



ПРОГРАММА УПРАВЛЕНИЯ
МНОГОФУНКЦИОНАЛЬНОЙ ПЛАТФОРМОЙ ТАКТОВОЙ СЕТЕВОЙ
СИНХРОНИЗАЦИИ
«СОНАТА-У»
Описание программы
МБСЕ.00010-01 13 01

АННОТАЦИЯ

В настоящем документе приведено описание Программного изделия (ПИ) «Программа управления многофункциональной платформой тактовой сетевой синхронизации «СОНАТА-У», а также указан состав технических средств, необходимых для его функционирования.

В разделе «Общие сведения» приведено обозначение и наименование ПИ, перечислено системное программное обеспечение, необходимое для функционирования ПИ, а также указан язык программирования, на котором написано ПИ.

В разделе «Функциональное назначение» указано назначение ПИ и приведены задачи, которые могут быть решены с его помощью. Сообщаются сведения о функциональных ограничениях на применение ПИ.

В разделе «Описание логической структуры» приведена структура ПИ с описанием функций составных частей и связей между ними. Описание логической структуры выполнено с учётом текста программы на исходном языке. Описан алгоритм работы при решении различных задач.

В разделе «Используемые технические средства» указаны элементы аппаратного обеспечения, которые необходимы для выполнения ПИ.

В разделе «Вызов и загрузка» указан способ вызова установленного ПИ с соответствующего носителя образа ПИ.

В разделе «Входные и выходные данные» указаны способ кодирования входных данных и виды выходных данных, а также характер их организации.

СОДЕРЖАНИЕ

1. Общие сведения.....	4
1.1. Обозначение и наименование программы.....	4
1.2. Программное обеспечение, необходимое для функционирования программы.	4
1.3. Языки программирования, на которых написана программа	4
2. Функциональное назначение	5
2.1. Назначение программы	5
2.2. Выполняемые функции	5
2.3. Классы решаемых задач	7
2.4. Функциональные ограничения на применение.....	7
3. Описание логической структуры.....	8
3.1. Алгоритм программы	8
3.2. Используемые методы.....	9
3.2.1. Методы настройки состава и конфигурации блоков МП ТСС	9
3.2.2. Методы получения статуса блоков МП ТСС.....	9
3.2.3. Методы обработки команд	10
3.2.4. Методы отображения входных данных.....	10
3.3. Структура программы.....	11
3.4. Связи программы с другими программами.....	17
3.5. Вариант развертывания ПИ	18
4. Используемые технические средства.....	19
5. Вызов и загрузка.....	20
5.1. Способ вызова программы.....	20
5.2. Объем программы	20
6. Входные и выходные данные.....	21

1. ОБЩИЕ СВЕДЕНИЯ

1.1. Обозначение и наименование программы

1.1.1. Наименование ПИ: Программа управления многофункциональной платформой тактовой сетевой синхронизации «СОНАТА-У» (далее – МП ТСС).

1.1.2. Обозначение ПИ: МБСЕ.00010-01.

1.2. Программное обеспечение, необходимое для функционирования программы

1.2.1. Установка ПИ производится на предварительно установленную операционную систему (ОС) Linux, в которой должны быть установлены следующие пакеты и программы с версиями не ниже указанной:

- Python версии 3.7+;
- Apache версии 2.4;
- PHP версии 7.0+;
- СУБД PostgreSQL 12+.

1.2.2. Штатным способом установки ПИ является перенос заранее подготовленного образа с внешнего носителя (например, карты памяти) в постоянную память (eMMC) одноплатного компьютера.

1.3. Языки программирования, на которых написана программа

1.3.1. ПИ разработано на языках JavaScript, Python, PHP.

2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

2.1. Назначение программы

2.1.1. ПИ предназначено для управления МП ТСС с использованием графического Web-интерфейса, мониторинга технического состояния блоков, входящих в состав МП ТСС, переключения между основными и резервными блоками, вычисления и отображения параметров качества синхросигналов, управления параметрами синхронизации времени (фазы).

2.2. Выполняемые функции

2.2.1. ПИ обеспечивает выполнение функции контроля и управления МП ТСС в следующих областях:

- области управления обработки неисправностей;
- области управления качеством синхросигналов;
- области управления конфигурацией;
- области управления безопасностью.

2.2.2. В области управления обработки неисправностей ПИ обеспечивает выполнение следующих функций:

- 1) обнаружения и локализации неисправностей;
- 2) индикации неисправностей входного сигнала;
- 3) ведения журнала истории событий и аварий с указанием:
 - блока (платы) – источника события;
 - типа события;
 - времени возникновения события;
- 4) отдельный просмотр событий в зависимости от их состояния: активное, маскированное, закрытое (подтвержденное);
- 5) маскирование событий и настройка категорий событий;
- 6) оповещение пользователей об аварийных событиях;
- 7) гибкие возможности по фильтрации и сортировке событий в электронном журнале;
- 8) резервное копирование и архивирование данных.

2.2.3. В области управления качеством синхросигналов ПИ обеспечивает выполнение следующих функций:

- получение и обработка данных с результатами измерений временных отклонений синхросигналов с целью вычисления максимальной относительной ошибки временного интервала (МООВИ) и девиации временного интервала (ДВИ) по каждому из каналов

- контроля параметров входных сигналов и сравнение их с устанавливаемыми масками;

- вывода результатов измерений и расчетов (МООВИ, ДВИ и др.) в графическом и табличном видах;

- анализа результатов измерений.

2.2.4. В области управления конфигурацией ПИ обеспечивает выполнение следующих функций:

1) получение инвентарной информации с плат (версий и дат прошивок, заводских номеров);

2) для плат: ввод в эксплуатацию / вывод из эксплуатации;

3) для входных сигналов:

- выбора канала;

- установки приоритетов;

- установки типа входного сигнала;

- установки уровня допустимого качества входного сигнала;

- ввод в эксплуатацию / вывод из эксплуатации входа;

4) для выходных сигналов:

- установки резервирования выходного сигнала;

- включения (выключения) выходного сигнала;

- установки уровня качества в формируемом сигнале 2 048 кбит/с;

- ввод в эксплуатацию / вывод из эксплуатации выхода;

5) в части управления:

- ведение нормативно-справочной информации;

- обеспечение Web-интерфейса для управления МП ТСС;

- интерактивное визуальное отображение структуры МП ТСС с отображением технического состояния плат, входных и выходных интерфейсов;
- включения (выключения) порта местного управления;
- установки скорости для последовательного порта;
- управления режимами работы и конфигурацией плат NTP и RTP.

2.2.5. В области управления безопасностью ПИ обеспечивает выполнение следующих функций:

- введения классов (ролей) пользователей: с разрешением только на просмотр, с разрешением на просмотр и конфигурирование, с разрешением на просмотр, конфигурирование и управление пользователями ПИ;
- гибкие возможности по блокировке пользователей: по количеству неудачных попыток, по периоду неактивности, на срок и др.;
- возможность задания политики паролей – допустимые символы и пр.;
- ведения групп пользователей;
- введения паролей и идентификаторов для пользователей.

2.2.6. В части обеспечения взаимодействия с внешними системами ПИ обеспечивает поддержку протокола SNMP.

2.3. Классы решаемых задач

2.3.1. ПИ может быть использовано для решения следующих задач:

- получение информации о техническом состоянии плат МП ТСС;
- получение информации о параметрах синхросигналов;
- переключение между основными и резервными платами;
- изменение параметров (режимов работы) выходных интерфейсов;
- переключение между входами синхросигналов;
- измерение временных отклонений и расчет МООВИ и ДВИ;
- управления режимами работы и конфигурацией плат NTP и RTP.

2.4. Функциональные ограничения на применение

ПИ предназначено для установки на плату управления, входящую в состав МП ТСС и построенную на базе одноплатного компьютера, работающего под управлением операционной системы Linux.

3. ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

3.1. Алгоритм программы

3.1.1. Блок-схема алгоритма функционирования ПИ представлена на рис. 1.

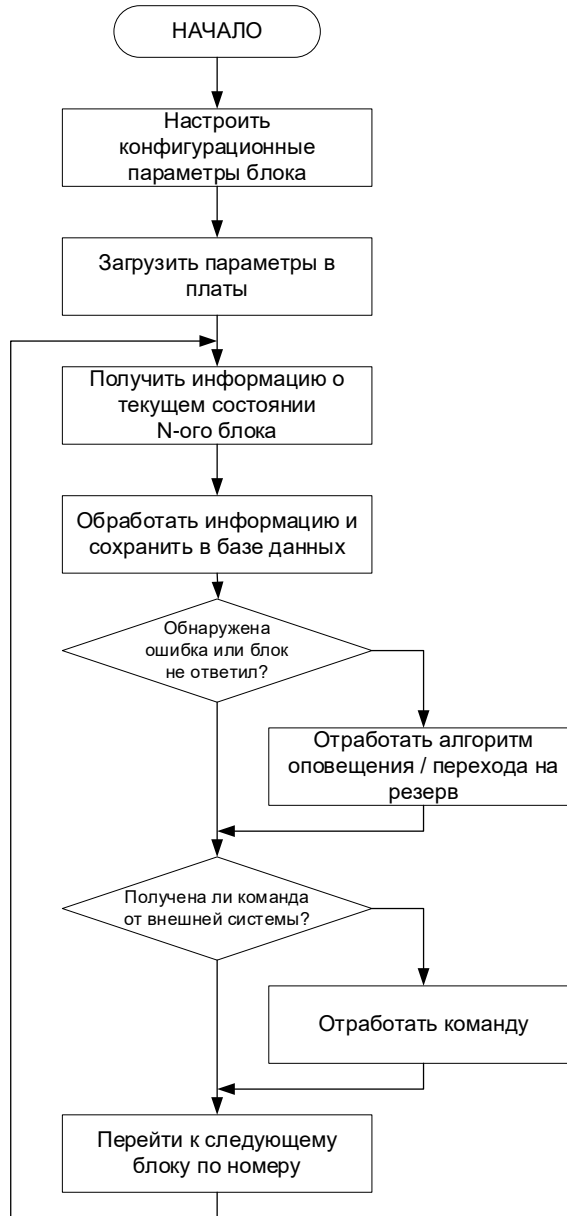


Рисунок 1 – Блок-схема алгоритма функционирования ПИ

3.1.2. В общем случае алгоритм функционирования ПИ предусматривает выполнение следующих шагов:

- 1) настройка состава блоков МП ТСС;
- 2) задание конфигурационных параметров блоков и их загрузка на соответствующие блоки;
- 3) запуск цикла сканирования получения статуса блоков;

4) получение информации о статусе блока и его сохранение в базе данных (в файле);

5) в случае обнаружения изменения статуса блока на «Авария» либо не получения ответа от платы в течение заданного интервала времени 3 раза подряд отработать алгоритм оповещения персонала и перехода на резерв, если таковой имеется, и переход определен настройками;

б) получение информации о временных отклонениях синхросигналов и вычисление МООВИ и ДВИ;

7) проверка наличия команды во входной очереди команд;

8) отработка команды при ее наличии;

9) отображение результатов вычислений параметров синхросигналов;

10) переход к следующему блоку для опроса.

3.2. Используемые методы

3.2.1. Методы настройки состава и конфигурации блоков МП ТСС

3.2.1.1. Состав и конфигурация плат МП ТСС задается посредством Web-интерфейса.

3.2.1.2. Информация о составе и конфигурации блоков сохраняется в базе данных и используется для последующего отображения внешнего вида МП ТСС и циклического опроса блоков.

3.2.1.3. Настройка состав МП ТСС подразумевает задание:

– перечня блоков;

– типов блоков;

– перечня входных / выходных интерфейсов с указанием их типов и режимов работы.

3.2.1.4. Конфигурация блоков зависит от их типа.

3.2.2. Методы получения статуса блоков МП ТСС

3.2.2.1. Получение статусов, характеризующих техническое состояние блоков МП ТСС, осуществляется путем циклического обращения к каждому блоку, получения информации о техническом состоянии блока, сохранения полученной информа-

ции в базе данных, анализа информации и выполнения заранее заданных действий, обусловленных техническим состоянием.

3.2.2.2. Периодичность опроса блоков зависит от количества установленных блоков в оборудовании. Один цикл опроса, за который обеспечивается получения статусной и дополнительной информации по всем блокам занимает не более 3 с.

3.2.2.3. В рамках опроса блоков также запрашиваются параметры временных отклонений синхросигнала по всем входам для расчета МООВИ и ДВИ.

3.2.2.4. При обнаружении отклонений параметров блоков от заданных норм осуществляется оповещение персонала путем изменения индикации внешнего вида блока на панели управления Web-интерфейса, а также путем отправки аварийных сообщений (TRAP).

3.2.3. Методы обработки команд

3.2.3.1. Команды для МП ТСС в целом или отдельных плат поступают от пользователей посредством работы с Web-интерфейсом.

3.2.3.2. Команды помещаются в очередь команд, которая периодически опрашивается в рамках цикла опроса блоков МП ТСС.

3.2.3.3. Между опросами двух блоков выполняется все команды, адресованные опрашиваемому блоку.

3.2.3.4. Информация по командам, результатам выполнения команд, источникам команд сохраняется в базе данных и является доступной для просмотра уполномоченными пользователями.

3.2.3.5. Результат выполнения команды также возвращается отправителю.

3.2.4. Методы отображения входных данных

3.2.4.1. В ПИ предусмотрено два режиме отображения входных и выходных данных:

- в графическом формате;
- в текстовом формате, включая табличное представление данных.

3.2.4.2. Для отображения используется Web-интерфейс.

3.2.4.3. Переключение между режимами отображения данных осуществляется через меню Web-интерфейса.

3.3. Структура программы

3.3.1. ПИ характеризуется двумя видами структур: структурой исходных текстов ПИ и структурой исполняемых модулей ПИ.

3.3.2. Основные исходные тексты ПИ представляют собой четыре группы программных модулей:

- Web-интерфейса администрирования, реализованного на языках программирования JavaScript и PHP;

- модуля опроса плат, реализованного на языке программирования Python;

- модуля поддержки SNMP, реализованного на языке программирования Python;

- модуля контроля состояния платы управления, реализованного на языке программирования Python.

3.3.3. Web-интерфейс состоит из серверной и клиентской частей.

3.3.4. Серверная часть Web-интерфейса реализована на базе скриптового языка PHP (требуется PHP версии не ниже 7). Для доступа к базе данных используется структурированный язык запросов SQL.

3.3.5. Клиентская часть Web-интерфейса реализована с использованием языка гипертекстной разметки HTML, каскадных таблиц стилей CSS, скриптового языка JavaScript, библиотек jQuery, jQuery UI, плагина jTable, а также свободного набора инструментов для создания сайтов и веб-приложений Bootstrap версии 4.3.1, включающего HTML- и CSS-шаблоны оформления.

3.3.6. Исходные тексты ПИ расположены в следующих каталогах на одном-платном компьютере:

- Web-интерфейс: каталог /var/www/html;

- модули, реализованные на базе языка программирования python: каталог /usr/sbin/kmt3;

3.3.7. Работа с исходными текстами может проводиться в любом текстовом редакторе. Рекомендуется использовать текстовый редактор с подсветкой синтаксиса.

3.3.8. Для работы ПИ компиляция исходных текстов не требуется. Исполняемые модули создаются автоматически при первичном обращении к ним.

3.3.9. Обобщенная структура взаимосвязей файлов Web-интерфейса приведена на рис. 2.

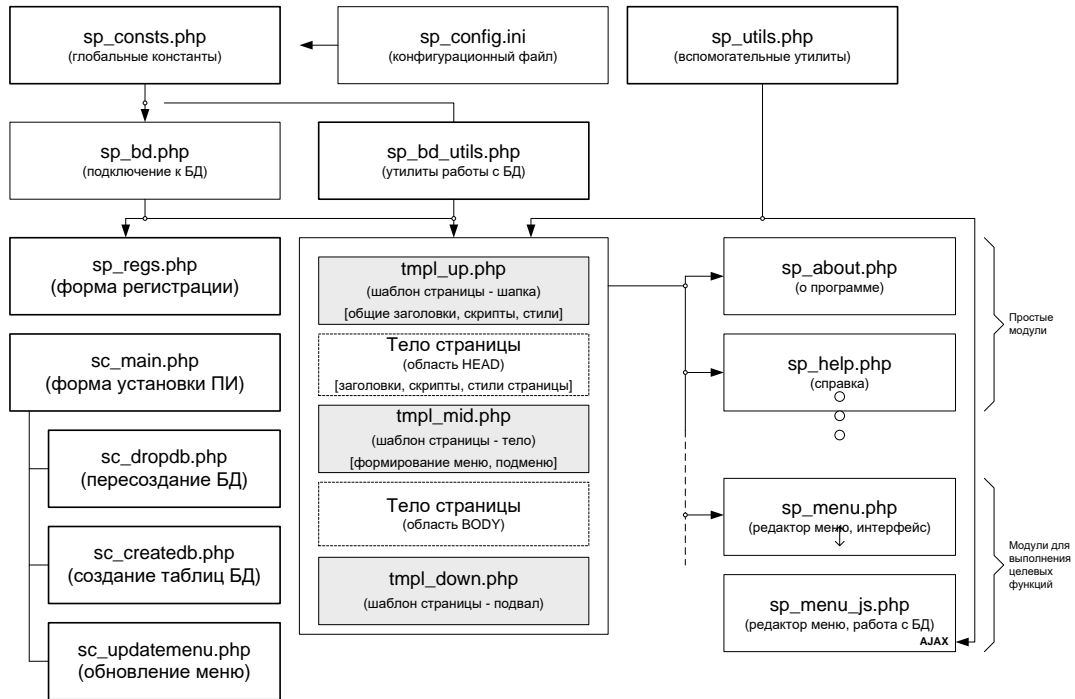


Рисунок 2

3.3.10. Обобщенная структура ПИ с указанием взаимосвязей между элементами платы управления и платами МП ТСС, представлена на рис. 3.

Обобщенная структура ПИ

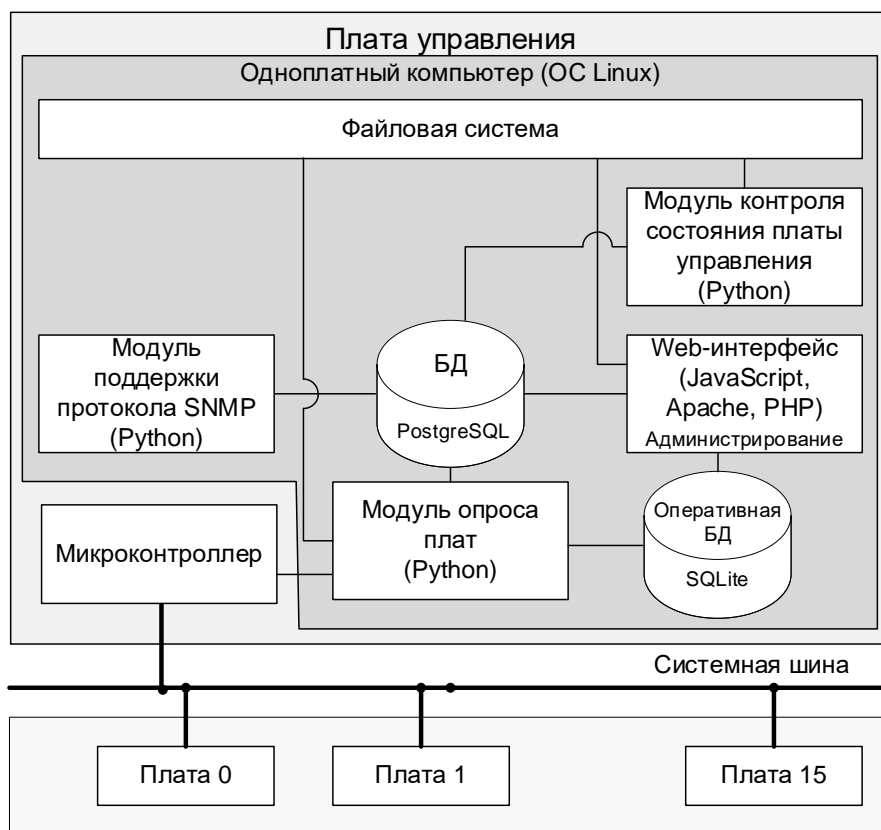


Рисунок 3

3.3.11. Обобщенная информация об основных модулях Web-интерфейса приведена в таблице 1.

Таблица 1

Идентификатор файла/каталога	Примечание
fonts	Каталог со шрифтами
images	Каталог с изображениями
library	Каталог с библиотеками (классами) PHP для работы с файлами в формате Word и Excel
load	Каталог с исходными данными для заполнения БД при установке ПИ
scripts	Каталог со скриптами javascript
styles	Каталог со стилями css
templates	Каталог с шаблонами страниц
vendor	Каталог со вспомогательными компонентами для клиентской части Web-интерфейса
sc_config.ini	Конфигурационные параметры для первичной

Идентификатор файла/каталога	Примечание
	установки и настройки ПИ
sc_consts.php	Константы для первичной установки и настройки ПИ
sp_about.php	О программе
sp_bd.php	Функции работы с основной БД
sp_bd_sys.php	Функции работы с системной БД
sp_bd_utils.php	Вспомогательные функции работы с БД
sp_captcha.php	Формирование капчи (для страницы регистрации)
sp_commands.php	Команды управления оборудованием
sp_common_js.php	Общие функции для формирования ответов на AJAX-запросы
sp_config.ini	Файл с конфигурационными параметрами
sp_consts.php	Общие константы
sp_fileupload.php	Выгрузка файлов
sp_fileupload_raw.php	Выгрузка файлов в исходном формате
sp_front.php	Внешний вид оборудования синхронизации
sp_front_js.php	Внешний вид оборудования синхронизации. Обработка AJAX-запросов от sp_front.php
sp_functional_js.php	Целевые функции (взаимодействие с оборудованием)
sp_help.php	Справочная система
sp_helpedit.php	Редактирование справки. Внешний вид
sp_helpedit_js.php	Редактирование справки. Обработка AJAX-запросов от sp_helpedit.php
sp_logcmdplate.php	Журнал передачи команд на оборудование. Внешний вид
sp_logcmdplate_js.php	Журнал передачи команд на оборудование. Обработка AJAX-запросов от модуля sp_logcmdplate.php
sp_logsession.php	Журнал просмотра сессий пользователей. Внешний вид
sp_logsession_js.php	Журнал просмотра сессий пользователей. Обработка AJAX-запросов от модуля sp_session.php
sp_logupload.php	Журнал загрузки файлов. Внешний вид
sp_logupload_js.php	Журнал загрузки файлов. Обработка AJAX-запросов от модуля sp_logupload.php

Идентификатор файла/каталога	Примечание
sp_loguser.php	Просмотр журнала действий пользователей. Внешний вид
sp_loguser_js.php	Просмотр журнала действий пользователей. Обработка AJAX-запросов от модуля sp_userlog.php
sp_mask.php	Справочник масок для оценки МОВИ (МООВИ) и ДВИ. Внешний вид
sp_mask_js.php	Журнал загрузки файлов. Обработка AJAX- запросов от модуля sp_mask.php
sp_menu.php	Работа с меню. Внешний вид
sp_menu_js.php	Работа с меню. Обработка AJAX-запросов от модуля sp_menu.php
sp_plate.php	Картотека плат. Внешний вид
sp_plate_js.php	Картотека плат. Обработка AJAX-запросов от модуля sp_plate.php
sp_regs.php	Регистрация в web-интерфейсе
sp_role.php	Роли пользователей. Внешний вид
sp_role_js.php	Роли пользователей. Обработка AJAX-запросов от модуля sp_role.php
sp_setting.php	Конфигурационные параметры. Внешний вид
sp_setting_js.php	Конфигурационные параметры. Обработка AJAX-запросов от модуля sp_setting.php
sp_statusinput.php	Справочник статусов входов. Внешний вид
sp_statusinput_js.php	Справочник статусов входов. Обработка AJAX- запросов от модуля sp_statusinput.php
sp_statusoutput.php	Справочник статусов выходов. Внешний вид
sp_statusoutput_js.php	Справочник статусов выходов. Обработка AJAX-запросов от модуля sp_statusoutput.php
sp_statusplate.php	Справочник статусов плат. Внешний вид
sp_statusplate_js.php	Справочник статусов плат. Обработка AJAX- запросов от модуля sp_statusplate.php
sp_typeinput.php	Справочник типов входов. Внешний вид
sp_typeinput_js.php	Справочник типов входов. Обработка AJAX- запросов от модуля sp_typeinput.php
sp_typeoutput.php	Справочник типов выходов. Внешний вид
sp_typeoutput_js.php	Справочник типов выходов. Обработка AJAX-

Идентификатор файла/каталога	Примечание
	запросов от модуля sp_typeoutput.php
sp_typeplate.php	Справочник типов плат. Внешний вид
sp_typeplate_js.php	Справочник типов плат. Обработка AJAX-запросов от модуля sp_typeplate.php
sp_upload.php	Загрузка файлов
sp_upload_data.php	Загрузка данных
sp_upload_utils.php	Вспомогательные утилиты для загрузки
sp_user.php	Картотека пользователей. Внешний вид
sp_user_js.php	Картотека пользователей. Обработка AJAX-запросов от модуля sp_user.php
sp_utils.php	Утилиты
sp_utils_crc8.php	Вычисление кода CRC8
sp_utils_excel.php	Вспомогательные функции для формирования файлов Excel
sp_utils_files.php	Вспомогательные функции для формирования файлов
sp_utils_template.php	Вспомогательные функции для формирования шаблонов
tmpl_down.php	Шаблон страницы (подвал)
tmpl_down_full.php	Шаблон страницы (подвал). Полноформатный вывод (с электронными журналами)
tmpl_down_mini.php	Шаблон страницы (подвал). Сокращенный вывод (без электронных журналов)
tmpl_mid.php	Шаблон страницы (тело)
tmpl_mid_full.php	Шаблон страницы (тело). Полноформатный вывод (с электронными журналами)
tmpl_mid_mini.php	Шаблон страницы (тело). Сокращенный вывод (без электронных журналов)
tmpl_up.php	Шаблон страницы (шапка)
tmpl_up_full.php	Шаблон страницы (шапка). Полноформатный вывод (с электронными журналами)
tmpl_up_mini.php	Шаблон страницы (шапка). Сокращенный вывод (без электронных журналов)

3.3.12. Обобщенная информация об основных модулях, реализованных на базе языка программирования python, приведена в таблице 2. Обобщенная информация о

вспомогательных модулях, реализованных на базе языка программирования python, приведена в таблице 3.

Таблица 2 – Основные модули

Идентификатор файла	Примечание
kmt_agent.py kmt_send_trap.py SONATAMIB.py	Модуль поддержки SNMP: - агент - отправка TRAP - описание MIB
kmt_monitor_udp_ovi.py	Модуль опроса плат
kmt_get_status.py	Модуль контроля состояния платы управления

Таблица 3 – Вспомогательные модули

Идентификатор файла	Примечание
kmt_db_sqlite_alter.py kmt_db_sqlite_create.py kmt_db_sqlite_create_first.py kmt_db_sqlite_from_pg.py kmt_db_sqlite_log_create.py kmt_db_sqlite_ovi_create.py	Модули для создания и измерения баз данных. Запускаются один раз в случае полной переустановки ПИ
kmt_id.py	Идентификационные данные
kmt_leds.py	Контроль работы остальных модулей, выполнение внутренних команд
kmt_search_sonata.py	Модуль поиска МП ТСС в локальной сети
kmt_trap_reciever.py	Модуль получения и регистрации во временной базе данных сообщений TRAP. Используется для отладки ПИ
kmt_zero.py	Сброс базы данных в исходное состояние

3.3.13. Все модули ПИ запускаются автоматически при загрузке операционной системы.

3.3.14. Все исполняемые модули пользуются сервисами ОС.

3.4. Связи программы с другими программами

3.4.1. Для функционирования модулей опроса блоков и поддержки протокола SNMP необходимо наличие установленного пакета rsysnmp.

3.4.2. Для функционирования Web-интерфейса администрирования необходимо наличие Web-сервера Apache с PHP.

3.5. Вариант развертывания ПИ

3.5.1. Вариант развертывания ПИ приведен на рис. 4.

3.5.2. ПИ устанавливается на одноплатный компьютер, входящий в состав платы управления МП ТСС, локальный терминал оператора (ЛТО) подключается к МП ТСС через стык Ethernet, на ЛТО установлено только системное обеспечение (операционная система, браузер и офисный пакет).

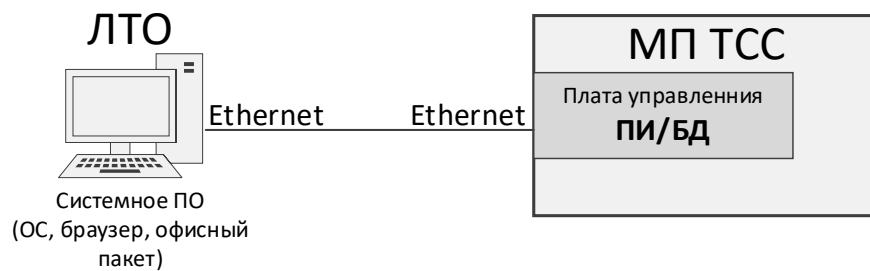


Рисунок 4

4. ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА

4.1. Для обеспечения функционирования ПИ используется одноплатный компьютер со следующими характеристиками

- четырехъядерный процессор с частотой 2 ГГц;
- объем ОЗУ 2 Гбайт;
- объем постоянной памяти eMMC 16 ГБ;
- наличие трех последовательных портов UART;
- наличие интерфейса Ethernet.

4.2. Архитектура компьютера обеспечивает возможность установки на него Linux-подобной операционной системы (например, Ubuntu).

4.3. Для взаимодействия с микроконтроллером используется один из трех последовательных портов UART.

5. ВЫЗОВ И ЗАГРУЗКА

5.1. Способ вызова программы

5.1.1. Вызов программы осуществляется автоматически при загрузке операционной системы.

5.1.2. В ходе выполнения программы осуществляется постоянный контроль работы модулей и автоматический перезапуск сбойного модуля.

5.2. Объем программы

5.2.1. Объем программы, включая базы данных, составляет около 25 Мбайт.

5.2.2. В ходе выполнения ПИ может использовать до 500 МБайт оперативной памяти.

6. ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ

6.1. Входные данные (команды) поступают:

- 1) от микроконтроллера, встроенного в плату управления МП ТСС, по последовательному порту (скорость обмена данными составляет 115 кбит/с);
- 2) от локального терминала через Ethernet;
- 3) от внешней системы управления с использованием протокола SNMP.

6.2. Входными данными для ПИ являются команды, которые формирует пользователь через Web-интерфейс. Допускается использование стандартного (текстового) терминала для передачи команд в оборудование. В качестве терминала может использоваться программа putty.

6.3. Выходные данные ПИ представляют собой результаты измерений и расчета параметров синхросигналов, отображаемые в табличном и графическом (графики) видах в реальном и отложенном времени.

6.4. На основе входных данных, полученных от пользователя, формируются команды для управления платами.

6.5. Формат данных для платы управления (между ПИ и оборудованием) представляет собой структурированный текст, включающий:

- 1) для выходных данных: команду с параметрами;
- 2) для выходных данных: ответ на команду.

6.6. Описание команд и ответов на них, передаваемых между ПИ и оборудованием, приведен в таблице 1.

Таблица 1

Блок	Команда	Описание, пример ответа
Блок управления	ls	Получить список установленных блоков. Возвращается перечень номеров установленных блоков (с 0 до 14), разделенных пробелами. Пример запроса: ls\r\n =1 5 6 7 8\r\n
Блок управления (не адресуется)	tie	Получить результаты измерений ОВИ для 14 каналов. Ожидаемый формат ответа:

Блок	Команда	Описание, пример ответа
		=ОВИ1 ОВИ2 ...\r\n Если измерения для какого-то канала не проводились, то в соответствующей позиции возвращается символ n. Возвращаются последние на момент запроса ОВИ. Если отсутствуют обе платы входов, то возвращается error. Пример запроса: tie\r\n =128 n 358 n n -453 1 2 3 4 5 6 7 8\r\n tie\r\n =error\r\n
Блоки входов (управление режимом «байпас»)	N bypass N bypass=on N bypass=off N bypass=auto	Управление режимом «байпас» для плат p7 и p8. bypass – получить текущее состояние bypass=on – включить режим «байпас» bypass=off – выключить режим «байпас» bypass=auto – включить автомат. «байпас» Пример запроса: p7 bypass\r\n =off\r\n p7 bypass=auto\r\n ok\r\n
Блок управления (не адресуется)	time time= YYYYMMDD D HHMMSS	Получить текущее время. Формат ответа: 1) если время получено: =YYYYMMDD HHMMSS 2) если время не было получено: =none Пример запроса: time\r\n =20230307 121001\r\n
Все блоки + Корзина (p15)	N info	Получить информацию по блоку с адресом N. Допустимые значения N приведены в таблице 1. Возвращается: заводской номер, версия ПО, дата ПО, версия ПЛИС, дата ПЛИС. Поля разделяются пробелом. Если ПО или ПЛИС нет, то в соответствующих позициях записывается знак минус (-) (код 0x2D). Если блок не установлен, то возвращается none. При запросе информации по адресу p15 предлагается возвращать информацию по корзине. Пример запроса: p5 info\r\n =13423 1.5.2 20230302 1.4.1 20230302\r\n p7 info\r\n =none\r\n
Все блоки, которые можно перезапустить, включить	N=u N=d N=r	Включение (u или up), выключение (d или down) и перезагрузка блока (r или reboot). Плата управления команды включения и отключения (насчет отключения необходимо

Блок	Команда	Описание, пример ответа
или выключить		<p>уточнить) не поддерживает. Пример запроса: p5=u\r\n ok\r\n p7=r\r\n ok\r\n</p>
Блок усилителей (адреса p0, p4)	p0 status p4 status	<p>Запросить статус блока. Возвращается состояние (активный (a), резервный (r)) и режимы выходов (32 шт.). Для каждого выхода возвращается: m – 2 МГц, e – 2 Мбит/с, 1 – 10 МГц, 0 – отключен. Если блока нет, то возвращается none. Если блок выключен – down. Пример запроса: p0 status\r\n =a mmmeemmm0000mmmmeemmm00m0m\r\n p0 status\r\n =none\r\n p4 status\r\n =r mmmeemmm0000mmmmeemmm00m0m\r\n</p>
Блок синтезаторов (адреса p5, p6)	p5 status p6 status	<p>Запросить статус блока. Возвращается состояние (активный (a), резервный (r)) и режим работы (p – вхождение в синхронизм, s – синхронный режим, h – режим удержания, f – свободные колебания). Если блока нет, то возвращается none. Если блок выключен – down. Пример запроса: p5 status\r\n =a s\r\n p6 status\r\n =r p\r\n</p>
Блок входов (адреса p7, p8)	p7 status p8 status	<p>Запросить статус блока. Возвращается состояние (активный (a), резервный (r)) и состояние светодиодов входов: 0 – нет сигнала 1 – зеленый 2 – ошибка 3 – оранжевый 4 – мигающий оранжевый 5 – мигающий красный) Если блока нет, то возвращается none. Светодиоды для входов 1PPS и ГНСС. Предлагается записывать последовательно: ГНСС_1 1PPS_1 ГНСС_2 1PPS_2 Итого будет 10 значений состояний светодиодов. Если блока нет, то возвращается none. Если блок выключен – down.</p>

Блок	Команда	Описание, пример ответа
		Пример запроса: p7 status\r\n =a 1 0 0 2 3 1 0 0 0 0\r\n p8 status\r\n =r 1 0 0 2 3 1 0 0 0 0\r\n
Блок питания (адреса p10, p11)	p10 status p11 status	Запросить статус блока. Возвращается состояние вводов питания: 1 – питание есть, 0 – питания нет. Информация по вводам записывается последовательно – сначала по вводу 1, затем по вводу 2. Если блока нет, то возвращается none. Если блок выключен – down. Пример запроса: p10 status\r\n =10\r\n p11 status\r\n =11\r\n
Работа с масками		
Блок входов (адреса p7, p8)	mtie tdev	Получить установленную маску для МОВИ (mtie) или ДВИ (tdev). Возвращается default, если выбрана маска по умолчанию, иначе идентификатор маски в СУ и значение маски для 10-ти контрольных точек: 0.1, 1.0, 2.5, 6.3, 10.0, 16.0, 25.0, 30.0, 1000.0, 10000.0. Пример запроса: mtie\r\n =1 25 25 25 30 30 50 80 100 200 400\r\n tdev\r\n =1 5 5 5 10 10 20 30 30 30 30\r\n mtie\r\n =default\r\n
Блок входов (адреса p7, p8)	mtie=... tdev=...	Установить маску для МОВИ (mtie) или ДВИ (tdev). Передается default для установки маски по умолчанию, или идентификатор маски в СУ и непосредственно значение маски для 10-ти контрольных точек: 0.1, 1.0, 2.5, 6.3, 10.0, 16.0, 25.0, 30.0, 1000.0, 10000.0. Для установки масок по умолчанию default передается либо командой mtie либо tdev. Пример запроса: mtie=3 25 25 25 30 30 50 80 100 200 400\r\n ok\r\n tdev=3 5 5 5 10 10 20 30 30 30 30\r\n ok\r\n mtie=default\r\n ok\r\n
Входы (адреса 1 – 14)	x PL xN xN=P	Получение (запрос отдельно по каждой плате) / установка правил обработки масок. N – номер входа (1 – 14)

Блок	Команда	Описание, пример ответа
		<p>P – вид обработки: 0 – обработка в автоматическом режиме 1 – не обрабатываем МОВИ 2 – не обрабатываем ДВИ 3 – не обрабатываем оба параметра</p> <p>Пример запроса: x p7\r\n =2 0 0 0 0 0 0 0 0 0 0 0 1 3\r\n x p8\r\n =0 2 0 0 0 0 0 0 0 0 0 0 1 3\r\n x1\r\n =2\r\n x3=3\r\n ok\r\n</p>
Управление выходами		
Выход (адреса с o1 по o32)	oN	<p>Получить режим работы выхода N. Возвращается одно из следующих значений: m – 2 МГц, e – 2 Мбит/с, 1 – 10 МГц, 0 – отключен.</p> <p>Пример запроса: o3\r\n =m\r\n o17\r\n =0\r\n</p>
Выход (адреса с o1 по o32)	oN=M	<p>Установить режим работы M для выхода N. Передается одно из следующих значений: m – 2 МГц, e – 2 Мбит/с, 1 – 10 МГц, 0 – (ноль) отключен.</p> <p>Пример запроса: o3=m\r\n ok\r\n o17=e\r\n ok\r\n</p>
Управление SSM		
SSM	s PL	<p>Получить данные по SSM на входах. Возвращается строка с данными SSM для 6 входов. Значение: ‘-’ – данных нет, остальные значения согласно нормативным документам (4, 8 и пр.).</p> <p>Пример запроса: s p7\r\n =0 4 8 8 0 -\r\n s p8\r\n =0 4 8 8 0 -\r\n</p>
SSM (адреса 1 – 6)	sN=C I S _i O S _o	<p>Установить данные по обработке SSM для входа N (от 1 до 6). Параметры: C – канал приема (0-5) I – режим обработки входного SSM (a – автоматический, r –</p>

Блок	Команда	Описание, пример ответа
		<p>ручной) S_i – замещающее значение SSM (для ручного режима) O – режим формирования выходного SSM (а – автоматический, r – ручной) S_o – значение выходного SSM (для ручного режима) Пример запроса: s1=4 a 8 r 4\r\n ok\r\n s5=3 a 4 r 8\r\n ok\r\n</p>
SSM (адреса 1 – 6)	sN	<p>Вернуть параметры по обработке SSM для входа N (от 1 до 6). Описание параметров см. выше. Пример запроса: s1\r\n =4 a 8 r 4\r\n</p>
Управление входами		
Входы	i	<p>Вернуть данные по всем входам (всего 14). Параметры возвращаемых данных: 1 – отключен 2 – в работе (выбран) 4 – в работе (не выбран) 5 – не проходит по МОВИ 6 – не проходит по ДВИ 7 – не проходит по МОВИ и ДВИ a – SSM не верен b – сигнала нет f – все остальные комбинации Если активных плат нет, то возвращается error Пример запроса: i\r\n =2 4 4 5 5 6 1 1 1 1 1 1 1 1\r\n</p>
Входы	m	<p>Вернуть данные по типам сигналов на входах: 0 – нет сигнала 1 – 2048 кГц (меандр 2048 кГц) 2 – 1 МГц (меандр 1000 кГц) 3 – 5 МГц (меандр 5000 кГц) 4 – 10 МГц (меандр 10 000 кГц) 5 – E1 (2048 кбит/с) Если активных плат нет, то возвращается error Пример запроса: m\r\n =1 5 5 0 0 0\r\n</p>
Входы (адреса 1 - 14)	iN	<p>Выбрать вход N (от 1 до 14). Если выбрать порт нельзя по каким-то причинам, то возвращается error Пример запроса:</p>

Блок	Команда	Описание, пример ответа
		i1\r\n ok\r\n i5\r\n error\r\n
Входы (адреса 1 - 6)	iN=on iN=off	Ввести в эксплуатацию / вывести из эксплуатации вход N (от 1 до 14). Номер входа задается Пример запроса: i1=off\r\n ok\r\n i11=on\r\n ok\r\n
Входы	i=1 2 3 4 5 6 7 8 9 10 11 12 13 14	Установить приоритеты. Задается для каждого из входов (всего 14 значений). Пример запроса: i=1 2 4 3 5 6 7 8 9 10 11 12 13 14\r\n ok\r\n
Входы	g	Получить приоритеты. Возвращается 14 значений. Пример запроса: g\r\n =1 2 4 3 5 6 7 8 9 10 11 12 13 14\r\n
Управление станционной сигнализацией		
Блок управления	alarm1=1 alarm1=0 alarm2=1 alarm2=0 alarm3=1 alarm3=0 alarm4=1 alarm4=0	Замкнуть/включить (0) или разомкнуть/выключить (1) соответствующие контакты на интерфейсе подключения к станционной сигнализации. Пример запроса: alarm1=1\r\n ok\r\n
Работа с приемниками ГНСС		
Получение информации по режиму работы приемника ГНСС	gnss gnss=gp N S gnss=gl N S gnss=gn N S gnss=off1 gnss=off2 gnss=on1 gnss=on2	Получить информацию по режиму работы приемников. gp – установить режим GPS gl – установить режим ГЛОНАСС gn – установить совместный режим ГЛОНАСС/GPS off1 – отключить прием сигналов первым приемником off2 – отключить прием сигналов вторым приемником on1 – включить прием сигналов первым приемником on2 – включить прием сигналов вторым приемником N – минимальное количество спутников для бракования сигнала S – минимальный уровень сигнала Возвращаемые значения: gp, gl, gn, off1, off2, заданное минимальное количество спутников, минимально допустимый уровень сигнала.

Блок	Команда	Описание, пример ответа
		<p>Если приемника нет, то возвращается none.</p> <p>Пример запроса:</p> <pre>gnss\r\n =gl gl 3 30\r\n gnss=gp 4 25\r\n ok\r\n gnss\r\n =gp gp 4 25\r\n gnss=off1\r\n ok\r\n gnss\r\n =off gp 4 25\r\n</pre>
Получить данные по времени / позиционированию	rmc1 rmc2	<p>Получить последний рекомендованный минимальный набор данных от первого/второго приемника ГНСС (фактически, это копия последнего сообщения \$GPRMC, \$GNRMC или \$GLRMC в зависимости от включенного режима работы приемника ГНСС). Если приемника нет, то возвращается none.</p> <p>Пример запроса:</p> <pre>rmc1\r\n =\$GPRMC,123519,A,4807.038,N,01131.000,E,022.4,084.4,230 394,003.1,W*6A\r\n rmc2\r\n =none\r\n</pre>
Получение информации по спутникам	gsv1 gsv2	<p>В ответ на запрос передается накопленная на момент запроса информация по спутникам (по которым есть уровень сигнала) либо none, если приемника нет или спутники не обнаружены. Формат сообщения информации по спутникам:</p> <pre>=num,elevation,azimuth,snr,... \r\n</pre> <p>где,</p> <ul style="list-style-type: none"> num – номер спутника elevation – угол возвышения, в градусах azimuth – азимут в градусах snr – уровень сигнала <p>Пример запроса:</p> <pre>gsv1 =01,15,176,45,03,71,143,77,06,29,312,45,09,27,239,64\r\n gsv2 =none</pre>
Работа с платами NTP и PTP		
Чтение / установка параметров	p13 rN p13 rN=Z p13 save p14 rN	<p>Чтение регистра N для плат NTP (p13 rN) и PTP (p14 rN).</p> <p>Установка регистра N в значение Z для плат NTP (p13 rN=Z) и PTP (p14 rN=Z).</p> <p>Запись регистров в планы производится после получения команды p13 save или p14 save.</p>

Блок	Команда	Описание, пример ответа
	p14 rN=Z p14 save	Номера и назначение регистров приведены в таблице 3. Пример запроса: p13 r1 =0f1517450371\r\n p13 r2=43523451627\r\n ok\r\n p13 save\r\n ok\r\n



Акционерное общество

Научно-производственное предприятие «КОМЕТЕХ»
196006, г. Санкт-Петербург, вн.тер.г. муниципальный округ
Московская застава, ул. Парковая, д. 4, литера А, ком. 405
+7 (812) 407-25-04, mail@kometeh.ru
www.kometeh.ru