

УТВЕРЖДАЮ  
Генеральный директор  
ООО «ДЕЦИМА»  
\_\_\_\_\_ А.А. Шкляев  
“ \_\_\_\_\_ ” \_\_\_\_\_ 2022

Программный комплекс КОТМИ-14

Сервер ввода-вывода

Описание программы

Лист утверждения

ЯКШГ.00067-01 91 01-03 13 - ЛУ

Руководитель разработки

Начальник лаборатории

\_\_\_\_\_ А.В. Тумаков  
“ \_\_\_\_\_ ” \_\_\_\_\_ 2022

Ответственный исполнитель

Ведущий инженер-программист

\_\_\_\_\_ М.Ю. Дьяченко  
“ \_\_\_\_\_ ” \_\_\_\_\_ 2022

Инв.№ подл.	Подп. и дата
Взам. инв. №	Подп. и дата
Инв.№ дубл.	Подп. и дата

Утвержден  
ЯКШГ.00067-01 91 01-03 13 - ЛУ

**ПРОГРАММНЫЙ КОМПЛЕКС КОТМИ-14**  
**СЕРВЕР ВВОДА-ВЫВОДА**

**Описание программы**

**ЯКШГ.00067-01 91 01-03 13**

**Листов 16**

Инв.№ подл.	Подп. и дата
Взам. инв. №	Инв.№ дубл.
Подп. и дата	Подп. и дата

## **АННОТАЦИЯ**

В настоящем документе приведено общее описание программного обеспечения (ПО) сервера ввода-вывода программного комплекса (ПК) КОТМИ-14 (далее по тексту – сервер ввода-вывода), сведения о логической структуре и функционировании ПО.

Данный комплекс служит для создания многоуровневых диспетчерских информационно-управляющих систем реального времени.

**СОДЕРЖАНИЕ**

Лист

<b>1. ОБЩИЕ СВЕДЕНИЯ.....</b>	<b>4</b>
<b>2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ.....</b>	<b>5</b>
<b>3. ТРЕБОВАНИЯ К ТЕХНИЧЕСКИМ СРЕДСТВАМ.....</b>	<b>7</b>
<b>4. ОПИСАНИЕ РАБОТЫ .....</b>	<b>8</b>
4.1. ОБЩЕЕ ОПИСАНИЕ.....	8
4.2. Модули канального уровня .....	11
4.3. ТЕЛЕУПРАВЛЕНИЕ .....	11
4.4. РЕЗЕРВИРОВАНИЕ .....	12
4.5. КОНФИГУРАЦИОННЫЙ ФАЙЛ.....	13
4.6. ВОЗМОЖНОСТИ ПО ОБРАБОТКЕ ДАННЫХ .....	13
4.6.1. ПРИЕМ НЕСТАНДАРТНЫХ ТИПОВ ДАННЫХ В ПРОТОКОЛЕ MODBUS .....	13
4.6.2. ОБРАБОТКА ПРИНИМАЕМЫХ ЗНАЧЕНИЙ .....	13
4.6.3. «ВИРТУАЛЬНЫЙ» ПОРТ .....	14
4.6.4. ПОДАВЛЕНИЕ «ДРЕБЕЗГА» КОНТАКТОВ.....	14
4.6.5. СОСТАВНЫЕ ТЕЛЕСИГНАЛЫ .....	14
<b>ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ.....</b>	<b>15</b>

## **1. ОБЩИЕ СВЕДЕНИЯ**

Сервер ввода-вывода работает в составе ПК КОТМИ-14 выполняет функции коммуникационного программного сервера или роутера для этих систем.

Сервер ввода-вывода работает под управлением операционных систем семейства Windows или Linux. ПО сервера ввода-вывода использует стандартные сервисы, предоставляемые этими операционными системами.

Программа написана на языке C++. При разработке дополнительных протоколов, оформленных в соответствии с требованиями интерфейса расширения, могут использоваться любые языки и среды программирования, обеспечивающие построение DLL-модуля протокола.

## 2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

Сервер ввода-вывода работает в составе ПК КОТМИ-14 и выполняет функции коммуникационного программного сервера или роутера данных в сети устройств разного уровня (контроллеры, устройства RTU, SCADA–системы, ЦППС).

В режиме коммуникационного сервера (концентратор каналов или конвертор протоколов) сервер ввода-вывода принимает:

- данные от нижестоящих систем или устройств (контроллеры, устройства RTU, контролируемые пункты) и передает их в одну вышестоящую систему;
- команды от вышестоящей системы и передает их в заданную нижестоящую систему или устройство.

В режиме роутера данных в сети устройств разного уровня сервер ввода-вывода выполняет:

- маршрутизацию данных между системами и устройствами без явного выделения систем и устройств нижестоящего уровня и системы вышестоящего уровня;
- маршрутизацию данных при наличии в сети нескольких независимых систем вышестоящего уровня.

Данные режимы не являются взаимоисключающими и, в зависимости от настройки конфигурации, могут поддерживаться одновременно.

Сервер ввода-вывода обеспечивает:

- одновременный и независимый прием данных от всех подключенных систем и устройств;
- одновременную и независимую передачу данных всем подключенным системам и устройствам;
- взаимодействие с другими устройствами посредством последовательных (синхронных или асинхронных) каналов, IP (TCP или UDP) каналов, OPC соединений;
- хранение последних изменений по каждому параметру. Глубина хранения для каждого параметра может быть задана индивидуально.

Обмен и ретрансляция данных по каналам связи осуществляется по ряду стандартных протоколов, используемых в телемеханике. В настоящее время поддерживаются следующие протоколы:

- МЭК 61850-8-1 (MMS и GOOSE);

- МЭК 60870-5-101;
- МЭК 60870-5-103;
- МЭК 60870-5-104;
- OPC DA 3.00;
- Modbus RTU;
- Modbus ASCII;
- Modbus TCP;
- СЭТ;
- Меркурий 230;
- SPABus;
- MDE (КОТМИ-2010);
- MDX (КОТМИ-14).

Состав протоколов может быть расширен, в том числе, силами сторонних разработчиков. С этой целью был разработан специальный программный интерфейс взаимодействия с сервером ввода-вывода. Модули протоколов (DLL), написанные с учетом соглашений интерфейса, будут полноценно функционировать в системе наравне с «родными» протоколами сервера ввода-вывода.

### 3. ТРЕБОВАНИЯ К ТЕХНИЧЕСКИМ СРЕДСТВАМ

Сервер ввода-вывода (СВВ) предназначен для использования на промышленных или персональных компьютерах, работающих под управлением операционных систем:

- Операционная система «Альт 8 СП», разработки компании ООО «Базальт СПО».
- Операционная система специального назначения «Astra Linux Special Edition» РУСБ.10015-01 (очередное обновление 1.6), разработки компании ООО «РусБИТех-Астра».
- Операционная система специального назначения «Astra Linux Special Edition» РУСБ.10152-02 (очередное обновление 4.7), разработки компании ООО «РусБИТех-Астра».
- Операционная система специального назначения «Astra Linux Special Edition» РУСБ.10265-01 (очередное обновление 8.1), разработки компании ООО «РусБИТех-Астра».
- Microsoft Windows.

Для ПО сервера ввода-вывода предъявляются следующие минимальные аппаратные требования к ПЭВМ:

- Процессор:
  - Центральные процессоры с архитектурой «Эльбрус» (E2K): «Эльбрус-4С», «Эльбрус-8С» и «Эльбрус-8СВ», разработанные компанией АО «МЦСТ»;
  - Центральные процессоры с архитектурой ARMv8-A: «Baikal-M», разработанный компанией АО «Байкал Электроникс»;
  - Центральные процессоры с архитектурой x86 (Intel 80x86) и x86-64 (AMD64/Intel64/EM64T).
- Оперативная память — 4 ГБ.
- Дисковое пространство — 1 ГБ.
- Скорость канала связи Ethernet с сервером приложений (при использовании в составе ПК КОТМИ-14) — 10 Мбит/сек.
- Наличие последовательного интерфейса (COM-порта) для устройств, использующих для связи последовательный интерфейс.



## 4. ОПИСАНИЕ РАБОТЫ

### 4.1. Общее описание

Сервер ввода-вывода обеспечивает взаимодействие ПК КОТМИ-14 с различными внешними системами и устройствами (Рисунок 4.1).

Сервер ввода-вывода поддерживает радиальное и магистральное подключение устройств и внешних систем, и произвольное количество резервируемых линий связи до устройств и внешних систем.

Сервера ввода-вывода могут попарно резервироваться.

На одном компьютере может быть развернуто несколько серверов ввода-вывода, каждый со своей конфигурацией. Лицензия учитывает общее количество параметров всех серверов ввода-вывода на одном компьютере.

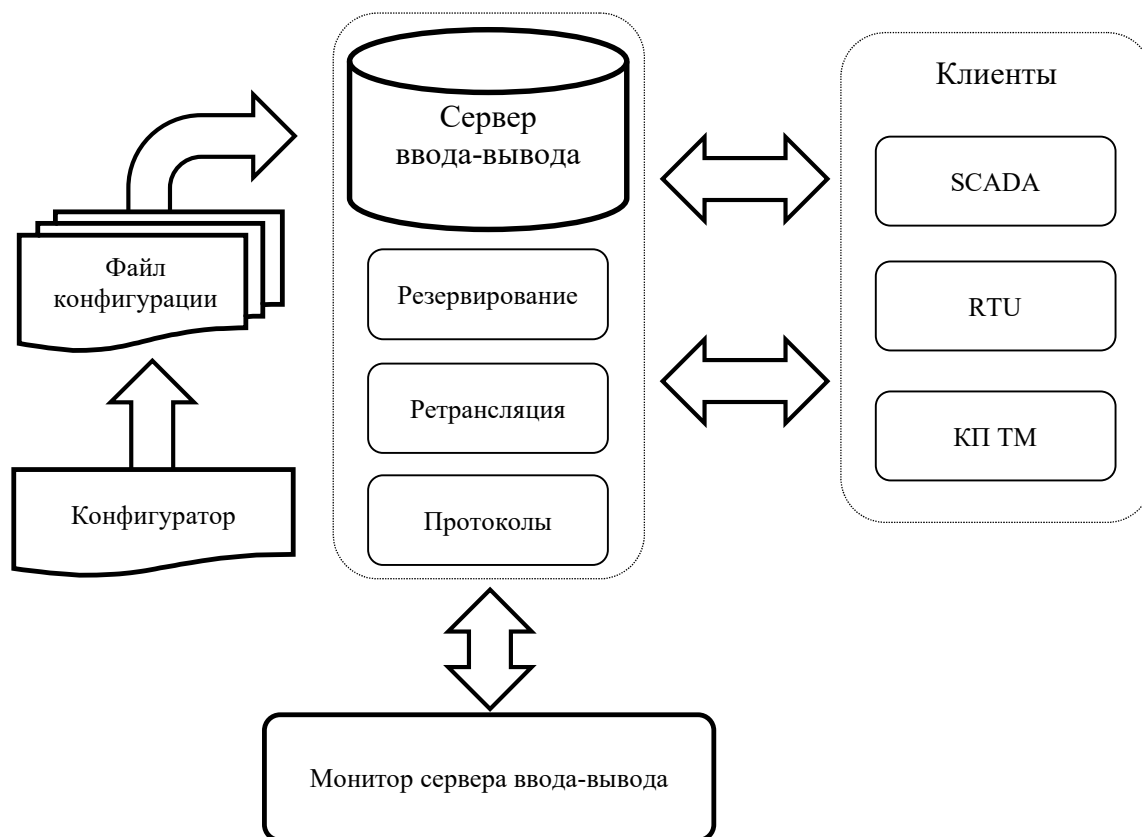


Рисунок 4.1 – Базовая архитектура и взаимодействие компонентов сервера ввода-вывода

Программа сервера ввода-вывода состоит из следующих основных частей:

- ядро системы;
- загрузчик конфигурационного файла;

- набор драйверов, реализующих канальные уровни разных протоколов (TCP, UDP, COM-порты);
- набор модулей, реализующих прикладные уровни разных протоколов.

Модули прикладных протоколов – отдельные dll-файлы, взаимодействующие с сервером ввода-вывода через специальный интерфейс расширения (API).

После запуска на этапе инициализации программа сервера ввода-вывода считывает настроечную информацию из указанного конфигурационного xml-файла и формирует внутреннюю рабочую структуру каналов, параметров и связей между ними. Затем осуществляется непосредственное соединение по каналам, прием и ретрансляция данных между настроенными протоколами.

В конфигурационном файле также задается схема резервирования, которую должен использовать сервер ввода-вывода. Поддерживается как резервирование нескольких экземпляров сервера ввода-вывода, так и резервирование каналов внутри одного сервера ввода-вывода.

Данные, принимаемые и передаваемые по каналам, имеют структуру, специфическую для используемого протокола. Для унификации обмена данными между различными протоколами внутри сервера ввода-вывода используются так называемые обобщенные типы данных (ОТД). Ретрансляция данных из одного протокола в другой внутри сервера ввода-вывода выполняется через структуру ОТД.

Подготовка конфигурационной информации осуществляется с помощью программы конфигулятора сервера ввода-вывода. Конфигуратор на основе существующих мета описаний протоколов и устройств позволяет удобным образом создавать конфигурации, описывающие направления, протоколы, устройства и параметры, участвующие в обмене, задать связи при ретрансляции данных, сформировать требуемую схему резервирования. Результатом конфигурирования является xml-файл специальной структуры, принимаемый сервером ввода-вывода в качестве входных данных.

Управление и наблюдение за работой сервера ввода-вывода осуществляется с помощью программы «Монитор сервера ввода-вывода».

Сервер ввода-вывода представляет собой консольное приложение (Рисунок 4.2), написанное на языке программирования C++, и работающее под операционными системами семейства Windows и Linux.

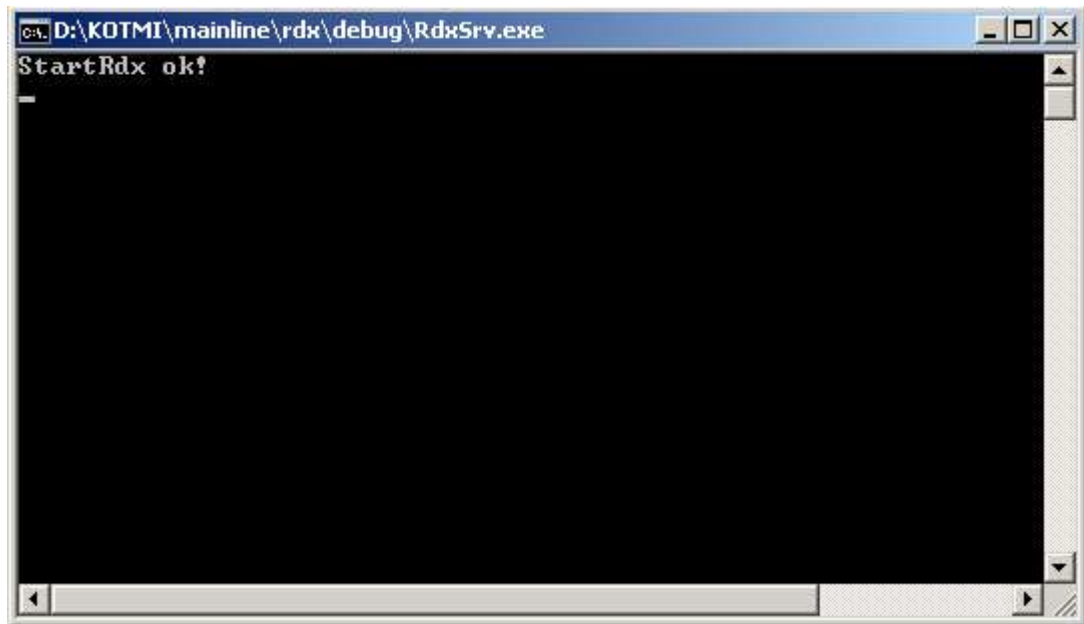


Рисунок 4.2 – Окно сервера ввода-вывода

В своей работе сервер ввода-вывода использует стандартные сервисы, предоставляемые этими операционными системами.

Функционирование сервера ввода-вывода подразделяется на три этапа:

**Старт и конфигурирование системы (start time).** На этом этапе работает один, основной поток процесса. Читается и разбирается конфигурационный xml-файл, заданный в параметрах командной строки. В соответствии с заданной конфигурацией формируется внутренняя структура направлений, каналов, параметров обмена и связей между ними. Настраивается требуемая схема резервирования для каналов, работающих в составе одного направления. После того как структура сформирована, последовательно запускаются на выполнение все потоки каналов ввода-вывода.

**Основной цикл обработки (run time).** Запущены и работают каналы ввода-вывода. В зависимости от состояния каналов (подключен-отключен), осуществляется прием данных, ретрансляция и передача, в соответствии со схемой стартовой конфигурации. Каждый канал обслуживается отдельным, событийно-управляемым потоком процесса. Основное состояние потока – это режим ожидания события («спящий режим»). Переход потока из «спящего» состояния в активное происходит по следующим событиям: событие таймера протокола, принятие данных из канала, появление данных для маршрутизации, завершение работы.

**Останов системы (stop time).** После того, как основной поток сервиса получил событие «завершение работы» он выполняет следующие действия: устанавливает события

«завершение работы» для потоков всех каналов, освобождает все, ранее созданный программные объекты, завершает работу процесса.

#### 4.2. Модули канального уровня

Модули канального уровня представляют собой набор коммуникационных подпрограмм, реализующих канальный уровень для заданного протокола. В зависимости от типа протокола (IP, асинхронный последовательный) это может быть один из следующих вариантов:

- **Стандартный СОМ порт.** Модуль, для работы с асинхронными протоколами. Работа с асинхронными протоколами осуществляется через стандартные СОМ порты системы;
- **TCP/IP сокет.** Модуль, для работы с сетевыми протоколами. Работа с сетевыми протоколами осуществляется посредством стандартной библиотеки сокетов Беркли.
- **UDP/IP сокет.** Модуль, для работы с сетевыми протоколами. Работа с сетевыми протоколами осуществляется посредством стандартной библиотеки сокетов Беркли.

#### 4.3. Телеуправление

Сервер ввода-вывода поддерживает режим телеуправления для следующих протоколов:

- МЭК 61850-8-1 (MMS и GOOSE);
- МЭК 60870-5-101;
- МЭК 60870-5-104;
- Modbus RTU;
- Modbus TCP;
- MDE (КОТМИ-2010);
- MDX (КОТМИ-14).

Команда телеуправления не может быть *сгенерирована* самим сервером ввода-вывода. Сервер ввода-вывода, в данном случае, является, исключительно, прозрачным устройством. Это касается, как принятия решения о передаче команды телеуправления, так и прикладной процедуры ее исполнения. То есть, один из абонентов сервера ввода-вывода, например, SCADA система, передает команду телеуправления, другому абоненту сервера связи, например, устройству RTU. При этом, оба абонента, SCADA система и устройство RTU, *ответственны* за корректность прикладной процедуры команды телеуправления, за возможность выполнить команду телеуправления, за проверку необходи-

мых таймаутов и т. д. Сервер ввода-вывода осуществляет только маршрутизацию пакетов, содержащих информацию о телеуправлении, между двумя абонентами.

В протоколах МЭК 870-5-101 и МЭК 870-5-104 существуют специальные типы данных для команд телеуправления: ASDU58 и ASDU59. Сервер ввода-вывода идентифицирует эти типы данных, как **команды**. Благодаря чему, информация с типами данных телеуправления не архивируется в сервере, в отличие от других, информационных, типов данных.

Для поддержки команд телеуправления в линиях с разными протоколами, модули, реализующие эти протоколы, должны использовать соответствующие обобщенные типы данных, предназначенные для выполнения команд ТУ.

При выполнении команд ТУ между разными протоколами, используется упрощенный, одноэтапный алгоритм. Результатом ТУ будет квитанция о выполнении или невозможности выполнения команды. Время ожидания квитанции задается соответствующими настройками на этапе конфигурирования параметров.

#### 4.4. Резервирование

Сервер ввода-вывода поддерживает следующие виды резервирования:

- резервирование хостов;
- резервирование каналов (в рамках одного хоста).

С точки зрения механизма резервирования серверов, статус каждого работающего сервера ввода-вывода может быть либо **online**, либо **standby**. Статус **standby** означает, что сервер находится в работающем состоянии с полностью развернутой run-time конфигурацией. При этом каналы связи не активированы. То есть, сервер *не устанавливает* соединения с абонентами (RTU устройствами, SCADA системами, другими серверами связи). Каналы, выполняющие функции резервирования и диагностики активируются вне зависимости от режима **online** или **standby**.

Если работает только один сервер ввода-вывода, то он и будет **online** сервером. Если работают одновременно два или несколько серверов, то в **online** режиме будет тот сервер, который стартовал раньше. Остальные будут ожидать в **standby** режиме, периодически осуществляя процедуру контроля работоспособности **online**-хоста. В случае потери связи или перехода **online**-хоста в режим **standby**, остальные сервера по встроенному алгоритму определяют и активизируют новый **online**-процесс. Сервер, работающий в **online** режиме, можно перевести в режим **standby**, и наоборот. Для этого в мониторе необходимо подать команду «Сменить статус».

Резервирование каналов в одном процессе сервера ввода-вывода, осуществляется в рамках понятия «направления». «Направление» представляет собой группу из одного или нескольких однотипных протоколов, работающих по разным каналам, но с одной и той же конфигурацией телемеханических параметров. В каждый момент в режиме *online* находится только один канал. Периодически, в процессе работы они тестируют состояние активного *online*-канала и, при необходимости, могут перераспределить *online*-статус в соответствии с приоритетами, заданными при конфигурации.

#### 4.5. Конфигурационный файл

Вся необходимая информация для работы программы **RdxServer.exe**, берется из конфигурационного xml-файла, задаваемого в параметрах командной строки при запуске сервера ввода-вывода.

#### 4.6. Возможности по обработке данных

##### 4.6.1. Прием нестандартных типов данных в протоколе Modbus

Многие устройства, работающие по протоколу Modbus, передают информацию в форматах отличных от стандартных функций этого протокола.

Сервер ввода-вывода может принимать и обрабатывать данные структурированные любым образом. Посредством конфигурационных параметров «**start**» и «**length**» задается поле, содержащее отдельный элемент информации в принимаемом пакете.

Длина элемента информации указывается в битах.

Начало и конец элемента информации может приходиться на любое место принимаемого пакета, не обязательно на границу байта.

Параметр «**type**» задает вариант преобразования этой информации.

##### 4.6.2. Обработка принимаемых значений

Каждому принимаемому элементу информации можно назначить процедуру обработки в сервере ввода-вывода.

Функции обработки:

- инверсия;
- смещение;
- умножение;
- обнуление;
- контроль апертуры.

#### 4.6.3. «Виртуальный» порт

Функциональность «виртуальный» порт - это возможность подключиться специализированным сторонним ПО к удаленному контроллеру.

Подключение осуществляется через канал, используемый сервером ввода-вывода для телеметрии.

При этом прием-передача телеметрии не прекращается.

#### 4.6.4. Подавление «дребезга» контактов

Сервер ввода-вывода имеет функцию подавления эффекта «дребезг» контактов. То есть, часто изменяющихся телесигналов. Каждому телесигналу, в конфигураторе, можно задать два параметра: **rate** (временной промежуток), **count** (количество изменений). Функция подавления работает следующим образом: если за временной промежуток **rate**, произошло **count** изменений телесигнала, то сигналам считается «дребезжащим». И дальнейшие его изменения не фиксируются системой. А остается последнее состояние из **count** значений. В последующем, если за временной промежуток **rate**, произошло меньше чем **count** изменений телесигнала, то сигналам считается не «дребезжащим». И дальнейшие его изменения фиксируются системой.

#### 4.6.5. Составные телесигналы

Сервер ввода-вывода может формировать составные телесигналы из принимаемых однобитных телесигналов. Логика формирования составных сигналов следующая:

ts1	ts2	значение	флаг
0	0	0	10000 RDX_FLAG_TS2_Intermediate
0	1	0	0
1	0	1	0
1	1	1	10000 RDX_FLAG_TS2_Intermediate

Данная функциональность поддерживается только протоколами МЭК-101 / 104.

**ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ**

- ОТД – обобщенный тип данных
- ПК - программный комплекс
- ПО – программное обеспечение



