импульсный вход (выходные параметры задаются значениями входных параметров) input_name.N — имя входного параметра элемента N output_name.N — имя выходного параметра элемента N 	Импульсное задание значения одного параметра или группы параметров
mplex	<ul style="list-style-type: none"> input — входной параметр output_count — количество выходных параметров 	Мультиплексор (дублирование значения со входа блока на указанное количество выходов блока)

Имя функции	Количество входных параметров	Описание
timer	<ul style="list-style-type: none"> · input_count — количество входов в одной группе · time_period — время в секундах · start_when_create — запуск таймера при создании. <p>Значение = 0 — таймер при создании не запустится если activate=1 Значение = 1 — таймер при создании запустится, если activate=1</p> <ul style="list-style-type: none"> · assign_output_when_create — принудительная инициализация выходов блока при создании. <p>Значение = 0 — выходы будут инициализированы значениями, сохранёнными в прикладной программе VSM Значение = 1 — выходы будут инициализированы значениями согласно текущему состоянию таймера</p> <ul style="list-style-type: none"> · activate — активация таймера <p>Значение = 0 — остановка таймера Значение = 1 — запуск таймера</p> <ul style="list-style-type: none"> · debug — отладка <p>Значение = 0 — отладка выключена Значение = 1 — отладка включена</p> <ul style="list-style-type: none"> · debug.end_time — время до срабатывания таймера в секундах (значение = 0, если сработал) · debug.idle_time — время в секундах с момента последнего срабатывания (значение = 0, если таймер ещё работает) 	Таймер
eventbox	<p>Настройки:</p> <ul style="list-style-type: none"> · events_count — количество событий <p>Входные параметры:</p> <ul style="list-style-type: none"> · input.activate — активность события <p>Значение = 0 — Неактивно Значение = 1 — Активно</p> <p>На этот параметр воздействует блок «schedule», этот параметр отображается на выходной параметр activate.</p> <ul style="list-style-type: none"> · input.default_value — значение, отображаемое на выходное value в случае, если событие неактивно (т.е. значение input.activate = 0) · input.value — значение, которое подставляет блок schedule в соответствии со значением, которое ассоциировано с активным событием · input.type_id — текстовый идентификатор, по которому блок schedule находит соответствующее событие среди множества событий eventbox <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output.activate — признак, что событие активно для коммутации с другими блоками <p>Значение = 0 — Неактивно Значение = 1 — Активно</p> <ul style="list-style-type: none"> · output.value — выходное значение, которое либо <p>Значение = input.default_value (если событие неактивно, т.е. значение input.activate = 0) Значение = input.value, которое подставит блок «schedule» (если событие активно, т.е. значение input.activate = 1)</p>	Коммутатор событий от блока «scheduler»

Имя функции	Количество входных параметров	Описание
persist_timer	<ul style="list-style-type: none"> input_count — количество входов в одной группе time_period — время в секундах start_when_create — запуск таймера при создании. Значение = 0 — таймер при создании не запустится если activate=1 Значение = 1 — таймер при создании запустится, если activate=1 assign_output_when_create — принудительная инициализация выходов блока при создании. Значение = 0 — выходы будут инициализированы значениями, сохранёнными в прикладной программе VSM Значение = 1 — выходы будут инициализированы значениями согласно текущему состоянию таймера activate — активация таймера Значение = 0 — остановка таймера Значение = 1 — запуск таймера (продолжение отсчёта) force_finish — Значение = 0 — нормальный режим Значение = 1 — принудительное завершение таймера до его срабатывания reset — сброс Значение = 1 — сброс состояния таймера, отсчёт с начала debug — отладка Значение = 0 — отладка выключена Значение = 1 — отладка включена debug.end_time — время до срабатывания таймера в секундах (значение = 0, если сработал) debug.idle_time — время в секундах с момента последнего срабатывания (значение = 0, если таймер ещё работает) 	Таймер с сохранением времени (обрабатывает оставшееся время после перезапуска сервисов VSM)
trigger	<ul style="list-style-type: none"> input_count — количество входов в одной группе activate — активация Значение = 0 — сброс триггера Значение = 1 — запуск триггера disable — вкл./выкл. Значение = 0 — выключение триггера Значение = 1 — включение триггера 	Триггер
unitrigger	<ul style="list-style-type: none"> mode — режим триггера Значение = 0 — Toggle, Значение = 1 — S-R FF, Значение = 2 — R-S FF set — входное значение reset — сброс Значение = 0 — сброс выключен Значение = 1 — сброса включен output — выход 	Универсальный триггер с тремя режимами
limit	<ul style="list-style-type: none"> input — входное значение функции limits_count — количество limit значений Параметры limit значения: <ul style="list-style-type: none"> limit — пороговое значение threshold — гистерезис high_value — значение при переходе границы limit low_value — значение до перехода границы limit 	Лимит с гистерезисом

Имя функции	Количество входных параметров	Описание
vfix	<p>Настройки:</p> <ul style="list-style-type: none"> fix_by_value — значение, которое инициирует режим фиксации. В этом режиме на выход всегда подается значение параметра fixed_output fixed_output — значение, которое подается на выход в зафиксированном режиме reset — сброс режима фиксации в нормальный режим (импульсный) persistent_reset — постоянный сброс (выключение режима фиксации) <p>Значение = 0 — Выключен Значение = 1 — Включен</p> <ul style="list-style-type: none"> can_reset_force — принудительный сброс <p>Значение = 0 — нормальный режим (сброс возможен только при соответствующих входных условиях) Значение = 1 — сброс, игнорируя входные условия</p> <p>Входные параметры:</p> <ul style="list-style-type: none"> input — вход функции <p>В нормально режиме значение со входа передается на выход</p> <ul style="list-style-type: none"> reset — сброс режима фиксации в нормальный режим <p>Выходные параметры:</p> <ul style="list-style-type: none"> output — выход функции <p>В нормально режиме значение на выход поступает из входа input В фиксированном режиме на выход поступает значение fixed_output</p> <ul style="list-style-type: none"> fix_state — текущий режим <p>Значение = 0 — Нормальный Значение = 1 — Фиксированный</p>	<p>Блок, транслирующий значение входа на выход в нормальном режиме или постоянное значение, равное fixed_output, в случае когда входное значение равняется fix_by_value (функция переходит в режим фиксации, игнорируя входное значение)</p>
graph	<p>Настройки:</p> <ul style="list-style-type: none"> count — число точек графика N.x — X-координата точки, где N — номер точки N.y — Y-координата точки, где N — номер точки xrange.begin — начало диапазона по оси X xrange.end — конец диапазона по оси X yrange.begin — начало диапазона по оси Y yrange.end — конец диапазона по оси Y <p>Входные параметры:</p> <ul style="list-style-type: none"> input — входное значение (X) <p>Выходные параметры:</p> <ul style="list-style-type: none"> output — выходное значение (Y) 	<p>Построение графика: задаются точки, которые соединяются отрезками. Вход — значение по оси X, выход — значение по оси Y</p>
dew_point	<p>Настройки:</p> <ul style="list-style-type: none"> digits_after_point — количество знаков после запятой для выходного расчетного значения <p>Входные параметры:</p> <ul style="list-style-type: none"> input_humidity — влажность в процентах (от 1 до 100) input_temperature — температура в градусах Цельсия (от -30 до 60) <p>Выходные параметры:</p> <ul style="list-style-type: none"> output — расчетное значение точки росы в градусах Цельсия 	<p>Блок для расчета точки росы по температуре и влажности</p>

Имя функции	Количество входных параметров	Описание
days	<p>Настройки:</p> <ul style="list-style-type: none"> enable — включение/выключение функции <p>Значение = 0 — Выключен Значение = 1 — Включен Если enable=0, то значение active=0 автоматически</p> <p>Входные параметры:</p> <ul style="list-style-type: none"> periods — интервалы времени в пределах суток <p>Разделители: «;», «,» Формат hh:[mm[:sec]]-hh:[mm[:sec]]. Пример: 1-2; 2:00-3:00; 3:00:00-4:00:00</p> <ul style="list-style-type: none"> days — список дней месяца через разделители «пробел», «;», «,» <p>День месяца: число от 1 до 31 (если значение = 0 — разрешены все дни). Пример: 1 2 3 5</p> <ul style="list-style-type: none"> wdays — дни недели <p>Список дней недели через разделители «пробел», «;», «,» Допустимы следующие значения (регистр не имеет значения): sun mon tue wed thu fri sat либо all - все дни недели разрешены. Вместо сокращенного имени дня недели можно использовать полное наименование, например: sunday monday</p> <p>Выходные параметры:</p> <ul style="list-style-type: none"> active — состояние планировщика <p>Значение = 0 — Заданный интервал не достигнут Значение = 1 — Заданный интервал наступил</p>	<p>Дневной планировщик.</p> <p>Срабатывает, когда все условия выполняются (диапазоны времени, дни месяца, дни недели). При срабатывании значение параметра active=1.</p> <p>Если установить значение параметра enable=0, то значение параметра active автоматически равно 0.</p>
pump	<p>Настройки:</p> <ul style="list-style-type: none"> disable — принудительное выключение блока <p>Значение = 0 — блок активен Значение = 1 — блок неактивен Если disable=1, то на выходе блока ноль</p> <ul style="list-style-type: none"> disable_inactive — таймер профилактической работы по неактивности <p>Значение = 0 — включен Значение = 1 — выключен</p> <p>Входные параметры:</p> <ul style="list-style-type: none"> condition1 — условие №1 (входные значения: 0 или 1) condition2 — условие №2 (входные значения: 0 или 1) <p>Срабатывание происходит по любому из условий</p> <ul style="list-style-type: none"> time_after_condition — целочисленное время в минутах <p>Количество минут сколько на выходе держится значение=1 при обнулении входных условий</p> <ul style="list-style-type: none"> time_after_inactive — целочисленное время в минутах <p>Количество минут сколько на выходе держится значение=1 во время профилактической работы</p> <ul style="list-style-type: none"> time_inactive — число с дробной частью в часах. Минимальное значение 0.02. <p>Запуск профилактической работы по неактивности через указанное количество часов. Продолжительность этой работы уточняется параметром time_after_inactive</p> <p>Выходные параметры:</p> <ul style="list-style-type: none"> output — выход (0 или 1) 	<p>Блок управления насосом</p>

Имя функции	Количество входных параметров	Описание
bits_pack	Настройки: · input_count — число входов Каждый вход — значение бита, начиная с младшего Входные параметры: · input.N — входной массив битовой маски Выходные параметры: · output — выходное значение, собранное из массива битовой маски	Сборка выходного значения по битовой маске
bits_unpack	Настройки: · output_count — число выходов. Каждый выход — значение бита, начиная с младшего Входные параметры: · input — входное значение Выходные параметры: · output.K — выходное значение битовой маски	Разложение входного значения в битовую маску
split_symbols	Настройки: · output_count — количество выходов Каждый выход ассоциируется с подстрокой в заданных пределах · output.N.start — стартовый символ строки, нумерация начинается с 1 · output.N.length — длина подстроки, начиная со стартового символа Входные параметры: · input — входная текстовая строка Выходные параметры: · output.N.value — выходная подстрока, индивидуальная для каждого выхода, где N — номер выхода	Выделение подстроки или символов из входного значения
split	Настройки: · input_count — число входов Каждый вход — это массив значений · Separators — разделители, по которым выделяются значения из массива По умолчанию: ' '; Входные параметры: · input.N — входной массив значений Выходные параметры: · output.K — выделенное значение из входного массива, где K — число образовавшихся после разложения массива элементов	Разложение массива на параметры по указанному разделителю
impulse_counter	Входные параметры: · input — входные значения Настройки: · enable — вкл./выкл. счётчика · reset — сброс счётчика Значение = 0 — норм. состояние Значение = 1 — сброс счётчика · direction — направление счётчика Значение=0 — Уменьшение счётчика на 1 Значение=1 — Увеличение счётчика на 1 · persistent — сохранение значения счётчика между перезапусками Значение = 0 — Значение не сохраняется Значение = 1 — Значение сохраняется Выходные параметры: · output — счетчик входных импульсов	Сбрасываемый счётчик импульсов (изменений параметра)

Имя функции	Количество входных параметров	Описание
impulse_maker	Настройки: <ul style="list-style-type: none"> enable — вкл./выкл. блока Значение = 0 — Выключить работу Значение = 1 — Включить работу <ul style="list-style-type: none"> impulse_count — количество генерируемых импульсов (от 0 до 4294967295) delay_between_impulses — пауза между импульсами в секундах Выходные параметры: <ul style="list-style-type: none"> output — выход генерируемых импульсов (каждый импульс это последовательность значений 0-1-0) 	Генератор заданного количества импульсов (0-1-0)
icounter		Блок полностью аналогичен блоку impulse_counter, но входное значение=0 не приводит к увеличению счётчика
time_counter	Входные параметры: <ul style="list-style-type: none"> input — входные значения Настройки: <ul style="list-style-type: none"> enable — вкл/выкл. счётчика. Если выкл. — счёт идёт, но выход счётчика не меняется до момента выставления enable в 1 <ul style="list-style-type: none"> reset — сброс счётчика Значение = 0 — норм. состояние Значение = 1 — сброс счётчика <ul style="list-style-type: none"> direction — направление счётчика Значение=0 — Уменьшение счётчика на 1 Значение=1 — Увеличение счётчика на 1 <ul style="list-style-type: none"> persistent — сохранение значения счётчика между перезапусками Значение = 0 — Значение не сохраняется Значение = 1 — Значение сохраняется Выходные параметры: <ul style="list-style-type: none"> output — счётчик фронта в секундах 	Счетчик фронта временной. Считает секунды, когда на входе держится единица. Если enable=0, то счёт продолжается внутри блока, а выходной счётчик останавливается
subst	Настройки: <ul style="list-style-type: none"> conditions_count — количество значений в таблице подстановки use_default_value — флаг для использования значения по умолчанию Значение = 0 — Не использовать. На выход блока подается входное значение, если не найдено соответствий Значение = 1 — Использовать. На выход блока подается значение по умолчанию, если не найдено совпадений <ul style="list-style-type: none"> default_value — значение по умолчанию, которое подается на выход в случае отсутствия совпадений Применимо если use_default_value=1 Входные параметры: <ul style="list-style-type: none"> input — входное значение condition.N.input_value — значение, которое требуется заменить, где N - индекс в таблице подстановки condition.N.subst_value — значение подстановки Выходные параметры: <ul style="list-style-type: none"> output — выход функции Если входное значение отсутствует в таблице подстановки, то на выход подаётся: а) значение по умолчанию, use_default_value = 1 б) Значение со входа блока, если use_default_value = 0	Блок для табличной подстановки значений

Имя функции	Количество входных параметров	Описание
commutate	<p>Настройки:</p> <ul style="list-style-type: none"> · items_count — количество входов/выходов в блоке · fill_uncommutated — заполнение нескоммутированных выходов <p>Значение = 0 — Выключено (по умолчанию) Значение = 1 — Включено. Заполнять нескоммутированные выходы значением fill_value</p> <ul style="list-style-type: none"> · fill_value — значение для заполнения нескоммутированных выходов. <p>Используется, если параметр fill_uncommutated=1</p> <p>Входные параметры:</p> <ul style="list-style-type: none"> · input.N.value — входное значение · input.N.index — номер выхода для коммутации (на этот выход подается значение) <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output.N — выход, где N - номер выхода. <p>Значение на выходе определяется настройками входных параметров</p> 	Блок для коммутации входящих значений с выходными по заданному индексу
temp_sensor	<p>Настройки:</p> <ul style="list-style-type: none"> · sensor_name — 1-wire идентификатор температурного датчика <p>Выходные параметры:</p> <ul style="list-style-type: none"> · value — значение с датчика · alarm — флаг аварии / отсутствия связи · port — номер порта, к которому подключен датчик 	Блок для получения показаний температурных датчиков
pt1	<p>Входные параметры:</p> <ul style="list-style-type: none"> · input — входное значение <p>Настройки:</p> <ul style="list-style-type: none"> · time_constant — временная константа · digits_after_point — число цифр после запятой на выходе · pass_through — флаг для пропуска значения со входа на выход без изменений (фактически, отключает работу блока) <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output — выходное значение 	Сглаживание входного сигнала. Рассчитывается по формуле: $Y = X * (1 - \exp(-step/T_i))$, где step — шаг в секундах, T_i — временная константа
trend	<p>Входные параметры:</p> <ul style="list-style-type: none"> · input_value — входное значение <p>Настройки:</p> <ul style="list-style-type: none"> · measure_count — количество измерений для подсчета тренда · digits_after_point — число знаков после запятой на выходе <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output.state — текущее состояние (число): <p>Значение = 0 — Убывание Значение = 1 — Покой Значение = 2 — Возрастание</p> <ul style="list-style-type: none"> · output.state.text — текущее состояние (текст): <p>Значение = down Значение = stable Значение = up</p> <ul style="list-style-type: none"> · output.state.speed_in_min — скорость изменения величины в минуту 	Блок «Тренд» для расчёта изменения величины за выбранное количество измерений. Блок считает среднее по заданному количеству измерений: рассчитывает величину, направление и скорость изменения величины

Имя функции	Количество входных параметров	Описание
ramp	<p>Входные параметры:</p> <ul style="list-style-type: none"> · direction — направление движения Значение = 0 — от точки В к точке А Значение = 1 — от точки А к точке В · pause — пауза Значение = 0 — Нормальный режим Значение = 1 — Остановка вычислений · reset — сброс до начальной точки (в зависимости от направления) <p>Настройки:</p> <ul style="list-style-type: none"> · point_a — значение точки А · point_b — значение точки В · period_up — время в секундах для направления от точки А к точке В · period_down — время в секундах для направления от точки В к точке А · digits_after_point — число знаков после запятой на выходе <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output.output_value — выходное значение · output.a_reachhead — точка А достигнута Значение = 0 — Нет Значение = 1 — Да · output.b_reachhead — точка В достигнута Значение = 0 — Нет Значение = 1 — Да 	Изменение значений от точки А к точке В за заданное время
step_regulator	<p>Настройки:</p> <ul style="list-style-type: none"> · steps_count — количество шагов регулятора (максимум 100) · begin_range — начало рабочего диапазона · end_range — конец рабочего диапазона · up_delay_seconds — задержка в секунда на включение шага · down_delay_seconds — задержка в секунда на отключение шага <p>Входные параметры:</p> <ul style="list-style-type: none"> · input — входное значение с ПИД-регулятора · force_off Значение = 0 — Нормальный режим Значение = 1 — Аварийное отключение, выходы шагов выставляются в 0 <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output.N.state — выход N-шага (0 или 1) 	Шаговый регулятор с настраиваемым числом шагов
max_sequence	<p>Входные параметры:</p> <ul style="list-style-type: none"> · input_value — входное значение <p>Настройки:</p> <ul style="list-style-type: none"> · digits_after_point — число знаков после запятой · enable — выключение/сброс <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output_max — максимум 	Расчёт максимума в последовательности значений
min_sequence	<p>Входные параметры:</p> <ul style="list-style-type: none"> · input_value — входное значение <p>Настройки:</p> <ul style="list-style-type: none"> · digits_after_point — число знаков после запятой · enable — выключение/сброс <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output_min — минимум 	Расчёт минимума в последовательности значений

Имя функции	Количество входных параметров	Описание
math_ sequence_func	Входные параметры: · input_value — входное значение Настройки: · digits_after_point — число знаков после запятой · enable — выключение/сброс Выходные параметры: · output_max — максимум · output_min — минимум · output_avg — арифметическое среднее	Расчёт минимума, максимума и арифметического среднего в последовательности значений
arithmetic_ average	Входные параметры: · input — входное значение Настройки: · measure_count — число измерений для подсчёта среднего Выходные параметры: · output — выходное значение	Расчёт арифметического среднего
device_delay_ defence	Входные параметры: · device.N.input — сигнал управления на устройство N Настройки: · items_count — количество контролируемых устройств (макс. 100) · c1_work_time — минимальное время работы устройства · c2_stop_time — минимальное время нахождения в стопе · c3_next_start_time — минимальное время до следующего запуска устройства · c4_start_delay — задержка старта следующего устройства · c5_stop_delay — задержка стопа следующего устройства · assign_outputs Значение = 0 — Выходы не меняются до фактического воздействия Значение = 1 — Обнуление выходов устройств при инициализации блока Выходные параметры: · device.N.output — выходной сигнал управления на устройство N	Запуск и останов группы устройств с заданными таймингами

Имя функции	Количество входных параметров	Описание
working_hours	<p>Входные параметры:</p> <ul style="list-style-type: none"> • enable — включение/отключение счетчика • reset — сброс счётчика <p>Настройки:</p> <ul style="list-style-type: none"> • refresh_time_sec — время обновления счетчика <p>Выходные параметры:</p> <p>Общее время работы:</p> <ul style="list-style-type: none"> • total.hours — часы наработки • total.mins — минуты наработки • total.formatted — комбинированное значение <p>Время работы за текущие сутки:</p> <ul style="list-style-type: none"> • day.hours — часы наработки • day.mins — минуты наработки • day.formatted — комбинированное значение <p>Время работы за предыдущие сутки:</p> <ul style="list-style-type: none"> • last_day.hours — часы наработки • last_day.mins — минуты наработки • last_day.formatted — комбинированное значение <p>Время работы за текущую неделю:</p> <ul style="list-style-type: none"> • week.hours — часы наработки • week.mins — минуты наработки • week.formatted — комбинированное значение <p>Время работы за прошлую неделю:</p> <ul style="list-style-type: none"> • last_week.hours — часы наработки • last_week.mins — минуты наработки • last_week.formatted — комбинированное значение <p>Время работы за текущий месяц:</p> <ul style="list-style-type: none"> • month.hours — часы наработки • month.mins — минуты наработки • month.formatted — комбинированное значение <p>Время работы за предыдущий месяц:</p> <ul style="list-style-type: none"> • last_month.hours — часы наработки • last_month.mins — минуты наработки • last_month.formatted — комбинированное значение <p>Время работы за текущий год:</p> <ul style="list-style-type: none"> • year.hours — часы наработки • year.mins — минуты наработки • year.formatted — комбинированное значение <p>Время работы за прошлый год:</p> <ul style="list-style-type: none"> • last_year.hours — часы наработки • last_year.mins — минуты наработки • last_year.formatted — комбинированное значение 	<p>Блок для подсчёта наработки в часах с сохранением значений между перезапусками и статистикой по предыдущим периодам</p>

Имя функции	Количество входных параметров	Описание
power_meter	<p>Входные параметры:</p> <ul style="list-style-type: none"> · input — входное значение с измерением текущей мощности <p>Настройки:</p> <ul style="list-style-type: none"> · items_count — количество тарифных зон (максимум 100) <p>Необходима как минимум одна зона для функционирования блока</p> <ul style="list-style-type: none"> · counter_offset — добавочное значение (shift) для выхода счетчика · cost_offset — добавочное значение цены (shift) к итоговой цене · digits_after_point — количество знаков после запятой для выходов счетчика <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output.total_cost — итоговое значение цены · output.total_count — итоговое значение счётчика <p>Настройки тарифной зоны:</p> <ul style="list-style-type: none"> · zone.N.price — цена за единицу · zone.N.range_begin — начальный интервал тарифной зоны в формате HH:MM:SS Например 23:59:00 Минуты и секунды можно не указывать, тогда они равны 0 · zone.N.range_end — конечный интервал тарифной зоны в формате HH:MM:SS <p>Выходы тарифной зоны:</p> <ul style="list-style-type: none"> · zone.N.output.counter — текущий счетчик тарифной зоны · zone.N.output.counter — текущая сумма оплаты по тарифной зоне 	Многотарифный счетчик электроэнергии

Имя функции	Количество входных параметров	Описание
math_expr	<p>Настройки:</p> <ul style="list-style-type: none"> · digits_after_point — количество знаков после запятой в выходном значении · input_names — список входных переменных Например: x1, y1, temp · expression — формула расчёта значений Например: sin(x1)+sqrt(y1)+temp · expression — синтаксис выражений, поддерживаемые операции <p>• "+" — стандартные математические операции • "-" — стандартные математические операции • "*" — стандартные математические операции • "/" — стандартные математические операции • "^" — возведение в степень • "(" — группировка выражений</p> <p>Поддерживаемые функции:</p> <ul style="list-style-type: none"> · pow(x,y) · sqrt(x) · log(x) · exp(x) · sin(x) · asin(x) · cos(x) · acos(x) · tan(x) · atan(x) · atan2(y,x) · abs(x) · fabs(x) · floor(x) · ceil(x) · round(x) <p>Входные параметры:</p> <ul style="list-style-type: none"> · input.NAME, где NAME — имя, перечисленное в input_names <p>Выходные параметры:</p> <ul style="list-style-type: none"> · output — выходное значение 	Расчёт значений по заданной формуле

7.2 Блок ПИД-регулирования

Настройки блока ПИД-регулирования осуществляются в группе параметров `soft.settings.pids.*`

За инстанцирование экземпляра ПИД-регулятора отвечает параметр `soft.settings.pids.active_pids` (активные экземпляры PID-регулятора) — список меток, разделяемых запятой или пробелом. Метка — имя экземпляра.

Экземпляр ПИД-регулятора

Метка, указанная в списке активных экземпляров ПИД-регулятора, создаёт группу параметров в `soft.settings.pids.profiles.NAME.*`, где NAME — имя метки. Добавление метки создает экземпляр ПИД-регулятора с указанным именем.

Настройки экземпляра ПИД-регулятора представлены в таблице 2. Имена параметров указаны без базового пути.

Таблица 2. Настройки экземпляра ПИД-регулятора

Имя параметра	Назначение
active_sequences	Активные последовательности управления. Список меток указывается через разделители «пробел» или «,»
active_hysteresises	Список активных гистерезисов. Список меток указывается через разделители «пробел» или «,»
active_controls	Список групп управления последовательностями. Список меток указывается через разделители «пробел» или «,»
auto_reset	Автоматический сброс ошибки ПИД-регулятора при изменении коэффициентов. Актуально, когда включена интегральная составляющая (0 — выключено, 1 — включено). Используется для отладки, по умолчанию выключен.
reset	Сброс текущей накапливаемой ошибки ПИД-регулятора. Актуально, когда включена интегральная составляющая
input	Входная величина, на основе которой рассчитывается ошибка
input_shift	Значение (сдвиг), которое суммируется со входной величиной перед расчетом ошибки
output	Выходная величина — ошибка
kp	Пропорциональная составляющая (по умолчанию = 1)
ki	Интегральная составляющая (по умолчанию = 0)
kd	Дифференциальная составляющая (по умолчанию = 0)
output_range	Диапазон выходного значения (ошибки) (по умолчанию от -0.5 до 0.5)
output_shift	Сдвиг выходного значения (ошибки) (по умолчанию 0.5)
point	Заданное значение регулятора (уставка)
point_shift	Сдвиг заданного значения регулятора, суммируется с заданным значением перед расчетом
period	Интервал вычисления ошибки в секундах
proportional_range	Зона пропорциональности (ЗП) — это число, характеризующее входной диапазон, который представляется как интервал от $(point - ЗП/2)$ до $(point + ЗП/2)$. Влияет на чувствительность ПИД-регулятора, т.е. на степень его реакции
slope	«Наклон» ошибки характеризует её направление (рост или уменьшение) в зависимости от входной величины. 0 = «\» — обратная косая черта, 1 = «/» — косая черта

Подчинённые блоки ПИД-регулятора

ПИД-регулятор имеет следующие подчинённые блоки (см. рисунок 42):

- Гистерезисы;
- Последовательности;
- Группы управления последовательностями.

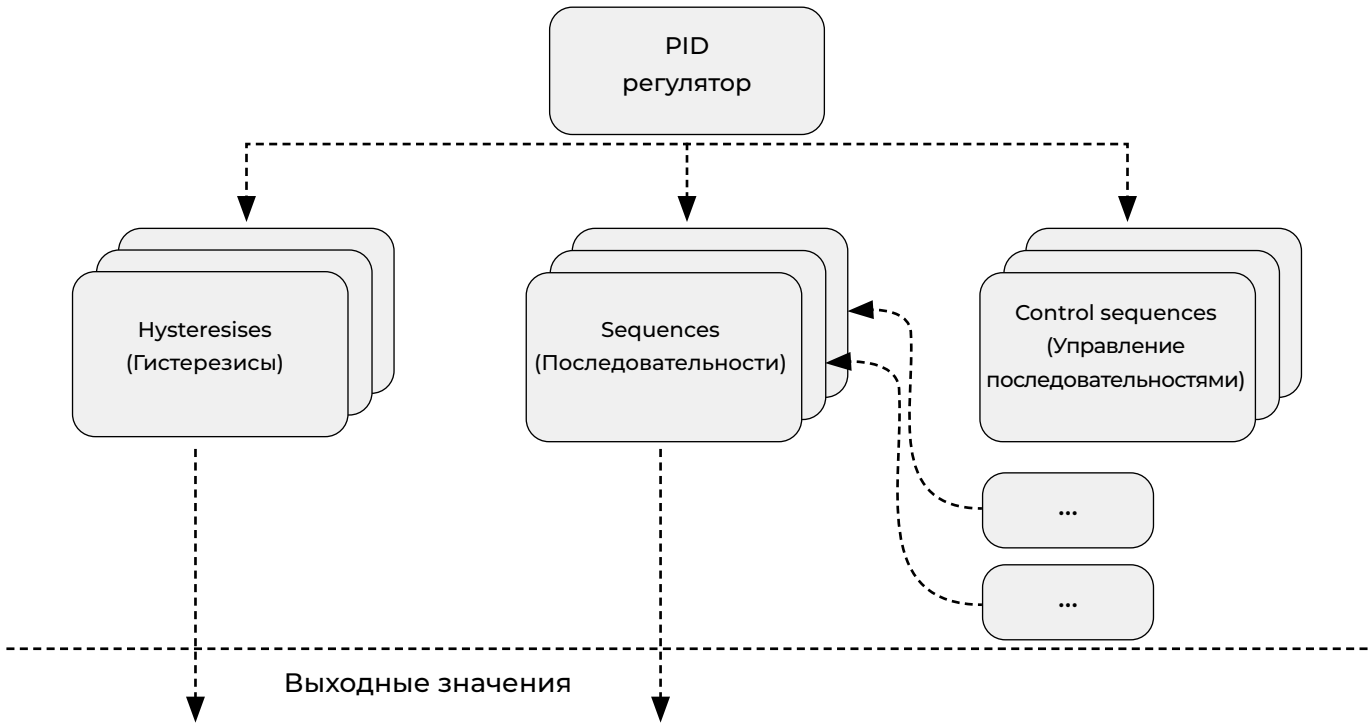


Рисунок 42. Подчинённые блоки ПИД-регулятора

ПИД-регулятор: последовательности (sequences)

Одна или множество последовательностей регулирования включаются в экземпляр ПИД-регулятора.

Последовательности перечисляются в параметре `active_sequences`.

Базовый путь последовательности:

`soft.settings.pids.profiles.PID_NAME.sequences.SEQ_NAME.*`, где `PID_NAME` — имя ПИД-регулятора, `SEQ_NAME` — имя последовательности.

Основная характеристика последовательности — график прямой, который задается двумя точками, а также ограничивается диапазоном. Входное значение — ошибка ПИД-регулятора, выходное значение — результат соотнесения ошибки с графиком.

Настройки последовательности представлены в таблице 3.

Таблица 3. Настройки последовательности

Имя параметра	Назначение
output	Выходное значение последовательности
output_min	Выходной диапазон: минимум (в процентах от 0 до 100)
output_max	Выходной диапазон: максимум (в процентах от 0 до 100)
point_a.x	Точка А графика, координата x
point_a.y	Точка А графика, координата y
point_b.x	Точка В графика, координата x
point_b.y	Точка В графика, координата y
inverse_slope	Зеркальное отображение наклона графика (от заданного) Значение = 0 — выключено, значение = 1 — включено

ПИД-регулятор: управление группой последовательностей (control sequences)

Одна или множество групп управления последовательностями включаются в ПИД-регулятор. Группы управления последовательностями перечисляются в параметре **active_controls** ПИД-регулятора.

Базовый путь группы управления: **soft.settings.pids.profiles.PID_NAME.controls.CONTROL_NAME.***, где **PID_NAME** — имя PID регулятора, **CONTROL_NAME** — имя группы управления последовательностями.

Группа управления последовательностями состоит из списка существующих последовательностей (sequences), которые указаны в параметре **active_sequences**. Для этих последовательностей указывается зона охвата в процентах и наклон графика. Порядок следования (слева направо) в соответствии с номером элемента в группе последовательности.

Группа управления последовательностями меняет точки задействованных последовательностей таким образом, чтобы разместить их в заданном порядке, в заданных пропорциях, в заданном диапазоне на оси графика.

В список управления последовательностями может включаться только та последовательность, которая существует (перечислена) в **active_sequences**.

Настройки группы управления последовательностями представлены в таблице 4.

Таблица 4. Настройки группы управления последовательностями

Имя параметра	Назначение
count	Количество элементов (последовательностей для управления)
range_start	Диапазон размещения на графике: начальная точка
range_end	Диапазон размещения на графике: конечная точка
replace_right	Политика замещения выключенных последовательностей. Значение = 0 — замена соседом слева, значение = 1 — замена соседом справа
error	Флаг ошибки. Значение = 0 — нет ошибки, значение = 1 — ошибка
error_message	Сообщение об ошибке

ПИД-регулятор: элемент группы управления последовательностями

После указания количества элементов (последовательностей для управления) в группе управления последовательностями создаются соответствующие группы параметров внутри группы управления последовательностями. Порядковый номер элемента соответствует порядку следования элемента слева направо.

Настройки элемента группы управления последовательностями представлены в таблице 5.

Таблица 5. Настройки элемента группы управления последовательностями

Имя параметра	Назначение
sequence_name	Имя последовательности для управления (должна быть в списке active_sequences ПИД-регулятора)
slope	Наклон графика: 0 = «/» — косая черта, 1 = «\» — обратная косая черта
width	Ширина графика в процентах

ПИД-регулятор: гистерезисы (hysteresises)

Один или множество гистерезисов включаются в экземпляр ПИД-регулятора. Гистерезисы перечисляются в параметре **active_hysteresises** (параметр последовательности регулирования).

Базовый путь гистерезиса:

soft.settings.pids.profiles.PID_NAME.hysteresises.HYST_NAME.*, где **PID_NAME** — имя ПИД-регулятора, **HYST_NAME** — имя гистерезиса.

Гистерезис описывает интервалы состояний, которые однозначно детерминированы.

Если значение не попадает ни в один из указанных интервалов, то выходное значение гистерезиса сохраняет своё последнее значение (не меняется).

Настройки гистерезиса представлены в таблице 6.

Таблица 6. Настройки гистерезиса

Имя параметра	Назначение
count	Количество интервалов
output	Выходное значение гистерезиса

PID: Интервалы гистерезиса

Гистерезис может содержать один или множество интервалов, описывающих состояние. Указываются интервалы тех диапазонов, которые явно определяют выходное значение. Если входное значение не попадает ни в один из заданных диапазонов, выходное значение сохраняет свое состояние (не меняется).

Настройки интервалов гистерезиса представлены в таблице 7.

Таблица 7. Настройки интервалов гистерезиса

Имя параметра	Назначение
x1	Начало интервала (точка $\geq x1$)
x2	Конечное значение интервала (точка $< x2$)
state	Значение состояния (для выхода гистерезиса)

7.3 Блоки сохранения данных

Блоки сохранения данных указываются в списке активных блоков сохранения в параметре

`soft.settings.store.active_store`.

После добавления в список блок сохранения создается по указанному пути:

`soft.settings.store.instances.STORE_NAME`, где `STORE_NAME` — имя блока сохранения.

Примечание: блок сохранения может работать как блок загрузки данных, если значение параметра `read_only=1`.

После создания блока необходимо указать его тип, задав параметр `type`:

- тип `self` — сохраняет значения параметров на собственных входах. Количество входов определяется параметром `items_count`;
- тип `by_name` — сохраняет значения произвольных параметров. Список параметров для сохранения задается параметром `names`;
- тип `by_name_ext` — аналогичен типу `by_name` с возможностью присвоить параметру значение по умолчанию при его отсутствии в прикладной программе.

Блок сохранения данных типа `self`

Параметры блока сохранения данных типа `self` представлены в таблице 8.

Таблица 8. Параметры блока сохранения данных типа `self`

Имя параметра	Назначение
filename	Имя файла для сохранения параметров
savetime	Время в секундах, ограничивающее обращение к диску на запись. Данные сохраняются по факту изменения, но не чаще раз в N секунд.
count	Количество входов для сохранения
read_only	Данные только для чтения: данные загружаются из файла, но никогда не сохраняются

Блок сохранения данных типа `by_name`

Параметры блока сохранения данных типа `by_name` представлены в таблице 9.

Таблица 9. Параметры блока сохранения данных типа `by_name`

Имя параметра	Назначение
<code>filename</code>	Имя файла для сохранения параметров
<code>savetime</code>	Время в секундах, ограничивающее обращение к диску на запись. Данные сохраняются по факту изменения, но не чаще раз в N секунд.
<code>names</code>	Имена параметров для сохранения. Указываются через разделители «пробел», «;», «,»
<code>read_only</code>	Данные только для чтения: данные загружаются из файла, но никогда не сохраняются

Блок сохранения данных типа `by_name_ext`

Параметры блока сохранения данных типа `by_name_ext` представлены в таблице 10.

Таблица 10. Параметры блока сохранения данных типа `by_name_ext`

Имя параметра	Назначение
<code>filename</code>	Имя файла для сохранения параметров
<code>savetime</code>	Время в секундах, ограничивающее обращение к диску на запись. Данные сохраняются по факту изменения, но не чаще раз в N секунд.
<code>read_only</code>	Данные только для чтения: данные загружаются из файла, но никогда не сохраняются
<code>items_count</code>	Количество элементов лога
<code>items.N.name</code>	Элемент: имя параметра (основное) для сохранения, где N - номер элемента
<code>items.N.init_by</code>	Элемент: имя параметра (вспомогательное), которым следует инициализировать основной параметр, если он не был задан
<code>items.N.default_value</code>	Элемент: значение по умолчанию для инициализации параметра, если основной и вспомогательный не заданы

Хранение файлов и контрольная сумма

При сохранении файла блока `store` система автоматически создаёт два файла:

- файл с именем, указанным в параметре `filename`
- теньевая копия файла с постфиксом `.shadow`

При выключении питания в момент открытия файла или его перезаписи данные файла могут быть утеряны. Для предотвращения этой ситуации запись дублируется в два файла. Если один файл был испорчен в момент отключения питания, второй остаётся целым.

Для контроля целостности данных в конец файлов блоков `store` добавляется контрольная сумма. Строка с контрольной суммой находится в конце файла и выглядит следующим образом: `crc16-signed:45702`

Файл автоматически подписывается при записи данных блока со стороны прикладного уровня. В случае, если корректировка файла блока `store` была выполнена вручную необходимо подписывать файл контрольной суммой каждый раз после его изменения.

- Подписать файл хранения контрольной суммой: `./logic_service --sign filename.txt`
- Проверить подпись файла хранения: `./logic_service --check filename.txt`

7.4 Блоки журналирования и контроля аварий (логгер)

Блоки журналирования и контроля аварий указываются в списке активных блоков журналирования в параметре `soft.settings.logger.active_loggers`. После добавления в список блок журналирования создается по указанному пути:

`soft.settings.logger.instances.LOGGER_NAME`, где `LOGGER_NAME` — имя блока журналирования.

После создания блока необходимо указать имя файла (журнала), а также количество элементов, записываемых в журнал. Каждый элемент связывается с отдельным параметром.

Блок журналирования может работать в двух режимах:

1. Простой (значение параметра `alarm_functions` = 0)

Блок работает как журнал без дополнительных функций обработки аварий. Простой режим задается блоку по умолчанию.

2. Расширенный (значение параметра `alarm_functions` = 1)

Включена дополнительная функция «контроль аварий», а также появляются дополнительные параметры.

Общий параметр для блоков журналирования и контроля аварий

Параметр `soft.settings.logger.current_mode` — текущий режим работы является общим для блоков журналирования и контроля аварий. Значение параметра влияет на работу логгера. Осуществляется запись и контроль тех элементов, которые связаны с текущим режимом работы оборудования.

Блок журналирования и контроля аварий

Параметры блока журналирования и контроля аварий представлены в таблице 11.

Таблица 11. Параметры блока журналирования и контроля аварий

Имя параметра	Назначение
<code>file_name</code>	Имя файла журнала
<code>items_count</code>	Количество элементов или журналируемых параметров
<code>alarm_functions</code>	Поддержка расширенных функций контроля и регистрации аварий Значение = 0 — выключено, значение = 1 — включено
<code>reset_alarms</code>	Сброс активных тревог внутри контролируемой группы параметров

Для каждого класса аварий имеется своя группа выходов, которые представлены в таблице 12.

Таблица 12. Выходы класса аварий

Имя параметра	Назначение
<code>output.N.alarm</code>	Общий флаг аварии, где N — класс аварии. Значение = 0 — авария отсутствует, значение = 1 — авария присутствует. Флаг выставляется, если среди контролируемой группы параметров зафиксирована авария
<code>output.N.alarm_group</code>	Группа зафиксированной аварии, где N — класс аварии
<code>output.N.alarm_identifier</code>	Идентификатор параметра, ассоциированного с аварией

Параметры элементов блока журналирования и контроля аварий

Параметры элементов блока журналирования и контроля аварий представлены в таблице 13.

Таблица 13. Параметры элементов блока журналирования и контроля аварий

Имя параметра	Назначение
identifier	Уникальный текстовый идентификатор элемента ВНИМАНИЕ! Правило задания идентификатора: идентификатор — текстовая метка без пробелов с использованием только латинских символов. Например: pv_pressure_out_of_range
param_name	Имя параметра, значение которого требуется журналировать
value	Значение параметра ВНИМАНИЕ! Используется в случае, если не указано имя параметра, а значение требуется получать посредством правил перенаправления параметров
policy	Политика журналирования: <ul style="list-style-type: none"> · change — по изменению · equal — по сравнению с эталонным значением · range — по диапазону · timer — по таймеру · chtimer — по изменению + по таймеру (change + timer)
operate_modes	Список режимов работы оборудования, при которых данный элемент активирован. Допустимые разделители: «пробел», «,», «;». Если значение=0 или не задано (пустое поле) — элемент активирован во всех режимах работы

Дополнительные параметры, если значение параметра policy = equal, представлены в таблице 14.

Таблица 14. Параметры политики «equal»

Имя параметра	Назначение
normal_value	Эталонное значение Если значение не равно эталонному, то через delay секунд возникает авария
delay	Время задержки срабатывания аварии в секундах

Дополнительные параметры, если значение параметра policy = range, представлены в таблице 15.

Таблица 15. Параметры политики «range»

Имя параметра	Назначение
min	Минимальное значение параметра
max	Максимальное значение параметра
delay	Время задержки срабатывания аварии в секундах
invert_rule	Инвертирование правила, т.е. проверка что не входит в диапазон

Дополнительные параметры, если значение параметра policy = timer, представлены в таблице 16.

Таблица 16. Параметры политики «timer»

Имя параметра	Назначение
timer	Интервал времени в секундах, через которые происходит сохранение параметра в лог

Дополнительные параметры, если значение параметра policy = change, представлены в таблице 17.

Таблица 17. Параметры политики «change»

Имя параметра	Назначение
delay	Время задержки срабатывания аварии в секундах
invert_rule	Инвертирование правила: авария, если значение параметра НЕ изменяется более чем delay секунд
	<p>ВНИМАНИЕ! Политика «change» в режиме тревог имеет практический смысл только в одном режиме:</p> <ul style="list-style-type: none"> · delay == 0, invert_rule == 0, авария всегда возведена · delay == N, invert_rule == 0, авария всегда возведена, через N секунд · delay == 0, invert_rule == 1, авария всегда возведена · delay == N, invert_rule == 1, авария возводится, если N секунд параметр не меняет значения. Это единственный режим имеющий практическое значение в режиме тревог для политики «change»

Параметры элемента, доступные при включенной функции контроля аварий (alarm_functions=1) представлены в таблице 18.

Таблица 18. Параметры элемента при включенной функции контроля аварий

Имя параметра	Назначение
alarm	Флаг аварии: значение = 0 — авария отсутствует, значение = 1 — авария присутствует. Выставляется автоматически, если контролируемый параметр вышел за указанный диапазон значений в соответствии с политикой проверки. Появление ошибки вызывает общий флаг аварии в блоке журнала
alarm_class	Класс аварии: Значение = 0 — информационное сообщение Значение = 1 — категория А Значение = 2 — категория В
alarm_group	Группа аварии
alarm_type	Тип аварии в пределах группы
alarm_auto_reset	Автоматический сброс аварии, если значение параметра вернулось в область допустимых значений и текущий operate_mode совпадает с выбранным. alarm_auto_reset и alarm_can_reset <u>НЕ ДОЛЖНЫ</u> быть включены (или выключены) одновременно
alarm_can_reset	Ручной сброс аварии
alarm_force_reset	При ручном сбросе тревога (безусловно) сбросится несмотря на текущий режим и допустимость входного значения

Техническая поддержка программного обеспечения осуществляется программистами компании ООО «РМ Вент» по адресу:

197198, Санкт-Петербург г., Ждановская ул., дом 29, литера А, ПОМЕЩ. 15-Н ПОМЕЩ. 20.

Контактный телефон: +7 (812) 321-61-21, электронная почта: info@pmvent.ru



Компания ООО «РМ Вент» является разработчиком программного обеспечения, систем автоматизации и диспетчеризации, а также производителем и поставщиком инженерного оборудования.

197198, Санкт-Петербург, ул. Ждановская, д. 29
8 (800) 550-61-21 (звонок по России бесплатный)
info@pmvent.ru
pmvent.ru