



АО «ПИК ПРОГРЕСС»

Юр. Адрес: 111024, Москва, Авиамоторная ул., д. 51А.
Почт.Адрес: 111024, Москва, Авиамоторная ул., д. 51А
тел.: (495) 365-50-25, 365-55-58

Программно-технический комплекс «Космотроника»

Программный комплекс «QMicro» для операционной системы «Нейтрино»

Руководство по эксплуатации

СШМК.00103-01 42 01

Листов 40

Инд. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата

АННОТАЦИЯ

В данном документе содержится информация для подготовки к работе и настройке программного комплекса «QMicro» для операционной системы ЗОСРВ «Нейтрино».

В разделе «Назначение программы» указано назначение и функции программы и сведения о программных и технических средствах, обеспечивающих ее выполнение.

В разделе «Условия выполнения и состав комплекса» указаны сведения о составных частях комплекса и их взаимодействию.

В разделе «Настройка программы» приведено описание действий по настройке программы на условия конкретного применения и приведены поясняющие примеры.

СОДЕРЖАНИЕ

АННОТАЦИЯ.....	2
СОДЕРЖАНИЕ	3
1. НАЗНАЧЕНИЕ ПРОГРАММЫ.....	5
2. УСЛОВИЯ ВЫПОЛНЕНИЯ	5
2.1. Составные части	6
2.1.2 Драйверы каналов связи	7
2.1.3 Драйверы ретрансляции информации	8
2.1.4 Драйверы устройств связи с объектом.....	8
2.1.5. Дополнительные модули	8
2.2 Подготовка файлов для загрузке в контроллер	9
2.2.1. Структура файла start.ini	9
2.2.1.1 Ключи раздела Disks.....	9
2.2.1.2 Ключи раздела Options	10
2.2.1.3. Раздел USO	10
2.2.1.4. Раздел Kanals.....	10
2.2.1.5. Раздел Retransl	11
2.2.1.6. Раздел Module.....	11
2.2.1.7. Раздел Programms	11
2.3 Подготовка контроллера к функционированию.....	11
2.3.1 Подготовка в режиме прямого соединения через последовательный порт.....	11
2.3.1.1 Соединение с контроллером	11
2.3.1.2 Запись или обновление программного обеспечения	12
2.3.1.3 Чтение и запись конфигурации контроллера.....	12
2.3.1.4 Чтение и запись файлов-описателей контроллера	13
2.3.2 Подготовка в режиме соединения по локальной сети через встроенный ftp-сервер.....	14
Приложение 1. Пример файла start.ini	16
Приложение 2. Драйверы USO	17
Драйвер логометров УП23/УП24.....	17
ДРАЙВЕР АНАЛОГОВОГО МУЛЬТИПЛЕКСОРА ДЛЯ ПЛАТЫ СШМК.758717.00077 ПИК “Прогресс”	17
ДРАЙВЕР СЧИТЫВАНИЯ ИНФОРМАЦИИ СО СЧЕТЧИКОВ “Альфа”	18
ЦИФРОВЫЕ ВХОДЫ-ВЫХОДЫ USO-КО.....	18
ЦИФРОВАЯ ПЛАТА 5600	18
ДРАЙВЕР ЦИФРО-АНАЛОГОВОГО ПРЕОБРАЗОВАТЕЛЯ.	20
ВИРТУАЛЬНОЕ USO.....	21

ДРАЙВЕР МОДУЛЕЙ КОНТРОЛЯ И УПРАВЛЕНИЯ ICRCON СЕРИИ I-7000	21
ДРАЙВЕР ОБМЕНА ИНФОРМАЦИЕЙ ПО ПРОТОКОЛУ MODBUS RTU.....	22
ДРАЙВЕР ОБМЕНА ИНФОРМАЦИЕЙ ПО ПРОТОКОЛУ MODBUS TCP	24
ДРАЙВЕР ЛОГОМЕТРОВ УП23/УП24	26
ДРАЙВЕР МОДУЛЕЙ КОНТРОЛЯ И УПРАВЛЕНИЯ УСО-М 16ТС, 8ТИ, 16ТУ.....	27
ДРАЙВЕР МОДУЛЕЙ КОНТРОЛЯ И УПРАВЛЕНИЯ УСО-М 64ТС, 32ТИ, 16ТУ, 8ЦАП	27
ДРАЙВЕР МОДУЛЕЙ КОНТРОЛЯ И УПРАВЛЕНИЯ УСО ТМ	28
ДРАЙВЕР УДАЛЁННОГО УСО НА БАЗЕ ЗОСРВ «Нейтрино».	30
Приложение 3. Драйверы каналов связи	31
Общие ключи для драйверов каналов связи	31
Драйвер обмена через последовательный порт	33
Драйвер обмена по радиоканалу.....	33
Драйвер обмена по сети	33
Драйвер обмена по локальной сети по протоколу UDP.....	34
Драйвер обмена по сети с АРМ	34
Драйвер передачи данных по протоколу ГОСТ Р МЭК 60870-5-104	35
Драйвер передачи информации по протоколу MODBUS TCP	35
Драйвер передачи информации по протоколу MODBUS RTU	36
Приложение 4. Драйверы ретрансляции	36
Драйвер ретрансляции через последовательный порт	36
Драйвер ретрансляции по локальной сети по протоколу UDP.....	36
Драйвер ретрансляции через последовательный порт на контроллер, работающий под управлением операционной системы DOS.	37
Приложение 5. Подключаемые модули	37
Вычисление значений расчетных сигналов	38
Переход на летнее и зимнее время.....	38
Управление сторожевым таймером	38
Управление журналом событий.....	38
Управление пультом оператора.....	39
Типовые проблемы и их решение.....	39
Техническая поддержка.....	40

1. НАЗНАЧЕНИЕ ПРОГРАММЫ

Программный комплекс «QМісто» предназначен для решения задач оперативного контроля и управления различными технологическими объектами. В соответствии со своим назначением комплекс выполняет следующие основные функции :

- Сбор информации об объекте управления в виде набора параметров различного типа (аналоговые, дискретные, импульсные)
- Первичную обработку собранной информации (фильтрация, сглаживание) и ее проверку на достоверность.
- Выполнение дополнительных расчетов по различным формулам
- Анализ изменений в состоянии обрабатываемых параметров
- Передачу собранной информации на верхний уровень по различным каналам связи
- Хранение собранной информации до момента ее передачи на верхний уровень
- Контроль и управление работой периферийных устройств (УСО, каналообразующая аппаратура, сторожевой таймер)
- Ведение различных архивов и протоколов (протокол старта системы, журнал событий, архив данных со счетчиков электроэнергии).
- Выполнение управляющих воздействий на объект управления в виде команд телеуправления и аналогового управления (ЦАП). Воздействие может осуществляться как автоматически, так и по команде с верхнего уровня системы.

Структура программного обеспечения позволяет включать в состав комплекс дополнительные модули, расширяющие набор выполняемых функций. Набор функций, выполняемых конкретным контроллером, зависит от его настроек.

2. УСЛОВИЯ ВЫПОЛНЕНИЯ

Комплекс программ предназначен для выполнения на промышленных контроллерах, построенных на базе архитектуры Intel x86 или ARM, в среде ЗОСРВ «Нейтрино». Детальные сведения о подходящих технических средствах изложены в руководстве пользователя ОС.

6

СШМК.00103-01 42 01

2.1. Составные части

Комплекс программ состоит из следующих компонентов:

- Программа `qmicro`, которая запускает ядро программного комплекса и отслеживает его работоспособность и, при необходимости, осуществляющая перезапуска ПО.
- Программы `qmicro`, являющейся ядром программного комплекса
- Драйверов каналов связи
- Драйверов ретрансляции информации
- Драйверов устройств связи с объектом (УСО)
- Дополнительных модулей

2.1.1 Программа `qmicro`

Программа `qmicro` является ядром программного комплекса и выполняет следующие действия:

- Загружает в оперативную память описания сигналов для конкретного контроллера
- Загружает в оперативную память последние сохраненные состояния сигналов
- В соответствии с настройками загружает драйвера устройств связи с объектами
- В соответствии с настройками загружает драйвера каналов связи
- В соответствии с настройками загружает драйвера ретрансляторов
- В соответствии с настройками загружает дополнительные модули
- В соответствии с настройками загружает дополнительные программы
- Протоколирует процесс запуска в протоколе старта (если в настройках задано формирование протокола старта комплекса)
- Формирует сообщение о загрузке комплекса для передачи по каналам связи и записи в журнале событий.

После выполнения этих действий программа `qmicro` переходит в режим штатного функционирования, который заключается в циклическом выполнении следующих действий:

- Тестирование загруженных драйверов каналов связи, ретрансляции и УСО. В случае сбоя какого-либо драйвера производится перезагрузка комплекса с записью в журнале событий.
- Запрос у драйверов УСО текущего состояния устройств и формирование событий о пропадании и восстановление связи с УСО.
- Получение от драйверов УСО значений измеряемых телесигналов и их обработка. Формирование событий по этим сигналам в соответствии с настройками для передачи на верхний уровень.
- Получение от модуля дорасчетов значений расчетных телеизмерений и их обработка. Формирование событий по этим сигналам в соответствии с настройками для передачи на верхний уровень.
- Получение от драйверов УСО значений импульсных входов.
- Получение от драйверов УСО значений измерений телеизмерений и их обработка. Выполнение первичной обработки этих сигналов (сглаживание, фильтрация,

СШМК.00103-01 42 01

оцифровка). Формирование событий по этим сигналам в соответствии с настройками для передачи на верхний уровень.

- Расчет значений интегральных телеизмерений. Формирование событий по этим сигналам в соответствии с настройками для передачи на верхний уровень.
- Получение от модуля дорасчетов значений расчетных телеизмерений и их обработка. Выполнение первичной обработки этих сигналов (сглаживание, фильтрация). Формирование событий по этим сигналам в соответствии с настройками для передачи на верхний уровень.
- Выполнение команд телеуправления в автоматическом режиме в соответствии с настройками.
- Сохранение текущих значений параметров в случае их изменения
- Сохранение описателей параметров в случае их изменения.

Кроме этого программа `qmicro` запускает две вспомогательные задачи. Первая задача отвечает за обмен информацией с драйверами канала связи. Вторая задача предназначена для организации ретрансляции информации на другие контроллеры и запускается, только на контроллерах, осуществляющих ретрансляцию информации на другие контроллеры.

2.1.2 Драйверы каналов связи

Драйверы каналов связи осуществляют обмен информацией между контроллером и Коммуникационным сервером “Космотроника”, используя различное коммуникационное оборудование (локальная сеть, порт RS-232/485, радиомодемы и т.п.). Для обмена с программами верхнего уровня других производителей используются драйверы каналов, поддерживающих различные протоколы обмена (Modbus, IEC 870-5-101, IEC 870-5-104 и т.п.). Работа таких драйверов зависит от специфики канала и протокола обмена.

Одновременно контроллер может вести обмен по одному или нескольким каналам. При этом в соответствии обмен по каждому каналу может осуществляться независимо (разные направления передачи информации) или информация передается только один раз по одному из группы каналов (резервирование каналов).

Драйверы каналов связи, обслуживающих обмен с Коммуникационным сервером “Космотроника”, выполняют следующие основные функции:

- Прием запросов от сервера ввода-вывода.
- Проверка правильности приема запроса (сверка контрольных сумм и флагов начала и конца посылки). Запросы не прошедшие процедуру проверки игнорируются.
- Проверка адресного поля запроса. Если адресное поле запроса не совпадает с адресом контроллера, то запрос игнорируется
- Проверка конечного адресата. Если в качестве конечного адресата указан адрес другого контроллера, то посылка передается задаче (программа `qmicro`) для ретрансляции запроса адресату.
- Если в качестве конечного адресата указан данный контроллер, то драйвер распаковывает запрос и подготавливает ответ на него самостоятельно, либо передает запрос на обработку задаче (программа `qmicro`), обеспечивающей обмен с драйверами каналов связи. В этом случае ответ на запрос формируется программой `qmicro` и передается драйверу канала для передачи.

В зависимости от типа и возможностей используемого канала передача может осуществляться только по запросу или по запросу и спорадически (инициативная передача).

При обмене с сервером ввода-вывода по каналу связи при необходимости производится коррекция текущего времени контроллера.

2.1.3 Драйверы ретрансляции информации

Драйверы ретрансляции используются для связи контроллера с Коммуникационным сервером «Космотроника» через промежуточные контроллеры. При этом, на промежуточных контроллерах должны быть установлены драйверы ретрансляции информации. Драйвер ретрансляции выполняет следующие функции:

- Принимают от задачи (программа qm1cgo), обслуживающей ретрансляцию данных запросы к удаленным КП и передает их по каналу связи.
- Принимает ответы на посланные запросы и передает их на обработку задаче (программа qm1cgo), обслуживающую ретрансляцию данных.

2.1.4 Драйверы устройств связи с объектом

Драйверы устройств связи с объектом являются промежуточным звеном между программой qm1cgo и измеряемыми параметрами. Для этого драйверы УСО выполняют следующие функции:

- Инициализация УСО при старте комплекса (при необходимости)
- Периодическое тестирование или проверка связи с УСО
- Синхронизация текущего времени УСО
- Чтение и передача в программу qm1cgo по запросу значений подключенных к обслуживаемым УСО телесигналов, телеизмерений (в виде кода АЦП или реального значения) и импульсных входов.
- Выполнение по запросу программы qm1cgo команд телеуправления.
- Обслуживание дополнительных запросов с верхнего уровня (чтение профилей нагрузки, чтение и запись регистров устройства и т.п.)

2.1.5. Дополнительные модули

Дополнительные модули оформлены в виде динамически подключаемых библиотек и используются для расширения функциональности комплекса. В настоящий момент реализованы следующие дополнительные модули:

- Модуль дорасчетов, позволяющий проводить вычисления значения расчетных параметров на основе уже имеющихся.
- Модуль, обеспечивающий автоматический переход контроллера на летнее и зимнее время.
- Модуль управления сторожевым таймером для контроллера ПТК «Космотроника», обеспечивающий перезагрузку контроллера, в случае сбоя программного обеспечения.
- Модуль ведения журнала событий.

2.2 Подготовка файлов для загрузке в контроллер

Для нормального функционирования в контроллер должно быть загружена и инициализирована операционная ЗОСРВ «Нейтрино», программный комплекс «QMicro» для ЗОСРВ «Нейтрино», файл с конфигурацией контроллера start.ini, двоичные файлы-описатели (файлы с расширением bin), описывающие УСО и сигналы контроллера.

Операционная система загружается и инициализируется на этапе изготовления контроллера. Этот процесс описывается отдельной инструкцией.

Программное обеспечение для контроллеров располагается в директории /flashdisk/progr. Первоначальная версия программного обеспечения загружается в контроллер на этапе изготовления. При необходимости обновления, программное обеспечение поставляется в электронном виде.

Внимание – файлы, относящиеся к программному обеспечению контроллера, должны иметь атрибут исполняемый. Атрибут исполняемый присваивается в ЗОСРВ «Нейтрино» командой `chmod +x <имя файла>` (для отдельного файла) или `chmod +x *` (для всех файлов директории).

Файл start.ini располагается в директории /flashdisk/progr. Структура файла описывается в следующем разделе.

Двоичные файлы описатели с расширением bin располагаются в директории /flashdisk/bin. Эти файлы создаются в конфигураторе контроллера.

2.2.1. Структура файла start.ini

Файл start.ini является текстовым файлом, описывающим конфигурацию контроллера. Работа программы qmicro начинается с чтения этого файла. Пример файла приведен в приложении 1. Файл start.ini состоит из разделов, а разделы из наборов ключей. В файле могут присутствовать следующие разделы:

- **Disks** – определяет структуру директорий контроллера
- **Options** – секция настроек ПО
- **Uso** – перечисляет драйверы УСО, которые должны быть загружены, с их настройками
- **Kanals** – перечисляет драйверы каналов связи, которые должны быть загружены, с их настройками
- **Module** - перечисляет подгружаемые модули, которые должны быть загружены, с их настройками
- **Retransl** - перечисляет драйверы ретрансляторов, которые должны быть загружены, с их настройками
- **Programms** - перечисляет внешние программы, которые должны быть загружены, с их настройками

2.2.1.1 Ключи раздела Disks

В разделе Disks файла start.ini допускаются следующие ключи:

- **FileStartProtokol** – описывает имя файла с протоколом старта системы. Протокол старта системы является текстовым файлом, в котором отображаются

10

СШМК.00103-01 42 01

выполненные операции и ошибки, возникшие при загрузке и инициализации программ

- FlashDiskPath – указывает каталог, где хранятся двоичные файлы-описатели с расширением bin
- StaticDiskPath - указывает каталог, где сохраняются текущие значения параметров и не переданная на верхний уровень информация. Информация сохраняется на случай внезапной перезагрузки
- SwopDiskPath - указывает каталог, где сохраняются файлы при дистанционном конфигурировании контроллера.
- SavePeriod – периодичность в сек, с которой записываются изменения текущих значений параметров на диск. По умолчанию – 0 сек – запись производится по каждому изменению. Если значение -1 – запись производится не будет.

Типовая конфигурация раздела выглядит следующим образом:

[Disks]

FileStartProtokol=/flashdisk/start
FlashDiskPath=/flashdisk/bin/
StaticDiskPath=/flashdisk/sram
SwopDiskPath=/flashdisk/swop

Внимание – изменение параметров по умолчанию ключей раздела Disks может сделать невозможным дистанционное конфигурирование контроллера!

2.2.1.2 Ключи раздела Options

В разделе Options файла start.ini допускаются следующие ключи:

SendFirstValue – разрешает (On) или запрещает (Off) посылку первого значения параметра после старта комплекса, если последнее переданное значение не было прочитано из статической памяти. Значение по умолчанию – On.

2.2.1.3. Раздел USO

Раздел USO может содержать один или несколько ключей Uso. Каждый ключ содержит имя запускаемого драйвера USO с его настройками. Список доступных драйверов USO с их настройками приведен в приложении 2.

2.2.1.4. Раздел Kanals

Раздел Kanals может содержать ключи Kanal. Каждый ключ содержит имя запускаемого драйвера канала связи с его настройками. Список доступных драйверов каналов с их настройками приведен в приложении 3.

2.2.1.5. Раздел Retransl

Раздел Retransl может содержать ключи Retransl. Каждый ключ содержит имя запускаемого драйвера ретрансляции с его настройками. Список доступных драйверов ретрансляции с их настройками приведен в приложении 4.

2.2.1.6. Раздел Module

Раздел Module может содержать ключи Module. Каждый ключ содержит имя запускаемого подгружаемого модуля с его настройками. Список доступных подгружаемых модулей их настройками приведен в приложении 5.

2.2.1.7. Раздел Programms

Раздел Programms может содержать ключи Programm. Каждый ключ содержит имя внешней программы с параметрами запуска, запускаемой при старте комплекса.

2.3 Подготовка контроллера к функционированию

Подготовка контроллера к функционированию в этом режиме состоит из следующих этапов:

- Соединение с контроллером
- Запись или обновление программного обеспечения
- Чтение и запись конфигурации контроллера
- Чтение и запись файлов-описателей контроллера

2.3.1 Подготовка в режиме прямого соединения через последовательный порт

2.3.1.1 Соединение с контроллером

Соединение с контроллером производится с помощью программы Microsoft® HyperTerminal или аналогичной коммуникационной программы. Для установления соединения необходимо подключиться к второму последовательному порту (COM2) контроллера нуль-модемным кабелем. Установить в коммуникационной программе параметры соединения – скорость 9600, 8, N, 1. **Затем снять перемычку, отвечающую за аппаратную перезагрузку контроллера** и перезапустить контроллер. После этого на экране должны появиться сообщения о загрузке BIOS и операционной системы. Затем на экран будет выдано сообщение:

For exit to Operating System press <Esc>. TimeOut is 5 seconds

.....

После появления этого сообщение в течении 5 сек нужно нажать клавишу <Esc>.

На экране должен появиться приглашающий символ #. Появление этого символа означает, что соединение установлено.

2.3.1.2 Запись или обновление программного обеспечения

Для обновления или записи программного обеспечения контроллера нужно все необходимые для записи или обновления программы записать в отдельную папку на компьютере. Затем набрать в коммуникационной программе команду:

mk crrprog

После выполнения этой команды контроллер переходит в ожидание приема файлов по протоколу Z-модем. Затем в меню программы Microsoft@ HyperTerminal выбрать пункт меню **Передача -> Отправить файл**. В появившемся окошке либо в строке **Имя файла** ввести имя директория с записываемыми файлами и добавить символы *, либо выбрать пункт меню **Обзор** выбрать любой файл, а затем изменить его имя на *. В результате, если файлы находятся в каталоге **c:\bin\kp2**, то строка Имя файла должна выглядеть следующим образом:

c:\bin\kp2*

В пункте меню **Протокол** должно быть выбрано:

Z-модем с восст. после сбоя

Для начала копирования нужно выбрать пункт меню **Отправить**.

После завершения копирования файлов нужно в окне коммуникационной программы нажать клавишу <Enter> - должен появиться приглашающий символ #.

Для увеличения скорости передачи можно перед командой **mk crrprog** ввести команду:

mk 115

Затем в коммуникационной программе необходимо изменить настройки подключения на 115200,N,8,1. После этого нужно нажать клавишу <Enter> - должен появиться приглашающий символ #. После этого нужно ввести команду **mk crrprog** и произвести выбор файлов для копирования. Для возвращения к скорости подключения 9600 нужно ввести команду:

mk 9600

и затем переключить коммуникационную программу в режим 9600, N, 8, 1 и нажать клавишу <Enter>.

2.3.1.3 Чтение и запись конфигурации контроллера

Файл с описанием конфигурации контроллера называется start.ini. Это текстовый файл, который может быть отредактирован в любом текстовом редакторе.

Внимание: при чтении файлов из контроллера, некоторые коммуникационные программы (в том числе Microsoft@ HyperTerminal) могут изменить регистр в имени файла. С точки зрения операционных систем семейства Windows файлы с именами start.ini и START.INI – это один и тот же файл. Для контроллера, работающего под управлением операционной системой ЗОСРВ «Нейтрино» - файлы с именами start.ini и START.INI – это два разных файлов. Поэтому после редактирования перед последующей записью в контроллер записываемый файл необходимо переименовать в start.ini.

В файле указаны рабочие директории программы, а также содержится перечень используемых драйверов УСО, каналов связи, драйверов ретрансляции, дополнительных модулей с их настройками.

Для чтения текущего файла введите в окне коммуникационной программы команду:

mk rwstart

Файл будет передан коммуникационной программе. Чтобы узнать, в какую директорию программа Microsoft@ HyperTerminal запишет (записала) принятый файл, нужно открыть пункт меню **Передача -> Принять файл**. Имя, высвечиваемое в окне **Имя файла** и является директориум, куда помещаются принятые файлы. В этом же меню можно изменить каталог по умолчанию, куда будут записываться принятые файлы.

Запись измененного файла start.ini в контролер может быть произведена при обновлении программного обеспечения контроллера (см. п.2.3.1.2.). Для этого в директории с файлами на компьютере должен присутствовать и файл start.ini. Можно произвести запись файла start.ini отдельно. Все действия для этого аналогичны действиям, изложенным в пункте 2.3.1.2, только вместо команды **mk cpprog** нужно выполнить команду **mk cpstart**.

2.3.1.4 Чтение и запись файлов-описателей контроллера

Файлы-описатели контроллера представляют собой двоичные файлы с расширением *.bin. В них содержится информация о подключенных к контроллеру УСО, измеряемых и расчетных параметрах и т.п. Чтение и запись файлов осуществляется аналогично пунктам 2.3.1.2 и 2.3.1.3. Для чтения файлов из контроллера нужно выполнить команду:

mk rwbin

Для записи файлов в контроллер нужно выполнить команду:

mk cpbin

Перед началом копирования можно также увеличить скорость обмена между компьютером и контроллером с помощью команды **mk 115**.

Внимание, при выполнении команды **mk cpbin, все старые файлы с расширением *.bin будут удалены из контроллера. Эта команда предполагает копирование полного набора файла описателей для контроллера.**

Внимание: при чтении файлов из контроллера, некоторые коммуникационные программы (в том числе Microsoft® HyperTerminal) могут изменить регистр в имени файла. С точки зрения операционных систем семейства Windows файлы с именами *ti32.ini* и *TI32.INI* – это один и тот же файл. Для контроллера, работающего под управлением операционной системой ЗОСРВ «Нейтрино» - файлы с именами *ti32.ini* и *TI32.INI* – это два разных файлов. Поэтому после чтения перед последующей записью в контроллер имена записываемых файлов необходимо перевести в нижний регистр..

2.3.2 Подготовка в режиме соединения по локальной сети через встроенный ftp-сервер.

Соединение с контроллером производится с помощью любого ftp-клиента. Соединение по локальной сети через встроенный ftp-сервер возможна, только для контроллеров, имеющих сетевую карту. При этом компьютер и контроллер должны находиться в одной подсети, или соответствующим образом должна быть настроена маршрутизация внутри сети. **Для работы через встроенный ftp-сервер не требуется остановки или перезагрузки контроллера.**

Для того чтобы узнать IP адрес контроллера, к нему нужно подключиться через последовательный порт (п.2.3.1.1) и выполнить команду **ifconfig**. На экран коммуникационной программы будет выведено следующее сообщение:

```
# ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 33192
    inet 127.0.0.1 netmask 0xff000000
en0: flags=80008843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST,SHIM> mtu 1500
    address: 00:11:22:00:44:00
    media: Ethernet 10baseT full-duplex
    status: active
    inet 193.24.4.200 netmask 0xfffff00 broadcast 193.24.4.255
```

Далее для установления соединения нужно настроить ftp-клиент на соединение по IP-адресу контроллера и ввести имя пользователя и пароль. После этого можно производить чтение и запись файлов. Чтение, запись и изменение атрибутов файлов производится в соответствии с инструкцией на используемого ftp-клиента.

Для обновления программного обеспечения необходимо переписать требуемые файлы в директорий */flashdisk/progr* и присвоить им атрибут исполняемый (777).

Для чтения/записи файла *start.ini* необходимо скопировать файл *start.ini* из/в директорий */flashdisk/progr*.

Для чтения/записи файлов-описателей контроллера необходимо скопировать файлы с расширением *bin* из/в директорий */flashdisk/bin*.

После перезаписи файлов в контроллер, чтобы изменения вступили в силу, нужно перезагрузить контроллер. Для перезагрузки контроллера через ftp-соединение нужно создать файл с именем **restart** (содержимое файла не имеет значения) и переписать его в контроллер в директорию /flashdisk/progr. После перезаписи файла, контроллер будет автоматически перезагружен.

Приложение 1. Пример файла start.ini**[Disks]**

FileStartProtokol=/flashdisk/start
FlashDiskPath=/flashdisk/bin/
StaticDiskPath=/flashdisk/sram
SwopDiskPath=/flashdisk/swop

[Uso]

Uso=c5600
Uso=Aqalpha p/dev/ser4 s9600 d2 t1 A /flashdisk k

[Kanals]

Kanal=tnc a4 p/dev/ser1 s38400 n2 d-
Kanal=sercom a4 p/dev/ser3 s9600 n2 w5

[Module]

Module=librashet.so
Module=libtmkorr.so
Module=libjevent.so o/flashdisk/jevent.cfg f/flashdisk/event1
Module=libwatch586.so

[Retransl]

Retransl=udpretransl p2126

[Programms]

Приложение 2. Драйверы УСО

Имя драйвера	Поддерживаемые устройства	Типы УСО
amux	Аналоговый мультиплексор 32 входа	10
Aqalpha	Счетчики электроэнергии “Альфа”	9, 38, 39
c5600	Цифровые входы 5600	3
cap	Цифро-Аналоговый преобразователь	32
fixuso	Виртуальное УСО	250
icrcon	Драйвер модулей серии I-7000	20
mdbf	Драйвер протокола Modbus RTU	60
grp	Драйвер логометров УП23/УП24	59
usom	Драйвер модулей контроля и управления 16ТС, 8ТИ, 16ТУ	37
usom2	Драйвер модулей контроля и управления 64ТС, 32ТИ, 16ТУ, 8ЦАП	62

ДРАЙВЕР АНАЛОГОВОГО МУЛЬТИПЛЕКСОРА ДЛЯ ПЛАТЫ СШМК.758717.00077 ПИК “Прогресс”

Аналоговый мультиплексор позволяет одновременно подключить до 32 аналоговых сигналов. Всего драйвер может обслуживать до 4 мультиплексоров, подключенных к аналоговой плате. Для работы с мультиплексорами необходимо включить в файл **start.ini** драйвер **amux**. Независимо от количества используемых мультиплексоров драйвер **amux** только один раз.

Для каждого мультиплексора должны быть заданы следующие параметры адреса:

- **Port** – базовый адрес порта платы
- **NomAV** - номер аналогового входа, к которому подключен мультиплексор (1 - 4);

Для каждого мультиплексора должны быть заданы следующие параметры инициализации:

Rej – режим работы входа:

- 0 - 0V – 5V
- 1 - -5V – 5V
- 2 - 0 – 10V
- 3 - -10 – 10V

Диапазон кодов АЦП, задаваемый при описании аналоговых сигналов зависит от заданного режима измерения (однополярный, двухполярный), номинального выходного сигнала датчика и номинала резистора, установленного на входе мультиплексора. Возможные комбинации для режимов (0-5V и -5V-5V) приведены в таблице.

Номинал Датчика	Номинал резистора	Однополярный режим (0-5V)	Двуполярный режим (-5V – 5V)

18
СШМК.00103-01 42 01

		Мин. Код	Макс.код	Мин.код	Макс.код
0 - 5 mA	1 Ком	0	4095	0	2047
0 - 20 mA	250 Ом	0	4095	0	2047
0 - 20 mA	249 Ом	0	4079	0	2039
4 - 20 mA	250 Ом	819	4095	410	2047
4 - 20 mA	249 Ом	816	4079	408	2039
0 -5 В	нет	0	4095	0	2047
-5 - +5 В	нет	--	--	- 2047	2047

Пример вставки драйвера в файл **start.ini** :

Uso = amux

ДРАЙВЕР СЧИТЫВАНИЯ ИНФОРМАЦИИ СО СЧЕТЧИКОВ “Альфа”

Драйвер считывания данных со счетчиков электроэнергии “Альфа” **Agalpha** поддерживает различные виды счетчиков. Поддерживаемые типы счетчиков и правило их описания изложены в документе СШМК.00103-02 31 01-1 “Программно-аппаратный комплекс “Космотроника” Программа сбора информации со счетчиков Alpha, EvroAlpha, A1700, A1800, A3. Описание применения. ”

Пример вставки драйвера в файл **start.ini** :

Uso = Aqalpha p/dev/ser4 s9600 d2 t1 A /flashdisk k

ЦИФРОВЫЕ ВХОДЫ-ВЫХОДЫ УСО-КО

Цифровые входы-выходы УСО КО позволяет одновременно подключить до 16 дискретных входов или параметров телеуправления. Для работы с цифровыми входами-выходами УСО КО необходимо включить в файл **start.ini** драйвер **gpio86dx**. Тип УСО – 1.

Пример вставки драйвера в файл **start.ini** :

Uso=gpio86dx

ЦИФРОВАЯ ПЛАТА 5600

Цифровая плата 5600 позволяет одновременно подключить до 96 (5600 - 96) дискретных или импульсных входов или параметров телеуправления. Для работы с платами 5600 необходимо включить в файл **start.ini** драйвер **c5600**. Независимо от количества используемых плат 5600 драйвер **c5600** загружается только один раз.

Для каждой платы должны быть заданы следующие параметры адреса:

СШМК.00103-01 42 01

- ◆ адрес УСО - если в качестве значения данного параметра указан 0, то для доступа к первой плате по умолчанию будет использоваться базовый адрес 100Н, для второй - 110Н и т.д.

Для каждой платы должны быть заданы в качестве параметров инициализации маски входов, приведенные в таблице.

20
СШМК.00103-01 42 01

Маска Инициализации	Использование IN - входы OUT - выходы			
	1 – 4	5 – 8	9 -16	17 – 24
	25 –28	29 – 32	33 - 40	41 – 48
	49 – 52	53 – 56	57 - 64	65 – 72
	73 – 76	77 – 80	81 - 88	89 – 96
80	OUT	OUT	OUT	OUT
81	IN	OUT	OUT	OUT
82	OUT	OUT	OUT	IN
83	IN	OUT	OUT	IN
88	OUT	IN	OUT	OUT
89	IN	IN	OUT	OUT
8A	OUT	IN	OUT	IN
8B	IN	IN	OUT	IN
90	OUT	OUT	IN	OUT
91	IN	OUT	IN	OUT
92	OUT	OUT	IN	IN
93	IN	OUT	IN	IN
98	OUT	IN	IN	OUT
99	IN	IN	IN	OUT
9A	OUT	IN	IN	IN
9B	IN	IN	IN	IN

В масках 8A, 8B, 9A, 9B содержатся ЛАТИНСКИЕ буквы.

Если плата 5600 имеет меньше 96 входов, то для неиспользуемых входов задается маска 9B .

Пример вставки драйвера в файл **start.ini** :

Uso = c5600

ДРАЙВЕР ЦИФРО-АНАЛОГОВОГО ПРЕОБРАЗОВАТЕЛЯ.

Независимо от количества используемых преобразователей, драйвер запускается только один раз.

Для каждого модуля должны быть заданы следующие параметры адреса:

- адрес выхода на шине ISA

В качестве параметра инициализации должен быть задан идентификатор параметра, значение которого выдается на выход модуля.

Пример вставки драйвера в файл **start.ini** :

Uso = cap

21
СШМК.00103-01 42 01
ВИРТУАЛЬНОЕ УСО

Виртуальное УСО используются для описания фиктивных сигналов, используемых в каких-либо формулах расчета. Для работы с виртуальными УСО не требуется включать никаких дополнительных драйверов в файл **start.ini**.

ДРАЙВЕР МОДУЛЕЙ КОНТРОЛЯ И УПРАВЛЕНИЯ ICPCON СЕРИИ I-7000

Модули контроля и управления серии I-7000 объединяются в сеть RS-485 и подключаются к последовательному порту контроллера. Модули могут иметь различное назначение (измерение аналоговых сигналов, цифровой ввод, цифровой вывод, подсчет числа импульсов и т.п.)

Для работы с модулями необходимо включить в файл **start.ini** драйвер **icpcn**. Независимо от количества счетчиков в сети драйвер загружается только один раз. Формат запуска драйвера:

Uso = icpcn p<serp> s<speed>

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод)

По умолчанию обмен ведется на скорости **9600** бод.

Для каждого модуля должны быть заданы следующие параметры в поле адреса:

- адрес модуля в сети RS-485.

Для аналоговых модулей с диапазоном измерения, отличным от -10 - +10 В должен быть задан один из возможных диапазонов измерения.

Номер диапазона	Диапазон	Мин. код АЦП	Код АЦП для нулевого значения	Макс. код АЦП
1	-10 - +10 В	-32768	0	32767
2	-5 - +5 В	-32768	0	32767
3	-1 - +1 В	-32768	0	32767
4	-500 - +500 мВ	-32768	0	32767
5	-150 - +150 мВ	-32768	0	32767
6	-20 - +20 мА	-32768	0	32767

Для каждого модуля можно дополнительно описать два телесигнала с адресами 98 и 99. ТС98 (Есть связь с модулем) принимает значение 1 когда модуль отвечает контроллеру по сети RS-485 и 0 – в противном случае. ТС99 (Нет связи с модулем) принимает значение 1 когда модуль не отвечает контроллеру по сети RS-485 и 0 – в противном случае.

Пример вставки драйвера в файл **start.ini** :

Uso = icpcn p/dev/ser3 s9600

СШМК.00103-01 42 01

ДРАЙВЕР ОБМЕНА ИНФОРМАЦИЕЙ ПО ПРОТОКОЛУ MODBUS RTU

Драйвер обмена информацией по протоколу Modbus RTU поддерживает следующие функции протокола:

- 1,2,101,102 – для получения состояния телесигналов
- 3, 103, 113, 123, 133, 143,153, 4, 104, 114, 124, 134, 144,154 – для получения значения телеизмерений
- 5,105 – для телеуправления

Кроме этого драйвер поддерживает функции 6, 16, 106, 116 для записи данных в устройство.

Формат запуска драйвера:

Uso = mdbuf p<serp> s<speed> b<bits> n<stopbits> h<parity> w<pause> r<razd> e<err>
m<mnoj>

или

Uso = mdbf80 p<serp> s<speed> b<bits> n<stopbits> h<parity> w<pause> r<razd> e<err>
m<mnoj>

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод)

bits – количество бит в байте (7 или 8) - по умолчанию 8

stopbits – количество стоповых бит (1 или 2) – по умолчанию 1

parity – контроль четности (n – отсутствует, e - контроль четности o – контроль нечетности) – по умолчанию n (контроль отсутствует)

pause - пауза между посылками в мсек (по умолчанию – 0)

razd – если ключ имеет значение '-', то драйвер осуществляет монополярный захват последовательного порта. Если значение '+' (по умолчанию), то драйвер работает в режиме разделяемого порта.

err - количество ошибок при обмене с устройством, после чего вырабатывается сигнал "Нет связи с устройством" . Если значение больше 255, то сигнал об отсутствии связи вырабатываться не будет.

mnoj – множитель, определяющий величину тайм-аута обмена с устройством (по умолчанию – 10). При увеличении параметра время ожидания ответа от устройства увеличивается, при уменьшении – уменьшается.

Драйвер позволяет считывать информацию блоками (до 32 параметров в блоке).

Для каждого блока должны быть заданы следующие параметры адреса:

- адрес контроллера Modbus в сети RS-485)

Для каждого блока должны быть заданы следующие параметры инициализации:

СШМК.00103-01 42 01

параметры инициализации описываются двумя строками:

- номер используемой функции протокола Modbus
- адрес начального регистра в блоке

Для драйвера mdbf для блоков с использованием функции 1,2,101,102 могут быть описаны телесигналы с адресами от 1 до 32.

Для драйвера mdbf80 для блоков с использованием функции 1,2,101,102 могут быть описаны телесигналы с адресами от 1 до 80.

Для драйвера mdbf для блоков с использованием функций 3, 103, 113, 123, 133, 143,153, 4, 104, 114, 124, 134, 144,154 могут быть описаны телеизмерения с адресами от 1 до 32.

Для драйвера mdbf80 для блоков с использованием функций 3, 103, 113, 123, 133, 143,153, 4, 104, 114, 124, 134, 144,154 могут быть описаны телеизмерения с адресами от 1 до 80.

Для драйвера mdbf для блоков с использованием функции 5,105 могут быть описаны параметры телеуправления с адресами от 1 до 32.

Для драйвера mdbf80 для блоков с использованием функции 5,105 могут быть описаны параметры телеуправления с адресами от 1 до 80.

Дополнительно для любого блока может быть описан телесигнал с адресом 91 – нет связи с блоком и 90 – есть связь с блоком.

Описания функций получения телеизмерений:

3 – стандартная функция Modbus.

103 – запрос по функции 3, ответ интерпретируется как число с плавающей точкой инверсное.

113 – запрос по функции 3, ответ интерпретируется как число с плавающей точкой без инверсии.

123 – запрос по функции 3, ответ интерпретируется как 4 байтное беззнаковое число с инверсией.

133 – запрос по функции 3, ответ интерпретируется как 4 байтное беззнаковое число без инверсии.

143 – запрос по функции 3, ответ интерпретируется как 4 байтное число со знаком с инверсией.

153 – запрос по функции 3, ответ интерпретируется как 4 байтное число со знаком без инверсии.

4 – стандартная функция Modbus.

104 – запрос по функции 4, ответ интерпретируется как число с плавающей точкой инверсное.

114 – запрос по функции 4, ответ интерпретируется как число с плавающей точкой без инверсии.

124 – запрос по функции 4, ответ интерпретируется как 4 байтное беззнаковое число с инверсией.

134 – запрос по функции 4, ответ интерпретируется как 4 байтное беззнаковое число без инверсии.

144 – запрос по функции 4, ответ интерпретируется как 4 байтное число со знаком с инверсией.

154 – запрос по функции 4, ответ интерпретируется как 4 байтное число со знаком без инверсии.

Описания функций получения телесигналов:

1 и 2 – стандартные функции Modbus

СШМК.00103-01 42 01

101 – производится чтение по функции 3 и выделение соответствующего бита в прочитанном регистре

102 – производится чтение по функции 4 и выделение соответствующего бита в прочитанном регистре

Описания функций телеуправления:

5 - стандартные функция Modbus

105 – телеуправление для устройств УСО-К

Описание функций записи в устройство:

6 и 16 – стандартные функции Modbus с автоматической синхронизацией данных контроллера с устройством

106 и 116 – функции Modbus 6 и 16, синхронизация данных контроллера с устройством производится по срабатыванию телеуправления с адресом 100.

Пример вставки драйвера в файл **start.ini** :

```
Uso = mdbf p/dev/ser3 s9600
```

ДРАЙВЕР ОБМЕНА ИНФОРМАЦИЕЙ ПО ПРОТОКОЛУ MODBUS TCP

Драйвер обмена информацией по протоколу Modbus TCP поддерживает следующие функции протокола:

- 1,2,101,102 – для получения состояния телесигналов
- 3, 103, 113, 123, 133, 143,153, 4, 104, 114, 124, 134, 144,154 – для получения значения телеизмерений
- 5 – для телеуправления

Кроме этого драйвер поддерживает функции 6, 16, 106, 116 для записи данных в устройство.

Формат запуска драйвера:

```
Uso = mdbtcp i<ipaddr w<pause>
```

где **ipaddr** – ip-адрес контроллера, с которого устанавливается соединение с устройствами
pause - пауза между посылками в мсек (по умолчанию – 0)

Драйвер позволяет считывать информацию блоками (до 32 параметров в блоке).

Для каждого блока должны быть заданы следующие параметры адреса:

- адрес контроллера Modbus в сети RS-485.
- IP адрес контроллера
- Порт для подключения

Для каждого блока должны быть заданы следующие параметры инициализации:

параметры инициализации описываются тремя строками:

- номер используемой функции протокола Modbus
- адрес начального регистра в блоке

Для блоков с использованием функции 1,2,101,102 могут быть описаны телесигналы с адресами от 1 до 32.

Для блоков с использованием функций 3, 103, 113, 123, 133, 143,153, 4, 104, 114, 124, 134, 144,154 могут быть описаны телеизмерения с адресами от 1 до 32.

Для блоков с использованием функции 5,105 могут быть описаны параметры телеуправления с адресами от 1 до 32.

Дополнительно для любого блока может быть описан телесигнал с адресом 91 – нет связи с блоком и 90 – есть связь с блоком.

Описания функций получения телеизмерений:

3 – стандартная функция Modbus.

103 – запрос по функции 3, ответ интерпретируется как число с плавающей точкой инверсное.

113 – запрос по функции 3, ответ интерпретируется как число с плавающей точкой без инверсии.

123 – запрос по функции 3, ответ интерпретируется как 4 байтное беззнаковое число с инверсией.

133 – запрос по функции 3, ответ интерпретируется как 4 байтное беззнаковое число без инверсии.

143 – запрос по функции 3, ответ интерпретируется как 4 байтное число со знаком с инверсией.

153 – запрос по функции 3, ответ интерпретируется как 4 байтное число со знаком без инверсии.

4 – стандартная функция Modbus.

104 – запрос по функции 4, ответ интерпретируется как число с плавающей точкой инверсное.

114 – запрос по функции 4, ответ интерпретируется как число с плавающей точкой без инверсии.

124 – запрос по функции 4, ответ интерпретируется как 4 байтное беззнаковое число с инверсией.

134 – запрос по функции 4, ответ интерпретируется как 4 байтное беззнаковое число без инверсии.

144 – запрос по функции 4, ответ интерпретируется как 4 байтное число со знаком с инверсией.

154 – запрос по функции 4, ответ интерпретируется как 4 байтное число со знаком без инверсии.

Описания функций получения телесигналов:

1 и 2 – стандартные функции Modbus

СШМК.00103-01 42 01

101 – производится чтение по функции 3 и выделение соответствующего бита в прочитанном регистре

102 – производится чтение по функции 4 и выделение соответствующего бита в прочитанном регистре

Описания функций телеуправления:

5 - стандартные функция Modbus

105 – телеуправление для устройств УСО-К

Описание функций записи в устройство:

6 и 16 – стандартные функции Modbus с автоматической синхронизацией данных контроллера с устройством

106 и 116 – функции Modbus 6 и 16, синхронизация данных контроллера с устройством производится по срабатыванию телеуправления с адресом 100.

Пример вставки драйвера в файл **start.ini** :

```
Uso = mdbtcp i193.24.4.50 w10
```

ДРАЙВЕР ЛОГОМЕТРОВ УП23/УП24

Логометры УП23/УП24 подключаются к последовательному порту контроллера через интерфейс RS232 / RS485. Для работы с модулями необходимо включить в файл **start.ini** драйвер **rpn**. Логометры должны быть настроены на работу по протоколу Modbus ASCII. Независимо от количества контроллеров в сети драйвер загружается только один раз. Формат запуска драйвера:

```
Uso = rpn p<serp> s<speed>
```

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод)

Для каждого модуля должны быть заданы следующие параметры адреса:

– адрес модуля в сети RS-485.

Параметры инициализации для модуля не требуются.

Пример вставки драйвера в файл **start.ini** :

```
Uso = rpn p/dev/ser3 s9600
```

СШМК.00103-01 42 01**ДРАЙВЕР МОДУЛЕЙ КОНТРОЛЯ И УПРАВЛЕНИЯ УСО-М 16ТС, 8ТИ, 16ТУ**

Модули контроля и управления серии УСО-М объединяются в сеть RS-485 и подключаются к последовательному порту контроллера. Каждый модуль может включать в себя дискретные или импульсные входы, аналоговые входы, датчик измерения температуры, выходы телеуправления.

Для работы с модулями необходимо включить в файл **start.ini** драйвер **usom**. Независимо от количества модулей в сети драйвер загружается только один раз. Формат запуска драйвера:

```
Uso = usom p<serp> s<speed>
```

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод)

По умолчанию обмен ведется на скорости **9600** бод.

При описании модуля первый телесигналы или счетчики импульсов имеют адреса с 1 по 16, параметр температуры – адрес 100, телеизмерения – адреса 101-108, параметры телеуправления – адреса 201-216.

При описании температурного датчика указывается диапазон измеряемых кодов АЦП - -110,200, значения, соответствующие кодам АЦП - -55,100.

Для параметров телеизмерений минимальный код АЦП равен 0, максимальный 16383.

Для каждого модуля можно дополнительно описать два телесигнала с адресами 90 и 91. ТС90 (Есть связь с модулем) принимает значение 1 когда модуль отвечает контроллеру по сети RS-485 и 0 – в противном случае. ТС91 (Нет связи с модулем) принимает значение 1 когда модуль не отвечает контроллеру по сети RS-485 и 0 – в противном случае.

Для каждого модуля должны быть заданы следующие параметры адреса:

– адрес модуля в сети RS-485.

Параметры инициализации для модуля не требуются.

Пример вставки драйвера в файл **start.ini** :

```
Uso = usom p/dev/ser3 s9600
```

ДРАЙВЕР МОДУЛЕЙ КОНТРОЛЯ И УПРАВЛЕНИЯ УСО-М 64ТС, 32ТИ, 16ТУ, 8ЦАП

Модули контроля и управления серии УСО-М объединяются в сеть RS-485 и подключаются к последовательному порту контроллера. Каждый модуль может включать в себя

СШМК.00103-01 42 01

дискретные или импульсные входы, аналоговые входы, датчик измерения температуры, выходы телеуправления или каналы цифро-аналогового преобразователя (ЦАП).

Для работы с модулями необходимо включить в файл **start.ini** один из драйверов **usom2**. Независимо от количества модулей в сети драйвер загружается только один раз. Формат запуска драйвера:

```
Uso = usom2 p<serp> s<speed>
```

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод)

По умолчанию обмен ведется на скорости **9600** бод.

При описании модуля первый телесигналы имеют адреса с 1 по 64, счетчики импульсов имеют адреса с 1 по 16, параметр температуры – адрес 100, телеизмерения – адреса 1-32, параметры телеуправления – адреса 1-16.

При описании температурного датчика указывается диапазон измеряемых кодов АЦП - -110,200, значения соответствующие кодам АЦП - -55,100.

Для параметров телеизмерений минимальный код АЦП равен 0, максимальный 16383.

Для каждого модуля можно дополнительно описать два телесигнала с адресами 90 и 91. ТС90 (Есть связь с модулем) принимает значение 1 когда модуль отвечает контроллеру по сети RS-485 и 0 – в противном случае. ТС91 (Нет связи с модулем) принимает значение 1 когда модуль не отвечает контроллеру по сети RS-485 и 0 – в противном случае.

Для каждого модуля должны быть заданы следующие параметры адреса:

– адрес модуля в сети RS-485.

Для каждого модуля должны быть заданы следующие параметры инициализации:

– тип модуля (1 – 64 ТС, 2 – 32 ТИ, 3 – 16 ТУ, 4 ЦАП)

– для ЦАП дополнительно должен быть задан идентификатор параметра, значение которого подается на выход ЦАП.

Пример вставки драйвера в файл **start.ini** :

```
Uso = usom2 p/dev/ser3 s9600
```

Выход ЦАП имеет диапазон кодов 0 – 4095.

ДРАЙВЕР МОДУЛЕЙ КОНТРОЛЯ И УПРАВЛЕНИЯ УСО ТМ

Модули контроля и управления серии УСО-ТМ объединяются в сеть RS-485 и подключаются к последовательному порту контроллера. Каждый модуль может включать в себя дискретные или импульсные входы, аналоговые входы, датчик измерения температуры, выходы телеуправления или каналы цифро-аналогового преобразователя (ЦАП), а также встроенный прибор технического учета электроэнергии.

СШМК.00103-01 42 01

Для работы с модулями необходимо включить в файл **start.ini** один из драйверов **usotm**. Независимо от количества модулей в сети драйвер загружается только один раз. Формат запуска драйвера:

Uso = usotm p<serp> s<speed>

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод)

По умолчанию обмен ведется на скорости **9600** бод.

Текущие параметры УСО ТМ имеют следующие адреса:

Адрес	Параметр
1	Активная мощность потребленная
2	Активная мощность выданная
3	Реактивная мощность потребленная
4	Реактивная мощность выданная
5	Полная мощность
6	Частота
7	Ток фазы А
8	Ток фазы В
9	Ток фазы С
10	Напряжение фазы А
11	Напряжение фазы В
12	Напряжение фазы С
13	Температура с измерителя мощности
21 - 52	Значение дополнительных аналоговых входов
101...	Значения телесигналов
151 ...	Значения импульсных входов
201 ...	Параметры телеуправления
90	Телесигнал – есть связь с устройством
91	Телесигнал – нет связи с устройством
100	Температура с платы ТИ внутренняя
99	Температура с платы ТИ наружная

Для каждого модуля должны быть заданы следующие параметры адреса:

– адрес модуля в сети RS-485.

Для каждого модуля должны быть заданы следующие параметры инициализации:

- Коэффициент трансформации тока
- Коэффициент трансформации напряжения

Пример вставки драйвера в файл **start.ini** :

Uso = usotm p/dev/ser3 s9600

СШМК.00103-01 42 01

ДРАЙВЕР УДАЛЁННОГО УСО НА БАЗЕ ЗОСРВ «Нейтрино».

Драйвер удалённого УСО на базе контроллера ЗОСРВ «Нейтрино» используется для передачи параметров из удалённого в головной контроллер через последовательный порт.

Формат запуска драйвера:

```
Us0 = serpr p<serp> s<speed> b<bits> n<stopbits> h<parity> w<pause> e<err> m<mnoj>  
r<srez>
```

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод)

bits – количество бит в байте (7 или 8) - по умолчанию 8

stopbits – количество стоповых бит (1 или 2) – по умолчанию 1

parity – контроль четности (n – отсутствует, e - контроль четности o – контроль нечетности) – по умолчанию n (контроль отсутствует)

pause - пауза между посылками в мсек (по умолчанию – 0)

err - количество ошибок при обмене с устройством, после чего вырабатывается сигнал “Нет связи с устройством” . Если значение больше 255, то сигнал об отсутствии связи вырабатываться не будет.

mnoj – множитель, определяющий величину тайм-аута обмена с устройством (по умолчанию – 10). При увеличении параметра время ожидания ответа от устройства увеличивается, при уменьшении – уменьшается.

srez – период запроса среза с удалённого контроллера в секундах. По умолчанию - 600 (10 минут).

На удалённом контроллере для передачи данных используется драйвер канала sercom.

При описании параметров на головном КП должны соблюдаться следующие правила:

- Номер группы и номер параметра должны совпадать на головном и удалённом КП.
- Константы на головном КП описываются как телеизмерения.
- Адреса параметров на головном КП могут быть любыми, но не должны пересекаться.
- Адреса 90 и 91 зарезервированы для телесигналов “Есть связь с УСО” и “Нет связи с УСО”
- Автоматическое генерирование сигналов для головного КП может быть сделано с помощью программы KpView.exe через пункт меню “Таблицы для головного КП”.

Приложение 3. Драйверы каналов связи

Имя драйвера	Краткое описание	Имя соответствующего драйвера на сервере ввода- вывода	Имя драйвера ретрансляции
sercom	Обмен через последовательный порт	rs485.exe	rs485retransl
tnc	Обмен по радиоканалу	tnc_2	отсутствует
tcpqnx	Обмен по сети	tcpqnx	отсутствует
udpqnx	Обмен по локальной сети по протоколу UDP	отсутствует	udpretransl
tcparm	Обмен по сети с АРМ	tcparm	отсутствует

Общие ключи для драйверов каналов связи

Все драйверы каналов связи поддерживают следующий необязательный набор ключей:

a<adress> **b**<timereinit> **c**<comp> **e**<sizee> **f**<fsrez> **g**<filtr> **h**<noprogram> **k**<ktime> **l**<lennp> **n**<nombuf> **o**<sizeo> **r**<sizee> **t**<sizee> **u**<nochv> **w**<timep> **x**<timesave> **y**<sendconst> **q**<addkeys> **-k**<ctrl>

- **adress** – адрес контроллера в канале связи (по умолчанию адрес контроллера берется из прошивки)
- **timereinit** – величина паузы в работе канала в секундах, после чего проводится его переинициализация.
- **comp** – включает (“+”) или выключает (“-“) сжатие информации при передаче по каналу связи. По умолчанию сжатие включено.
- **sizee** – размер буфера событий в килобайтах (по умолчанию – 20). В этом буфере хранятся события об изменениях значений и состояний параметров, а также системная информация (перезагрузки, коррекции времени КП и УСО). Вся информация хранится в буфере до момента ее передачи на сервер ввода-вывода. При заполнении буфера наиболее старые события будут из него удаляться, чтобы освободить место для вновь происшедших событий.
- **fsrez** – задает формат среза, передаваемого с контроллера на сервер – обычный (“-“) или расширенный (“+”). При передаче в обычном формате в срезе присутствует одна метка времени (время отправки среза) для всех передаваемых параметров. При передаче в расширенном формате в срезе для каждого параметра передается время последнего измерения. Расширенный срез имеет больший размер при том же наборе параметров. По умолчанию срез передается в обычном формате.

СШМК.00103-01 42 01

- **filtr** – имя файла фильтра. Если указан файл фильтра, то по каналу будет передаваться информация по параметрам, которые есть в файле фильтра. Файл фильтра представляет текстовый файл, в каждой строке которого содержатся два числа – номер группы и номер параметра.
- **noprogram** - разрешает (“+”) или запрещает (“-“) удаленное программирование контроллера по каналу связи. По умолчанию удаленное программирование разрешено.
- **vertime** – используется для запрета коррекции времени контроллера по данному каналу связи. Если коррекция запрещена – вместо vertime должен стоять знак минус (“-“), если разрешена – плюс (“+”). По умолчанию коррекция времени разрешена.
- **lenp** – длина пачки, передаваемой по каналу связи. Размер пачки по умолчанию различен для различных каналов связи.
- **nombuf** – номер буфера (от 1-10), создаваемого каналом связи для хранения событий. Этот параметр требуется только при резервировании каналов связи. Тогда для основного и резервного канала должен быть указан один и тот же номер буфера. В этом случае, событие, переданное по одному из резервируемых каналов, удаляется из буфера, что предотвращает дублирование информации. По умолчанию для каждого канала создается свой буфер событий. Этот режим используется для передачи информации с контроллера на разные сервера ввода-вывода.
- **sizeo** – размер буфера событий в килобайтах, передаваемых за один сеанс связи. По умолчанию равен размеру трансферного буфера. Размер этого буфера не может превышать размер трансферного буфера.
- **sizer** - размер буфера периодической информации в килобайтах (по умолчанию – 20). В этом буфере хранится периодическая информация о значениях телеизмерений (трехминутные значения, тридцатиминутные значения и т.п.). Вся информация хранится в буфере до момента ее передачи на сервер ввода-вывода. При заполнении буфера более старая информация будет из него удаляться, чтобы освободить место для новой.
- **sizet** – размер трансферного буфера в килобайтах (по умолчанию – 30). Буфер используется для хранения промежуточных данных до момента их передачи серверу ввода-вывода.
- **timep** – пауза между пачками в миллисекундах, при передаче данных на сервер. Эта величина по умолчанию различна для разных каналов связи. Изменение этого параметра в сторону увеличения может потребоваться для нестабильно работающих каналов связи.
- **sendconst** - разрешает (“+”) или запрещает (“-“) передачу значений уставок в срезе. По умолчанию передача значений уставок разрешена.
- **nochv** - разрешает (“+”) или запрещает (“-“) изменение значений параметров по каналу связи. По умолчанию изменение значений параметров разрешено.
- **timesave** – период сохранения буфера событий в сек. Параметр задает периодичность сохранения буфера событий на диск. Параметр может принимать значение -1 – сохранение отключено, 0 – производится сохранение при каждом изменении буфера. Значение по умолчанию 0.
- **addkeys** – дополнительные ключи для драйвера канала, могут принимать значения:

ctrl – активирует мониторинг за работой драйвера. Задаёт время в минутах, если по истечении которого не будет принято ни одного байта, то будет произведена перезагрузка комплекса с записью в журнале событий о сбое канала.

Драйвер обмена через последовательный порт

Драйвер обмена через последовательный порт используется для обмена с сервером по интерфейсу RS-232/422/485. Кроме общих ключей драйвер поддерживает следующие ключи:

```
sercom p<serp> s<speed>
```

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод)

Пример записи в **start.ini**:

```
Kanal = sercom a1 p/dev/ser3 s9600 n2 w5
```

Драйвер обмена по радиоканалу

Драйвер обмена через последовательный порт используется для обмена с сервером через модем. Кроме общих ключей драйвер поддерживает следующие ключи:

```
tnc p<serp> s<speed> v<vers> d<dsr>
```

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена с модемом (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод) – *обычно 38400 бод.*

vers - версия прошивки модема (1 или 2). Необязательный параметр. Версия 1 используется для модема со скоростью 2400 бод. По умолчанию -2.

dsr – включает (“+”) или выключает (“-“) управление модемом по линиям DSR/RTS. Необязательный параметр. По умолчанию управление линией включено. **Внимание: при использовании модема P9 управление модемом по линиям DSR/RTS должно быть выключено.**

Пример записи в **start.ini**:

```
Kanal = tnc a9 p/dev/ser1 s38400 n2 d-
```

Драйвер обмена по сети

Драйвер обмена по сети используется для обмена с сервером через локальную сеть.

Чтобы такой обмен был возможен, на контроллере должен быть установлен правильный TCP адрес и при необходимости прописан шлюз до сервера. На контроллере и сервере можно проверить доступность соединения с помощью утилиты **ping**, входящей в состав операционной системы ЗОСРВ «Нейтрино» и Microsoft Windows. Кроме общих ключей драйвер поддерживает следующие ключи:

СШМК.00103-01 42 01

```
tcpqnx i<mytcp,myport,servertcp,serverport> z<init>
```

где **mytcp** – ТСП адрес контроллера.

myport – порт контроллера в сети.

servertcp – ТСП адрес сервера, с которым производится обмен.

serverport – порт сервера в сети.

init – включает (“+”) или выключает (“-”) инициативный режим. При включенном инициативном режиме при возникновении новых событий контроллер сам выходит на соединение с сервером для передачи информации. При выключенном инициативном режиме контроллер передает информацию только по запросу сервера ввода-вывода. По умолчанию инициативный режим включен. *Внимание: когда канал используется в качестве одного из резервируемых (основного или резервного), инициативный режим должен быть отключен. На сервере при этом в файле StartWrt.mdb в таблице LoadDrv в поле RejimDrv должен присутствовать ключ ZaproEvent=Enable;*

Пример записи в **start.ini**:

```
Kanal=tcpqnx a3 i10.49.4.24,2155,10.42.18.2,2155 z- n2
```

Драйвер обмена по локальной сети по протоколу UDP

Драйвер обмена по локальной сети по протоколу UDP используется для обмена с сервером через промежуточный контроллер, находящийся в локальной сети. Контроллер, через который осуществляется ретрансляция, может связываться с сервером ввода-вывода по любому каналу связи. Кроме общих ключей драйвер поддерживает следующие ключи:

```
udpqnx p<port>
```

где **port** – номер порта UDP, через который осуществляется обмен. Такой же номер порта должен быть указан для драйвера ретрансляции `udptransl` на контроллере, через который осуществляется ретрансляция.

Пример записи в **start.ini**:

```
Kanal=udpqnx a9 p2126 n2
```

Драйвер обмена по сети с АРМ

Драйвер обмена по сети используется для обмена с АРМом через локальную сеть, минуя сервер ввода-вывода. Чтобы такой обмен был возможен, на контроллере должен быть установлен правильный ТСП адрес и при необходимости прописан шлюз до АРМа. На контроллере и АРМе можно проверить доступность соединения с помощью утилиты **ping**,

СШМК.00103-01 42 01

входящей в состав операционной системы ЗОСРВ «Нейтрино». и Microsoft Windows. Кроме общих ключей драйвер поддерживает следующие ключи:

```
tcparm i<mytcp,myport,armtcp,armport>
```

где **mytcp** – TCP адрес контроллера.

myport – порт контроллера в сети.

armtcp – TCP адрес АРМа, с которым производится обмен.

armport – порт АРМа в сети.

Пример записи в **start.ini**:

```
Kanal= tcparm a3 i10.49.4.24,2155,10.42.18.2,2155 z- n2
```

Драйвер передачи данных по протоколу ГОСТ Р МЭК 60870-5-104

Драйвер осуществляет передачу данных с контроллера по протоколу ГОСТ Р МЭК 60870-5-104 Кроме общих ключей драйвер поддерживает следующие ключи:

```
p104send i<mytcp,myport,sizecausetrans,sizeadrasdu,sizeobjadr>
```

где **mytcp** – TCP адрес контроллера.

myport – порт контроллера в сети (по умолчанию 2404).

sizecausetrans – размер поля причина передачи в байтах (по умолчанию 2).

sizeadrasdu – размер поля адреса ASDU в байтах (по умолчанию 2).

sizeobjadr – размер поля адреса информационного объекта в байтах (по умолчанию 3).

.

Пример записи в **start.ini**:

```
Kanal=p104send i10.49.4.24,2404
```

Драйвер передачи информации по протоколу MODBUS TCP

Формат запуска драйвера:

```
Kanals = mdbtcp retr i<ipaddr,port>
```

где **ipaddr** – IP-адрес с которого осуществляется передача

port – TCP порт, с которого осуществляется передача (по умолчанию – 502)

36
СШМК.00103-01 42 01

Драйвер передачи информации по протоколу MODBUS RTU

Формат запуска драйвера:

Kanals = mdbrturetr p<serp> s<speed> b<bits> n<stopbits> h<parity> w<pause>

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод)

bits – количество бит в байте (7 или 8) - по умолчанию 8

stopbits – количество стоповых бит (1 или 2) – по умолчанию 1

parity – контроль четности (n – отсутствует, e - контроль четности o – контроль нечетности) – по умолчанию n (контроль отсутствует)

pause - пауза между посылками в мсек (по умолчанию – 0)

Приложение 4. Драйверы ретрансляции

Имя драйвера	Краткое описание
rs485retransl	Ретрансляция через последовательный порт
udpretransl	Ретрансляция по локальной сети по протоколу UDP
rs485dosretransl	Ретрансляция через последовательный порт на контроллер, работающий под операционной системой DOS

Драйвер ретрансляции через последовательный порт

Драйвер rs485retransl предназначен для ретрансляции запросов удаленным контроллерам и передачи их ответов серверу ввода-вывода через последовательный порт по протоколу RS-232/422/485. Драйвер поддерживает следующие ключи:

Retransl = rs485retransl p<serp> s<speed>

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (/dev/ser1, dev/ser2 и т.п.)

speed - скорость обмена через последовательный порт (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод).

Пример записи в **start.ini**:

Retransl = rs485retransl p/dev/ser1 s9600

Драйвер ретрансляции по локальной сети по протоколу UDP

Драйвер `udpretransl` предназначен для ретрансляции запросов удаленным контроллерам и передачи их ответов серверу ввода-вывода по локальной сети по протоколу UDP. Драйвер поддерживает следующие ключи:

`Retransl = udpretransl p<port>`

где **port** – номер порта UDP, через который осуществляется обмен. Такой же номер порта должен быть указан для драйвера канала `udrqnх` на удаленных контроллерах, на которые осуществляется ретрансляция.

Пример записи в **start.ini**:

`Retransl = udpretransl p2126`

Драйвер ретрансляции через последовательный порт на контроллер, работающий под управлением операционной системы DOS.

Драйвер `rs485dosretransl` предназначен для ретрансляции запросов удаленным контроллерам и передачи их ответов серверу ввода-вывода через последовательный порт по протоколу RS-232/422/485. Драйвер поддерживает следующие ключи:

`Retransl = rs485dosretransl p<serp> s<speed>`

где **serp** - имя последовательного порта, через который должен вестись обмен информацией с устройствами (`/dev/ser1`, `dev/ser2` и т.п.)

speed - скорость обмена через последовательный порт (одно из значений 1200, 2400, 4800, 9600, 19200, 38400, 57600 бод, 115200 бод).

Пример записи в **start.ini**:

`Retransl = rs485dosretransl p/dev/ser1 s9600`

Приложение 5. Подключаемые модули

Имя модуля	Краткое описание
<code>librshet.so</code>	Вычисляет значения расчетных параметров
<code>libtmkorr.so</code>	Обеспечивает переход на летнее и зимнее время
<code>libwatch586.so</code>	Управляет сторожевым таймером
<code>libjevent.so</code>	Управление журналом событий

Вычисление значений расчетных сигналов

Модуль `librashet.so` по запросам от программы `qmicro` вычисляет значения расчетных сигналов. При запуске модуля никакие ключи не требуются.

Пример записи в **start.ini**:

`Module=librashet.so`

Переход на летнее и зимнее время

Модуль `libtmkorr.so` обеспечивает автоматический перевод времени контроллера на летнее и зимнее время. При запуске модуля никакие ключи не требуются.

Пример записи в **start.ini**:

`Module=libtmkorr.so`

Управление сторожевым таймером

Модули `libwatch586.so` для ZF-586 и `libwatch86dx.so` для УСО КО обеспечивают по запросам от программы `qmicro` управление сторожевым таймером контроллера. В случае, если запросы от программы `qmicro` перестают поступать, сторожевой таймер производит автоматический перезапуск контроллера. Для активации сторожевого таймера на контроллере должен быть одет соответствующий джампер. При запуске модуля никакие ключи не требуются.

Пример записи в **start.ini** для ZF-586:

`Module=libwatch586.so`

Пример записи в **start.ini** для УСО КО:

`Module=libwatch86dx.so`

Управление журналом событий

Модуль `libjevent.so` обеспечивает управление журналом событий на контроллере. При запуске модуля могут использоваться следующие ключи:

`Module = libjevent.so s<size> o<fileconfig> f<filejournal>`

size – максимальное количество записей в журнале (по умолчанию 5000). При достижении максимального количества записей к имени файла с журналом добавляется расширение “.old” и он сохраняется под этим именем и создается новый пустой файл журнала.

fileconfig – имя файла с конфигурацией журнала

filejournal – имя файла с журналом событий.

Подробная информация о настройке журнала событий содержится в документе “ Программно-технический комплекс “Космотроника” Программа ведения журнала событий. Описание применения. СШМК.00103-04 31 01-2”

Пример записи в **start.ini**:

```
Module = libjevent.so o/flashdisk/jevent.cfg f/flashdisk/event1
```

Управление пультом оператора

Модуль **libterm.so** используется для организации диалога с помощью пульта оператора. Работа с пультом описана в документе “ Программно-технический комплекс “Космотроника” Организация диалога с помощью пульта оператора. Описание применения. СШМК.00103-01 42 02”

Пример записи в **start.ini**:

```
Module=libterm.so
```

Типовые проблемы и их решение

- 1. Не удаётся подключиться к контроллеру через ftp-соединение.**
Для проверки настройки сетевого соединения необходимо подключиться к контроллеру через последовательный порт в терминальном режиме.
Просмотр текущих сетевых настроек производится утилитой **ifconfig**.
Для тестирования связи с контроллером используется утилита **ping**.
- 2. Не поступает информация с устройства связи с объектом (УСО).**
Проверить наличие записи о драйвере УСО в файле старта системы (**start.ini**).
Через ftp-соединение проверить наличие драйвера УСО в директории **/flashdisk/progr**.
- 3. Не происходит соединение контроллера с верхним уровнем по каналу связи.**
Проверить наличие записи драйвера канала связи в файле старта системы (**start.ini**).
Проверить правильность параметров запуска драйвера канала связи в файле старта системы (**start.ini**) (Номер последовательного порта, скорость обмена и т.п.).
Через ftp-соединение проверить наличие канала связи в директории **/flashdisk/progr**.

Техническая поддержка

Пользователи ПО могут обратиться за технической поддержкой, направив возникающие вопросы на адрес электронной почты pik.support@pikprogress.ru или зарегистрировав обращение на портале технической поддержки <http://www.pikprogress.ru/support.html>. Для этого необходимо перейти на сайт компании АО «ПИК ПРОГРЕСС» по адресу: www.pikprogress.ru, затем перейти в раздел технической поддержки (пункт меню «Поддержка») и оформить заявку, введя данные в соответствующие поля формы. Также заявку на техподдержку можно отправить по адресу 105118, г. Москва, ул. Вольная, д.19.