

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «РЕЛЕМАТИКА МД»

Руководство системного программиста

АИПБ.09.04.001-3.0 32

Авторские права на данный документ принадлежат ООО «Релематика-НКУ», 2023.
Данный документ не может быть полностью или частично воспроизведен, скопирован,
распространен без разрешения ООО «Релематика-НКУ».

Адрес предприятия-изготовителя:

428020, Чувашская Республика, г. Чебоксары, пр. И. Яковлева, д. 1, пом.13, каб. 80
ООО «Релематика-НКУ»
Тел.: +7(8352) 244-777
E-mail: info@relematika-nku.ru

Аннотация

Данный документ содержит описание применения и эксплуатации программного обеспечения «Релематика МД» системным программистом или иным сотрудником, в должностные обязанности которого входят данные виды работ. Данная программа предназначена для работы в системах автоматизации в качестве ПО для сбора, обработки и передачи данных как компонент ПТК «UniSCADA», так и самостоятельно.

Содержание

| | | |
|--------|---|----|
| 1 | Общие сведения о программе | 6 |
| 1.1 | Назначение и функции | 6 |
| 1.2 | Сведения о технических и программных средствах, обеспечивающих выполнение данной программы..... | 6 |
| 2 | Структура программы | 6 |
| 3 | Установка и настройка программы | 7 |
| 3.1 | Установка программы | 7 |
| 3.1.1 | Установка на ОС «Windows» | 7 |
| 3.1.2 | Установка на ОС «Linux» | 8 |
| 3.2 | Параметры командной строки | 18 |
| 3.3 | Файл «config.xml» | 18 |
| 3.4 | Файл проекта | 19 |
| 3.5 | Базовый объект | 20 |
| 3.6 | Объект «OPCUA_SERVER» | 21 |
| 3.7 | Объект «OPCUA_CLIENT» | 21 |
| 3.8 | Объект «IEC104_SERVER» | 22 |
| 3.9 | Объект «IEC104_CLIENT» | 24 |
| 3.10 | Объект «IEC61850_SERVER» | 25 |
| 3.11 | Объект «IEC61850_CLIENT» | 26 |
| 3.11.1 | Параметры подключения устройства | 26 |
| 3.11.2 | Параметры отчетов..... | 27 |
| 3.11.3 | Параметры команд | 27 |
| 3.11.4 | Параметры считывания файлов | 28 |
| 3.12 | Объект «SNMPv3_OBJ» | 29 |
| 3.13 | Объект «DirControl» | 31 |
| 3.14 | Объект формирования единой осциллограммы «R58601»..... | 32 |
| 3.15 | Объект «EventLog» | 34 |
| 3.16 | Объект «Mux» | 35 |
| 3.17 | Объект «Demux» | 36 |
| 3.18 | Объект «Modbus Slave» | 36 |
| 3.19 | Объект «MODBUS_CLIENT»..... | 38 |
| 3.19.1 | Конфигурация объекта «MODBUS_DLL_CONFIG»..... | 42 |
| 3.19.2 | Общая информация по работе со специализированными устройствами | 49 |
| 3.19.3 | Особые замечания для терминалов SEPAM | 53 |
| 3.19.4 | Особые замечания для терминалов MiCOM..... | 54 |
| 3.19.5 | Особые замечания для терминалов БМЦС | 54 |
| 3.19.6 | Сообщения оператору | 54 |
| | Приложение А (справочное) Пример файла конфигурации «config.xml» | 56 |
| | Приложение Б (справочное) Пример файла проекта..... | 57 |
| | Приложение В (справочное) Пример конфигурации объекта «OPCUA_SERVER» и объекта «IEC61850_CLIENT» | 58 |
| | Приложение Г (обязательное) Шаблоны подстановки для объекта формирования единой осциллограммы «R58601» | 60 |
| | Приложение Д (справочное) Пример скрипта создания БД для объекта формирования единой осциллограммы «R58601» | 61 |
| | Приложение Е (обязательное) Шаблоны подстановки данных сигнала в запросе на добавление данных в БД объекта «EventLog» | 65 |
| | Приложение Ж (справочное) Пример скрипта создания БД для объекта «EventLog» | 66 |

| | |
|--|----|
| Приложение И | 69 |
| Приложение К..... | 71 |
| (справочное) Описание служебных параметров для устройств Modbus в зависимости от производителей и их типов | 71 |
| Приложение Л (справочное) Параметры объекта «MODBUS_DLL_CONFIG» | 74 |
| Приложение М..... | 80 |
| (обязательное) Список сокращений | 80 |

1 Общие сведения о программе

1.1 Назначение и функции

Программное обеспечение «Релематика МД» (далее – ПО «Релематика МД») предназначено для работы в системах автоматизации как в качестве отдельной самостоятельной программы, так и в качестве компонента ПТК «UniSCADA». Программа является кроссплатформенной разработкой, что обеспечивает ее выполнение на разных операционных системах, указанных в 1.2.

Функциональным назначением ПО «Релематика МД» является сбор, обработка и передача данных.

Программа выполняет следующие функции:

- считывание файлов конфигурации в формате «*.xml»;
- создавать объекты следующих типов:
 - 1) OPC UA сервер (модуль OPCUA_Server);
 - 2) OPC UA клиент (модуль OPCUA_Client);
 - 3) сервер МЭК 60870-5-104 (IEC104_SERVER);
 - 4) клиент МЭК 60870-5-104 (IEC104_CLIENT);
 - 5) сервер 61850 (IEC61850_Server);
 - 6) клиент 61850 (IEC61850_Client);
 - 7) клиент SNMPV3;
 - 8) объект формирования единой осциллограммы «R58601»;
 - 9) EventLog;
 - 10) Mux, Demux;
 - 11) «Modbus Slave»;
 - 12) «MODBUS_CLIENT»;
 - 13) «MODBUS_DLL_CONFIG»;
- обеспечение механизмом передачи сигналов из одного объекта в другой;
- ограничение объема занимаемого места на диске входных и выходных каталогов.

1.2 Сведения о технических и программных средствах, обеспечивающих выполнение данной программы

Для обеспечения функционирования компонента необходимы следующие характеристики ПЭВМ:

- процессор с тактовой частотой не менее 1,8 ГГц;
- оперативная память объемом не менее 2 ГБ;
- объем свободного места на диске не менее 3 ГБ.

Системными программными средствами, используемые программой, являются:

- ОС Windows 7, 8, 8.1, 10 и выше, Windows Server 2008, 2012, 2016 и выше;
- Linux (Astra Linux 1.7, ALT Linux 10).

2 Структура программы

ПО «Релематика МД» является модульной программой. На рисунке 1 приведена структура ПО «Релематика МД».

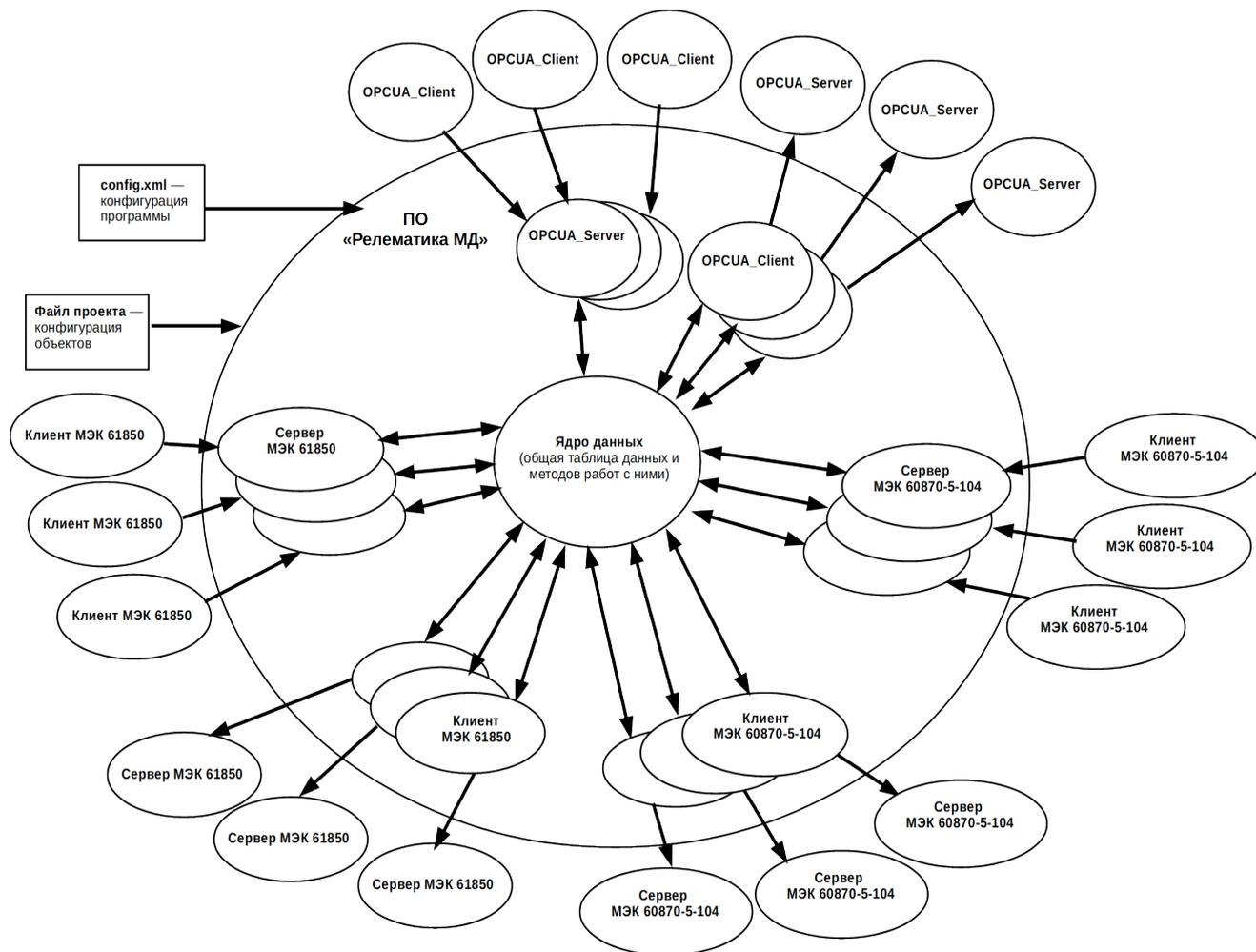


Рисунок 1 – Структура ПО «Релематика МД»

Главным модулем ПО «Релематика МД» является ядро данных, через который происходит взаимодействие всех объектов программы (механизм передачи сигналов из одного объекта в другой). Ядро данных содержит общую таблицу данных и методов работ с ними, т.е. хранит данные и обеспечивает к ним доступ остальным объектам программы. Остальные модули независимы друг от друга, и каждый работает в своем потоке. Следовательно, ПО «Релематика МД» является многопоточной программой.

Объект ПО «Релематика МД» – это класс, реализующий:

- какой-либо протокол обмена с внешними объектами и устройствами;
- обработку данных;
- работу с файлами;
- взаимодействие с базами данных.

Считывание объектов происходит через файл проекта, в котором задаются объекты с нужными параметрами. В конфигурационном файле «config.xml» задаются настройки инициализации ПО.

3 Установка и настройка программы

3.1 Установка программы

ПО «Релематика МД» предоставляется в виде архива «RMD_install.zip» с исполнительным файлом и необходимыми библиотеками. В данном архиве содержатся еще два архива «RMD3xR_Win.zip» и «RMD_18_05_ASTR1_7_3.zip» для установки на ОС «Windows» и ОС «Linux» соответственно.

3.1.1 Установка на ОС «Windows»

Перед установкой программы на ОС «Windows» необходимо установить программу «VC_redist.x64.exe».

Далее распаковать архив «RMD3xR_Win.zip» с исполнительным файлом ПО «Релематика МД» для ОС «Windows» в рабочий каталог. Исполнительный файл программы называется «RMD.exe».

Затем зарегистрировать ПО как службу (3.2).

3.1.2 Установка на ОС «Linux»

Установка программы на ОС «Linux» обязательно должна производиться человеком, прошедшим обучение по применению операционных систем семейства ASTRA LINUX.

Порядок действий установки ПО на ОС «Linux»:

- 1) Перейти к полученной архивной папке с исполнительным файлом ПО «Релематика МД» и драйверами (рисунок 2): *Компьютер > Накопители > Съемный носитель > RMD_install* (в данном случае полученная папка находится на съемном носителе);

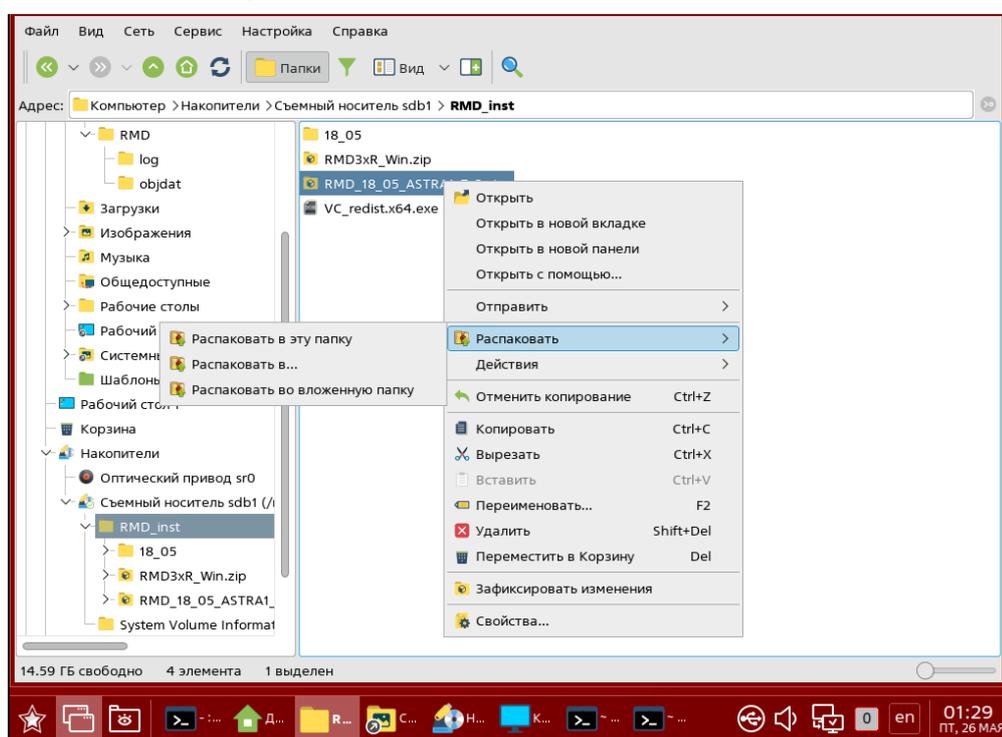


Рисунок 2 – Переход к полученной папке с исполнительным файлом ПО «Релематика МД» и необходимыми библиотеками

Далее выбрать архив «RMD_18_05_ASTR1_7_3.zip» для установки на ОС «Linux» и распаковать ее в корень в папку «RMD_inst».

- 2) Создать рабочую папку «RMD» (можно любое пользовательское наименование) в корне (рисунок 3): *«Компьютер > Файловая система > home > Домашняя»* (можно задать другой путь размещения);

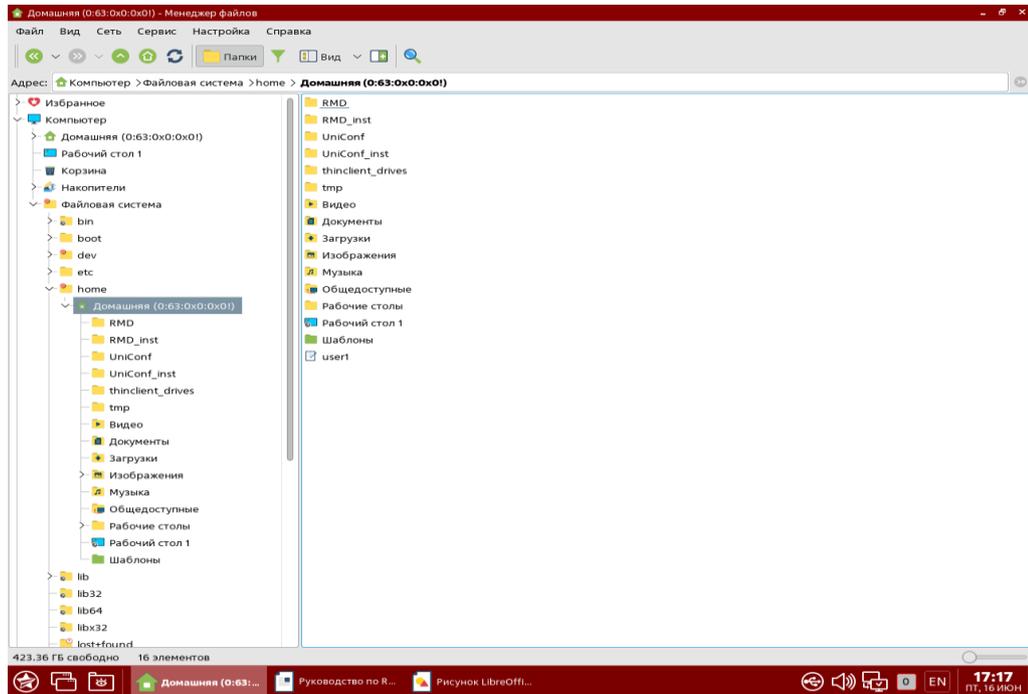


Рисунок 3 – Создание папки «RMD»

- 3) Из папки «RMD_inst» (полученная папка) скопировать файлы: «RMD», «config.xml» и файл проекта (в данном случае файл «cfg1011.xml»).

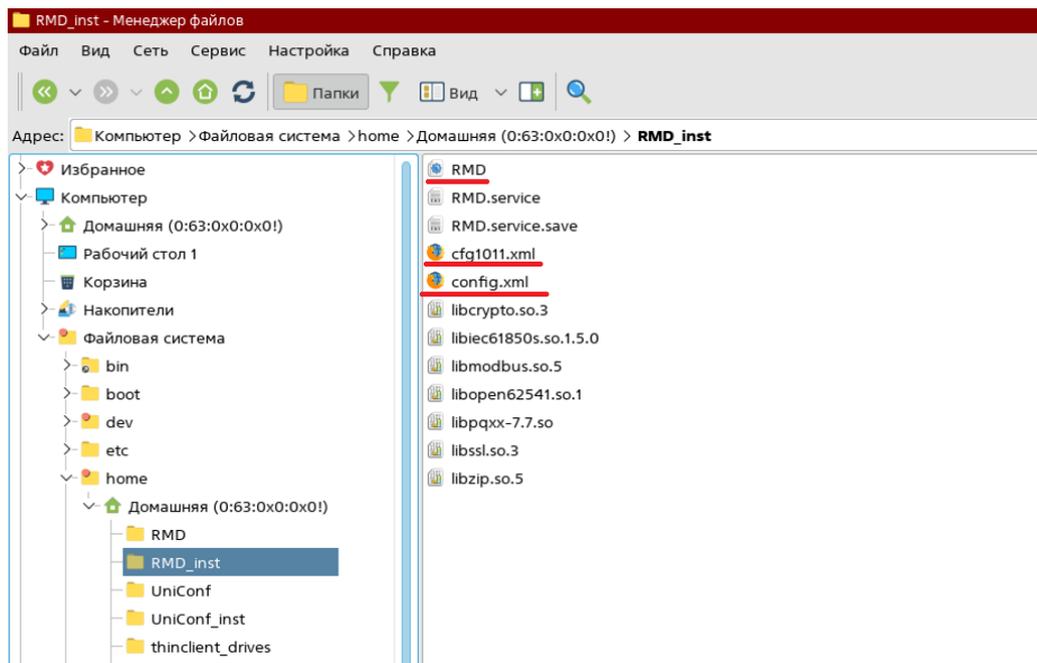


Рисунок 4 – Копируемые файлы из полученной папки
Вставить в созданную папку «RMD»;

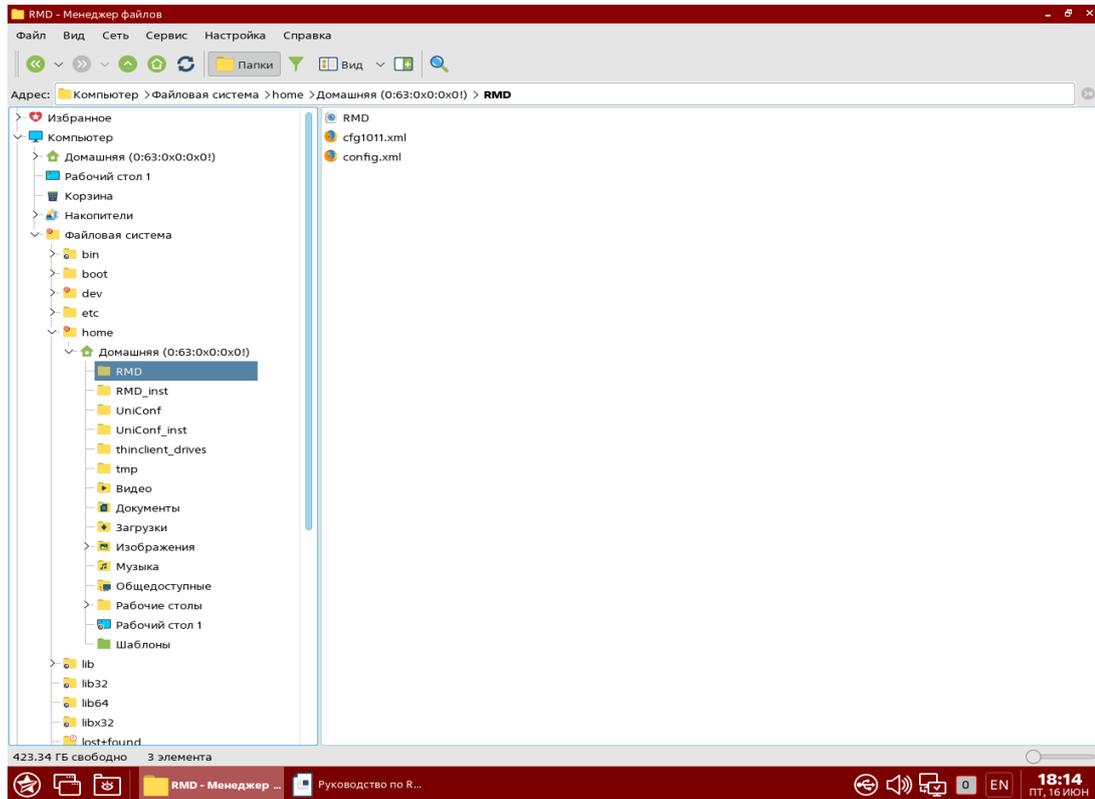


Рисунок 5 – Вставка скопированных файлов в рабочую папку «RMD»

- 4) Зайти в терминал Fly (командная строка) набрав комбинацию «Alt+T» или выбрав «меню Пуск-Системные - Терминал Fly» (рисунок 6);

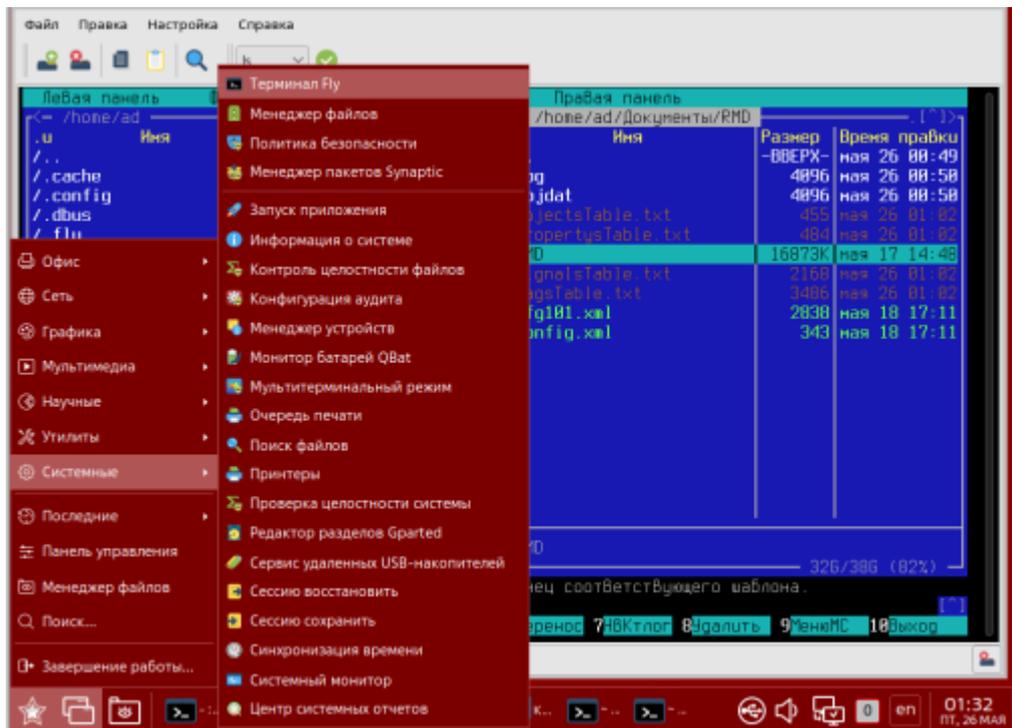


Рисунок 6 – Вызов терминала Fly

Прописать в терминале Fly команду «sudo mc» (повышенные права) (рисунок 7).

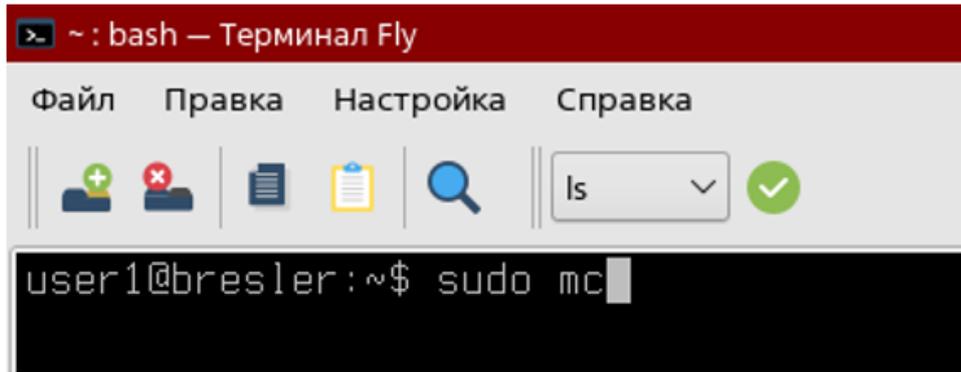


Рисунок 7 – Выполнение команды «sudo mc»

- 5) В левом экране терминала Fly открыть папку «RMD_inst» (полученная папка), в правом экране открыть папку «lib» (usr > local > lib) (рисунок 8).

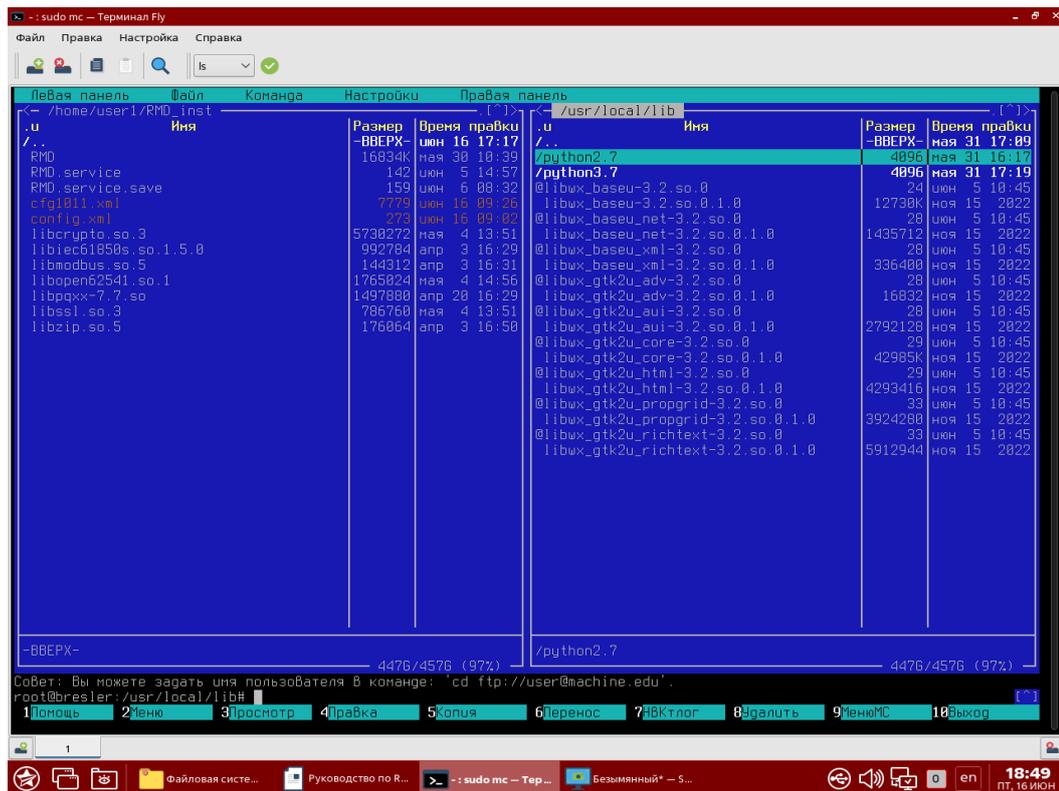


Рисунок 8 – Открытые папки «RMD_inst» и «lib» в терминале Fly

Примечание – Переключение между экранами происходит с помощью кнопки «Tab».

- 6) Из папки «RMD_inst» (полученная папка) скопировать файлы библиотек в папку «lib» (рисунки 9 и 10).

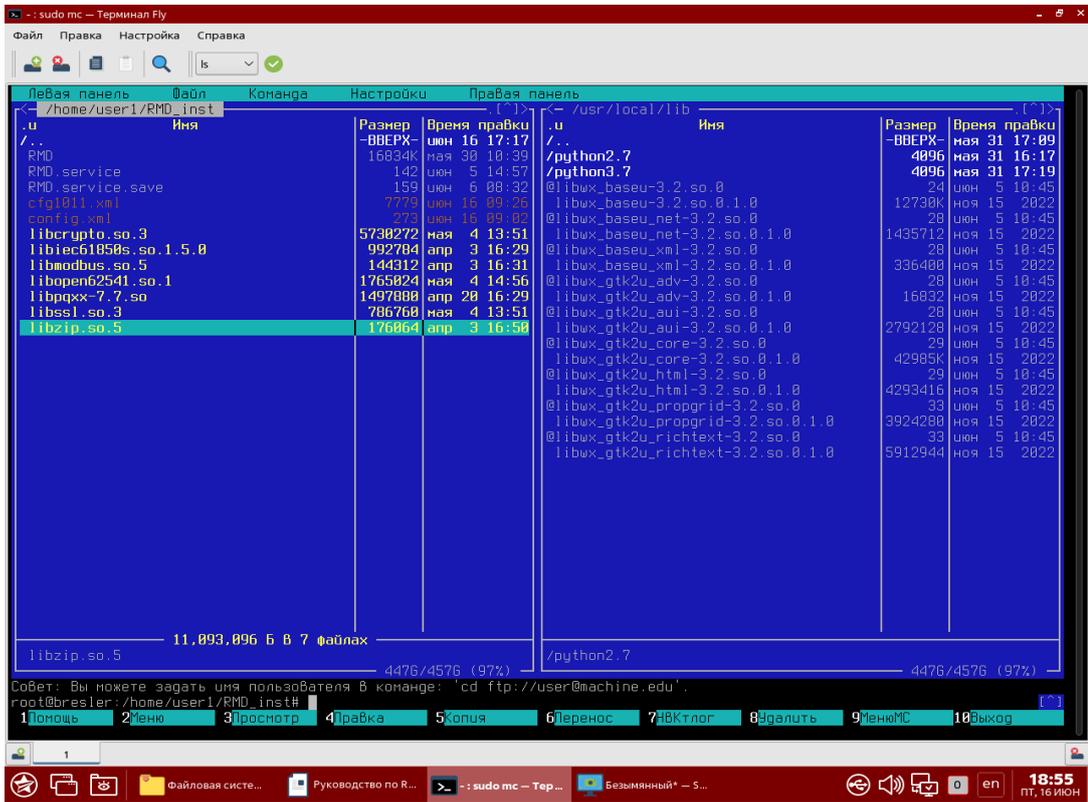


Рисунок 9 – Копирование файлов библиотек в папке «RMD_inst»

Чтобы выделить файлы, нажать кнопку на клавиатуре «Insert», и после выделения всех файлов нажать кнопки на клавиатуре: `F5 > Enter`;

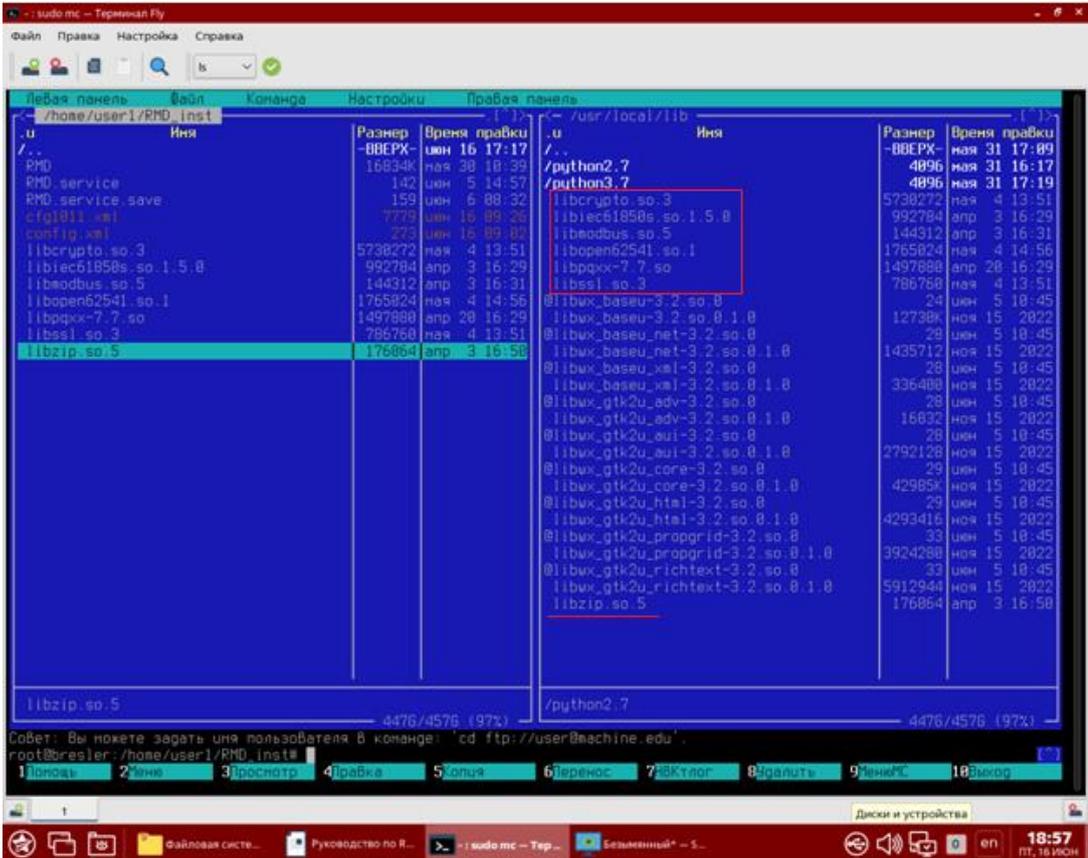


Рисунок 10 – Вставка файлов библиотек в папку «lib»

- 7) Открыть «RMD.service» (находится в полученной папке «RMD_inst») через блокнот «Kate».

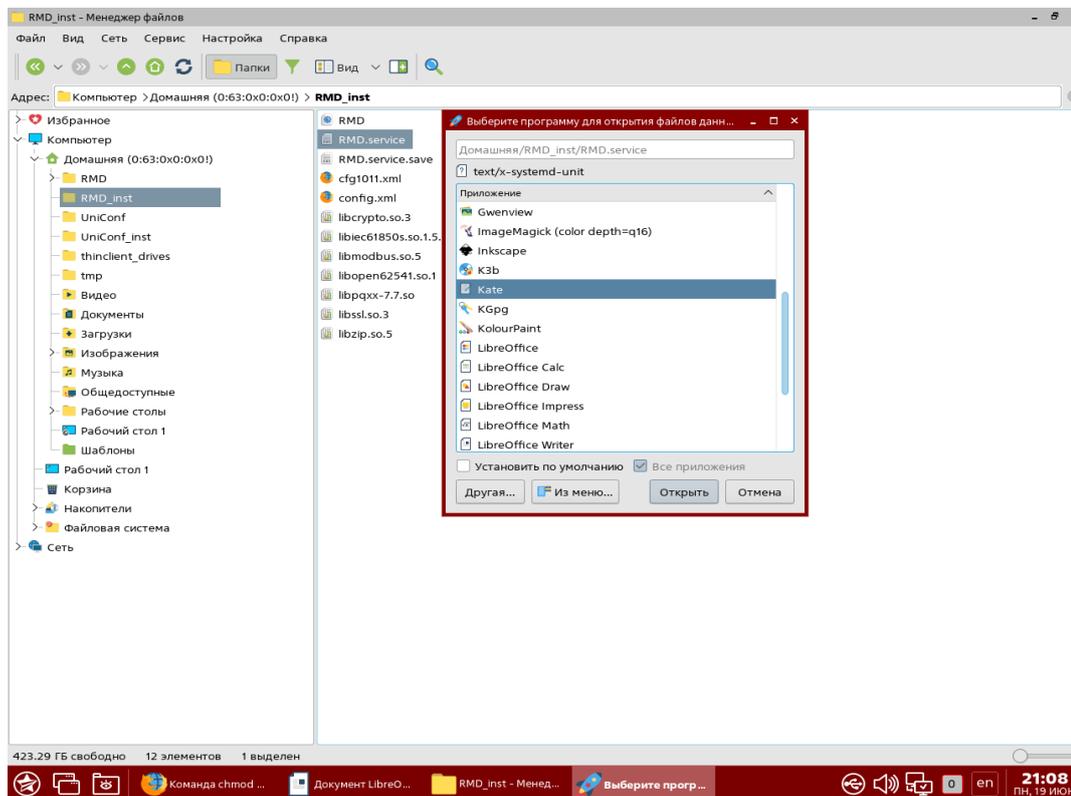


Рисунок 11 – Открытие «RMD.service» через блокнот «Kate»

В открывшемся окне записать (рисунок 12):

- Description = задать имя «демона»;
- WorkingDirectory = рабочая директория, где хранится файл «RMD»;
- ExecStart = путь к исполняемому файлу «RMD».

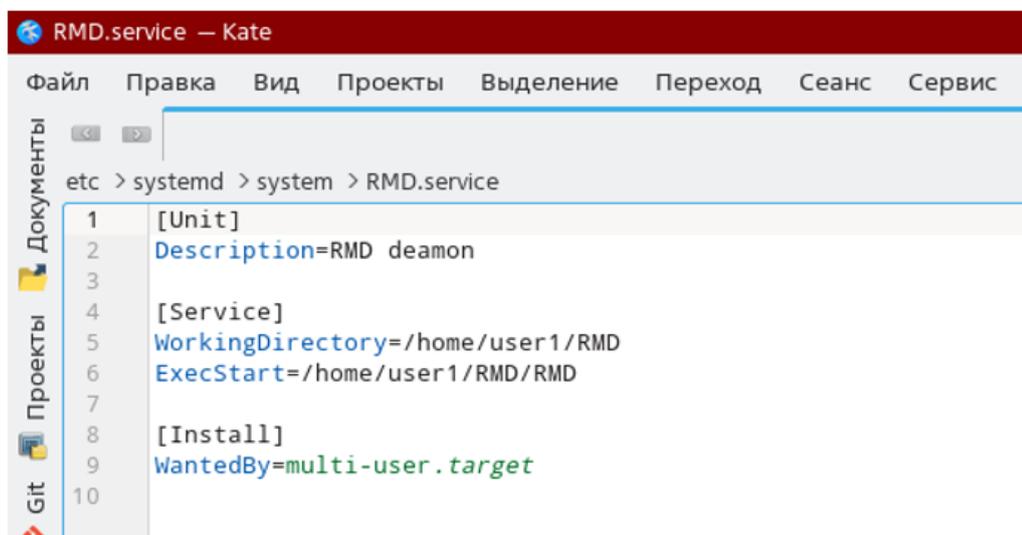


Рисунок 12 – Конфигурация «RMD.service» через блокнот «Kate»

Изменения сохранить;

- 8) В терминале Fly в левом экране открыть папку «RMD_inst» (полученная папка), в правом экране открыть папку «system» (etc > systemd > system) (рисунок 13).

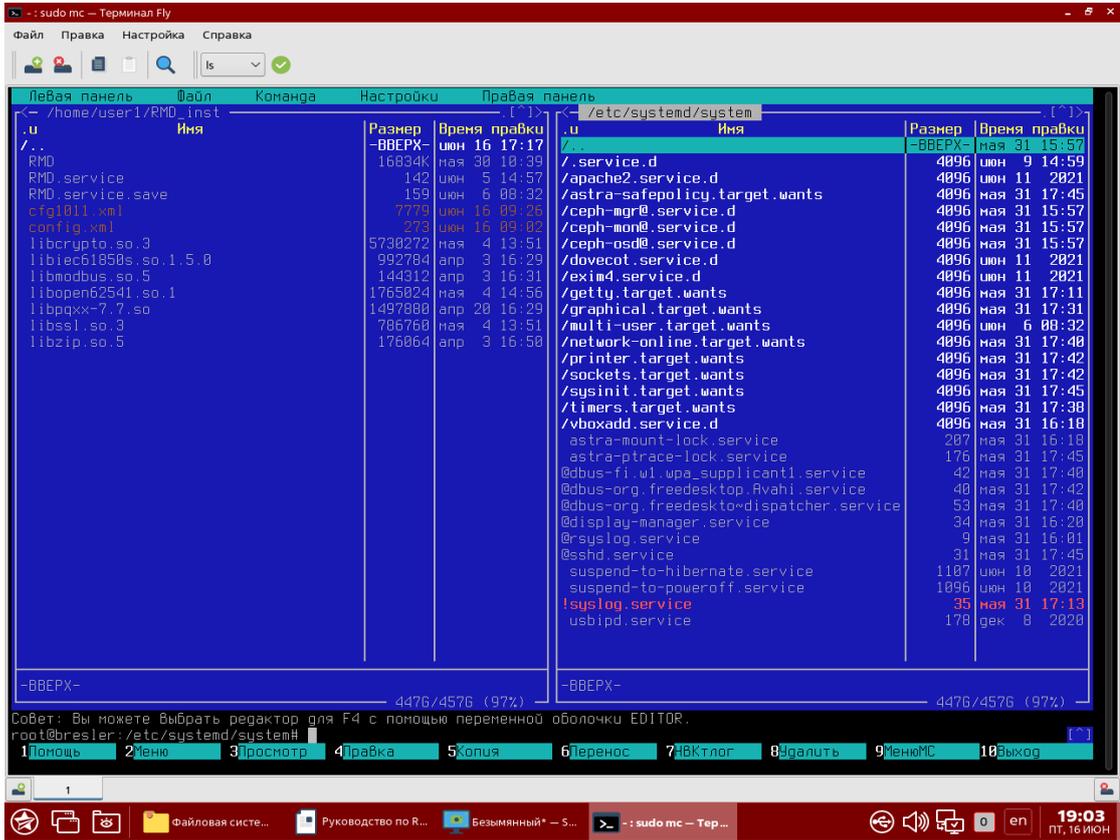


Рисунок 13 – Открытые папки «RMD_inst» и «system» в терминале Fly

9) Из папки «RMD_inst» (полученная папка) скопировать файл «RMD.service» и вставить в папку «system» (рисунки 14 и 15).

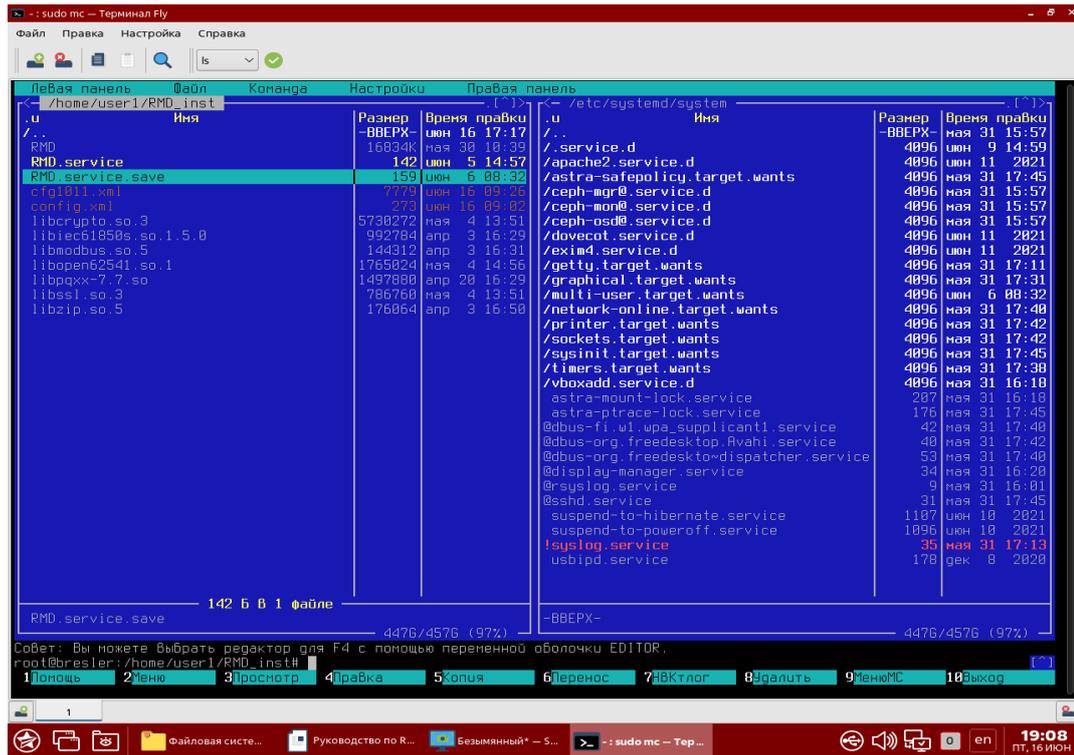


Рисунок 14 – Копирование файла «RMD.service» в папку «RMD_inst»

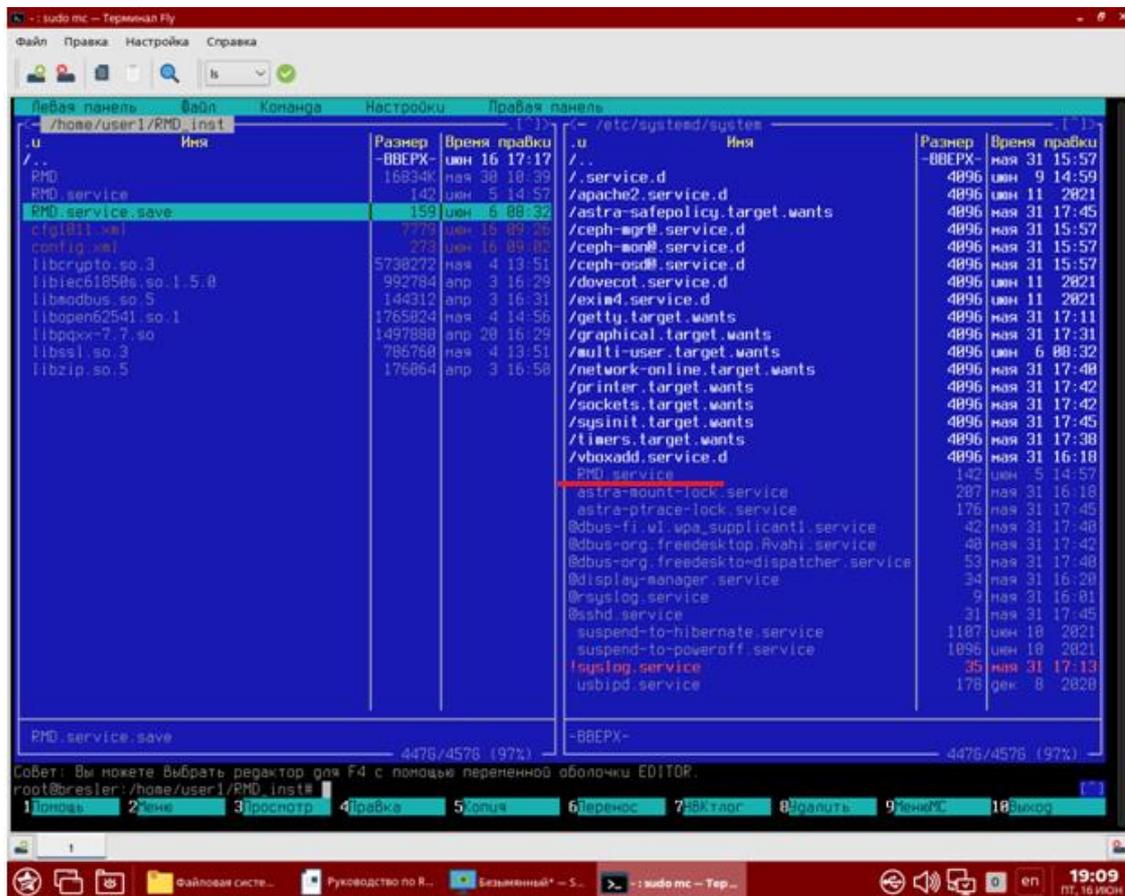


Рисунок 15 – Вставка файла «RMD.service» в папку «system»

10) В терминале в левом окне найти файл «RMD» (рисунок 16): «`home > user1>RMD`» (может храниться в другом месте, зависит от того, куда был сохранен файл по пункту 2);

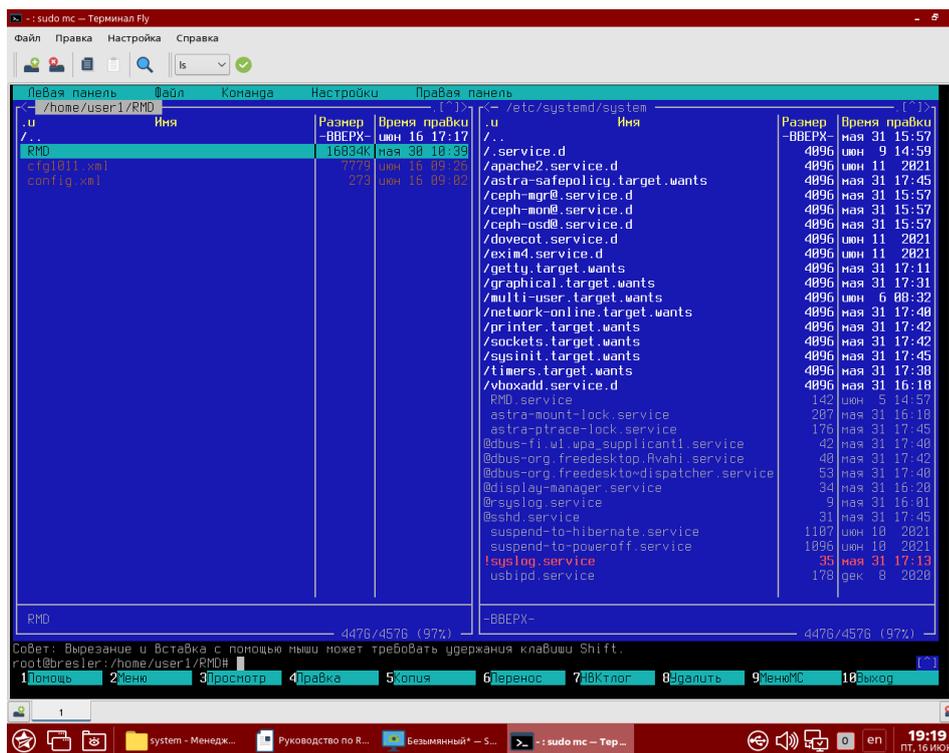


Рисунок 16 – Открытие файла «RMD» в терминале Fly

11) Дать расширенные права файлу «RMD». Для этого имеется два варианта:
1-ый вариант:

- выбрать: *Файл > Права доступа* (рисунок 17);
- отметить пункты указанные на рисунке 17, затем выбрать «Установить»;

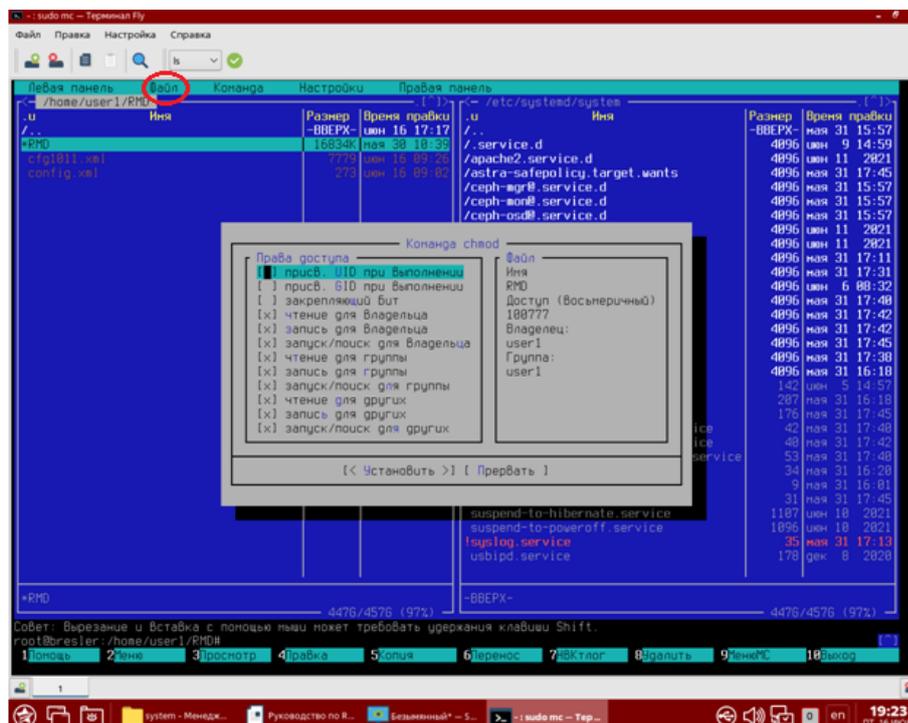


Рисунок 17 – Установка расширенных прав файлу «RMD» с помощью первого варианта
2-ой вариант:

- в предыдущем окне через комбинацию клавиш «*Ctrl+O*» задать расширенные права через команду «*chmod 777 RMD*» (рисунок 18);
- вернуться назад (команда «*Ctrl+O*»);

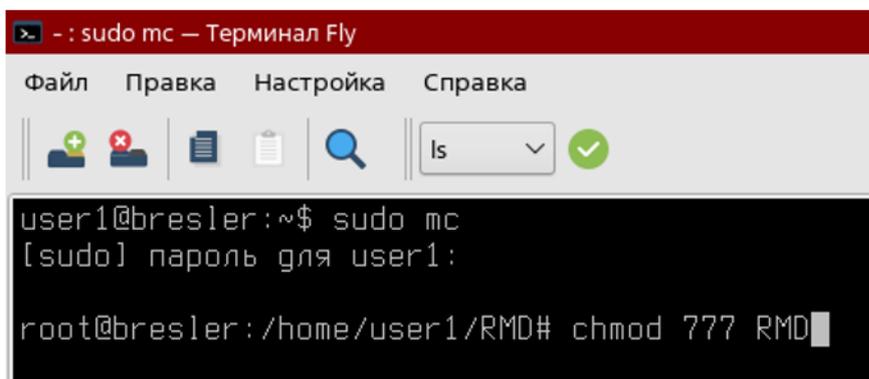


Рисунок 18 – Установка расширенных прав файлу «RMD» с помощью второго варианта

- 12) Выйти из данной панели и в терминале Fly прописать команду «*sudo systemctl daemon-reload*» – перезагрузка «демонов» (рисунок 19);

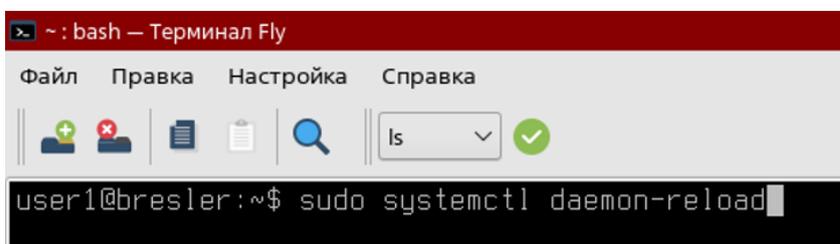


Рисунок 19 – Выполнение команды перезагрузки «демонов»

- 13) Запустить ПО «Релематика МД» в режиме «демона» с помощью команды «*sudo systemctl start RMD*» (рисунок 20).

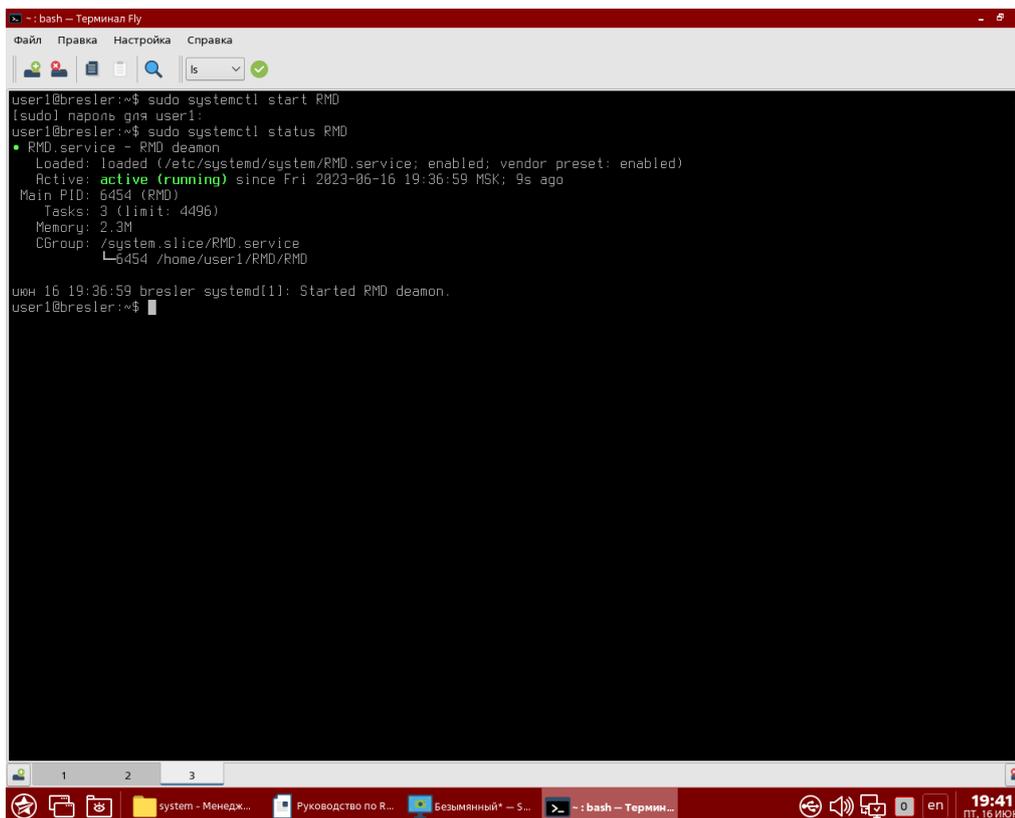


Рисунок 20 – Запуск ПО «Релематика МД»

Запуск RMD: *sudo systemctl start RMD*;
 Статус RMD: *sudo systemctl status RMD*;
 Остановка RMD: *sudo systemctl stop RMD*;
 Автозапуск RMD: *sudo systemctl enable RMD*.

3.2 Параметры командной строки

В ПО «Релематика МД» для регистрации/разрегистрации служб используются параметры командной строки, указанные в таблицах 1 и 2. Для идентификации параметра командной строки перед его названием следует указывать символ «-» или «/».

Пример вводимой строки – C:\Relematika\MD>rmd.exe /s.

C:\Relematika\new\MD>rmd.exe /u.

Примечание – Все действия в командной строке необходимо выполнять с правами администратора.

Таблица 1 – Параметры командной строки, используемые ПО «Релематика МД» для ОС «Windows»

| Параметр | Значение |
|----------|--|
| /s | Зарегистрировать как системный сервис (служба) |
| /u | Разрегистрация сервиса (Uninstall) |

Таблица 2 – Параметры командной строки, используемые ПО «Релематика МД» для ОС «Linux»

| Параметр | Значение |
|----------|--------------------------|
| /d | Запуск в режиме «демона» |

3.3 Файл «config.xml»

Файл конфигурации «config.xml» предназначен для инициализации ПО. Файл содержит ссылку на файл проекта и следующие настройки ПО:

- возможность назначать имя сервису;
- возможность задавать размер и количество логов программы.

Пример «config.xml» файла приведен в приложении А.

Узел «UniIniData» включает в себя:

- атрибут «ModificationDate» – обязательный атрибут файла конфигурации, содержащий информацию о дате его модификации;
- узел регистрации службы «Register». В параметре «ServiceName» задается имя сервиса, если в данном параметре имя сервиса не задано, то задается имя по умолчанию «RelematikaMD»;
- узел «LocalPCHost». Данный узел обязателен и содержит атрибут «cfgfile», в котором задаются путь и имя файла проекта, описывающего внутренние объекты программы.

Узел «LocalPCHost» содержит следующие атрибуты:

- «cfgfile» – путь и имя файла проекта, описывающего внутренние объекты ПО (обязательный атрибут);

Кроме того, узел «LocalPCHost» имеет следующие узлы-параметры:

- «Logs» – опциональный узел, который задает размер и количество лог-файлов программы.

Узел «Logs» содержит следующие атрибуты:

- «CountStr» – максимальное количество строк в лог-файле, опциональный атрибут, по умолчанию задано 10000 строк (0/4294967295) Минимальное значение данного параметра 1000. Если пользователь задал количество строк меньше 1000, то программа самостоятельно увеличит данное значение до 1000;

- «CountFile» – максимальное количество лог-файлов. Опциональный атрибут (0/65535), по умолчанию общее количество логов программы задано: $x \cdot 2 + 1$, где x – количество объектов,

что и является минимальным значением данного параметра. Причем необходимо учесть, что лог-файлы ведутся лишь для тех объектов, которые должны запускаться на данном компьютере. Если пользователь задал данному параметру значение меньше, чем $x \cdot 2 + 1$, то программа самостоятельно увеличит количество логов до минимального значения.

Примечание – При создании лог-файла и при наличии файла с таким именем – расширение старого файла меняется на «*.bak» расширение.

- LogLevel – параметр уровня лога (необязательный параметр). Численные представления уровней лога приведены в таблице 3;

- Path – путь записи папок «log» и «oldlog». В данной директории при работе программы будут создаваться папки «log» и «oldlog» для лог-файлов ПО «Релематика МД». Необязательный параметр, если же данный параметр не задан, то лог-файлы будут записываться в папки «log» и «oldlog» в директории установки ПО «Релематика МД».

Пример – `<Logs CountFile="100" CountStr="800000" LogLevel="5" Path="F:\Relematika\"/>`.

Таблица 3 – Категория и уровни лог-сообщений

| Уровень | Тип лог-сообщений | Описание |
|---------|-------------------|--------------------------------------|
| 1 | E1 | Ошибка |
| 2 | W2 | Предупреждение |
| 3 | M3 | Сообщение программы |
| 4 | D4 | Информация для отладки (LL_DEBUG) |
| 5 | T5 | Информация для трассировки программы |

Примечание – По умолчанию программа включает в лог-файлы сообщения с типами M3, W2 и E1.

3.4 Файл проекта

Файл проекта описывает внутренние объекты ПО, их тип и количество, задает типы получаемых, обрабатываемых и передаваемых данных, задает направления передачи сигналов из одного объекта в другой.

Пример файла проекта приведен в приложении Б.

Узел «rmdcfgdata» включает в себя:

- атрибут «ModificationDate» – обязательный атрибут, который содержит информацию о дате модификации данного файла;

- атрибут «version» – версия формата файла конфигурации (обязательный атрибут). Для программы старше версии 3.0 должен быть равен 2;

- объекты ПО (задается объект и его конфигурация).

Объект задается в формате: «rmdcfgdata/{Тип объекта}». Содержит следующие атрибуты:

- «name» – имя объекта ПО (обязательный атрибут);

- «host» – имя или IP-адрес хоста, на котором будет создаваться объект ПО (обязательный атрибут).

Объекту задаются свойства в формате: «rmdcfgdata/{Тип объекта}/property». Содержат следующие атрибуты:

- «name» – имя свойства (обязательный атрибут);

- «value» – значение свойства (обязательный атрибут).

Объекты могут иметь служебные (системные) теги, которые задаются в формате: «rmdcfgdata/{Тип объекта}/servicetag». Содержат следующие атрибуты:

- «name» – имя сигнала (обязательный атрибут);

- «value» – внешнее представление сигнала (обязательный атрибут);

- «vartype» – тип данных сигнала (обязательный атрибут);

- «sendto» – адрес передачи сигнала (опциональный атрибут, задается только при передаче данных от одного объекта к другому).

Сигналы (теги) в объекте задаются в формате: «rmdcfgdata/{Тип объекта}/tag». Содержат следующие атрибуты:

- «name» – имя сигнала (обязательный атрибут);
- «value» – внешнее представление сигнала (обязательный атрибут);
- «vartype» – тип данных сигнала (обязательный атрибут);
- «k» – коэффициент (опциональный атрибут). Не может быть равен 0, используется для типов VT_I4, VT_R4, VT_R8. Для типов VT_BOOL k=-1 означает инверсию логического сигнала. Коэффициент используется при передаче/приеме данных в другие объекты. При этом при передаче данных происходит умножение, а при приеме - деление;
- «sendto» – адрес передачи сигнала. Опциональный атрибут, задается только при передаче данных от одного объекта к другому, причем возможна передача сигнала сразу в несколько объектов через «,» (максимум до 10 объектов). **Пример** – *sendto="OPC\ARIS.Signal11,Obj60870_1\Signal11,Obj60870_3\Signal11"*;
- «typebuf» – опциональный атрибут, определяет тип буферизации данных. Тип «MEM» обеспечивает буферизацию в оперативной памяти, тип «HDD» – на жестком диске. Если данный атрибут не задан, то буферизация не ведется;
- «deadband» – порог чувствительности аналоговых сигналов, задается в абсолютных величинах (опциональный атрибут и по умолчанию не задан);
- «group» - опциональный атрибут, необходим для разделения данных по группам;
- дополнительные атрибуты, используемые в рамках конкретного объекта.

3.5 Базовый объект

ПО «Релематика МД» имеет базовый объект (родительский объект). Все остальные объекты наследуются от базового объекта, т.е. все свойства и служебные сигналы базового объекта применимы и к другим внутренним объектам программы.

Базовый объект имеет свойства, указанные в таблице 4.

Таблица 4 – Свойства базового объекта

| Свойство | Описание |
|------------------------|--|
| TimeOut | Внутренний период ожидания основного цикла объекта в миллисекундах. Опциональное свойство, по умолчанию задано 3 мс |
| TimeOutSetNotConnect | Таймаут, после которого подается сигнал «NotConnected» (в миллисекундах). Опциональное свойство, по умолчанию задано 3000 мс |
| TimeOutConnection | Интервал между попытками подключения в миллисекундах. Опциональный параметр, по умолчанию задано 10000 мс |
| TimeOutCheckConnection | Интервал между проверками подключения в миллисекундах. Опциональное свойство, по умолчанию задано 1000 мс |
| TimeZona | Метка времени UTC. Необязательный параметр, имеет значения: -12...+12, local (зона должна браться из системных настроек компьютера). По умолчанию задано 0. При получении данных «из вне» объект из метки времени источника сигнала вычитает «TimeZona». При передаче данных «наружу» объект к метке времени прибавляет «TimeZona» |

В таблице 5 представлены служебные сигналы базового объекта.

Таблица 5 – Служебные сигналы базового объекта

| Сигнал | Тип | Описание |
|--------------|---------|--|
| NotConnected | VT_BOOL | Отслеживание состояния подключения (нет связи с устройством) |

| | | |
|----|---------|---|
| IU | VT_BOOL | Управление состоянием объекта (включение/выключение объекта) |
|----|---------|---|

Пример – `<servicetag name="NotConnected" value="NotConnected" vartype="VT_BOOL" sendto="OPCUASERVER\SNMPCln_1.NotConnected"/>`

`<servicetag name="IU" value="IU" vartype="VT_BOOL" sendto="OPCUASERVER\SNMPCln_1.IU"/>`

3.6 Объект «OPCUA_SERVER»

Объект «OPCUA_SERVER» обеспечивает обмен данными с внешними OPC UA клиентами. Пример объекта приведен в приложении В.

OPC UA сервер имеет свойства, указанные в таблице 6.

Таблица 6 – Свойства объекта «OPCUA_SERVER»

| Свойство | Описание |
|------------------|--|
| NameServer | Имя OPCUA сервера для подключения внешних OPCUA клиентов. Опциональный параметр, по умолчанию имя «Relematika OPCUA Server» |
| TCPPort | Порт прослушивания OPCUA сервера. Опциональный параметр, по умолчанию порт 4840 |
| Certificate | Сертификат и ключ сервера в двоичном виде. Опциональные свойства. Если будет отсутствовать хотя бы один из данных параметров, то значения свойств «AllowUnencrypted» и «AcceptAllCerts» устанавливаются в «1» |
| PrivateKey | |
| AllowUnencrypted | Разрешение на нешифрованное соединение с сервером: «0» – запрещено, «1» – разрешено. Опциональное свойство, по умолчанию «1» |
| AcceptAllCerts | Разрешение на подключение клиенту с любым сертификатом: «0» – запрещено, «1» – разрешено. Опциональное свойство, по умолчанию «1». Если запрещено, то допускаются лишь клиенты с сертификатом как у сервера |
| UsersCount | Количество пользователей сервера. Опциональное свойство, по умолчанию «0». Максимальное значение «256». Если установлено «0», то разрешается анонимное подключение к серверу |
| Username1 | Имя пользователя и пароль в зашифрованном виде. Имя каждого пользователя должно быть уникальным. В конец названия свойства добавляется порядковый номер пользователя. Максимальный номер пользователя определяется в свойстве «UsersCount» |
| Password1 | |

Сигналы в данном объекте задаются согласно синтаксису, описанному в 3.4

3.7 Объект «OPCUA_CLIENT»

В программе реализован объект «OPCUA_CLIENT». Данный объект обеспечивает:

- подключение к внешним OPC UA серверам;
- обмен данными по стандарту OPC UA.

В файле конфигурации проекта OPC UA клиент задается следующим образом:

```
<OPCUA_CLIENT name="UAClient">
  <property name="ServerIP" value="127.0.0.1"/>
  <property name="ServerPort" value="48010"/>
</OPCUA_CLIENT>
```

```

    <property name="Certificate" value="Сертификат в двоичном
виде"/>
    <property name="PrivateKey" value="Ключ в двоичном виде"/>
    <property name="Username" value="Имя пользователя"/>
    <property name="Password" value="Пароль в зашифрованном виде"/>
    <property name="SecurityMode" value="1"/>

    <tag name="Tag1" value="Demo.Dynamic.Scalar.Boolean"
vartype="VT_BOOL" namespace="2" k="0.1" sendto="UAServer\Tag1"/>
    <tag name="Tag2" value="Demo.Dynamic.Scalar.Int32" vartype="VT_I4"
namespace="2" sendto="UAServer\Tag2"/>
    <tag name="Tag3" value="Demo.Static.Scalar.Int32" vartype="VT_I4"
namespace="2" sendto="UAServer\Tag3"/>
</OPCUA_CLIENT>.

```

При этом в соответствии настройкам OPC UA клиента в объект «OPCUA_SERVER» добавить:

```

<OPCUA_SERVER name="UAServer">
    <tag name="Tag1" value="Boolean_1" vartype="VT_BOOL"/>
    <tag name="Tag2" value="Int32_1" vartype="VT_I4"/>
    <tag name="Tag3" value="Int32_Wr" vartype="VT_I4"/>
</OPCUA_SERVER>.

```

OPC UA клиент имеет свойства, указанные в таблице 7.

Таблица 7 – Свойства объекта «OPCUA_CLIENT»

| Свойство | Описание |
|--------------|---|
| ServerIP | IP-адрес внешнего сервера, к которому подключается объект (обязательный параметр объекта) |
| ServerPort | TCP-порт внешнего сервера (обязательный параметр объекта) |
| Certificate | Сертификат и ключ клиента в двоичном виде. Опциональные свойства. Если отсутствует хотя бы один из данных параметров, то значение свойства «SecurityMode» будет установлено в «1» |
| PrivateKey | |
| Username | Имя пользователя и пароль в зашифрованном виде. Опциональные свойства объекта. Если не указан один из параметров, то будет устанавливаться анонимное подключение с сервером |
| Password | |
| SecurityMode | Режим безопасности подключения. Возможны следующие значения: <ul style="list-style-type: none"> – «1» – None; – «2» – Sign; – «3» – Sign and Encrypt. Опциональное свойство. Значение по умолчанию – «1» |

Сигналы в данном объекте задаются согласно синтаксису, описанному в 3.4. Объект дополнительно имеет опциональный атрибут «namespace» – пространство имен (область действия сигнала). Диапазон возможных значений: 0–255. Значение по умолчанию – «2», при необходимости задать другую область сигнала – в конфигурации указывается данный атрибут с иным возможным значением.

3.8 Объект «IEC104_SERVER»

Объект «IEC104_SERVER» обеспечивает передачу данных по стандарту ГОСТ Р МЭК 60870-5-104-2004. При обмене данных программа поддерживает стандартные ASDU пакеты: 1, 3, 9, 13, 30, 31, 36, 37, 45, 46, 47, 100, 103.

В файле конфигурации проекта сервер МЭК 60870-5-104 задается следующим образом:

```
<IEC104_SERVER name="Obj60870">
  <property name="IPAddress" value="127.0.0.1"/>
  <property name="TCPPort" value="2404"/>
  <property name="TimeOut" value="50"/>
  <property name="TimeOutCheckConnection" value="1000"/>
  <property name="TimeOutConnection" value="3000"/>
  <property name="TimeOutT0" value="1000"/>
  <property name="TimeOutT1" value="15000"/>
  <property name="TimeOutT2" value="10000"/>
  <property name="TimeOutT3" value="20000"/>
  <property name="TimeOutSetNotConnect" value="1000"/>
  <property name="StationAddress" value="1"/>
  <property name="AcknowledgeK" value="12"/>
  <property name="AcknowledgeW" value="8"/>
  <servicetag name="IU" value="IU" vartype="VT_BOOL"
sendto="OPC\Obj60870.IU"/>
  <servicetag name="NotConnected" value="NotConnected"
vartype="VT_BOOL" sendto="OPC\Obj60870.NotConnected"/>
  <tag name="Signal1" value="1" asdu="13" vartype="VT_R4" />
  <tag name="Signal2" value="2" asdu="13" vartype="VT_R4" />
  <tag name="Signal3" value="3" asdu="1" vartype="VT_BOOL" />
  <tag name="Signal4" value="1000" asdu="36" vartype="VT_R4" />
  <tag name="Signal5" value="1001" asdu="36" vartype="VT_R4" />
  <tag name="Signal6" value="1002" asdu="36" vartype="VT_R4" />
  <tag name="Signal7" value="1003" asdu="13" vartype="VT_R4" />
  <tag name="Signal8" value="1004" asdu="13" vartype="VT_R4" />
  <tag name="Signal9" value="1005" asdu="13" vartype="VT_R4" />
  <tag name="Signal10" value="2000" asdu="1" vartype="VT_BOOL" />
  <tag name="Signal11" value="2001" asdu="1" vartype="VT_BOOL" />
  <tag name="Signal" value="666" asdu="45" vartype="VT_BOOL"
sendto="OPC\ARIS.Signal"/>
</IEC104_SERVER>.
```

Данному объекту задаются следующие служебные параметры:

- IPAddress – IP-адрес сервера/APM, на котором запускается данный объект (обязательное свойство объекта);
- TCPPort – порт прослушивания сервера/APM (опциональный параметр, по умолчанию задан порт 2404);
- StationAddress – общий адрес ASDU сервера/APM, на котором запускается данный объект (обязательный параметр объекта);
- TimeOutT0 – Time-out при установлении соединения, $t_0 = (1-255)$ с (опциональное свойство, по умолчанию задано значение 1000 мс);
- TimeOutT1 – Time-out при посылке или тестировании APDU, $t_1 = (1-255)$ с (опциональное свойство, по умолчанию задано значение 15000 мс);
- TimeOutT2 – Time-out для подтверждения в случае отсутствия сообщения с данными, $t_2 = (1-255)$ с, причем $t_2 < t_1$ (опциональное свойство, по умолчанию задано значение 10000 мс);
- TimeOutT3 – Time-out для посылки блоков тестирования в случае долгого простоя, $t_3 = (1-255)$ с (опциональное свойство, по умолчанию задано значение 20000 мс);
- AcknowledgeK – максимальная разность между переменной состояния передачи и номером последнего подтверждения, $k = 1-32767$ (количество принятых I-пакетов до подтверждения). Опциональный узел, по умолчанию задано 12 пакетов;
- AcknowledgeW – последнее подтверждение после приема w APDU формата I, $w = 1-32767$, $w < 2/3 \cdot k$ (максимальное число неподтверждённых пакетов). Данный узел – опция, по умолчанию задано: value="8".

Сигналы задаются согласно синтаксису, описанному в 3.4. У сигналов данного объекта в атрибуте «value» указывается адрес сигнала согласно стандарту ГОСТ Р МЭК 60870-5-104-2004

(обязательный параметр). Объект дополнительно содержит обязательный атрибут «asdu», в котором задается тип пакета.

Важное уточнение: к одному объекту, в один момент времени, может быть подключен только один клиент.

Также стоит отметить, что свойство typebuf="MEM" указывает на то, что сигнал необходимо поместить в буфер, по умолчанию не буферизируется.

3.9 Объект «IEC104_CLIENT»

Объект «IEC104_CLIENT» (клиент МЭК 60870-5-104) инициализирует соединения с устройствами (внешними серверами МЭК 60870-5-104) и получает от них данные. В файле конфигурации проекта объект задается следующим образом:

```
<IEC104_CLIENT name="Obj60870_2">
  <property name="IPAddress" value="192.168.102.200"/>
  <property name="TCPPort" value="2404"/>
  <property name="TimeOut" value="50"/>
  <property name="TimeOutCheckConnection" value="1000"/>
  <property name="TimeOutConnection" value="3000"/>
  <property name="TimeOutT0" value="1000"/>
  <property name="TimeOutT1" value="15000"/>
  <property name="TimeOutT2" value="10000"/>
  <property name="TimeOutT3" value="20000"/>
  <property name="TimeOutGI" value="60000"/>
  <property name="TimeOutSetNotConnect" value="1000"/>
  <property name="StationAddress" value="1"/>
  <property name="AcknowledgeK" value="12"/>
  <property name="AcknowledgeW" value="8"/>
  <property name="UseASDU70" value="0"/>
  <servicetag name="IU" value="IU" vartype="VT_BOOL"
sendto="OPC\Obj60870_2.IU"/>
  <servicetag name="NotConnected" value="NotConnected"
vartype="VT_BOOL" sendto="OPC\Obj60870_2.NotConnected"/>
  <tag name="SSignal1" value="1" asdu="13" vartype="VT_R4"
sendto="OPC\ARIS.Signal1,Obj60870_1\Signal1,Obj60870_3\Signal1"/>
  <tag name="SSignal2" value="2" asdu="13" vartype="VT_R4"
sendto="OPC\ARIS.Signal2,Obj60870_1\Signal2,Obj60870_3\Signal2"/>
  <tag name="SSignal3" value="3" asdu="1" vartype="VT_BOOL"
sendto="OPC\ARIS.Signal3,Obj60870_1\Signal3,Obj60870_3\Signal3"/>
  <tag name="SSignal4" value="1000" asdu="36" vartype="VT_R4"
typebuf="MEM"
sendto="OPC\ARIS.Signal4,Obj60870_1\Signal4,Obj60870_3\Signal4"/>
  <tag name="SSignal5" value="1001" asdu="36" vartype="VT_R4"
typebuf="MEM"
sendto="OPC\ARIS.Signal5,Obj60870_1\Signal5,Obj60870_3\Signal5"/>
  <tag name="SSignal6" value="1002" asdu="36" vartype="VT_R4"
typebuf="MEM"
sendto="OPC\ARIS.Signal6,Obj60870_1\Signal6,Obj60870_3\Signal6"/>
  <tag name="SSignal7" value="1003" asdu="13" vartype="VT_R4"
sendto="OPC\ARIS.Signal7,Obj60870_1\Signal7,Obj60870_3\Signal7"/>
  <tag name="SSignal8" value="1004" asdu="13" vartype="VT_R4"
sendto="OPC\ARIS.Signal8,Obj60870_1\Signal8,Obj60870_3\Signal8"/>
  <tag name="SSignal9" value="1005" asdu="13" vartype="VT_R4"
sendto="OPC\ARIS.Signal9,Obj60870_1\Signal9,Obj60870_3\Signal9"/>
  <tag name="SSignal10" value="2000" asdu="1" vartype="VT_BOOL"
sendto="OPC\ARIS.Signal10,Obj60870_1\Signal10,Obj60870_3\Signal10"/>
  <tag name="SSignal11" value="2001" asdu="1" vartype="VT_BOOL"
sendto="OPC\ARIS.Signal11,Obj60870_1\Signal11,Obj60870_3\Signal11"/>
  <tag name="SSignal" value="666" asdu="45" vartype="VT_BOOL"/>
</IEC104_CLIENT>.
```

Клиент МЭК 60870-5-104 имеет следующие параметры:

- IPAddress – адрес оборудования (обязательный параметр объекта);
- Port – порт оборудования, по которому передаются данные по стандарту ГОСТ Р МЭК 60870-5-104-2004 (опциональный параметр, по умолчанию задан порт 2404);
- StationAddress – ASDU-адрес устройства, указываемый в посылках (обязательный атрибут);
 - TimeOutT0 – Time-out при установлении соединения, $t_0 = (1-255)$ с (опциональное свойство, по умолчанию задано значение 1000 мс);
 - TimeOutT1 – Time-out при посылке или тестировании APDU, $t_1 = (1-255)$ с (опциональное свойство, по умолчанию задано значение 15000 мс);
 - TimeOutT2 – Time-out для подтверждения в случае отсутствия сообщения с данными, $t_2 = (1-255)$ с, причем $t_2 < t_1$ (опциональное свойство, по умолчанию задано значение 10000 мс);
 - TimeOutT3 – Time-out для посылки блоков тестирования в случае долгого простоя, $t_3 = (1-255)$ с (опциональное свойство, по умолчанию задано значение 20000 мс);
 - TimeOutGI – Time-out общего опроса. Опциональный параметр, по умолчанию задано 0 мс – значит общий опрос никогда не выполняется. Если параметр задан и отличен от 0, то с данной периодичностью выполняется общий опрос;
 - AcknowledgeK – максимальная разность между переменной состояния передачи и номером последнего подтверждения, $k = 1-32767$ (количество принятых I-пакетов до подтверждения). Опциональный узел, по умолчанию задано 12 пакетов;
 - AcknowledgeW – последнее подтверждение после приема w APDU формата I, $w = 1-32767$, $w < 2/3 \cdot k$ (максимальное число неподтверждённых пакетов). Данный узел – опция, по умолчанию задано: value="8".

Сигналы задаются согласно синтаксису, описанному в 3.4. У сигналов данного объекта в атрибуте «value» указывается адрес сигнала согласно стандарту ГОСТ Р МЭК 60870-5-104-2004 (обязательный параметр). Объект дополнительно содержит обязательный атрибут «asdu», в котором задается тип пакета.

3.10 Объект «IEC61850_SERVER»

В ПО «Релематика МД» реализован обмен данных по стандарту МЭК 61850 в режиме сервера. Для этого, в программе создается объект «IEC61850_SERVER».

Функции объекта:

- возможность подключения устройств (внешних клиентов МЭК 61850);
- считывание файла конфигурации объектной модели сервера 61850 в формате «*.cfg»;
- передача файлов с указанного места.

В файле конфигурации проекта сервер 61850 задается следующим образом:

```
<IEC61850_SERVER name="obj_MMS" >
  <property name="IPAddress" value="127.0.0.1"/>
  <property name="ConfigFile" value="modell1.cfg"/>
  <property name="FileStore" value="filestore"/>
  <property name="ReportName" group="1"
value="TEMPLATEGenericIO/LLN0.BR.Measurements01"/>
  <property name="Tag1" group="1"
value="TEMPLATEGenericIO/GGIO1.AnIn1[MX]mag.f#q=q#t=t" vartype="VT_I4" />
  <property name="Control" group="2" value="SBOw"/>
  <tag name="Tag2" group="2"
value="TEMPLATEGenericIO/GGIO1.SPCS01#q=connect#t=utclocal" typebuf="MEM"
vartype="VT_I4" sendto="IPCOPCServer\Tag2"/>
</IEC61850_SERVER>.
```

В файл конфигурации объекта «OPCUA_SERVER» добавить:

```
<OPCUA_SERVER name="IPCOPCServer">
  <tag name="Tag1" value="GGIO1.AnIn1" vartype="VT_I4"
sendto="obj_MMS\Tag1\1"/>
  <tag name="Tag2" value="GGIO1.SPCS01" vartype="VT_I4"/>
</OPCUA_SERVER>.
```

В таблице 8 представлены свойства данного объекта.

Таблица 8 – Свойства объекта «IEC61850_SERVER»

| Свойство | Описание |
|------------|---|
| IPAddress | IP-адрес сервера (обязательный параметр объекта. Адрес должен быть версии IPv4 в формате четырёх десятичных чисел (от 0 до 255)) |
| TCPPort | TCP-порт, через который происходит соединение с сервером (необязательный параметр, по умолчанию 102) |
| ConfigFile | Файл объектной модели сервера (обязательный параметр). Если в данном параметре указано только имя, то поиск файла происходит в той же папке, где расположен исполняемый файл «gmd.exe». Если же указан и путь, то поиск файлов будет происходить по заданному пути. <i>Пример – <property name="ConfigFile" value="D:\Relematika\MD\model.cfg"/></i> |
| FileStore | Путь к папке с файлами. Данный параметр обязателен для передачи файлов внешним клиентам МЭК 61850 с указанного места |
| ReportName | Свойство группы отчета (обязательный параметр при передаче данных через отчеты) |
| Control | Свойство группы тегов управления, имеет следующие возможные значения (типы управления устройством): <ul style="list-style-type: none"> – DO – direct control with normal security; – DOw – direct control with enhanced security; – SBO – select before operate with normal security; – SBOw – select before operate with enhanced security. Параметр обязателен и необходимо задавать |

Сигналы задаются согласно синтаксису, описанному в 3.4. У сигналов данного объекта в атрибуте «value» указывается адрес значения сигнала, метки времени и качества сигнала.

3.11 Объект «IEC61850_CLIENT»

Объект «IEC61850_CLIENT» (клиент 61850) обеспечивает прием данных по протоколу МЭК 61850 от внешнего сервера МЭК 61850. Для конфигурирования объекта имеются следующие группы параметров:

- параметры подключения устройства (внешнего сервера МЭК 61850);
- параметры отчетов;
- параметры команд;
- параметры считывания файлов.

Пример конфигурации клиента 61850 приведен в приложении В.

3.11.1 Параметры подключения устройства

Для настройки соединения с внешним сервером МЭК 61850 объекту задаются следующие параметры:

- IPAddress – IP-адрес устройства (обязательный параметр). Адрес должен быть версии IPv4 в формате четырёх десятичных чисел (от 0 до 255);
- TCPPort – TCP-порт, через который происходит соединение с устройством (необязательный параметр, по умолчанию задан порт 102);
- MaxCountErrorConnect – максимальное количество попыток соединения с сервером (опциональный параметр, по умолчанию данный параметр равен возможному максимальному значению, т.е. бесконечности, поэтому попытки соединения с сервером будут продолжаться до тех пор, пока не достигнут подключения (0/65535));

- TimeOutNotConnect – период выставления признака «Нет соединения с сервером». Необязательный параметр, по умолчанию задано 10000 мс (0/2147483647). Отсчитывается с момента потери соединения с сервером.

3.11.2 Параметры отчетов

Объект имеет возможность получать данные с устройства в виде отчетов. Для конфигурирования процесса задаются следующие служебные параметры:

- ReportName – имя отчета (имя отчета задается такое же, как и на устройстве). Параметр является обязательным;
- IntgPd – период отправки информации с устройства. Необязательный параметр, по умолчанию данный период не используется (0/2147483647);
- ReportOpt – причина формирования отчета в устройстве, имеет следующие возможные значения:

- 1) TRG_OPT_DATA_CHANGED – данные изменились;
- 2) TRG_OPT_QUALITY_CHANGED – качество изменилось;
- 3) TRG_OPT_DATA_UPDATE – данные обновились;
- 4) TRG_OPT_INTEGRITY – периодический опрос;
- 5) TRG_OPT_GI – общий опрос.

Ввод нескольких причин, по которым формируется отчет, происходит через знак «|».

*Пример – TRG_OPT_DATA_CHANGED/TRG_OPT_QUALITY_CHANGED/
TRG_OPT_DATA_UPDATE/TRG_OPT_GI.*

Если данный параметр не указывать в программе, то отчеты будут приходиться по заданным в устройстве настройкам;

- OptFlds – опциональные поля выводимые в отчете. Данному параметру предусмотрены следующие возможные значения (поля):

- 1) RPT_OPT_SEQ_NUM;
- 2) RPT_OPT_TIME_STAMP;
- 3) RPT_OPT_REASON_FOR_INCLUSION;
- 4) RPT_OPT_DATA_SET;
- 5) RPT_OPT_DATA_REFERENCE;
- 6) RPT_OPT_BUFFER_OVERFLOW;
- 7) RPT_OPT_ENTRY_ID;
- 8) RPT_OPT_CONF_REV.

Примечание – Для устройства данный параметр необязателен: если параметр в программе не задан, то устройство будет отправлять отчеты с обязательными полями (в отчете согласно стандарту МЭК 61850 имеются обязательные и опциональные поля).

По умолчанию в программе заданы следующие опциональные поля, включаемые в отчеты: «RPT_OPT_DATA_REFERENCE|RPT_OPT_REASON_FOR_INCLUSION|RPT_OPT_ENTRY_ID».

Сигналы задаются согласно синтаксису, описанному в 3.4. Для данного объекта в параметре «value» указывается адрес значения сигнала, метки времени и качества сигнала.

3.11.3 Параметры команд

Клиент 61850 имеет возможность посылать в устройство команды. Для подачи тегов управления на устройство задаются следующие служебные параметры:

- OriginIdent – идентификатор происхождения. Данный параметр описан в стандарте МЭК 61850, имеет текстовое значение. Является необязательным параметром объекта и по умолчанию задано значение «rmd»;
- OriginCat – данный параметр описан в стандарте МЭК 61850. Принимает следующие возможные значения (статусы):

- 1) not-supported;
- 2) bay-control;
- 3) station-control;
- 4) remote-control;
- 5) automatic-bay;

- 6) automatic-station;
- 7) automatic-remote;
- 8) maintenance.

Параметр не обязателен и по умолчанию задано значение «station-control»;

- BlockSendData – сигнал отключения передачи данных. Опциональное свойство, по умолчанию задано значение «0»;

- BlockReadCmd – сигнал приема команды. Опциональное свойство, по умолчанию задано значение «0»;

- Control – абстрактный сервис управления согласно стандарту МЭК 61850 (данный параметр обязателен и необходимо задавать). Параметр имеет следующие возможные значения (типы управления устройством):

- 1) DO – direct control with normal security;
- 2) DOw – direct control with enhanced security;
- 3) SBO – select before operate with normal security;
- 4) SBOw – select before operate with enhanced security.

3.11.4 Параметры считывания файлов

Клиент 61850 также обеспечивает возможность считывания файлов с устройства. Для считывания файлов с устройства задаются параметры:

- GF_Dir (обязательный параметр для считывания файлов) – каталог для считанных файлов с устройства с использованием макросов. В пути и имени файлов необходимо использовать следующие макросы:

- 1) [YYYY] – год;
- 2) [MM] – месяц;
- 3) [DD] – день;
- 4) [HH] – часы;
- 5) [NN] – минуты;
- 6) [ADR] – адрес устройства;
- 7) [FN] – имя файла на устройстве.

Пример – `<property name="GF_Dir" group="0" value="D:\OSC\[YYYY]_[MM]_[DD]_[HH]_[NN]BMRZ_[ADR]"/>`.

Расширение файла является таким же, как и в устройстве;

- GF_Dir2 – каталог для второй копии считанных файлов (резервный каталог). Макросы являются аналогичными. Данный параметр необязателен, и, если таковой не задан, файлы считываются в едином экземпляре;

Пример – `<property name="GF_Dir2" group="0" value="D:\OSC_COPY\[YYYY]_[MM]_[DD]_[HH]_[NN]BMRZ_[ADR]"/>`.

- GF_IEDDir (необязательный параметр) – адрес поиска файлов в устройстве.

Пример – `<property name="GF_IEDDir" group="0" value="/COMTRADE"/>`.

Если параметр не задан – поиск выполняется с корневого узла («NULL»);

- GF_Ext – параметр, задаваемый расширение файлов для поиска на сервере.

Пример – `<property name="GF_Ext" group="0" value="cfg"/>`.

Параметр необязателен, если таковой не задан – считываются все файлы;

- GF_TimeOut – период опроса устройства на наличие новых файлов, в миллисекундах. Минимальное значение 100 мс, максимальное – 100 000 000 мс. Необязательный параметр, по умолчанию задано 10 000 мс;

Пример вводимой строки – `<property name="GF_TimeOut" value="10000"/>`.

- GF_Start (необязательный параметр) – тег запуска считывания файла.

Пример – `<property name="GF_Start" type="TAG_OPC_DA" value="GF_Start" vartype="VT_BOOL"/>`.

Если данный параметр в процессе выполнения программы примет значение «1», то выполняется событийный цикл считывания файлов.

Для данного объекта задается следующий алгоритм считывания файлов:

1) в память ПО «Релематика МД» считывается список ранее считанных файлов из файла истории <каталог rmd.exe>\objdat\<имя объекта>.dat, если считать не удалось – значит список пуст;

2) если соединение с устройством доступно, запрашивается список файлов по параметру ="GF_IEDDir" (если данный параметр не задан, запрашивается весь список доступных файлов);

3) если список файлов на устройстве не пустой, файл истории очищается;

4) для каждой группы выполняется следующая последовательность действий:

a) запрашивается список файлов по параметру ="GF_IEDDir" (если данный параметр не задан, запрашивается весь список доступных файлов с устройства);

b) каждый файл из списка устройства проверяется на наличие в ранее считанном списке файла истории (в памяти ПО «Релематика МД»);

c) если файл входит в список скаченных файлов, то имя файла записывается в файл истории;

d) если файл не входит в список скаченных файлов, то считывается данный файл из терминала и записывается в папку с указанными макросами (если параметр «GF_Dir2» установлен, то записывается в две заданные папки). Затем имя файла записывается в файл истории;

5) ожидается таймаут «GF_TimeOut» и данный алгоритм переходит в пункт 1 для опроса устройства на наличие новых файлов. В случае, когда тег «GF_Start» принимает значение «1», данный период прерывается и запускается пункт 1 данного алгоритма.

3.12 Объект «SNMPv3_OBJ»

Объект «SNMPv3_OBJ» в ПО «Релематика МД» обеспечивает обмен данных по протоколу SNMP.

В файле конфигурации проекта объект «SNMPv3_OBJ» задается следующим образом:

```
<SNMPv3_OBJ name="SNMPCln_1">
  <property name="IPAddress" value="192.168.102.231"/>
  <property name="Port" value="161"/>
  <property name="TagUpdateRate" value="7000"/>
  <property name="NoAnswerCount" value="3"/>
  <property name="Version" value="3"/>
  <property name="Auth" value="1"/>
  <property name="UserName" value="admin"/>
  <property name="AuthAlgorithm" value="2"/>
  <property name="AuthPassword" value="{пароль в зашифрованном
виде}"/>
  <property name="PrivAlgorithm" value="4"/>
  <property name="PrivPassword" value="{пароль в зашифрованном
виде}"/>
  <property name="TrapsEnabled" value="0"/>
  <property name="ReadCommunity" value="public"/>
  <property name="WriteCommunity" value="private"/>
  <servicetag name="NotConnected" value="NotConnected"
vartype="VT_BOOL" sendto="OPCUASERVER\SNMPCln_1.NotConnected"/>
  <servicetag name="TagIU" value="IU" vartype="VT_BOOL"
sendto="OPCUASERVER\SNMPCln_1.TagIU"/>
  <tag name="Signal_1" value=".1.3.6.1.4.1.8691.7.6.1.10.2.0"
vartype="VT_I4" sendto="OPCUASERVER\SNMPCln_1.Signal_1"/>
  <tag name="Signal_2" trap="1" value=".1.3.6.1.4.1.8691.7.6.2.3"
vartype="VT_I4" sendto="OPCUASERVER\SNMPCln_1.Signal_2"/>
</SNMPv3_OBJ>.
```

При этом в объект «OPCUA_SERVER» добавить:

```
<OPCUA_SERVER name="OPCUASERVER">
  <tag name="SNMPCln_1.NotConnected"
value="SNMPCln_1.NotConnected" vartype="VT_BOOL"/>
  <tag name="SNMPCln_1.TagIU" value="SNMPCln_1.TagIU"
vartype="VT_BOOL" sendto="SNMPCln_1\TagIU"/>
  <tag name="SNMPCln_1.Signal_1" value="SNMPCln_1.Signal_1"
vartype="VT_I4"/>
  <tag name="SNMPCln_1.Signal_2" value="SNMPCln_1.Signal_2"
vartype="VT_I4"/>
</OPCUA_SERVER>.
```

Свойства объекта представлены в таблице 9.

Таблица 9 – Свойства объекта «SNMPv3_OBJ»

| Свойство | Описание |
|---------------|--|
| Auth | Способ аутентификации, возможные значения: 1) «noAuthNoPriv» или «1» (по умолчанию) – без аутентификации и шифрования, пароли передаются в открытом виде; 2) «authNoPriv» или «2» – аутентификация без шифрования траффика; 3) «authPriv» или «3» – аутентификация и шифрование, максимальный уровень защищенности. Обязательный параметр, если параметр «Version» = 3 |
| UserName | Имя пользователя, обязательный параметр, если «Version» = 3 |
| AuthAlgorithm | Алгоритм аутентификации, возможные значения: «MD5» (или «2») или «SHA» (или «3»). Данный параметр должен использоваться, если выбраны способы аутентификации «authNoPriv» или «authPriv». Обязательный параметр, если «Version» = 3 |
| AuthPassword | Пароль аутентификации, содержимое значения должно быть зашифровано. Данный параметр должен использоваться, если выбраны способы аутентификации «authNoPriv» или «authPriv». Обязательный параметр, если «Version» = 3 |
| PrivAlgorithm | Алгоритм шифрования, возможные значения: «AES128» (или «4») или «DES» (или «2»). Данный параметр должен использоваться, если выбран способ аутентификации «authPriv». Обязательный параметр, если «Version» = 3 |
| PrivPassword | Пароль шифрования, содержимое значения должно быть в закодированном виде. Данный параметр должен использоваться, если выбран способ аутентификации «authPriv». Обязательный параметр, если «Version» = 3 |
| TrapsEnabled | Обработка ловушек SNMP, возможные значения: «1» – включено, «0» – отключено. Значение по умолчанию «0», необязательный параметр, имеет значение, только если «Version» = 3 |
| IPAddress | Адрес опрашиваемого устройства, обязательный параметр |
| Port | Порт опрашиваемого устройства, необязательный параметр, значение по умолчанию – 161. Диапазон от 1 до 65535 |

| Свойство | Описание |
|----------------|---|
| TagUpdateRate | Период опроса значений в миллисекундах. Необязательный параметр, значение по умолчанию – 1000. Диапазон от 10 до 2147483647 |
| ReadCommunity | Группа для чтения, необязательный параметр, имеет значение, только если «Version» = 1 или 2, по умолчанию равен «public» |
| WriteCommunity | Группа для записи, необязательный параметр, имеет значение, только если «Version» = 1 или 2, по умолчанию равен «private» |
| NoAnswerCount | Свойство определяет количество неотвеченных запросов для выставления NotConnected = true. Необязательный параметр (по умолчанию = 3). Диапазон от 1 до 1000 |
| Version | Версия SNMP |

Сигналы задаются согласно синтаксису, описанному в 3.4. У сигналов (не служебных) данного объекта в атрибуте «value» задается OID согласно протолку SNMP (если значение начинается с символа «t», то данный OID является типа «trap»).

3.13 Объект «DirControl»

ПО «Релематика МД» обеспечивает ограничение объема занимаемого места на диске входных и выходных каталогов по времени хранения и/или суммарному размеру хранения данных. Для этого в программе создан объект «DirControl».

Объект «DirControl» обеспечивает выполнение следующих функций:

- возможность ограничить объем занимаемого места на диске входных и выходных каталогов по времени хранения и/или суммарному размеру хранения данных;
- контролирование до 10 каталогов;
- возможность проверки вложенных каталогов;
- проверка занимаемого места происходит через определенный интервал;
- удаление пустых каталогов;
- удаление всей группы COMTRADE файлов при удалении одного из них, т.е. файлы с одинаковым названием но с разными расширениями («.cfg», «.hdr», «.dat», «.inf»).

В файле конфигурации объект «DirControl» задается следующим образом:

```
<DIR_CONTROL name="DIRCONTROL">
  <property name="Dirs"
value="D:/Folder1//D:/Folder2//D:/Folder3"/>
  <property name="MaxSize" value="0"/>
  <property name="DaysOfStorage" value="0"/>
  <property name="CheckInterval" value="1"/>
  <property name="NestedDir" value="1"/>
</DIR_CONTROL>
```

Свойства объекта представлены в таблице 10.

Таблица 10 – Свойства объекта «DirControl»

| Свойство | Описание |
|---------------|--|
| Dirs | Список каталогов. Разделены знаком «//». Допустимо до 10 каталогов |
| MaxSize | Максимальный объем хранения (в ГБ). Диапазон от 0 до 100 000, значение по умолчанию 0 – не проводить проверку |
| DaysOfStorage | Время хранения (в днях). Диапазон временного интервала 0–10 000, значение по умолчанию 0 – не проводить проверку |
| CheckInterval | Интервал проверки каталогов (в часах). Диапазон 1–1000, значение по умолчанию 1 |

| Свойство | Описание |
|-----------|--|
| NestedDir | Параметр учета вложенных каталогов. Значение 0 или 1, по умолчанию = 1 |

3.14 Объект формирования единой осциллограммы «R58601»

Объект «R58601» реализован для формирования единой осциллограммы из считанных осциллограмм. По окончании считывания осциллограмм выполняется процедура их «склеивания». Процедура запускается по входному сигналу или периодически с заданным интервалом.

В файле конфигурации объект «R58601» задается следующим образом:

```
<R58601 name="R58601">
  <property name="InputDirNumber" value="1"/>
  <property name="InputDir1" value="D:/InputDir1"/>
  <property name="OutputDirs" value="D:/OutputDir1//
D:/OutputDir2//D:/OutputDir3//D:/OutputDir4"/>
  <property name="OutputFile" value="D:/OutputFile"/>
  <property name="DBConnStr" value="{строка подключения в
зашифрованном виде}"/>
  <property name="DBMaxFileSize" value="0"/>
  <property name="DBMaxStorageTime" value="0"/>
  <property name="DBCheckInterval" value="0"/>
  <property name="OscTimeRange" value="100"/>
  <property name="OscInputTimeout" value="60"/>
  <property name="OscWaitTimeout" value="10000"/>
  <property name="OscMaxDaysAge" value="0"/>
  <property name="Configuration" value="D:/58601Conf"/>
  <servicetag name="TagStartIN" value="StartIN"
vartype="VT_BOOL"/>
  <servicetag name="TagReadiness" value="Readiness"
vartype="VT_BOOL" sendto="OPCUASERVER\TagReadiness"/>
  <servicetag name="TagStatus" value="Status"
vartype="VT_LPSTR" sendto="OPCUASERVER\TagStatus"/>
</R58601>.
```

При этом в объект «OPCUA_SERVER» добавить:

```
<OPCUA_SERVER name="OPCUASERVER">
  <tag name="TagReadiness" value="Readiness" vartype="VT_BOOL"/>
  <tag name="TagStatus" value="Status" vartype="VT_LPSTR"/>

  <tag name="TagStartIN" value="StartIN" vartype="VT_BOOL"
sendto="R58601\TagStartIN"/>
</OPCUA_SERVER>.
```

Объект «R58601» имеет параметры, указанные в таблице 11.

Таблица 11 – Свойства объекта «R58601»

| Свойство | Описание |
|----------------|---|
| InputDirNumber | Количество входных каталогов. Диапазон от 1 до 1000 (обязательный параметр) |
| InputDir1 | Путь до входного каталога (обязательный параметр). В конец названия тега добавляется порядковый номер каталога. Максимальный номер каталога определен в свойстве «InputDirNumber» |

| Свойство | Описание |
|------------------|---|
| OutputDirs | Пути выходных каталогов (до 4 шт.) с возможностью использования шаблонов подстановки, указанных в приложении Г (обязательный параметр). Каталоги разделены знаком «//» |
| OutputFile | Имя и путь выходного файла с возможностью использования шаблонов подстановки, указанных в приложении Г (обязательный параметр) |
| DBConnStr | Строка подключения к БД «Postgre SQL» (обязательный параметр). Зашифрованная строка подключения содержит: 1) имя_пользователя; 2) пароль; 3) имя_базы_данных; 4) адрес_сервера; 5) порт. При отсутствии БД и необходимости ее создания в приложении ДД указан пример SQL-скрипта создания БД для объекта «R58601» |
| DBMaxFileSize | Максимальный объем (в Гб) хранения данных в БД. Диапазон максимального объема от 0 до 100 000. Опциональный параметр, значение по умолчанию: «0» – не проводить проверку |
| DBMaxStorageTime | Максимальное время (в днях) хранения данных в БД. Диапазон временного интервала от 0 до 10 000. Опциональный параметр, значение по умолчанию: «0» – не проводить проверку |
| DBCheckInterval | Интервал проверки БД (в часах). Диапазон от 0 до 1000. Опциональный параметр, значение по умолчанию: «0» – не проводить проверку |
| OscTimeRange | Время разбега осциллограмм (в миллисекундах) – Тосц. Диапазон от 0 до 1 000. Опциональный параметр, по умолчанию Тосц = 100. Если значение больше 1 000 – выставляется 1 000 (с записью ошибки в лог программы). Если значение Тосц < 0 – выставляется 100 (с записью ошибки в лог программы) |
| OscInputTimeout | Таймаут проверки входных данных (в секундах) – Твх. Диапазон от 0 до 90 000. Если Твх = 0, таймер не запускается. Опциональный параметр, значение по умолчанию: 1 минута (Твх = 60). Если значение больше 90 000 – выставляется 90 000 (с записью ошибки в лог программы). Если значение меньше 0 – выставляется 0 (с записью ошибки в лог программы) |
| OscWaitTimeout | Таймаут ожидания осциллограмм от других терминалов (в миллисекундах) – Тож. Диапазон от 0 до 60 000. Опциональный параметр, по умолчанию задано 10 секунд (Тож = 10 000). Если значение больше 60 000 – выставляется 60 000 (с записью ошибки в лог программы). Если значение меньше 0 – выставляется 10 000 (с записью ошибки в лог программы). Таймаут ожидания должен быть $\leq (Твх * 1000) / 2$ (исключение – таймаут проверки = 0). Если условие не выполняется Тож = $(Твх * 1000) / 2$ |
| OscMaxDaysAge | Допустимый возраст входящей осциллограммы (в днях). Диапазон от 0 до 1000. Опциональный параметр, значение по умолчанию: «0» – принимать все осциллограммы |
| Configuration | Файл конфигурации 58601 (генерируется МИКРА) (обязательный параметр) |

Объект не инициализируется, если неверно задан один из следующих параметров: InputDirNumber, InputDir1, OutputDirs, OutputFile, DBConnStr. При этом в лог программы выводится сообщение об ошибке.

Объект «R58601» имеет три служебных сигнала, описанные в таблице 12.

Таблица 12 – Служебные сигналы объекта «R58601»

| Сигнал | Тип | Описание |
|--------------|----------|---|
| TagStartIN | VT_BOOL | Сигнал о старте «склейки», принимается объектом от OPCUA_SERVER |
| TagReadiness | VT_BOOL | Сигнал готовности «склейки», отправляется в OPCUA_SERVER |
| TagStatus | VT_LPSTR | Сигнал результата «склеивания», отправляется в OPCUA_SERVER |

3.15 Объект «EventLog»

В ПО «Релематика МД» реализован функционал записи событий в базу данных. Для этого предусмотрен объект «EventLog». При изменении сигнала данный объект производит запись об этом в БД.

В файле конфигурации объект «EventLog» задается следующим образом:

```
<EVENT_LOG name="EVENTLOG">
  <property name="DBConnStr" value="{строка подключения в
зашифрованном виде}"/>
  <property name="DBQuery" value="INSERT INTO events (name,
value, vartype_id, timestamp, quality_id) VALUES ({name}, {value},
{vartype}, {timestamp}, {quality})"/>
  <tag name="InputTag" value="InputTag" vartype="VT_BOOL"/>
</EVENT_LOG>.
```

При этом в объект «OPCUA_SERVER» добавить:

```
<OPCUA_SERVER name="OPCUASERVER">
  <tag name="InputTag" value="InputTag" vartype="VT_BOOL"
typebuf="MEM" sendto="EVENTLOG\InputTag"/>
</OPCUA_SERVER>.
```

Объект «EventLog» имеет два обязательных параметра, указанные в таблице 13. Если неверно задано одно из свойств, то объект не инициализируется.

Таблица 13 – Свойства объекта «EventLog»

| Свойство | Описание |
|-----------|--|
| DBConnStr | Строка подключения к БД «Postgre SQL». Зашифрованная строка подключения содержит: 1) имя_пользователя; 2) пароль; 3) имя_базы_данных; 4) адрес_сервера; 5) порт |
| DBQuery | Строка с запросом на добавление данных в БД с использованием подстановки данных сигнала согласно шаблону, указанному в приложении Е |

В вышеуказанном примере объекта «EventLog» изменения сигнала «InputTag» (<tag name="InputTag" value="InputTag" vartype="VT_BOOL"/>) записываются в БД, указанной в параметре «DBConnStr», согласно запросу «DBQuery».

Если БД не имеется и необходимо создать ее, то в приложении Ж указан пример SQL-скрипта создания БД для объекта «EventLog».

3.16 Объект «Mux»

В ПО «Релематика МД» реализован функционал объединения группы из трех входящих сигналов в один. Для этого предусмотрен объект «Mux». Выходной сигнал составляется из следующих сигналов: значение (value), временная метка (datetimestamp), качество (quality). Результирующий сигнал может отправляться в другие объекты.

В файле конфигурации объект «Mux» задается следующим образом:

```
<MUX_OBJ name="MUX_1">
  <property name="TimeOutInput" value="100"/>
  <tag name="Signal1" value="out" group="1" vartype="VT_I4"
sendto="OPC\MUX_1.Signal1"/>
  <tag name="Signal1_val" value="val" group="1" vartype="VT_I4"/>
  <tag name="Signal1_qual" value="qual" group="1" vartype="VT_I4"/>
  <tag name="Signal1_ts" value="ts" group="1" vartype="VT_I8"/>
  <tag name="Signal2" value="out" group="2" vartype="VT_BOOL"
sendto="OPC\MUX_1.Signal2"/>
  <tag name="Signal2_val" value="val" group="2" vartype="VT_BOOL"/>
  <tag name="Signal2_qual" value="qual" group="2" vartype="VT_I4"/>
  <tag name="Signal2_ts" value="ts" group="2" vartype="VT_I8"/>
</MUX_OBJ>.
```

При этом в объект «OPCUA_SERVER» добавить:

```
<OPCUA_SERVER name="OPC" host="">
  <tag name="MUX_1.Signal1" value="MUX_1.Signal1" vartype="VT_I4"/>
  <tag name="MUX_1.Signal1_val" value="MUX_1.Signal1_val"
vartype="VT_I4" sendto="MUX_1\Signal1_val\1"/>
  <tag name="MUX_1.Signal1_qual" value="MUX_1.Signal1_qual"
vartype="VT_I4" sendto="MUX_1\Signal1_qual\1"/>
  <tag name="MUX_1.Signal1_ts" value="MUX_1.Signal1_ts"
vartype="VT_I8" sendto="MUX_1\Signal1_ts\1"/>
  <tag name="MUX_1.Signal2" value="MUX_1.Signal2" vartype="VT_BOOL"/>
  <tag name="MUX_1.Signal2_val" value="MUX_1.Signal2_val"
vartype="VT_BOOL" sendto="MUX_1\Signal2_val\2"/>
  <tag name="MUX_1.Signal2_qual" value="MUX_1.Signal2_qual"
vartype="VT_I4" sendto="MUX_1\Signal2_qual\2"/>
  <tag name="MUX_1.Signal2_ts" value="MUX_1.Signal2_ts"
vartype="VT_I8" sendto="MUX_1\Signal2_ts\2"/>
</OPCUA_SERVER>.
```

Объект «Mux» имеет один опциональный параметр:

- TimeOutInput – время ожидания входящих сигналов группы после получения одного из них, перед их объединением. Значение по умолчанию – 100 мс.

Сигналы задаются согласно синтаксису, описанному в 3.4.

Выходной сигнал объединяется из трех сигналов одной группы (таблица 14).

Таблица 14 – Таблица сигналов объекта «Mux»

| Тип сигнала | name | value | vartype | group |
|-----------------|--------------|-------|--------------------------------------|--------------------------------|
| Выходной сигнал | Имя сигнала. | out | Соответствует типу выходного сигнала | Номер группы выходного сигнала |
| Значение | | val | Соответствует типу выходного сигнала | |
| Качество | | qual | VT_I4 | |
| Метка времени | | ts | VT_I8 | |

3.17 Объект «Demux»

Объект Demux разбирает входящие сигналы на два составляющих сигнала: временная метка (timestamp), качество (quality). Составляющие сигналы могут отправляться в другие объекты.

В файле конфигурации объект «Demux» задается следующим образом:

```
<DEMUX_OBJ name="DEMUX_1">
  <tag name="Signal1" value="in" group="1" vartype="VT_I4"/>
  <tag name="Signal1_qual" value="qual" group="1" vartype="VT_I4"
sendto="OPC\DEMUX_1.Signal1_qual"/>
  <tag name="Signal1_ts" value="ts" group="1" vartype="VT_I8"
sendto="OPC\DEMUX_1.Signal1_ts"/>
  <tag name="Signal2" value="in" group="2" vartype="VT_BOOL"/>
  <tag name="Signal2_qual" value="qual" group="2" vartype="VT_I4"
sendto="OPC\DEMUX_1.Signal2_qual"/>
  <tag name="Signal2_ts" value="ts" group="2" vartype="VT_I8"
sendto="OPC\DEMUX_1.Signal2_ts"/>
</DEMUX_OBJ>.
```

В файл конфигурации объекта OPCUA_SERVER добавить:

```
<OPCUA_SERVER name="OPC" host="">
  <tag name="Signal1" value="Signal1" vartype="VT_I4"
sendto="DEMUX_1\Signal1\1"/>
  <tag name="Signal2" value="Signal2" vartype="VT_I8"
sendto="DEMUX_1\Signal2\2"/>
  <tag name="DEMUX_1.Signal1_qual" value="DEMUX_1.Signal1_qual"
vartype="VT_I4"/>
  <tag name="DEMUX_1.Signal1_ts" value="DEMUX_1.Signal1_ts"
vartype="VT_I8"/>
  <tag name="DEMUX_1.Signal2_qual" value="DEMUX_1.Signal2_qual"
vartype="VT_I4"/>
  <tag name="DEMUX_1.Signal2_ts" value="DEMUX_1.Signal2_ts"
vartype="VT_I8"/>
</OPCUA_SERVER>.
```

Сигналы задаются согласно синтаксису, описанному в 3.4.

Входной сигнал разбивается на два сигнала одной группы (таблица 15).

Таблица 15 – Таблица сигналов объекта «Demux»

| Тип сигнала | name | value | vartype | group |
|----------------|-------------|-------|-------------------------------------|-------------------------------|
| Входной сигнал | Имя сигнала | in | Соответствует типу входного сигнала | Номер группы входного сигнала |
| Качество | | qual | VT_I4 | |
| Метка времени | | ts | VT_I8 | |

3.18 Объект «Modbus Slave»

В программе реализован функционал передачи данных по протоколу Modbus (TCP). Для этого разработан объект «Modbus Slave».

Объект «Modbus Slave» обеспечивает выполнение следующих функций:

- передача данных по протоколу Modbus-TCP;
- поддержка стандартных функций протокола Modbus: 1, 2, 3, 4, 5, 6, 16;

- отдельное адресное пространство для регистровых функций (3, 4, 6, 16) и для битовых функций (1, 2, 5), пул адресов от 0x0000 до 0xFFFF;
- массив запрашиваемых элементов данных может содержать «разрывы» в адресном пространстве, но размер всего запрашиваемого пакета вместе со служебной информацией не превышает 253 байта. Незаданные в конфигурации элементы данных имеют значение «0»;
- порядок битов при запросе битовых данных – прямой;
- порядок байт в регистрах Modbus – конфигурируемый;
- поддерживаемые типы данных Modbus: INT16, INT32, FLOAT32, DOUBLE64, BIT;
- каждый элемент данных, размещенный в пуле адресов, приращивает значение адреса на величину своего размера, минимальный размер элемента данных 2 байта (1 регистр Modbus) – для регистровых функций 3,4,6,16; если элемент данных битовый, количество передаваемых бит выравнивается по 1 байту;
- широковещательные запросы не поддерживаются;
- запросы на синхронизацию времени не поддерживаются;
- адреса объекта лежат в диапазоне от 1 до 255;
- при некорректном запросе объект возвращает по протоколу стандартный код ошибки, соответствующий ситуации;
- в объекте зарезервирован сигнал NotConnected для определения наличия подключения мастера Modbus;
- поддерживает возможность указывать коэффициент для аналоговых сигналов;
- согласно стандарту, для записи доступны только Coils (регистры флагов и выходы, функция чтения – 1, функция записи – 5) и Holding Registers (регистры, предназначенные для хранения настроек, функция чтения – 3, функция записи – 6, 16). Если для сигнала задана функция 2 или 4, то запись сигнала по протоколу Modbus будет НЕВОЗМОЖНА.

Пример объекта:

```
<MODBUS_SLAVE name="ModbusSlave" host="">
  <property name="Mode" value="TCP"/>
  <property name="IPAddress" value="127.0.0.1"/>
  <property name="Port" value="502"/>
  <property name="ModbusAddress" value="1"/>

  <servicetag name="NotConnected" value="NotConnected"
vartype="VT_BOOL" sendto="OPC\NotConnected"/>
  <tag name="BIT_value1" value="100" function = "1" vartype="VT_BOOL"
sendto="OPC\Bit_value1"/>
  <tag name="BIT_value2" value="101" function = "1" vartype="VT_BOOL"
sendto="OPC\Bit_value2"/>
  <tag name="REG_value1_INT16" value="100" function = "3"
vartype="VT_I2" sendto="OPC\REG_value1"/>
  <tag name="REG_value2_INT32" value="101" function = "3"
vartype="VT_I4" sendto="OPC\REG_value2"/>
  <tag name="REG_value3_FLOAT32" value="103" function = "3"
vartype="VT_R4" sendto="OPC\REG_value3"/>
  <tag name="REG_value4_DOUBLE64" value="105" function = "3"
vartype="VT_R8" k="0.1" sendto="OPC\REG_value4"/>
  <tag name="REG_value6_INT16" value="0x73" function = "3"
vartype="VT_I2" sendto="OPC\REG_value6"/>
</MODBUS_SLAVE>.
```

При этом в объект «OPC_DA_SERVER» добавить:

```
<OPCUA_SERVER name="OPC "host="">
  <tag name="NotConnected" type="TAG OPC_DA_SERVER"
value="NotConnected" vartype="VT_BOOL"/>
  <tag name="BIT_value1" value="BIT_value1" vartype="VT_BOOL"/>
```

```

<tag name="BIT_value2" value="BIT_value2" vartype="VT_BOOL"/>
<tag name="REG_value1" value="REG_value1" vartype="VT_I2"/>
<tag name="REG_value2" value="REG_value2" vartype="VT_I4"/>
<tag name="REG_value3" value="REG_value3" vartype="VT_R4"/>
<tag name="REG_value4" value="REG_value4" vartype="VT_R8"/>
<tag name="REG_value6" value="REG_value6" vartype="VT_I2"/>
</OPCUA_SERVER>.

```

Объект «Modbus Slave» имеет свойства, описанные в таблице 16 .

Таблица 16 – Свойства объекта «Modbus Slave»

| Свойство | Описание |
|---------------|--|
| Mode | Режим работы. Возможные значения: TCP (в первом релизе – только TCP) |
| ModbusAddress | Modbus-адрес объекта |
| IPAddress | IP-адрес объекта |
| Port | Порт, ожидающий подключения мастера |

Сигналы объекта задаются атрибутами, описанными в таблице 17.

Таблица 17 – Атрибуты сигналов объекта «Modbus Slave»

| Атрибут | Описание |
|----------|--|
| name | Имя сигнала |
| value | Адрес сигнала, например «109», допускается 16-ричное представление: 0xNNNN |
| function | Функция Modbus для чтения, функции 1 и 2, 3 и 4 не равнозначны(!), также этот параметр задает тип регистра Modbus: <ul style="list-style-type: none"> – f=1 Coils, чтение и запись; – f=2 Discrete Inputs, только чтение; – f=3 Holding Registers, чтение и запись; – f=4 Input Registers, только чтение |
| vartype | Параметр типа данных Modbus, возможные значения: <ul style="list-style-type: none"> – VT_BOOL – для битов; – VT_I2 – INT16, диапазон -32768..32767; – VT_I4 – INT32 диапазон -2 147 483 648.. 2 147 483 647; – VT_R4 – FLOAT32 диапазон 3,4E +/- 38; – VT_R8 – DOUBLE64 диапазон 1,7E +/- 308 |
| k | Коэффициент. Не допустимо значение 0 |

3.19 Объект «MODBUS_CLIENT»

Для опроса устройств по протоколу Modbus (TCP) разработан объект «MODBUS_CLIENT».

Объект «MODBUS_CLIENT» строится в виде иерархического дерева, состоящего из линий, станций и сигналов (рисунок 21).

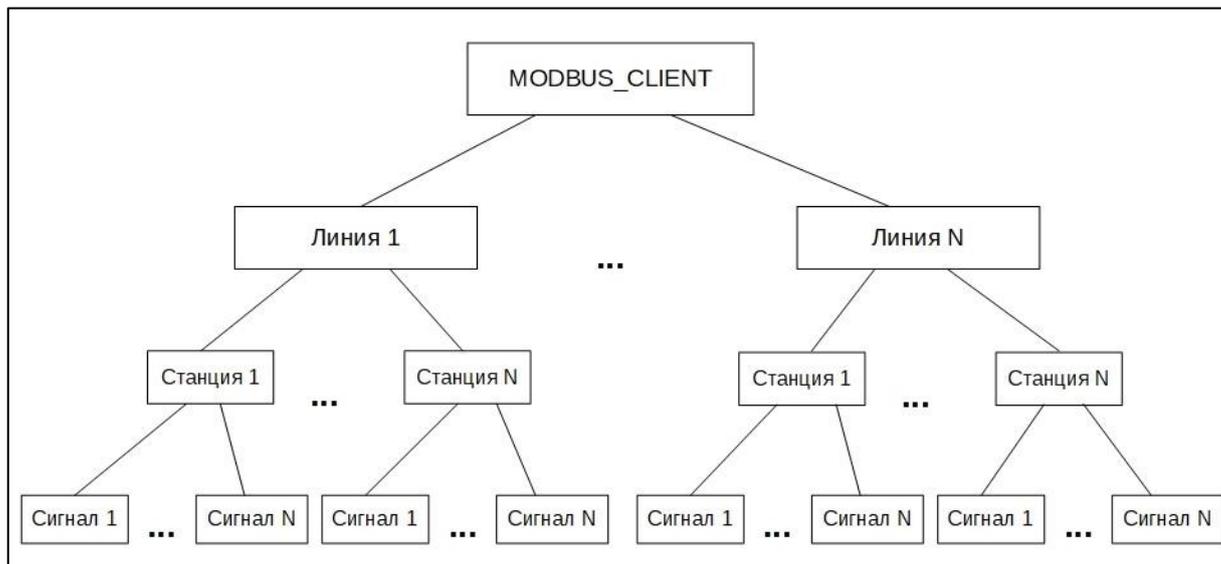


Рисунок 21 – Объектная модель MODBUS клиента

Объект необходим для сбора данных (аналоговых и дискретных) и управления оборудованием, подключенным к последовательным асинхронным портам ввода/вывода (COM-портам) или TCP/IP-портам через преобразователи полевой шины по протоколу Modbus.

Линия – объект, описывающий последовательный асинхронный порт ввода/вывода (COM-порт) или TCP/IP-порт и содержащий дочерние объекты – станции.

Станция – адресуемый Modbus-модуль (устройство).

Сигнал – объект, описывающий сигнал или группу сигналов.

Пример объекта «MODBUS_CLIENT» указан в приложении И.

Для конфигурирования объекта «MODBUS_CLIENT» задается следующий служебный параметр:

– IsActiveStart – тэг определения активности объекта при загрузке программы. При запуске программы данный параметр может принимать значение «0» или «1».

Пример строки – `<property name="IsActiveStart" value="0/1"/>`.

При значении «1» объект активен и принимает данные, при значении «0» – обрывает входную информацию и приводит объект в не активное состояние.

Данный объект имеет и служебный сигнал:

– IsActiveTag – тег активности объекта (тип «VT_BOOL»). У данного тега важен его атрибут «value», в котором задается значение «IsActive», что означает, что когда он имеет значение «1», то объект активен.

Пример строки – `<tag name="IsActiveTag" value="IsActive" vartype="VT_BOOL"/>`.

Объект имеет следующие теги:

– ServerOFF – режим работы объекта (значение «0»/«1», тип «VT_BOOL»). При записи в данный тег значения «1» все линии объекта отключаются, и опрос всего оборудования прекращается. При записи значения «0» все линии включаются. Данный параметр доступен и для чтения, и для записи (далее доступ – RW);

– ServerID – строковый тег, содержащий имя компьютера, на котором запущен объект, тип «VT_BSTR» (доступен только для чтения, далее доступ – R);

– ServerALIVE – тег, определяющий возможность вывода информации о событиях “EVENTS” в специальные теги, тип «VT_I4». Проверяется один раз в секунду. Если значение тега равно нулю, события будут считываться из устройства в буфер, но выводиться не будут. Для вывода событий необходимо увеличивать значение тега циклически не реже, чем один раз в секунду. Данный параметр предназначен только для устройств SEPAМ, БМРЗ, БМПА, БМПС, СИРИУС (старого типа), СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС, MiCOM P123, MiCOM P124, MiCOM P127. Доступ – RW;

– ServerLogLevel – текущий уровень лога, тип «VT_I4». Возможные значения: «1» – обычный уровень и «2» – отладочный уровень. Все остальные значения игнорируются. Доступ – RW.

Для настройки линий (следующая ступень иерархии) для объекта предусмотрены следующие служебные параметры:

– IU – тег использования линии (значение «0»/«1», тип «VT_BOOL»). При записи в данный тег значения «0» линия объекта отключается, и опрос всего оборудования на линии прекращается. При записи «1» линия включается. Начальное значение тега может быть задано в конфигурации объекта «MODBUS_DLL_CONFIG» (3.19.1). Доступ – RW;

– NotConnected – тег наличия связи с устройствами на линии (значение «0»/«1», тип «VT_BOOL»). Если хотя бы одно устройство на линии отвечает на запросы, значение данного тега равно «0», иначе – «1». Если линия отключена от опроса, то значение всегда «0». Доступ – R;

– LineHealth – тег работоспособности линии, тип «VT_I4». Значение тега прибавляется объектом на один каждую секунду (инкремент). Минимальное значение тега «0», максимальное – «65535». После достижения максимального значения параметр сбрасывается на «0» и отсчет начинается сначала. Доступ – R. По умолчанию данный параметр отключен. Для его включения используется параметр конфигурации объекта «LineHealthOFF» = «0», который задается в объекте «MODBUS_DLL_CONFIG»;

– LineTimeCorrection – значение параметра TSCR (коррекции времени), тип «VT_I4». Максимальное значение «1000», минимальное «0». Доступ – RW;

– LineStat – время опроса всех устройств на линии в миллисекундах, тип «VT_I4». Доступ – R.

Для настройки станций на линии в объекте предусмотрены следующие служебные теги:

– IU – тег использования станции (значение «0»/«1», тип «VT_BOOL»). При записи в данный тег значения «0» станция отключается, и опрос оборудования прекращается. При записи «1» – станция включается. Начальное значение тега может быть задано в конфигурации объекта «MODBUS_DLL_CONFIG». При значении параметра «ServerOFF» равно «1» – изменять значение тега «IU» станции в клиенте невозможно, об этом будет выдаваться сообщение в теге «REQ_RESULT». Доступ тега – RW;

– NotConnected – тег наличия связи с устройством (значение «0»/«1», тип «VT_BOOL»). Если устройство отвечает на запросы, значение данного тега равно «0», иначе – «1». Если станция отключена от опроса, то значение всегда «0». Доступ – R;

– TSCmd – тег команды на синхронизацию времени (значение «0»/«1», тип «VT_BOOL»). При записи в данный тег значения «1» станции посылается запрос на синхронизацию времени. После выполнения команды тег принимает значение «0». Доступ – RW;

– REQ_RESULT – тег результата (статуса) команды (запроса), тип «VT_BSTR». Может принимать различные значения с описанием ошибки или статуса запроса при обмене данными с устройством. Доступ – R;

– OSCStatus – статус считывания осциллограмм, тип «VT_BSTR». Создается только для устройств, имеющих функцию записи и хранения осциллограмм. Содержит краткую информацию об осциллограммах в устройстве и статусе считывания:

- 1) Q (Quantity) – количество осциллограмм в устройстве;
- 2) DLN (Downloading Now) – номер считываемой в данный момент осциллограммы;
- 3) PRGS (Progress) – прогресс скачивания осциллограммы в процентах (допустимые значения: N/A – не доступный, AllDone – все осциллограммы скачаны);
- 4) DT (DateTIme) – дата-время всех осциллограмм в устройстве.

Данный тег создается лишь в том случае, если в конфигурации не запрещено автоматическое считывание осциллограмм. Доступ – R;

– OSCAUTO – статус автоматического считывания осциллограмм (значение «0»/«1», тип «VT_BOOL»). Создается только для устройств, имеющих функцию записи и хранения осциллограмм. Если значение тега равно «1», то автоматическое считывание включено, иначе – выключено. Доступ – RW;

– OSCREFRESHLIST – тег для подачи команды на обновление списка осциллограмм в устройстве (имеет тип «VT_BOOL» для обычных устройств, для устройств типа 150, 747 и 1000 имеет тип «VT_I4»). Создается только для устройств, имеющих функцию записи и хранения осциллограмм. Для подачи команды требуется записать в тег значение «1». После исполнения команды тег принимает нулевое значение. Доступ – RW;

– OSCTODL – тег для подачи команды на скачивание с устройства осциллограммы с определенным номером, тип «VT_I4». Создается только для устройств, имеющих функцию записи и хранения осциллограмм. Номер осциллограммы, записываемый в тег, не должен быть ≤ 0 и не должен быть больше количества осциллограмм, имеющихся на данный момент в устройстве. В случае, если в текущий момент времени происходит автоматическое скачивание, то оно прерывается. После выполнения команды, если значение параметра «OSCAUTO» равно «1», автоматическое скачивание возобновится. Доступ – RW;

– Commands.ReadZoneX – командный тег (тип «VT_I4») для опроса зоны по запросу, где X – это номер зоны. Необходимо записать в тег значение «1».

При конфигурировании станций в зависимости от производителей устройств Modbus и их типов дополнительно предусматриваются служебные параметры, приведенные в приложении КИИ.

Объект «MODBUS_CLIENT» имеет следующие характеристики:

- максимальное количество линий – 255;
- максимальное количество станций на линии – 32;
- общее максимальное количество точек ввода-вывода – 65535.

При этом поддерживает следующие устройства:

- устройства производства НТЦ «Механотроника»:
 - 1) БМРЗ – 0,4АВ, БМРЗ-04-01АВ;
 - 2) БМРЗ – 0,4ВВ, БМРЗ-04-01ВВ;
 - 3) БМПА – 0,4, БМПА-04-01;
 - 4) БМЦС;
 - 5) БМПА, БМРЗ, БМЦС со свободно-конфигурируемой картой адресов;
- устройства производства «Schneider Electric»:
 - 1) Sepam – 2000;
 - 2) Sepam – 80;
 - 3) Sepam – 40;
 - 4) Modicon;
 - 5) MiCOM P123;
 - 6) MiCOM P124;
 - 7) MiCOM P127;
- устройства производства «Радиус»:
 - 1) Сириус-Л;
 - 2) Сириус-2Л;
 - 3) Сириус-2-В v3;
 - 4) Сириус-2-С v3;
 - 5) Сириус-2-Л v3;
 - 6) Сириус-ГС v3;
 - 7) Сириус v2 (старые);
- устройства производства ООО «Релематика»:
 - 1) TOP 200-16;
 - 2) TOP 300;
- другие устройства, поддерживающие протокол Modbus RTU и ModbusTCP.

Вышеперечисленные устройства делятся на «обычные» устройства и на устройства со свободно-конфигурируемой картой адресов. «Обычными» устройствами являются стандартные терминалы, у которых жестко задаются адреса сигналов и команд. Адреса сигналов и команд данных терминалов описываются в шаблонах в формате «*.xml», а все шаблоны хранятся в

едином файле «Templates.xml». В данном файле каждое устройство имеет уникальный цифровой идентификатор типа (Type). Устройствами со свободно-конфигурируемой картой адресов являются терминалы производства ООО «Механотроника» (DeviceType = 150), иные устройства, поддерживающие протокол MODBUS (DeviceType = 747), терминалы производства ООО «Релематика» TOP 200-16 и TOP 300 (DeviceType = 1000).

Дополнительно для конфигурации объекта «MODBUS_CLIENT» в файле проекта ПО «Релематика МД» создается объект «MODBUS_DLL_CONFIG». Данный объект содержит внешние настройки объекта «MODBUS_CLIENT», сети, устройств, с которыми объект должен обмениваться данными, а также в нем прописываются настройки опрашиваемых зон устройств.

3.19.1 Конфигурация объекта «MODBUS_DLL_CONFIG»

Пример объекта «MODBUS_DLL_CONFIG»:

```
<MODBUS_DLL_CONFIG Name="ModBus" ServerOFF="0" CheckKeyTimeout="60"
LogLevel="1" LineHealthOFF="0">
  <Lines Name="Линии связи">
    <Line Name="Line" TCP="1" IPAddr="10.10.5.57" Port="4001"
InUse="1" OffTagInUse="0" MaxTimeout="1000" RetryLimit="2" TimeSynchro="0"
TimeSynchroPeriod="" ModbusTCP="0" TSCR="0">
      <Station Name="BEMPRU-TL2BKZ" InUse="1" ModbusAddr="1"
DeviceType="3002" DACDelay="200" ZoneGetWrite="1" SETS="1"
SetsCheckEnable="1" SetsCheckPeriod="1" INFO="1" OSCDL="1"
OSCPATH="C:\OSC\BEMPRU-TL2BKZ" EVENTS="1" EAUTOACK="1" EVTimeFromPC="1">
        <Tags>
          <Tag RegularBegin="54" RegularSize="1"
RegularFreq="1000"/>
          <Tag RegularBegin="31" RegularSize="1"
RegularFreq="1000"/>
          <Tag RegularBegin="30" RegularSize="1"
RegularFreq="1000"/>
        </Tags>
      </Station>
    </Line>
  </Lines>
</MODBUS_DLL_CONFIG>
```

Для настройки объекта задаются следующие параметры (подробное описание параметров конфигурации приведено в таблице Л.1 приложения Л):

- Name – имя объекта латиницей (должно быть уникальным), которое будет составной частью имени OPC-тега. Обязательный параметр;
- ServerOFF – отключение объекта (значение «0»/«1»). При значении «1» опрос всех устройств прекращается. Определяет начальное значение тега «ServerOFF» (параметр конфигурирования объекта «MODBUS_CLIENT»). Необязательный параметр, значение по умолчанию «1» (отключен);
- CheckKeyTimeout – период опроса наличия ключа GUARDANT в секундах. Опциональный параметр, значение по умолчанию «60». Диапазон допустимых значений: 5...600;
- LogLevel – включение/отключение расширенного/отладочного лога при первоначальном запуске (значение «1»/«2»). Значение «1» – обычный лог, значение «2» – расширенный/отладочный лог. Необязательный параметр, при его отсутствии задается обычный режим логирования;
- LineHealthOFF – флаг включения тега «LineHealth». Флаг имеет значения «0»/«1», где значение «0» означает, что тег «LineHealth» включен, а значение «1» – тег «LineHealth» отключен. Параметр необязательный и по умолчанию имеет значение «1».

В объекте также содержатся все линии устройств. Для линий устройств создаются специальные блоки <Lines> и <Line>.

Блок <Lines> представляет секцию линий связи, количество которых соответствует количеству реальных физических линий связи с терминалами.

Пример:

```
<Lines Name="">
  <Line ... >
  </Line>
</Lines>.
```

Блок <Lines> имеет лишь один параметр: «Name» – имя секции линий связи. Необязательный параметр, по умолчанию задается имя «Линии связи».

Блок <Line> представляет настройки одной определенной линии связи.

Пример:

```
<Line Name="" TCP="" ModbusTCP="" IPAddr="" IPAddrReserve="" Port=""
ComNumber="" BaudRate="" DataBits="" StopBits="" Parity="" InUse=""
OffTagInUse="" MaxTimeout="" RetryLimit="" TimeSynchro=""
TimeSynchroPeriod="" TSCR="">
  <Station ...>
  </Station>
</Line>.
```

В данном блоке задаются следующие параметры линии связи:

- Name – имя линии латиницей (должно быть уникальным), которое будет составной частью имени OPC-тега. Обязательный параметр;

- TCP – параметр для настройки типа обмена. Значение «1» – обмен по протоколу TCP. В случае отсутствия параметра или значения параметра, отличного от «1», параметры настройки TCP обмена игнорируются, и объект будет пытаться обмениваться данными по последовательному порту. Если значение равно «1», то параметры настройки последовательного порта указывать необязательно (могут отсутствовать);

- ModbusTCP – обмен с устройством по протоколу Modbus TCP (значение «0»/«1»). Значение «1» – обмен с устройством будет осуществляться по протоколу Modbus TCP, значение «0» – обмен будет осуществляться по протоколу Modbus RTU. Опциональный параметр, по умолчанию, задано значение «0»;

- IPAddr – основной IP-адрес опрашиваемого устройства (преобразователя интерфейсов). Обязательный параметр. Если параметр «TCP» равен «0», то данный параметр игнорируется. Если параметр «IPAddrReserve» = «1», то объект пытается подключиться по основному IP-адресу три раза. Если безрезультатно, пытается подключиться к резервному IP-адресу. Иначе, в случае, если «IPAddrReserve» = «0» или если данный параметр отсутствует, объект пытается подключиться к опрашиваемому устройству до тех пор, пока не подключится или пока объект не остановят. Период попыток подключения составляет 10 с;

- IPAddrReserve – резервный IP-адрес опрашиваемого устройства (преобразователя интерфейсов). Необязательный параметр и может отсутствовать. Если параметр «TCP» равен «0», то данный параметр игнорируется. Если параметр «IPAddrReserve» равен «1», то объект пытается подключиться к опрашиваемому устройству по резервному IP-адресу три раза. Если безрезультатно, то пытается подключиться по основному IP-адресу три раза. И так будет продолжаться до тех пор, пока не подключится или пока объект не остановят. Период попыток подключения составляет 10 с;

- Port – IP-порт опрашиваемого устройства (преобразователя интерфейсов). Обязательный параметр. Если параметр «TCP» равен «0», то данный параметр игнорируется;

- ComNumber – номер последовательного порта, существующего в системе. Обязательный параметр. Если параметр «TCP» равен «1», то данный параметр игнорируется;

- BaudRate – скорость последовательного порта. Обязательный параметр. Если параметр «TCP» равен «1», то данный параметр игнорируется;

- DataBits – число бит данных. Обязательный параметр. Если параметр «TCP» равен «1», то данный параметр игнорируется;

- StopBits – число стоповых битов. Обязательный параметр. Если параметр «TCP» равен «1», то данный параметр игнорируется;

- Parity – четность. Может принимать значения: «0» – без проверки четности (none), «1» – нечет (odd) и «2» – чёт (even). Обязательный параметр. Если параметр «TCP» равен «1», то данный параметр игнорируется;
- InUse – флаг опроса всей линии при старте объекта (значение «0»/«1», где значение «0» – не опрашивать, значение «1» – в опросе). Обязательный параметр;
- OffTagInUse – флаг перехода тега «InUse» в невидимый режим (значение «0»/«1»). Значение «1» – параметр «InUse» невидимый тег, значение «0» – видимый. Необязательный параметр, и если данный параметр отсутствует, то тег «InUse» видимый;
- MaxTimeout – таймаут ожидания ответа от устройства в миллисекундах. Обязательный параметр;
- RetryLimit – количество повторной отправки того же самого запроса при ошибке. Обязательный параметр;
- TimeSynchro – флаг синхронизации времени на устройстве (значение «0»/«1»). Значение «1» – синхронизировать время на устройстве, значение «0» – не синхронизировать. Обязательный параметр;
- TimeSynchroPeriod – период синхронизации времени на устройстве в миллисекундах. Опциональный параметр, по умолчанию имеет значение «60000». Если параметр «TimeSynchro» не равен «1», то данный параметр игнорируется;
- TSCR – значение коррекции времени в миллисекундах. Прибавляется к значению времени в пакете синхронизации. Необязательный параметр, по умолчанию имеет значение «0».

В блоке <Line> кроме настроек линии связи содержатся блоки <Station>. Блок <Station> представляет настройки связи конкретного физического устройства, а также параметры для настройки специализированных устройств (терминалов).

Пример:

```
<Station Name="" InUse="" ModbusAddr="" DeviceType="" WR_Mode=""
SessionQuota="" EVENTS="" EAUTOACK="" EFREQ="" EUPDT="" EVTimeFromPC=""
SETS="" PSW="" INFO="" IFREQ="" OSCPath="" CMDID="" OSCDL="" DACDelay=""
ZoneGetWrite="">
  <Tags>
  </Tags>
</Station>.
```

Количество секций <Station> соответствует количеству устройств на физической линии связи, описанной в секции <Line>. Блок <Station> имеет следующие параметры:

- Name – имя устройства латиницей (должно быть уникальным), которое будет составной частью имени OPC-тега. Обязательный параметр;
- InUse – флаг опроса устройства при старте объекта (значение «0»/«1»). Значение «1» – в опросе, «0» – не опрашивать. Обязательный параметр;
- ModbusAddr – modbus адрес устройства. Обязательный параметр;
- DeviceType – уникальный цифровой идентификатор типа устройства. Обязательный параметр. Значение данного параметра соответствует значению параметра «Type» блока «Unit» файла «Templates.xml»;
- DeviceName – уникальный строковый идентификатор типа устройства. Необязательный параметр в случае, когда присутствует параметр «DeviceType»;
- WR_Mode – флаг включения режима записи значений в теги для устройств, поддерживающих протокол Modbus RTU и ModbusTCP (значение «0»/«1», где значение «0» – отключен, значение «1» – включен). Необязательный параметр. Необходимо использовать в сочетании с параметром блока <Tag> (блок параметров одного запроса к устройству) «DirectAccess» = «1», а также параметром зоны «FuncWR» = «x» в шаблоне устройств (файл «Templates.xml»), где «x» – номер функции для записи (5, 6, 15, 16);
- SessionQuota – квота количества запросов к одному терминалу за сессию. Сессия терминала – это период времени, когда идет опрос определенного устройства на линии. Необязательный параметр. Значение по умолчанию – 2 (рекомендуется).

ВНИМАНИЕ!!! Изменение данного параметра существенно влияет на скорость опроса всех терминалов на линии;

- EVENTS – флаг работы с журналом событий (значение «0»/«1»). Значение «1» разрешает работу с журналом событий терминала. Необязательный параметр, по умолчанию задано значение «0»;
- EAUTOACK – флаг работы в режиме автоквитирования (значение «0»/«1»). Значение «1» разрешает работу в режиме автоквитирования (считал, значит квитировал событие). Значение «0» переключает считывание событий терминала в режим «ручного» квитирования (считал, сохранил на жесткий диск, отквитировал). Необязательный параметр и по умолчанию задано значение «0». **Данный параметр предназначен только для устройств MiCOM P123, MiCOM P124, MiCOM P127;**
- EFREQ – частота запросов к журналу событий устройства в миллисекундах. Необязательный параметр, и по умолчанию равен «1000»;
- EUPDT – частота обновления событийных тегов в миллисекундах. Необязательный параметр, и по умолчанию равен «1000»;
- EVTimeFromPC – флаг обновления тега «EVTime» (приложение **Ошибка! Источник ссылки не найден.**) из ПК (значение «0»/«1»). При значении «1» время обновления тега «EVTime» выставляется из ПК. Необязательный параметр и по умолчанию равен «0»;
- SETS – флаг вывода уставок устройства в OPC-теги в группе «SETS» (значение «0»/«1»). Значение «1» – выводить уставки в теги, «0» – не выводить. Необязательный параметр и по умолчанию задано значение «0»;
- PSW – параметр автоматического ввода пароля при попытке записи значений. **Данный параметр предназначен только для терминалов СИРИУС v3.** PSW = «9999» – запись значения «9999» в поле пароля, «0» – не вводить. Необязательный параметр, по умолчанию равен «0»;
- INFO – флаг вывода информации об авариях/срабатываниях устройства в OPC-теги в группе «INFO» (значение «0»/«1»). Значение «1» – выводить информацию в теги, значение «0» – не выводить. **Данный параметр предназначен только для устройств: БМРЗ, БМПА, БМЦС, СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС, СИРИУС (старого типа), MiCOM P123, MiCOM P124, MiCOM P127.** Необязательный параметр, по умолчанию задано значение «0»;
- IFREQ – частота запросов для считывания номера последнего срабатывания в устройстве в миллисекундах. Необязательный параметр и по умолчанию равен «1000». **Данный параметр предназначен только для устройств: СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС, СИРИУС (старого типа), MiCOM P123, MiCOM P124, MiCOM P127;**
- OSCPath – путь сохранения осциллограмм и событий, считанных с устройства. Необязательный параметр. В случае его отсутствия осциллограммы сохраняются в **C:\Relematika\OSC\ИМЯ_ЛИНИИ\ИМЯ_СТАНЦИИ**. Есть возможность использовать следующие макросы: [YYYY] – год, [MM] – месяц, [DD] – день, [HH] – час, [NN] – минута, [ADR] – адрес устройства;
- CMDID – уникальные цифровые идентификаторы (ID) командных тегов для устройств. Команды в виде тегов заносятся в группу «Commands». Необязательный параметр и может отсутствовать;
- OSCDL – флаг автоматического считывания осциллограмм с устройства. Значение «0» – не скачивать осциллограммы с устройства, любое другое значение или отсутствие параметра – скачивать. Необязательный параметр;
- DACDelay – задержка на отправку запроса главной зоны в миллисекундах. Необязательный параметр, значение по умолчанию «200»;
- ZoneGetWrite – флаг вывода зоны чтения/записи по запросу устройства в OPC-теги в группе «Zone_GetWrite» (значение «0»/«1»). Значение «1» – выводить уставки в теги, «0» – не выводить. Необязательный параметр, по умолчанию имеет значение «0». **Данный параметр предназначен только для устройств: СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС;**
- SetsCheckEnable – параметр, разрешающий фоновое чтение уставок (значение «0»/«1», где значение «0» – отключен, значение «1» – включен). **Данный параметр**

предназначен только для устройств Seram-80. Параметр опциональный, и по умолчанию имеет значение «0»;

– SetsCheckPeriod – период проверки уставок в минутах при фоновом считывании уставок. **Данный параметр предназначен только для устройств Seram-80.** Является обязательным при включении фонового считывания уставок с терминала Seram-80;

– DSTFile – имя специального файла с расширением «*.dst», необходимый для преобразования осциллограмм, хранящихся в устройствах производства ООО «Механотроника» (DeviceType = 150), в формат «COMTRADE», т.к. в данных устройствах осциллограммы хранятся в собственном числовом формате. Для этого необходимо заранее, с помощью ПО «Конфигуратор-МТ», скачать с устройства данный файл и поместить его в папку «DST» в рабочем каталоге ПО «Релематика МД», например, «D:\Relematika\MD\DST». Кроме этого, в папке установки ПО «Релематика МД» находятся три динамически загружаемых библиотеки (dll): «OscillHelper.dll», «Utils.dll», «ZipArchive.dll». После преобразования создаются три файла осциллограммы: dat, cfg, hdr. Также на диске остаётся оригинальный файл с устройства с расширением sto. Данный параметр необязательный и может отсутствовать.

Блок <Tags> представляет секцию настройки запросов к устройству.

Пример:

```
<Tags>
  <Tag .../>
</Tags>.
```

Блок <Tag> содержит параметры одного запроса к устройству (зоны опроса).

Пример:

```
<Tag RegularBegin="" RegularSize="" RegularFreq="" MainDAC=""/>
<Tag RegularBegin="" RegularSize="" RegularFreq=""
ValueFromEvents=""/>
<Tag RegularBegin="" RegularSize="" RegularFreq="" Writable=""/>
<Tag RegularBegin="" RegularSize="" RegularFreq="" DirectAccess=""/>
<Tag DemandBegin="" DemandSize=""/>.
```

Зона опроса представляет собой набор объединенных данных в адресном пространстве устройства. Одновременно можно опрашивать несколько зон, задав начальную зону параметром «RegularBegin» или «DemandBegin», в зависимости от желаемого типа опроса. В первом случае зона или группа зон будет опрашиваться регулярно с периодом опроса заданным «RegularFreq», во втором – при записи в тег опроса зоны «Commands.ReadZoneX» значения «1». Параметры «RegularSize» и «DemandSize» задают количество опрашиваемых зон (одним запросом) соответственно.

ВНИМАНИЕ! Начальный адрес данных каждой следующей зоны должен быть следующим за конечным адресом данных каждой предыдущей зоны. Эта информация содержится в шаблоне устройств.

Данный блок для «обычных» устройств имеет следующие параметры запроса:

– RegularBegin – номер зоны указанный в шаблоне устройств (файл «Templates.xml») в параметре «Num» блока зоны (<Zone Num="1" ... >);

– RegularSize – количество зон, опрашиваемых одним запросом (если в параметре «RegularBegin» указан номер зоны «3», а в «RegularSize» – «2», то произойдет чтение зон под номером 3 и 4);

– RegularFreq – желательный период опроса зоны или группы зон в миллисекундах;

– MainDAC – флаг главной зоны (группы зон) (значение «0»/«1», где значение «0» – отключен, значение «1» – включен). Означает опрос данной зоны (группы зон) будет происходить сразу (вне очереди) после подачи управляющей команды на устройство. Для каждого устройства указывается хотя бы одна зона для регулярного опроса с параметром «MainDAC»;

– DemandBegin – номер начальной зоны опроса группы зон по запросу (команде). Необязательный параметр и может отсутствовать. Не может быть в секции вместе с параметрами «RegularXXX»;

- DemandSize – количество зон, опрашиваемых одним запросом. Не может быть в секции вместе с параметрами «RegularXXX»;
- ValueFromEvents – флаг обновления дискретных OPC-тегов через новые события (значение «0»/«1», где значение «0» – отключен, значение «1» – включен). Работает только с дискретными входами, логическими уравнениями, телесигнализацией и КТС. Необязательный параметр, имеет значение по умолчанию «0» (выключен). **Только для терминалов СЕПАМ(!):**
- DirectAccess – флаг включения режима записи (значение «0»/«1», где значение «0» – отключен, значение «1» – включен). Необязательный параметр и по умолчанию имеет значение «0». Работает только в сочетании с параметром «WR_Mode» = «1». Становятся доступны функции записи 5, 6, 15, 16 (битовые и регистровые (тип INT, UINT) зоны);
- OnlyWrite – параметр, необходимый в случаях, когда по указанному адресу памяти возможно производить только запись значений (отсутствует возможность чтения). В этом случае в конфигурации необходимо задать параметры в следующем порядке: <Tag RegularBegin="1" RegularSize="1" RegularFreq="0" DirectAccess="1" OnlyWrite="1"/>. Все остальные условия параметров «DirectAccess», «WR_Mode» должны быть соблюдены.
Для устройств типа 150, 747 и 1000 блок <Tag> имеет следующие параметры:
 - BitBegin* – начальный адрес бита или битов, где * – имена OPC-тегов в данном случае будут формироваться так: Bit_0xNNNN_Fn_MMM, где NNNN – начальный адрес запроса в HEX, n – номер функции модбас, а MMM – номер бита в массиве;
Пример 1 – BitBegin= «0x100».
 - Пример 2 – Line1.BMRZ_NEW.Bit_F1_0x0001_005.**
- BitsCount – количество битов для чтения, обязателен, если указан параметр «BitBegin». Например: BitsCount = «25»;
- RegBegin** – начальный адрес регистра или регистров для чтения, где ** – имена OPC-тегов в данном случае будут формироваться так: Reg_0xNNNN_Fn_MMM, где NNNN – начальный адрес запроса в HEX, n – номер функции Modbus, а MMM – номер регистра в массиве;
Пример – Line1.BMRZ_NEW.Reg_F65_0xFF9A_000.
- RegsCount – количество регистров для чтения. Обязателен, если указан параметр «RegBegin»;
- Func – номер функции. Обязательный параметр. Для битовых тегов могут быть указаны функции 1, 2. Для регистровых тегов – функции 3, 4, 65;
- Freq – период опроса заданных адресов в миллисекундах. Обязательный параметр;
- DataType – обязательный параметр, если заданы параметры «RegBegin» и «RegsCount», иначе не учитывается. Определяет тип данных в устройстве для конфигурируемых параметров, т.е. все данные в диапазоне адресов будут интерпретироваться как данный тип. Возможные типы данных указаны в таблице 18;
- OPCDT – обязательный параметр, если заданы параметры «RegBegin» и «RegsCount», иначе не учитывается. Задаёт тип OPC-тега. Возможные типы данных указаны в таблице 7;
- Size – обязательный параметр, если заданы параметры «RegBegin» и «RegsCount», иначе не учитывается. Задаёт размер типа данных в регистрах (2 байта). Например, если указано два регистра, то считывается четыре байта;
- Writable – флаг записи значений в тег для указанного диапазона адресов (значение «0»/«1», где значение «0» – отключен, значение «1» – включен). Необязательный параметр, по умолчанию имеет значение «0» (выключен). Для записи битов используется функция «5», для записи регистров – функция «16»;
- CreateTags – флаг возможности создания тегов и запросов для описания событий (значения «0»/«1»). При значении «1» – теги и запросы создаваться не будут, при значении «0» – будут создаваться. Применимо только к битовым зонам. Необязательный параметр, имеет значение по умолчанию «1». **Предназначен только для устройств ООО «Релематика» (DeviceType=1000);**

- Desc – строковое описание бита/регистра. Все описания заканчиваются разделителем «:», в том числе и конечное;
- MainDAC – флаг главной зоны (группы зон) (значение «0»/«1», где значение «0» – отключен, значение «1» – включен). Означает опрос данной зоны (группы зон) сразу после подачи управляющей команды на устройство. Данный параметр указывается только для устройств типа 150.

Таблица 18 – Типы данных Modbus, поддерживаемые устройствами 150, 747 и 1000

| DataType | OPCDT | Комментарий |
|-----------------|-------|---|
| UINT | 0 | |
| UINT16_01 | 0 | |
| UINT16_10 | 0 | |
| UINT32 | 0 | |
| UINT32_SWAPREG | 0 | |
| UINT32_0123 | 0 | |
| UINT32_1032 | 0 | |
| INT | 0 | |
| INT32 | 0 | |
| INT32_SWAPREG | 0 | |
| INT16_01 | 0 | |
| INT16_10 | 0 | |
| INT32_1032 | 0 | |
| INT32_0123 | 0 | |
| INTA | 0 | |
| UINTA | 0 | |
| INT32A | 0 | |
| UINT32A | 0 | |
| INTB | 0 | |
| UINTB | 0 | |
| INT32B | 0 | |
| UINT32B | 0 | |
| REAL | 1 | |
| HIBYTE | 0 | Старший байт регистра |
| LOBYTE | 0 | Младший байт регистра |
| FLOAT | 1 | |
| FLOAT_0123 | 1 | |
| FLOAT_1032 | 1 | |
| DOUBLE_01234567 | 1 | |
| DOUBLE_76543210 | 1 | |
| DOUBLE_10325476 | 1 | |
| DOUBLE_67452301 | 1 | |
| DOUBLE_23016745 | 1 | |
| DOUBLE_54761032 | 1 | |
| DOUBLE_32107654 | 1 | |
| DOUBLE_45670123 | 1 | |
| DATETIME | 2 | Строковый тип для вывода в тег времени устройства, используется только для устройств DataType = 150 и DataType = 1000 |

После имени типа цифрами показан порядок байт.

OPCDT = OPC Data Type – тип данных OPC тега. Возможные значения:

- 0 – INTEGER (VT_I4);

- 1 – REAL (VR_R8);
- 2 – STRING (VT_BSTR);
- 3 – BOOL (VT_BOOL).

Пример записи блоков и всех параметров в файле конфигурации (значения параметров в примере не является прототипом для применения):

```
<MODBUS_DLL_CONFIG Name="ModBus" ServerOFF="0" CheckKeyTimeout"60"
DemoOFF="1" LogLevel="1" Flush="1">
  <Lines Name="Линии связи">
    <Line Name="Line" TCP="0" ModbusTCP="1" IPAddr="192.172.100.105"
Port="4001" ComNumber="16" BaudRate="9600" DataBits="8" StopBits="1"
Parity="2" InUse="1" OffTagInUse="1" MaxTimeout="500" RetryLimit="2"
TimeSynchro="1" TimeSynchroPeriod="10000" TSCR="1000" >
      <Station Name="24_OL" InUse="1" ModbusAddr="1" DeviceType="5"
WR_Mode="0" SessionQuota="2" EVENTS="0" EAUTOACK="0" EFREQ="1000"
EUPDT="1000" EVTimeFromPC="0" SETS="0" PSW="0" INFO="0" IFREQ="0"
OSCPATH="C:\OSC\ [YYYY]_[MM]_[DD]\[HH]_[NN]\Устройство_[ADR]" CMDID="6034,
6035, 6036, 6037, 6038, 6039, 6051, 6052, 6053" OSCDL="0" DACDelay="0"
ZoneGetWrite="1" >
        <Tags>
          <Tag RegularBegin="7" RegularSize="1" RegularFreq="2000"
MainDAC="1" />
          <Tag RegularBegin="1" RegularSize="1" RegularFreq="500"
Writable="1" />
          <Tag RegularBegin="2" RegularSize="1" RegularFreq="500"
DirectAccess="1" />
          <Tag RegularBegin="3" RegularSize="1" RegularFreq="1000"
ValueFromEvents="1" />
          <Tag DemandBegin="4" DemandSize="1" />
        </Tags>
      </Station>
    </Line>
  </Lines>
</MODBUS_DLL_CONFIG>.
```

3.19.2 Общая информация по работе со специализированными устройствами

3.19.2.1 Уставки

Режим работы с уставками поддерживается следующими терминалами:

- SEPAМ 20;
- SEPAМ 40;
- SEPAМ 80;
- SEPAМ 2000;
- БМЦС;
- БМРЗ;
- БМПА;
- СИРИУС (старого типа);
- СИРИУС-2-В;
- СИРИУС-2-С;
- СИРИУС-2-Л;
- СИРИУС-2-ГС;
- MiCOM P123;
- MiCOM P124;
- MiCOM P127.

Для включения уставок необходимо в конфигурации объекта «MODBUS_DLL_CONFIG» включить параметр «SETS» = «1». Для считывания уставок необходимо добавить в конфигурацию объекта «MODBUS_CLIENT» OPC-тег «Get_Set», а для записи – OPC-тег

«Write_Set». Выбор активного набора уставок в тестовом режиме для терминалов СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС осуществляется с помощью OPC-тега «Sets.ActiveSet» (приложение К).

Таким образом, для чтения уставок необходимо записать значение «1» в OPC-тег «Get_Set». Для изменения уставок необходимо сначала их считать (Get_Set), далее записать в OPC-тег уставки новое значение и записать значение «1» в OPC-тег «Write_Set», подав тем самым команду на запись. А для выбора активного набора уставок необходимо записать в OPC-тег «Sets.ActiveSet» номер набора уставок. Каждый набор уставок находится в OPC-группе «SetsX», где X – это номер набора уставок.

Пример – «Sets2» – второй набор уставок.

3.19.2.2 Зона чтения/записи по запросу

Режим работы с зоной поддерживается следующими терминалами:

- СИРИУС-2-В;
- СИРИУС-2-С;
- СИРИУС-2-Л;
- СИРИУС-2-ГС.

Для включения зоны необходимо в конфигурации «MODBUS_DLL_CONFIG» добавить параметр «ZoneGetWrite» = «1» (приложение Л), а также в шаблоне устройств необходимо добавить параметр «GetWrite» = «1» в зону. Данная зона имеет параметр «Internal» = «1».

Для чтения зоны необходимо записать значение «1» в OPC-тег «GetValues_X» (тег задается в конфигурации объекта «MODBUS_CLIENT» (приложение К)), где X – это название ID зоны в шаблоне устройств.

Для изменения зоны необходимо сначала их считать (GetValues_X), далее записать в необходимый OPC-тег зоны новое значение и записать значение «1» в OPC-тег «WriteValues_X», подав тем самым команду на запись, где X – это название ID зоны в шаблоне устройств. OPC-теги задаются в конфигурации объекта «MODBUS_CLIENT» (приложение К).

3.19.2.3 События

Режим считывания событий поддерживается следующими терминалами:

- SEPAМ 20;
- SEPAМ 40;
- SEPAМ 80;
- SEPAМ 2000;
- БМЦС;
- MICROLOGIC;
- СИРИУС (старого типа);
- СИРИУС-2-В;
- СИРИУС-2-С;
- СИРИУС-2-Л;
- СИРИУС-2-ГС;
- MiCOM (P123);
- MiCOM (P124);
- MiCOM (P127).

Для включения событий необходимо в конфигурации объекта добавить параметр «EVENTS» = «1».

Для терминалов MiCOM P123, MiCOM P124, MiCOM P127: в терминале есть два режима чтения событий:

- автоматический (считали, значит отквитировали);
- полуавтоматический (считали, далее отдельным запросом отквитировали).

Для включения в MiCOM режима автоматического квитирования необходимо добавить дополнительный параметр «EAUTOACK» = «1», в противном случае будет активен полуавтоматический режим.

Для терминалов SEPAМ 20, SEPAМ 40, SEPAМ 80, SEPAМ 2000, БМЦС, СИРИУС (старого типа), СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС, МiСОМ Р123, МiСОМ Р124, МiСОМ Р127 в режиме считывания событий имеются следующие особенности:

- события автоматически записываются в файл формата «*.csv» на жесткий диск по умолчанию в папку C:\Relematika\OSC\ИМЯ_ЛИНИИ\ИМЯ_СТАНЦИИ\EVENTS. Чтобы создать новый каталог для хранения журнала событий необходимо в конфигурации добавить параметр «OSCPATH» = «C:\ИМЯ_ПАПКИ». Папка «EVENTS» создается автоматически в новом каталоге;

- каталог с событиями хранится в каталоге с осциллограммами. Если в параметре «OSCPATH» присутствуют макросы, то каталог для событий будет располагаться в каталоге с первым макросом.

Для терминалов SEPAМ 20, SEPAМ 40, SEPAМ 80, SEPAМ 2000, БМЦС, MICROLOGIC, СИРИУС (старого типа), СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС, МiСОМ Р123, МiСОМ Р124, МiСОМ Р127 в режиме считывания событий имеются следующие особенности:

- для вывода новых событий используются теги «EVTime» (приложение К) и «ServerALIVE». Для этого необходимо записать в тег «ServerALIVE» значение «1», в результате чего в теге «EVTime» отобразится информационная строка о последних новых событиях;

- для обновления OPC-тегов дискретных сигналов по новым событиям в конфигурации необходимо в теге станции с регулярным опросом добавить параметр «ValueFromEvents» = «1» (только для терминалов СЕРАМ). При этом, если данный параметр выставлен, то события будут включены вне зависимости от включения/отключения параметра «EVENTS»;

- для обновления времени событий в теге «EVTime» (приложение К) из ПК необходимо использовать в конфигурации параметр в станции «EVTimeFromPC» = «1» (не реализовано в терминалах СИРИУС, МiСОМ); в терминале СИРИУС (старого типа) при первом запуске объекта первое считанное событие не выводится, т.к. оно используется для логики выявления изменения дискретных сигналов.

3.19.2.4 Срабатывания

Режим работы срабатываний поддерживается следующими терминалами:

- БМРЗ;
- БМПА;
- СИРИУС (старого типа);
- СИРИУС-2-В;
- СИРИУС-2-С;
- СИРИУС-2-Л;
- СИРИУС-2-ГС;
- МiСОМ Р123;
- МiСОМ Р124;
- МiСОМ Р127.

Для включения/отключения данного режима необходимо в конфигурационном файле объекта в узле <Station> параметру «INFO» записать значение «1»/«0» (INFO = «1»).

У терминалов БМРЗ и БМПА для того, чтобы считать нужные данные, необходимо записать номер аварии в OPC-тег «Commands.Damage_Num» (приложение К), который задается в конфигурации объекта «MODBUS_CLIENT».

У терминалов СИРИУС (старого типа), СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС имеются следующие особенности для работы в режиме срабатываний:

- номер последнего срабатывания отображается в OPC-теге «INFO.NumberLastDamage» (приложение К) (задается в конфигурации объекта «MODBUS_CLIENT»), частота запроса которого регулируется в файле конфигурации объекта через параметр «IFREQ»=«2000» (значение в миллисекундах). Значение по умолчанию в теге задано «0»;

- максимальное допустимое количество срабатываний содержится в OPC-теге «INFO.MaxNumberDamage» (приложение К) (отсутствует в терминалах СИРИУС (старого типа));

- данные срабатывания содержатся в группе «INFO.InfoTags» (для терминалов СИРИУС (старого типа) в группе «Info»);

- для того, чтобы считать нужные данные, необходимо записать номер срабатывания в OPC-тег «INFO.Commands.GetInfoDamage_ByNumber».

Для терминалов MiCOM P123, MiCOM P124, MiCOM P127 имеются следующие особенности для работы в режиме срабатываний:

- номер последнего срабатывания отображается в OPC-теге «Info.NumberLastDamage» (приложение К), частота запроса которого регулируется в файле конфигурации объекта через параметр «IFREQ»=«2000» (значение в миллисекундах). Значение по умолчанию в теге задано «0»;

- максимальное допустимое количество срабатываний содержится в OPC-теге «Info.MaxNumberDamage» (приложение К);

- данные срабатывания содержатся в группе «Info.Data»;

- для того, чтобы считать нужные данные, необходимо записать номер срабатывания в OPC-тег «Info.Commands.GetInfoDamage_ByNumber»;

- для того, чтобы выводить данные о самом новом срабатывании автоматически, необходимо в OPC-тег «Info.AUTO_MODE» (приложение К) записать значение «1». По умолчанию автоматический режим отключен (значение OPC-тега «Info.AUTO_MODE» равно «0») – ручной режим);

- в автоматическом режиме команда «Info.Commands.GetInfoDamage_ByNumber» отключена;

- в ручном режиме заданное значение OPC-тега «Info.Commands.GetInfoDamage_ByNumber» фиксируется (не изменяется);

- расчет токов происходит по внутренним коэффициентам терминала, которые находятся в уставках. Данные коэффициенты необходимо задать перед эксплуатацией объекта.

В ручном режиме, в случае успешного выполнения запроса, значение тега «REQ_RESULT» будет содержать сообщение «#0 ALL OK» об успешном завершении операции. В автоматическом режиме вывод сообщений об успешном считывании срабатываний в тег «REQ_RESULT» отключен.

Автоматический режим помогает осуществлять быстрый поиск последнего срабатывания. Рекомендуется использовать:

- одномоментно – включить, дождаться результата поиска последнего срабатывания, отключить;

- постоянно – с увеличенной частотой опроса (IFREQ = «20000» или выше), чтобы не нагружать канал связи.

Командные OPC-теги «Commands.Damage_Num», «INFO.Commands.GetInfoDamage_ByNumber» и «Info.Commands.GetInfoDamage_ByNumber» сбрасываются только в случае, если в них записаны недопустимые значения, в остальных случаях записанное значение остается неизменным.

3.19.2.5 Осциллограммы

Режим скачивания осциллограмм поддерживается следующими терминалами:

- SEPAМ 20;
- SEPAМ 40;
- SEPAМ 80;
- SEPAМ 2000;
- СИРИУС-2-В;
- СИРИУС-2-С;
- СИРИУС-2-Л;
- СИРИУС-2-ГС;
- MiCOM P123;

- MiCOM P124;
- MiCOM P127;
- БМРЗ.

Автоматическое скачивание осциллограмм включено по умолчанию. Для того, чтобы отключить данный режим, необходимо в конфигурации объекта добавить параметр «OSCDL» = «0».

Осциллограммы автоматически записываются в файлы на жесткий диск по умолчанию в папку C:\Relematika\OSC\ИМЯ_ЛИНИИ\ИМЯ_СТАНЦИИ\. Чтобы создать новый каталог для хранения осциллограмм необходимо в конфигурации добавить параметр «OSCPATH» = «C:\ИМЯ_ПАПКИ». В параметре «OSCPATH» допустимо применение следующих макросов (макрос – символьное имя, заменяемое значением из метки времени скачанной осциллограммы):

- [YYYY] – год;
- [MM] – месяц;
- [DD] – день;
- [HH] – часы;
- [NN] – минуты;
- [ADR] – адрес устройства.

Пример – OSCPPath="C:\Sepam\[YYYY]_[MM]_[DD]_[HH]_[NN]\Устройство_[ADR]".

В каталоге с осциллограммами хранится каталог с событиями. Если в параметре «OSCPATH» присутствуют макросы, то каталог для событий будет располагаться в каталоге с первым макросом. Для примера выше это будет каталог: C:\Sepam\EVENTS.

В режиме скачивания осциллограмм по умолчанию включены следующие OPC-теги:

- OSCAUTO – включение/отключение автоматического скачивания осциллограмм;
- OSCREFRESHLIST – флаг обновления списка осциллограмм;
- OSCStatus – информационная строка статуса считывания осциллограммы;
- OSCTODL – команда на ручное скачивание осциллограммы (номер осциллограммы).

3.19.2.6 Командные теги, телеуправление

Режим телеуправления поддерживается следующими терминалами:

- SEPAM 20;
- SEPAM 40;
- SEPAM 80;
- SEPAM 2000;
- БМПА;
- Micrologic;
- СИРИУС (старого типа);
- СИРИУС-2-В;
- СИРИУС-2-С;
- СИРИУС-2-Л;
- СИРИУС-2-ГС;
- MiCOM P123;
- MiCOM P124;
- MiCOM P127.

У терминалов SEPAM, MiCOM P123, MiCOM P124, MiCOM P127, СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС для включения командных тегов необходимо в конфигурационном файле объекта записать в узле <Station> в параметр «CMDID» уникальные цифровые идентификаторы (ID) командных тегов. Для отключения командных тегов параметр «CMDID» необходимо оставить пустым: CMDID="".

Для терминалов СИРИУС (старого типа) командные теги включены всегда.

3.19.3 Особые замечания для терминалов SEPAM

В именах, считанных и сохраненных на диске осциллограмм терминалов Seram-80 производства «Schneider Electric», указано время первой выборки, а не время пуска осциллографа. Реальная метка времени пуска содержится в конфигурационном файле осциллограммы.

Также, при подаче команды на принудительное чтение осциллограммы с устройства, осциллограмма начнет считываться через 90 с. Это сделано для исключения выдачи ошибок терминалом при попытке прервать текущее скачивание.

3.19.4 Особые замечания для терминалов MiCOM

В работе с терминалами MiCOM после того, как все осциллограммы будут скачаны в автоматическом режиме, объект начнет проверять наличие новых осциллограмм с интервалом 60 с.

3.19.5 Особые замечания для терминалов БМЦС

Если в конфигурации объекта для терминалов БМЦС параметру «EVENTS» выставлено значение «1», то события читаются в файл на диск ПК автоматически: по умолчанию в папку C:\MBOSC\ИМЯ_ЛИНИИ\ИМЯ_СТАНЦИИ\EVENTS. В тег «EVTime» (приложение К), при необходимости, можно вывести события вручную посредством OPC-тега «ServerALIVE».

В случае, отсутствия параметра «EVENTS» или если «EVENTS» равен «0», то считывание событий переключается на ручной режим, при этом выводится тег «EVRead».

3.19.6 Сообщения оператору

Сообщения оператору отображаются через OPC-тег «REQ_RESULT». Варианты сообщений представлены в таблице 19.

Таблица 19 – Варианты сообщений оператору

| Сообщение | Описание |
|---------------------------|---|
| #0 ALL OK | Запрос принят терминалом, пришел корректный ответ |
| #1 ILLEGAL FUNCTION | В запросе используется не поддерживаемая терминалом функция (код ошибки – 01) |
| #2 ILLEGAL DATA ADDRESS | Неверная адресация данных в запросе (код ошибки – 02) |
| #3 ILLEGAL DATA VALUE | Неверные данные в запросе (код ошибки – 03) |
| #4 SLAVE DEVICE FAILURE | Устройство (терминал) не исправно (код ошибки – 04) |
| #5 SLAVE ACK | Запрос принят (код ошибки – 05) |
| #6 SLAVE BUSY | Устройство занято, повторить позднее (код ошибки – 06) |
| #7 SLAVE NACK | Выполнение запроса невозможно (код ошибки – 07) |
| #8 UNCKNOWN ERROR | Неопознанная ошибка |
| #9 LATE ANSWER | Неверный адрес устройства в ответе |
| #10 SLAVE TIMEOUT | Таймаут ожидания ответа от устройства |
| #11 SLAVE CRC ERROR | Неверная контрольная сумма в ответе от устройства |
| #12 COMM INIT FATAL ERROR | Ошибка инициализации коммуникационного порта |
| #13 COMM FATAL ERROR | Ошибка порта, возникшая в процессе работы |
| #16 STATION NOT IN USE | Станция выключена из опроса (при попытке записи в теги станции) |

| Сообщение | Описание |
|---|---|
| #17 LINE NOT IN USE | Линия не используется (при попытке записи в теги станции) |
| #18 STATION NOT CONNECTED | Устройство не отвечает (при попытке записи в теги станции) |
| #19 REQ PERIOD TOO SMALL | Заданные периоды запросов слишком малы для текущего коммуникационного канала. Опрос продолжается, но период опроса не укладывается в рамки, заданные в конфигурационном файле в параметре «RegularFreq» |
| #22 ZONE IS NOT READ YET | Зона в данный момент не считана (возникает при попытке записи значений еще не считанных уставок) |
| #24 SLAVE EXCHANGE IDENTIFIER ERROR | Идентификатор обмена в ответном пакете по протоколу Modbus TCP не соответствует идентификатору обмена соответствующего запроса |
| #25 SLAVE ERROR GETEWAY TARGET DEVICE FAILED TO RESPOND | Устройство не отвечает (не в сети). Возникает по протоколу Modbus TCP. Специализированное использование в сочетании со шлюзами |
| #26 SLAVE ERROR GETEWAY PATH UNAVAILABLE | Возникает по протоколу Modbus TCP. Специализированное использование в сочетании со шлюзами |
| #50 GET SETS OK | Уставки успешно прочитаны |
| #52 WRITE SETS OK | Пришел корректный ответ от устройства на запись уставок |
| #55 EXECUTE (R-W) | Объект в процессе чтения-записи команд |
| #100 DEVICE NOT SUPPORT REMOTE CFG READ | Только для Seram (уставки). Не поддерживается дистанционное чтение конфигурации устройства |
| #101 CARTRIDGE VER GREATER THAN 94/42 | Только для Seram (уставки). Неверная версия картриджа |
| #102 ANSWER NOT READY. RETRY | Только для Seram (уставки). Ответ не готов, повторите попытку |
| #103 NO FUNC OR REMOTE READING DISBL | Только для Seram (уставки). Функция защиты отсутствует или удаленное чтение конфигурации устройства запрещено |

Приложение А
(справочное)
Пример файла конфигурации «config.xml»

```
<UniIniData ModificationDate="14.09.2022 11:06:05">  
  <Register ServiceName="RelematikaMD3"  
    <LocalIPCHost cfgfile="ipccfg.xml">  
      <Logs CountStr="10000" CountFile="3" LogLevel="1" Path=""/>  
    </LocalIPCHost>  
  </UniIniData>
```

Приложение Б (справочное) Пример файла проекта

```

<rmdcfgdata ModificationDate="21.07.2022 15:45:14" version="3">
  <OPCUA_SERVER name="OPCUASERVER">
    <property name="NameServer" value="RMD3xOPCUAServerTest"/>
    <property name="TCPPort" value="4840"/>
    <!-- <tag name="IU_Client" value="IU_Client" vartype="VT_BOOL"
sendto="OPCUACLIENT\IU"/>
    <tag name="NC_Client" value="NC_Client" vartype="VT_BOOL"/>
    <tag name="Byte" value="16" vartype="VT_UI1"/>
    <tag name="Int32" value="32" vartype="VT_I4"/> -->
    <tag name="TagReadiness" value="Readiness" vartype="VT_BOOL"/>
    <tag name="TagStatus" value="Status" vartype="VT_LPSTR"/>

    <tag name="TagStartIN" value="StartIN" vartype="VT_BOOL"
sendto="ObjR58601\TagStartIN"/>
  </OPCUA_SERVER>

  <R58601 name=" R58601">
    <property name="InputDirNumber" value="1"/>
    <property name="InputDir1" value="D:/r58601test/r860"/>
    <property name="OutputDirs" value="D:/r58601test/r860out"/>
    <property name="OutputFile" value="D:/OutputFile"/>
    <property name="DBConnStr" value="user=postgres password=admin
dbname=R58601 host=localhost port=5432"/>
    <property name="DBMaxFileSize" value="0"/>
    <property name="DBMaxStorageTime" value="0"/>
    <property name="DBCheckInterval" value="0"/>
    <property name="OscTimeRange" value="100"/>
    <property name="OscInputTimeout" value="1"/>
    <property name="OscWaitTimeout" value="1000"/>
    <property name="OscMaxAge" value="0"/>
    <property name="Configuration"
value="D:/r58601test/P58601.pcfg"/>
    <tag name="TagStartIN" value="StartIN" vartype="VT_BOOL"/>
    <tag name="TagReadiness" value="Readiness" vartype="VT_BOOL"
sendto="UAServer\TagReadiness"/>
    <tag name="TagStatus" value="Status" vartype="VT_LPSTR"
sendto="UAServer\TagStatus"/>
  </R58601>
</rmdcfgdata>

```

Приложение В
(справочное)
**Пример конфигурации объекта «OPCUA_SERVER» и объекта
«IEC61850_CLIENT»**

```
<OPCUA_SERVER name="OPC" >
  <property name="NameServer" value="OPCUAServer"/>
  <property name="Timeout" value="200"/>
  <property name="Certificate" value="Сертификат в двоичном виде"/>
  <property name="PrivateKey" value="Ключ в двоичном виде"/>
  <property name="AllowUnencrypted" value="0"/>
  <property name="AcceptAllCerts" value="0"/>
  <property name="UsersCount" value="1"/>
  <property name="Username1" value="Имя пользователя 1"/>
  <property name="Password1" value="Пароль пользователя 1 в зашифрованном
  виде"/>
  <tag name="PhV_phSA_mag_f" value="PhV_phSA_mag_f" vartype="VT_R8"/>
  <tag name="PhV_phSB_ang_f" value="PhV_phSB_ang_f" vartype="VT_R8"/>
  <tag name="PhV_phSB_mag_f" value="PhV_phSB_mag_f" vartype="VT_R8"/>
  <tag name="PhV_phSC_ang_f" value="PhV_phSC_ang_f" vartype="VT_R8"/>
  <tag name="PhV_phSC_mag_f" value="PhV_phSC_mag_f" vartype="VT_R8"/>
  <tag name="PhV_neut_ang_f" value="PhV_neut_ang_f" vartype="VT_R8"/>
  <tag name="PhV_neut_mag_f" value="PhV_neut_mag_f" vartype="VT_R8"/>
  <tag name="IU" value="IU" vartype="VT_BOOL" sendto="TOR\IU"/>
</OPCUA_SERVER>
```

При этом в объекте «IEC61850_CLIENT» задать:

```
<IEC61850_CLIENT name="TOR" >
  <property name="IPAddress" value="192.168.102.200"/>
  <property name="TCPPort" value="102"/>
  <property name="MaxCountErrorConnect" value="0"/>
  <property name="TimeoutNotConnect" value="1000"/>
  <property name="GF_Dir"
value="d:/ossC/[YYYY]_[MM]_[DD]/[HH]_[NN]/BMRZ_[ADR]"/>
  <property name="GF_Dir2"
value="d:/ossC2/[YYYY]_[MM]_[DD]/[HH]_[NN]/BMRZ_[ADR]"/>
  <property name="GF_TimeOut" value="90000" />
  <property name="GF_Ext" value="zip" />
  <property name="IntgPd" group="1" value="100"/>
  <property name="ReportName" group="1"
value="TOR_300MEAS/LLN0.BR.Report_MEAS102"/>
  <property name="ReportOpt" group="1" value="TRG_OPT_INTEGRITY"/>
  <property name="OptFlds" group="1"
value="RPT_OPT_SEQ_NUM|RPT_OPT_TIME_STAMP|RPT_OPT_REASON_FOR_INCLUSION|RPT
_OPT_DATA_SET|RPT_OPT_DATA_REFERENCE|RPT_OPT_BUFFER_OVERFLOW|RPT_OPT_ENTRY
_ID|RPT_OPT_CONF_REV"/>
  <tag name="IU" value="IU" vartype="VT_BOOL"/>
  <tag name="PhV_phSA_mag_f" group="1"
value="TOR_300MEAS/MMXU1.PhV.phSA[MX]cVal.mag.f#q#t=t" vartype="VT_R8"
sendto="OPC\PhV_phSA_mag_f"/>
```

```
<tag name="PhV_phsB_ang_f" group="1"
value="TOR_300MEAS/MMXU1.PhV.phsB[MX]cVal.ang.f#q=q#t=t" vartype="VT_R8"
sendto="OPC\PhV_phsB_ang_f"/>
<tag name="PhV_phsB_mag_f" group="1"
value="TOR_300MEAS/MMXU1.PhV.phsB[MX]cVal.mag.f#q=q#t=t" vartype="VT_R8"
sendto="OPC\PhV_phsB_mag_f"/>
<tag name="PhV_phsC_ang_f" group="1"
value="TOR_300MEAS/MMXU1.PhV.phsC[MX]cVal.ang.f#q=q#t=t" vartype="VT_R8"
sendto="OPC\PhV_phsC_ang_f"/>
<tag name="PhV_phsC_mag_f" group="1"
value="TOR_300MEAS/MMXU1.PhV.phsC[MX]cVal.mag.f#q=q#t=t" vartype="VT_R8"
sendto="OPC\PhV_phsC_mag_f"/>
<tag name="PhV_neut_ang_f" group="1"
value="TOR_300MEAS/MMXU1.PhV.neut[MX]cVal.ang.f#q=q#t=t" vartype="VT_R8"
deadband="10" coefficient="0.1" sendto="OPC\PhV_neut_ang_f"/>
<tag name="PhV_neut_mag_f" group="1"
value="TOR_300MEAS/MMXU1.PhV.neut[MX]cVal.mag.f#q=q#t=t" vartype="VT_R8"
sendto="OPC\PhV_neut_mag_f"/>
</IEC61850_CLIENT>
```

Приложение Г
(обязательное)
Шаблоны подстановки для объекта формирования единой
осциллограммы «R58601»

“[YYYY]” – год начала склейки,
“[YY]” – год начала склейки (последние 2 цифры),
“[MM]” – месяц начала склейки,
“[DD]” - день начала склейки,
“[HH]” – час начала склейки,
“[NN]” – минута начала склейки,
“[SS]” – секунда начала склейки,
“[MSS]” – миллисекунда начала склейки.
“[AYYYYY]” – год аварии,
“[AYY]” – год аварии (последние 2 цифры),
“[AMM]” – месяц аварии,
“[ADD]” - день аварии,
“[AHH]” – час аварии,
“[ANN]” – минута аварии,
“[ASS]” – секунда аварии,
“[AMSS]” – миллисекунда аварии,
“[FDEF]” – использовать имя, сгенерированное объектом «R58601», шаблон только для выходных файлов.

Приложение Д
(справочное)
Пример скрипта создания БД для объекта формирования единой
осциллограммы «R58601»

```
-- PostgreSQL database dump
--

-- Dumped from database version 14.4
-- Dumped by pg_dump version 15.0

-- Started on 2022-12-12 16:07:08

SET statement_timeout = 0;
SET lock_timeout = 0;
SET idle_in_transaction_session_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SELECT pg_catalog.set_config('search_path', '', false);
SET check_function_bodies = false;
SET xmloption = content;
SET client_min_messages = warning;
SET row_security = off;

--
-- TOC entry 4 (class 2615 OID 2200)
-- Name: public; Type: SCHEMA; Schema: -; Owner: -
--

-- *not* creating schema, since initdb creates it

SET default_table_access_method = heap;

--
-- TOC entry 210 (class 1259 OID 16542)
-- Name: waveforms_groups; Type: TABLE; Schema: public; Owner: -
--

CREATE TABLE public.waveforms_groups (
    id integer NOT NULL,
    accident_timestamp timestamp with time zone NOT NULL
);

--
-- TOC entry 209 (class 1259 OID 16541)
-- Name: WaveformsGroups_id_seq; Type: SEQUENCE; Schema: public; Owner: -
--

CREATE SEQUENCE public."WaveformsGroups_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
```

```

--
-- TOC entry 3330 (class 0 OID 0)
-- Dependencies: 209
-- Name: WaveformsGroups_id_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: -
--

ALTER SEQUENCE public."WaveformsGroups_id_seq" OWNED BY public.waveforms_groups.id;

--
-- TOC entry 213 (class 1259 OID 16570)
-- Name: waveforms_processings; Type: TABLE; Schema: public; Owner: -
--

CREATE TABLE public.waveforms_processings (
    id integer NOT NULL,
    waveform_group_id integer NOT NULL,
    result text NOT NULL,
    "timestamp" timestamp with time zone NOT NULL
);

--
-- TOC entry 211 (class 1259 OID 16568)
-- Name: WaveformsProcessings_id_seq; Type: SEQUENCE; Schema: public; Owner: -
--

CREATE SEQUENCE public."WaveformsProcessings_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

--
-- TOC entry 3331 (class 0 OID 0)
-- Dependencies: 211
-- Name: WaveformsProcessings_id_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: -
--

ALTER SEQUENCE public."WaveformsProcessings_id_seq" OWNED BY public.waveforms_processings.id;

--
-- TOC entry 212 (class 1259 OID 16569)
-- Name: WaveformsProcessings_waveform_group_id_seq; Type: SEQUENCE; Schema: public; Owner: -
--

CREATE SEQUENCE public."WaveformsProcessings_waveform_group_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

--
-- TOC entry 3332 (class 0 OID 0)
-- Dependencies: 212

```

```

-- Name: WaveformsProcessings_waveform_group_id_seq; Type: SEQUENCE OWNED BY; Schema:
public; Owner: -
--

ALTER SEQUENCE public."WaveformsProcessings_waveform_group_id_seq" OWNED BY
public.waveforms_processings.waveform_group_id;

--
-- TOC entry 214 (class 1259 OID 17185)
-- Name: waveforms; Type: TABLE; Schema: public; Owner: -
--

CREATE TABLE public.waveforms (
  id integer NOT NULL,
  file_name text NOT NULL,
  waveform_group_id integer NOT NULL,
  data oid NOT NULL,
  file_size bigint NOT NULL,
  file_upload_timestamp timestamp with time zone NOT NULL,
  is_out boolean NOT NULL
);

--
-- TOC entry 215 (class 1259 OID 17232)
-- Name: waveforms_id_seq; Type: SEQUENCE; Schema: public; Owner: -
--

ALTER TABLE public.waveforms ALTER COLUMN id ADD GENERATED ALWAYS AS IDENTITY (
  SEQUENCE NAME public.waveforms_id_seq
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1
);

--
-- TOC entry 3175 (class 2604 OID 16593)
-- Name: waveforms_groups id; Type: DEFAULT; Schema: public; Owner: -
--

ALTER TABLE ONLY public.waveforms_groups ALTER COLUMN id SET DEFAULT
nextval('public."WaveformsGroups_id_seq"::regclass);

--
-- TOC entry 3176 (class 2604 OID 16610)
-- Name: waveforms_processings id; Type: DEFAULT; Schema: public; Owner: -
--

ALTER TABLE ONLY public.waveforms_processings ALTER COLUMN id SET DEFAULT
nextval('public."WaveformsProcessings_id_seq"::regclass);

--
-- TOC entry 3177 (class 2604 OID 16619)
-- Name: waveforms_processings waveform_group_id; Type: DEFAULT; Schema: public; Owner: -
--

```

```
ALTER TABLE ONLY public.waveforms_processings ALTER COLUMN waveform_group_id SET
DEFAULT nextval('public."WaveformsProcessings_waveform_group_id_seq"::regclass);
```

```
--
-- TOC entry 3179 (class 2606 OID 16595)
-- Name: waveforms_groups WaveformsGroups_pkey; Type: CONSTRAINT; Schema: public; Owner: -
--
```

```
ALTER TABLE ONLY public.waveforms_groups
ADD CONSTRAINT "WaveformsGroups_pkey" PRIMARY KEY (id);
```

```
--
-- TOC entry 3183 (class 2606 OID 17191)
-- Name: waveforms WaveformsProcessing_pkey; Type: CONSTRAINT; Schema: public; Owner: -
--
```

```
ALTER TABLE ONLY public.waveforms
ADD CONSTRAINT "WaveformsProcessing_pkey" PRIMARY KEY (id);
```

```
--
-- TOC entry 3181 (class 2606 OID 16612)
-- Name: waveforms_processings WaveformsProcessings_pkey; Type: CONSTRAINT; Schema: public;
Owner: -
--
```

```
ALTER TABLE ONLY public.waveforms_processings
ADD CONSTRAINT "WaveformsProcessings_pkey" PRIMARY KEY (id);
```

```
--
-- TOC entry 3185 (class 2606 OID 17192)
-- Name: waveforms WaveformsProcessing_waveform_group_id_fkey; Type: FK CONSTRAINT; Schema:
public; Owner: -
--
```

```
ALTER TABLE ONLY public.waveforms
ADD CONSTRAINT "WaveformsProcessing_waveform_group_id_fkey" FOREIGN KEY
(waveform_group_id) REFERENCES public.waveforms_groups(id);
```

```
--
-- TOC entry 3184 (class 2606 OID 16620)
-- Name: waveforms_processings WaveformsProcessings_waveform_group_id_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: -
--
```

```
ALTER TABLE ONLY public.waveforms_processings
ADD CONSTRAINT "WaveformsProcessings_waveform_group_id_fkey" FOREIGN KEY
(waveform_group_id) REFERENCES public.waveforms_groups(id);
```

```
-- Completed on 2022-12-12 16:07:08
```

```
--
-- PostgreSQL database dump complete
--
```

Приложение Е
(обязательное)

Шаблоны подстановки данных сигнала в запросе на добавление данных в БД объекта «EventLog»

Таблица Е.1 – Шаблоны подстановки данных сигнала в запросе на добавление данных в БД

| Шаблон | Тип данных | Описание |
|-------------|---|----------------------|
| {name} | text | Имя сигнала |
| {value} | text | Значение сигнала |
| {vartype} | integer, возможные значения: 1) VT_EMPTY = 0; 2) VT_NULL = 1; 3) VT_I2 = 2; 4) VT_I4 = 3; 5) VT_R4 = 4; 6) VT_R8 = 5; 7) VT_BSTR = 8; 8) VT_BOOL = 11; 9) VT_I1 = 16; 10) VT_LPSTR = 30 | Тип значения сигнала |
| {timestamp} | timestamp | Метка времени |
| {quality} | integer, возможные значения: 1) BAD = 0; 2) UNCERTAIN = 64; 3) GOOD = 192; 4) CONFIG_ERROR = 4; 5) LOCAL_OVERRIDE = 216; 6) LAST_KNOWN = 20 | Качество сигнала |

Приложение Ж
(справочное)
Пример скрипта создания БД для объекта «EventLog»

```
-- PostgreSQL database dump

-- Dumped from database version 14.4
-- Dumped by pg_dump version 15.0

-- Started on 2022-12-13 11:42:58

SET statement_timeout = 0;
SET lock_timeout = 0;
SET idle_in_transaction_session_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SELECT pg_catalog.set_config('search_path', '', false);
SET check_function_bodies = false;
SET xmloption = content;
SET client_min_messages = warning;
SET row_security = off;

--
-- TOC entry 4 (class 2615 OID 2200)
-- Name: public; Type: SCHEMA; Schema: -; Owner: -
--

-- *not* creating schema, since initdb creates it

SET default_table_access_method = heap;

--
-- TOC entry 212 (class 1259 OID 16523)
-- Name: events; Type: TABLE; Schema: public; Owner: -
--

CREATE TABLE public.events (
    id bigint NOT NULL,
    value text,
    vartype_id integer,
    "timestamp" timestamp with time zone,
    quality_id integer,
    name text
);

--
-- TOC entry 211 (class 1259 OID 16522)
-- Name: Events_id_seq; Type: SEQUENCE; Schema: public; Owner: -
--

CREATE SEQUENCE public."Events_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
```

```

CACHE 1;

--
-- TOC entry 3325 (class 0 OID 0)
-- Dependencies: 211
-- Name: Events_id_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: -
--

ALTER SEQUENCE public."Events_id_seq" OWNED BY public.events.id;

--
-- TOC entry 210 (class 1259 OID 16515)
-- Name: sig_quality; Type: TABLE; Schema: public; Owner: -
--

CREATE TABLE public.sig_quality (
  id integer NOT NULL,
  name text NOT NULL
);

--
-- TOC entry 209 (class 1259 OID 16508)
-- Name: sig_vartypes; Type: TABLE; Schema: public; Owner: -
--

CREATE TABLE public.sig_vartypes (
  id integer NOT NULL,
  name text NOT NULL
);

--
-- TOC entry 3172 (class 2604 OID 16526)
-- Name: events id; Type: DEFAULT; Schema: public; Owner: -
--

ALTER TABLE ONLY public.events ALTER COLUMN id SET DEFAULT
nextval('public."Events_id_seq"::regclass);

--
-- TOC entry 3178 (class 2606 OID 16530)
-- Name: events Events_pkey; Type: CONSTRAINT; Schema: public; Owner: -
--

ALTER TABLE ONLY public.events
  ADD CONSTRAINT "Events_pkey" PRIMARY KEY (id);

--
-- TOC entry 3176 (class 2606 OID 16521)
-- Name: sig_quality SigQuality_pkey; Type: CONSTRAINT; Schema: public; Owner: -
--

ALTER TABLE ONLY public.sig_quality
  ADD CONSTRAINT "SigQuality_pkey" PRIMARY KEY (id);

```

```
--  
-- TOC entry 3174 (class 2606 OID 16514)  
-- Name: sig_vartypes SigVartypes_pkey; Type: CONSTRAINT; Schema: public; Owner: -  
--
```

```
ALTER TABLE ONLY public.sig_vartypes  
  ADD CONSTRAINT "SigVartypes_pkey" PRIMARY KEY (id);
```

```
--  
-- TOC entry 3179 (class 2606 OID 16536)  
-- Name: events Events_quality_id_fkey; Type: FK CONSTRAINT; Schema: public; Owner: -  
--
```

```
ALTER TABLE ONLY public.events  
  ADD CONSTRAINT "Events_quality_id_fkey" FOREIGN KEY (quality_id) REFERENCES  
public.sig_quality(id);
```

```
--  
-- TOC entry 3180 (class 2606 OID 16531)  
-- Name: events Events_vartype_id_fkey; Type: FK CONSTRAINT; Schema: public; Owner: -  
--
```

```
ALTER TABLE ONLY public.events  
  ADD CONSTRAINT "Events_vartype_id_fkey" FOREIGN KEY (vartype_id) REFERENCES  
public.sig_vartypes(id);
```

```
-- Completed on 2022-12-13 11:43:00
```

```
--  
-- PostgreSQL database dump complete  
--
```

Приложение И (справочное)

Пример объекта «MODBUS_CLIENT»

```

<MODBUS_CLIENT name="ModBus" host="ASUE_USPD2">
  <property name="TimeOut" value="10"/>
  <property name="IsActiveStart" value="1"/>
  <tag name="IsActiveTag" value="IsActive" vartype="VT_BOOL"/>
  <tag name="Line.BEMPRU-TL2BKZ.NotConnected" value="Line.BEMPRU-
TL2BKZ.NotConnected" vartype="VT_BOOL" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.NotConnected"/>
  <tag name="Line.BEMPRU-TL2BKZ.IU" value="Line.BEMPRU-TL2BKZ.IU"
vartype="VT_BOOL" typebuf="MEM" sendto="OPC\ModBus.Line.BEMPRU-
TL2BKZ.IU"/>
  <tag name="Line.BEMPRU-TL2BKZ.REQ_RESULT" value="Line.BEMPRU-
TL2BKZ.REQ_RESULT" vartype="VT_LPSTR" sendto="OPC\ModBus.Line.BEMPRU-
TL2BKZ.REQ_RESULT"/>
  <tag name="Line.BEMPRU-TL2BKZ.TSCmd" value="Line.BEMPRU-
TL2BKZ.TSCmd" vartype="VT_BOOL" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.TSCmd"/>
  <tag name="Line.BEMPRU-TL2BKZ.OSCAUTO" value="Line.BEMPRU-
TL2BKZ.OSCAUTO" vartype="VT_BOOL" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.OSCAUTO"/>
  <tag name="Line.BEMPRU-TL2BKZ.OSCREFRESHLIST" value="Line.BEMPRU-
TL2BKZ.OSCREFRESHLIST" vartype="VT_I4" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.OSCREFRESHLIST"/>
  <tag name="Line.BEMPRU-TL2BKZ.OSCStatus" value="Line.BEMPRU-
TL2BKZ.OSCStatus" vartype="VT_LPSTR" sendto="OPC\ModBus.Line.BEMPRU-
TL2BKZ.OSCStatus"/>
  <tag name="Line.BEMPRU-TL2BKZ.OSCTODL" value="Line.BEMPRU-
TL2BKZ.OSCTODL" vartype="VT_I4" sendto="OPC\ModBus.Line.BEMPRU-
TL2BKZ.OSCTODL"/>
  <tag name="Line.BEMPRU-TL2BKZ.Svet_BEMP_32172" value="Line.BEMPRU-
TL2BKZ.Svet_BEMP_32172" vartype="VT_I4" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.Svet_BEMP_32172"/>
  <tag name="Line.BEMPRU-TL2BKZ.Svet_BEMP_16011" value="Line.BEMPRU-
TL2BKZ.Svet_BEMP_16011" vartype="VT_I4" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.Svet_BEMP_16011"/>
  <tag name="Line.BEMPRU-TL2BKZ.Vid_neyt" value="Line.BEMPRU-
TL2BKZ.Vid_neyt" vartype="VT_I4" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.Vid_neyt"/>
  <tag name="Line.BEMPRU-TL2BKZ.U0_srab_ZZ" value="Line.BEMPRU-
TL2BKZ.U0_srab_ZZ" vartype="VT_I4" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.U0_srab_ZZ"/>
  <tag name="Line.BEMPRU-TL2BKZ.Tip_otst" value="Line.BEMPRU-
TL2BKZ.Tip_otst" vartype="VT_I4" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.Tip_otst"/>
  <tag name="Line.BEMPRU-TL2BKZ.Soot_I0I1" value="Line.BEMPRU-
TL2BKZ.Soot_I0I1" vartype="VT_I4" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.Soot_I0I1"/>
  <tag name="Line.BEMPRU-TL2BKZ.Soot_I0I2" value="Line.BEMPRU-
TL2BKZ.Soot_I0I2" vartype="VT_I4" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.Soot_I0I2"/>
  <tag name="Line.BEMPRU-TL2BKZ.Soot_I2I1" value="Line.BEMPRU-
TL2BKZ.Soot_I2I1" vartype="VT_I4" typebuf="MEM"
sendto="OPC\ModBus.Line.BEMPRU-TL2BKZ.Soot_I2I1"/>
  <tag name="Line.BEMPRU-TL2BKZ.Events.EVTime" value="Line.BEMPRU-
TL2BKZ.Events.EVTime" vartype="VT_LPSTR" sendto="OPC\ModBus.Line.BEMPRU-
TL2BKZ.Events.EVTime"/>

```

```

    <tag name="Line.BEMPRU-TL2BKZ.SetsCheckEnable" value="Line.BEMPRU-
    TL2BKZ.SetsCheckEnable" vartype="VT_BOOL" sendto="OPC\ModBus.Line.BEMPRU-
    TL2BKZ.SetsCheckEnable"/>
    <tag name="Line.BEMPRU-TL2BKZ.SetsCheckPeriod" value="Line.BEMPRU-
    TL2BKZ.SetsCheckPeriod" vartype="VT_I4" sendto="OPC\ModBus.Line.BEMPRU-
    TL2BKZ.SetsCheckPeriod"/>
    <tag name="Line.BEMPRU-TL2BKZ.TimerSetsCheck" value="Line.BEMPRU-
    TL2BKZ.TimerSetsCheck" vartype="VT_I4" sendto="OPC\ModBus.Line.BEMPRU-
    TL2BKZ.TimerSetsCheck"/>
    <tag name="Line.BEMPRU-TL2BKZ.Sets.Get_Sets" value="Line.BEMPRU-
    TL2BKZ.Sets.Get_Sets" vartype="VT_BOOL" sendto="OPC\ModBus.Line.BEMPRU-
    TL2BKZ.Sets.Get_Sets"/>
    <tag name="Line.BEMPRU-TL2BKZ.Sets.Write_Sets" value="Line.BEMPRU-
    TL2BKZ.Sets.Write_Sets" vartype="VT_BOOL" sendto="OPC\ModBus.Line.BEMPRU-
    TL2BKZ.Sets.Write_Sets"/>
</MODBUS_CLIENT>

```

Приложение К
(справочное)

Описание служебных параметров для устройств Modbus в зависимости от производителей и их типов

Таблица К.1 – Описание служебных параметров для устройств Modbus в зависимости от производителей и их типов

| Имя тега | Тип данных | Описание |
|---|------------|---|
| СИРИУС-2-В (тест), СИРИУС-2-С (тест), СИРИУС-2-Л (тест), СИРИУС-2-ГС (тест) | | |
| Zone_GetWrite.GetValues_X | VT_BOOL | Тег подачи команды на считывание зоны. Создается, если в конфигурации разрешена работа с зоной чтения/записи по запросу. Для подачи команды следует записать в тег значение «1». Доступ – RW |
| Zone_GetWrite.WriteValues_X | VT_BOOL | Тег подачи команды на запись зоны. Создается, если в конфигурации разрешена работа с зоной чтения/записи по запросу. Для подачи команды следует записать в тег значение «1». Доступ – RW |
| Sets.ActiveSet | VT_I4 | Тег для выбора активного набора уставок (для выбора необходимо записать номер набора уставок) |
| Sepam-2000, Sepam-80, БМЦС, MICROLOGIC, СИРИУС (старого типа), СИРИУС-2-В (тест), СИРИУС-2-С (тест), СИРИУС-2-Л (тест), СИРИУС-2-ГС (тест), MiCOM P123, MiCOM P124, MiCOM P127 | | |
| Events.EVTime | VT_BSTR | Тег метки времени и описания события. Содержит текстовое описание и метку времени события. Создается только, если считывание событий разрешено в конфигурации объекта. Доступ – R |
| Sepam-2000, Sepam-80, Sepam-40 | | |
| Sets.ANSIx.Get_Set | VT_BOOL | Тег подачи команды на считывание уставки. ANSI x – номер защиты. Создается, если в конфигурации разрешена работа с уставками. Для подачи команды следует записать в тег значение «1». Доступ – RW |
| Sets.ANSIx.Write_Set | VT_BOOL | Тег подачи команды на запись уставки. ANSI x – номер защиты. Создается, если в конфигурации разрешена работа с уставками. Для подачи команды следует записать в тег значение «1». Доступ – RW |
| Устройства производства НТЦ «Механотроника» | | |
| Commands.Read_Info | VT_BOOL | Тег подачи команды на считывание с устройств накопительной информации. Создается, если считывание накопительной информации разрешено в конфигурации объекта. Для |

| Имя тега | Тип данных | Описание |
|---|------------|---|
| | | подачи команды следует записать в тег значение «1». Доступ – RW |
| Sets.Get_Sets | VT_BOOL | Тег подачи команды на считывание уставок. Создается, если в конфигурации разрешена работа с уставками. Для подачи команды следует записать в тег значение «1». Доступ – RW |
| Sets.Write_Sets | VT_BOOL | Тег подачи команды на запись уставок. Создается, если в конфигурации разрешена работа с уставками. Для подачи команды следует записать в тег значение «1». Доступ – RW |
| Устройства производства НТЦ «Механотроника», кроме БМЦС | | |
| Commands.Damage_Num | VT_I4 | Тег команды на считывание информации о повреждении. В тег записывается номер повреждения. Создается, если считывание накопительной информации разрешено в конфигурации объекта. Доступ – RW |
| БМЦС | | |
| Events.EVRead | VT_BSTR | Тег метки времени и описания события. Содержит текстовое описание и метку времени события. Работает по умолчанию если считывание событий не разрешено в конфигурации объекта. Доступ – R. Используется только для считывания событий вручную |
| БМРЗ, БМПА, СИРИУС-2-В (тест), СИРИУС-2-С (тест), СИРИУС-2-Л (тест), СИРИУС-2-ГС (тест), СИРИУС (старого типа) | | |
| INFO.NumberLastDamage | VT_I4 | Тег номера последнего срабатывания. Создается, если в конфигурации объекта выставлен соответствующий параметр INFO = «1». Доступ – R |
| INFO.MaxNumberDamage | VT_I4 | Тег значения максимально допустимого количества срабатываний, которые может хранить терминал. Считывается с терминала только один раз при включении. Создается, если в конфигурации объекта выставлен соответствующий параметр INFO = «1». Доступ – R. <i>(Отсутствует в терминале СИРИУС (старого типа))</i> |
| INFO.Commands.GetInfoDamage_ByNumber | VT_I4 | Тег команды на считывание информации о срабатывании. В тег записывается номер срабатывания в пределах значения, указанного в теге INFO.MaxNumberDamage. Создается, если в конфигурации объекта выставлен соответствующий параметр INFO = «1». Доступ – RW |

| Имя тега | Тип данных | Описание |
|--|------------|---|
| СИРИУС (старого типа) | | |
| RemoteControl.Reset_Signaling | VT_BOOL | Командный тег телеуправления для сброса сигнализации. Создается всегда. Доступ – RW |
| RemoteControl.Turn_OFF | VT_BOOL | Командный тег телеуправления для отключения ключа дистанционно по ЛС. Создается всегда. Доступ – RW |
| RemoteControl.Turn_ON | VT_BOOL | Командный тег телеуправления для включения ключа дистанционно по ЛС. Создается всегда. Доступ – RW |
| MiCOM P123, MiCOM P124, MiCOM P127 | | |
| Info.NumberLastDamage | VT_I4 | Тег номера последнего срабатывания. Создается, если в конфигурации объекта выставлен соответствующий параметр INFO = «1». Доступ – R |
| Info.MaxNumberDamage | VT_I4 | Тег значения максимально допустимого количества срабатываний, которые может хранить терминал. Создается, если в конфигурации объекта выставлен соответствующий параметр INFO = «1». Доступ – R |
| Info.Commands. GetInfoDamage_ByNumber | VT_I4 | Тег команды на считывание информации о срабатывании. В тег записывается номер срабатывания в пределах значения, указанного в теге Info.MaxNumberDamage. Создается, если в конфигурации объекта выставлен соответствующий параметр INFO = «1». Доступ – RW |
| Info.AUTO_MODE | VT_BOOL | Тег команды на включение автоматического вывода данных самого нового срабатывания. По умолчанию значение «0». Создается, если в конфигурации объекта выставлен соответствующий параметр INFO = «1». Доступ – RW |

Приложение Л
(справочное)
Параметры объекта «MODBUS_DLL_CONFIG»

Таблица Л.1 – Параметры конфигурации объекта «MODBUS_DLL_CONFIG»

| Параметр | Описание |
|----------------------------|---|
| ServerOFF | Отключение объекта. При значении «1» опрос всех устройств прекращается. Определяет начальное значение тега «ServerOFF» (параметр конфигурирования объекта «MODBUS»). Необязательный параметр, значение по умолчанию «1» (отключен) |
| DemoOFF | Отключение демонстрационного режима. При значении «1» демонстрационный режим отключается (максимальное количество поддерживаемых терминалов – один). Необязательный параметр. Значение по умолчанию «0» (включен) |
| CheckKeyTimeout | Период опроса наличия ключа GUARDANT в секундах. Значение по умолчанию = «60». Диапазон допустимых значений: 5...600 |
| LogLevel | Включение/отключение расширенного/отладочного лога при первоначальном запуске. Значение «1» – обычный лог, значение «2» – расширенный/отладочный лог |
| NumLogs | Количество логов. При отсутствии данного параметра по умолчанию задаются значения: в обычном режиме «3», в отладочном режиме «100». При введении данного параметра количество логов в режиме обычном и в режиме отладки будут соответствовать заданному параметру, т.е. количество логов для разных режимов будут одинаковыми, согласно значению параметра «NumLogs». Диапазон допустимых значений: 1...65535 |
| SizeOneLogInKb | Размер одного лога в килобайтах. При отсутствии данного параметра по умолчанию задаются значения: в обычном режиме «3000», в отладочном режиме «50000». При введении данного параметра размер одного лога в режиме обычном и в режиме отладки будут соответствовать заданному параметру, т.е. размер одного лога для разных режимов будет одинаковым, согласно значению параметра «SizeOneLogInKb». Диапазон допустимых значений: (50 – 50000) Кб |
| Секция <Line> | |
| Name | Имя линии может быть произвольным (допускаются только латинские буквы, цифры и некоторые символы). По умолчанию равен «LineX», где X – порядковый номер линии |
| TCP | Параметр для настройки типа обмена. Значение «1» – обмен по TCP. В случае отсутствия параметра или значения параметра, отличного от «1», параметры настройки TCP обмена игнорируются, и объект будет пытаться обмениваться данными по последовательному порту. Если значение равно «1», то параметры настройки последовательного порта указывать не обязательно (могут отсутствовать) |
| ModbusTCP | Значение «1» – обмен с устройством будет осуществляться по протоколу ModbusTCP. По умолчанию равен «0» |

| Параметр | Описание |
|-------------------|---|
| ComNumber | Номер последовательного порта, существующего в системе. Если параметр TSP равен «1», то данный параметр игнорируется (может отсутствовать в секции) |
| BaudRate | Скорость последовательного порта. Если параметр TSP равен «1», то данный параметр игнорируется (может отсутствовать в секции) |
| DataBits | Число бит данных. Если параметр TSP равен «1», то данный параметр игнорируется (может отсутствовать в секции) |
| StopBits | Число стоповых битов. Если параметр TSP равен «1», то данный параметр игнорируется (может отсутствовать в секции) |
| Parity | Четность. Может принимать значения: «0» – без проверки четности (none), «1» – нечет (odd) и «2» – чёт (even). Если параметр TSP равен «1», то данный параметр игнорируется (может отсутствовать в секции) |
| IPAddr | Основной IP-адрес опрашиваемого устройства (преобразователя интерфейсов). Если параметр TSP равен «0», то данный параметр игнорируется (может отсутствовать в секции). Если параметр «IPAddrReserve» = «1» объект пытается подключиться три раза по основному IP-адресу. Если безрезультатно, пытается подключиться к резервному IP-адресу. Иначе, в случае если «IPAddrReserve» = «0» или если данный параметр отсутствует, объект пытается подключиться до тех пор, пока не подключиться или пока объект не остановят. Период попыток подключения составляет 10 с |
| IPAddrReserve | Резервный IP-адрес опрашиваемого устройства (преобразователя интерфейсов). Если параметр TSP равен «0», то данный параметр игнорируется (может отсутствовать в секции). Если параметр «IPAddrReserve» = «1» объект пытается подключиться три раза по резервному IP-адресу. Если безрезультатно, пытается подключиться три раза к основному IP-адресу. И так далее. Период попыток подключения составляет 10 с |
| Port | IP-порт опрашиваемого устройства (преобразователя интерфейсов). Если параметр TSP равен «0», то данный параметр игнорируется (может отсутствовать в секции) |
| InUse | Обязательный параметр. Флаг опроса всей линии при старте объекта. Значение «1» – в опросе, «0» – не опрашивать |
| OffTagInUse | Необязательный параметр. Флаг перехода тега «InUse» в невидимый режим. Значение «1» – «InUse» невидимый тег, «0» – видимый. Если параметр отсутствует тег «InUse» видимый |
| MaxTimeout | Обязательный параметр. Таймаут ожидания ответа от устройства в миллисекундах |
| RetryLimit | Обязательный параметр. Количество повторной отправки того же самого запроса при ошибке |
| TimeSynchro | Обязательный параметр. Флаг синхронизации времени на устройстве. Значение «1» – синхронизировать, «0» – не синхронизировать |
| TimeSynchroPeriod | Период синхронизации времени на устройстве. Если параметр «TimeSynchro» не равен «1», то данный параметр игнорируется (может отсутствовать в секции) |

| Параметр | Описание |
|-----------------------|---|
| TSCR | Значение коррекции времени в миллисекундах. Прибавляется к значению времени в пакете синхронизации. Необязательный параметр. По умолчанию значение «0» |
| Секция <Line/Station> | |
| Name | Имя станции. Может быть произвольным (допускаются только латинские буквы, цифры и некоторые символы). По умолчанию равен «StationX», где X – порядковый номер станции |
| InUse | Обязательный параметр. Флаг опроса устройства при старте объекта. Значение «1» – в опросе, «0» – не опрашивать |
| ModbusAddr | Обязательный параметр. Modbus адрес устройства |
| DeviceType | Уникальный цифровой идентификатор типа устройства. Обязательный параметр для специализированных устройств. Необязательный параметр для стандартных устройств, в случае, когда присутствует параметр «DeviceName», а в шаблоне устройств в блоке <Unit> присутствует параметр «Name». Если в конфигурационном файле параметр «DeviceType» указан, номер типа устройства ищется в шаблоне: параметр «Type» блока <Unit>. Если указанный в конфигурационном файле номер типа отсутствует в шаблоне, объект останавливается с ошибкой. Если в конфигурационном файле параметр «DeviceType» не указан, объект проверяет параметр «DeviceName». Если параметр «DeviceName» не указан, объект останавливается с ошибкой. Иначе, объект сравнивает в шаблоне устройства в блоке <Unit> параметр «Name» с параметром конфигурационного файла «DeviceName». Если данные параметры не совпадают, объект останавливается с ошибкой (тип устройства не указан и имя устройства не найдено в шаблоне) |
| DeviceName | Уникальный строковый идентификатор типа устройства. Необязательный параметр для специализированных устройств, в случае, когда присутствует параметр «DeviceType» (описание параметра «DeviceType» выше в данной таблице) |
| DACDelay | Задержка на отправку запроса главной зоны в миллисекундах. Необязательный параметр, значение по умолчанию «200» |
| ZoneGetWrite | Необязательный параметр. Флаг вывода зоны чтения/записи по запросу устройства в OPC-теги в группе «Zone_GetWrite». Значение «1» – выводить уставки в теги, «0» – не выводить. По умолчанию «0». Данный параметр предназначен только для устройств СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС |
| SETS | Необязательный параметр. Флаг вывода уставок устройства в OPC-теги в группе «Sets». Значение «1» – выводить уставки в теги, «0» – не выводить. По умолчанию «0» |
| INFO | Необязательный параметр. Флаг вывода информации об авариях/срабатываниях устройства в OPC-теги в группе «INFO». Значение «1» – выводить информацию в теги, значение «0» – не выводить. Данный параметр предназначен только для устройств БМРЗ, БМПА, БМЦС, СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС, СИРИУС (старого типа), MiCOM P123, MiCOM P124, MiCOM P127. По умолчанию «0» |

| Параметр | Описание |
|--------------|---|
| IFREQ | Необязательный параметр. Определяет частоту запросов для считывания номера последнего срабатывания в устройстве в миллисекундах. По умолчанию равен 1000. Данный параметр предназначен только для устройств СИРИУС-2-В, СИРИУС-2-С, СИРИУС-2-Л, СИРИУС-2-ГС, СИРИУС (старого типа), MiCOM P123, MiCOM P124, MiCOM P127 |
| OSCPATH | Необязательный параметр. Путь сохранения осциллограмм и событий, считанных с устройства. В случае его отсутствия осциллограммы сохраняются в C:\Relematika\OSC\ИМЯ_ЛИНИИ\ИМЯ_СТАНЦИИ. Есть возможность использовать следующие макросы: [YYYY] – год, [MM] – месяц, [DD] – день, [HH] – час, [NN] – минута, [ADR] – адрес устройства |
| CMDID | Необязательный параметр. Уникальные цифровые идентификаторы (ID) командных тегов для устройств. Команды в виде тегов заносятся в группу «» |
| OSCDL | Необязательный параметр. Флаг автоматического считывания осциллограмм с устройства. Значение «0» – не скачивать осциллограммы с устройства, любое другое значение или отсутствие параметра – скачивать |
| EVENTS | Необязательный параметр. Значение «1» разрешает работу с журналом событий терминала. По умолчанию равен «0» |
| EAUTOACK | Необязательный параметр. Значение «1» разрешает работу в режиме автоквитирования (считал, значит квитировал событие). Значение «0» переключает считывание событий терминала в режим «ручного» квитирования (считал, сохранил на жесткий диск, отквитировал). По умолчанию равен «0». Данный параметр предназначен только для устройств MiCOM P123, MiCOM P124, MiCOM P127 |
| EVTimeFromPC | Необязательный параметр. При значении «1» время обновления тега «EVTime» выставляется из ПК. По умолчанию равен «0» |
| EFREQ | Необязательный параметр. Определяет частоту запросов к журналу событий устройства в миллисекундах. По умолчанию равен 1000 |
| EUPDT | Необязательный параметр. Определяет частоту обновления событийных тегов в миллисекундах. По умолчанию равен 1000 |
| SessionQuota | Квота количества запросов к одному терминалу за сессию. Сессия терминала – это период времени, когда идет опрос определенного устройства на линии. Необязательный параметр. Значение по умолчанию – 2 (рекомендуется). ВНИМАНИЕ!!! Изменение этого параметра существенно влияет на скорость опроса всех терминалов на линии |
| PSW | Необязательный параметр. Необходим для автоматического ввода пароля при попытке записи значений только для терминалов СИРИУС v3. PSW = «9999» – запись значения «9999» в поле пароля, «0» – не вводить. По умолчанию равен «0» |
| WR_Mode | Флаг включения режима записи значений в теги для других устройств, поддерживающих протокол Modbus RTU и ModbusTCP. |

| Параметр | Описание |
|---|---|
| | Необходимо использовать в сочетании с параметром тега в конфигурации «DirectAccess» = «1», а также параметром зоны «FuncWR» = «x» в шаблоне, где «x» – номер функции для записи (5, 6, 15, 16) |
| <Line/Station/Tags> – секция тэгов (зон) | |
| <Line/Station/Tags/Tag> – секция тега (зоны) | |
| RegularBegin | Обязательный параметр. Номер начальной зоны опроса группы зон. Не может быть в секции вместе с параметрами «DemandXXX» |
| RegularSize | Обязательный параметр. Количество зон, опрашиваемых одним запросом. Не может быть в секции вместе с параметрами «DemandXXX» |
| RegularFreq | Обязательный параметр. Желательный период опроса зоны или группы зон в миллисекундах. Не может быть в секции вместе с параметрами «DemandXXX» |
| MainDAC | Обязательный параметр. Флаг главной зоны (группы зон). Означает опрос данной зоны (группы зон) сразу после подачи управляющей команды на устройство. Не может быть в секции вместе с параметрами «DemandXXX» |
| DemandBegin | Необязательный параметр. Номер начальной зоны опроса группы зон по запросу (команде). Не может быть в секции вместе с параметрами «RegularXXX» |
| DemandSize | Обязательный параметр. Количество зон, опрашиваемых одним запросом. Не может быть в секции вместе с параметрами «RegularXXX» |
| ValueFromEvents | Обновление дискретных OPC-тегов через новые события. Работает только с дискретными входами, логическими уравнениями, телесигнализацией и КТС. Значение по умолчанию «0» (выключен). Только для терминалов СЕПАМ(!) |
| DirectAccess | В сочетании с параметром «WR_Mode» = «1» доступны функции записи 5, 6, 15, 16 (битовые и регистровые зоны). Необходимо дополнительно в шаблоне в зоне указать параметр «FuncWR» или «FuncW». В сочетании с параметром «WR_Mode» = «0» (или отсутствием) актуален только для битовых зон. Только для зон с регулярным опросом. Количество зон должно быть один. Позволяет писать в битовые теги зоны без дополнительных команд и действий. Для записи используется функция 5 протокола Modbus |
| OnlyWrite | Параметр необходим в случаях, когда по указанному адресу памяти возможно производить только запись значений (отсутствует возможность чтения). В этом случае в конфигурации необходимо задать параметры в следующем порядке: <Tag RegularBegin="1" RegularSize="1" RegularFreq="0" DirectAccess="1" OnlyWrite="1"/>. Все остальные условия параметров «DirectAccess», «WR_Mode», «Func», «FuncW» должны быть соблюдены |
| Следующие параметры применимы, только если указан <Line/Station DeviceType ="747"> или <Line/Station DeviceType ="150"> | |
| BitBegin | Начальный адрес бита или битов. Например: «BitBegin» = «0x100» |
| BitsCount | Обязателен, если указан параметр «BitBegin». Количество битов для чтения. Например: «BitsCount» = «25» |

| Параметр | Описание |
|-----------|--|
| RegBegin | Начальный адрес регистра или регистров для чтения |
| RegsCount | Обязателен, если указан параметр «RegBegin». Количество регистров для чтения |
| Func | Обязательный параметр. Могут быть указаны функции 1, 2, 3, 4 |
| Freq | Обязательный параметр. Период опроса заданных адресов в миллисекундах |
| DataType | Обязательный параметр, только если заданы «RegBegin» и «RegsCount», иначе не учитывается. Определяет тип данных в устройстве для конфигурируемых параметров, т.е. все данные в диапазоне адресов будут интерпретироваться как данный тип |
| OPCDT | Обязательный параметр, только если заданы «RegBegin» и «RegsCount», иначе не учитывается. Задает тип OPC-тега |
| Size | Обязательный параметр, только если заданы «RegBegin» и «RegsCount», иначе не учитывается. Задает размер типа данных в регистрах (2 байта). Например, если указано 2 регистра, то считывается 4 байта |
| Writable | Возможные значения: 1/0. Необязательный параметр. Задает возможность записи в тег для указанного диапазона адресов. Для записи битов используется функция 5, для записи регистров используется функция 16 |

Приложение М
(обязательное)
Список сокращений

| | |
|--------|---|
| АРМ | – автоматизированное рабочее место |
| АСУ | – автоматизированная система управления |
| АСУ ТП | – автоматизированная система управления технологическим процессом |
| БД | – база данных |
| КА | – коммутационный аппарат |
| ПК | – персональный компьютер |
| ПО | – программное обеспечение |
| ПТК | – программно-технический комплекс |
| ПЭВМ | – персональная электронно-вычислительная машина |
| РФ | – Российская Федерация |
| ССПИ | – система сбора и передачи информации |
| ТУ | – телеуправление |