

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«РАКУРС-ИНЖИНИРИНГ»



Программный комплекс «Раскада»

Руководство пользователя

САНКТ-ПЕТЕРБУРГ
2024

Содержание

1 Общие сведения	2
2 Системные требования	3
3 Установка и удаление	4
3.1 Установка программного комплекса	4
3.2 Установка серверных служб	4
3.3 Удаление программного комплекса	5
4 Лицензионная политика и регистрация	6
4.1 Запрос демонстрационных ключей	6
4.2 Полная регистрация приложений	6
4.2.1 Серийные номера	6
4.2.2 Запрос регистрационных ключей	6
4.2.3 Применение регистрационных ключей	7
5 Состав программного комплекса	8
5.1 Среда разработки проектов RSP	8
5.2 Среда исполнения проектов RSP	8
5.3 Графический редактор мнемосхем Easy Graphics Editor	8
5.4 Служба записи и чтения архивных данных Fins Logger	8
5.5 Служба удаленного доступа к архивным данным Fins Mail	9
5.6 Служба сбора и обработки данных Fins Router	9
5.7 DCOM сервер протокола OPC DA 2.0	9
5.8 Программа редактирования шаблонов отчётов FastReport	9
5.9 Плагин сервера событий alrmsrv	9
5.10 Плагин сервера МЭК 60870-5-104	10
5.11 Плагин клиента МЭК 60870-5-104	10
5.12 Клиент для получения данных по протоколу OPC DA 2.0	10
5.13 Клиент для получения данных по протоколу OPC HDA 2.0	10
5.14 Плагин формирования предупреждений warnings	10
5.15 Библиотечный модуль сервера OPC UA opcuasrv	11
5.16 Библиотечный модуль клиента OPC UA opcuacli	11
5.17 Плагин для записи итоговых значений sumwrite	11
5.18 Тестовые утилиты	12
5.19 Утилиты регистрации компонентов	12
5.20 Вспомогательные программные компоненты и файлы	12
5.21 Остальное	12
6 Информация о разработчике и контакты	13
7 Перечень сокращений	14

1 Общие сведения

Программный комплекс (ПК) «Раскада» является SCADA-системой и предназначен для разработки программного обеспечения верхнего уровня автоматизированных систем управления (АСУ) с любой архитектурой:

- автономных АСУ;
- клиент-серверных АСУ;
- распределённых АСУ.

В совокупности с конфигурационными проектами АСУ, которые разрабатываются пользователями согласно данной документации, ПК решает следующие задачи:

- обмен информацией с устройствами, входящими в состав АСУ, то есть с ПЛК, измерительными приборами и модулями ввода/вывода;
- обработка информации, поступающей от промышленных объектов;
- отображение информации на экранах мониторов в понятной для человека форме;
- ведение базы данных (истории) с технологической информацией;
- аварийная и предупредительная сигнализация, управление тревожными сообщениями;
- формирование отчётов о ходе технологического процесса;
- дистанционное управление исполнительными механизмами.

ПК «Раскада» является пакетом гибких инструментов автоматизации и не решает перчисленные задачи в отрыве от конфигурационных проектов АСУ, разработанных пользователями. Таким образом, ПК и конкретный конфигурационный проект становятся неотъемлемой частью АСУ в качестве программного обеспечения верхнего уровня.

ПК можно условно разделить на серверную и клиентскую части. В серверную часть комплекса входят:

- служба записи и чтения архивных данных Fins Logger с плагинами (подробно о ней в руководстве FinsLoggerManual.pdf)
- служба удалённого доступа к архивным данным Fins Mail (подробно о ней в руководстве FinsMailManual.pdf)
- служба сбора и обработки данных Fins Router с плагинами и подключаемыми библиотечными модулями (подробно о ней в руководстве FinsRouterManual.pdf)

В клиентскую часть комплекса входят:

- графический редактор мнемосхем Easy Graphics Editor (подробно о нём в руководстве EasyEditManual.pdf)
- среда разработки и среда исполнения проектов RSP (подробно о них в руководстве RSPManual.pdf)

2 Системные требования

Общие минимальные требования к конфигурации компьютера для использования ПК «Раскада»:

- 4-х ядерный процессор с тактовой частотой 2,33 ГГц;
- 4/8 Гб оперативной памяти для 32/64-разрядных ОС;
- 1 Гб свободного места на накопителе.

Для серверной части комплекса могут быть дополнительные требования в зависимости от назначения компьютера в конфигурационном проекте АСУ:

- для возможности подключения служб к ПЛК — наличие необходимых портов типа RS232/RS422/RS485/Ethernet;
- для хранения больших объемов архивных данных — от 100 Гб свободного места на накопителе.

Для клиентской части комплекса требуется наличие видеокарты и монитора, поскольку работа с ней реализуется через пользовательский интерфейс. Кроме того, требования к конфигурации компьютера, на котором запускается среда исполнения проектов RSP, напрямую зависят от сложности конфигурационного проекта АСУ и требований к человеко-машинному интерфейсу, которые выражаются в:

- общем объеме проекта RSP (количестве и сложности страниц и мнемосхем);
- количестве и сложности скриптов, используемых в проекте RSP;
- числе опрашиваемых подключённых устройств и частоте их опроса.

В зависимости от вышеперечисленных условий может потребоваться более мощный процессор и больше оперативной памяти.

Поддерживаемые операционные системы (ОС)¹:

- Windows 8, 8.1 Home/Pro/Enterprise;
- Windows 10 Home/Pro/Enterprise/loT;
- Windows 11 Home/Pro/Enterprise/loT;
- Windows Server 2012/2012 R2/2016;
- Debian 8.0 и выше;
- Ubuntu 10.04 и выше;
- Astra Linux CE/SE 1.7.0 и выше.

Для ОС семейства Windows серверная часть комплекса также требует установки пакета Microsoft Visual C++ 2019 (или новее) Redistributable Package Hybrid x86 (для 32-х битных ОС) или x64 (для 64-х битных ОС). В современных операционных системах данный пакет уже установлен, однако для версий Windows 8 и 10 или Windows Server, выпущенных до 2020 года, его необходимо самостоятельно загрузить с сайта Microsoft и установить.

¹Состав ПК для ОС семейства Windows и ОС семейства Linux отличается. Подробнее об этом в разделе 5, а также индивидуальных руководствах каждого приложения.

3 Установка и удаление

3.1 Установка программного комплекса

Дистрибутивы ПК «Раскада» поставляются в виде архивов в кроссплатформенном формате ZIP. Доступны две версии дистрибутива:

- в 32-битной сборке под Windows;
- в 64-битной сборке под Linux.

Для установки комплекса достаточно распаковать дистрибутив ПК в каталог с правами на запись. Рекомендуются следующие каталоги установки:

- C:\Ruscada для Windows;
- /opt/ruscada для Linux.

Клиентская часть комплекса готова к использованию немедленно. Редактор мнемосхем EasyEdit, среда разработки проектов RSPDev и среда исполнения проектов RSPRun запускаются соответствующими исполняемыми файлами.

Для работы серверной части комплекса требуется корректная конфигурация, разработанная согласно индивидуальным руководствам каждого приложения, а также регистрация используемых приложений в качестве служб в системе.

3.2 Установка серверных служб

Служба записи и чтения архивных данных Fins Logger, служба удалённого доступа к архивным данным Fins Mail и служба сбора и обработки данных Fins Router — все приложения серверной части комплекса требуют для работы корректно оформленный файл конфигурации, а также предоставляют единый интерфейс командной строки, позволяющий устанавливать и запускать свои экземпляры.

Ключ Windows	Ключ Linux	Описание
/h	-h	(сокр. от help) вывести информацию о версии приложения, используемых параметрах и ключах командной строки, описанных в этой таблице
/r	-r	(сокр. от register) зарегистрировать приложение в качестве службы
/d	-d	(сокр. от debug) запустить приложение в режиме отладки, с выводом в консоль отладочной информации
/i=[name]	-i=[name]	(сокр. от instance) необязательный параметр, где [name] — имя экземпляра приложения, которое должно содержать только символы латинского алфавита, подчёркивания или цифры, и не начинаться на цифру; в случае регистрации как службы также должно не совпадать с именем других служб, установленных в системе; если параметр не указан, в качестве [name] будет использовано имя исполняемого файла
/c=[config]	-c=[config]	(сокр. от configuration) необязательный параметр, где [config] — путь до файла конфигурации приложения, который следует заключать в двойные кавычки, если он содержит пробелы; может быть указан как в относительном, так и абсолютном формате; если параметр не указан, в качестве [config] будет использован файл конфигурации, расположенный рядом с исполняемым файлом и имеющий имя экземпляра приложения
N/A	-daemon	запустить приложение как daemon-процесс
N/A	-stop	остановить daemon-процесс

При регистрации приложения в качестве службы она настраивается следующим образом:

- имя службы в системе — имя экземпляра приложения (параметр [name] , указанный при регистрации)
- отображаемое имя службы — наименование приложения плюс имя экземпляра в скобках
- при любом сбое работа службы восстанавливается немедленно (перезапуск через 0 сек.)
- для всех экземпляров служб создается отдельный журнал событий и ошибок в административной консоли Windows "Просмотр событий" в разделе "Журналы приложений и служб" - Ruscada (максимальный размер лога событий по умолчанию - 100 Мб)
- Настройка детализации вывода информации в лог событий производится через параметр LogLevel в секции Settings конфигурационного файла службы и может принимать следующие значения:
 - LogLevel=NOTSET - Выводить все сообщения кроме отладочных (значение по умолчанию)
 - LogLevel=ERROR - Выводить только сообщения об ошибках
 - LogLevel=WARNING - Выводить только сообщения об ошибках и предупреждениях
 - LogLevel=DEBUG - Выводить все сообщения включая отладочные

Для последующего управления установленной службой используется диспетчер управления службами (под Windows) или консольные команды для запуска, остановки, удаления служб и просмотра их состояния:

Команды Windows	Команды Linux
sc.exe start [name] sc.exe stop [name] sc.exe query [name] sc.exe delete [name]	systemctl enable [name] systemctl start [name] systemctl stop [name] systemctl status [name] systemctl disable [name]

3.3 Удаление программного комплекса

Для удаления комплекса достаточно выполнить следующие действия:

- остановить и удалить все зарегистрированные службы, используя консольные команды, приведённые выше;
- удалить каталог, в который был распакован дистрибутив ПК.

4 Лицензионная политика и регистрация

ПК «Раскада» защищён от копирования, и для его полноценной работы требуется зарегистрировать используемые приложения. В отсутствие регистрации все приложения серверной части комплекса, а также *среда исполнения проектов RSP* выводят ошибку в лог при запуске и прекращают работу через 1 час.

Приложения комплекса регистрируются по-отдельности — в том числе потому, что в зависимости от проекта АСУ могут использоваться в разном составе.

Кроме полной регистрации для приложений предусмотрен демонстрационный режим работы, в котором они не будут требовать перезапуска до 30 дней с момента генерации демо-ключа.

4.1 Запрос демонстрационных ключей

Для работы с приложениями комплекса в демонстрационном режиме следует обратиться по адресу `demo_request@rakurs.com` с указанием списка приложений, которые планируется использовать.

С компанией-разработчиком потребуется оформить *Соглашение об апробации*, после чего вам будет предоставлена актуальная стабильная сборка программного комплекса и набор демо-ключей.

Полученные демо-ключи необходимо прописать в соответствующие конфигурационные файлы каждого приложения в секции *Register* с ключом *SN*, например:

```
[Register]
SN=58A4-0D24-0653-D578-FAEA-282F-A0E4-B649
```

По истечении срока действия демо-ключа приложение сообщит об этом в лог и перейдёт в стандартный режим работы без регистрации.

4.2 Полная регистрация приложений

4.2.1 Серийные номера

Для регистрации каждого приложения требуется определить его серийный номер и запросить регистрационный ключ для этого номера.

Серийный номер представляет собой буквенно-цифровую последовательность, содержащую данные привязки конкретного экземпляра установленного приложения к конкретному компьютеру. При переустановке ОС или замене аппаратного обеспечения контроль соответствия серийного номера и ключа регистрации будет нарушен, и потребуется перерегистрация приложения.

Для определения серийного номера приложения следует запустить утилиту `register.exe`, входящую в состав ПК, на компьютере, где он установлен, и выбрать в выпадающем списке наименование приложения для регистрации. Серийный номер приложения выводится в поле *Serial Number*.

4.2.2 Запрос регистрационных ключей

Полученный серийный номер приложения следует использовать для запроса регистрационного ключа. Запросить ключ можно одним из следующих способов:

- обратиться в техническую поддержку по адресу `sc@rakurs.com`;
- самостоятельно через почтовый сервис регистрации по адресу `register@rakurs.spb.ru`.

При обращении в техническую поддержку убедитесь, что у вас есть действующий договор с компанией-разработчиком на поддержку проекта АСУ. Передайте весь список пар [*Наименование приложение / Серийный номер*] и ожидайте в ответ набор регистрационных ключей.

Для самостоятельного запроса ключей требуется предварительная регистрация в почтовом сервисе, которая обычно производится в процессе исполнения договора на реализацию проекта АСУ. Сообщите почтовый адрес, с которого будет осуществляться запрос

регистрационных ключей, компании-разработчику через ваши контакты по договору и дождитесь подтверждения регистрации в сервисе.

Чтобы получить регистрационный ключ приложения самостоятельно, следует отправить в сервис регистрации письмо с темой «serial *Serial Number*», где *Serial Number* — это серийный номер, полученный ранее, например:

```
serial 58A4-0D24-0653-D578-FAEA-282F-A0E4-B649
```

Содержимое письма значения не имеет. Для каждого серийного номера потребуется отправить отдельное письмо.

В ответном письме придёт ключ полной регистрации приложения — или ошибка, если вы не были вовремя зарегистрированы в сервисе, или же у вас закончился выделенный пакет лицензий. В этом случае уже придётся обращаться в техническую поддержку.

4.2.3 Применение регистрационных ключей

Полученные описанными выше способами регистрационные ключи необходимо прописать в соответствующие конфигурационные файлы каждого приложения в секции *Register* с ключом *Key*, например:

```
[Register]  
Key=25CF6F4B6D192B60-B92EDA57129106E0
```

Допускается вместо ключа *Key* использовать имя компьютера, а также одновременно указывать в этой секции несколько регистрационных ключей для обеспечения работы приложения на переносимом носителе, или при запуске приложения по сети, например:

```
[Register]  
RS01=25CF6F4B6D192B60-B92EDA57129106E0  
RS02=F1632F957EC7B0C5-149550F11E3A02C0  
RS03=EC17BF5AB41FE52B-84B3B7F513F96780
```


5 Состав программного комплекса

5.1 Среда разработки проектов RSP

Назначение: среда разработки *RSP Developer* предназначена для создания визуального интерфейса пользователя для систем контроля, управления и сбора данных промышленной автоматике.

Руководство пользователя: RSPManual.pdf.

Входит в дистрибутив: только для ОС Windows.

Исполняемый файл: rspdev.exe .

Файлы конфигурации:

- classes.ini — имена классов и их расположение (вкладки) на палитре визуальных компонентов;
- TrendsUser.ini — пользовательские настройки (файл создается автоматически);
- SavedTrends.ini — пользовательские наборы графиков (файл создается автоматически).

Тип программного компонента: редактор.

5.2 Среда исполнения проектов RSP

Назначение: среда исполнения *RSP Runtime* предназначена для запуска приложения, ранее сгенерированного в среде разработки *RSP Developer*.

Руководство пользователя: RSPManual.pdf.

Входит в дистрибутив: для ОС Windows / для ОС Linux.

Исполняемый файл: rsprun.exe / rsprun .

Файл конфигурации: rsprun.ini .

Тип программного компонента: среда исполнения.

5.3 Графический редактор мнемосхем Easy Graphics Editor

Назначение: графический редактор *Easy Graphics Editor* предназначен для разработки мнемосхем с возможностью предварительного просмотра в среде исполнения *RSP Runtime*.

Руководство пользователя: EasyEditManual.pdf.

Входит в дистрибутив: только для ОС Windows.

Исполняемый файл: EasyEdit.exe .

Тип программного компонента: редактор.

5.4 Служба записи и чтения архивных данных Fins Logger

Назначение: выполняет роль сервера для записи данных реального времени и доступа к базам данных архивов и конфигурации.

Руководство пользователя:

- FinsLoggerManual.pdf — общее руководство;
- RSPmanual.pdf — раздел 7.8.1 в части разработки конфигурации.

Входит в дистрибутив: для ОС Windows / для ОС Linux.

Исполняемый файл: FinsLogger.exe / FinsLogger .

Зависимости: sqlite3.dll — библиотека для поддержки SQLite.

Файл конфигурации: FinsLogger.ini .

Тип программного компонента: сервер.

Используемые таблицы конфигурационной БД: TimeIntervals, RefreshRates, Plants, PlcList, DeviceGroups, Devices, MsgCodes, PlcState .

5.5 Служба удаленного доступа к архивным данным Fins Mail

Назначение: предназначена для удалённого доступа к архивным данным, формируемым службой *Fins Logger*, позволяет передавать архивы данных по электронной почте.

Руководство пользователя: FinsMailManual.pdf.

Входит в дистрибутив: для ОС Windows / для ОС Linux.

Исполняемый файл: FinsMail.exe / FinsMail.

Зависимости: 7z.dll — библиотека для упаковки данных в архивы.

Файл конфигурации: FinsMail.ini.

Тип программного компонента: клиент или сервер.

5.6 Служба сбора и обработки данных Fins Router

Назначение: предназначена для обеспечения маршрутизации данных между сетями Ethernet, Controller Link, RS-422/485.

Руководство пользователя: FinsRouterManual.pdf.

Входит в дистрибутив: для ОС Windows / для ОС Linux.

Исполняемый файл: finssvc.exe / FinsRouter.

Файл конфигурации: finssvc.ini / FinsRouter.ini

Тип программного компонента: сервер.

5.7 DCOM сервер протокола OPC DA 2.0

Назначение: предназначен для передачи данных *FinsRouter* в сторонние приложения по протоколу OPC в соответствие со спецификацией OPC Data Access 2.0.

Руководство пользователя: FinsOPCManual.docx.

Входит в дистрибутив: только для ОС Windows.

Исполняемый файл: FinsOPC.exe.

Файл конфигурации: FinsOPC.ini.

Тип программного компонента: сервер.

5.8 Программа редактирования шаблонов отчётов FastReport

Назначение: редактирование шаблонов отчётов FastReport, используемых в проектах RSP.

Руководство пользователя: RSPManual.pdf — раздел 7.8.1 в части разработки отчётов.

Входит в дистрибутив: только для ОС Windows.

Исполняемый файл: ReportEdit3.exe.

Тип программного компонента: редактор.

5.9 Плагин сервера событий alrmsrv

Назначение: предназначен для построения распределённых систем сбора данных. Занимает промежуточное место между устройствами нижнего уровня (ПЛК и другие устройства сбора данных) и клиентскими приложениями.

Руководство пользователя: см. документ FinsRouterManual.pdf глава 5.1 «Плагин сервера событий alrmsrv».

Исполняемый файл: вместо исполняемого файла используется динамически загружаемая библиотека:

- для ОС Windows: alrmsrv.dll ;
- для ОС Linux: libalrmsrv.so .

Файл конфигурации:

- finssvc.ini - настройки плагина указываются в секции *FinsAddress_alarms*.

Тип программного компонента: плагин.

5.10 Плагин сервера МЭК 60870-5-104

Назначение: предназначен для передачи данных из памяти роутера по протоколу МЭК 60870-5-104 и работает в режиме сервера.

Руководство пользователя: см. документ `FinsRouterManual.pdf` глава 5.5.1 «Настройка плагина `iec104srv`».

Исполняемый файл: вместо исполняемого файла используется динамически загружаемая библиотека:

- для ОС Windows: `iec104srv.dll` ;
- для ОС Linux: `libiec104srv.so` .

Файл конфигурации: `finssvc.ini` – в секциях общих настроек и в настройках адресов.
Тип программного компонента: плагин.

5.11 Плагин клиента МЭК 60870-5-104

Назначение: предназначен для получения данных в память роутера по протоколу МЭК 60870-5-104 и работает в режиме клиента.

Руководство пользователя: см. документ `FinsRouterManual.pdf` глава 5.5.2 «Настройка плагина `iec104cli`».

Исполняемый файл: вместо исполняемого файла используется динамически загружаемая библиотека:

- для ОС Windows: `iec104cli.dll` ;
- для ОС Linux: `libiec104cli.so` .

Файл конфигурации: `finssvc.ini` – в секциях общих настроек и в настройках адресов.
Тип программного компонента: плагин.

5.12 Клиент для получения данных по протоколу OPC DA 2.0

Назначение: плагин `opcgate` предназначен для передачи данных в память роутера по протоколу OPC и работает в режиме клиента.

Плагин производит опрос по протоколу OPC заданного в файле конфигурации перечня параметров и помещает результаты опроса в сконфигурированные ячейки памяти `FinsRouter` .

Руководство пользователя: см. документ `FinsRouterManual.pdf` глава 5.6 «Плагин чтения данных по протоколу OPC DA 2.0 `opcgate`».

Исполняемый файл: вместо исполняемого файла используется динамически загружаемая библиотека `opcgate.dll` . Работает только под Windows.

Файл конфигурации: настройки модуля `opcgate` хранятся в соответствующих секциях файла `finssvc.ini` .

Тип программного компонента: клиент.

5.13 Клиент для получения данных по протоколу OPC HDA 2.0

Назначение: клиент для получения данных по протоколу OPC HDA 2.0

Исполняемый файл: вместо исполняемого файла используется динамически загружаемая библиотека `opchgate.dll` . Работает только под Windows.

Тип программного компонента: клиент.

5.14 Плагин формирования предупреждений `warnings`

Назначение: плагин `warnings` предназначен для формирования массива слов состояния предупреждений в соответствии с внутренним стандартом на основе входной информации в виде массива битов или массива целых чисел, формируемых клиентом IEC104 (одно- или двухэлементная информация). Плагин формирует историю событий, выполняет команды квитирования предупреждений и изменение настроек предупреждений (смена типа, маскирование предупреждений).

Руководство пользователя: см. документ `FinsRouterManual.pdf` глава 5.10 «Плагин формирования предупреждений warnings».

Исполняемый файл: вместо исполняемого файла используется динамически загружаемая библиотека:

- для ОС Windows: `warnings.dll` ;
- для ОС Linux: `libwarnings.so` .

Файл конфигурации: `finssvc.ini` . Настройки плагина указываются в секции типа `[1.1.0_warnings]` .

Тип программного компонента: плагин.

Используемые таблицы конфигурационной БД: `MsgCodes` .

5.15 Библиотечный модуль сервера OPC UA `opcuasrv`

Назначение: сервер OPC UA `opcuasrv` предназначен для передачи данных Fins Router в сторонние приложения по протоколу OPC UA. Сервер позволяет передавать данные реального времени через сервис подписки на теги с метками времени и признаками качества и содержит внутренний исторический буфер данных для обеспечения непрерывного процесса передачи данных в случае временного отключения клиентов.

Руководство пользователя: см. документ `FinsRouterManual.pdf` глава 6.1 «Библиотечный модуль сервера OPC UA `opcuasrv`».

Исполняемый файл: `opcuasrv.pyz`

Файл конфигурации: `finssvc.ini` . Базовые настройки библиотечного модуля хранятся в секции объявления модулей `[Modules]` .

Тип программного компонента: сервер.

Используемые таблицы конфигурационной БД. Настройки адресов данных и тегов библиотечного модуля хранятся в справочных таблицах конфигурационной БД проекта. Поддерживаются только таблицы данных в `csv` формате (но с расширением `txt`) . При этом в файле `tables.ini` обязательно должно присутствовать описание полей таблицы (см. подробнее в документе `RSPManual.pdf` – раздел «7.8.1 Компоненты трендов», там есть описание справочных таблиц для компонента `HistoryTrend`). Имя таблицы со списком адресов тегов может быть произвольным и задается в параметре `RootNames` .

5.16 Библиотечный модуль клиента OPC UA `opcuacli`

Назначение: клиент OPC UA `opcuacli` предназначен для получения данных в Fins Router из устройств или от сторонних приложений по протоколу OPC UA. Клиент позволяет получать данные реального времени через сервис подписки на теги с метками времени и признаками качества.

Руководство пользователя: см. документ `FinsRouterManual.pdf` глава 6.2 «Библиотечный модуль клиента OPC UA `opcuacli`».

Исполняемый файл: `opcuacli.pyz`

Файл конфигурации: `finssvc.ini` . Базовые настройки библиотечного модуля хранятся в секции объявления модулей `[Modules]` .

Тип программного компонента: клиент.

Используемые таблицы конфигурационной БД – то же самое, что и для сервера OPC UA. См. раздел 5.15.

5.17 Плагин для записи итоговых значений `sumwrite`

Назначение. Плагин представляет собой динамическую библиотеку и записывает итоговые (средние, максимальные, минимальные и мгновенные) значения для всех буферов типа `TAreaBuffer` в двухмерные таблицы.

Руководство пользователя: см. документ `FinsLoggerManual.pdf` глава 5 «Плагин для записи итоговых значений `sumwrite`».

Исполняемый файл: вместо исполняемого файла используется динамически загружаемая библиотека:

- для ОС Windows: `SumWrite.dll` ;
- для ОС Linux: `libsumwrite.so` .

Файл конфигурации: Настройки плагина описываются в конфигурационном файле `FinsLogger.ini` в секции `[SumWrite]`.

Тип программного компонента: плагин.

5.18 Тестовые утилиты

Исполняемый файл `iec104test.exe` представляет собой утилиту, которая устанавливает связь с подчиненным устройством по протоколу МЭК-60870-5-104, после чего происходит получение данных с объекта, при этом первоначально формируется команда с кодом 20 — общий опрос (*interrogated by station interrogation*) и обработка полученных данных, а затем происходит спорадический прием по изменению значения переменных.

Исполняемый файл `hltest.exe` установить связь по одному из стандартных протоколов — тех, что подключаются через `ConnectionString` и опросить регистры/банки данных с выводом значений в разных формах представления.

5.19 Утилиты регистрации компонентов

Утилита `register.exe` предназначена для регистрации ряда компонентов программного комплекса «Раскада». Утилита позволяет выбрать программный компонент из списка и ввести для него серийный номер и код активации.

Утилита `AllRegister.exe` позволяет отмечать используемые программные компоненты комплекса «Раскада» при помощи чекбоксов и вводить серийные номера продуктов, а также сохранять эту информацию в текстовый файл или в буфер обмена (для отмеченных компонентов).

5.20 Вспомогательные программные компоненты и файлы

`7z.dll` — динамическая библиотека для поддержки работы с архивными файлами формата 7z.

`classes.ini` — конфигурационный файл панели компонентов программы *RSP Developer*. Содержит имена классов и их расположение (вкладки) на палитре визуальных компонентов.

`FinsOPC.reg` — файл с ключами системного реестра Windows для DCOM сервера протокола OPC DA 2.0 `FinsOPC.exe`.

`icons16.bmp` — набор иконок для их отображения в дереве проекта программы *RSP Developer*.

`install.ps1` — PowerShell-скрипт для установки дистрибутива в определенном формате.

`libey32.dll` — динамическая библиотека, которая нужна для шифрования протокола TCP/IP.

`split104.dll` — модуль синхронизации данных резервированных клиентов по протоколу IEC60870-5-104.

`SQLite3.dll` — динамическая библиотека, содержащая средства командной строки для управления базой данных SQLite.

`ssleay32.dll` — совместно используемая библиотека OpenSSL.

5.21 Остальное

`docs` — руководства пользователя;

`locale` — файлы локализации приложений;

`py` — портативный интерпретатор *Python*, а также все используемые библиотеки;

`templates` — шаблоны страниц *RSP Developer*.

6 Информация о разработчике и контакты

ПК «Раскада» разработан, развивается и поддерживается ООО «Ракурс-инжиниринг», расположенным по адресу:

198515, г. Санкт-Петербург, пос. Стрельна, ул. Связи, дом 30, лит. А.

По вопросам обновления, добавления функциональности и устранения неисправностей ПК следует обращаться в сервисный центр ООО «Ракурс-инжиниринг» по следующим реквизитам:

- телефон: +7 (812) 252-59-09
- e-mail: sc@rakurs.com

Официальный сайт разработчика: www.rakurs.com

7 Перечень сокращений

АСУ	- автоматизированная система управления
БД	- база данных
ПО	- программное обеспечение
ООО	- общество с ограниченной ответственностью
ОС	- операционная система
ПК	- программный комплекс
ПЛК	- программируемый логический контроллер
OPC	- (англ. Open Platform Communications) - это набор программных технологий, которые предоставляют единый интерфейс для управления различными устройствами и обмена данными.
OPC DA	- (англ. Data Access) наиболее распространённый стандарт. Описывает набор функций обмена данными в реальном времени с ПЛК и другими устройствами.
OPC UA	- (англ. OPC Unified Architecture) - последняя по времени выпуска спецификация OPC, которая основана не на технологии Microsoft COM, а на архитектуре SOA (Service Oriented Architecture - «Архитектура, ориентированная на сервисы») с целью обмена информацией и обеспечения совместимости с множеством различных аппаратно-программных платформ.