

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования «Южно-Российский государственный
политехнический университет (НПИ) имени М.И. Платова»

На правах рукописи



Муженко Александр Сергеевич

**МЕТОДЫ И МОДЕЛИ ОЦЕНКИ ЭФФЕКТИВНОСТИ ПРОЦЕССОВ
МЕЖМОДУЛЬНОГО ВЗАИМОДЕЙСТВИЯ В СИСТЕМАХ СБОРА И
ОБРАБОТКИ ИНФОРМАЦИИ ЭЛЕКТРИЧЕСКИХ ПОДСТАНЦИЙ**

Специальность: 2.3.1. Системный анализ, управление
и обработка информации, статистика

ДИССЕРТАЦИЯ

на соискание ученой степени кандидата технических наук

Научный руководитель:
доктор технических наук
Плотников Дмитрий Александрович

Новочеркасск 2025 г.

Оглавление

| | |
|---|-----------|
| Введение..... | 5 |
| 1 Обзор и анализ существующих методов и технических средств контроля электрических подстанций..... | 15 |
| 1.1 Общая характеристика и классификация подстанций | 15 |
| 1.2 Анализ электрической подстанции, как объекта контроля и управления | 19 |
| 1.2.1 Основное технологическое оборудование подстанции | 19 |
| 1.2.2 Задачи мониторинга и управления оборудованием подстанции | 23 |
| 1.2.3 Типовой состав технологических функций АСУ ТП подстанции..... | 29 |
| 1.2.4 Типовые структуры и информационное взаимодействие в АСУ ТП подстанции..... | 31 |
| 1.3 Анализ требований к системам сбора и обработки информации в составе АСУ ТП подстанции | 37 |
| 1.3.1 Нормативные требования к контроллеру присоединения..... | 37 |
| 1.3.2 Оценка числа измерительных и управляющих каналов контроллера присоединения..... | 41 |
| 1.3.3 Анализ контроллера присоединения как системы сбора и обработки информации | 46 |
| 1.4 Постановка задач исследований | 48 |
| 1.4.1 Разработка номенклатуры модулей систем сбора и обработки информации электрических подстанций и выбор интерфейса межмодульного взаимодействия..... | 48 |
| 1.4.2 Разработка средств описания модульных систем сбора и обработки информации | 49 |
| 1.4.3 Создание методов и моделей для оценки параметров межмодульного информационного взаимодействия..... | 51 |
| 1.5 Выводы по главе 1 | 53 |
| 2 Разработка модульной структуры системы сбора и обработки информации электрической подстанции и метода её описания..... | 55 |

| | |
|---|-----------|
| 2.1 Обоснование и разработка номенклатуры модулей системы сбора и обработки информации | 55 |
| 2.2 Анализ информационных потоков системы..... | 56 |
| 2.3 Использование интерфейса <i>CAN FD</i> и протокола <i>CANopen FD</i> для организации межмодульного взаимодействия в системах сбора и обработки информации | 59 |
| 2.3.1 Обоснование выбора интерфейса <i>CAN FD</i> и протокола <i>CANopen FD</i> ... | 59 |
| 2.3.2 Общая характеристика протокола <i>CANopen FD</i> | 61 |
| 2.3.3 Анализ основных служб протокола <i>CANopen FD</i> | 64 |
| 2.3.4 Объектный словарь и средства описания устройств..... | 75 |
| 2.4 Представление межмодульных информационных связей и разработка метода описания системы..... | 76 |
| 2.5 Выводы по главе 2..... | 87 |
| 3 Разработка методов и моделей для оценки параметров межмодульного взаимодействия в системах сбора и обработки информации | 88 |
| 3.1 Разработка метода оценки времени передачи сообщений интерфейса <i>CAN FD</i> | 89 |
| 3.1.1 Анализ структуры сообщений | 89 |
| 3.1.2 Разработка метода оценки времени передачи сообщений..... | 91 |
| 3.2 Разработка усовершенствованного вероятностного метода и модели для оценки параметров межмодульного взаимодействия | 95 |
| 3.2.1 Анализ системы сбора и обработки информации как системы массового обслуживания | 95 |
| 3.2.2 Разработка усовершенствованного вероятностного метода оценки параметров | 99 |
| 3.2.3 Разработка усовершенствованной вероятностной модели системы сбора и обработки информации | 104 |
| 3.3 Разработка имитационной модели для оценки параметров межмодульного взаимодействия..... | 107 |
| 3.4 Оценка адекватности разработанных моделей | 113 |

| | |
|---|------------|
| 3.5 Выводы по главе 3..... | 121 |
| 4 Практическая реализация разработанных методов и моделей..... | 123 |
| 4.1 Использование усовершенствованной вероятностной модели для оценки параметров межмодульного взаимодействия в контроллере присоединения электрической подстанции 10-110 кВ..... | 123 |
| 4.2 Использование усовершенствованной вероятностной модели для оценки параметров взаимодействия устройств управления дугогасящими реакторами . | 129 |
| 4.3 Использование метода оценки времени передачи сообщений для исследования межмодульного взаимодействия в регистраторе событий | 135 |
| 4.4 Аппаратная реализация модулей систем сбора обработки информации подстанции | 136 |
| 4.5 Выводы по главе 4..... | 140 |
| Заключение | 141 |
| Список сокращений | 144 |
| Список используемой литературы | 146 |
| Приложение А. Структурные схемы информационных связей, компоновки элементов ПТК АСУ ТП ПС | 157 |
| Приложение Б. Исходные тексты основных модулей разработанного программного обеспечения | 165 |
| Б.1 Вероятностная модель для исследования параметров межмодульного взаимодействия в системе | 165 |
| Б.2 Имитационная модель для исследования параметров межмодульного взаимодействия в системе | 171 |
| Приложение В. Выборки результатов моделирования с использованием вероятностной и имитационной моделей | 174 |
| Приложение Г. Документы о внедрении результатов работы | 180 |
| Приложение Д. Свидетельства о регистрации программ для ЭВМ по теме исследования..... | 183 |

Введение

Актуальность темы. Модульные системы сбора, обработки информации и управления получают всё более широкое распространение в различных областях промышленности, медицины, обороны, энергетики и иных отраслей, открывая новые возможности для оптимизации и масштабирования производственных процессов. Этот подход позволяет предприятиям создавать адаптивные решения, которые легко изменяются в зависимости от производственных задач, технологических требований или изменений в рыночной ситуации.

Модульная система отличается от традиционных монолитных конструкций своей гибкостью. Каждый компонент представляет собой самостоятельный модуль, который может быть добавлен, заменён или модернизирован без необходимости полной перестройки системы.

Одним из примеров успешного внедрения модульных управляющих систем является автомобильная промышленность. Современные заводы используют модульные производственные линии на базе микропроцессорной техники для сборки автомобилей различных моделей на одной платформе. Это позволяет значительно сократить издержки и быстро адаптироваться к изменениям в спросе.

Другой пример – фармацевтическая отрасль, где модульные производственные системы используются для выпуска различных лекарственных форм на одной линии. Это особенно важно для компаний, работающих с небольшими партиями продукции или адаптирующихся к изменениям нормативных требований.

Также можно упомянуть системы управления морского и военно-морского флота РФ, в которых применяется интегрированная система управления платформой технических средств корабля. Это единый интегрированный цифровой вычислительно-управляющий комплекс на базе микропроцессорной техники, который обеспечивает контроль и управление главной энергетической установкой, электроэнергетической системой, общекорабельными системами, движением и маневрированием корабля. Аппаратные и программные средства

системы имеют модульную структуру для обеспечения максимальной эксплуатационной надёжности и функциональной гибкости.

Ярким примером области промышленности, в которой от параметров используемых модульных систем сбора и обработки информации зависит эффективность отрасли в целом, является электроэнергетика. Российская Федерация, по данным отчета Системного оператора о функционировании Единой Энергетической системы РФ за 2024 год, является одним из крупнейших мировых производителей электроэнергии. Вследствие существенного объёма производства, передачи и потребления электроэнергии особое значение приобретает обеспечение технико-экономической эффективности функционирования электрических подстанций (ПС), являющихся ключевым звеном в системе энергораспределения.

Электрические подстанции (ПС) – это связующие элементы между электростанциями и конечными потребителями. Объекты такого типа должны принимать электроэнергию от электростанций и распределять её с высокой надёжностью и эффективностью между потребителями. Для обеспечения надёжности и эффективности функционирования ПС используются автоматизированные системы управления технологическим процессом (АСУ ТП), одним из ключевых элементов которых является контроллер присоединения (КП).

Стандартом организации «Единая Энергетическая система РФ» (СТО 56947007-29.240.30.010-2008) определено более 60 типовых схем подстанций, различающихся сложностью организации, но содержащих однотипное оборудование: трансформаторы, выключатели, разъединители, заземляющие ножи, однако количество этих элементов, в зависимости от сложности схемы может существенно различаться. КП выполняет сбор и предварительную обработку информации, поступающей с оборудования ПС, а также формирует сигналы управления им, т.е. является микропроцессорной системой сбора и обработки информации (ССОИ) ПС.

Ввиду большого разнообразия возможных конфигураций ПС и, следовательно, разного количества измеряемых значений и нуждающихся в управлении узлов, возникают существенно различающиеся требования к

информационной ёмкости ССОИ ПС. Обеспечить необходимую информационную ёмкость можно различными способами, один из которых – применение модульной структуры. Однако, при использовании модульной структуры важно иметь средства оценки параметров межмодульного взаимодействия, чтобы обеспечить надежную и безотказную работу модульной системы. Следует отметить, что эта проблема актуальна не только для электроэнергетики, но и для любой другой области применения модульных систем.

В процессе анализа предметной области выяснилось, что существуют решения для узко направленных задач, но нет универсальных средств оценки параметров межмодульного взаимодействия в ССОИ. Поэтому обеспечение необходимой информационной ёмкости для каждой конкретной подстанции и разработка универсальных средств оценки параметров межмодульного взаимодействия в системах сбора и обработки информации является актуальной задачей, решение которой имеет существенное значение для развития электроэнергетической отрасли страны.

Исследования выполнялись в соответствии с Перечнем критических технологий Российской Федерации, утверждённым Указом Президента РФ от 7 июля 2011 г. № 899 (разделы «Технологии информационных, управляющих, навигационных систем», «Технологии создания энергосберегающих систем транспортировки, распределения и использования энергии»), в рамках научного направления «Теория, принципы, моделирование и технологии проектирования информационных, измерительных и управляющих систем», утвержденного Учёным советом ЮРГПУ(НПИ) 29 декабря 2021 года, протокол №5.

Степень разработанности темы диссертационного исследования.

Оценка эффективности процессов межмодульного взаимодействия в ССОИ на начальных этапах проектирования обычно выполняется с помощью математического и имитационного моделирования, в том числе с использованием стохастических моделей, теории массового обслуживания, цифровых двойников.

Вопросы математического моделирования процессов в ССОИ и оценки эффективности их функционирования подробно рассмотрены в работах

Аллакулиева Ю.Б., Боровкова А.И., Глазунова Ю.В., Масселя Л.В., Родионовой С.Н., Ротанова Е.Г., Скурихина В.А., Шаховского А.В. и других исследователей [1-5], однако, несмотря на разнообразие подходов к оценке эффективности ССОИ, многие исследования обладают общими недостатками, такими как весьма узкая область применения и недостаточность эмпирических данных, подтверждающих эффективность предложенных методов в смежных областях.

С общими методологическими подходами к моделированию информационных процессов в модульных системах можно ознакомиться в [6, 7, 8, 9, 10, 11], однако авторы [7, 8, 9] подчеркивают, что, несмотря на наличие унифицированных концептуальных основ, создание адекватной модели для конкретной системы представляет собой крайне сложную задачу, требующую учёта множества разнообразных аспектов. При создании модели необходимо учитывать особенности среды передачи данных, используемых протоколов, служб, также необходимо принимать во внимание специфику объекта, для которого разрабатывается модель.

Известны готовые программные средства моделирования взаимодействия [12, 13, 14, 15, 16] для широко распространённых универсальных технологий проводной (*Ethernet* [17]) и беспроводной (*GSM* [18], *UMTS* [19], *Wi-Fi* [20], *Bluetooth* [21]) связи. Для этих технологий существуют доступные и готовые к использованию библиотеки, содержащие описания стандартных протоколов и типовых устройств. Однако более специализированные современные промышленные интерфейсы межмодульного взаимодействия, такие, как *CAN* [22], *CAN FD* [23], *RS-485* [24], *SPI* [25], имеют свою специфику и требуют разработки отдельных библиотек для упомянутых средств моделирования, что по сложности сопоставимо с созданием новой модели.

Целью диссертационной работы является повышение эффективности функционирования систем сбора и обработки информации электроэнергетических подстанций (ССОИ ПС) за счёт улучшения их технических и эксплуатационных характеристик путём организации надёжного межмодульного взаимодействия в

ССОИ ПС с применением современных методов получения, анализа и обработки информации.

Для достижения поставленной цели в представленной работе решены **следующие задачи:**

– выполнен обзор и анализ публикаций, научных трудов и других источников по теме исследования;

– обоснована необходимость использования модульной структуры ССОИ в составе АСУ ТП ПС;

– обоснованы критерии эффективности процессов межмодульного взаимодействия в ССОИ ПС;

– обоснована целесообразность использования современного интерфейса *CAN FD* и протокола *CANopen FD* для организации межмодульного взаимодействия в ССОИ;

– разработан метод описания модульной ССОИ, использующей протокол *CANopen FD* и интерфейс *CAN FD*, с учетом межмодульных информационных связей;

– разработан метод оценки времени передачи сообщений в ССОИ, использующих интерфейс *CAN FD*, позволяющий определить верхнюю и нижнюю границы времени передачи;

– разработаны усовершенствованные вероятностные метод и модель для предварительной оценки параметров межмодульного взаимодействия в ССОИ, использующих протокол *CANopen FD*;

– разработана имитационная модель для детального исследования процессов межмодульного взаимодействия в ССОИ, использующих протокол *CANopen FD*;

– разработано программное обеспечение, реализующее вероятностную и имитационную модели и позволяющее исследовать параметры межмодульного взаимодействия, оценить степень их влияния на качество связи, а также устройства, использующие предложенные методы.

Объектом исследования являются модульные системы сбора и обработки информации электрических подстанций (ССОИ ПС).

Предметом исследования является межмодульное взаимодействие в ССОИ ПС, методы его описания и оценки его параметров.

Методы исследования. Методология исследования базируется на комплексном применении теоретического анализа, компьютерного моделирования и физического эксперимента. Для решения задач исследования используются методы математического и системного анализа, статистики, обобщённого и объектно-ориентированного программирования. Для практической реализации результатов работы использованы программные продукты *Microsoft Visual Studio*, *Matlab*, язык программирования *C#*, язык разметки и инициализации объектов в топологиях платформы *.NET XAML*, расширяемый язык разметки *XML* и язык описания структуры документа *XSD*.

Достоверность и обоснованность научных положений и выводов, сформулированных в диссертационной работе, подтверждается обоснованностью принятых допущений, корректным применением методов математического и системного анализа, теории алгоритмов и программ; корректным применением математического аппарата при получении аналитических выражений, подтверждением теоретических положений результатами имитационного компьютерного моделирования и физических экспериментов.

Тема работы соответствует технической отрасли науки паспорта научной специальности 2.3.1. Системный анализ, управление и обработка информации, статистика по направлениям исследований п. 3 «Разработка критериев и моделей описания и оценки эффективности решения задач системного анализа, оптимизации, управления, принятия решений, обработки информации и искусственного интеллекта», п. 4 «Разработка методов и алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений, обработки информации и искусственного интеллекта», п. 11 «Методы и алгоритмы прогнозирования и оценки эффективности, качества, надежности функционирования сложных систем управления и их элементов»,

п. 12 «Визуализация, трансформация и анализ информации на основе компьютерных методов обработки информации» и п.15 «Теоретический анализ и экспериментальное исследование функционирования элементов систем управления в нормальных и специальных условиях с целью улучшения технико-экономических и эксплуатационных характеристик».

Научная новизна работы состоит в следующем.

1. Разработано формализованное описание модульной ССОИ на базе интерфейса *CAN FD*, отличающееся явным представлением межмодульных информационных связей и предотвращающее возможность возникновения основных ошибок конфигурирования системы (соответствует направлению исследования п.3 паспорта специальности 2.3.1, страницы 76–86 диссертации).

2. Разработан метод оценки времени передачи сообщений *CAN FD*, формируемых в процессе функционирования модулей ССОИ, отличающийся тем, что он позволяет определить верхнюю, нижнюю и среднюю оценки времени передачи с учетом структуры сообщений, и обеспечивающий более высокую точность оценки по сравнению с известными методами (соответствует направлению исследования п.4 паспорта специальности 2.3.1, страницы 89–95 диссертации).

3. Разработаны усовершенствованные вероятностные метод и модель для предварительной оценки параметров межмодульного взаимодействия в ССОИ, построенных на базе интерфейса *CAN FD* и протокола *CANopen FD*, отличающиеся использованием информации о вероятности изменения контролируемых сигналов и позволяющие оперативно оценить зависимость критически важных параметров межмодульного взаимодействия от конфигурации конкретной системы (соответствует направлениям исследования п. 11 и п.15 паспорта специальности 2.3.1, страницы 95–107 диссертации).

4. Разработана имитационная модель для детального исследования процессов межмодульного взаимодействия в ССОИ, построенных на базе интерфейса *CAN FD* и протокола *CANopen FD*, отличающаяся тем, что на основе обобщенного описания системы формирует параметры конфигурации входящих в

неё модулей и путём имитационного моделирования позволяет с высокой точностью оценить основные параметры процессов межмодульного взаимодействия (соответствует направлениям исследования п.12 и п.15 паспорта специальности 2.3.1, страницы 107–113 диссертации).

5. Показана универсальность разработанных методов и моделей, обеспечивающая возможность их использования для конфигурирования и оценки параметров не только ССОИ ПС, но и других модульных систем (страницы 129–136 диссертации).

На защиту выносятся:

1. метод описания модульной ССОИ, использующей протокол *CANopen FD* и интерфейс *CAN FD*, с учетом межмодульных информационных связей;

2. метод оценки времени передачи сообщений *CAN FD* для моделирования межмодульного взаимодействия в ССОИ, позволяющий оценить верхнюю и нижнюю границы времени передачи с учетом длины сообщения и возможной вставки бит;

3. усовершенствованные вероятностные метод и модель для предварительной оценки параметров межмодульного взаимодействия в ССОИ, позволяющие на ранних этапах проектирования модульных систем оперативно оценить критически важные параметры межмодульного взаимодействия, определяющие эффективность работы этих систем;

4. имитационная модель для детального исследования межмодульного взаимодействия в ССОИ на базе интерфейса *CAN FD* и протокола *CANopen FD*, позволяющая с учётом параметров настройки каждого модуля системы оценить параметры межмодульного взаимодействия с целью выявления перегрузки сети и недопустимых задержек доставки сообщений, влияющих на эффективность функционирования системы;

5. устройства и программное обеспечение, реализующие разработанные методы и модели.

Теоретическая значимость диссертационной работы заключается в разработке методов описания модульных систем и оценки эффективности процессов межмодульного взаимодействия в ССОИ.

Практическая значимость работы заключается в применении разработанных методов, моделей и программного обеспечения при проектировании современных ССОИ ПС, эксплуатируемых в энергосетях Российской Федерации. Перечисленные средства позволяют существенно повысить эффективность, надёжность и качество ССОИ ПС путём уменьшения сроков проектирования, исключения ошибок при формировании внутрисистемных информационных связей и, как следствие, повысить безопасность и экономическую эффективность эксплуатации ПС.

Реализация результатов работы. Разработанные в процессе подготовки диссертации методы и модели использованы в ООО «СКБ Электронного приборостроения» (г. Черноголовка) при проектировании универсального модульного контроллера присоединения, находящегося в стадии подготовки серийного производства.

Разработанные вероятностная и имитационная модели внедрены в учебный процесс ЮРГПУ (НПИ) и используются в курсах «Проектирование цифровых вычислительных (управляющих) устройств», «Микроконтроллеры и микропроцессоры», «Отладочные средства микропроцессорных систем».

Апробация работы. Основные положения и результаты диссертационной работы докладывались и обсуждались на конференциях: Международная научно-техническая конференция молодых учёных БТГУ им. В.Г. Шухова, г. Белгород, 2023 г.; 19-я международная научно-практическая конференция «Безопасность ядерной энергетики» на базе ВИТИ НИЯУ МИФИ, г. Волгодонск, 2023 г.; 20-я международная научно-практическая конференция «Безопасность ядерной энергетики» на базе ВИТИ НИЯУ МИФИ, г. Волгодонск, 2024 г.; 46-я международная научно-техническая конференция «Кибернетика энергетических систем» на базе ЮРГПУ (НПИ), г. Новочеркасск, 2024 г.

Публикации по теме диссертации. По результатам исследований опубликовано 13 печатных работ, из них 5 – в журналах, рекомендованных ВАК, получено 2 свидетельства о регистрации программного обеспечения.

Структура и объём работы. Диссертационная работа состоит из введения, четырёх глав, заключения и приложений. Она содержит 156 страниц основного текста, 46 рисунков, 9 таблиц, список литературы из 95 наименований и 5 приложений на 29 страницах.

1 ОБЗОР И АНАЛИЗ СУЩЕСТВУЮЩИХ МЕТОДОВ И ТЕХНИЧЕСКИХ СРЕДСТВ КОНТРОЛЯ ЭЛЕКТРИЧЕСКИХ ПОДСТАНЦИЙ

1.1 Общая характеристика и классификация подстанций

Электрическая подстанция (ПС) [26] представляет собой сложный технический объект, предназначенный для приема, преобразования и распределения электрической энергии. Согласно ГОСТ 19431 [27], ПС включают в себя преобразователи электрической энергии, а также системы управления, распределительные и вспомогательные устройства.

Классификация электрических подстанций

Электрические подстанции классифицируются по ряду признаков, определяющих их технические характеристики, назначение и конфигурацию в сети электроснабжения [28]:

- по уровню высокого напряжения, на который рассчитана ПС;
- по количеству силовых трансформаторов;
- по принадлежности и функциональной роли:

1) районные ПС – предназначены для электроснабжения отдельных административных или промышленных районов;

2) системные ПС – узловые элементы энергосистемы, выполняющие функции преобразования напряжения и распределения потоков мощности между различными участками сети;

3) потребительские ПС – обеспечивают электроснабжение конкретных предприятий или объектов;

- по конструктивному исполнению:

1) открытые распределительные устройства (ОРУ) – размещаются на открытом воздухе, обеспечивая удобный доступ к оборудованию, но требуют значительных площадей для своего размещения;

2) закрытые распределительные устройства (ЗРУ) – расположены в зданиях, что повышает уровень защиты от атмосферных воздействий, но увеличивает стоимость сооружения;

3) комплектные распределительные устройства (КРУ) – компактные решения в металлических шкафах, обеспечивающие минимизацию занимаемой площади и упрощение монтажа;

– по положению в энергосистеме (рис. 1.1):

1) тупиковые ПС – подключаются одной или двумя радиально подключенными линиями электропередачи (ЛЭП), не питающими другие объекты;

2) ответвительные ПС – подключаются к одной или двум магистральным ЛЭП посредством ответвлений; основная линия продолжает питание других подстанций;

3) проходные ПС – интегрируются в сеть путем «врезки» в две линии электропередачи с двусторонним питанием;

4) узловые ПС – формируют узел электроснабжения, соединяя три и более линий, обеспечивая перераспределение потоков мощности.

Компоновка трансформаторного оборудования

На большинстве электрических подстанций устанавливаются два силовых трансформатора, что позволяет повысить надежность энергоснабжения потребителей и обеспечить гибкость эксплуатации. Однотрансформаторные ПС могут применяться для питания объектов с пониженной степенью надежности или использоваться в качестве первой очереди при строительстве двухтрансформаторной подстанции с возможностью последующего расширения.

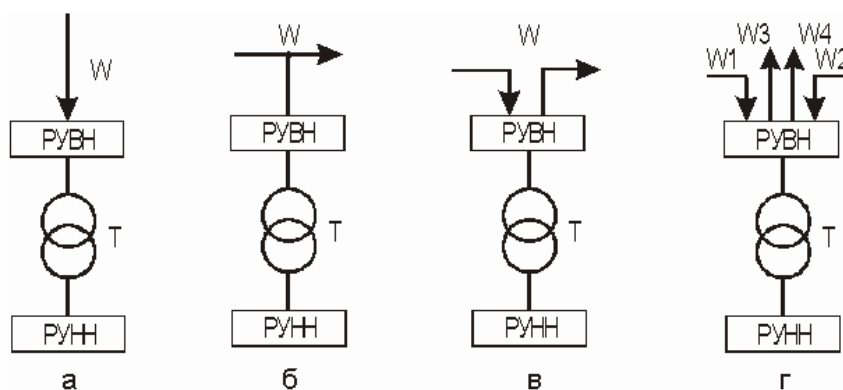


Рисунок 1.1 – Типы подстанций в зависимости от положения в энергосистеме:
а – тупиковая; б – ответвительная; в – проходная; г – узловая (РУВН – распределительное устройство высокого напряжения; РУНН – распределительное устройство низкого напряжения)

Далее будут рассмотрены факторы, определяющие выбор структурной схемы электрической подстанции. Они представляют собой комплекс технических и эксплуатационных параметров, которые необходимо учитывать при проектировании и модернизации ПС [29].

1. Значение и функциональная роль ПС в энергосистеме.

Структурная схема ПС в значительной степени определяется её назначением в рамках энергосистемы. ПС могут выполнять различные функции, включая электроснабжение промышленных предприятий или административных районов, обеспечение связи между различными частями энергосистемы.

2. Положение ПС в энергосистеме, а также топология и уровни напряжения прилегающих сетей.

ПС могут располагаться в узловых точках энергосистемы, где обеспечивается перераспределение потоков электрической мощности между различными объектами сети. В таких случаях схемные решения должны учитывать необходимость сохранения транзитных потоков мощности, что особенно важно для магистральных узлов электроснабжения.

3. Категория потребителей по степени надежности электроснабжения.

Важнейшим критерием при проектировании ПС является уровень надежности электроснабжения, который определяется категорией подключенных электроприемников.

Электроприемники I категории – объекты, отключение которых может привести к возникновению угрозы жизни людей, значительным экономическим потерям или нарушению сложных технологических процессов. Данная категория требует двух независимых источников питания, а приоритетные объекты дополнительно снабжаются третьим резервным источником с автоматическим вводом резерва.

Электроприемники II категории – объекты, нарушение электроснабжения которых приводит к массовым экономическим потерям, включая снижение объемов производства, простои промышленного оборудования и транспорта. Для

них рекомендуется двухстороннее резервирование с возможностью оперативного переключения дежурным персоналом или выездной бригадой.

Электроприемники III категории – прочие потребители, не подпадающие под I и II категории. Для них допустимо электроснабжение от одного источника, при условии, что перерывы на ремонт и замену поврежденных элементов длятся не более 24 часов.

4. Перспективы развития ПС и прилегающих сетей.

При проектировании ПС необходимо учитывать возможные перспективы её расширения, а также прогнозируемые изменения в энергосистеме. Развитие инфраструктуры может включать увеличение мощности за счет установки дополнительных силовых трансформаторов, изменение конфигурации сети в связи с ростом энергопотребления, внедрение современных технологий, включая цифровые ПС и интеллектуальные системы управления энергосетями.

С учетом этих факторов выбирается структурная схема ПС, обеспечивающая не только текущие, но и будущие потребности энергосистемы (рис. 1.2) [29].

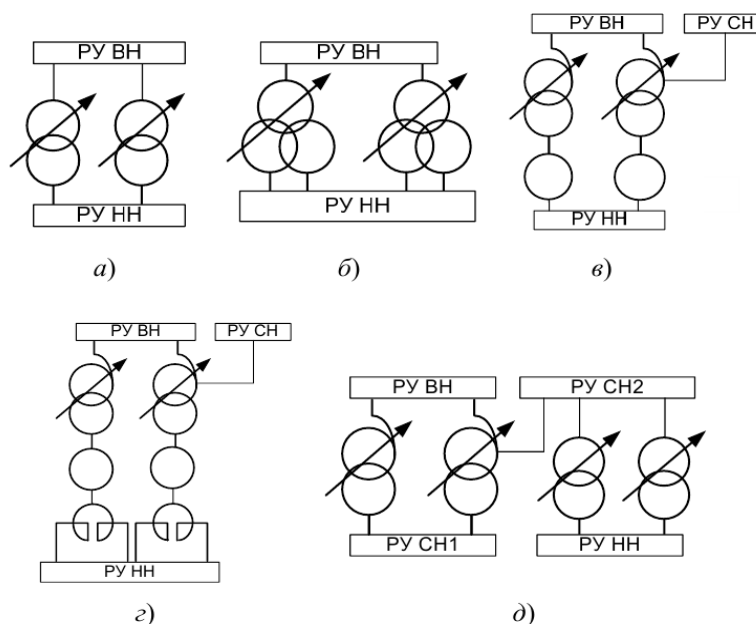


Рисунок 1.2 – Типовые структурные схемы ПС:

а – схема с двухобмоточными трансформаторами; *б* – схема с трансформаторами с расщеплённой обмоткой; *в* – схема с автотрансформаторами и регулировочными трансформаторами; *г* – схема с автотрансформаторами, регулировочными трансформаторами и реакторами; *д* – схема с автотрансформаторами и четырьмя РУ

1.2 Анализ электрической подстанции, как объекта контроля и управления

1.2.1 Основное технологическое оборудование подстанции

При проектировании ПС используется определенный состав основного оборудования. В состав основного оборудования любой подстанции входят: силовой трансформатор, шины, силовые коммутационные аппараты, системы измерения, управления, защит, автоматики, сигнализации [30].

Энергетическое оборудование электрических ПС функционально и конструктивно подразделяется на две основные категории:

- силовые устройства, обеспечивающие передачу и распределение электрической энергии по основным цепям электроснабжения;
- вторичные устройства, выполняющие функции сбора информации, защиты, управления и автоматизации технологических процессов в первичных цепях.

Силовое оборудование ПС может размещаться как на ОРУ, так и в ЗРУ, в зависимости от конструктивных и эксплуатационных требований. Вторичные системы, в свою очередь, монтируются на релейных панелях, в специализированных шкафах управления, а также в отдельных ячейках КРУ.

Рассмотрим более подробно основные разновидности оборудования ПС, имеющего отношение к тематике исследований, его особенности и характеристики.

Коммутационные аппараты

Основными типами коммутационных аппаратов являются выключатели, разъединители, отделители.

Высоковольтный выключатель – коммутационный аппарат, предназначенный для оперативного включения и отключения отдельных цепей или электрооборудования в энергосистеме в нормальных или аварийных режимах с обязательным гашением электрической дуги. По методу гашения дуги высоковольтные выключатели классифицируются на вакуумные (используются в электрических сетях напряжением 3-35 кВ), электромагнитные (6 и 10 кВ),

воздушные (свыше 110 кВ), масляные (до 220 кВ), элегазовые. Также выключатели разделяют по виду изоляции токоведущих частей (внутренние и наружные).

Обязательными требованиями к высоковольтным выключателям являются:

- безопасность;
- высокая скорость работы;
- надёжная эксплуатация;
- простота обслуживания.

Разъединитель представляет собой коммутационное устройство (аппарат), предназначенное для разрыва электрической цепи с обязательной визуальной индикацией разомкнутого состояния. В высоковольтных электрических схемах разъединители устанавливаются последовательно с выключателями, обеспечивая дополнительную безопасность при проведении ремонтных и эксплуатационных работ. Конструктивная простота разъединителей по сравнению с силовыми выключателями объясняется их функциональной спецификой – отсутствие необходимости в гашении электрической дуги при размыкании контактов.

Разъединители могут быть классифицированы по следующим признакам: по типу подвижного контактного ножа (поворотные рубящие, качающиеся); по роду установки (внутренние и наружные); по способу управления (ручные, моторизованные, дистанционные); по номинальному напряжению (до 35 кВ, 110-220 кВ и 330-500 кВ).

К разъединителям предъявляются следующие требования:

- обеспечение очевидного разрыва цепи (визуально контролируемого разъединения);
- стойкость к динамическим и термическим нагрузкам, возникающим при коммутации;
- наличие изоляции, гарантирующей надёжную работу при различных климатических воздействиях (дождь, снег, налесь);
- работоспособность в неблагоприятных эксплуатационных условиях;
- эргономичность конструкции, облегчающая техническое обслуживание.

Отделитель представляет из себя коммутационный аппарат, предназначенный для изолирования аварийного участка сети от основного электроснабжения. В сетях напряжением до 110 кВ используются трехполюсные устройства, а в высоковольтных сетях (220 кВ и выше) применяются однополюсные отделители (по одному на фазу). Для повышения коммутационной способности (возможности отключения токов до 630 А) применяются элегазовые отделители. По методу управления отделители выполняются с автоматическим отключением и ручным включением.

В штатных режимах эксплуатации электротехническое оборудование ПС не представляет непосредственной опасности для персонала, так как его металлические корпуса изолированы от токоведущих частей. Однако при возникновении аварийных ситуаций (повреждение изоляции, замыкание на землю) существует риск возникновения потенциала на металлических частях корпуса, что может привести к поражению электрическим током.

В целях обеспечения электробезопасности персонала все металлические элементы электроустановок, включая корпуса оборудования и конструкции, находящиеся в непосредственной близости, подлежат **заземлению**. Заземляющее устройство соединяет металлические части электрооборудования с контуром заземления ПС, обеспечивая отвод опасного потенциала и значительно уменьшая вероятность возникновения опасности для жизни человека.

Одним из устройств, предназначенных для создания надёжного и видимого соединения токоведущих частей электроустановки с землёй является **заземляющий нож (ЗН)**. Конструктивно ЗН представляет собой металлический стержень или пластину с контактами, которые при включении плотно прижимаются к токоведущим частям и заземляющему устройству. К ЗН предъявляются следующие требования: стационарность (при напряжениях до 1000 В могут использоваться переносные ЗН, свыше 1000 В должны применяться стационарные ЗН), наличие указателей отключенного и включенного положения (указатели должны быть установлены на приводах ЗН).

Силовые трансформаторы

При транспортировке электрической энергии на значительные расстояния неизбежно возникают энергетические потери, связанные с сопротивлением проводников и другими факторами. Для минимизации этих потерь в энергосистемах применяется принцип трансформации напряжения, а весь процесс передачи электроэнергии включает несколько ключевых этапов:

- 1) генерация электроэнергии на электростанциях, где напряжение, создаваемое генераторами, изначально имеет относительно низкую амплитуду;
- 2) повышение напряжения на трансформаторных ПС, обеспечивающее снижение потерь при транспортировке электроэнергии по ЛЭП;
- 3) транспортировка энергии по высоковольтным ЛЭП, соединяющим различные узлы энергосистемы;
- 4) понижение напряжения на трансформаторных ПС, где электроэнергия адаптируется под параметры, требуемые конечными потребителями.

Ключевыми элементами, обеспечивающими процесс трансформации электроэнергии (этапы 2 и 4), являются **силовые трансформаторы** и **автотрансформаторы**. Силовые трансформаторы выполняют преобразование напряжения за счет электромагнитной индукции между первичной и вторичной обмотками, а автотрансформаторы имеют общую обмотку для входного и выходного напряжений, что позволяет им обеспечивать более плавное регулирование параметров сети и повышенную энергетическую эффективность.

Использование данных устройств на подстанциях различного уровня напряжения обеспечивает эффективную и надежную передачу электрической энергии, снижает потери и стабилизирует параметры сети, что критически важно для устойчивой работы всей энергосистемы.

При вводе в эксплуатацию и работе силовых устройств ПС выполняется измерение и вычисление следующих основных параметров [31]:

- номинальная полная мощность ($S_{ном}$, кВА);
- номинальное линейное напряжение ($U_{л.ном}$, В или кВ);
- номинальный линейный ток ($I_{л.ном}$, А);

- номинальная частота (f , Гц);
- напряжение короткого замыкания (U_k , В);
- уровень, давление и температура диэлектрического и теплоотводящего масла.

Измерительные трансформаторы

Измерительные трансформаторы используются для информационного обмена между силовыми цепями и системами мониторинга, защиты и управления. Их функциональное назначение определяется необходимостью точного воспроизведения электрических процессов, протекающих в первичной силовой сети, с последующей передачей преобразованных параметров во вторичную сеть.

По принципу работы и конструктивным особенностям измерительные трансформаторы подразделяются на два основных типа:

- **трансформаторы тока (ТТ)** – предназначены для пропорционального преобразования величины электрического тока, проходящего через первичную обмотку, в сигнал меньшей величины, удобный для использования в измерительных приборах, устройствах релейной защиты и автоматики;

- **трансформаторы напряжения (ТН)** – обеспечивают понижение первичного напряжения до безопасных и стандартизованных значений, используемых в измерительных цепях и системах релейной защиты.

Несмотря на различие в назначении, оба типа устройств основаны на общих физических принципах электромагнитной трансформации, но при этом обладают специфическими конструктивными решениями и схемами включения в энергетическую инфраструктуру подстанции. Их корректное функционирование играет важнейшую роль в обеспечении точности измерений, надежности защиты и стабильности процессов управления энергосистемой.

1.2.2 Задачи мониторинга и управления оборудованием подстанции

Мониторинг состояния оборудования ПС является критически важным по следующим причинам.

1. Обеспечение надежности электроснабжения. Трансформаторные и распределительные ПС являются ключевыми элементами в электрических сетях. Контроль их состояния помогает предотвратить аварийные ситуации и обеспечивать стабильное и непрерывное электроснабжение потребителей.

2. Безопасность эксплуатации. Ненадлежащее состояние оборудования может привести к авариям и пожарам, которые могут вызвать серьезные последствия для персонала и окружающей инфраструктуры. Регулярный контроль помогает вовремя выявлять и устранять потенциальные опасности.

3. Продление срока службы оборудования. Регулярный мониторинг состояния и профилактическое обслуживание могут значительно продлить срок службы трансформаторов и других компонентов, что в долгосрочной перспективе снижает капитальные затраты на замену оборудования.

4. Экономическая эффективность. Профилактическое обслуживание и своевременное выявление неисправностей могут значительно снизить затраты на ремонт и замену оборудования.

5. Снижение потерь энергии. Неисправное или изношенное оборудование может приводить к значительным потерям энергии в процессе трансформации, передачи и распределения. Регулярный контроль помогает поддерживать эффективность работы ПС на высоком уровне.

6. Соответствие нормативным требованиям. В Российской Федерации, как и большинстве других стран, существует ряд нормативных актов и стандартов, регулирующих эксплуатацию и техническое обслуживание ПС. Контроль состояния помогает обеспечить соответствие этим требованиям и избежать штрафов и санкций.

7. Защита окружающей среды. Утечки масла и других опасных веществ из трансформаторов могут наносить вред окружающей среде. Регулярные проверки позволяют своевременно выявлять и устранять такие проблемы, минимизируя экологические риски.

8. Управление рисками. Операторы ПС могут лучше управлять рисками, связанными с эксплуатацией оборудования, если они владеют актуальной

информацией о его состоянии. Это позволяет принимать обоснованные решения о необходимости ремонта или модернизации.

Контроль состояния ПС включает в себя различные методы диагностики, такие как тепловизионное обследование, анализ вибраций, мониторинг уровня масла и его качества, тестирование электрических параметров и другие процедуры, которые помогают обеспечивать надежность, безопасность электросетевого комплекса и зачастую способствуют предотвращению аварийных ситуаций.

Аварийные ситуации – внезапное и непредвиденное событие, которое приводит к нарушению нормального функционирования оборудования или системы, создавая угрозу для безопасности людей, имущества или окружающей среды. На электрических ПС аварийные ситуации могут включать различные типы событий, от технических неисправностей до природных катастроф.

Далее приведем классификацию источников возникновения аварийных ситуаций.

1. Технические неисправности и износ оборудования

– Старение и деградация материалов. Со временем изоляционные материалы, масло, контакты и другие компоненты теряют свои свойства, что приводит к снижению надежности и повышенному риску аварий.

– Дефекты производства. Нарушения в процессе производства или сборки трансформаторов могут стать причиной преждевременных отказов.

– Ненадлежащее техническое обслуживание. Отсутствие регулярного и качественного обслуживания может привести к накоплению мелких неисправностей, которые в итоге вызывают серьезные аварии.

2. Эксплуатационные ошибки

– Ошибки персонала. Неправильное управление, ошибки при переключении оборудования или нарушение процедур безопасности могут привести к аварийным ситуациям.

– Неправильная настройка и регулировка. Неверная настройка защитных устройств и релейной защиты может привести к отсутствию их срабатывания в аварийной ситуации или, наоборот, вызвать ложные срабатывания.

3. Воздействие окружающей среды

– Экстремальные погодные условия. Грозы, наводнения, ураганы и другие экстремальные погодные условия могут повредить оборудование или вызвать короткие замыкания.

– Загрязнения и коррозия. Пыль, грязь, влага и агрессивные химические вещества могут способствовать коррозии и износу оборудования.

– Температурные колебания. Резкие изменения температуры могут вызывать механические напряжения в материалах и приводить к их повреждению.

4. Электрические перегрузки и замыкания

– Перегрузка оборудования. Превышение номинальных нагрузок может привести к перегреву и повреждению трансформаторов и других компонентов.

– Короткие замыкания. Замыкания в электрических цепях могут вызвать значительные токи, которые повреждают оборудование и вызывают аварии.

5. Механические повреждения

– Вандализм и кражи. Умышленное повреждение оборудования или кража металлических частей может нарушить нормальную работу подстанции.

– Аварии и катастрофы. Падение деревьев, дорожно-транспортные происшествия или другие несчастные случаи могут механически повредить оборудование подстанции.

6. Недостатки в проектировании и строительстве

– Проектные ошибки. Недостатки в проектировании подстанции могут привести к недостаточной надежности и безопасности ее работы.

– Нарушения при строительстве. Ошибки или нарушения при строительстве и монтаже оборудования могут стать причиной последующих аварий.

7. Проблемы с изоляцией

– Пробой изоляции. Повреждение или деградация изоляции может привести к коротким замыканиям и другим аварийным ситуациям.

– Увлажнение изоляции. Попадание влаги в изоляцию может снизить ее диэлектрические свойства и вызвать повреждения.

8. Проблемы с охлаждением

– Недостаточное охлаждение. Неисправности в системе охлаждения могут привести к перегреву трансформаторов и других компонентов, что может вызвать их повреждение.

– Загрязнение охлаждающей жидкости. Загрязнения в охлаждающем масле или других жидкостях могут снизить эффективность охлаждения и вызвать перегрев.

Одним из примеров аварии на электрической ПС в Российской Федерации является авария на ПС "Московская" в Московской области 20 августа 2020 года.

Авария произошла из-за возгорания на ПС "Московская" компании "Мосэнерго". По предварительным данным, причиной возгорания стала эксплуатация неисправного электрооборудования. В результате аварии был выведен из работы один из трансформаторов ПС, что привело к частичному отключению электропитания в некоторых районах Московской области. Более пятидесяти тысяч человек временно остались без электричества, а предприятия и организация остановили свою работу, что повлекло за собой существенные экономические потери.

Этот инцидент подчеркивает важность регулярного обслуживания и проверки электрооборудования на подстанциях, а также необходимость строгого соблюдения мер безопасности для предотвращения подобных аварийных ситуаций.

Из сказанного следует, что систематический и непрерывный контроль состояния ПС позволяет определить исправность оборудования и, как следствие, предотвратить возникновение аварийных ситуаций.

Непрерывный контроль и управление состоянием оборудования ПС обеспечивают АСУ ТП ПС, к которым, согласно [32], предъявляются следующие требования.

1. Общие требования:

– обеспечение максимальной эффективности решения производственных задач по обеспечению транспортировки электроэнергии в электросетевом комплексе;

– повышение надежности за счет снижения ошибок персонала;

– переход к использованию подстанций без постоянного присутствия обслуживающего персонала.

2. Типовой состав задач АСУ ТП:

– управление (технологическое или диспетчерское), в том числе обеспечение возможности дистанционного управления;

– информационная поддержка и контроль систем релейной защиты и автоматики (РЗА) и других специализированных систем автоматического управления / регулирования;

– регистрация параметров переходных процессов в аварийных режимах;

– мониторинг состояния основного технологического оборудования;

– автоматизация вспомогательных технологических процессов;

– обеспечение информационной и общей безопасности.

3. Задачи управления (технологического или диспетчерского):

– измерение параметров технологических (в т.ч. электроэнергетических) режимов работы оборудования ПС и отходящих ЛЭП, параметров окружающей среды на ПС в темпе реального времени, расчет необходимых для отображения и выдачи агрегированных и/или производных значений (среднее, интегральное, производной и т.п.), ведение архивов данных;

– контроль положения коммутационного оборудования, контроль нарушения заданной схемы работы подстанции (топологии), формирование, передача и регистрация (запись) сигналов о переключениях и их характеристиках (время, величина коммутируемого тока, значение напряжения и т.д.);

– дистанционное управление из АРМ ОП, ЦУС, ДЦ АО «СО ЕЭС» коммутационными аппаратами, заземляющими ножами, разъединителями, функциями устройств РЗА, управляемыми средствами регулирования

технологического режима оборудования ПС (положение РПН, регулируемые СКРМ, фазоповоротные устройства, устройства продольной компенсации, вставки постоянного тока и пр.), контроль формирования команд ДУ и блокировок, контроль выполнения команд ДУ, регистрация (запись) команд ДУ и результатов их выполнения;

– автоматизированное формирование и исполнение оптимальных карт проведения операций (автоматизированных бланков переключений, типовых автоматизированных бланков переключений);

– автоматический контроль проведения переключений, защита от неправильных действий персонала;

– автоматизированное выполнение переключений.

4. Задачи мониторинга состояния и эксплуатации основного технологического оборудования.

АСУ ТП ПС должна обеспечивать получение и доступ пользователя к информации о выполняемых в смежных системах или комплексах следующих задач мониторинга состояния и эксплуатации основного технологического оборудования:

– мониторинг работы первичного оборудования, в том числе автотрансформаторов (трансформаторов) и устройств РПН, синхронных компенсаторов и др.;

– контроль состояния элегазового оборудования;

– контроль состояния ОПН, разрядников;

– контроль срабатывания устройств КИВ;

– контроль состояния ЩПТ, АБ, ЩСН.

1.2.3 Типовой состав технологических функций АСУ ТП подстанции

Технологические функции АСУ ТП обеспечивают решение задач, рассмотренных в параграфе 1.2.2. В нормативном документе [32] регламентирован широкий спектр функций, критически важных для обеспечения надежности и

эффективности работы энергетических объектов. Приведем обзор лишь основных, разделив их на шесть групп:

1. Мониторинг и визуализация:

– сбор, обработка и представление информации о текущем состоянии технологического оборудования;

– отображение данных в режиме реального времени на автоматизированных рабочих местах (АРМ) оперативного персонала с возможностью архивирования информации и последующего анализа;

– индикация отклонений от нормы в режиме мнемосхем, включающая динамическое отображение технологических параметров и состояния оборудования.

2. Технологическая предупредительная и аварийная сигнализация, включающая:

– регистрацию предупредительных и аварийных событий;

– контроль выхода параметров за допустимые эксплуатационные пределы;

– автоматизированную фильтрацию и обработку поступающих сигналов.

Также в эту группу входит вывод критически важных сообщений на терминалы управления и рабочие станции персонала, что обеспечивает своевременную реакцию на нештатные ситуации.

3. Автоматизированное и дистанционное управление коммутационными аппаратами (КА) ПС (см. параграф 1.2.1).

4. Обмен информацией в рамках энергосистемы:

– передача оперативной информации в центры управления сетями (ЦУС), региональные и объединенные диспетчерские управления (РДУ, ОДУ);

– обмен неоперативными технологическими данными с магистральными и первичными энергосистемами (МЭС, ПМЭС) в целях оптимизации сетевого взаимодействия.

5. Диагностика и учет ресурса оборудования:

– мониторинг работоспособности первичного оборудования, в том числе автоматизированный контроль параметров КА;

- оценка остаточного ресурса КА ПС, позволяющая прогнозировать необходимость их технического обслуживания и модернизации;

- диагностика измерительных систем устройств РЗА, а также аппаратуры АСУ ТП и других средств измерения.

6. Автоматизированный мониторинг климатических факторов, включающий:

- контроль температурного режима в помещениях и на ОРУ;

- оценку влажности, атмосферного давления и других параметров, влияющих на работу оборудования ПС.

В стандарте организации [32] приведена полная таблица соответствия задач и технологических функций АСУ ТП.

1.2.4 Типовые структуры и информационное взаимодействие в АСУ ТП подстанции

Основные требования к программно-техническим средствам и комплексам АСУ ТП ПС единой национальной электрической сети (ЕНЭС), изложенные в данном параграфе, обеспечивают выполнение функций и решение задач, описанных в [32] и в параграфах 1.2.2, 1.2.3.

Программно-технический комплекс (ПТК) АСУ ТП, согласно [33], может быть выполнен в трех различных вариантах архитектуры.

Архитектура построения ПС I типа. Информационное взаимодействие между интеллектуальными электронными устройствами (ИЭУ) АСУ ТП выполняется при помощи объектно-ориентированных сообщений (протокол *GOOSE*) согласно стандарту МЭК 61850-8-1 [34], а обмен всей информацией между ИЭУ релейной защиты (РЗА) осуществляется дискретными и аналоговыми электрическими сигналами, передаваемыми по контрольному кабелю; информационный обмен со станционным уровнем (*SCADA*) осуществляется по цифровому протоколу *MMS* [35, 36]; измерения тока и напряжения передаются в виде электрических аналоговых сигналов с использованием контрольных кабелей.

Архитектура построения ПС II типа. Информационное взаимодействие между ИЭУ выполняется при помощи объектно-ориентированных сообщений (протокол *GOOSE*) согласно стандарту МЭК 61850-8-1; информационный обмен со станционным уровнем (*SCADA*) осуществляется по цифровому протоколу *MMS*; измерения тока и напряжения передаются в виде электрических аналоговых сигналов с использованием контрольных кабелей.

Архитектура построения ПС III типа. Информационное взаимодействие между ИЭУ РЗА выполняется при помощи объектно-ориентированных сообщений (протокол *GOOSE*) согласно стандарту МЭК 61850-8-1; информация от измерительных устройств тока и напряжения передается в цифровом виде с использованием протокола передачи мгновенных значений (*SV*) согласно стандарту МЭК 61850-9-2 [37]; информационный обмен со станционным уровнем (*SCADA*) осуществляется по цифровому протоколу *MMS*.

АСУ ТП ПС должна разрабатываться как интегрированная, иерархически распределенная человеко-машинная система, обеспечивающая комплексное выполнение функций управления, мониторинга, сбора, обработки, хранения и передачи технологической информации. Данная система должна обеспечивать надежное взаимодействие всех уровней управления электрической ПС, а также соответствовать требованиям информационной безопасности и нормативно-правовым актам.

При проектировании АСУ ТП ПС необходимо учитывать положения Федерального закона от 26.07.2017 № 187-ФЗ «О безопасности критической информационной инфраструктуры Российской Федерации» [38], а также действующую на момент разработки Единую техническую политику в электросетевом комплексе.

Одним из ключевых требований к АСУ ТП является возможность масштабирования и модернизации как на аппаратном, так и на программном уровнях. Архитектура системы должна предусматривать реализацию трехуровневой модели программно-технических средств: полевого уровня, уровня присоединения и станционного уровня [33]. Рассмотрим эти уровни подробнее.

Полевой уровень представляет собой основу для первичного сбора и передачи измерительной информации, обеспечивая интерфейс между первичным оборудованием и вычислительными комплексами АСУ ТП. В его состав входят:

- первичные преобразователи (датчики), которые не входят в стандартный комплект оборудования;
- цифровые блоки и объединяющие устройства для работы с цифровыми и оптическими трансформаторами тока и напряжения.

Все устройства должны устанавливаться в непосредственной близости от контролируемого оборудования. При этом в случае комплектных поставок измерительные преобразователи могут входить в состав первичного оборудования, оставаясь частью полевого уровня, но не входя в состав программно-технического комплекса (ПТК).

Уровень присоединения выполняет функции агрегации данных от полевого уровня, обеспечивая первичную обработку информации, анализ событий и передачу управляющих команд. В его состав входят:

- контроллеры присоединения (КП), осуществляющие централизованный сбор, обработку данных, а также формирующие сигналы управления;
- интеллектуальные электронные устройства РЗА, обладающие функциями автоматического управления;
- интегрируемые устройства смежных систем РЗА.

Станционный уровень отвечает за централизованное управление, обработку и хранение информации, а также за интеграцию с диспетчерскими центрами и системами верхнего уровня. В его состав входит всевозможное сетевое оборудование (в том числе маршрутизаторы и межсетевые экраны, обеспечивающие формирование демилитаризованной зоны), серверы разного назначения (АСУ ТП, удаленной обработки информации, сбора низкоприоритетной информации и т.д.), а также системы безопасности и отказоустойчивости (гарантированное электропитание, резервное копирование и хранение данных, обнаружение вторжений на сетевом уровне).

Современные автоматизированные системы управления электрическими ПС предъявляют повышенные требования к надежности и скорости обмена данными. В данной области одним из ключевых стандартов является упомянутый ранее МЭК 61850-8-1, в рамках которого определены различные механизмы передачи информации.

Одним из механизмов является использование сообщений протокола *MMS*, входящего в состав МЭК 61850-8-1 и реализующего клиент-серверную модель обмена данными. Этот механизм используется преимущественно для взаимодействия между ИЭУ и системами верхнего уровня, а также между отдельными ИЭУ. Протокол *MMS* реализует функции опроса и мониторинга, обеспечивая высокую степень формализации и стандартизации передаваемых данных.

Вторым механизмом является применение сообщений протокола *GOOSE* [34], который также определен в рамках МЭК 61850-8-1 и предназначен для быстрого обмена информацией о событиях между ИЭУ внутри ЛВС. Он позволяет минимизировать задержки при передаче команд на коммутацию и при реагировании на аварийные ситуации, что существенно повышает общую оперативность работы всей системы. Его основные характеристики включают:

- механизм многопоточности (*multicast*) распространения данных, что обеспечивает быструю доставку сообщений нескольким получателям без необходимости подтверждения приема;

- малую задержку передачи (менее 4 мс), что критически важно для РЗА и ПА.

Стандарт [33] определяет следующие структуры информационного взаимодействия в клиент-серверной технологии, подробно показанные в Приложении А.

1. Информационное взаимодействие КП и преобразователя дискретных сигналов (ПДС) посредством цифровых связей (сообщений «*GOOSE*» согласно стандарту МЭК 61850-8-1) в ЛВС (рис. 1.3).

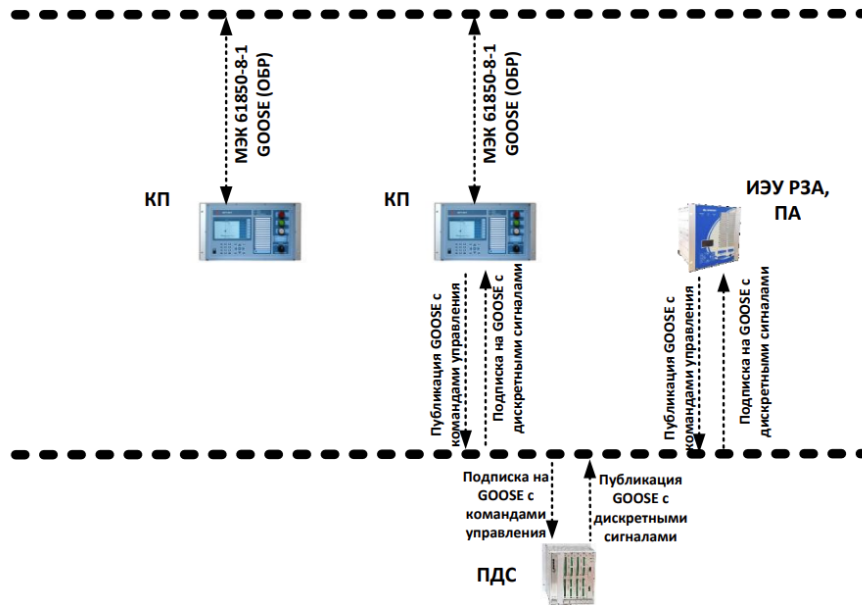


Рисунок 1.3– Информационное взаимодействие КП – ПДС
(архитектура II)

2. Информационное взаимодействие КП и серверов телемеханики (ТМ) АСУ ТП посредством цифровых связей *MMS*, согласно стандарту МЭК 61850-8-1 (рис. 1.4).

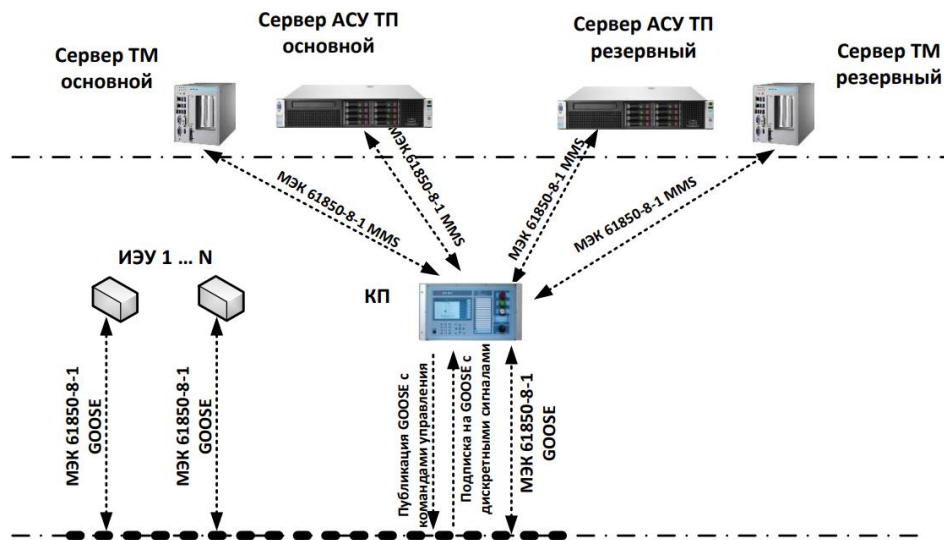


Рисунок 1.4– Информационное взаимодействие КП – серверы ТМ и АСУ ТП
(архитектура II и III)

3. Информационное взаимодействие измерительного преобразователя (ИП) с КП, серверами ТМ и АСУ ТП (архитектура II) посредством цифровых связей *MMS*, согласно стандарту МЭК 61850-8-1 (рис. 1.5).

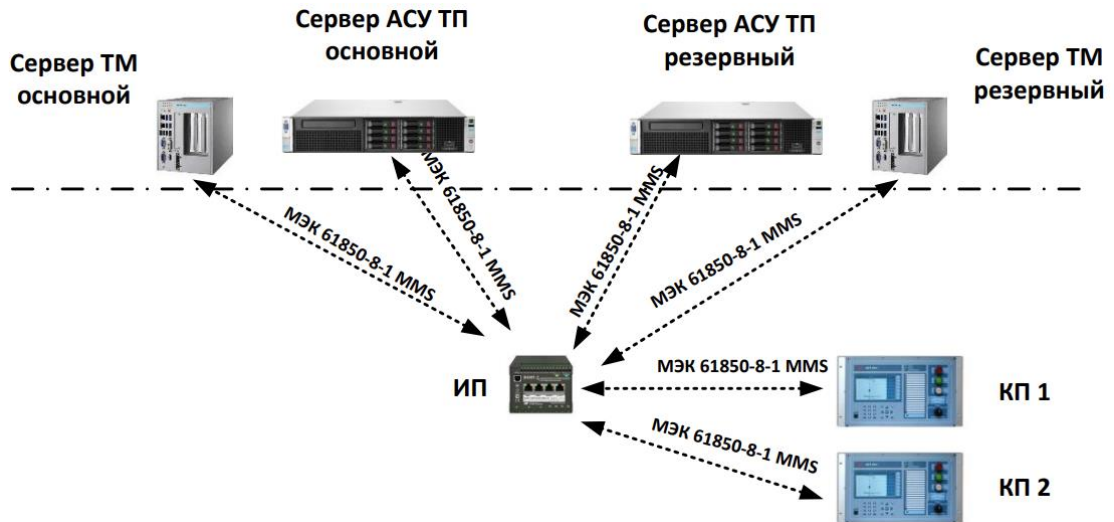


Рисунок 1.5 – Информационное взаимодействие ИП – КП, серверы ТМ и АСУ ТП (архитектура II)

4. Информационное взаимодействие ПДС и интерфейсного модуля цифрового измерительного трансформатора с КП, ИЭУ РЗА и ПА посредством цифровых связей SV, согласно стандарту МЭК 61850-9-2 (рис. 1.6).

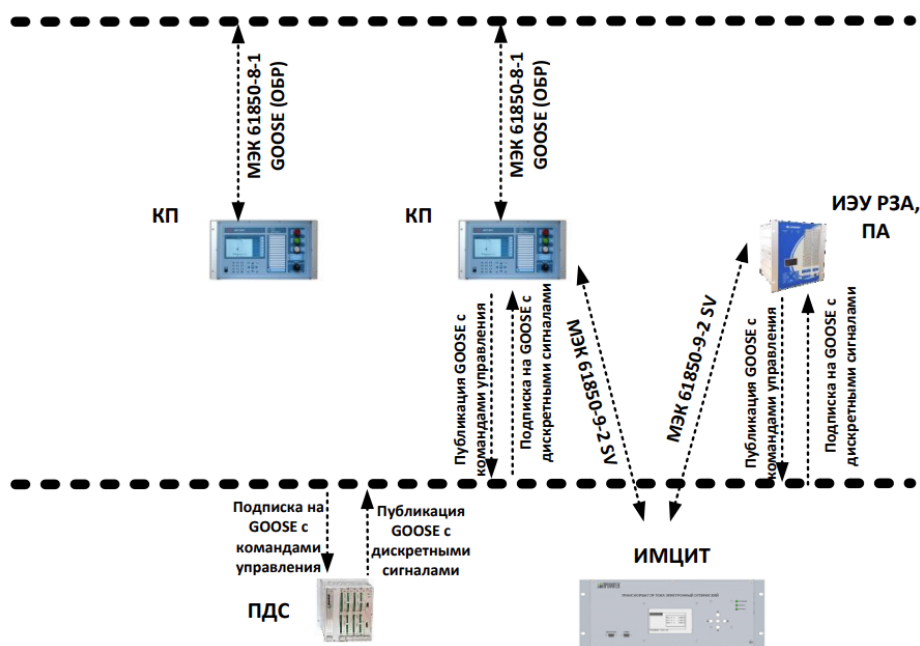


Рисунок 1.6 – Информационное взаимодействие ПДС, ИМЦИТ – КП, ИЭУ РЗА, ПА (архитектура III)

1.3 Анализ требований к системам сбора и обработки информации в составе АСУ ТП подстанции

В процессе анализа типовых структурных схем организации и информационного взаимодействия в АСУ ТП ПС, подробно рассмотренных в параграфе 1.2.4, было выявлено, что КП занимает ключевое место на уровне присоединения АСУ ТП ПС, выполняя важные функции измерения входных аналоговых и дискретных сигналов, их последующей обработки, управления отдельными узлами всей системы. От надежности и безотказности функционирования КП не в последнюю очередь зависит качество работы всей АСУ ТП ПС. Это подтверждается стандартом организации [39], согласно которому **«Контроллеры присоединения** предназначены для измерения, присвоения метки времени, сбора, обработки аналоговых сигналов и двухпозиционных сигналов о состоянии оборудования по присоединению, обмена данными с верхними уровнями управления и управления коммутационными аппаратами с организацией блокировок. Контроллеры присоединения рассматриваются как самостоятельные изделия, которые могут, в том числе, применяться для автоматизации технологических процессов на подстанциях ЕНЭС для присоединений с уровнем напряжения 110-750 кВ».

КП разрабатываются ОАО «НТЦ ФСК ЕЭС», однако, разработкой этих устройств могут заниматься и компании, не относящиеся к ЕЭС, например: ООО «Систел» или ФГУП «ЭЗАН». Независимо от компании-разработчика, готовое устройство должно в обязательном порядке соответствовать нормативным требованиям к конструкции, вводу аналоговых сигналов, вводу\выводу дискретных сигналов, обработке информации, климатическим характеристикам и т.д. упомянутого ранее стандарта организации [39].

1.3.1 Нормативные требования к контроллеру присоединения

Функциональные характеристики КП в рамках требований ПАО «ФСК ЕЭС» регламентируются рядом ключевых параметров, определяющих их роль в

автоматизированных системах управления электрическими ПС. Согласно нормативным требованиям [39], далее представлены основные задачи, которые должен выполнять КП.

– Обработка и анализ данных с первичного оборудования. КП должен осуществлять сбор, измерение и преобразование аналоговых и дискретных сигналов, поступающих от вторичных обмоток измерительных трансформаторов тока (ТТ), напряжения (ТН), блокировочных контактов первичного оборудования, устройств РЗА, а также датчиков и измерительных преобразователей.

– Формирование управляющих команд. КП должен обеспечивать формирование и передачу управляющих сигналов к КА, устройствам РПН и прочему исполнительному оборудованию.

– Синхронизация времени. Для координации процессов в автоматизированной системе управления требуется синхронизация системного времени по сигналам систем единого времени, что критически важно для обеспечения точности присваиваемых временных меток и согласованного функционирования системы.

– Реализация программных блокировок. КП должен поддерживать программные механизмы оперативных блокировок, предотвращающие некорректное или несанкционированное изменение режимов работы оборудования.

– Передача данных на верхний уровень системы управления. КП должен передавать собранную информацию на верхний уровень различными способами, такими как спорадическая передача (по апертуре), циклическая отправка данных с заданной периодичностью, передача по запросу оператора или системы.

– Взаимодействие с другими элементами АСУ ТП. КП должен обеспечивать интеграцию с терминалами релейной защиты и противоаварийной автоматики, а также с контроллерами других присоединений и АСУ ТП среднего уровня. Для обмена данными между КП и другими интеллектуальными устройствами АСУ ТП используются протоколы *IEC 61850-8-1 (MMS, GOOSE)*, обеспечивающие стандартизированную коммуникацию. КП должны поддерживать

протокол *PRP*, что позволяет реализовать повышенную отказоустойчивость передачи данных за счет дублирования информационных потоков.

– Резервирование управления КА при неисправности верхнего или среднего уровней АСУ ТП).

– Обеспечение самодиагностики элементов системы.

Основными **конструктивными** требованиями согласно [39, 40] является промышленное исполнение КП, наличие блочного каркаса, модульность конструкции с возможностью замены неисправного сменного элемента, охлаждение контроллера за счет естественной конвекции, наличие сигнальных дискретных выходов (не менее двух: «Отсутствие питания» и «Аварийное обслуживание»). Степень защиты устройства должна быть не хуже *IP20*. Конструкция устройства должна обеспечивать возможность размещения в шкафах с защитной оболочкой до *IP54* без дополнительной вентиляции.

Развитие цифровых систем управления электрическими ПС предъявляет строгие требования к **вводу аналоговых сигналов**, поступающих от измерительных ТТ и ТН. В соответствии с современными стандартами и регламентами автоматизированных систем управления, ключевыми характеристиками ввода аналоговых сигналов являются следующие.

1. Основные требования к конфигурации аналоговых входов:

– количество аналоговых каналов должно составлять не менее четырех входов для сигналов от ТТ и четырех входов для сигналов от ТН;

– необходимо предусмотреть возможность увеличения числа входных каналов за счет установки дополнительных модулей в состав КП, что обеспечивает масштабируемость системы и гибкость конфигурации.

2. Временная привязка и обработка данных. Для обеспечения точности анализа и синхронизации событий каждому измерению должна присваиваться временная метка с точностью не хуже $\pm 1,0$ мс.

3. Алгоритмы первичной обработки аналоговых данных. При обработке аналоговых сигналов должны выполняться следующие вычислительные операции:

– масштабирование;

- расчет производных параметров;
- контроль граничных условий – анализ динамики изменений измеренных параметров с определением выхода за установленные пороговые значения (апертурный контроль); границы допустимых значений должны формироваться относительно последнего зафиксированного достоверного измерения;
- приоритетная обработка аномальных сигналов – если один или несколько сигналов выходят за пределы установленной апертуры, система должна осуществлять их поочередную детализированную обработку, что позволяет минимизировать вероятность ложных срабатываний и повысить точность диагностики состояния оборудования ПС.

Кроме того, модули ввода аналоговых сигналов в составе КП должны измерять и вычислять параметры, упомянутые в параграфе 1.2.1 (действующие значения напряжений по каждой фазе и линейные; действующие значения токов по каждой фазе; частота сети; активная мощность пофазно и суммарная; реактивная мощность пофазно и суммарная; полная мощность пофазно и суммарная; коэффициент активной мощности пофазно и общий). Для всех измеренных и вычисленных параметров абсолютная погрешность должна быть не более $\pm 0,5$ %, для частоты сети – не более $\pm 0,01$ %.

Требования к **дискретным входам**:

- количество дискретных входов не менее 16 шт. с возможностью установки дополнительных модулей в контроллер;
- должна быть предусмотрена гальваническая изоляция дискретных входов контроллера;
- периодичность опроса сигналов 0,5 - 1,0 мс.

Требования к **дискретным выходам**:

- выходные контактные устройства должны обеспечивать гальваническое разделение контроллера присоединения с внешними цепями;
- количество сигнальных дискретных выходов не менее 2.

Требования к **коммуникационным модулям и протоколам интеграции**:

- КП должен осуществлять обмен с оборудованием вышестоящих уровней по протоколу IEC 61850-8-1 (MMS);
- для организации оперативных блокировок применяется протокол IEC 61850-8-1 (GOOSE);
- КП должен поддерживать интерфейсы 10/100Base – TX (обязательно) и 10/100 Base – FX (опционально).

1.3.2 Оценка числа измерительных и управляющих каналов контроллера присоединения

В стандарте организации [39] сформулированы лишь минимальные требования к числу измерительных и управляющих каналов КП, но при этом оговорена возможность увеличения этого числа. Следовательно, при проектировании КП необходимо оценить и максимальное количество каналов, чтобы предусмотреть конструктивную возможность установки необходимого числа модулей и обеспечить запас пропускной способности интерфейса межмодульного взаимодействия. С целью оценки максимального числа измерительных и управляющих каналов КП автором выполнен анализ типовых схем электрических ПС.

Стандартом организации [41] установлено минимальное количество типовых схем, охватывающих большинство встречающихся на практике случаев проектирования и реконструкции действующих ПС, позволяющих обеспечить надежность и живучесть ПС и достичь экономичных унифицированных решений. Все типовые схемы разделены по классам напряжений от 6 до 750 кВ: 6 кВ (3 схемы); 20 кВ (2 схемы); 35 кВ (6 схем); 110 кВ (17 схем); 220 кВ (19 схем); 330 кВ (7 схем); 500 кВ (6 схем); 750 кВ (5 схем).

В ходе анализа типовых схем организации ПС было выявлено, что состав основного оборудования, описанного в параграфе 1.2.1, для каждой ПС во многом одинаков, но количество элементов различных типов в зависимости от сложности схемы может существенно различаться. Выполним оценку количества основного оборудования подстанции на примере трех схем класса напряжения 330 кВ.

Наиболее простая схема №330-3Н (Блок «линия-трансформатор» с выключателем), показанная на рис. 1.7, имеет в своем составе один измерительный трансформатор напряжения (ТН) и один трансформатор тока (ТТ), два выключателя нагрузки (ВН) и четыре заземляющих ножа (ЗН).

Показанная на рис. 1.8 схема средней сложности №330-6Н (Треугольник) имеет в своем составе 2 ТН и 3 ТТ с возможностью установки еще одного ТН и двух ТТ при проектировании ПС, 6 ВН, 4 разъединителя и 16 ЗН.

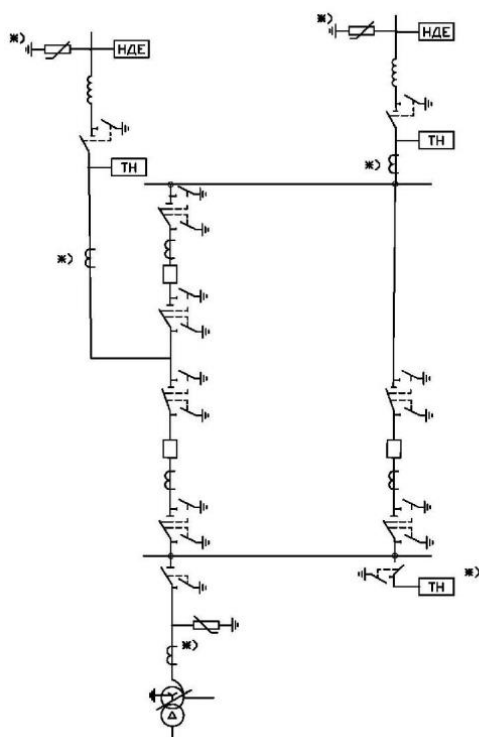
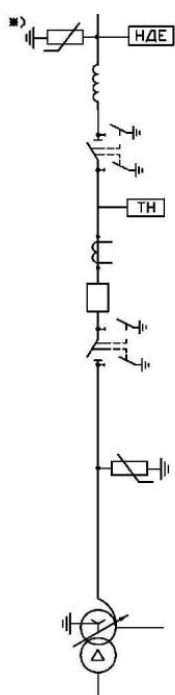


Рисунок 1.7 – Типовая схема ПС №330-3Н.
Блок «линия-трансформатор» с выключателем

Рисунок 1.8 – Типовая схема ПС №330-6Н.
Треугольник

Последняя в данном классе напряжения наиболее сложная схема №330-17 (Полуторная схема), показанная на рис. 1.9, имеет уже 10 ТН и 12 ТТ с возможностью установки еще 12 ТТ при проектировании, 23 ВН, 14 разъединителей и 60 ЗН.

КП, согласно указанным в параграфе 1.3.1, требованиям должен выполнять сбор и обработку аналоговой и дискретной информации с оборудования ПС, а также формировать сигналы управления КА ПС. Чтобы обеспечить выполнение

этих требований, КП должен иметь соответствующую **информационную ёмкость**, которая характеризуется количеством дискретных входов, дискретных выходов, аналоговых входов, а также некоторыми дополнительными функциями, которые могут быть реализованы в контроллере.

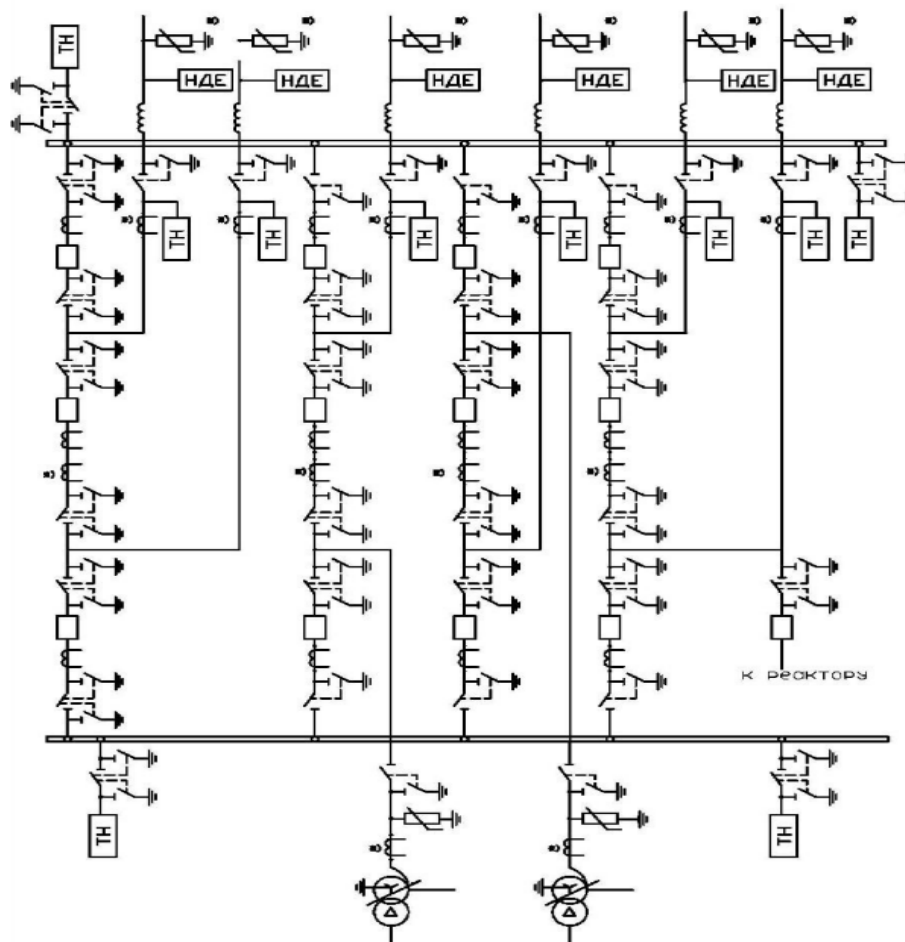


Рисунок 1.9 – Типовая схема ПС №330-17. Полуторная

Количество входов и выходов КП зависит от количества измерительных и управляющих каналов, а оценка их числа включает анализ типов контролируемых параметров, функций защиты и автоматики, а также имеющихся локальных систем мониторинга и управления, и сильно изменяется в зависимости от схемы подстанции.

Формирование структуры измерительных и управляющих каналов в КП определяется характером передаваемых сигналов. В соответствии с функциональным назначением и технологическими особенностями процессов передачи данных выделяются три ключевые группы сигналов.

1. Сигналы телеизмерений (ТИ) – передают текущие значения электрических параметров, определяющих технологический режим функционирования электроэнергетического оборудования. Эти данные обеспечивают основу для анализа состояния системы, оперативного прогнозирования её работы и автоматизированного управления энергообъектами.

2. Сигналы телесигнализации (ТС) – отражают дискретные события и переходные процессы, возникающие в электрической сети. К данной категории относятся изменения состояния КА (включено, отключено), срабатывание устройств релейной защиты и автоматики и аварийно-предупредительная сигнализация, информирующая о критических отклонениях параметров сети или сбоях в работе оборудования.

3. Сигналы телеуправления (ТУ) – представляют собой команды дистанционного управления КА и иными исполнительными устройствами ПС. Эти команды даются операторами диспетчерских пунктов или автоматически генерируются системой управления для обеспечения быстрого реагирования на изменения в энергосистеме.

Стандартом организации [42] установлен необходимый объем и источники передаваемой телеинформации об электроэнергетическом режиме, состоянии электрической сети и сетевого оборудования.

Сигналы ТИ. Основными источниками данных о параметрах работы электрооборудования являются измерительные ТТ и ТН, обеспечивающие высокоточную регистрацию режимных характеристик.

Для контроля токовых нагрузок используются ТТ с номинальным выходным током 1 или 5 А. Измерение напряжения осуществляется с применением ТН с выходным напряжением 58 или 100 В.

Важным требованием является исключение промежуточных аналоговых преобразователей, что позволяет минимизировать погрешности и потери информации. Оцифровка поступающих измерительных данных осуществляется непосредственно в КП или в связанном с ним по цифровому интерфейсу

измерительном преобразователе, что повышает точность и оперативность обработки информации.

С помощью ТТ измеряется действующее значение фазного тока (I_a, I_b, I_c) и ток нулевой последовательности (I_0) – по одному измерительному каналу на параметр. С помощью ТН измеряется действующее значение фазного напряжения (U_a, U_b, U_c) и напряжение нулевой последовательности (U_0) – по одному измерительному каналу на параметр.

Сигналы ТС. Основными источниками сигналов событий являются блок-контакты и концевые выключатели силовых коммутационных аппаратов (ВН, отделителей, короткозамыкателей, разъединителей и ЗН, тележек выкатных элементов КРУ). Количество каналов для получения информации о состоянии КА может сильно варьироваться в зависимости от конструкции КА. Традиционно сложилось так, что изменение положения КА контролируется в системе по состоянию одновременно двух контактов – нормально-разомкнутый и нормально-замкнутый. Это исключает ошибку в оценке положения КА, позволяет контролировать длительность промежуточного (оба контакта разомкнуты) и недопустимого (оба контакта замкнуты) состояния и отображать эти состояния на пульте контроля дежурного персонала. Таким образом, для КА контролируется, как минимум, положение главных контактов (контакт «Включен» и контакт «Отключен»), всего 2 сигнала. Кроме того, может контролироваться состояние привода – 1 сигнал, срабатывание релейной защиты (отключение защитой) – 1 сигнал, блокировка включения – 1 сигнал, признак ручного управления – 1 сигнал.

Сигналы ТУ. В качестве сигналов ТУ (команд) используются дискретные сигналы, при помощи которых обеспечивается управление оборудованием ПС, а именно электромагнитами включения и отключения в приводах ВН, ЗН или реле команды. В ответственных случаях (например, для управления КА) обычно используется два сигнала: «Включить» и «Отключить».

Для рассмотренных ранее типовых схем ПС с учётом типов и количества применяемого в них оборудования выполнена оценка числа измерительных и управляющих каналов КП, обеспечивающих получение всей необходимой

телеинформации и передачу команд управления. Результаты оценки представлены в табл. 1.1.

Таблица 1.1 – Количество измерительных и управляющих каналов КП для типовых схем ПС.

| Типовая схема | Измерительные трансформаторы | | | | Выключатели, разъединители | | | | Заземляющие ножи | | | |
|---------------|------------------------------|-------------|----|----|----------------------------|-------------|-------|-------|------------------|-------------|--------|----|
| | Кол-во | Тип сигнала | | | Кол-во | Тип сигнала | | | Кол-во | Тип сигнала | | |
| | | ТИ | ТС | ТУ | | ТИ | ТС | ТУ | | ТИ | ТС | ТУ |
| №330-3Н | 2 | 8 | - | - | 2 | - | 2-4 | 2-4 | 4 | - | 4-8 | - |
| №330-6Н | 8 | 32 | - | - | 10 | - | 10-20 | 10-20 | 16 | - | 16-32 | - |
| №330-17 | 34 | 136 | - | - | 37 | - | 37-74 | 37-74 | 60 | - | 60-120 | - |

Как видно из таблицы, количество измерительных и управляющих каналов КП для различных типовых схем ПС может отличаться более чем в 10 раз, что требует применения КП с существенно различающейся информационной ёмкостью.

1.3.3 Анализ контроллера присоединения как системы сбора и обработки информации

В результате анализа нормативных требований к КП (параграф 1.3.1) становится ясно, что основные функции КП схожи с функциями систем сбора и обработки информации (ССОИ). Так, например, в [43] сказано, что «**Система сбора и обработки информации** — это комплекс аппаратных и программных средств, предназначенных для сбора, хранения, обработки и передачи данных, используемых в процессе управления или принятия решений». КП выполняет все ключевые функции ССОИ: сбор информации (измерение значений токов и напряжений, получение состояния КА), её обработку (вычисление активной, реактивной и полной мощностей, выявление выхода значения за заданные пределы), передачу (информационное взаимодействие со стационарным уровнем АСУ ТП ПС), а также управление (формирование команд управления КА), то есть является частной реализацией ССОИ на электрических ПС.

ССОИ, в широком смысле, для различных производственных сфер могут быть реализованы в разнообразном исполнении, начиная от модульной конструкции устройства под определённые требования до монолитной с большим запасом информационной ёмкости и вычислительной способности в едином устройстве. Часто вычисление многих требуемых параметров выполняется удалёнными серверами, а измеренные значения сохраняются в облачных хранилищах, такой подход позволяет снизить нагрузку на локальные вычислительные блоки ССОИ.

К ССОИ ПС предъявляются рассмотренные ранее в параграфе 1.3.1 нормативные требования, согласно которым она должна иметь модульную структуру и выполнять сбор, обработку, передачу информации и управление на месте, то есть непосредственно на ПС. Объясняется это необходимостью повышения функциональной надёжности ССОИ за счёт реализации функций самодиагностики, упрощения обслуживания ССОИ («горячая» замена однотипных составных частей (модулей) в случае неисправности, установка дополнительных модулей при необходимости увеличить информационную ёмкость системы), рационального (экономически обоснованного) использования запаса информационной ёмкости. Последний пункт определяет, в частности, экономическую нецелесообразность использования моноблочных ССОИ с информационной ёмкостью, значительно превышающей необходимую на конкретном присоединении определённой подстанции.

Подводя итог вышесказанного, можно сделать вывод, что при проектировании ПС для каждого присоединения должны разрабатываться ССОИ, соответствующие требованиям «ФСК ЕЭС» и обладающие необходимой информационной ёмкостью за счёт использования модульной структуры.

При проектировании модульных ССОИ остро встаёт вопрос организации надёжного межмодульного взаимодействия, гарантирующего обмен высокоприоритетной информацией без потерь и обеспечивающего достаточную скорость передачи низкоприоритетной информации.

1.4 Постановка задач исследований

1.4.1 Разработка номенклатуры модулей систем сбора и обработки информации электрических подстанций и выбор интерфейса межмодульного взаимодействия

Стандартом [39] и выполненным в параграфе 1.3.3 анализом подтверждается необходимость использования модульной структуры при проектировании ССОИ ПС. Система должна содержать определённый перечень модулей, необходимых для обеспечения сбора и передачи измеренных параметров, управления оборудованием ПС, а также обеспечения собственных нужд. Модульный состав системы определяется на этапе проектирования и зависит от особенностей присоединения, для которого она разрабатывается, но в неё обязательно входят модули ввода унифицированных аналоговых сигналов, ввода и вывода дискретных сигналов, коммуникационные модули, процессорный модуль и модули источников питания.

Однако нормативная документация содержит лишь общие требования к конструктивному исполнению ССОИ ПС и не конкретизирует типы и количество сигналов каждого модуля, способы взаимодействия модулей друг с другом. Поэтому одной из первых задач, возникающих при проектировании масштабируемых модульных систем, является разработка номенклатуры модулей, обеспечивающей выполнение необходимых функций и не противоречащей предъявляемым требованиям.

Второй задачей, связанной с реализацией модульной структуры, является выбор интерфейса и протокола взаимодействия модулей. Для организации межмодульного взаимодействия могут использоваться различные интерфейсы, отличающиеся пропускной способностью, надёжностью, стоимостью реализации и возможностью использования тех или иных протоколов. Правильный выбор интерфейса и протокола позволит обеспечить выполнение нормативных требований по быстродействию ССОИ и реализовать передачу данных без потерь с сохранением экономически обоснованной и целесообразной стоимости изделия.

Количество входных и выходных сигналов каждого модуля, а также общее количество модулей в системе непосредственно влияют на интенсивность межмодульного взаимодействия и, следовательно, определяют выбор используемого интерфейса и протокола. Таким образом, вслед за разработкой номенклатуры модулей следует выполнить анализ распространённых промышленных интерфейсов, предназначенных для создания систем, работающих в режиме реального времени, обеспечивающих высокую достоверность обмена данными, отличающихся невысокой стоимостью аппаратной реализации, имеющих развитый механизм автоматического обнаружения ошибок и эффективные протоколы обмена данными, после чего выбрать из них наиболее соответствующий предъявляемым требованиям организации надежного межмодульного взаимодействия.

1.4.2 Разработка средств описания модульных систем сбора и обработки информации

Модульная структура современных ССОИ ПС позволяет с использованием сравнительно небольшой номенклатуры унифицированных модулей скомпоновать систему, обеспечивающую наиболее эффективное решение конкретной прикладной задачи для конкретного объекта. Однако для этой структуры характерны и все недостатки сетевой организации. Наиболее важный из них – существенное влияние качества сетевого взаимодействия на показатели эффективности функционирования системы и на её общую работоспособность. Более того, на основные параметры сетевого взаимодействия – скорость обмена данными, своевременность их доставки, вероятность искажения информации – непосредственно влияет сложность конкретной системы и число взаимодействующих модулей, входящих в её состав, которое в разных реализациях ССОИ ПС может изменяться в весьма широких пределах.

С ростом количества модулей возрастает и вероятность возникновения ошибок в настройке межмодульного взаимодействия, причём обнаружить их зачастую можно только на этапе ввода ССОИ в эксплуатацию или на еще более

поздних этапах функционирования ПС, что приводит к существенным материальным затратам на их устранение, а иногда и к аварийным ситуациям. Зачастую причиной упомянутых ошибок является отсутствие адекватных средств описания ССОИ, как совокупности взаимодействующих модулей. В результате становится возможной несогласованная настройка параметров приёмников и передатчиков информации.

Хорошим примером системного подхода к описанию всех аспектов взаимодействия компонентов внутри многоуровневой информационно-измерительной и управляющей системы является комплекс стандартов МЭК 61850 [35, 36, 37]. Он реализует объектно-ориентированную модель, в которой технологическое оборудование (трансформаторы, КА, датчики и т.п.) представляется в виде объектов, имеющих некоторые атрибуты (свойства) и поведение (функции). Объекты взаимодействуют друг с другом посредством стандартизированных сообщений, что обеспечивает совместимость компонентов системы и единообразие способов их взаимодействия. Однако для описания межмодульного взаимодействия в ССОИ ПС стандарты МЭК 61850 являются избыточными, поскольку ориентированы в первую очередь на сложные многоуровневые системы в электроэнергетике. Кроме того, в качестве среды передачи данных они подразумевают использование сетей *Ethernet* [17] и семейства протоколов *TCP/IP* [44, 45] что, с одной стороны, сильно ограничивает выбор интерфейса и протокола, а с другой – приводит к неоправданным материальным затратам на реализацию сравнительно сложного интерфейса, избыточного для решаемой задачи.

Для других широко используемых в промышленности и, в частности, в энергетике интерфейсов и протоколов (МЭК 60870 [46], *Modbus* [47], *CANopen* [48]) также существуют средства описания параметров взаимодействия, но их общим недостатком является ориентированность на описание отдельных модулей, а не системы в целом. В результате формируется множество отдельных описаний, согласованность которых не гарантируется.

Примером универсального средства описания систем и взаимодействия их элементов является язык *UML* [49], позволяющий в наглядном графическом виде показать компоненты системы и информационные потоки между ними. Но *UML* представляет собой лишь один из удобных инструментов описания, в то время как разработка с его помощью самого описания рассматриваемого класса систем является отдельной непростой задачей.

Следовательно, чтобы значительно сократить количество возникающих ошибок настройки межмодульного взаимодействия и, как следствие, обеспечить качество функционирования модульных ССОИ ПС требуется разработка средств описания этих систем, учитывающих особенности решаемых задач.

1.4.3 Создание методов и моделей для оценки параметров межмодульного информационного взаимодействия

В процессе проектирования модульных систем, в частности ССОИ ПС, одной из важнейших задач является организация эффективного и надёжного взаимодействия модулей системы. При этом эффективность и надёжность определяются по-разному в зависимости от того, к данным какого типа применяются эти понятия.

В модульных ССОИ обычно можно выделить два типа данных, которыми модули обмениваются друг с другом.

1. Основные, как правило высокоприоритетные, данные, потеря или несвоевременная доставка которых критическим образом влияет на корректную работу всей ССОИ. Критерием эффективности межмодульного взаимодействия для таких данных является их гарантированная доставка в течение оговоренного интервала времени, определяемого на этапе проектирования ССОИ. Вероятность их потери в исправной ССОИ равна нулю, поскольку потеря рассматривается как отказ системы. Если какая-то часть данных была искажена в процессе передачи, то используемый протокол взаимодействия модулей должен обеспечить их повторную доставку в течение того же интервала времени.

2. Вспомогательные, как правило низкоприоритетные, данные, потеря или задержка передачи которых не приводит к нарушению работоспособности ССОИ, но влияет на время выполнения некоторых второстепенных функций. К таким функциям можно отнести, например, передачу предварительно сохранённых осциллограмм, протоколов событий, параметров настройки. Как и в предыдущем случае, при искажении некоторой части этих данных они должны быть переданы повторно, но жёсткие требования к времени передачи не предъявляются. Таким образом, для этих данных критерием эффективности межмодульного взаимодействия является скорость их передачи, требуемое значение которой определяется на этапе проектирования ССОИ.

Любой интерфейс передачи данных имеет ограниченную пропускную способность, поэтому возможность доставки данных первого типа в течение заданного интервала времени, как и скорость передачи данных второго типа, зависят от интенсивности межмодульного взаимодействия. Сама же интенсивность непосредственно зависит от количества используемых в системе модулей и параметров их настройки, которые, в свою очередь, определяются конфигурацией конкретного объекта, для которого проектируется ССОИ.

Таким образом, при неправильно выбранных параметрах настройки модулей в конкретной системе возможны ситуации, когда часть данных будет доставляться модулям-получателям с недопустимой задержкой или не будет доставляться вовсе. Чем на более ранних этапах проектирования будут выявлены эти проблемы, тем меньших материальных затрат потребует их решение. С другой стороны, сформулировать общие правила оптимальной настройки межмодульного взаимодействия для каждого объекта не представляется возможным в силу большого разнообразия характеристик этих объектов (см. параграф 1.3.2). В контексте решения данной проблемы общепризнанным подходом считается применение методов математического и имитационного моделирования [50].

С общими методологическими подходами к моделированию сетей передачи данных можно ознакомиться в [6, 7, 8, 9, 10, 11], однако авторы [7, 8, 9] подчеркивают, что, несмотря на наличие унифицированных концептуальных

основ, создание адекватной модели для конкретной системы представляет собой крайне сложную задачу, требующую учёта множества разнообразных аспектов. При создании модели сети необходимо учитывать особенности среды передачи данных, используемых протоколов, служб, также необходимо принимать во внимание специфику объекта, для которого разрабатывается модель.

Известны инструменты моделирования сетей [12, 13, 14, 15, 16] для широко распространённых технологий проводной (*Ethernet*) и беспроводной (*GSM* [18], *UMTS* [19], *Wi-Fi* [20], *Bluetooth* [21]) связи. Для этих технологий существуют доступные и готовые к использованию библиотеки, содержащие описания стандартных протоколов и типовых устройств. Однако менее распространённые промышленные интерфейсы межмодульного взаимодействия, такие, как *CAN* [22], *CAN FD* [23], *RS-485* [24], *SPI* [25], имеют свою специфику и требуют разработки отдельных библиотек для упомянутых средств моделирования, что по сложности сопоставимо с разработкой новой модели.

Таким образом, возникает необходимость в разработке методов и реализующих их моделей для анализа межмодульного взаимодействия в ССОИ, учитывающих особенности решаемых задач, в частности, выбранный интерфейс и протокол межмодульного взаимодействия.

1.5 Выводы по главе 1

В первой главе диссертации получены следующие основные результаты.

1. Рассмотрена классификация электрических подстанций (ПС), определён состав их основного оборудования. Выполнен анализ ПС как объекта контроля и управления, сформулированы задачи мониторинга и управления ПС. Рассмотрены типовые структурные схемы ПС и факторы, влияющие на выбор структурной схемы. Результаты анализа позволили сделать вывод, что в зависимости от используемой типовой схемы состав и количество оборудования ПС может сильно различаться.

2. Рассмотрены источники возникновения аварийных ситуаций на ПС, обосновано использование АСУ ТП, способствующих предотвращению подобных

ситуаций. Определены задачи и функции АСУ ТП, рассмотрены архитектуры построения и типовые структуры организации. В ходе анализа структурных схем и информационных связей АСУ ТП ПС было выявлено, что контроллер присоединения (КП) является одним из ключевых узлов информационного и управляющего взаимодействия.

3. Выполнен анализ нормативных требований к КП, а именно требования к конструкции, к входным и выходным сигналам, к информации, сбор и обработку и передачу которой выполняет КП. Выполнена оценка числа измерительных и управляющих каналов КП, на основании которой сделан вывод, что количество необходимых каналов для разных типовых схем ПС может отличаться в десятки раз, что требует применения КП с существенно различающейся информационной ёмкостью.

4. Выполнен анализ КП как системы сбора и обработки информации (ССОИ). Показано, что КП является частной реализацией ССОИ для электрических ПС. Сделан вывод, что при проектировании ПС для каждого присоединения необходима разработка ССОИ с существенно различающимися требованиями к информационной ёмкости.

5. Сформулированы задачи исследований, решение которых обеспечит повышение эффективности, надёжности и качества разрабатываемых ССОИ ПС:

- разработка номенклатуры модулей ССОИ ПС и выбор интерфейса межмодульного взаимодействия;
- разработка средств описания модульных ССОИ;
- создание методов и моделей для оценки параметров межмодульного информационного взаимодействия в ССОИ.

2 РАЗРАБОТКА МОДУЛЬНОЙ СТРУКТУРЫ СИСТЕМЫ СБОРА И ОБРАБОТКИ ИНФОРМАЦИИ ЭЛЕКТРИЧЕСКОЙ ПОДСТАНЦИИ И МЕТОДА ЕЁ ОПИСАНИЯ

2.1 Обоснование и разработка номенклатуры модулей системы сбора и обработки информации

Общие требования к конструктивному оформлению ССОИ ПС сформулированы в [40, 51], однако номенклатура модулей, количество входных и выходных сигналов нормативными документами не регламентируются. Поэтому в первую очередь решалась задача обоснования набора типов модулей, предназначенных для использования в системе.

Исходя из типовых задач, решаемых ССОИ ПС и описанных в параграфе 1.3, были предложены следующие типы модулей и обоснованы их основные характеристики.

Процессорный модуль ПМ. Предназначен для реализации общего алгоритма функционирования ССОИ, координации работы её модулей. Для синхронизации ССОИ с единым астрономическим временем имеет встроенный *GPS*-модуль. Для реализации протокола резервирования *PRP* [52] предусмотрены два порта *SFP*.

Коммуникационный модуль КМ. Предназначен для обеспечения взаимодействия с измерительными преобразователями, использующими последовательные интерфейсы. В зависимости от исполнения имеет до трёх портов *RS-422/485* и до двух портов *RS-232* с гальванической развязкой.

Модуль прямого ввода МПВ ($4 \times I$, $4 \times U$). Предназначен для измерения токов и напряжений присоединения и вычисления дополнительных параметров (активная и реактивная мощность, частота, фазы, линейные напряжения, коэффициент мощности). Имеет 4 входа для измерения напряжений фаз и нулевой последовательности (U_a, U_b, U_c, U_0) и 4 входа для измерения токов (I_a, I_b, I_c, I_0).

Модуль аналогового ввода МАВв ($8 \times AI$). Имеет восемь унифицированных аналоговых входов и предназначен для измерения токов и напряжений в других контролируемых точках, а также иных аналоговых сигналов.

Модуль дискретного ввода МДВв ($20 \times DI$). Имеет 20 дискретных входов и предназначен для ввода сигналов ТС.

Модуль дискретного вывода МДВыв ($8 \times DO$). Имеет 8 дискретных выходов и предназначен для вывода сигналов ТУ.

Количество входов МПВ определяется функциями, выполняемыми этим модулем. Количество входов и выходов МАВв, МДВв, МДВыв выбрано максимальным с учётом конструктивных ограничений, установленных в [40, 51]. В частности, учитывались требования к тепловому режиму модулей и ССОИ в целом, к электрической прочности изоляции между отдельными входами и выходами, к конструктивному исполнению коммутационных элементов (разъёмов).

Разработанная номенклатура позволяет адаптировать ССОИ к конкретной ПС путём установки необходимого количества модулей. На примере рассмотренных ранее типовых схем ПС 330 кВ и с учётом сведений о количестве входных и выходных сигналов, полученных в параграфе 1.3.2, выполнена оценка количества необходимых модулей для обеспечения нужд АСУ ТП ПС (табл. 2.1).

Таблица 2.1 – Количество необходимых модулей ССОИ для типовых схем ПС.

| Типовая схема | Модуль прямого ввода ($4 \times I$, $4 \times U$), шт. | Модуль аналогового ввода ($8 \times AI$), шт. | Модуль дискретного ввода ($20 \times DI$), шт. | Модуль дискретного вывода ($8 \times DO$), шт. |
|---------------|---|---|--|--|
| Схема №330-3Н | 1 | 1 | 2 | 1 |
| Схема №330-6Н | 3 | 2 | 4 | 3 |
| Схема №330-17 | 6 | 10 | 12 | 10 |

2.2 Анализ информационных потоков системы

Современные модульные ССОИ характеризуются необходимостью эффективного обмена данными (см. параграф 1.4.3) между различными

функциональными узлами, отличающимися по требованиям к скорости, надежности и приоритету передачи. Анализ информационных потоков в модульной ССОИ, структура которой показана на рис. 2.1, позволяет выделить несколько ключевых категорий взаимодействия, ранжированных по их критической значимости для функционирования системы (от критически важных до наиболее низкоприоритетных, но в некоторых случаях необходимых).

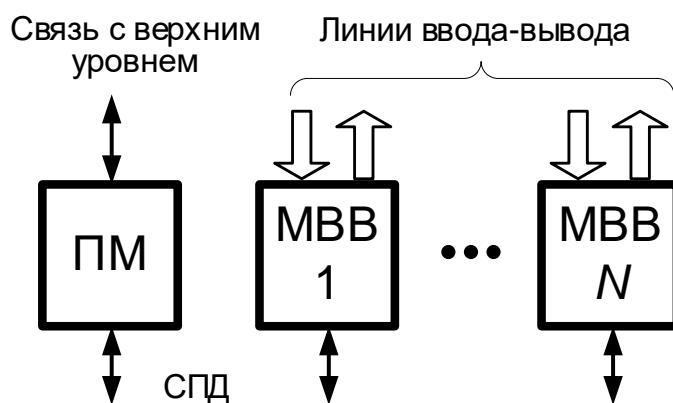


Рисунок 2.1 - Модульная структура ССОИ
 ПМ – процессорный модуль; МВВ – модули ввода-вывода;
 СВД – сеть передачи данных.

1. Управляющие сообщения процессорного модуля.

Центральным элементом информационного обмена является процессорный модуль (ПМ), который осуществляет диспетчеризацию и координацию работы системы посредством передачи управляющих команд на модули ввода\вывода ($МВВ_1 \dots МВВ_N$). Эти команды задают режим функционирования отдельных модулей («отключен», «сервисный режим», «рабочий режим» и др.) и обеспечивают корректное переключение состояний системы.

2. Сообщения синхронизации.

Для обеспечения временной когерентности процессов обработки данных ПМ формирует синхронизирующие сообщения, передаваемые всем модулям системы с некоторой частотой (для разных модульных систем эта частота может изменяться). Этот механизм позволяет гарантировать единство отсчёта времени

для всех модулей ССОИ, критически важное для обеспечения их согласованного функционирования.

3. Передача измеренных и вычисленных данных

В ответ на получение синхронизирующего сигнала любой модуль системы может передавать информационные сообщения, содержащие текущие измеренные или вычисленные величины. Передача осуществляется синхронно через фиксированные интервалы синхронизации или асинхронно (досрочно) при детектировании значительных изменений передаваемых параметров, что в некоторых случаях позволяет снизить нагрузку на сеть передачи данных без потери оперативности передачи критически важных данных.

4. Передача диагностической информации и флагов состояния.

Помимо значений измеренных величин, модули системы передают диагностические сообщения, содержащие бинарные признаки (флаги) состояния контролируемых параметров. Эти сообщения имеют фиксированную периодичность отправки, но могут быть сгенерированы досрочно при возникновении аварийных или предаварийных ситуаций. Получателями таких данных являются ПМ и модули аналогичного типа, что обеспечивает функции самодиагностики, активацию защитных механизмов и сигнализацию аварийных состояний.

5. Мониторинг состояния модулей ССОИ и диагностика их неисправностей.

Для контроля работоспособности отдельных компонентов системы модули периодически отправляют сообщения о своем текущем состоянии, включающие данные о текущем режиме работы, наличии ошибок, критических изменениях в функционировании (исправен модуль или нет). Эти сообщения принимаются ПМ, который фиксирует отказ оборудования, инициируя соответствующую аварийную сигнализацию, если в течение заданного временного интервала от одного из модулей не поступает диагностическая информация.

6. Обмен низкоприоритетной информацией.

Наряду с вышеописанными высокоприоритетными потоками, в системе осуществляется передача второстепенных данных, таких как протоколы событий,

сохранённые осциллограммы сигналов или параметры конфигурации системы. Эти данные не требуют жесткой привязки к циклам синхронизации, что позволяет гибко управлять загрузкой интерфейса межмодульного взаимодействия и минимизировать влияние передачи вспомогательных данных на работу критически важных подсистем.

Все информационные потоки, за исключением первого и шестого, возникают в модульной ССОИ только в её рабочем режиме. Управляющие сообщения ПМ и обмен низкоприоритетной информацией возможны в любом режиме работы.

2.3 Использование интерфейса *CAN FD* и протокола *CANopen FD* для организации межмодульного взаимодействия в системах сбора и обработки информации

2.3.1 Обоснование выбора интерфейса *CAN FD* и протокола *CANopen FD*

Организация эффективного межмодульного взаимодействия в ССОИ ПС, рассмотренной в параграфе 1.4.1, является залогом успешного функционирования всей АСУ ТП, а выбор интерфейса взаимодействия – первоочередной задачей при проектировании ССОИ. В результате анализа наиболее широко известных интерфейсов, с использованием которых возможна организация взаимодействия модулей в ССОИ, таких как *Ethernet* [17], *RS-485* [24], *SPI* [25], *I2C* [53], *Backplane Bus* [54], оптические интерфейсы, было выявлено, что они в той или иной степени обладают существенными недостатками, среди которых можно отметить высокую стоимость реализации, сложность установки и обслуживания, низкую помехозащищенность, низкую пропускную способность, отсутствие механизма обнаружения и предотвращения ошибок, ограниченное количество подключаемых устройств. В значительной степени лишён этих недостатков современный промышленный интерфейс *CAN FD*.

Интерфейс *CAN FD* (*CAN with Flexible Data-rate*) [23] является расширением оригинального интерфейса *CAN* [22] - стандарта промышленной сети,

ориентированного, прежде всего, на объединение в единую сеть различных исполнительных устройств и датчиков.

Интерфейс *CAN* предназначен для надежной передачи и приема данных с датчиков, команд управления и обнаружения ошибок взаимодействия между электронными устройствами, контроллерами. Интерфейс реализован с использованием шинной топологии, режим передачи – последовательный, широкополосный, пакетный.

Основное различие между классическим *CAN* и *CAN FD* заключается в большем объеме информации, передаваемой в сообщении, и в возможности изменения скорости передачи данных в пределах одного сообщения. Используя *CAN FD*, взаимодействующие устройства могут динамически переключаться между различными скоростями передачи и выбирать наиболее подходящую длину сообщений для обеспечения оптимальной загрузки сети передачи данных.

Размер полезной нагрузки в каждом сообщении *CAN FD* увеличен до 64 байт по сравнению с 8 байтами в классическом *CAN* (рис. 2.2).

| | Number of Data Bytes | Data Length Code | | | |
|--------------------------------|----------------------|------------------|------|------|------|
| | | DLC3 | DLC2 | DLC1 | DLC0 |
| Codes in CAN and CAN FD Format | 0 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 0 | 0 | 1 |
| | 2 | 0 | 0 | 1 | 0 |
| | 3 | 0 | 0 | 1 | 1 |
| | 4 | 0 | 1 | 0 | 0 |
| | 5 | 0 | 1 | 0 | 1 |
| | 6 | 0 | 1 | 1 | 0 |
| | 7 | 0 | 1 | 1 | 1 |
| CAN Format | 8 | 1 | 0/1 | 0/1 | 0/1 |
| Codes in CAN FD Format | 8 | 1 | 0 | 0 | 0 |
| | 12 | 1 | 0 | 0 | 1 |
| | 16 | 1 | 0 | 1 | 0 |
| | 20 | 1 | 0 | 1 | 1 |
| | 24 | 1 | 1 | 0 | 0 |
| | 32 | 1 | 1 | 0 | 1 |
| | 48 | 1 | 1 | 1 | 0 |
| | 64 | 1 | 1 | 1 | 1 |

Рисунок 2.2 – Сравнение размера поля данных в сообщениях *CAN* и *CAN FD*

Скорость передачи данных в *CAN FD* до 5-8 раз выше, чем в классическом *CAN* (от 5 до 8 Мбит/с только для полезной нагрузки данных, скорость арбитражных битов по-прежнему ограничена максимальным значением 1 Мбит/с для совместимости с *CAN*). Спецификация протокола *CAN FD* также включает в себя некоторые другие улучшения, такие как более эффективное обнаружение ошибок в полученном сообщении и переключение на более быструю или более медленную скорость передачи данных по мере необходимости.

Использование интерфейса *CAN FD* позволяет обеспечить надёжное межмодульное взаимодействие с высокой скоростью. Технология *bit stuffing* [23] и проверка контрольной суммы обеспечивают высокую достоверность передаваемой по шине информации.

В качестве недостатка *CAN FD* применительно к решаемой задаче можно отметить довольно значительную долю служебной информации [23] в каждом сообщении (не менее 13 %), однако некоторые протоколы более высокого уровня (в частности, *CANopen FD*) в существенной степени устраняют этот недостаток.

2.3.2 Общая характеристика протокола *CANopen FD*

Интерфейс *CAN FD* реализует только два нижних уровня модели взаимодействия *OSI* [55] – физический и канальный. Этого может быть достаточно для решения простейших задач межмодульного взаимодействия. Однако если в состав ССОИ входят десятки модулей, которые обмениваются значениями сотен параметров, возникает необходимость использования протоколов более высокого уровня, обеспечивающих адресацию, синхронизацию, стандартизированный обмен основной и служебной информацией и многие другие функции.

Анализ информационных потоков, выполненный в параграфе 2.2, позволяет говорить о целесообразности применения протокола *CANopen FD* [56] для организации межмодульного взаимодействия в сложных распределённых системах. Данный вывод обусловлен рядом особенностей этого протокола, включая улучшенные характеристики передачи данных и оптимизированные механизмы установления связи. Кроме того, службы протокола дают возможность

без дополнительных усилий реализовать иерархию сообщений и правила обмена ими, обоснованные в параграфе 2.2.

CANopen FD представляет собой усовершенствованный высокоуровневый протокол общего назначения, являющийся модифицированной версией протокола *CANopen* [48]. Он предназначен для обеспечения эффективного взаимодействия устройств в промышленных автоматизированных системах, а также в бортовых микроконтроллерных сетях. Существенным отличием от классического протокола *CANopen* является расширенное поле данных, а также упрощённая процедура установления связи между передающим и принимающим узлом. В результате этого снижается объём служебной информации, передаваемой по сети, что повышает её пропускную способность и снижает нагрузку на вычислительные ресурсы узлов.

Архитектурно *CANopen FD* опирается на физический и канальный уровни модели *OSI* [55], которые реализуются посредством интерфейса *CAN FD* [23]. В рамках стандарта *CANopen FD* определяются схемы адресации, а также несколько специализированных протоколов прикладного уровня, адаптированных к различным профилям устройств. Данные протоколы обеспечивают управление сетью, мониторинг состояния узлов и обмен данными между ними. Кроме того, в *CANopen FD* предусмотрены механизмы сегментации и десегментации сообщений [57], что способствует эффективной передаче крупных блоков информации.

Стандартом [56] предусмотрены следующие основные службы (услуги связи), функционирующие на сеансовом уровне *CANopen FD*:

- служба конфигурирования устройства (*USDO*);
- службы управления сетью (*NMT* и *Heartbeat*);
- служба передачи данных процесса в реальном времени (*PDO*);
- службы синхронизации шины (*SYNC* и *TIME*);
- службы диагностики (*EMCY* и частично *USDO*);
- объектный словарь (*OBD*).

Вначале рассмотрим принцип функционирования протокола *CANopen FD*, а затем более подробно проанализируем назначение и работу каждой службы.

Объектный словарь и средства его описания рассматриваются в отдельном параграфе 2.3.4.

Принцип взаимодействия элементов протокола *CANopen FD* показан на рис. 2.3.

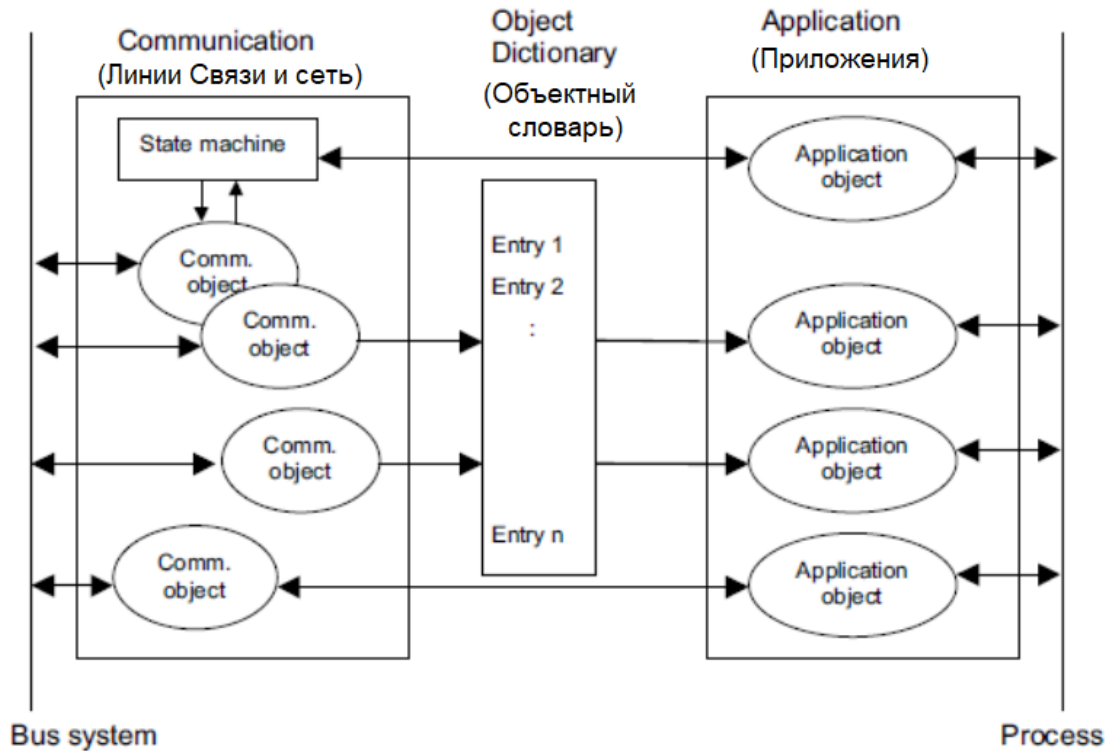


Рисунок 2.3 – Принцип функционирования элементов *CANopen FD*

Прикладной уровень приложений (*Application Layer*) в контексте сети *CANopen FD* представляет собой совокупность механизмов, предназначенных для организации обмена данными в реальном времени, а также для конфигурирования параметров взаимодействия сетевых узлов. Данный уровень обеспечивает логическую структуризацию функциональных компонентов системы посредством выделения отдельных сервисных объектов, обладающих специфическими характеристиками и назначением [58].

Ключевая роль сервисных объектов заключается в предоставлении основного функционала протокола, включая совокупность служб, регулирующих процессы передачи и обработки информации. Структура и спецификация каждой службы регламентируются стандартами, определяющими правила

функционирования сетевого взаимодействия. Взаимодействие приложений осуществляется путём вызова соответствующих служб на прикладном уровне, что обеспечивает согласованность процессов обмена данными. Для инициирования данного процесса объект передаёт данные через сеть *CAN FD*, используя соответствующий сервисный объект другого узла, что гарантирует корректность и надёжность коммуникации [56].

Основополагающим механизмом взаимодействия между приложениями на прикладном уровне являются примитивы службы (*Service Primitives*), которые определяют способы обмена информацией. В рамках данного механизма выделяют четыре ключевых типа примитивов:

- запрос (*request*) – инициируемый службой *application* запрос доступа к другой службе, что позволяет организовать взаимодействие между сетевыми узлами;
- индикация (*indication*) – передаваемое в службу *application* сообщение, информирующее о наступлении внутреннего события, инициированного приложением;
- ответ (*response*) – реакция приложения на ранее полученную индикацию, обеспечивающая согласованность обработки запросов;
- подтверждение (*confirmation*) – сообщение службы *application*, информирующее приложение о результате обработки ранее поданного запроса.

2.3.3 Анализ основных служб протокола *CANopen FD*

Службы конфигурирования устройства (*USDO*)

Спецификация протокола [56] определяет набор специализированных служб *USDO*, предназначенных для эффективного взаимодействия с объектным словарем *CANopen FD*, обеспечивая механизмы чтения и записи данных. Принципиальным аспектом функционирования данных служб является клиент-серверная модель, где инициатива всегда исходит от клиента *USDO*, формирующего запрос на выполнение операции. В отличие от классического обмена данными по *SDO* в *CANopen*, протокол *USDO* дополнительно передает

информацию о получателе, что расширяет функциональные возможности системы. Данный механизм позволяет реализовать различные схемы адресации, включая одноадресную, многоадресную и широковещательную, обеспечивая масштабируемость сети и возможность взаимодействия всех узлов *CANopen FD* в едином информационном пространстве.

Функциональное назначение служб *USDO* охватывает широкий спектр задач, включая конфигурирование устройств, диагностику, а также обмен данными процесса. В рамках протокола *CANopen FD* различают локальные и удаленные службы *USDO*. Локальные службы предназначены для взаимодействия в пределах одного сетевого сегмента, что определяется наличием идентичного сетевого идентификатора. Удаленные службы, напротив, обеспечивают связь между узлами, принадлежащими различным сегментам сети с разными сетевыми идентификаторами.

Системная архитектура *USDO* предполагает обязательное подтверждение выполнения каждой операции. В случае чтения данные передаются обратно клиенту *USDO*, а в случае записи сервер *USDO* генерирует подтверждающее сообщение, сигнализируя о корректном сохранении информации в объектном словаре. Для корректного функционирования клиент *USDO* обязан поддерживать механизм контроля времени ожидания отклика (тайм-аут), в течение которого полученные ответы считаются действительными. Истечение данного интервала приводит к игнорированию ответов, поступивших с задержкой.

Службы *USDO* подразделяются на ускоренные и сегментированные. Ускоренные службы осуществляют передачу данных в рамках единственного пакета, что возможно при небольшом объеме полезной нагрузки. Сегментированные службы, напротив, позволяют передавать большие массивы информации, разбивая их на несколько последовательных пакетов, каждый из которых требует подтверждения. Для оптимизации пропускной способности *CANopen FD* предусматривает поддержку массовых сервисов *USDO*, в которых подтверждается не каждый пакет, а вся группа переданных данных.

При возникновении сбоев в обработке запроса либо клиент, либо сервер *USDO* формируют специальное сообщение об отмене операции, содержащее 4-байтовый код ошибки, позволяющий идентифицировать причину отказа. Концепция *USDO* предусматривает преднастроенную систему широковещательной, многопользовательской и одноканальной связи, устраняя необходимость в ручном назначении идентификаторов доступа. Данный подход упрощает развёртывание сетевых приложений, так как позволяет запускать их исключительно на основе механизмов *USDO*. Дополнительно требуется поддержка стандартных служб *Heartbeat* и *EMCY* для мониторинга работоспособности сети.

Протокол *USDO* поддерживает обмен сообщениями различной длины в зависимости от типа службы. Каждое сообщение включает служебные данные, содержащие спецификатор команды и 24-разрядный адрес параметра объектного словаря (индекс и субиндекс), определяющий, какие данные подлежат чтению или записи. Пример передачи одного сообщения *USDO* показан на рис. 2.4.

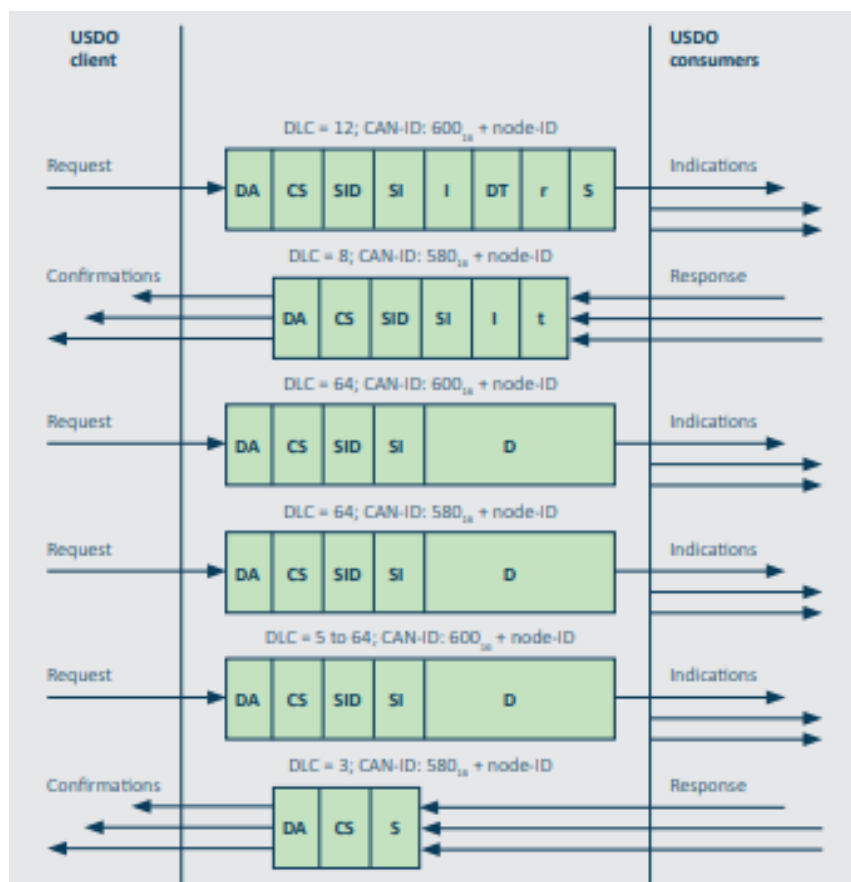


Рисунок 2.4 – Передача одного сообщения службой *USDO* протокола *CANopen FD*

В целом архитектура *USDO* предусматривает поддержку как несегментированных сообщений, размер которых не превышает 64 байта (лимит полезной нагрузки уровня канала в *CAN FD*), так и сегментированных механизмов передачи. При превышении ограничения в 64 байта протокол *CANopen FD* осуществляет разбиение передаваемых данных на последовательные пакеты, которые затем восстанавливаются принимающим узлом. Таким образом, *USDO* обеспечивает эффективное управление данными в распределённых промышленных сетях, повышая надёжность и гибкость взаимодействия устройств.

Службы управления сетью (*NMT* и *Heartbeat*)

В рамках управления сетью *CANopen FD* применяется архитектурный принцип «ведущий узел – ведомый узел» (*master-slave*), обеспечивающий централизованное администрирование узлов посредством ведущего устройства *NMT*. Данное устройство осуществляет контроль состояния ведомых узлов, которые могут находиться в одном из трех predetermined режимов функционирования:

- предоперационный режим: допускается использование всех служб *CANopen FD*, за исключением передачи данных процесса (*PDO*);
- операционный режим: обеспечивается полный доступ ко всем службам *CANopen FD*;
- режим остановки: функциональность ограничивается обработкой сообщений *NMT* и *Heartbeat*, остальные службы недоступны.

Команды управления, инициируемые ведущим устройством *NMT*, подтверждаются посредством передачи сообщений *Heartbeat*, что гарантирует синхронизацию процессов на уровне программных приложений. Формат команды *NMT* имеет двухбайтовую структуру, где один байт содержит код команды, а второй – идентификатор целевого узла. Специальное значение идентификатора узла "0" предназначено для широковещательной команды, требующей исполнения всеми узлами сети. Данная команда передается интерфейсом *CAN FD* с

идентификатором "0", обладающим наивысшим приоритетом в шине передачи данных.

Ответное сообщение *Heartbeat* содержит однобайтовый код текущего состояния устройства (предоперационное, операционное или остановленное) и передается с идентификатором *CAN FD*, скорректированным с учетом идентификатора узла (рис. 2.5). Частота отправки сообщений *Heartbeat* регулируется пользователем через параметр «время производителя *Heartbeat*», настраиваемый в объектном словаре *CANopen FD* [57].

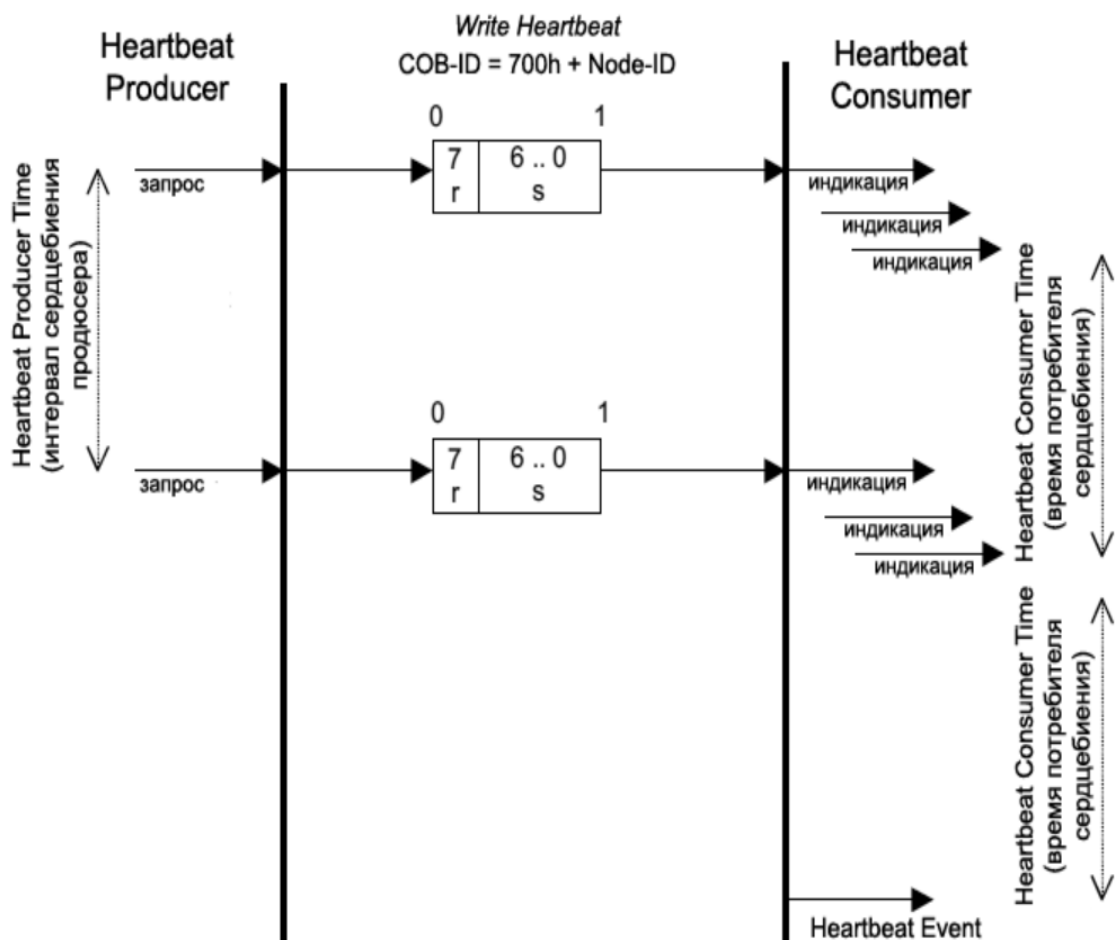


Рисунок 2.5 – Диаграмма работы службы *Heartbeat*

После включения питания узел автоматически переходит в предоперационное состояние, ожидая команды инициализации от ведущего устройства *NMT*. Управляющие команды инициализации различаются по функционалу: первая иницирует восстановление параметров связи до значений по умолчанию, вторая – сбрасывает не только параметры связи, но и параметры

профиля устройства. В обоих случаях узел после сброса возвращается в предоперационный режим.

Режим остановки *NMT* применяется для приостановки обмена *USDO* и *PDO* между устройствами, разрешая при этом обработку только сообщений *NMT* и *Heartbeat*.

Применение службы *Heartbeat* обеспечивает гарантированное подтверждение выполнения команд *NMT*, а также выполняет мониторинг доступности сетевых узлов. В ситуациях, когда обмен *PDO* инициируется только при изменении состояния, получатели сообщений не могут однозначно определить неработоспособность того или иного узла. Регулярные сообщения *Heartbeat* гарантируют контроль состояния всех узлов, устраняя описанную неопределенность.

Ответственность за корректную настройку параметров службы *Heartbeat* несут разработчики модульных систем, обеспечивая надежность работы сети.

Служба синхронизации шины (SYNC)

В рамках сетевого взаимодействия *CANopen FD* предусмотрен специальный механизм синхронизации, обеспечивающий координацию процессов в реальном времени. Данный механизм базируется на передаче сообщений синхронизации (*SYNC*), генерируемых определенным узлом сети, именуемым *SYNC-producer*. Этот узел выполняет функцию централизованного источника временных маркеров, что позволяет другим устройствам синхронизировать внутренние процессы с источником времени и между собой.

Для корректного функционирования службы *SYNC* в сети должны быть сконфигурированы узлы-потребители сообщений синхронизации, обозначаемые как *SYNC-consumer*. Эти узлы воспринимают и интерпретируют *SYNC*-сообщения, регулируя выполнение своих задач в соответствии с поступающими сигналами.

Синхронизационный объект транслируется в широковещательном режиме ко всем узлам сети, однако его обработка осуществляется исключительно устройствами, настроенными на прием таких сообщений. Для обеспечения

приоритетного доступа к шине *CAN FD* сообщениям синхронизации присваивается идентификатор с наивысшим уровнем приоритета, что гарантирует минимальные задержки в их передаче.

Взаимодействие устройств, работающих в синхронном режиме, осуществляется посредством обработки сообщений синхронизации от *SYNC-producer*, что позволяет им согласовывать внутренние процессы и временные метки. Для приложений, требующих более высокой точности временной синхронизации, предусмотрена возможность использования механизма синхронизации с повышенным разрешением, обеспечивающего минимальные временные различия между узлами.

Служба *PDO*

В рамках протокола *CANopen FD* функциональные особенности службы передачи данных процесса (*PDO*) во многом схожи с аналогом в классическом *CANopen*. Основной задачей *PDO* является передача чувствительных ко времени данных процесса, таких как измеряемые параметры и управляющие команды. Данная служба реализует архитектурную модель «поставщик-потребитель» (рис. 2.6).

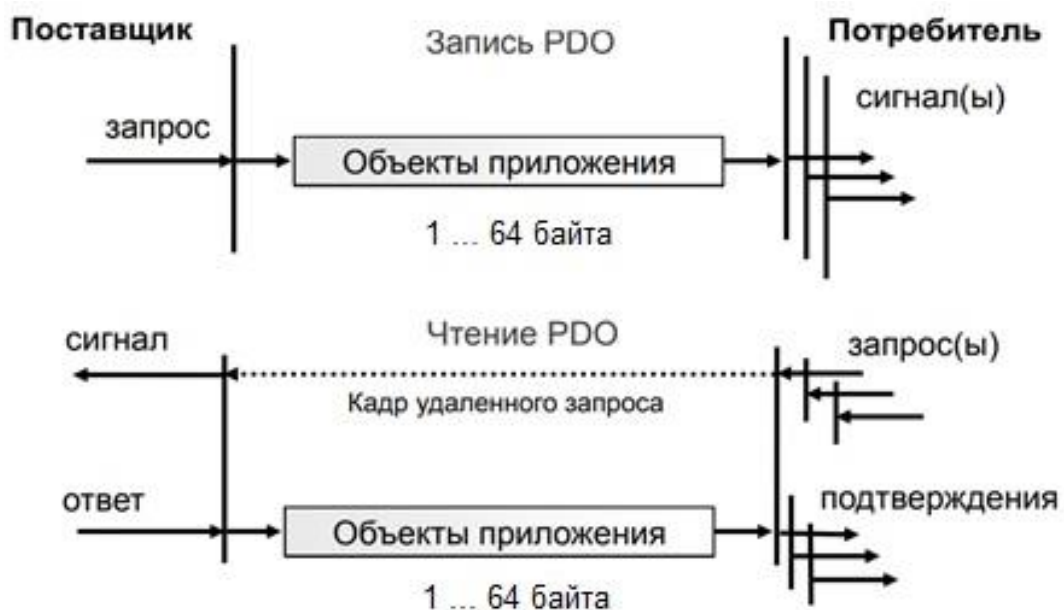


Рисунок 2.6 – Архитектура «поставщик-потребитель» службы *PDO*

Сообщения *PDO* в *CANopen FD* могут содержать до 64 байт данных процесса, которые инкапсулируются в единый кадр данных *CAN FD*. Для обозначения каждого *PDO* используется идентификатор *CAN-ID*, определяемый параметром *COB-ID*, который содержится в объектном словаре *CANopen FD*. *COB-ID* входит в состав параметров связи *PDO*, наряду с параметром типа передачи, определяющим режим отправки сообщений.

Передача *PDO* в *CANopen FD* может осуществляться как в асинхронном, так и в синхронном режимах (рис. 2.7). В асинхронном режиме отправка инициируется изменением значений входных данных либо истечением таймера события, что обеспечивает регулярную передачу. Однако таймер событий действует локально на уровне отдельного устройства и может быть не синхронизирован с остальными узлами сети.

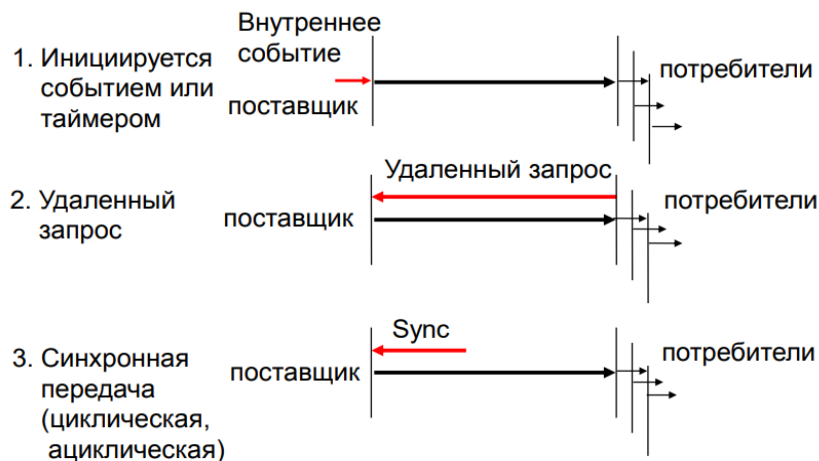


Рисунок 2.7 – Режимы вызова *PDO*

Синхронный режим передачи *PDO* является ключевой функциональной особенностью *CANopen FD*. Он основан на механизме периодической генерации сообщения *SYNC*, которое передается в сети с высоким приоритетом. Сообщение *SYNC* служит триггером для одновременной отправки всех синхронных *PDO*, всеми модулями, обеспечивая временную согласованность устройств.

Для согласования скорости работы разнородных устройств *CANopen FD* предусмотрена возможность настройки передачи синхронных сообщений в ответ на каждое второе, третье и вплоть до 240-го сообщения *SYNC*. Конфигурируя этот

параметр, можно управлять последовательностью запуска различных групп *PDO*, оптимизируя нагрузку на шину.

Еще одной полезной возможностью службы *PDO* является механизм блокировки передачи сообщений. Его суть заключается в настройке временной задержки после передачи, предотвращающей монополизацию сети приоритетными *PDO*. Это обеспечивает более справедливый арбитраж шины и даёт возможность передавать менее приоритетные сообщения. Данный механизм особенно полезен для передачи *PDO*, инициируемой изменением состояния.

Для организации эффективной передачи *PDO* предусмотрен дополнительный набор параметров, управляющих содержанием сообщений. Один *PDO* может включать несколько различных переменных процесса. В объектном словаре определен массив параметров разметки (сопоставления) *PDO*, задающий адреса (16-битный индекс и 8-битный субиндекс) для извлечения передаваемых значений из словаря.

Оптимизация взаимодействия *PDO* в сети *CANopen FD* может осуществляться несколькими методами в зависимости от требований к системе:

- приоритет *PDO* - настройка параметра *COB-ID* с соответствующим значением *CAN-ID* позволяет изменить приоритет передачи;
- связывание *PDO* – указание множества идентификаторов принимаемых *PDO* позволяет организовать подписку только на определенные передаваемые *PDO*, включая поддержку многоадресной и широковещательной передачи;
- планирование *PDO* - настройка типа передачи *PDO* определяет режим отправки (по изменению состояния, периодическая передача или синхронный режим). Аналогично, параметр типа для принимаемых *PDO* задает режим обработки полученных данных (немедленная обработка или синхронизация с *SYNC*);
- сопоставление *PDO* - конфигурирование набора параметров разметки *PDO* определяет, какие данные процесса включены в передаваемые *PDO* или куда записываются данные их принятых *PDO*.

Служба диагностики сети (*EMCY*)

Протоколом *CANopen FD* предусмотрена служба аварийных сообщений (*EMCY*), которая выполняет функцию оперативного оповещения о возникновении неисправностей как на уровне сетевого взаимодействия, так и в контексте функционирования прикладных процессов. Данный механизм реализован в рамках модели взаимодействия «поставщик – потребитель», где устройство-источник инициирует передачу *EMCY*-сообщений при обнаружении критических событий или ошибок.

Формат *EMCY*-сообщения в *CANopen FD* соответствует расширенной структуре, включающей 20 байт данных, и значительно превосходит классический формат *EMCY* (который используется в классическом *CANopen - Legacy EMCY*), т.к. содержит намного больше информации о возникшей неисправности (рис. 2.8).

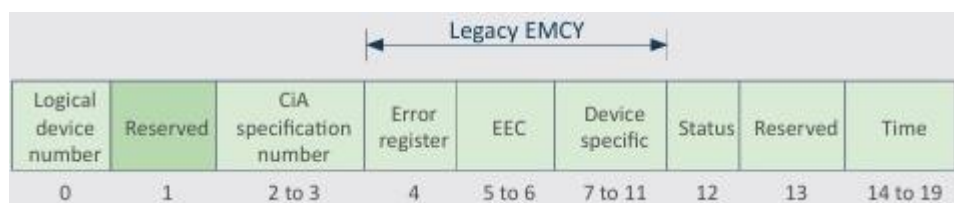


Рисунок 2.8 – Различия между передаваемыми сообщениями *EMCY – CANopen FD* и *Legacy EMCY - CANopen*

Из рис. 2.8 видно, что сообщение *Legacy EMCY* классического *CANopen* передает поля:

- регистр ошибок (*error register*) - 1 байт — индикатор текущего состояния устройства в контексте ошибок;
- код аварийной ошибки (*EEC*) - 2 байта — идентификатор неисправности согласно спецификации *CANopen*;
- специфические признаки устройства (*device specific*) - 5 байт — используются в зависимости от прикладного профиля.

Служба *EMCY* протокола *CANopen FD*, дополнительно к названным полям, передает:

- логический номер устройства (*logical device number*) - 1 байт – связывает сообщение с конкретным узлом сети;
- номер спецификации *CiA* (*CiA specification number*) - 2 байта – указывает номер профиля, которому соответствует устройство;
- состояние ошибки (*status*) - 1 байт – определяет характер неисправности (восстанавливаемая/невосстанавливаемая);
- временная метка (*time*) - 6 байт – фиксирует момент возникновения ошибки.

Механизм обработки ошибок базируется на состоянии регистра ошибок, который представляет собой объект словаря *CANopen FD* и управляет внутренней машиной состояний системы *EMCY*. В случае обнаружения неисправности устройство выполняет запись в регистр ошибок, инициируя таким образом передачу *EMCY*-сообщения для оповещения получателей о текущем состоянии. Если в процессе эксплуатации выявляются дополнительные ошибки, устройство остается в аварийном режиме до полного устранения всех критических событий.

Для корректного функционирования системы должен быть верно настроен механизм обработки *EMCY*-сообщений на стороне потребителей путём определения соответствующих параметров в объектном словаре *CANopen FD*. Конфигурация осуществляется через параметр *Emergency Consumer*, представляющий собой массив, содержащий все *COB-ID* аварийных сообщений, подлежащих обработке. Следует учитывать, что *COB-ID* представляет собой 32-битное значение, включающее идентификатор *CAN-ID* и 3 управляющих бита, определяющих особенности передачи данных.

В типовом варианте аварийное сообщение передается в одном кадре *CAN FD*, используя идентификатор с высоким приоритетом. Реакция системы на полученные *EMCY*-сообщения зависит от конкретной реализации.

Для гибкости конфигурации система позволяет изменять значение *COB-ID EMCY* путем записи в словарь объектов *CANopen FD* посредством службы *USDO*. Это обеспечивает адаптацию службы *EMCY* под конкретные требования прикладной системы.

2.3.4 Объектный словарь и средства описания устройств

Одним из основных понятий *CANopen FD* является объектный словарь (*Object Dictionary, OBD*), который представляет собой таблицу и выполняет функцию унифицированного интерфейса обмена данными, обеспечивая стандартизированный доступ к параметрам конфигурации, диагностической информации и текущим операционным данным устройства.

Концепция объектного словаря базируется на использовании адресации объектов, организованных в соответствии с двухуровневой иерархической структурой. Для идентификации объектов применяются два параметра:

- индекс (*index*) – уникальный идентификатор объекта в словаре, эквивалентный адресу этого объекта;
- субиндекс (*subindex*) – дополнительный идентификатор, используемый для доступа к компонентам составных объектов, таких как массивы или записи.

Каждая запись словаря (объект) имеет 16-разрядный индекс; в рамках одного объекта может быть определено до 256 подобъектов, обращение к которым выполняется с использованием восьмиразрядного субиндекса.

Объектный словарь каждого узла сети *CANopen FD* является его неотъемлемой частью и содержит набор данных, необходимых для функционирования устройства в сети. В зависимости от типа информации объекты словаря могут содержать текущие значения дискретных и аналоговых входов\выходов, управляющие параметры выходных сигналов, конфигурационные данные, определяющие поведение устройства и т.д.

Каждая запись в *OBD* характеризуется набором атрибутов, основными из них являются: тип объекта – классифицирует объект (простое значение, массив, запись); имя объекта – описание объекта, облегчающее его идентификацию; тип данных – определяет формат хранимых данных (логический, целочисленный, число с плавающей точкой, символьный и т. д.); атрибуты доступа – устанавливают права на чтение и запись (только чтение – *RO*, только запись – *WO*, чтение/запись – *RW*, неизменяемый параметр – *Const*).

Операции чтения и записи параметров *OBD* позволяют осуществлять обмен данными между устройствами, обеспечивая координацию их работы. Например, одно устройство может передать управляющий сигнал, изменяя значение соответствующего объекта в словаре другого узла, что приведет к модификации состояния исполнительного механизма.

Считывание значений из объектного словаря позволяет получать актуальную информацию о состоянии устройства, включая накопленные диагностические данные. Для реализации взаимодействия используются рассмотренные ранее службы, такие как *USDO* для асинхронного обмена данными и *PDO* для высокоскоростной передачи информации в реальном времени.

Для стандартизации описания объектных словарей в *CANopen FD* используются специальные файлы *Electronic Data Sheet (EDS-файлы)* [59], представляющие из себя текстовые документы, содержащие структурированную информацию о конфигурации устройства, включая список поддерживаемых объектов, диапазоны допустимых значений параметров, значения по умолчанию, типы данных.

В дополнение к *EDS*-файлам используются файлы конфигурации устройств (*Device Configuration File, DCF-файл*) [59]. Они содержат конкретные параметры настройки устройства и отличаются от *EDS* тем, что описывают не общие характеристики оборудования, а специфическую конфигурацию, применяемую в заданной системе. Однако, использование стандартных средств протокола *CANopen FD (EDS- и DCF-файлов)* при описании системы может стать причиной трудно выявляемых проблем. Речь о причинах и возможных проблемах при составлении описания системы пойдет в параграфе 2.4.

2.4 Представление межмодульных информационных связей и разработка метода описания системы

Как было сказано в параграфе 2.3.2, для описания модульной системы на базе протокола *CANopen FD* имеются стандартные средства: *EDS*- и *DCF*-файлы. Основной проблемой такого описания модульной системы является независимость

разметок передаваемых и принимаемых *PDO* (*TPDO* и *RPDO* соответственно) во взаимодействующих модулях.

Пусть, например, в *PDO* первого типа с заданной периодичностью передаются текущие измеренные значения входных сигналов, *PDO* второго типа содержит признаки состояния модуля и передаётся только при их изменении. Содержание каждого *PDO* задаётся параметрами его разметки (*PDO Mapping*, рис. 2.9) при настройке модулей.

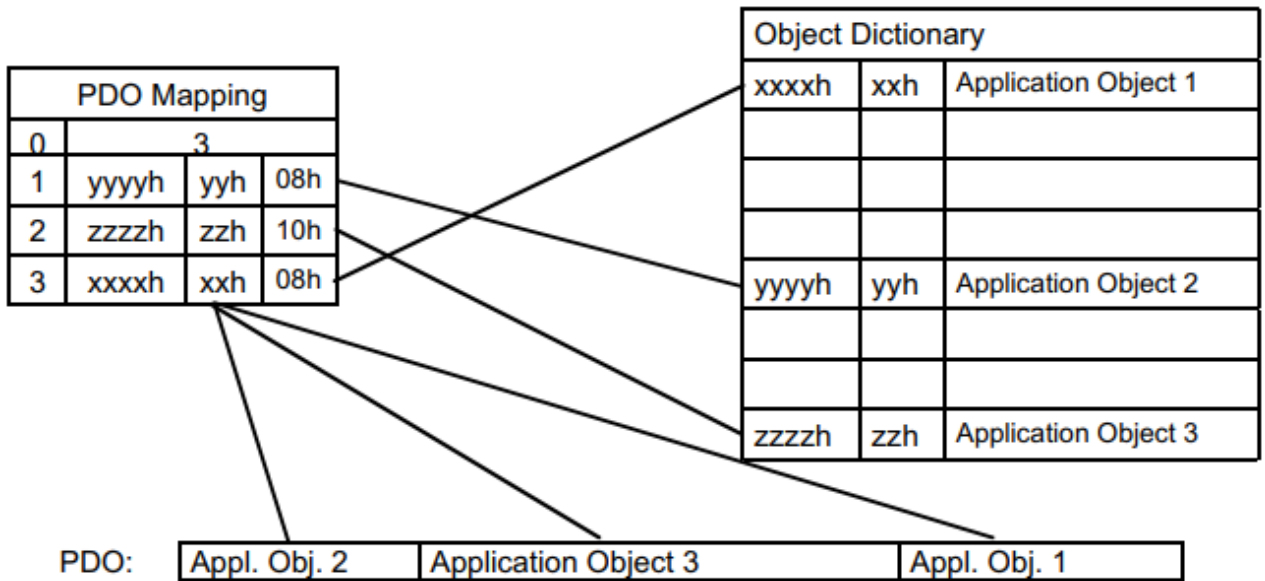


Рисунок 2.9 – Разметка *PDO* (*PDO Mapping*)

Набор этих параметров одинаков для *PDO*, формируемых некоторым модулем (*TPDO*), и для *PDO*, принимаемых модулем (*RPDO*), однако их назначение несколько отличается. Разметка *TPDO* определяет, содержимое каких объектов словаря и в каком порядке следует расположить в передаваемом сообщении. Разметка *RPDO* показывает, данные какого размера в какой последовательности нужно извлечь из принятого сообщения и в какие объекты словаря их записать. На рис. 2.10 показан пример передачи значений фазного напряжения, измеренных модулем 1, в процессорный модуль системы (модуль *N*) с использованием службы *PDO*.

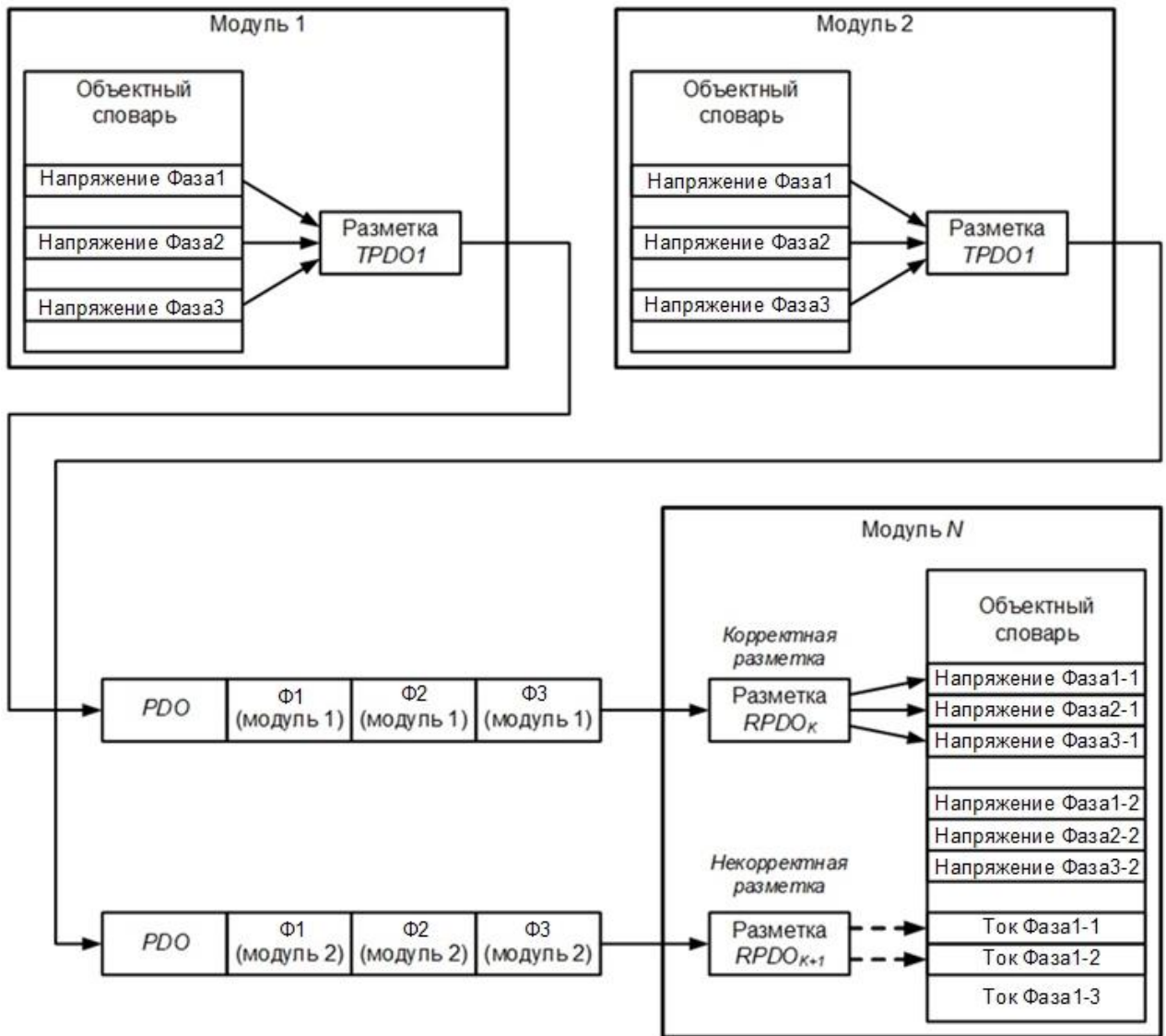


Рисунок 2.10 – Пример организации межмодульного взаимодействия в ССОИ ПС с использованием службы *PDO*

Описанный механизм обмена данными обладает чрезвычайной гибкостью, но при проектировании ССОИ в процессе её адаптации к конкретному объекту управления он требует настройки с учётом специфики этого объекта. Поскольку количество и типы модулей, используемых в системе, а также набор контролируемых параметров для разных объектов управления могут отличаться, разработчики должны указать, какие модули конкретного варианта системы являются источниками некоторых наборов данных, а какие – приёмниками, то есть задать информационные связи между объектными словарями модулей. В процессе настройки этих связей перед вводом ССОИ в эксплуатацию возможно возникновение трудно выявляемых ошибок. Дело в том, что в соответствии со

стандартом [59] разметки *TPDO* и *RPDO* взаимодействующих модулей настраиваются независимо в процессе создания отдельных *DCF*-файлов конфигурации для каждого модуля. Это может привести к ошибочной ситуации, изображённой на рис. 2.10 для модулей 2 и N . Пусть, например, *TPDO1* модуля 1 настроено для передачи текущих значений фазного напряжения в некоторой точке измерения (Фаза1, Фаза2, Фаза3), а *TPDO1* модуля 2 – для передачи аналогичных параметров, измеренных в другой точке. Эти данные передаются службой *PDO* в сеть и принимаются модулем N . Если разметка *RPDO* принимающего модуля настроена правильно ($RPDO_k$), то она помещает полученные параметры напряжения в соответствующие объекты словаря. Если же при настройке *RPDO* допущена ошибка ($RPDO_{k+1}$), то часть параметров напряжения, измеренных модулем 2, запишется в объекты словаря, предназначенные для хранения других величин, а часть и вовсе будет потеряна.

Проблему можно решить путём дополнения описания ССОИ, содержащего сведения о конфигурации модулей, явным описанием информационного взаимодействия между модулями. Для этого в описании следует определить некие информационные объекты, соответствующие *PDO* и включающие в себя все необходимые сведения о содержании, источнике и приёмниках каждого сообщения.

Для формирования структуры описания модульной ССОИ, основанной на протоколе *CANopen FD*, разработана схема информационных потоков (рис. 2.11). В соответствии с ней ССОИ представлена в виде множества модулей $\{m_k\}: k = \{1, 2, \dots, M\}$, где M – общее количество модулей. Параметры, измеряемые или вычисляемые k -м модулем, образуют множество X_k , параметры, получаемые от других модулей – множество Y_k . Каждый k -й модуль с помощью сообщений *PDO* передаёт другим модулям параметры из собственного множества X_k , а принимающие модули с номерами $i = \{1, 2, \dots, M\}, i \neq k$ в соответствии со своими настройками могут записать полученные данные в множества Y_i . Каждое сообщение имеет приоритет, определяемый классом сообщения $h \in \{1 \dots H\}$, где H – количество классов. Приоритет тем выше, чем меньше номер класса. Сообщения с

высшим приоритетом передаются и, соответственно, доставляются в первую очередь.

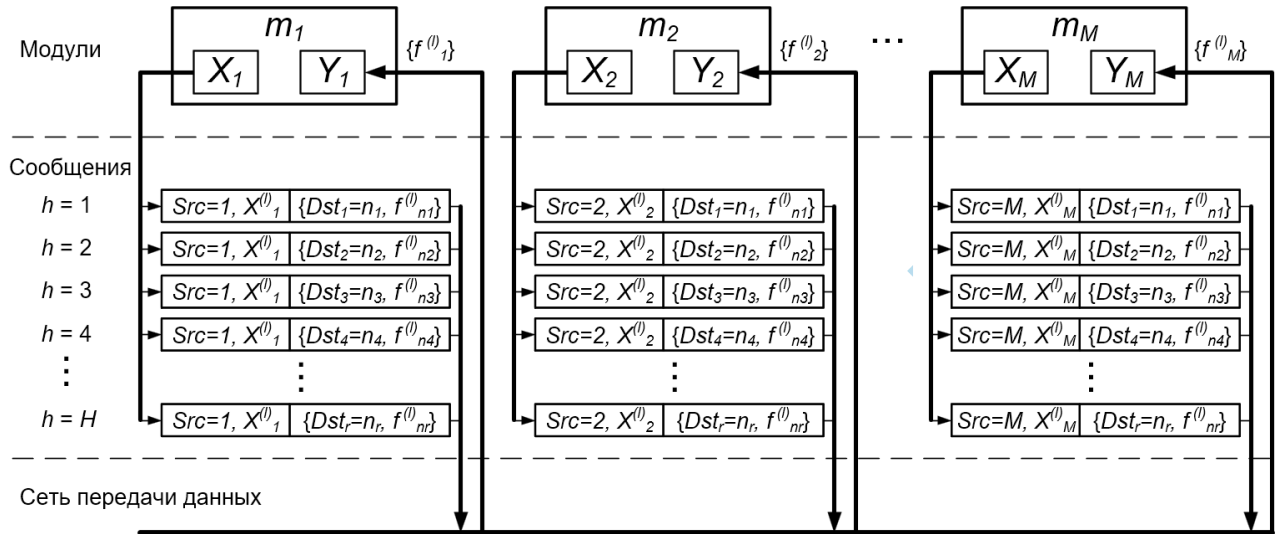


Рисунок 2.11 – Схема информационных потоков в модульной ССОИ

Таким образом, сообщение $l_{h,k}$, передаваемое k -м модулем и относящееся к классу h , можно описать информационным объектом, включающим следующие сведения:

- номер модуля-источника сообщения Src ($Src = k$);
- множество параметров $X_k^{(l)} \subset X_k$, содержащихся в сообщении;
- список модулей-приёмников, в котором для каждого r -го модуля в поле Dst указан его номер в системе n_r , а также задано соответствие $f_{nr}^{(l)}: X_k^{(l)} \rightarrow Y_{nr}$, отображающее полученные в сообщении данные на множество Y_{nr} .

Кроме перечисленных сведений в описание сообщения включаются и другие предусмотренные стандартом [59] параметры PDO , которые не имеют прямого отношения к разработанному методу и в связи с этим на рис. 2.11 не показаны.

На основе разработанной схемы создан и подробно изложен автором в [60] метод описания ССОИ, рассматривающий систему как совокупность модулей, взаимодействующих путём обмена сообщениями. На верхнем уровне описания определены две группы параметров (рис. 2.12): параметры модулей ($Nodes$) и параметры сообщений ($Messages$).

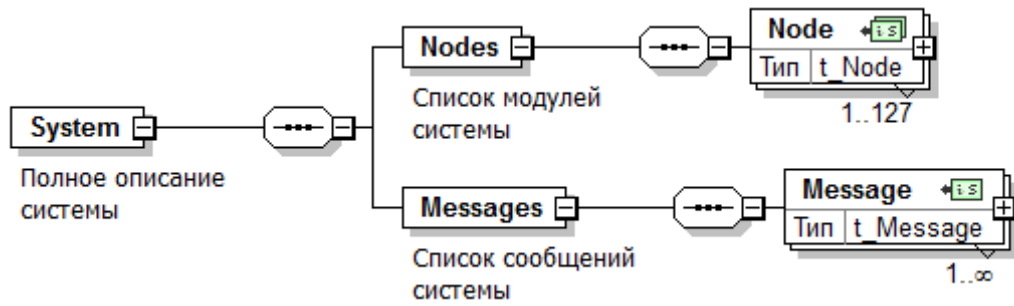


Рисунок 2.12 – Общая структура описания модульной ССОИ

Список модулей *Nodes* в соответствии с [56] не может содержать более 127 записей. Предельное количество записей в списке *Messages*, то есть максимально допустимое число всех возможных типов сообщений в системе, в явном виде не задано, поскольку индивидуальные ограничения накладываются на количества типов сообщений, формируемых каждым модулем в отдельности. Это естественным образом ограничивает размер списка *Messages*.

Описания модулей m_k , содержащиеся в списке *Nodes*, имеют структуру, показанную на рис. 2.13. Для каждого модуля задаются некоторые общие параметры (*attributes*) и описывается объектный словарь (*OBD*). В числе общих параметров следует особо отметить идентификатор (адрес) узла в сети *NodeID* и уникальный индекс модуля *idx*, с использованием которого другие объекты описания ссылаются на модуль. Применение в качестве индекса параметра *NodeID* признано нецелесообразным, поскольку он может изменяться в процессе настройки, что повлечёт за собой необходимость выявления и коррекции множества ссылок. Оставшиеся общие параметры являются вспомогательными, их перечень при необходимости может дополняться и изменяться.

Список *OBD* содержит описания отдельных объектов словаря *Object* типа $t_ObdEntry$. Для каждого объекта задаётся уникальный индекс *idx*, необходимый для ссылок на этот объект из других частей описания, а также параметры в соответствии с [41]: индекс и субиндекс (*ObdIndex*), разрядность данных в битах (*BitSize*), режим доступа (*Access*), признак возможности приёма или передачи в *PDO* (*PdoMap*) и ряд других, которые для простоты на схеме не показаны. Множества вычисляемых и принимаемых величин (X_k и Y_k соответственно)

задаются неявно на основе значений параметров *PdoMap* и *Access* всей совокупности объектов модуля. Прежде всего, для элементов каждого из множеств должна быть разрешена передача службой *PDO* (*PdoMap = true*). Из всей совокупности этих элементов в множество X_k входят объекты с режимом доступа «Только чтение» (*Access = ro*), а в множество Y_k – с режимом доступа «Запись» или «Чтение и запись» (*Access = {wo | rw}*). Такой способ задания X_k и Y_k предотвращает избыточность описания и, как следствие, исключает возможность несогласованности его элементов.

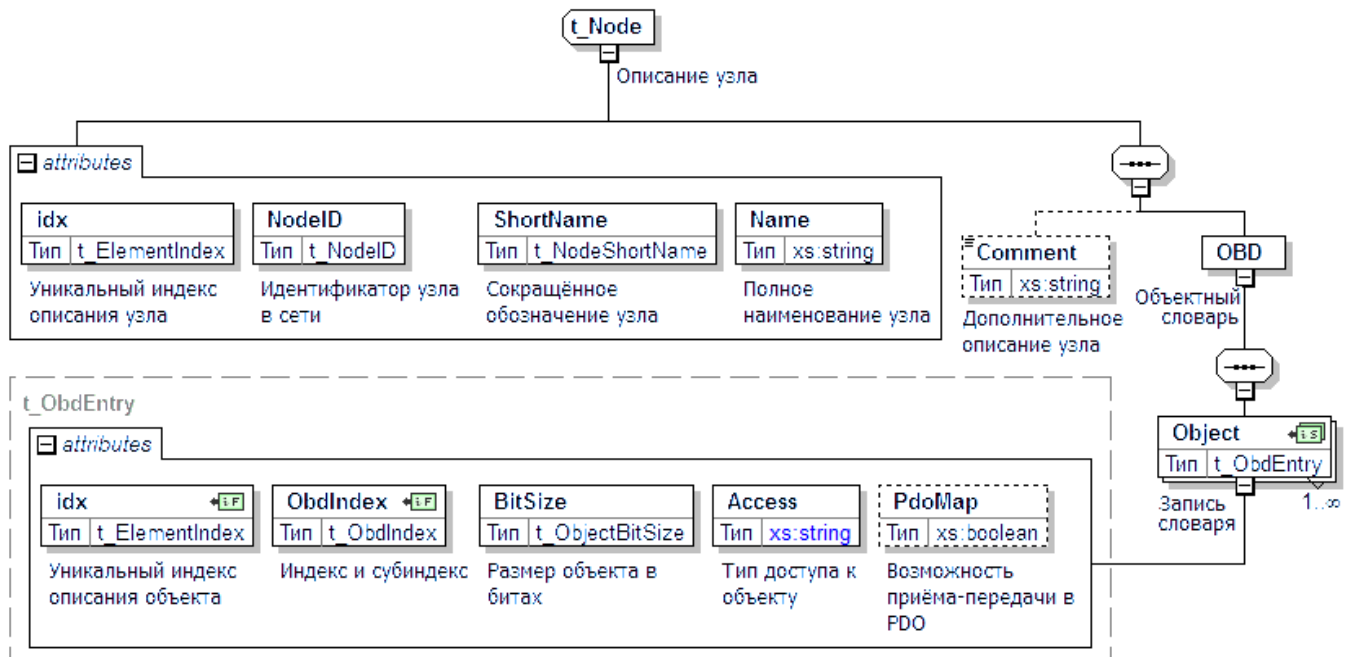


Рисунок 2.13 – Описание модуля ССОИ

Для описания сообщений ССОИ используется структура данных, упрощённая схема которой показана на рис. 2.14.

Эта структура, аналогично структуре *t_Node*, содержит набор общих параметров (*attributes*), а также описание источника и приёмников сообщения (*Source* и *DestList* соответственно). Назначение параметра *idx* аналогично описанному ранее для других структур, приоритет (класс) сообщения *h* определяется параметром *Priority*.

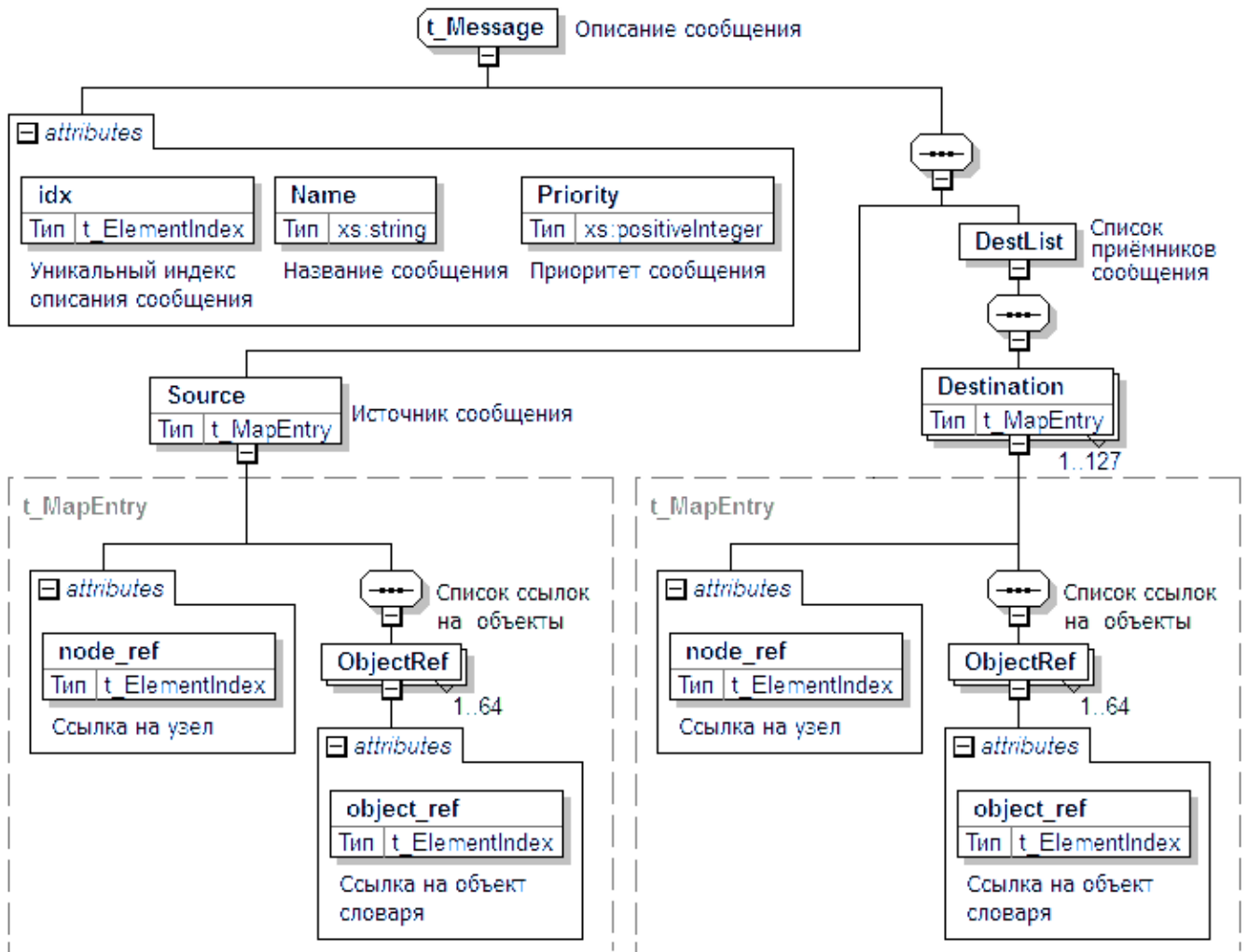


Рисунок 2.14 – Описание сообщения

Источник *Source* и список приёмников *DestList* сообщения описываются одинаковыми структурами типа *t_MapEntry*. Различие заключается лишь в том, что любое сообщение всегда имеет только один источник, а приёмников может быть несколько. В структуре *Source* указывается индекс модуля, передающего сообщение, и список ссылок на объекты из словаря этого модуля, определяющий содержание сообщения, то есть множество $X_k^{(l)}$. Список *DestList* содержит одну или несколько аналогичных структур, в каждой из которых указан индекс принимающего модуля и список ссылок на объекты из его словаря, в которые будут записаны данные, содержащиеся в сообщении. Отображение $f_{nr}^{(l)}: X_k^{(l)} \rightarrow Y_{nr}$ задаётся порядком перечисления индексов объектов в списках *ObjectRef* источника и приёмников.

Некоторые параметры объектов словаря, оговоренные в [57], но не относящиеся непосредственно к разработанному методу – такие, как режимы передачи, значения по умолчанию и ряд других – в приведённом описании для простоты не упомянуты, однако при практической реализации метода указываются в соответствующих структурах данных.

Разработанный метод описания ССОИ реализован автором с использованием языка *XML* [61]. *XML* - расширяемый язык разметки, который позволяет определять и хранить данные унифицированным способом. *XML* обеспечивает обмен информацией между вычислительными системами или отдельными приложениями, такими как веб-сайты, базы данных и т.п. Файл *XML* представляет из себя иерархический набор описаний объектов, что позволяет легко представить в нём структуры, показанные на рисунках 2.12, 2.13, 2.14.

Использование языка *XML* обеспечивает следующие преимущества.

1. Простота модификации. *XML*-описание хранится в текстовом формате и имеет иерархическую структуру, доступную для восприятия человеком. За счёт этого небольшие изменения описания можно выполнить любой программой для редактирования текста без специального ПО.

2. Расширяемость описания. Язык *XML* допускает использование произвольного количества тегов и атрибутов, определяемых разработчиком, в том числе и внутри уже существующих структур. За счёт этого к описанию любого объекта ССОИ можно добавить необходимое количество параметров с сохранением как минимум прямой совместимости исходных и дополненных описаний. Обратной совместимости можно добиться путём использования значений по умолчанию для элементов, отсутствующих в исходном описании.

3. Универсальность использования описания. Поскольку язык *XML* стандартизирован и широко распространён, для него существует множество средств разработки и поддержки, что позволяет использовать для создания описаний ССОИ не только специальное, но и универсальное ПО (например, *JSON* и *XML Editor* [62]).

4. Целостность структуры описания. Структуру *XML*-документа можно описать путём составления так называемой схемы документа, а впоследствии автоматически проверить на соответствие этому описанию при помощи широко распространённых стандартных средств.

Разработанный метод описания ССОИ имеет ряд преимуществ по сравнению со стандартным методом, использующим *EDS* и *DCF* файлы и задающим межмодульное взаимодействие путём создания независимых конфигураций для каждого модуля системы.

1. Взаимодействие модулей описывается явным образом в едином информационном объекте (*XML*-файле) и наглядно связывает источник и приёмники данных, что упрощает создание, модификацию и анализ описания системы.

2. При создании описания системы возникает возможность выявления сообщений, которые передаются, но не принимаются ни одним модулем (с пустым списком приёмников). Они увеличивают нагрузку на сеть, но не несут полезной информации.

3. Исключаются ситуации, когда приёмники будут настроены на получение данных, не имеющих источника. Это относится как к сообщениям в целом (сообщения с пустым полем *Source* недопустимы), так и к отдельным параметрам (списки *ObjRef* источника и приёмников должны соответствовать друг другу по количеству и типам данных, а также ссылаться на существующие объекты словарей).

4. Обеспечивается возможность выявления объектов словаря, значения которых должны быть получены от других модулей, но передача этих значений отсутствует.

5. Исключается несоответствие типов данных: для этого в описании каждого сообщения отображения $f_{nr}^{(l)}: X_k^{(l)} \rightarrow Y_{nr}$ проверяются на совместимость передаваемых и принимаемых данных;

6. На основе единого описания ССОИ возможно автоматизированное создание согласованных друг с другом стандартных файлов конфигурации

DCF [59] для отдельных модулей, что при необходимости позволяет использовать для настройки системы ПО сторонних производителей. Возможно решение и обратной задачи: автоматическое формирование *XML*-описания ССОИ на основе имеющихся отдельных *DCF*-файлов с проверкой его полноты и согласованности.

7. Описание ССОИ может быть использовано для автоматизации формирования фрагментов исходного текста ПО модулей – в первую очередь структур, описывающих объектные словари.

С использованием метода описания модульной ССОИ автором разработана на языке *XSD* [64] схема проверки *XML*-описания, обеспечивающая контроль полноты и непротиворечивости описания каждой конкретной системы. На схему получено свидетельство о регистрации ПО [63] (Приложение Д). С использованием этой схемы могут быть автоматически выявлены следующие ошибки и потенциальные проблемы настройки межмодульного взаимодействия:

- сообщения, не имеющие источника;
- сообщения, не имеющие получателей;
- сообщения без содержимого (пустые);
- передача одних и тех же параметров в нескольких сообщениях;
- передача отсутствующего (несуществующего) параметра;
- параметры, доступные для передачи, но не передаваемые ни в одном сообщении;
- отсутствие источника у принимаемого параметра;
- несоответствие типов передаваемого и принимаемого параметров.

Опыт применения схемы показал, что благодаря описанным решениям все ошибки несогласованной или неполной настройки отдельных модулей и, как следствие, возможность потери или искажения данных по этой причине выявляются и устраняются ещё на этапе подготовки конфигурации ССОИ в условиях предприятия-изготовителя, а не в процессе пусконаладочных работ, как это было ранее до внедрения разработанного метода.

2.5 Выводы по главе 2

Во второй главе диссертации получены следующие основные результаты.

1. Разработана номенклатура модулей ССОИ ПС, позволяющая адаптировать систему к определенным условиям эксплуатации посредством установки необходимого количества модулей, соответствующих разработанной номенклатуре.

2. Выполнен анализ информационных потоков в современных ССОИ. Рассмотрены информационные потоки в порядке убывания их важности для функционирования системы.

3. Рассмотрены наиболее широко распространённые интерфейсы, предоставляющие возможность организации взаимодействия модулей в ССОИ, выявлены их преимущества и недостатки. Предложено и обосновано использование интерфейса *CAN FD* при организации межмодульного взаимодействия в ССОИ.

4. На основе анализа информационных потоков в ССОИ сделан вывод о целесообразности использования протокола общего назначения *CANopen FD* для реализации межмодульного взаимодействия. Описаны и проанализированы основные службы протокола.

5. Выявлена проблема описания модульной системы с использованием стандартных средств протокола *CANopen FD*, обусловленная независимым определением одного и того же сообщения в передающем и принимающих модулях, вследствие чего становится возможной несогласованная настройка модулей.

6. Разработан метод описания ССОИ, рассматривающий систему как совокупность модулей, для которых явным образом задано информационное взаимодействие. Показаны преимущества и практическая значимость метода. С использованием метода разработана схема проверки *XML*-описания, при помощи которой автоматически выявляются ошибки и потенциальные проблемы настройки межмодульного взаимодействия. Получено свидетельство о регистрации ПО.

3 РАЗРАБОТКА МЕТОДОВ И МОДЕЛЕЙ ДЛЯ ОЦЕНКИ ПАРАМЕТРОВ МЕЖМОДУЛЬНОГО ВЗАИМОДЕЙСТВИЯ В СИСТЕМАХ СБОРА И ОБРАБОТКИ ИНФОРМАЦИИ

Как показано в параграфе 1.3.3, ССОИ имеют, как правило, модульную структуру (рис. 3.1), обеспечивающую возможность адаптации системы к широкому кругу объектов за счёт добавления необходимого числа модулей ввода-вывода. В параграфе 1.4.3 показано, что одним из существенных факторов, ограничивающих возможности такой адаптации, является нагрузка y сети передачи данных (СПД) между модулями, которая определяется интенсивностями λ_i потоков сообщений от модулей ССОИ и интенсивностью их обслуживания (передачи) μ :

$$y = \frac{1}{\mu} \sum_{i=0}^N \lambda_i .$$

В случае, если нагрузка превысит единицу, то есть суммарная интенсивность потоков сообщений от модулей станет больше интенсивности их передачи по СПД, возможна несвоевременная доставка или потеря некоторой части сообщений и, как следствие, некорректная работа ССОИ.

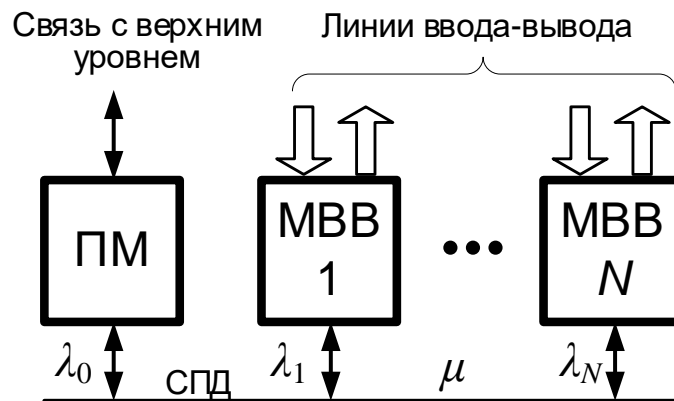


Рисунок 3.1 – Модульная структура ССОИ

ПМ – процессорный модуль; МВВ – модули ввода-вывода; СПД – сеть передачи данных, λ_i – интенсивность потока сообщений i -го модуля, μ – интенсивность передачи сообщений по СПД.

Интенсивности λ_i потоков сообщений от каждого модуля существенно зависят от задач, решаемых конкретной ССОИ, и от особенностей объекта наблюдения (управления). Интенсивность передачи сообщений μ определяется

выбранным интерфейсом и протоколом взаимодействия и для шинной топологии сети, показанной на рис. 3.1, обычно может быть вычислена как $\mu = l / b$, где b – время передачи одного сообщения. Таким образом, наличие возможности определения величины b позволяет на самых ранних этапах проектирования ССОИ оценить нагрузку СПД и обоснованно выбрать тип и параметры интерфейса, обеспечивающие его работу без перегрузки. Следовательно, необходимо разработать метод оценки времени передачи сообщений b выбранного интерфейса *CAN FD*.

3.1 Разработка метода оценки времени передачи сообщений интерфейса *CAN FD*

3.1.1 Анализ структуры сообщений

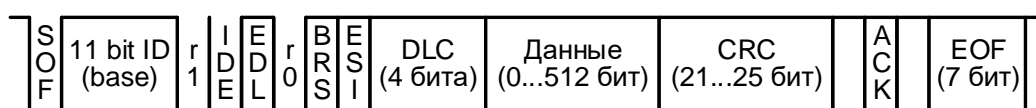
Одним из широко распространённых, высоконадёжных и хорошо зарекомендовавших себя интерфейсов для организации межмодульного взаимодействия является интерфейс *CAN*, метод оценки времени доставки сообщений для которого разработан и описан в [65, 66]. Однако постоянно возрастающие требования к объёму информации, регистрируемой и обрабатываемой ССОИ, и, соответственно, к пропускной способности СПД, привели к созданию более совершенной версии интерфейса *CAN FD*, обеспечивающей в некоторых случаях более чем восьмикратное увеличение эффективной скорости передачи данных [23]. Несмотря на общее сходство этих интерфейсов, структура сообщений *CAN FD* и способ их передачи заметно отличаются от *CAN* и требуют использования иных методов оценки времени передачи.

В [67, 68] предложены методы оценки времени передачи сообщений для интерфейсов *CAN* и *CAN FD*, однако они не рассматривают расширенный формат сообщений, а при вычислении количества дополнительных битов (о них будет сказано далее) не принимают во внимание наличие битов-разделителей с фиксированными значениями. Кроме того, авторы метода [68] основывались на

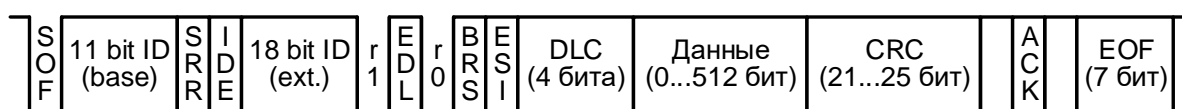
исходном описании интерфейса *CAN FD*, в которое в процессе стандартизации были внесены изменения [69, 70], влияющие на время передачи. Перечисленные недостатки явились поводом для разработки автором более совершенного и универсального метода, описанного далее.

В формате *CAN FD* возможна передача сообщений трёх типов: *Data Frame*, *Error Frame* и *Overload Frame*, причём для обмена данными используется только *Data Frame*, а сообщения двух оставшихся типов применяются для сигнализации о нештатных ситуациях [23]. Поскольку в правильно спроектированной системе нештатные ситуации возникают крайне редко, под временем передачи сообщения *b* в дальнейшем будем подразумевать время передачи *Data Frame*: T_{DF} .

Для оценки времени T_{DF} проанализируем формат *Data Frame*. В соответствии с [23] интерфейс *CAN FD*, аналогично классическому *CAN*, реализует два варианта *Data Frame*: стандартный (*Standard*, рис. 3.2а) и расширенный (*Extended*, рис. 3.2б). Как видно из рисунка, сообщения содержат определенный набор полей. Полное описание и особенности использования каждого из них приведены в [23], в данном параграфе рассмотрены лишь факторы, влияющие на время передачи сообщения.



а) - стандартный формат



б) - расширенный формат

Рисунок 3.2 – Форматы сообщений *CAN FD* типа *Data Frame*

Формат сообщения задается полем *IDE*: для стандартного формата его значение равно нулю, для расширенного – единице. Все поля, за исключением полей данных и контрольного кода (*CRC*), имеют фиксированную длину, показанную на рис. 3.2 (по умолчанию подразумевается длина 1 бит). Размер поля данных определяется содержимым поля *DLC*, которое может принимать значения

от 0 до 15. При $DLC \in \{0, 1, \dots, 8\}$ количество байтов в поле данных равно значению DLC . Если же $DLC \in \{9, 10, \dots, 15\}$, то значениям в этом диапазоне соответствуют следующие количества байтов в поле данных: 12, 16, 20, 24, 32, 48, 64.

3.1.2 Разработка метода оценки времени передачи сообщений

В процессе разработки метода прежде всего была выполнена оценка количества бит в поле данных. Несложно заметить, что функция $N_{DATA}(DLC)$, связывающая количество бит в поле данных с целочисленным значением поля DLC , является кусочно-линейной:

$$N_{DATA}(DLC) = \begin{cases} 8DLC, & 0 \leq DLC \leq 8 \\ 32DLC - 192, & 9 \leq DLC \leq 12 \\ 128DLC - 1408, & 13 \leq DLC \leq 15 \end{cases} . \quad (3.1)$$

Размер поля CRC зависит от размера поля данных: при $DLC \leq 10$ используется 17-разрядный контрольный код, при $DLC > 10$ – 21-разрядный. Перед контрольным кодом передаётся 3-разрядный счётчик автоматически вставленных битов (о них будет сказано далее), дополненный битом контроля чётности, после контрольного кода – бит $CRC Delimiter$. Следовательно, суммарное количество битов в поле CRC может быть равно 22 или 26. Кроме того, в $CAN FD$ по сравнению с CAN изменены правила автоматической вставки битов в поле CRC : теперь дополнительные биты находятся на фиксированных позициях. Первый дополнительный бит вставляется непосредственно перед полем CRC , остальные – после каждого четвёртого проверочного бита. Таким образом, при использовании $CRC-17$ общая длина поля CRC будет равна 28 битам (22 основных и 6 дополнительных), а при использовании $CRC-21$ – 33 битам (26 основных и 7 дополнительных):

$$N_{CRC}(DLC) = \begin{cases} 28, & DLC \leq 10 \\ 33, & DLC > 10 \end{cases} . \quad (3.2)$$

При передаче нескольких сообщений типа *Data Frame* друг за другом между ними вставляется разделитель *Interframe Space (IFS)* размером 3 бита, который также следует учитывать при оценке времени передачи.

В отличие от классического *CAN*, различные части сообщения *CAN FD* могут передаваться с различной битовой скоростью. Номинальная скорость $Baud_N$ соответствует спецификации *CAN* и может достигать 1 Мбит/с. С этой скоростью передаётся начало сообщения до поля *BRS*, в котором содержится признак изменения скорости. При $BRS=0$ остаток сообщения также передаётся на номинальной скорости. Если же $BRS=1$, то поля от *ESI* до *CRC* включительно передаются на альтернативной скорости (скорости данных $Baud_D$), а остаток сообщения – на номинальной. Переключение с номинальной скорости $Baud_N$ на скорость данных $Baud_D$ выполняется в процессе передачи бита *BRS*, обратное переключение – в процессе передачи бита *CRC Delimiter*, завершающего поле *CRC*. Значения $Baud_N$ и $Baud_D$ выбираются на этапе проектирования сети *CAN FD* и должны быть одинаковы для всех взаимодействующих узлов.

Поля *SOF*, *11-bit ID*, *r1*, *IDE*, *EDL*, *r0*, *BRS*, *ACK*, *ACK Delimiter*, *EOF* и *IFS* в сообщениях стандартного формата, а также в дополнение к ним поля *18-bit ID* и *SRR* в сообщениях расширенного формата, всегда передаются на номинальной скорости $Baud_N$. Количество битов в этих полях N_0 определяется выражением:

$$N_0(IDE) = 29 + 19 \cdot IDE. \quad (3.3)$$

На скорости данных $Baud_D$ передаются поля *ESI*, *DLC*, *CRC* и поле данных. Количество соответствующих битов N_1 с использованием (3.1), (3.2) и с учётом суммарной длины полей *ESI* и *DLC*, равной 5 битам, вычисляется по формуле:

$$N_1(DLC) = 5 + N_{DATA}(DLC) + N_{CRC}(DLC). \quad (3.4)$$

При записи выражений (3.3) и (3.4) условно принято, что бит *BRS* передаётся полностью на номинальной скорости, а бит *CRC Delimiter* – на скорости данных. Это допущение обосновано тем, что согласно [69] суммарное время передачи указанных битов в любом случае должно быть равно $T_N + T_D$, где

$T_N = 1 / \text{Baud}_N$, $T_D = 1 / \text{Baud}_D$ – время передачи одного бита на номинальной скорости и на скорости данных соответственно.

Однако даже зная количество битов N_0 и N_1 в сообщении, а также время передачи бита T_N и T_D , точно вычислить время передачи всего сообщения невозможно. Это связано с использованием технологии вставки битов (*bit stuffing*), суть которой заключается в следующем: если передатчик обнаружит в потоке передаваемых данных пять последовательных битов с одинаковыми значениями, то он автоматически вставляет в поток дополнительный бит с противоположным значением. Приемник, получая данные, автоматически удаляет дополнительные биты. Описанная процедура выполняется для предотвращения длительного нахождения линии связи в неизменном доминантном (ноль) или рецессивном (единица) состоянии, поскольку это мешает синхронизации приемников с потоком передаваемых данных.

Вставка битов не оказывает влияния на результат обмена данными, но увеличивает время передачи сообщения, причем количество дополнительных битов будет зависеть от содержания конкретного сообщения и поэтому в общем случае является случайной величиной.

В наиболее благоприятном случае число дополнительных битов равно нулю (за исключением фиксированных битов в поле *CRC*, которые уже учтены в (3.2)). Тогда минимальное время передачи сообщения T_{DF_MIN} с учетом (3.3) и (3.4) будет определяться как:

$$T_{DF_MIN} = N_0 T_N + N_1 (T_N + (T_D - T_N) BRS). \quad (3.5)$$

Далее рассмотрим наименее благоприятный случай. Если в сообщении стандартного формата поле «*11 bit ID*» равно $0x000$, то в начале передаются 14 битов одинаковой полярности: *SOF*, «*11 bit ID*», *r1* и *IDE*, из-за чего в сообщение добавляются два дополнительных бита. Для сообщений расширенного формата наихудшая ситуация возникает в случае, когда поле «*11 bit ID*» равно $0x7FF$, а «*18 bit ID*» – $0x3FFFF$: в этом случае после поля *SOF* будет передан 31 бит одинаковой полярности, что приведёт к вставке шести дополнительных битов. Затем перед полем *EDL* и после него дважды происходит изменение уровня

(сначала с 0 в 1, затем с 1 в 0), обнуляющее счетчик последовательных битов. Все рассмотренные дополнительные биты передаются на номинальной скорости $Baud_N$, поскольку добавляются перед полем BRS , где происходит изменение скорости. Их количество определяется выражением:

$$N_{0S}(IDE) = 2 + 4 \cdot IDE. \quad (3.6)$$

Максимальное количество дополнительных битов N'_{1S} в полях $r0$, BRS , ESI , DLC и «Данные» будет добавлено при передаче нулевых данных (при $DLC \in \{3, 15\}$ – единичных), их число можно вычислить по формуле (3.7):

$$N'_{1S} = \lfloor N_{DATA}(DLC)/5 \rfloor + A, \quad (3.7)$$

где $A=1$ при $DLC \in \{3, 8, 14, 15\}$; иначе $A=0$.

В полях ACK , $ACK Delimiter$, EOF и IFS вставка битов не выполняется.

Следует отметить, что выражение (3.7) справедливо лишь для наиболее вероятной ситуации, когда $BRS = 1$ и $ESI = 0$, то есть когда используется альтернативная скорость и передающий узел находится в состоянии «*Error Active*». Использование в режиме $CAN FD$ только номинальной скорости ($BRS = 0$) представляется нецелесообразным, а переход передающего узла в состояние «*Error Passive*» ($ESI = 1$) возможен лишь при большом количестве ошибок обмена данными и в правильно функционирующей сети маловероятен. Тем не менее, если вариант использования только номинальной скорости ($BRS = 0$) не исключается, то в общем виде максимальное количество дополнительных битов N_{1S} с учётом (3.7) можно определить так:

$$N_{1S} = \lfloor N_{DATA}(DLC)/5 \rfloor + A + (1 - BRS)B, \quad (3.8)$$

где $B=1$ при $DLC < 4$; иначе $B=0$.

Анализ ситуации при $ESI = 1$ показывает, что в случае $DLC = 14$ и $BRS = 1$ к значению N_{DS} , полученному с использованием (3.8), следует добавить единицу. Однако это не окажет существенного влияния на точность оценки времени передачи сообщений, поскольку переходу узла в состояние «*Error Passive*» предшествует передача большого количества сообщений типа «*Error Frame*» и повторная передача принятых с ошибкой сообщений типа «*Data Frame*». Это

обстоятельство влияет на эффективность обмена данными в значительно большей степени, чем вставка одного дополнительного бита, а изучение его влияния должно выполняться при исследовании более высоких уровней сетевого взаимодействия.

Обобщая приведённые рассуждения, с учётом (3.5), (3.6) и (3.8) запишем выражение для оценки максимального времени передачи сообщения T_{DF_MAX} :

$$T_{DF_MAX} = (N_0 + N_{0S})T_N + (N_1 + N_{1S})(T_N + (T_D - T_N)BRS). \quad (3.9)$$

Полученные результаты разработки метода докладывались и обсуждались на международной конференции [71].

3.2 Разработка усовершенствованного вероятностного метода и модели для оценки параметров межмодульного взаимодействия

3.2.1 Анализ системы сбора и обработки информации как системы массового обслуживания

В реальных условиях проектирования ССОИ невозможно представить систему, состоящую исключительно из модулей одного типа, передающих только однотипные данные. Напротив, в большинстве случаев архитектура таких систем является гетерогенной и включает в себя модули различных типов, каждый из которых выполняет специализированные функции в рамках общей структуры сбора и управления.

Представленная на рис. 3.3 структурная схема в упрощенном виде демонстрирует организацию системы, состоящей из множества модулей различных типов M . Количество модулей каждого типа обозначено как m_k , где k принимает значение от 1 до M .

Взаимодействие между модулями обеспечивается посредством единой сети передачи данных (СПД), в качестве которой может использоваться практически любой шинный интерфейс: *CAN*, *CAN FD*, *I2C*, *RS-485* и т.д. Такая структура позволяет реализовать гибкую масштабируемую ССОИ, которую легко адаптировать к изменяющимся условиям эксплуатации.

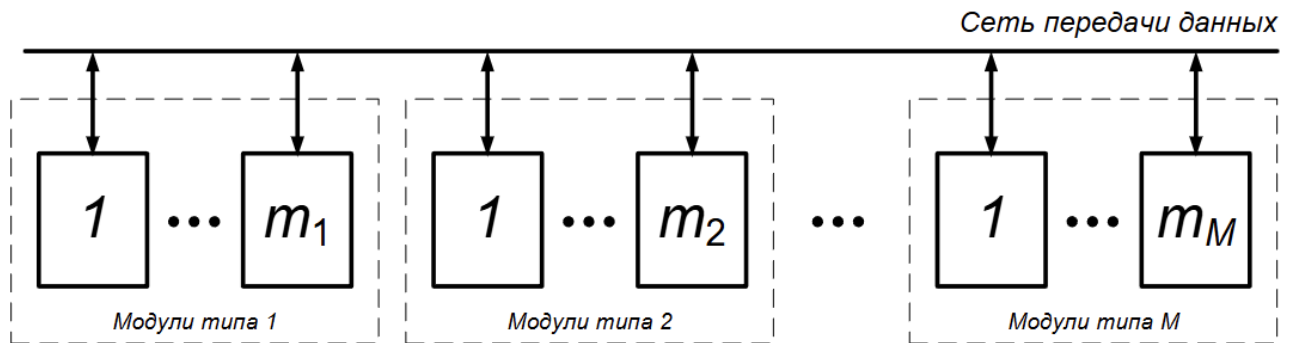


Рисунок 3.3 – Структурная схема гетерогенной модульной ССОИ
(M – количество типов модулей, m_k ($k = \{1, 2, \dots, M\}$) – количество модулей k -го типа)

В работах [72, 73] представлены методы и инструменты, предназначенные для исследования функционирования модульных систем путем разработки имитационных моделей коммуникационных процессов в системах, использующих протокол *CANopen*. Однако предложенные подходы обладают ограниченной вычислительной эффективностью и не обеспечивают оперативную оценку всех возможных конфигураций системы при варьировании параметров настройки модулей.

В дополнение к этому, известен вероятностный метод, а также соответствующая ему модель, разработанная для анализа параметров межмодульного взаимодействия в схожих системах [74]. Однако данный метод основан на использовании устаревшего интерфейса *CAN* в качестве основной СПД, что существенно ограничивает его применение в условиях высоконагруженных ССОИ и изменяющихся условий эксплуатации.

Учитывая вышеизложенные недостатки известных методов, возникает необходимость разработки усовершенствованного вероятностного метода, который позволил бы решать поставленные задачи с применением математического аппарата теории массового обслуживания. Разработанный автором метод [75] ориентирован на формализованную оценку параметров межмодульного взаимодействия в ССОИ, использующих в качестве СПД высокоскоростную шину *CAN FD* и протокол обмена данными *CANopen FD*. При этом область применения метода не ограничивается исследованием ССОИ ПС, он

может быть использован для анализа любых модульных систем, функционирующих на основе данного интерфейса и протокола, т.е. отличается универсальностью.

От количества и параметров настройки модулей в системе зависит интенсивность потока передаваемых данных. Исходя из этого, возникает необходимость решения следующих задач (параграф 1.4.3):

- оценка возможности своевременной передачи высокоприоритетных (ВПД) и низкоприоритетных данных (НПД) от всех модулей посредством СПД;
- оценка возможности изменения параметров настройки модулей для оптимизации работы системы.

Рассмотрим исследуемую ССОИ как систему массового обслуживания (рис. 3.4). В процессе функционирования системы модули генерируют потоки заявок (сообщений), направляемых в СПД с различными интенсивностями $\lambda_1, \lambda_2, \dots, \lambda_H$, где H обозначает общее число классов заявок. При этом, чем выше номер класса, тем ниже приоритет соответствующей заявки.

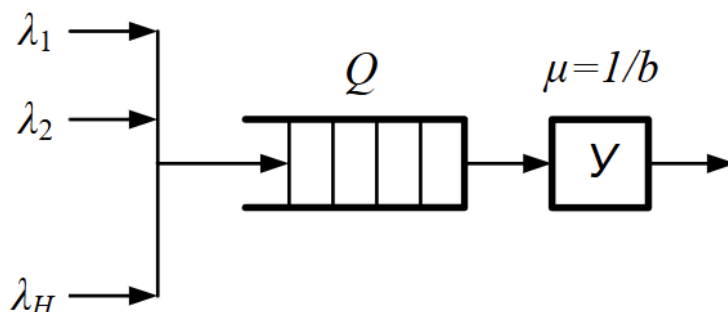


Рисунок 3.4 – Система массового обслуживания (СМО) - модель ССОИ:
 λ_h – интенсивность h -го потока заявок, H – число классов заявок, Q – буфер,
 U – обрабатывающее устройство, μ – интенсивность обслуживания, b – время обслуживания

Поступающие в СПД сообщения аккумулируются в буфере Q , где формируется очередь обработки сообщений, организованная в соответствии с уровнями приоритета обслуживания, что позволяет упорядочить поток передаваемых данных.

Обработка накопленных заявок осуществляется блоком управления U , который функционально эквивалентен сети передачи данных *CAN FD*. Реализация

процесса обработки осуществляется в соответствии с механизмом относительной приоритизации заявок.

Интенсивность обслуживания μ определяется как обратная величина времени передачи одного сообщения b :

$$\mu = \frac{1}{b}.$$

В дальнейшем будут детально рассмотрены ключевые параметры, присущие системам массового обслуживания (СМО), а также соответствующие им характеристики в контексте функционирования ССОИ.

В ССОИ модули формируют заявки (сообщения) следующих классов:

- сообщения службы синхронизации (*SYNC*), имеющие наивысший приоритет $h = 1$ и передаваемые ведущим модулем периодически для координации работы других модулей системы;

- сообщения службы *PDO*, имеющие несколько уровней приоритета $h = \{2, 3, \dots, H-2\}$ и предназначенные для передачи модулями текущих данных;

- сообщения службы *Heartbeat* (*HBEAT*) с приоритетом $h = H-1$, формируемые периодически каждым модулем с целью подтверждения работоспособности;

- сообщения службы *USDO*, имеющие низший приоритет H и предназначенные для передачи сравнительно больших объёмов НПД.

В [74] показано, что интенсивность потока сообщений *SYNC* определяется выражением:

$$\lambda_1 = 1/T_{SYNC}. \quad (3.10)$$

Интенсивность потока сообщений *HBEAT* вычисляется как:

$$\lambda_{H-1} = \frac{1}{T_{SYNC}} \sum_{k=1}^M \frac{m_k}{B_k}, \quad (3.11)$$

где M – количество типов модулей в рассматриваемой системе;

m_k – количество модулей k -го типа;

B_k – период передачи сообщений *HBEAT* модулями k -го типа, выраженный в циклах синхронизации.

Интенсивность потока заявок PDO $\lambda_{k,h}$ (заявки класса h , формируемые одним модулем k -го типа) вычисляется с помощью выражения:

$$\lambda_{k,h} = \min \left(\frac{1}{T_{SYNC}}, a_{k,h} \frac{(1 - e^{-a_{k,h} B_{k,h} T_{SYNC}})^2}{1 - (1 + a_{k,h} B_{k,h} T_{SYNC}) e^{-a_{k,h} B_{k,h} T_{SYNC}}} + \frac{e^{-a_{k,h} B_{k,h} T_{SYNC}}}{B_{k,h} T_{SYNC}} \right), \quad (3.12)$$

$$a_{k,h} = \sum_{i=1}^{V_{k,h}} a_{k,h,i}, \quad (3.13)$$

где $B_{k,h}$ – период передачи сообщений класса h модулем k -го типа, выраженный в циклах синхронизации;

$a_{k,h,i}$ – параметр экспоненциального распределения вероятностей выхода i -й измеряемой и передаваемой в сообщении (k, h) величины за пределы зоны нечувствительности.

Интенсивность потока заявок одного класса h ($h \in \{1, \dots, H-1\}$) всех модулей ССОИ вычисляется как:

$$\lambda_h = \sum_{k=1}^M m_k \lambda_{k,h}, \quad (3.14)$$

где M – количество типов модулей в ССОИ;
 m_k – количество модулей k -го типа.

Суммарная интенсивность потока сообщений от модулей ССОИ (без учета НП-заявок H службы $USDO$) с использованием (3.10), (3.11), (3.12), (3.14) может быть вычислена как:

$$\lambda_S = \sum_{h=1}^{H-1} \lambda_h, \quad (3.15)$$

где H – количество классов заявок.

3.2.2 Разработка усовершенствованного вероятностного метода оценки параметров

В [74] подробно рассмотрен метод определения параметров взаимодействия модулей в системах управления защитой турбоагрегата по вибрации, использующих протокол $CANopen$ на базе интерфейса CAN . Эти системы, подобно рассматриваемой, также представляют собой СМО с неоднородным потоком

заявок, однако спецификой систем вибромониторинга является измерение однотипных сигналов вибрации. В [74] показано, что вероятности их изменения за определённый интервал времени примерно одинаковы, поэтому в методе-прототипе используется общий параметр экспоненциального распределения вероятности для всех сигналов. В рассматриваемой ССОИ сигналы разнородны и вероятности их изменения существенно различаются, поэтому описанный подход приведёт к увеличению погрешности результатов моделирования. Более того, в методе, разработанном в данной диссертации и описанном в [75], предлагается использование более современного протокола *CANopen FD* [56] на базе интерфейса *CAN FD*. Одно из основных отличий этого протокола заключается в службе приёма/передачи служебных данных *USDO* (в *CANopen* использовалась служба *SDO*), которая была существенно доработана для обеспечения эффективной передачи больших объёмов данных. Остальные службы *CANopen FD* (*SYNC*, *PDO*, *HBEAT*) в основном остались без изменений и подробно описываются в [74]. В связи с этим возникла необходимость разработки усовершенствованного вероятностного метода, учитывающего неодинаковые вероятности изменения сигналов и особенности используемого протокола.

В параграфах 3.1.1 и 3.1.2 показано, что время передачи одного сообщения по шине *CAN FD* зависит от многих факторов, основными из них являются скорость работы интерфейса, формат и содержание сообщения (в зависимости от содержания изменяется количество дополнительных битов). Время передачи сообщения влияет на интенсивность обслуживания поступающих заявок. *CAN FD* отличается от классического *CAN* тем, что определенные части сообщения *CAN FD* передаются с разной битовой скоростью. Одно целое сообщение *CAN FD* состоит из двух групп полей: служебные поля и поля данных в соответствии со структурой сообщения [23]. Служебные поля передаются на номинальной скорости V_{aud_N} , которая может достигать до 1 Мбит/с, поля данных – на повышенной скорости (до 8 Мбит/с). Один бит поля данных будет передан за время равное $T_D = 1 / V_{aud_D}$, биты остальных полей – за время $T_N = 1 / V_{aud_N}$ каждый (подробное описание полей и количество бит в них, также выражения для определения минимального и

максимального времени передачи сообщения приведены в параграфах 3.1.1 и 3.1.2).

Основными выражениями для вычисления количества бит в служебных полях N_0 и в поле данных N_1 являются:

$$N_0 = 29 + 19 \cdot IDE, \quad (3.16)$$

$$N_1(DLC) = 5 + N_{DATA}(DLC) + N_{CRC}(DLC), \quad (3.17)$$

где N_{DATA} – длина основного блока данных;

N_{CRC} – длина поля контрольной суммы (28 или 33 бита);

IDE – признак расширенного формата сообщения;

DLC – код длины поля данных.

Таким образом, минимальное время передачи сообщения *CAN FD* с учётом (3.16) и (3.17), можно определить с использованием выражения:

$$t_{\min} = N_0 T_N + N_1 (T_N + (T_D - T_N) BRS), \quad (3.18)$$

где BRS – признак использования альтернативной скорости.

При передаче любого сообщения по шине *CAN FD*, практически гарантированно будут автоматически вставлены дополнительные биты [76, 77, 78]. Выражения для подсчета максимального количества дополнительных битов, добавляемых в служебных полях (N_{0S}) и в полях данных (N_{1S}), обоснованы в параграфе 3.1.2 и описаны автором в [76]:

$$N_{0S}(IDE) = 2 + 4 \cdot IDE, \quad (3.19)$$

$$N_{1S} = \lfloor N_{DATA}(DLC) / 5 \rfloor + A + (1 - BRS)B, \quad (3.20)$$

где $A=1$ при $DLC \in \{3, 8, 14, 15\}$; иначе $A=0$;

$B=1$ при $DLC < 4$; иначе $B=0$.

Наименее благоприятный случай возникает при максимальном добавлении дополнительных битов, а время передачи для этого случая с учетом (3.18), (3.19) и (3.20) определяется как:

$$t_{\max} = (N_0 + N_{0S}) T_N + (N_1 + N_{1S}) (T_N + (T_D - T_N) BRS).$$

Определить точное количество автоматически добавляемых бит, ввиду специфичности передаваемых сообщений и разнообразия их содержания, в данном

вероятностном методе не представляется возможным. С целью учета того факта, что дополнительные биты увеличивают общее время передачи всего сообщения, введем коэффициент η , определяющий степень этого увеличения.

$$t_{bs} = t_{\min} + \eta(t_{\max} - t_{\min}), \quad (3.21)$$

где t_{bs} – время передачи сообщения с учетом дополнительных бит;
 $\eta \in [0, 1]$ – коэффициент увеличения времени передачи.

Среднее время обслуживания заявки b в системе, работающей без перегрузки, с учетом (3.15) можно выразить как:

$$b = \frac{\lambda_1 T_{1,1} + \sum_{k=1}^M (m_k \sum_{h=2}^{H-1} \lambda_{k,h} t_{k,h})}{\lambda_S}, \quad (3.22)$$

где $t_{k,h}$ – время передачи t_{bs} сообщения h -класса, сформированного модулем k -го типа.

Нагрузка системы y является показателем нормального функционирования ССОИ и вычисляется выражением $\mu = 1 / b$. Если нагрузка сети окажется больше единицы, то это свидетельствует о неправильной работе ССОИ ввиду перегрузки сети передачи данных. Поэтому для корректной работы системы в целом к нагрузке предъявляется строгое требование $y < 1$, это обеспечит своевременную передачу как минимум высокоприоритетных сообщений.

НПД не являются критически важными для корректного функционирования ССОИ, но позволяют получить дополнительную информацию о состоянии отдельных элементов и параметрах работы системы. Примерами таких данных являются отчеты о функционировании системы, о возникающих событиях, осциллограммы. Поэтому передавать такие данные все-таки нужно, но в свободное от передачи ВПД время. Для обеспечения возможности учета передачи НП-сообщений в период отсутствия нагрузки на СПД, введем коэффициент простоя ρ , вычисляемый выражением $\rho = 1 - y$.

Средняя скорость обслуживания НП-заявок зависит от времени передачи НП-сообщений, которое в свою очередь сильно зависит от количества необходимой служебной информации и числа дополнительных битов. НПД передаются посредством использования службы *USDO* в режиме блочной передачи

данных [79]. Один объект данных передаётся службой *USDO* с использованием множества сообщений нескольких типов с различной длиной поля данных: запрос на установку связи (тип 1, длина 12 байт), подтверждение установки связи (тип 2, длина 8 байт), сообщения с данными (тип 3, длина 64 байта), последнее сообщение с данными (тип 4, длина переменная), подтверждение приёма (тип 5, длина 3 байта). Сообщения типа 1 и 2 передаются однократно. Сообщения типа 3 содержат по 60 байт полезных данных, поэтому их количество зависит от размера передаваемых данных в байтах N . Сообщение типа 4 передаётся однократно и содержит остаток полезных данных, если N не кратно 60. Сообщения типа 5 подтверждают приём некоторого количества Z сообщений типов 3 и 4.

Количество бит N_0 , передаваемых в каждом сообщении на номинальной скорости, одинаково для всех сообщений *USDO* и определяется (3.16). Количество бит N_1^i , передаваемых на альтернативной скорости, зависит от типа сообщения i , то есть от длины поля данных, и определяется (3.17).

Суммарное количество бит, передаваемых на номинальной скорости $N_0^{\text{сумм}}$ и альтернативной скорости $N_1^{\text{сумм}}$ с учетом (3.16), (3.17), (3.19), (3.20) и с использованием коэффициента вставки дополнительных битов η :

$$N_0^{\text{сумм}} = \left(2 + \left\lfloor \frac{N}{60} \right\rfloor + \left(\left\lceil \frac{N}{60} \right\rceil - \left\lfloor \frac{N}{60} \right\rfloor \right) + \left\lceil \frac{\left\lceil \frac{N}{60} \right\rceil}{Z} \right\rfloor \right) (N_0 + N_{0s}\eta), \quad (3.23)$$

$$N_1^{\text{сумм}} = N_1(12) + N_{1s}(12)\eta + N_1(8) + N_{1s}(8)\eta + N_1(64) + N_{1s}(64)\eta + N_1(N \bmod 60) + N_{1s}(N \bmod 60)\eta + \left\lceil \left(\left\lceil \frac{N}{60} \right\rceil \right) / Z \right\rceil (N_1 + N_{1s}\eta). \quad (3.24)$$

Используя выражения (3.23), (3.24) и зная время, за которое передаются биты N_0 и N_1 , можно определить теоретическое суммарное время передачи одного *USDO*:

$$T^{\text{сумм}} = N_0^{\text{сумм}} \cdot T_N + N_1^{\text{сумм}} \cdot T_D.$$

Фактическое время передачи зависит от коэффициента простоя ρ и вычисляется как:

$$T_{\text{факт}}^{\text{сумм}} = \frac{T^{\text{сумм}}}{\rho}. \quad (3.25)$$

Используя выражение (3.25) и зная размер передаваемого объекта данных N , можно определить среднюю скорость передачи этого объекта в байтах в секунду службой *USDO*:

$$Q_H = \frac{N}{T_{\text{факт}}^{\text{сумм}}}. \quad (3.26)$$

3.2.3 Разработка усовершенствованной вероятностной модели системы сбора и обработки информации

Разработанный в параграфе 3.2.2 усовершенствованный вероятностный метод оценки параметров реализован в вероятностной модели, предназначенной для анализа различных конфигураций ССОИ и изучения влияния параметров настройки входящих в неё модулей на нагрузку СПД.

Составлен набор необходимых исходных данных для исследования межмодульного взаимодействия в ССОИ:

– $BitRate_N$ – номинальная скорость передачи данных по шине *CAN FD*, бит/с;

– $BitRate_D$ – альтернативная скорость передачи данных по шине *CAN FD*, бит/с;

– $TSYNC$ – продолжительность цикла синхронизации ССОИ, с;

– η – коэффициент увеличения времени передачи в результате автоматической вставки бит (*bit stuffing*);

– AJ – трёхмерный массив параметров $\alpha_{k,h,j}$ экспоненциального распределения вероятностей выхода за пределы зоны нечувствительности для каждого контролируемого сигнала (см. выражение (3.13));

– α – усреднённый параметр экспоненциального распределения вероятностей выхода измеряемого или вычисляемого значения, передаваемого в *PDO*, за пределы зоны нечувствительности (может использоваться вместо массива AJ);

– H – количество классов заявок;

– M – число типов модулей в ССОИ;

– $\mathbf{M} = [m_1 \ m_2 \ \dots \ m_M]$ – вектор, содержащий количество модулей каждого типа;

– \mathbf{B} – матрица периодичности отправки сообщений;

– \mathbf{D} – матрица длин сообщений (*DLC*) с диапазоном значений от 0 до 64 или -1, если модуль не передаёт сообщения данного класса;

$$B = (b_{k,h})_{\substack{k=1,\overline{M} \\ h=1,\overline{H-1}}}, \quad D = (d_{k,h})_{\substack{k=1,\overline{M} \\ h=1,\overline{H-1}}};$$

– \mathbf{V} – матрица, содержащая количество отдельных параметров, передаваемых в каждом *PDO*;

– \mathbf{LM} – матрица режима вычисления интенсивностей потоков заявок, задающая для каждого сообщения один из трёх режимов передачи:

«0» – фиксированная периодичность отправки заявок;

«1» – фиксированная периодичность с возможностью досрочной передачи;

«2» – специальный режим вычисления интенсивности потока:

$$V = (v_{k,h})_{\substack{k=1,\overline{M} \\ h=1,\overline{H-1}}}, \quad LM = (lm_{k,h})_{\substack{k=1,\overline{M} \\ h=1,\overline{H-1}}}.$$

Использование матрицы \mathbf{LM} обусловлено необходимостью моделирования передачи сообщений, содержащих признаки выхода некоторой величины за границы заданного допустимого диапазона значений. Определить закон распределения интервалов времени между сообщениями такого типа не представляется возможным, поскольку он зависит от текущего режима работы и состояния оборудования объекта наблюдения.

Для иллюстрации рассмотрим пример из области мониторинга электрических ПС. Так, если температура трансформатора ПС остаётся в допустимых пределах, то соответствующий дискретный флаг сигнализации не изменяется и сохраняет нулевое значение. Однако при достижении порогового уровня даже незначительные колебания температуры могут вызывать частые изменения флага сигнализации, что приведёт к резкому увеличению количества заявок на передачу сообщения, содержащего этот флаг. Для таких случаев в \mathbf{LM} предусмотрен режим «2», в котором интенсивность потока заявок определяется

параметром $StressMode$: $StressMode = 0$ – минимальная интенсивность потока; $StressMode = 1$ – максимальная интенсивность потока (имитация предельных нагрузок СПД).

Основные этапы работы модели

1. Формирование матрицы экспоненциального распределения временных интервалов между заявками каждого класса от каждого типа модулей (матрица A) различается для случаев:

– если параметры экспоненциального распределения для каждого контролируемого сигнала примерно одинаковы, то в модели может использоваться общий параметр a и тогда формирование матрицы A выполняется следующим образом:

$$A = (a_{k,h}) = aV;$$

– если есть необходимость в получении более точного результата моделирования, можно использовать отдельные параметры $\alpha_{k,h,j}$ экспоненциального распределения для каждого контролируемого сигнала, заданные в массиве AJ , в этом случае элементы матрицы A вычисляются по формуле (3.13).

2. Формирование матрицы длительностей передачи сообщений:

$$T = (t_{k,h})_{\substack{k=1,\overline{M} \\ h=1,\overline{H-1}}} : t_{k,h} = F_1(d_{k,h}, \eta),$$

где $F_1(d_{k,h}, \eta)$ – функция, определяемая выражением (3.25).

3. Расчёт матрицы интенсивностей потоков сообщений каждого класса, отправляемых одним модулем каждого типа:

$$L = (l_{k,h})_{\substack{k=1,\overline{M} \\ h=1,\overline{H-1}}} : l_{k,h} = \begin{cases} 0 & \text{при } b_{k,h} = 0 \\ (T_{SYNC} b_{k,h})^{-1} & \text{при } lm_{k,h} = 0 \\ F_2(T_{SYNC}, b_{k,h}, a_{k,h}) & \text{при } lm_{k,h} = 1 \\ (T_{SYNC} b_{k,h}) & \text{при } lm_{k,h} = 2 \text{ и } StressMode = 0 \\ T_{SYNC}^{-1} & \text{при } lm_{k,h} = 2 \text{ и } StressMode = 1 \end{cases},$$

где $F_2(T_{SYNC}, b_{k,h}, a_{k,h})$ – функция, определяемая выражением (3.12).

4. Формирование матрицы интенсивностей потоков заявок каждого класса от всех модулей каждого типа:

$$\Lambda = (\lambda_{k,h})_{\substack{k=1,\overline{M} \\ h=1,H-1}} = L \circ (E \oplus M),$$

где E – единичный вектор размерности $H - 1$.

5. Определение средней длительности обслуживания одной заявки b (в соответствии с выражением (3.22)):

$$X = (x_{k,h})_{\substack{k=1,\overline{M} \\ h=1,H-1}} = T \circ \Lambda, \quad b = \lambda_S^{-1} \sum_{k,h} x_{k,h}.$$

6. Расчёт нагрузки системы y и коэффициента простоя ρ по формулам:

$$y = \lambda_S \cdot b, \quad \rho = 1 - y.$$

7. Определение скорости передачи низкоприоритетных данных Q_H , рассчитываемой согласно выражению (3.26).

На усовершенствованную вероятностную модель получено свидетельство о регистрации ПО (Приложение Д) [80], исходный текст разработанного ПО представлен в Приложении Б.

3.3 Разработка имитационной модели для оценки параметров межмодульного взаимодействия

Разработанная в параграфе 3.2.3 вероятностная модель ССОИ позволяет оперативно оценить принципиальную возможность реализации работоспособной системы с заданными параметрами СПД, количеством и типами модулей и типами сообщений. Однако эта модель не учитывает все параметры настройки отдельных модулей, возможные сбои в работе модулей, нетипичные изменения входных сигналов.

Указанных недостатков лишена разработанная имитационная модель для исследования межмодульного взаимодействия в ССОИ [81]. Модель создана на основе разработанного метода оценки времени доставки сообщений по шине *CAN FD* (параграф 3.1.2), представленной модульной структуры ССОИ (параграф 2.1), а также с использованием разработанного описания модульной

системы (параграф 2.4).

Имитационная модель имеет иерархическую структуру, показанную на рис. 3.5, и состоит из необходимого количества модулей верхнего уровня (рис. 3.6) и модели сети передачи данных (рис. 3.7).

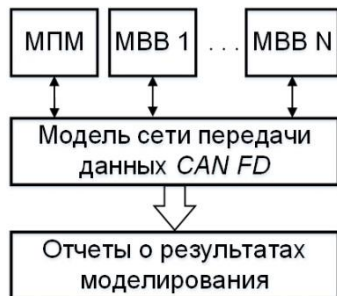


Рисунок 3.5 – Иерархическая структура имитационной модели модульной ССОИ

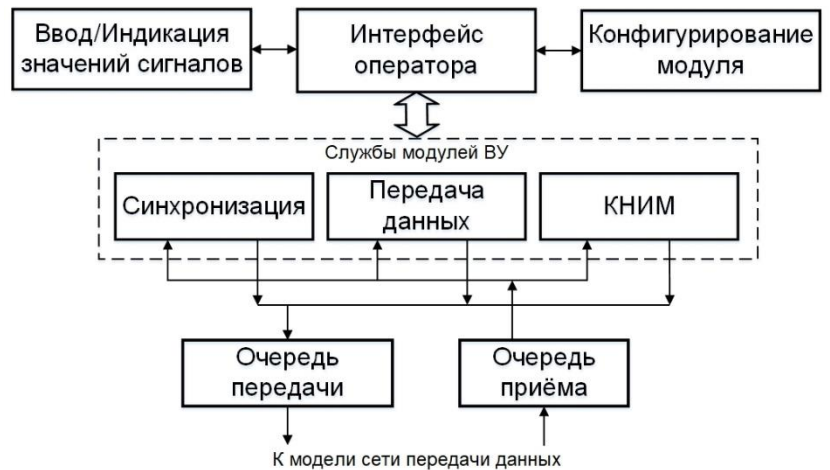


Рисунок 3.6 – Модель модуля верхнего уровня

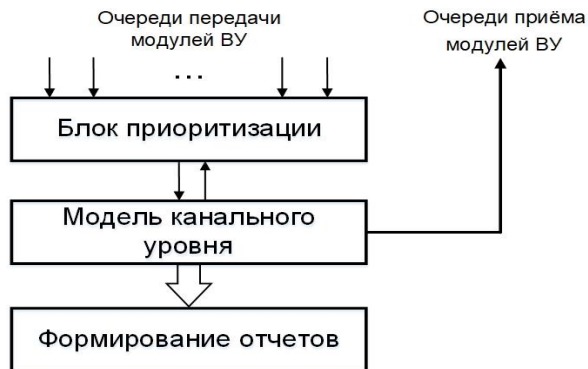


Рисунок 3.7 – Модель сети передачи данных (нижний уровень)

Модели модулей верхнего уровня выполняют следующие функции.

1. Микропроцессорный модуль (МПМ). С точки зрения сетевого взаимодействия его функцией является только формирование сообщений синхронизации. Параметрами модели этого модуля являются период передачи сообщений и их идентификатор.

2. Модуль дискретного ввода. Анализирует введенные оператором или

считанные из файла состояния входных дискретных сигналов, выявляет их изменения и информирует об этом МПМ путём передачи сообщений. Кроме того, формирует сообщения службы контроля наличия и исправности модуля *Heartbeat*. Параметрами модели являются идентификаторы сообщений данных и режимы их передачи (синхронный и/или асинхронный, период отправки синхронных сообщений), а также идентификатор и период передачи сообщений службы *Heartbeat*.

3. Модуль аналогового ввода. Анализирует введённые оператором или считанные из файла значения входных аналоговых сигналов, выявляет их отклонения от текущего уровня более, чем на заданную величину, и информирует об этом ПМ. Параметры модели аналогичны модулю дискретного ввода, дополнительно задаётся величина отклонения, в случае превышения которой формируется асинхронное сообщение данных.

4. Модуль дискретного вывода. При работе ССОИ этот модуль получает от ПМ команды на изменение состояния выходных сигналов. Такие команды подаются сравнительно редко и существенного влияния на сетевое взаимодействие не оказывают, поэтому модель модуля формирует лишь сообщения службы *Heartbeat*.

Созданная имитационная модель работает следующим образом. При запуске программы в основном окне имеется возможность добавить необходимые модули или загрузить готовую конфигурацию ССОИ из *XML*-файла описания (рис. 3.8).

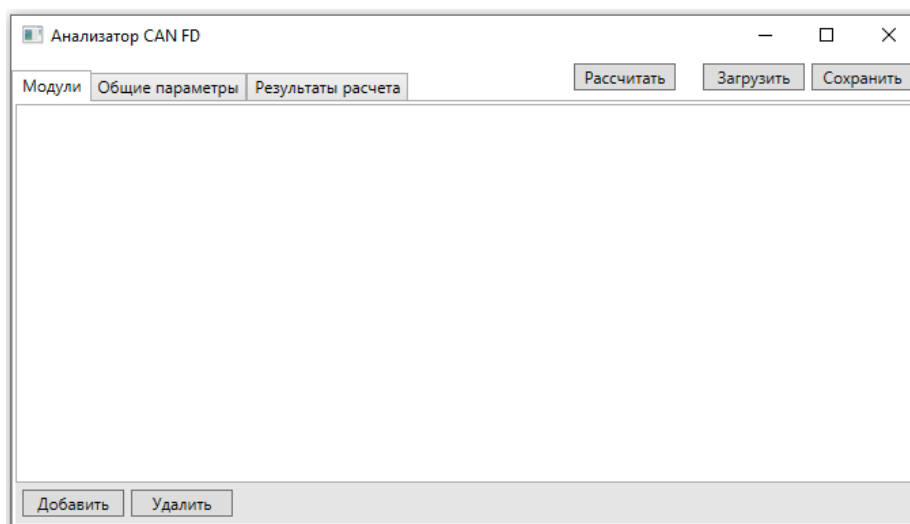


Рисунок 3.8 – Основное окно имитационной модели в начале работы

В интерфейсах добавленных модулей задаются описанные выше параметры их работы (рис. 3.9).

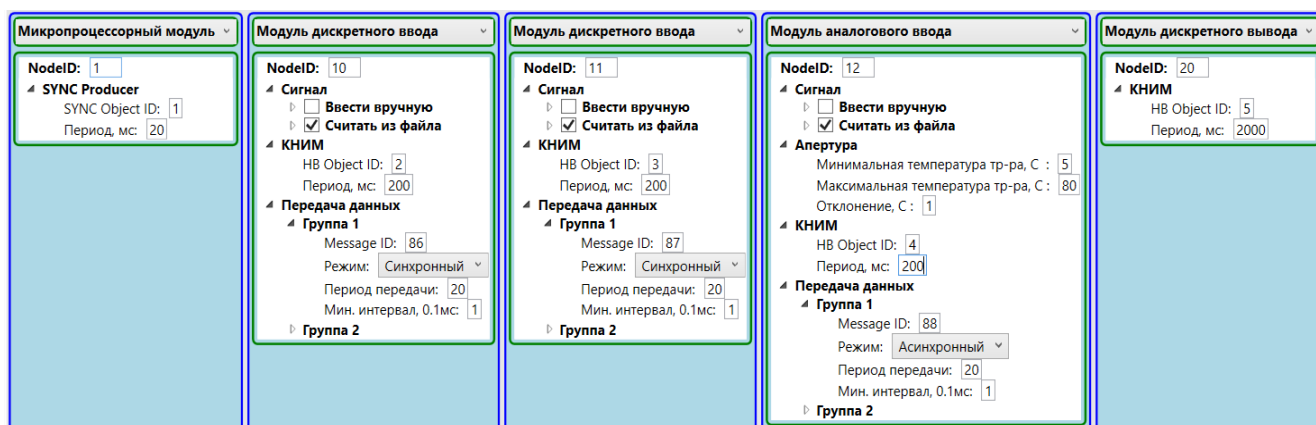


Рисунок 3.9 – Пример готовой конфигурации модулей

Во вкладке «Общие параметры» задаются номинальная и альтернативная скорости обмена, а также режим моделирования вставки битов (рис. 3.10). Вставка битов, подробно рассмотренная в параграфах 3.1.2 и 3.2.2, может существенно влиять на время передачи сообщения. Поэтому предусмотрено три режима её моделирования:

– режим «Нет» – дополнительные биты не вставляются, следовательно, передается только сообщение, длина которого рассчитывается по формулам (3.16) и (3.17);

– режим «Максимум» – к вычисленной длине сообщения добавляется максимально возможное количество дополнительных битов, рассчитываемое по формулам (3.19) и (3.20);

– режим «Реально» – каждое сообщение анализируется моделью с целью определения фактического количества дополнительных битов для корректного приёма/передачи согласно технологии *bit stuffing* (наиболее точный, но самый медленный вариант).

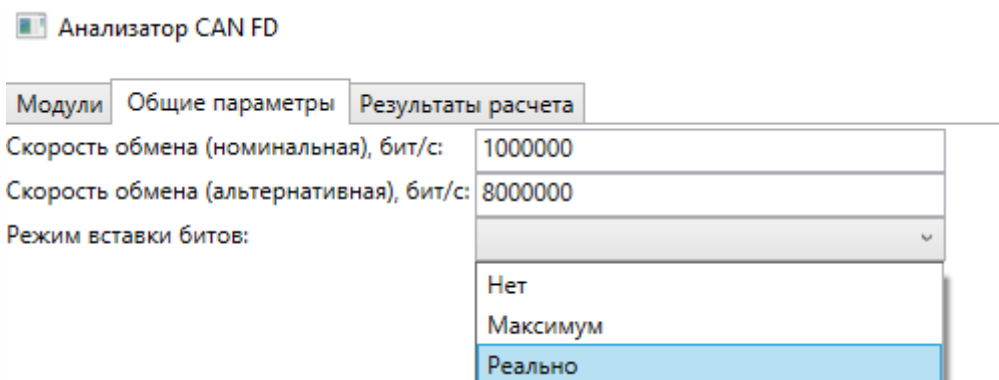


Рисунок 3.10 – Настройка общих параметров

После указания всех необходимых параметров конфигурации и нажатия кнопки «Рассчитать» модель начинает имитировать работу служб протокола *CANopen FD* для каждого модуля, формируя необходимые сообщения и направляя их на вход модели СПД. Модель СПД анализирует полученные сообщения, выстраивает их в порядке убывания приоритетов и направляет на входы моделей отдельных модулей, службы которых обрабатывают полученные данные и реагируют на них соответствующим образом. Процесс циклически повторяется в течение заданного оператором времени с целью накопления массива информации достаточного для анализа объёма.

В результате работы модели ССОИ создаётся массив данных, содержащий для каждого сообщения информацию о модуле-источнике, о времени постановки в очередь, времени начала передачи и длительности передачи (рис. 3.11). На основании этих сведений можно вычислить нагрузку сети в течение заданного интервала времени, выявить факты недопустимой задержки при отправке отдельных сообщений и скорректировать параметры настройки модулей для

обеспечения надёжного сетевого взаимодействия. Пример визуализации результатов анализа полученных данных программой приведён на рис. 3.12.

Анализатор CAN FD

Модули | Общие параметры | Результаты расчета | Рассчитать | Загрузить | Сохранить

| Модуль | Сообщение | Id | В очередь, мкс | Начало ПРД, мкс | Время передачи, мкс | Доставлено, мкс | Время доставки, мкс |
|--------|-------------------------|----|----------------|-----------------|---------------------|-----------------|---------------------|
| МПМ | Запрос 1 | 1 | 0 | 0 | 44 | 44 | 44 |
| MBV3 | Ответ 1 (нет изменений) | 30 | 52 | 52 | 44 | 96 | 44 |
| MBV1 | Ответ 1 (изменения) | 10 | 53 | 99 | 108 | 207 | 154 |
| MBV1 | Ответ 1 (блокировка) | 11 | 53 | 210 | 108 | 318 | 265 |
| MBV2 | Ответ 1 (нет изменений) | 20 | 53 | 321 | 44 | 365 | 312 |
| МПМ | Запрос 2 | 1 | 1000 | 1000 | 44 | 1044 | 44 |
| MBV2 | Ответ 2 (изменения) | 20 | 1053 | 1053 | 108 | 1161 | 108 |
| MBV1 | Ответ 2 (нет изменений) | 10 | 1058 | 1164 | 44 | 1208 | 150 |
| MBV2 | Ответ 2 (блокировка) | 21 | 1053 | 1211 | 108 | 1319 | 266 |
| MBV3 | Ответ 2 (нет изменений) | 30 | 1058 | 1322 | 44 | 1366 | 308 |

Рисунок 3.11 – Результаты работы имитационной модели в табличном виде

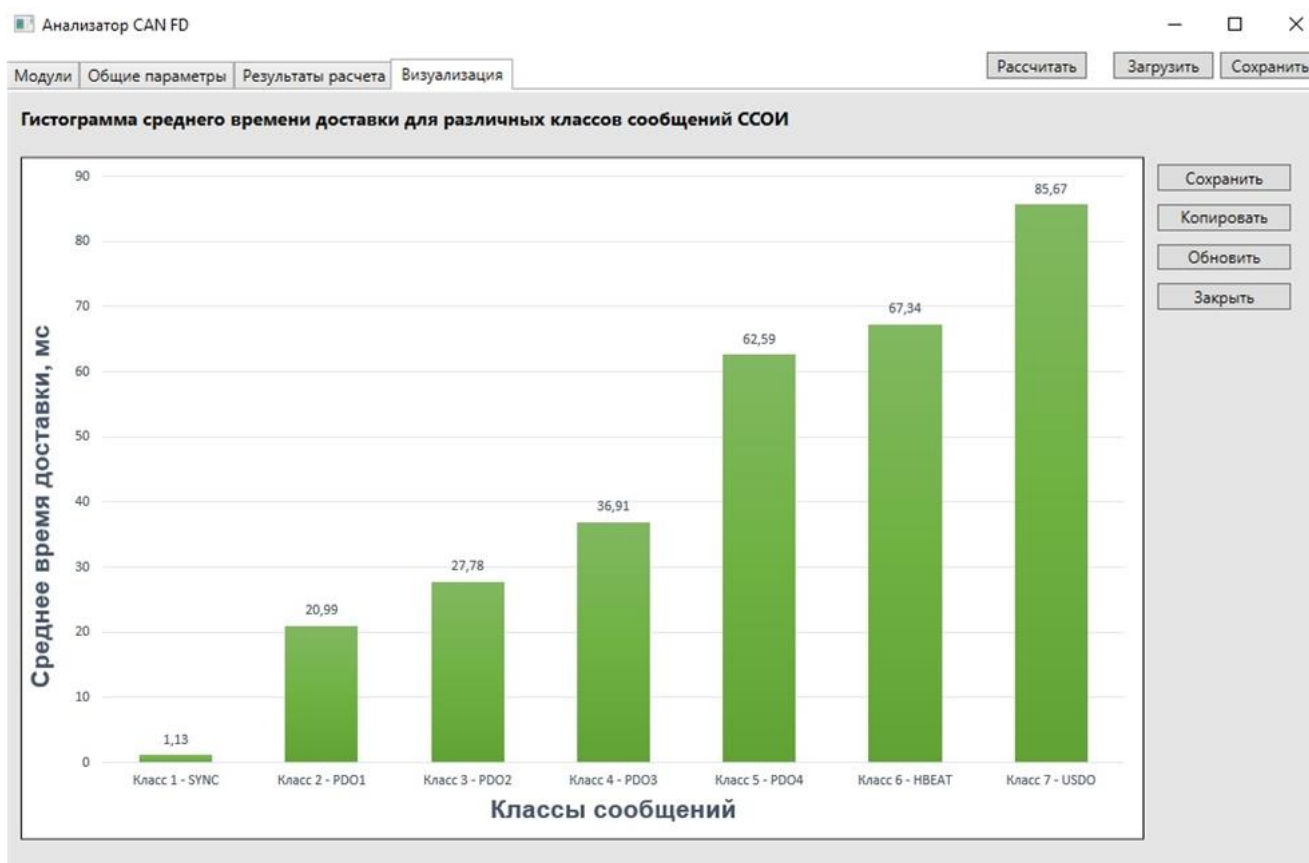


Рисунок 3.12 – Статистическая обработка результатов моделирования

Исходные тексты основных модулей разработанного ПО, реализующего работу имитационной модели, представлены в Приложении Б.. Результаты разработки имитационной модели докладывались и обсуждались на международной конференции [81].

Параметры сконфигурированной и отлаженной модели ССОИ можно сохранить в *XML*-файл описания для дальнейшего использования при следующем моделировании или при настройке реальной системы (см. параграф 2.4). Аналогично *XML*-файл конфигурации, считанный с реальной ССОИ, можно загрузить в имитационную модель с целью детального исследования межмодульного взаимодействия в конкретной системе.

3.4 Оценка адекватности разработанных моделей

Для получения ответа на вопрос, насколько хорошо разработанные вероятностная и имитационная модели представляют моделируемую ССОИ, необходимо оценить их адекватности. Один из подходов к оценке адекватности заключается в сравнении выходов моделируемой и реальной системы при одинаковых значениях входов [82, 83, 84, 85]. Данные, получаемые при выполнении моделирования и натурного эксперимента, являются статистическими, поэтому для оценки адекватности моделей применяются методы статистической теории оценивания и проверки гипотез [82].

Адекватность исследуемой модели можно оценить различными методами:

- по средним значениям откликов модели и реальной системы (*t*-критерий Стьюдента);
- по дисперсиям отклонений откликов модели от среднего значения откликов реальной системы (*F*-критерий Фишера, критерий согласия χ^2 , критерий Колмогорова-Смирнова).

Для исследования разработанных моделей были выбраны методы на основе *t*-критерия Стьюдента и *F*-критерия Фишера [86, 87], поскольку они позволяют выполнить комплексный анализ результатов эксперимента, подходят для большинства реальных данных, имеющих небольшие выборки, и отличаются простотой применения.

В процессе оценки адекватности вероятностной и имитационной моделей были проведены натурные эксперименты и выполнено моделирование ССОИ в конфигурации, представленной в таблице 3.1.

Таблица 3.1 – Типы модулей реальной ССОИ

| Тип | Наименование модуля | Количество модулей в ССОИ |
|-----|--|---------------------------|
| 1 | Процессорный модуль | 1 |
| 2 | Модуль измерения линейных токов и напряжений | 4 |
| 3 | Модуль аналогового ввода | 8 |
| 4 | Модуль дискретного ввода | 6 |
| 5 | Модуль дискретного вывода | 3 |

При исследовании реальной ССОИ проводилось несколько серий экспериментов на разных скоростях шины *CAN FD* (100 кбит/с, 200 кбит/с, 500 кбит/с). Серия для каждой скорости включала 30 наблюдений, в каждом эксперименте на вход системы подавались контролируемые сигналы, после чего в течение 10 секунд измерялась средняя нагрузка сети (Приложение В.1, В.3, В.5, «колонка 2») и средняя скорость передачи низкоприоритетных данных *USDO* (Приложение В.2, В.4, В.6, «колонка 2»). Графики изменения входных сигналов в каждом эксперименте сохранялись в файл с помощью многоканального регистратора.

На следующем этапе исследовалась имитационная модель ССОИ в той же конфигурации (табл. 3.1). Аналогично предыдущему этапу выполнялось несколько серий наблюдений для различных скоростей шины *CAN FD*, каждая серия содержала 30 наблюдений. В каждом наблюдении на вход модели подавались предварительно сохранённые графики входных сигналов и в течение 10 секунд регистрировалась последовательность сообщений, после чего вычислялась средняя нагрузка СПД (Приложение В.1, В.3, В.5, «колонка 3») и средняя скорость передачи низкоприоритетных данных (Приложение В.2, В.4, В.6, «колонка 3»).

На третьем этапе исследовалась вероятностная модель в той же конфигурации (табл. 3.1). Поскольку эта модель оперирует не значениями конкретных сигналов, а лишь распределением вероятности их изменения в течение некоторого времени, для нее в качестве исходных данных использовались

усреднённые параметры экспоненциального распределения вероятностей выхода измеряемой величины за пределы зоны нечувствительности (см. параграф 3.2.1), полученные в результате статистического анализа информации, накопленной ранее на электрических ПС. Из-за использования усреднённых характеристик в каждом наблюдении на выходе вероятностной модели получались одни и те же значения нагрузки СПД (Приложение В.1, В.3, В.5, «колонка 4») и скорости передачи низкоприоритетных данных (Приложение В.2, В.4, В.6, «колонка 4»).

После получения необходимого объёма исходных данных была выполнена оценка адекватности моделей с использованием нескольких критериев, способы применения которых описаны далее.

t-критерий Стьюдента

Общий алгоритм оценки адекватности моделей с использованием *t*-критерия Стьюдента можно представить следующим образом.

1. Вычисляются средние значения откликов модели ($\bar{Y}_{\text{мод}}$) и реальной системы ($\bar{Y}_{\text{эксн}}$) как среднее арифметическое всех значений в соответствующей выборке:

$$\bar{Y}_{\text{мод}} = \frac{1}{N} \sum_{i=1}^N Y_i, \quad (3.27)$$

$$\bar{Y}_{\text{эксн}} = \frac{1}{N} \sum_{i=1}^N Y_i, \quad (3.28)$$

где N – количество наблюдений моделирования и экспериментов на реальной ССОИ;

Y_i – i -е значение в выборке.

Если с точки зрения исследуемых процессов средние значения близки, то можно считать, что модель в среднем воспроизводит реальное поведение системы.

2. Вычисляются дисперсии S^2 откликов модели и реальной системы:

$$S_{\text{мод}}^2 = \frac{1}{N-1} \sum_{i=1}^N \left(Y_i - \bar{Y}_{\text{мод}} \right)^2, \quad (3.29)$$

$$S_{\text{эксн}}^2 = \frac{1}{N-1} \sum_{i=1}^N \left(Y_i - \bar{Y}_{\text{эксн}} \right)^2. \quad (3.30)$$

Если дисперсии близки, значит, модель правильно воспроизводит вариативность данных.

3. Если средние значения и дисперсии близки, проверяется их различие с помощью t -критерия Стьюдента, вычисляемого как:

$$t = \frac{\bar{Y}_{\text{мод}} - \bar{Y}_{\text{эксн}}}{\sqrt{\frac{S_{\text{мод}}^2 + S_{\text{эксн}}^2}{N}}}. \quad (3.31)$$

4. Находится табличное значение t -критерия в соответствии с уровнем значимости a (обычно, $a = 0,05$ [82]) и числом степеней свободы df . Число степеней свободы рассчитывают с использованием формулы Саттерауэйта (3.32) и округляют до целого:

$$df = \frac{\left(\frac{S_{\text{мод}}^2 + S_{\text{эксн}}^2}{N}\right)^2}{\frac{\left(\frac{S_{\text{мод}}^2}{N}\right)^2}{N-1} + \frac{\left(\frac{S_{\text{эксн}}^2}{N}\right)^2}{N-1}}. \quad (3.32)$$

5. Рассчитанный по формуле (3.31) t -критерий сравнивается с табличным [82]. Если $t > t_{\text{табл}}$ – модель неадекватная (должна быть отвергнута, как недостаточно точная), если $t \leq t_{\text{табл}}$ – адекватная.

F-критерий Фишера

Оценка адекватности моделей с использованием F -критерия Фишера выполняется следующим образом.

1. Рассчитывается F -критерий Фишера и число степеней свободы для двух выборок:

$$F = \frac{S_{\text{max}}^2}{S_{\text{min}}^2}, \quad (3.33)$$

$$dfN = N - 1, \quad (3.34)$$

где S_{max}^2 – большая из двух дисперсий;

S_{min}^2 – меньшая из двух дисперсий.

2. Рассчитанный по формуле (3.33) F -критерий Фишера сравнивается с табличным $F_{\text{табл}}$ [82]. Если $F > F_{\text{табл}}$ – модель неадекватная (должна быть отвергнута, как недостаточно точная), если $F \leq F_{\text{табл}}$ – адекватная.

Дополнительные оценки

1. Рассчитывается средняя относительная погрешность между выборками и делается вывод о её допустимости при решении рассматриваемого класса задач:

$$\delta_{\text{сред}} = \frac{1}{N} \sum_{i=1}^N \left| \frac{Y_{i_эксн} - \bar{Y}_{\text{мод}}}{Y_{i_эксн}} \right| \times 100\% . \quad (3.35)$$

2. Рассчитывается разница (в процентах) между минимальным и максимальным значением выборок и делается вывод о её допустимости:

$$\delta_{\text{max}} = \frac{|Y_{\text{min}} - Y_{\text{max}}|}{\frac{Y_{\text{min}} + Y_{\text{max}}}{2}} \times 100\% . \quad (3.36)$$

3. Определяется доверительный интервал, показывающий диапазон, в котором с заданной вероятностью P (обычно $P=0,95$ [82]) находится истинное среднее значение экспериментальной выборки:

$$CI_{\text{эксн}} = \bar{Y}_{\text{эксн}} \pm t_{a/2} \frac{S_{\text{эксн}}}{\sqrt{N}} . \quad (3.37)$$

где $S_{\text{мод}}$ – стандартное отклонение выборки моделирования, определяется как $\sqrt{S_{\text{мод}}^2}$;

$S_{\text{эксн}}$ – стандартное отклонение выборки натурального эксперимента, определяется как $\sqrt{S_{\text{эксн}}^2}$;

$t_{a/2}$ – квантиль распределения Стьюдента для заданного уровня значимости $a = 1 - P$.

4. Если вычисленное среднее значение $\bar{Y}_{\text{мод}}$ попадает в доверительный интервал $CI_{\text{эксн}}$, то делается вывод о хорошем совпадении результатов моделирования с экспериментальными данными.

| Скорость <i>CAN FD</i> , кбит/с | 500 | | 200 | | 100 | |
|---|-----------------|-------------------------|-----------------|-------------------------|-----------------|-------------------------|
| | Нагрузка СПД | Скорость <i>USDO</i> | Нагрузка СПД | Скорость <i>USDO</i> | Нагрузка СПД | Скорость <i>USDO</i> |
| F (3.33) | 1,03 | 1,06 | 1,08 | 1,07 | 1,02 | 1,01 |
| $F \leq F_{\text{табл}}$ | да | да | да | да | да | да |
| $\delta_{\text{сред.}}$, % (3.35) | 0,16 | 0,21 | 0,39 | 0,24 | 0,26 | 0,46 |
| $\delta_{\text{макс.}}$, % (3.36) | 0,53 | 0,66 | 1,26 | 0,69 | 0,97 | 1,17 |
| Доверительный интервал, $CI_{\text{эксн}}$ (3.37) | 23,81±0,32 % | 27501±194 байт/с | 48,08±0,43 % | 8090±66 байт/с | 76,35±0,53 % | 2270±26 байт/с |
| $\bar{Y}_{\text{мод}} \in CI_{\text{эксн}}$ | да | да | да | да | да | да |

Анализируя полученные результаты оценки, можно отметить хорошее совпадение средних значений выборок ($\bar{Y}_{\text{мод}}$, $\bar{Y}_{\text{эксн}}$) и среднеквадратичных отклонений ($S_{\text{мод}}^2$, $S_{\text{эксн}}^2$), что предварительно указывает на адекватность модели.

Сравнение рассчитанных t -критериев Стьюдента с табличным значением позволяет сделать вывод об адекватности модели. Аналогичный результат дает использование F -критерия Фишера.

Средняя относительная погрешность моделирования ($\delta_{\text{сред}}$) во всех случаях не превышает 0,46 %, что применительно к решаемой задаче можно считать очень хорошим результатом. Максимальная относительная погрешность моделирования ($\delta_{\text{макс}}$) равна 1,26 %, что также является хорошим результатом. Средние значения выборок $\bar{Y}_{\text{мод}}$, полученные в результате моделирования, во всех случаях попадают в доверительный интервал CI экспериментальной выборки.

Обобщая промежуточные выводы можно сделать итоговый вывод об адекватности имитационной модели и хорошем совпадении результатов моделирования с экспериментальными данными.

Аналогичным образом была выполнена оценка адекватности вероятностной модели. Поскольку в этом случае среднеквадратичное отклонение результатов моделирования равно нулю, F -критерий Фишера не использовался, так как в выражении (3.33) возникает деление на ноль. Число степеней свободы $df = 29$ (согласно (3.32)), в этом случае табличное значение t -критерия Стьюдента $t_{\text{табл}}$ выбрано равным 1,669 [82]. Результаты оценки приведены в таблице 3.3.

Таблица 3.3 – Результаты оценки адекватности вероятностной модели.

| Скорость <i>CAN FD</i> , кбит/с | 500 | | 200 | | 100 | |
|--|-----------------|---------------------|-----------------|-------------------|-----------------|-------------------|
| | Нагрузка СПД | Скорость USDO | Нагрузка СПД | Скорость USDO | Нагрузка СПД | Скорость USDO |
| $\bar{Y}_{\text{мод}}$ (3.27) | 23,74 % | 27530 байт/с | 48,28 % | 8130 байт/с | 75,92 % | 2269 байт/с |
| $\bar{Y}_{\text{эксп}}$ (3.28) | 23,81 % | 27501 байт/с | 48,08 % | 8090 байт/с | 76,35 % | 2270 байт/с |
| $\bar{Y}_{\text{мод}} \approx \bar{Y}_{\text{эксп}}$ | да | да | да | да | да | да |
| $S_{\text{мод}}^2$ (3.29) | 0 | 0 | 0 | 0 | 0 | 0 |
| $S_{\text{эксп}}^2$ (3.30) | 0,733 | 269403 | 1,31 | 30366 | 1,988 | 4953 |
| t (3.31) | 0,433 | 0,302 | 0,941 | 1,271 | 1,65 | 0,08 |
| $t \leq t_{\text{табл}}$ | да | да | да | да | да | да |
| $\delta_{\text{сред.}}, \%$ (3.35) | 2,83 | 1,68 | 1,94 | 1,59 | 1,43 | 2,46 |
| $\delta_{\text{мах.}}, \%$ (3.36) | 7,66 | 3,92 | 6,52 | 4,86 | 5,44 | 5,80 |
| Доверительный интервал, CI (3.37) | 23,81±0,32 % | 27501±194 байт/с | 48,08±0,43 % | 8090±66 байт/с | 76,35±0,53 % | 2270±26 байт/с |
| $\bar{Y}_{\text{мод}} \in CI_{\text{эксп}}$ | да | да | да | да | да | да |

Анализируя полученные результаты оценки вероятностной модели, можно отметить хорошее совпадение средних значений выборок ($\bar{Y}_{\text{мод}}$, $\bar{Y}_{\text{экс}}$), это позволяет предварительно говорить об адекватности модели. В пользу этого утверждения также говорит сравнение рассчитанных t -критериев Стьюдента с табличным значением – выполняется условие $t \leq t_{\text{табл}}$.

Средняя относительная погрешность моделирования ($\delta_{\text{сред}}$) во всех случаях не превышает 2,83 %, что применительно к задаче, решаемой вероятностной моделью (предварительная оперативная оценка основных параметров межмодульного взаимодействия), можно считать хорошим результатом. Максимальная относительная погрешность моделирования ($\delta_{\text{макс}}$) равна 7,66 %, что также является удовлетворительным результатом. Средние значения выборок $\bar{Y}_{\text{мод}}$, полученные в результате моделирования, во всех случаях попадают в доверительный интервал CI экспериментальной выборки.

Выполнив анализ результатов оценки, можно обоснованно говорить об адекватности вероятностной модели с приемлемым совпадением (в рамках решаемых задач) результатов моделирования и экспериментальных данных.

Таким образом подтверждена адекватность разработанных вероятностной и имитационной моделей.

3.5 Выводы по главе 3

В третьей главе диссертации получены следующие основные результаты.

1. Обоснована необходимость оценки параметров межмодульного взаимодействия на как можно более ранних этапах проектирования модульной системы. Выявлен один из основных факторов, влияющих на возможность правильного функционирования ССОИ – нагрузка на сеть передачи данных.

2. Разработан метод оценки времени передачи сообщений по шине *CAN FD*, формируемых в процессе функционирования модулей ССОИ, отличающийся от известных тем, что позволяет определить верхнюю, нижнюю и среднюю оценки времени доставки с учетом дополнительных битов. На основе анализа

стандартного и расширенного формата сообщений *CAN FD* составлены и описаны выражения, позволяющие определить минимальное и максимальное время передачи сообщения для наиболее и наименее благоприятных случаев соответственно.

3. На основе анализа ССОИ как системы массового обслуживания и с использованием метода оценки времени передачи сообщений разработан усовершенствованный вероятностный метод оценки основных параметров межмодульного взаимодействия: нагрузки сети и скорости передачи низкоприоритетных данных.

4. С использованием усовершенствованного вероятностного метода в среде *MatLab* реализована вероятностная модель для оперативной оценки параметров межмодульного взаимодействия, позволяющая исследовать различные конфигурации ССОИ и влияние параметров настройки входящих в неё модулей на нагрузку сети.

5. Разработана имитационная модель для детального исследования межмодульного взаимодействия в ССОИ на базе интерфейса *CAN FD*, использующая метод оценки времени передачи сообщений и *XML*-описание системы.

6. С использованием t -критерия Стьюдента и F -критерия Фишера доказана адекватность разработанных моделей.

4 ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ РАЗРАБОТАННЫХ МЕТОДОВ И МОДЕЛЕЙ

4.1 Использование усовершенствованной вероятностной модели для оценки параметров межмодульного взаимодействия в контроллере присоединения электрической подстанции 10-110 кВ

Для анализа характеристик функционирования проектируемой ССОИ ПС в условиях передачи осциллограмм токов и напряжений в режиме реального времени была использована усовершенствованная вероятностная модель (параграф 3.2.3). Цель моделирования заключалась в исследовании влияния конфигурации модулей ССОИ реальной ПС напряжением 10-110 кВ и скорости работы интерфейса межмодульного взаимодействия на параметры сети передачи данных (СПД).

Исследуемая ССОИ включает в себя пять функционально различающихся типов модулей (табл. 4.1).

Таблица 4.1 – Типы модулей ССОИ ПС 10-110 кВ

| Тип, № | Наименование модуля | Количество модулей в ССОИ |
|--------|--|---------------------------|
| 1 | Процессорный модуль | 1 |
| 2 | Модуль измерения линейных токов и напряжений | 2 |
| 3 | Модуль аналогового ввода | 10 |
| 4 | Модуль дискретного ввода | 8 |
| 5 | Модуль дискретного вывода | 2 |

Для процессорного модуля определена периодичность формирования высокоприоритетных синхронизационных сообщений *SYNC* (сообщения класса 1), которая составляет $T_{SYNC} = 100$ мс.

Остальные модули системы передают от 1 до 4 сообщений *PDO* (классы 2-5), при этом каждое передаваемое сообщение включает от 1 до 24 индивидуальных значений, среди которых присутствуют:

- измеренные параметры (мгновенные значения токов и напряжений);
- вычисленные параметры (линейные токи и напряжения, полная мощность и т.д.);
- дискретные флаги состояния (положение КА, сигналы о срабатывании защит или изменении режимов работы).

Передача этих данных осуществляется периодически с частотой 1 раз в секунду, однако возможен и асинхронный режим – в случае изменения хотя бы одного из передаваемых значений сообщение *PDO* формируется и передается досрочно.

В дополнение к вышесказанному, каждый модуль сети раз в секунду генерирует сообщения *HBEAT* (класс 6), предназначенные для контроля целостности системы и мониторинга состояния узлов.

Помимо передачи *PDO*, процессорный модуль выполняет обмен служебной информацией с каждым модулем сети по протоколу *USDO* (сообщения класса 7), что необходимо для поддержания работоспособности системы, корректного выполнения функций управления и приёма-передачи низкоприоритетных данных сравнительно большого объёма.

Для реализации всех возможностей интерфейса *CAN FD* использованы две скорости передачи: номинальная (1 Мбит/с) и альтернативная (8 Мбит/с).

На основе описанного режима работы ССОИ были сформированы исходные параметры вероятностной модели (табл. 4.2).

Таблица 4.2 – Исходные параметры вероятностной модели.

| Параметр | Обозначение параметра в модели | Обозначение матрицы в модели | Значение | Пояснение |
|--------------------------|--------------------------------|------------------------------|------------------|-------------------------------------|
| Классы заявок | <i>H</i> | — | 7 | <i>SYNC, PDO (2-5), HBEAT, USDO</i> |
| Количество типов модулей | <i>M</i> | — | 5 | Общее количество типов модулей |
| Распределение по типам | — | <i>M</i> | [1, 2, 10, 8, 2] | Распределение модулей по типам |

| Параметр | Обозначение параметра в модели | Обозначение матрицы в модели | Значение | Пояснение |
|-------------------------------|--------------------------------|------------------------------|------------|---|
| Периодичность отправки заявок | — | B | Изменяемое | Определяет временные характеристики передачи данных |
| Длина сообщения | — | D | Изменяемое | Рассчитывает размерность передаваемых сообщений |
| Кол-во значений в сообщении | — | V | Изменяемое | Отражает кол-во отдельных значений в каждом сообщении |
| Интенсивность передачи | — | LM | Изменяемое | Задаёт режим передачи |

Значение «изменяемое» указывает на возможность исследования работы данной конфигурации ССОИ с различными параметрами формирования и передачи сообщений. Коэффициент автоматической вставки дополнительных бит η принят равным 0.491 на основе анализа большого числа сообщений в реальной ССОИ на базе интерфейса *CAN FD* и с учётом сведений, приведённых в [77, 78].

Для количественного определения параметра экспоненциального распределения α проведен детальный анализ множества ранее зарегистрированных графиков изменения электрических параметров ПС на продолжительных временных интервалах с целью оценки вероятности выхода измеряемого значения за пределы диапазона заданной ширины. В результате анализа получена зависимость параметра α от ширины зоны нечувствительности δ (рис. 4.1). Для дальнейшего использования данная зависимость была аппроксимирована функцией (4.1):

$$\alpha = 3.616 \cdot e^{-1.898} + 0.0609. \quad (4.1)$$

Результаты моделирования

На первом этапе моделирования работы ССОИ ПС 10-110 кВ была получена зависимость загрузки СПД от ширины зоны нечувствительности для разных режимов работы ССОИ ($StressMode = 0$) и ($StressMode = 1$), подробно

рассмотренных в параграфе 3.2.3. Полученные зависимости представлены на рис. 4.2а.

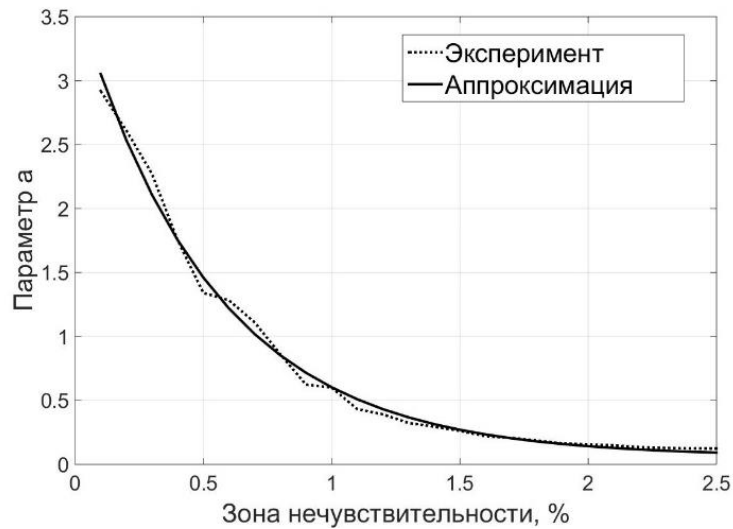


Рисунок 4.1 – Зависимость усреднённого параметра экспоненциального распределения от ширины зоны нечувствительности

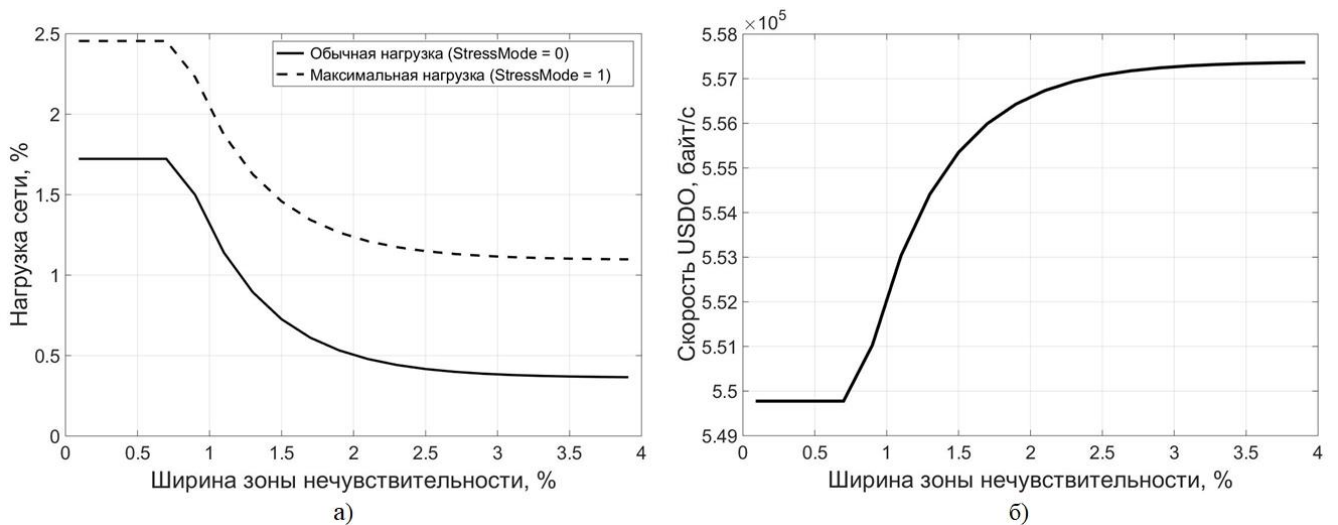


Рисунок 4.2 – Влияние ширины зоны нечувствительности на параметры СПД:
а – нагрузка y ; б – скорость передачи низкоприоритетных данных Q_H

Анализируя полученные зависимости, нетрудно заметить, что при значениях $\delta > 3\%$ влияние данного параметра на нагрузку СПД становится несущественным, тогда как при $\delta < 0,7\%$ нагрузка сети достигает максимума и перестает изменяться. Для рассмотренных режимов работы нагрузка на сеть составляет всего 2,47 % при максимально возможной скорости передачи данных.

Далее была получена зависимость скорости передачи низкоприоритетных данных (осциллограмм) Q_H от ширины зоны нечувствительности δ в условиях обычной нагрузки ($StressMode = 0$) (рис. 4.26). Из графика видно, что при максимальной нагрузке на СПД (при $\delta = 0,7\%$), низкоприоритетные данные передаются со скоростью около 550 кбайт/с. В случае, если низкоприоритетными данными являются осциллограммы (одна осциллограмма имеет размер 4 Кбайта), то она будет передана примерно за 7,3 мс. Таким образом, в режиме реального времени с периодичностью обновления 100 мс СПД может обеспечить передачу до 13 осциллограмм.

Дальнейший анализ результатов моделирования позволил сделать следующие выводы.

– Ширину зоны нечувствительности δ правильно будет выбрать равной 0,7 %, поскольку дальнейшее её уменьшение не увеличивает частоту передачи данных, а увеличение δ не оказывает значительного влияния на снижение нагрузки на СПД, однако снижает чувствительность ССОИ к изменениям электрических параметров.

– Для всех исследованных режимов работы коэффициент простоя СПД ρ остается выше 97 %, что свидетельствует о наличии значительного запаса пропускной способности сети. В таком случае весь запас можно использовать для передачи низкоприоритетных данных, а если такой необходимости нет, то целесообразно снизить скорость работы интерфейса *CAN FD*. Вследствие такого решения повысится помехоустойчивость шины и снизится тепловыделение модулей ССОИ.

При анализе результатов первого этапа моделирования возникла необходимость оценить зависимость параметров сети от скорости передачи данных по шине *CAN FD* при ширине зоны нечувствительности $\delta = 0,7\%$. Результаты оценки представлены на рис. 4.3.

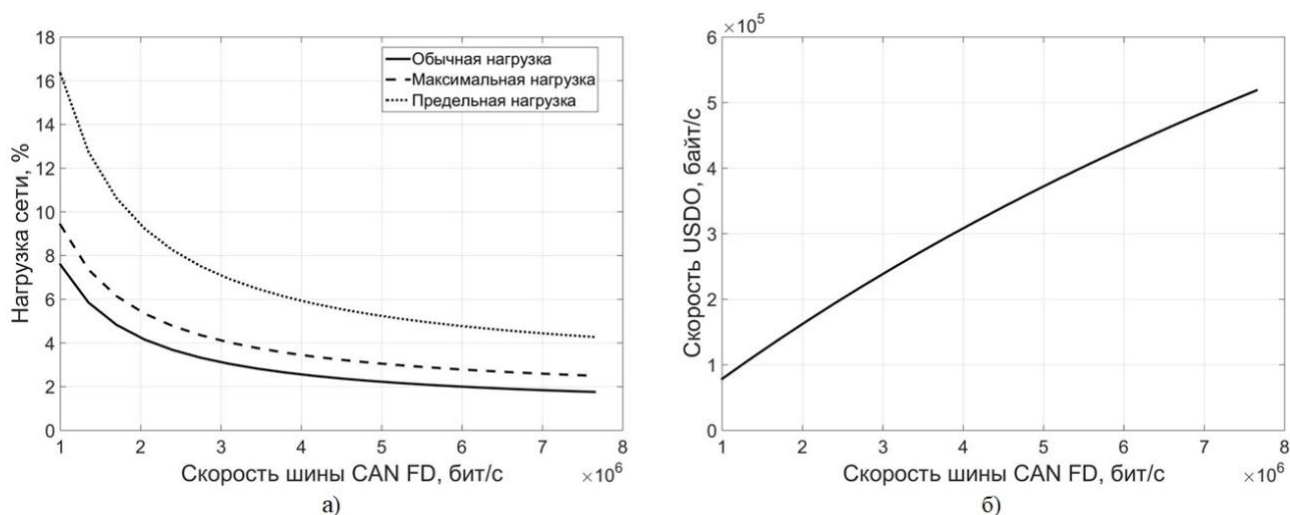


Рисунок 4.3 – Влияние скорости шины *CAN FD* на параметры СПД: а – нагрузка; б – скорость передачи низкоприоритетных данных

Выполнив анализ результатов, представленных на рис. 4.3а, можно сделать вывод, что даже при восьмикратном снижении скорости передачи данных в сети *CAN FD* (до 1 Мбит/с) не наблюдается возникновение перегрузки сети при передаче высокоприоритетных сообщений. Предельная нагрузка не превышает 17 % в текущей конфигурации ССОИ, что гарантирует своевременную доставку критически важных высокоприоритетных данных.

Рисунок 4.3б показывает практически линейную зависимость скорости передачи низкоприоритетных данных от скорости работы шины *CAN FD*. При снижении скорости работы интерфейса до минимального значения 1 Мбит/с скорость передачи низкоприоритетных данных уменьшается с 550 кбайт/с до 86,95 кбайт/с. Однако даже при такой скорости СПД сохраняет способность передавать до двух осциллограмм в режиме реального времени.

На основании полученных данных был сделан вывод, что для рассматриваемой конфигурации ССОИ целесообразно выбрать скорость работы интерфейса *CAN FD* из диапазона 2...4 Мбит/с. Это позволит снизить тепловыделение приёмопередатчиков *CAN FD* за счёт соответствующего уменьшения их тактовой частоты в 4...2 раза, при этом сохранится возможность передачи низкоприоритетных данных без ухудшения эксплуатационных характеристик системы.

Подход к использованию вероятностной модели и полученные с её помощью результаты докладывались на международной конференции [88].

4.2 Использование усовершенствованной вероятностной модели для оценки параметров взаимодействия устройств управления дугогасящими реакторами

Область применения разработанных методов и моделей не ограничивается исследованием ССОИ ПС, они могут успешно использоваться для изучения других модульных систем на основе интерфейса *CAN FD* и протокола *CANopen FD*. В частности, автором выполнено моделирование подобной системы, объединяющей устройства управления дугогасящими реакторами.

Дугогасящий реактор (ДГР) предназначен для компенсации емкостных токов замыкания на землю в сетях 6...35 кВ с изолированной нейтралью. Его основная функция заключается в уменьшении ионизации дуги, возникающей при однофазных замыканиях на землю, и создании условий для её быстрого гашения, что повышает надёжность и стабильность работы сети [89, 90, 91, 92] (рис. 4.4).

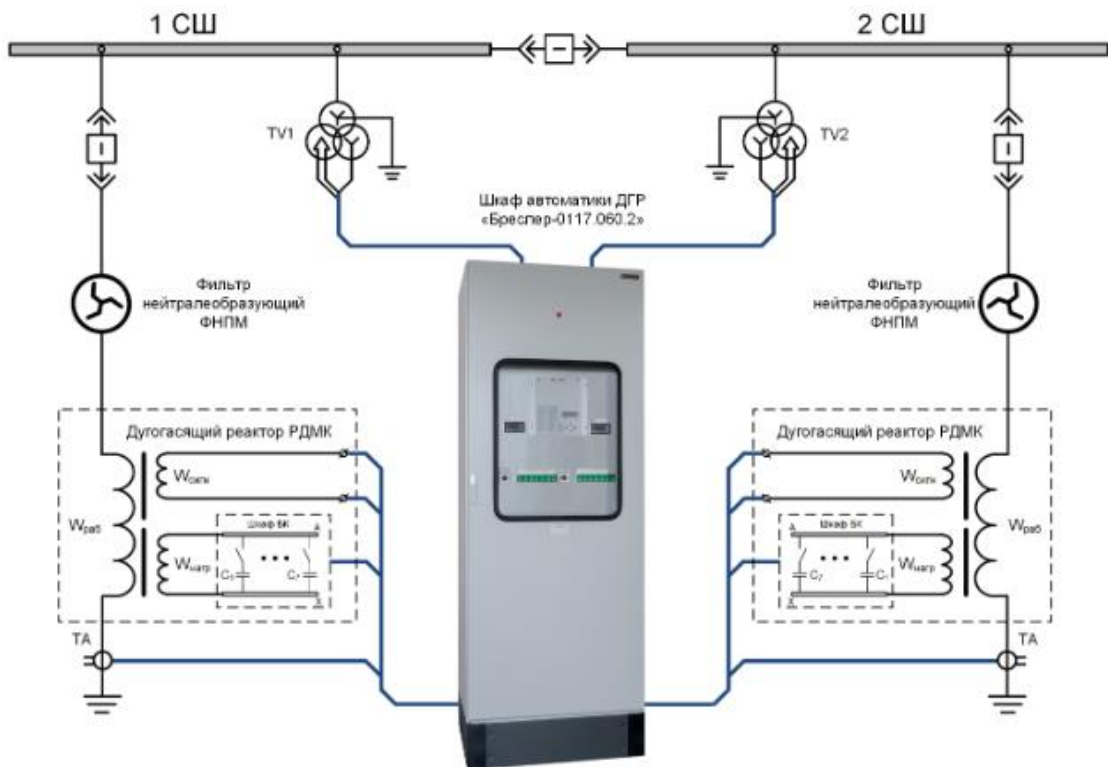


Рисунок 4.4 – Использование дугогасящих реакторов

ДГР представляет собой регулируемую катушку, путём изменения индуктивности которой выполняется настройка контура нулевой последовательности сети (КНПС) в резонанс с промышленной частотой 50 Гц. Для изменения индуктивности используется микропроцессорное устройство управления (УУ). УУ должно измерять текущий уровень расстройки КНПС [93, 94, 95] и формировать сигналы управления приводом сердечника ДГР с целью приведения этого уровня к номинальному значению.

На одной ПС могут использоваться несколько ДГР, причем в ряде случаев необходима их согласованная работа. В связи с этим УУ разных ДГР должны обмениваться друг с другом информацией о состоянии контролируемой сети и управляемого реактора. Кроме того, эту информацию необходимо передавать на АРМ оператора. Данные, которыми обмениваются УУ между собой и АРМ, включают текущие значения электрических параметров, осциллограммы, диагностические и событийные данные. Также УУ принимают от АРМ управляющие команды, профили работы, сигналы синхронизации и иную служебную информацию.

Для организации такого взаимодействия УУ, разработанные с участием автора [89], объединяются в сеть на базе интерфейса *CAN FD* и протокола *CANopen FD*. Общая протяженность этой сети может достигать нескольких сотен метров, причем чем ниже скорость передачи данных, тем больше максимально возможная длина линии связи. В связи с этим возникла задача определения минимально допустимой скорости интерфейса *CAN FD*, обеспечивающей своевременную доставку высокоприоритетных данных и приемлемую скорость обмена низкоприоритетными данными между УУ ДГР и АРМ оператора, находящимися на значительном расстоянии.

Для решения поставленной задачи была использована разработанная вероятностная модель (параграф 3.2.3). Исследуемая сеть ДГР включает в себя два функционально различающихся типа устройств (табл. 4.3).

Таблица 4.3 – Типы устройств сети ДГР

| Тип, № | Наименование устройства | Количество устройств в сети |
|--------|-------------------------|-----------------------------|
| 1 | Ведущие (АРМ оператора) | 1 |
| 2 | УУ ДГР | 18 |

Ведущее устройство формирует высокоприоритетные сообщения синхронизации *SYNC* (сообщения класса 1) с периодичностью $T_{SYNC} = 1000$ мс.

Остальные устройства сети в ответ на сообщение синхронизации могут передавать до 5 сообщений *PDO* (классы 2–6), при этом каждое передаваемое сообщение включает до 2 индивидуальных значений, среди которых присутствуют:

- напряжение и ток в контуре нулевой последовательности;
- текущий уровень расстройки КНПС;
- частота сети;
- положение регулирующего элемента;
- время последней и следующей корректировки резонансной настройки.

Передача этих данных осуществляется периодически с частотой 1 раз в секунду, за исключением данных о времени последней и последующей корректировки, которые передаются раз в минуту. Кроме того, каждое устройство сети раз в секунду генерирует сообщения *HBEAT* (класс 7), предназначенные для контроля целостности системы и мониторинга состояния узлов.

Помимо передачи *PDO*, ведущий модуль выполняет обмен низкоприоритетной информацией с каждым УУ по протоколу *USDO* (сообщения класса 8). Эта информация включает осциллограммы токов и напряжений, а также команды настройки УУ.

Скорость работы интерфейса *CAN FD* вначале была выбрана максимально возможной, т.е. номинальная скорость задана равной 1 Мбит/с, альтернативная – 8 Мбит/с.

Аналогично параграфу 4.1, на основе описанного режима работы СПД были сформированы исходные параметры вероятностной модели (табл. 4.4).

Коэффициент автоматической вставки дополнительных бит η соответствует принятому в параграфе 4.1. Параметр экспоненциального распределения a при моделировании не использовался, поскольку в рассматриваемой системе досрочная передача PDO не предусмотрена.

Таблица 4.4 – Исходные параметры вероятностной модели.

| Параметр | Обозначение параметра в модели | Обозначение матрицы в модели | Значение | Пояснение |
|-------------------------------|--------------------------------|------------------------------|------------|---|
| Классы заявок | H | — | 8 | $SYNC, PDO (2-6), HBEAT, USDO$ |
| Количество типов модулей | M | — | 2 | Общее количество типов модулей |
| Распределение по типам | — | M | [1, 18] | Распределение модулей по типам |
| Периодичность отправки заявок | — | B | Изменяемое | Определяет временные характеристики передачи данных |
| Длина сообщения | — | D | Изменяемое | Рассчитывает размерность передаваемых сообщений |
| Кол-во значений в сообщении | — | V | Изменяемое | Отражает кол-во отдельных значений в каждом сообщении |
| Интенсивность передачи | — | LM | Изменяемое | Задаёт режим передачи |

Результаты моделирования.

На первом этапе моделирования работы сети была получена зависимость нагрузки СПД от скорости работы интерфейса $CAN FD$ для разных режимов работы ($StressMode = 0$) и ($StressMode = 1$) в максимально возможном диапазоне скоростей. Результаты моделирования представлены на рис. 4.5.

Анализируя полученные графики нетрудно заметить, что в данной конфигурации сети нецелесообразно использовать максимально возможную скорость работы интерфейса $CAN FD$, т.к. в таком случае предельная нагрузка на СПД составляет всего 0,42 %. Более того, десятикратное снижение скорости

работы *CAN FD* (до 1 Мбит/с) повышает нагрузку на СПД незначительно – всего до 1,1 %.

Такая низкая нагрузка на СПД позволяет еще сильнее снизить скорость работы *CAN FD*, увеличив тем самым максимальную длину линии связи. Согласно [23], при скорости работы *CAN FD* = 1 Мбит/с длина линии связи может достигать лишь 40 метров, а при 12 кбит/с – свыше 3000 метров.

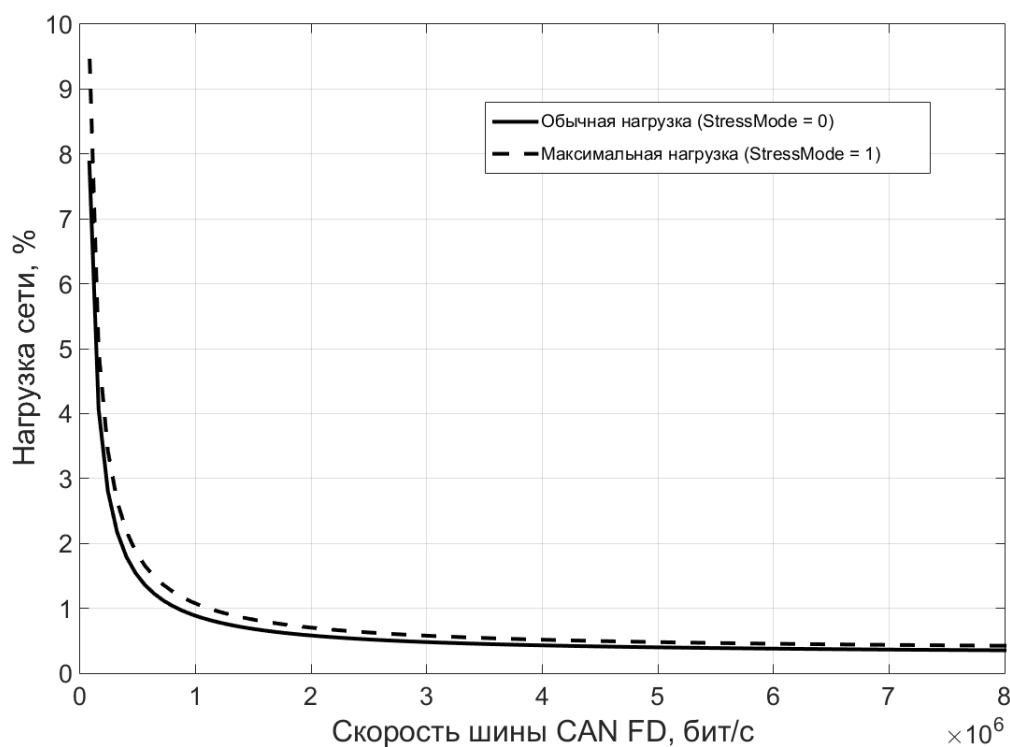


Рисунок 4.5 – Зависимость нагрузки на СПД от скорости работы шины *CAN FD* (до 8 Мбит/с)

На следующем этапе моделирования было исследовано, приводит ли использование скорости 12 кбит/с к перегрузке СПД. Результаты моделирования представлены на рис. 4.6.

Из графика видно, что использование скорости шины *CAN FD* в 12 кбит/с не приводит к перегрузке СПД даже в режиме максимальной нагрузки (*StressMode* = 1), однако нагрузка на СПД при этом достигает 89 %, т.е. практически безопасного максимума. В связи с этим целесообразно поднять скорость работы интерфейса *CAN FD* до 20 кбит/с, обеспечив тем самым умеренную нагрузку на СПД (53,4 % при *StressMode* = 0 и 64,5 % *StressMode* = 1) и запас для передачи

низкоприоритетных данных (осциллограмм токов и напряжений). Максимальная длина линии связи при этом уменьшится незначительно – примерно до 2950 м.

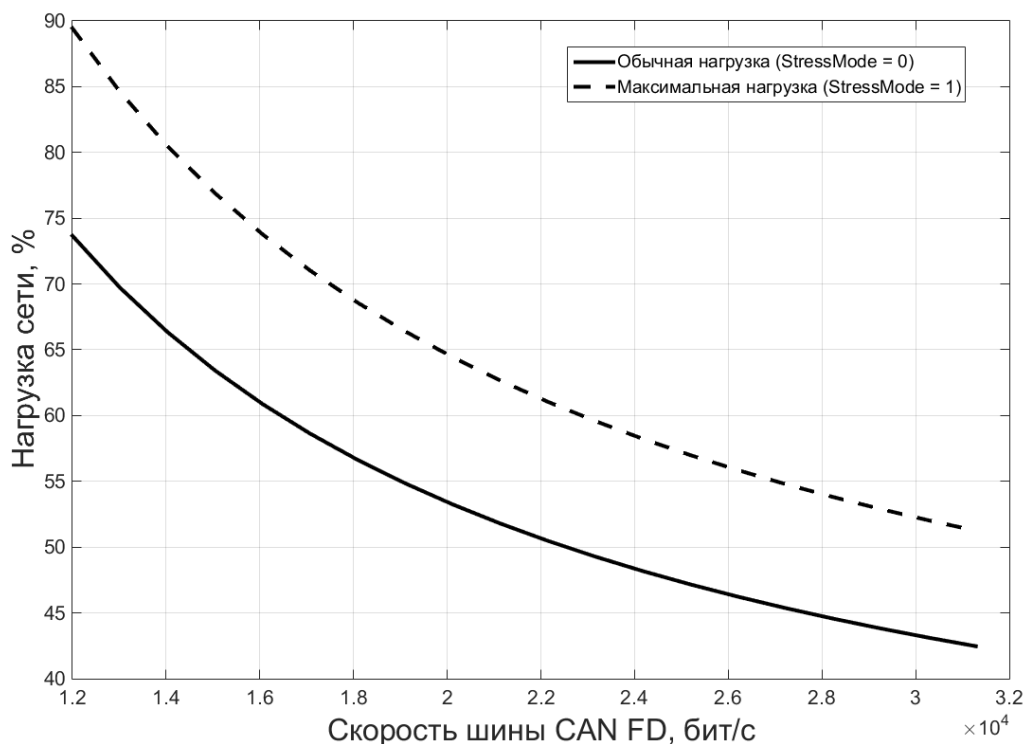


Рисунок 4.6 – Зависимость нагрузки на СПД от скорости работы шины *CAN FD* (от 12 кбит/с)

Далее была получена зависимость скорости передачи низкоприоритетных данных (*USDO*) от скорости работы шины *CAN FD* (рис. 4.7).

Из графика видно, что при скорости интерфейса *CAN FD* 20 кбит/с низкоприоритетные данные передаются со скоростью около 870 байт/с. В таком режиме одна осциллограмма размером 4 Кбайта будет передана примерно за 4,7 с. Поскольку в рассматриваемой системе наблюдение осциллограмм в режиме реального времени не требуется, а сами осциллограммы используются лишь при лабораторном анализе аварийных ситуаций, этот результат можно считать приемлемым.

На основании результатов моделирования сделан вывод, что для рассматриваемой конфигурации сети УУ ДГР целесообразно выбрать скорость работы интерфейса *CAN FD* равной 20 Кбит/с, что позволит соединить УУ линией связи длиной до 2950 м (с учётом маршрута прокладки кабеля), обеспечив

гарантированную доставку высокоприоритетных данных и приемлемую скорость передачи низкоприоритетных данных.

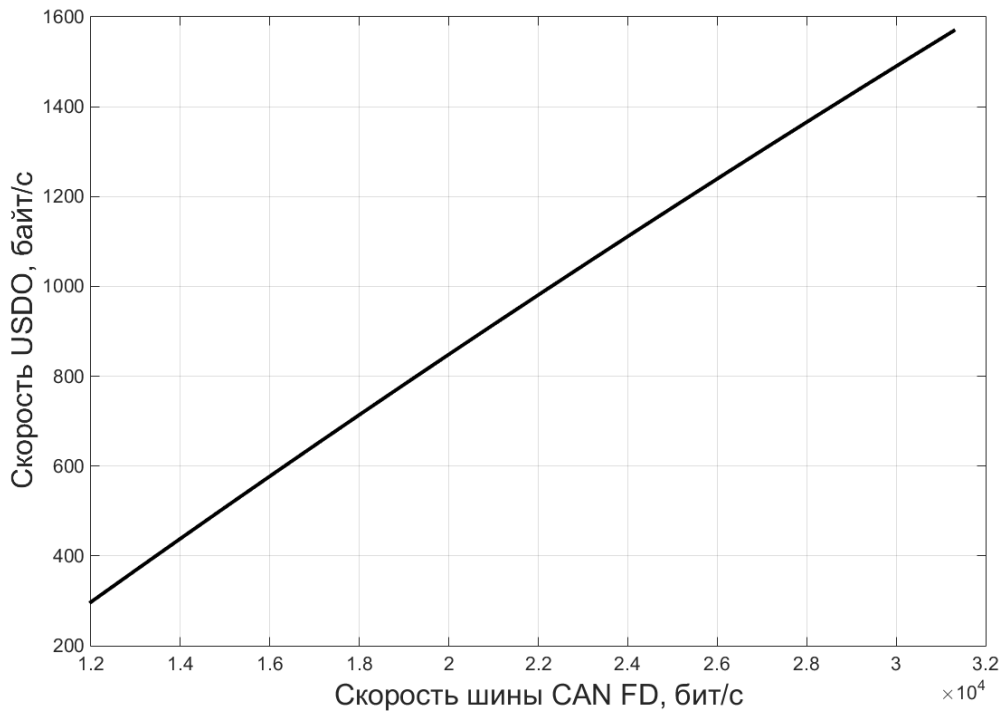


Рисунок 4.7 – Зависимость скорости передачи *USDO* от скорости работы шины *CAN FD*

4.3 Использование метода оценки времени передачи сообщений для исследования межмодульного взаимодействия в регистраторе событий

Универсальность характерна не только для созданных моделей, но и для разработанного метода оценки времени передачи сообщений по шине *CAN FD* (параграф 3.1.2). Хотя он в первую очередь предназначен для использования в вероятностной и имитационной моделях, однако, может применяться и независимо от них. Так, предложенный метод был использован при исследовании работы модульного регистратора событий, предназначенного для записи и анализа событий в системе защиты турбоагрегатов электрических станций.

В регистраторе используется упрощённый протокол межмодульного взаимодействия, подразумевающий периодическое формирование сообщений синхронизации процессорным модулем и ответную передачу фиксированного набора сообщений каждым модулем ввода. Поскольку все сообщения передаются синхронно и безусловно, необходимость использования вероятностной или

имитационной модели отсутствует, достаточно лишь при помощи разработанного метода вычислить суммарную длительность передачи всех сообщений в течение одного цикла синхронизации.

В результате вычислений было выявлено, что замена интерфейса *CAN* на *CAN FD* без изменения скорости передачи существенно снижает нагрузку СПД (с 0.72 до 0.35) в текущей конфигурации изделия, обеспечивая возможность подключения дополнительных модулей ввода данных при сохранении периода их опроса 2 мс (рис. 4.8). Если же подключение дополнительных модулей не требуется, то период опроса можно уменьшить до 1 мс, повысив таким образом скорость реакции регистратора на изменения входных сигналов при сохранении запаса по нагрузке СПД около 30 % на случай возникновения ошибок передачи / приёма отдельных сообщений.

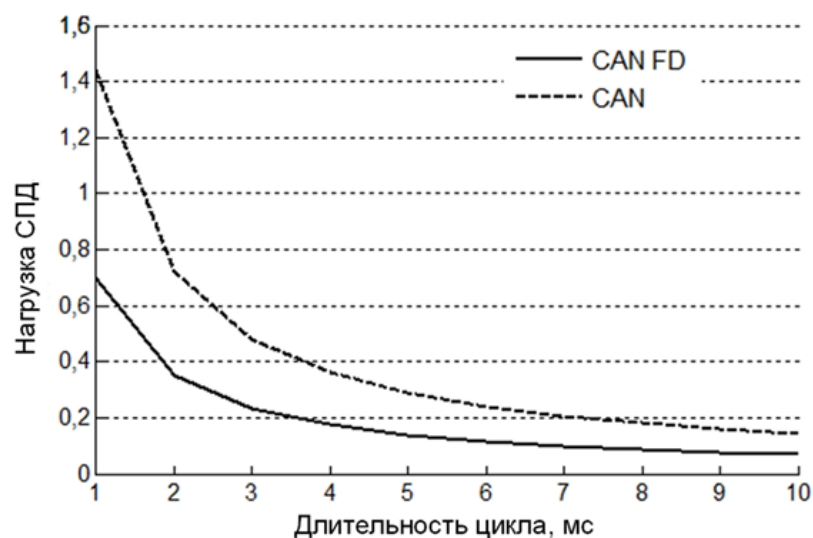


Рисунок 4.8 – Исследование нагрузки СПД модульного регистратора событий

4.4 Аппаратная реализация модулей систем сбора обработки информации подстанции

В соответствии с обоснованной в параграфе 1.4.1 номенклатурой были разработаны функциональные схемы модулей ввода\вывода сигналов ССОИ, использованные при проектировании КП в ООО «СКБ Электронного приборостроения» (Приложение Г).

Модуль ввода аналоговых сигналов

На рис. 4.9 изображена функциональная схема модуля ввода аналоговых сигналов.

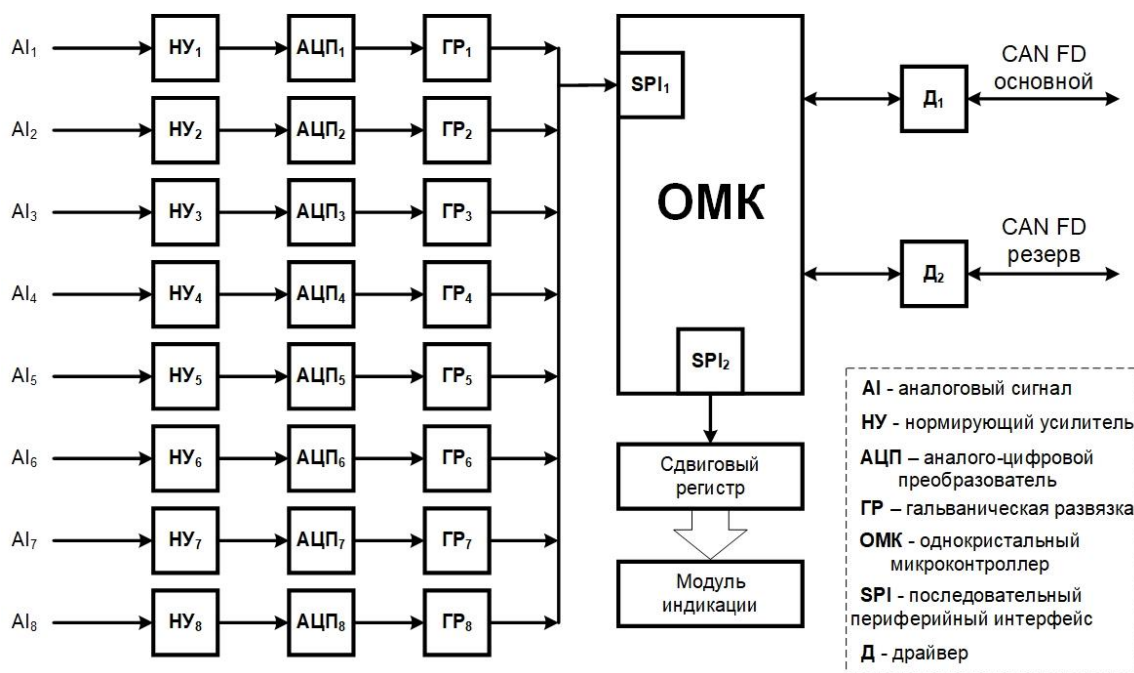


Рисунок 4.9 – Функциональная схема модуля ввода аналоговых сигналов

Модуль содержит 8 линий для ввода измеренных аналоговых значений. Нормирующие усилители $НУ$ преобразуют входные сигналы к виду, пригодному для ввода в аналого-цифровой преобразователь ($АЦП$). Далее аналоговые сигналы преобразуются в цифровую форму с помощью $АЦП$. Для защиты внутренних цепей модуля от недопустимо высокой разности потенциалов на входах и улучшения помехозащищённости модуля используется гальваническая развязка ($ГР$), причём с целью повышения точности измерений она установлена не в аналоговой части схемы, а после каждого $АЦП$ на цифровом последовательном интерфейсе SPI , связывающем $АЦП$ с однокристальным микроконтроллером $ОМК$. Цифровая обработка (фильтрация, усреднение, анализ и т.д.) сигнала выполняется $ОМК$. Результаты обработки в виде сообщений протокола *CANopen FD* передаются через драйверы $Д_1$ и $Д_2$ на основную и резервную шины *CAN FD*. Модуль индикации обеспечивает локальное отображение состояния входных сигналов: превышение пороговых уровней и исправность входов.

Модуль ввода дискретных сигналов

Функциональная схема модуля ввода дискретных сигналов приведена на рис. 4.10.

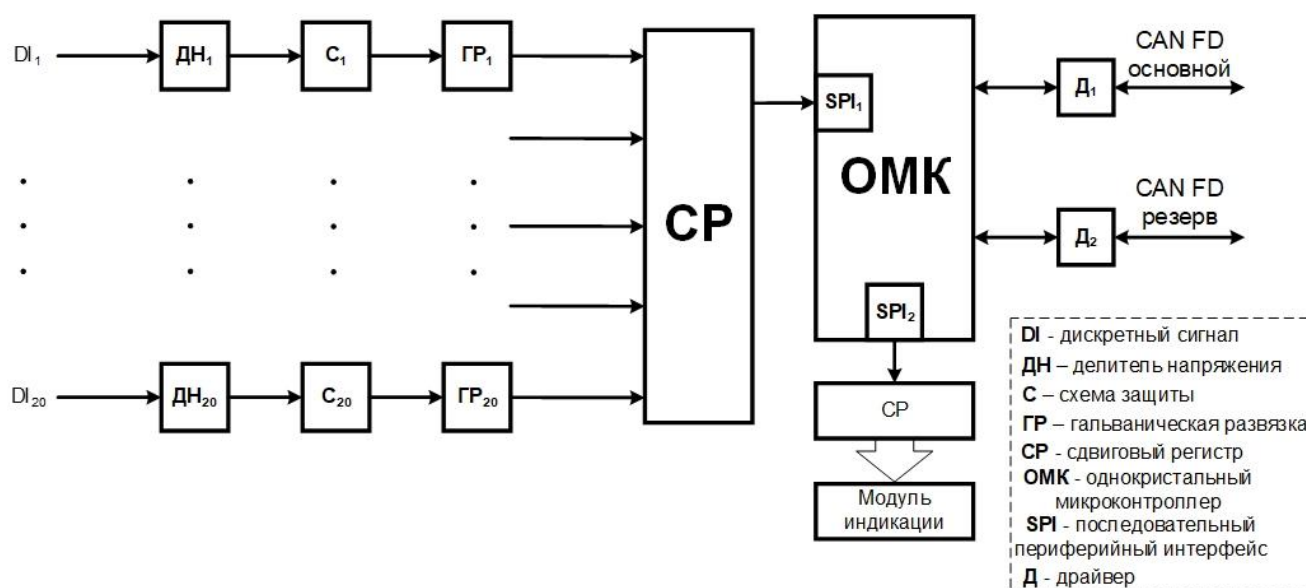


Рисунок 4.10 – Функциональная схема модуля ввода дискретных сигналов

Данная схема содержит 20 входов для ввода дискретных сигналов, снабжённых делителями напряжения (ДН) для приведения сигналов к уровню, пригодному для ввода в ОМК, схемами защиты от перенапряжений и импульсных помех (С) и гальванической развязкой (ГР). Ввиду большого количества входных сигналов для их ввода в ОМК используется сдвиговый регистр (СР), преобразующий 20-разрядный входной параллельный код в последовательный. ОМК выполняет считывание состояния входов, выявляет изменения состояния и передает эти данные в виде сообщений протокола *CANopen FD* на шину *CAN FD*. Аналогично схеме (рис. 4.9) имеется драйвер $Д_2$ резервной линии интерфейса *CAN FD* и модуль индикации.

Модуль вывода дискретных сигналов

Функциональная схема модуля вывода дискретных сигналов показана на рис. 4.11.

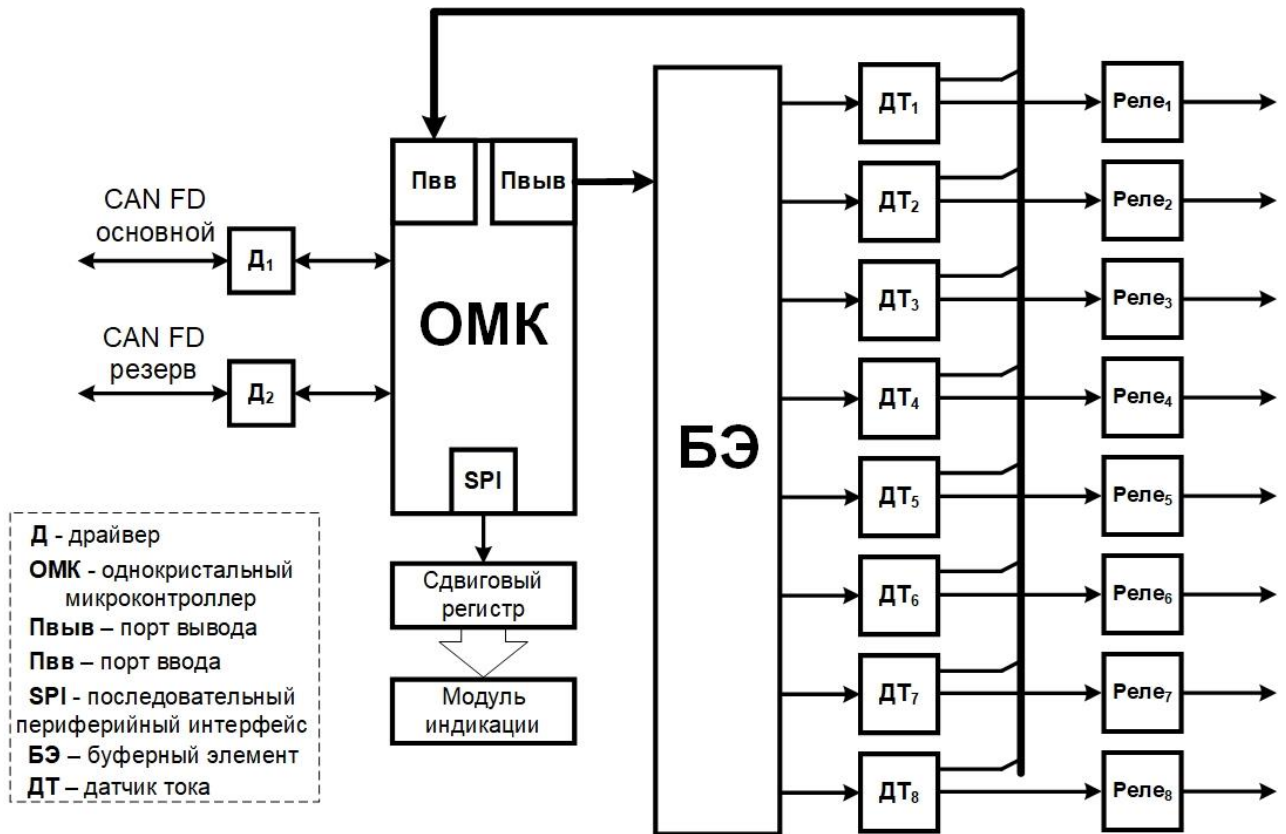


Рисунок 4.11 – Функциональная схема модуля вывода дискретных сигналов

Через основной и резервный драйверы шины *CAN FD* (D_1 и D_2 соответственно) управляющие команды протокола *CANopen FD*, сформированные процессорным модулем, поступают в **ОМК**. **ОМК** преобразует их в состояния выходных сигналов и выдаёт в порт вывода (**Пвыв**). Буферный элемент (**БЭ**) снижает нагрузку на выходы **ОМК**, выполняя функции усиления сигнала и подавления импульсных помех, проникающих в модуль по выходным цепям. Выходы **БЭ** управляют обмотками реле, обеспечивающих формирование выходных сигналов. Датчики тока (**ДТ**) предназначены для контроля исправности обмоток реле. Выходные сигналы **ДТ** через порт ввода **Пвв** поступают в **ОМК** и сопоставляются им с сигналами управления обмотками; в случае несовпадения формируется признак ошибки.

4.5 Выводы по главе 4

В четвёртой главе диссертации получены следующие основные результаты.

1. С использованием разработанной вероятностной модели исследовано межмодульное взаимодействие в ССОИ ПС 10-110 кВ, получены зависимости критически важных параметров взаимодействия от ширины зоны нечувствительности и скорости шины *CAN FD*. На основе анализа результатов моделирования сформулированы рекомендации по уменьшению скорости шины до 2...4 Мбит/с с целью снижения тепловыделения модулей.

2. Показана универсальность разработанных методов и моделей на примере исследования процессов обмена данными в сети устройств управления дугогасящими реакторами. С целью обеспечения максимальной длины линии связи путём анализа результатов моделирования определена минимальная скорость шины *CAN FD* 20 кбит/с, обеспечивающая гарантированную своевременную доставку высокоприоритетных данных и приемлемую скорость передачи низкоприоритетных данных; при этом допустимая длина линии составила 2950 м.

3. Показано, что разработанный метод оценки времени передачи сообщений по шине *CAN FD* может применяться не только в вероятностной и имитационной моделях, но и самостоятельно. Путём использования метода при исследовании процессов межмодульного взаимодействия в регистраторе событий показано, что замена интерфейса *CAN* на *CAN FD* значительно снижает нагрузку на СПД в текущей конфигурации изделия (с 0.72 до 0.35), что обеспечивает возможность подключения дополнительных модулей или уменьшения периода их опроса с сохранением запаса пропускной способности СПД.

4. В соответствии с представленной в параграфе 2.1 номенклатурой разработаны функциональные схемы отдельных модулей ССОИ. Разработанные схемы использованы компанией ООО «СКБ Электронного приборостроения» при проектировании универсального модульного контроллера присоединения.

Заключение

В диссертационной работе в результате выполнения теоретических исследований, разработки методов, моделей, программ и устройств, проведения численных и физических экспериментов решена актуальная научно-техническая задача повышения эффективности функционирования и улучшения технико-экономических характеристик ССОИ путём организации надёжного межмодульного взаимодействия.

Основные результаты диссертационной работы заключаются в следующем.

1. На основе аналитического обзора публикаций, научных трудов и других источников по теме исследования обоснована необходимость использования модульной структуры ССОИ и интерфейса *CAN FD*, реализующего межмодульное взаимодействие. Обоснована необходимость разработки метода описания модульной ССОИ на базе протокола *CANopen FD* и интерфейса *CAN FD*, учитывающего межмодульные информационные связи.

2. Разработан метод описания модульной ССОИ, использующей протокол *CANopen FD* и интерфейс *CAN FD*, позволяющий полностью исключить основные ошибки настройки межмодульного взаимодействия и за счёт этого сократить время конфигурирования системы с нескольких дней до нескольких часов.

3. Разработан метод оценки времени передачи сообщений *CAN FD* для моделирования межмодульного взаимодействия в ССОИ, позволяющий исключить погрешность оценки времени передачи сообщений, связанную с наличием дополнительных битов. У известных методов эта погрешность может достигать 18 %.

4. Разработаны усовершенствованные вероятностные метод и модель для предварительной оценки параметров межмодульного взаимодействия в ССОИ, функционирующих на базе протокола *CANopen FD* и интерфейса *CAN FD*, позволяющие оперативно оценить критически важные параметры межмодульного взаимодействия на самых ранних этапах проектирования ССОИ. Использование модели для исследования конкретной конфигурации ССОИ позволило уменьшить

тактовую частоту приёмопередатчика *CAN FD* в 2-4 раза, что значительно снижает энергопотребление и тепловыделение модулей ССОИ. Показано, что результаты моделирования соответствуют экспериментальным данным, полученными при исследовании реальной системы, с погрешностью не более 7,7 %.

5. Разработана имитационная модель для детального исследования межмодульного взаимодействия в ССОИ на базе интерфейса *CAN FD* и протокола *CANopen FD*, позволяющая в полном объеме исследовать все необходимые параметры межмодульного взаимодействия и путём изменения конфигурации модулей выбрать значения этих параметров, обеспечивающие наилучшую эффективность функционирования ССОИ. Показано, что результаты моделирования соответствуют экспериментальным данным, полученными при исследовании реальной системы, с погрешностью не более 1,3 %.

6. Разработано программное обеспечение, реализующее вероятностную и имитационную модели и позволяющее исследовать параметры межмодульного взаимодействия, оценить степень их влияния на качество связи.

7. С использованием созданных методов, моделей и программного обеспечения разработаны функциональные схемы модулей универсального контроллера присоединения, рассчитанного на работу без перегрузки при установке от 1 до 10 модулей ввода\вывода в основной корпус с возможностью расширения до 127 модулей за счёт использования дополнительных блочных каркасов.

Разработанные методы и модели использованы в ООО «СКБ Электронного приборостроения» (г. Черноголовка) при проектировании универсального модульного контроллера присоединения, находящегося в стадии подготовки серийного производства. Во время выполнения работы зарегистрированы результаты интеллектуальной деятельности (два свидетельства о регистрации программ для ЭВМ). Разработанные вероятностная и имитационная модели внедрены в учебный процесс ЮРГПУ (НПИ) и используются в курсах «Микроконтроллеры и микропроцессоры», «Отладочные средства

микропроцессорных систем», «Проектирование цифровых вычислительных (управляющих) устройств».

Таким образом, при выполнении диссертационного исследования решены все поставленные задачи, достигнута цель работы. Разработаны методы и модели, направленные на решение актуальной научно-технической задачи, связанной с заблаговременной оценкой эффективности межмодульного взаимодействия в ССОИ на самых ранних этапах её проектирования, где под эффективностью понимается:

- гарантированная доставка высокоприоритетных данных за требуемое время, регламентированное документами той области, в которой применяется ССОИ;
- достижение приемлемой скорости передачи низкоприоритетных данных, определяемой условиями эксплуатации конкретной системы.

Результаты диссертационного исследования могут быть использованы организациями, занимающимися проектированием и эксплуатацией модульных систем на базе интерфейса *CAN FD* и протокола *CANopen FD* для снижения вероятности возникновения ошибок настройки межмодульного взаимодействия с сопутствующим уменьшением времени конфигурирования модульной системы.

Дальнейшие исследования в этой области могут быть направлены на поиск путей оптимизации протокола взаимодействия модулей в ССОИ с целью снижения нагрузки сети передачи данных путём обоснованного объединения отдельных передаваемых параметров в сообщения с учётом вероятности изменения этих параметров, требований к времени реакции на изменения, а также времени передачи сообщений и количества сопутствующей служебной информации.

Список сокращений

- АВР — автоматический ввод резерва;
- АПВ — автоматическое повторное включение;
- АРМ — автоматизированное рабочее место;
- АСУ ТП — автоматизированная система управления технологическим процессом;
- БМ — блок мониторинга;
- ВН — высшее напряжение;
- ВУ — верхний уровень;
- ДГР — дугогасящий реактор;
- ДМЗ — демилитаризованная зона;
- ЕНЭС — Единая национальная электрическая сеть;
- ИД — интеллектуальный датчик;
- ИМЦИТ — интерфейсный модуль цифрового измерительного трансформатора;
- ИЭУ — интеллектуальное электронное устройство;
- КА — коммутационный аппарат;
- КЗ — короткое замыкание;
- КНПС — контур нулевой последовательности сети
- КП — контроллер присоединения;
- КПД — коэффициент полезного действия;
- КРУ — комплектное распределительное устройство;
- КРУН — комплектное распределительное устройство наружного исполнения;
- МАВв — модуль аналогового ввода;
- МДВв — модуль дискретного ввода;
- МДВыв — модуль дискретного вывода;
- МПМ — микропроцессорный модуль;
- МЭК — международная энергетическая комиссия;
- НН — низшее напряжение;
- ННТ — наиболее нагретая точка;
- НТП — нормы технологического проектирования;

ОПН — ограничитель перенапряжения;
ПА — противоаварийная автоматика;
ПАС — преобразователь аналоговых сигналов;
ПДС — преобразователь дискретных сигналов;
ПС — электрическая подстанция;
ПТК — программно-технический комплекс;
РЗА — релейная защита и автоматика;
РТСН — резервный трансформатор собственных нужд;
РУ — распределительное устройство;
СВ — секционный выключатель;
СВМ — система вибромониторинга;
СКВ — система контроля вибрации;
СМ — система мониторинга;
СН — среднее напряжение;
СПД — сеть передачи данных;
ССОИ — система сбора и обработки информации;
СУ — средний уровень;
ТН — трансформатор напряжения;
ТСН — трансформатор собственных нужд;
ТТ — трансформатор тока;
ШИП — шкаф измерительных преобразователей;
ЩСН — щит собственных нужд;
ЭЭС — электроэнергетическая система.

Список используемой литературы

1. Глазунов, Ю.В. Оптимизация потоков данных в цифровых энергетических системах / Ю.В. Глазунов, В.А. Скурихин – Известия РАН. Теория и системы управления. – 2020. – № 5. – С. 33–44.
2. Массель, Л.В. Семантическое моделирование при построении цифровых двойников энергетических объектов и систем / Л.В. Массель, А.Г. Массель – Онтология проектирования. 2023. Т13, №1 (47). С. 44-54. DOI:10.18287/2223-9537-2023-13-1-44-54.
3. Ротанов, Е.Г. Оптимизация сбора и обработки информации для повышения эффективности автоматизированных систем / Е.Г. Ротанов, А.В. Шаховской, С.Н. Родионова – Инновации и инвестиции. 2023. №6. С. 32-36.
4. Аллакулиев, Ю.Б. Анализ проблемных вопросов оценки эффективности системы сбора, передачи и отображения информации берегового центра управления автономными необитаемыми подводными аппаратами и обоснование концептуальных направлений их решения // Системы управления, связи и безопасности. 2019. №2. С. 143-161. DOI: 10.24411/2410-9916-2019-10207.
5. Боровков, А.И. Цифровые двойники в высокотехнологичной промышленности. Монография / А.И. Боровков, Ю.А. Рябов, Л.А. Щербина, Е.Р. Мартынец, В.В. Корчевская [и др.] – СПб.: Polytech Press, 2022. – 494 с. ISBN 978-5-7422-8006-4.
6. Wehrle K., Günes M., Gross J. Modeling and Tools for Network Simulation // Springer, Berlin, Heidelberg – 2010. – 545p. doi: 10.1007/978-3-642-12331-3.
7. Шеннон Р. Имитационное моделирование систем - искусство и наука / – М.: Мир, 1978. – 420 с.
8. Alizai M.H., Gao L., Kempf T., Landsiedel O. Tools and Modeling Approaches for Simulating Hardware and Systems [Электронный ресурс] / Springer, Berlin, Heidelberg, 2010. – 21 p. – URL: https://doi.org/10.1007/978-3-642-12331-3_7 (дата обращения: 27.11.2024).

9. Kunz G. Parallel Discrete Event Simulation [Электронный ресурс] / Springer, Berlin, Heidelberg, 2010. – 11 p. – URL: https://doi.org/10.1007/978-3-642-12331-3_8 (дата обращения: 28.11.2024).
10. Baynast A., Bohge M., Willkomm D., Gross J. Physical Layer Modeling [Электронный ресурс] / Springer, Berlin, Heidelberg, 2010. – 38 p. – URL: https://doi.org/10.1007/978-3-642-12331-3_9 (дата обращения: 28.11.2024).
11. Mühleisen M., Bültmann D., Klagges K., Schinnenburg M. Link Layer Modeling [Электронный ресурс] / Springer, Berlin, Heidelberg, 2010. – 18 p. – URL: https://doi.org/10.1007/978-3-642-12331-3_10 (дата обращения: 02.12.2024).
12. Breslau L., Estrin D., Fall K Et al. Advances in Network Simulation. Computer, – 2000. – № 33(5). pp. 59–67.
13. Riley G.F., Henderson T.R. The ns-3 Network Simulator [Электронный ресурс] / Springer, Berlin, Heidelberg, 2010. – 21 p. – URL: https://doi.org/10.1007/978-3-642-12331-3_2 (дата обращения: 10.12.2024).
14. Varga A. OMNeT++ [Электронный ресурс] / Springer, Berlin, Heidelberg, 2010. – 27 p. – URL: https://doi.org/10.1007/978-3-642-12331-3_3 (дата обращения: 13.12.2024).
15. Sommer J., Scharf J. IKR Simulation Library. [Электронный ресурс] / Springer, Berlin, Heidelberg, 2010. – 8 p. – URL: https://doi.org/10.1007/978-3-642-12331-3_4 (дата обращения: 13.12.2024).
16. Bültmann D., Mühleisen M., Max S. Open WNS. [Электронный ресурс] / Springer, Berlin, Heidelberg, 2010. – 13 p. – URL: https://doi.org/10.1007/978-3-642-12331-3_5 (дата обращения: 13.12.2024).
17. IEEE 802.3-2015 Standard for Ethernet. LAN/MAN Standards Committee. IEEE Computer Society [Электронный ресурс] / © 2015, IEEE Computer Society. - URL: https://www.onetel.de/wp-content/uploads/2016/11/802.3-2015_SECTION1.pdf (дата обращения: 02.01.2025).
18. GSM Specification / 3GPP, 1982. – URL: <https://www.3gpp.org/specifications-technologies/specifications-by-series/%20gsm-specifications> (дата обращения: 01.01.2025).

19. Universal Mobile Telecommunications System Specification / 3GPP, 03.04.2024. – 14 p. URL: [https:// https:// www.3gpp.org/ ftp/ Specs/ archive/ 23_series/ %2023.101/](https://www.3gpp.org/ftp/Specs/archive/23_series/%2023.101/) (дата обращения: 01.01.2025).

20. 802.11bc-2023 / IEEE Standard for Information Technology, 28 February 2024. – URL: [https:// ieeexplore.ieee.org/ servlet/opac?punumber=10456573](https://ieeexplore.ieee.org/servlet/opac?punumber=10456573) (дата обращения: 01.01.2025).

21. Bluetooth Specification and Documents / Bluetooth SIG Inc. – URL: [https:// www.bluetooth.com/ specifications/ specs/ %20?type=specs-docs](https://www.bluetooth.com/specifications/specs/%20?type=specs-docs) (дата обращения: 01.01.2025).

22. CAN Specification. Version 2.0 / Robert Bosch GmbH, 1991. – 73 p. URL: [http:// www.bosch-semiconductors.de / media / ubk_semiconductors / pdf_1 / canliteratur / can2spec.pdf](http://www.bosch-semiconductors.de/media/ubk_semiconductors/pdf_1/canliteratur/can2spec.pdf) (дата обращения: 02.01.2025).

23. CAN with Flexible Data-Rate; Version 1.1. Aug.2011 / Robert Bosch GmbH. – URL: [https:// www.semiconductors.bosch.de/ media/ pdf/ .../ canliteratur/ can_fd.pdf](https://www.semiconductors.bosch.de/media/pdf/.../canliteratur/can_fd.pdf) (дата обращения: 02.01.2025).

24. Interface Circuits for TIA/EIA-485 (RS-485). Application Report [Электронный ресурс] / © 2008, Texas Instruments Incorporated. – URL: [https:// www.ti.com/ lit/ an/ slla036d/ slla036d.pdf? ts=1636957884480](https://www.ti.com/lit/an/slla036d/slla036d.pdf?ts=1636957884480) (дата обращения: 02.01.2025).

25. SPI Block Guide V03.06, FreeScale Semiconductor. [Электронный ресурс] / 2008, FreeScale Semiconductor. – URL: [https:// www.freescale.com/ files/ microcontrollers/doc/ ref_manual/S12SPIV3.pdf](https://www.freescale.com/files/microcontrollers/doc/ref_manual/S12SPIV3.pdf) (дата обращения: 02.01.2025).

26. ГОСТ 24291-90 Электрическая часть электростанции и электрические сети. Термины и определения. - введ. 27.12.2005. – М.: Стандартинформ. – 2005. – 14 с.

27. ГОСТ 19431-2023 Энергетика и электрификация. Термины и определения. - введ. 01.12.2023. – М.: Стандартинформ. – 2023. – 8 с.

28. Бохан, А. Н. Проектирование подстанций систем электроснабжения: учеб. пособие / А. Н. Бохан; М-во образования Респ. Беларусь; Гомел. гос. техн. ун-т им. П. О. Сухого. – Гомель: ГГТУ им. П. О. Сухого, 2017. – 311 с.

29. Кокин, С.Е. Схемы электрических соединений подстанций / С.Е. Кокин, С.А. Дмитриев, А.И. Хальясмаа. – Екатеринбург: Изд-во Уральского университета, 2015. – 102 с.

30. СТО 56947007-29.240.10.248-2017 Нормы технологического проектирования подстанций переменного тока с высшим напряжением 35-750 кВ (НТП ПС). – введ. 25.08.2017. – Москва. – 135 с.

31. ГОСТ 52719-2007 Трансформаторы силовые. Общие технические требования. - введ. 09.05.2007. – М.: Стандартинформ. – 2007. – 45 с.

32. СТО 56947007-25.040.40.227-2016 Типовые технические требования к функциональной структуре автоматизированных систем управления технологическими процессами подстанций Единой национальной электрической сети (АСУ ТП ПС ЕНЭС). – введ. 26.09.2016. – Москва. – 97 с.

33. СТО 56947007-25.040.40.226-2016 Общие технические требования к АСУ ТП ПС ЕНЭС. Основные требования к программно-техническим средствам и комплексам. – введ. 26.09.2016. – изм. 06.10.2022. – М.: ПАО «ФСК ЕЭС». – 2016. – 71 с.

34. СТО 56947007-25.040.30.309-2020 Корпоративный профиль МЭК 61850. – введ. 05.10.2020. – изм. 24.03.2023. – М.: ПАО «ФСК ЕЭС». – 2020. – 257 с.

35. МЭК 61850-7-1-2009 Сети и системы связи на подстанциях. Часть 7. Базовая структура связи для подстанций и линейного оборудования. Раздел 1. Принципы и модели. – введ. 15.12.2009. - М.: Стандартинформ. – 2011. – 118 с.

36. МЭК 61850-7-4-2011 Сети и системы связи на подстанциях. Часть 7. Базовая структура связи для подстанций и линейного оборудования. Раздел 4. Совместимые классы логических узлов и классы данных. – введ. 13.12.2011. - М.: Стандартинформ. – 2013. – 122 с.

37. Техническое описание. Профиль SV МЭК 61850-9-2 Vinom3. – СПб. – 2017. – 35 с.

38. Федеральный закон «О безопасности критической информационной инфраструктуры Российской Федерации» от 26.07.2017 № 187-ФЗ (последняя редакция). - введ. 26.07.2017. – Москва.

39. СТО 34.01-6.2-002-2024 Контроллеры присоединения, преобразователи дискретных сигналов. Типовые технические требования. – введ. 24.05.2024. – М.: ПАО «Россети». – 2024. – 77 с.

40. ГОСТ Р МЭК 60917-1-2011 Модульный принцип построения базовых несущих конструкций для электронного оборудования. Часть 1. - введ. 14.12.2011. – М.: Стандартинформ. – 2013. – 23 с.

41. СТО 56947007-29.240.30.010-2008 Схемы принципиальные электрические распределительных устройств подстанций 35-750 кВ. Типовые решения. – введ. 20.12.2007. – М.: ОАО «ФСК ЕЭС». – 2007. – 132 с.

42. СТО 56947007-29.130.01.092-2011 Выбор видов и объемов телеинформации при проектировании систем сбора и передачи информации подстанций ЕНЭС для целей диспетчерского и технологического управления. – введ. 03.05.2011. – М.: ОАО «ФСК ЕЭС». – 2011. – 43 с.

43. Скворцова, Н.Н. Программно-аппаратные средства сбора, хранения и обработки информации. Учеб. пособие / Н.Н. Скворцова, А.К. Козак, Н.Н. Богачев [и др.] – М.: Инфра-Инженерия, 2024. – 124 с.

44. RFC 793 Transmission Control Protocol. DARPA Internet Program. Protocol Specification [Электронный ресурс] / September 1981, Information Processing Techniques Office. - URL: [https:// datatracker.ietf.org/ doc/ html/ %20rfc793](https://datatracker.ietf.org/doc/html/rfc793) (дата обращения: 02.01.2025).

45. RFC 791 Internet Protocol. DARPA Internet Program. Protocol Specification [Электронный ресурс] / September 1981, Information Processing Techniques Office. - URL: [https:// datatracker.ietf.org/ doc/ html/ %20rfc791](https://datatracker.ietf.org/doc/html/rfc791) (дата обращения: 02.01.2025).

46. ГОСТ Р МЭК 60870-5-103-2005 Устройства и системы телемеханики. Часть 5. Протоколы передачи. Раздел 103. Обобщающий стандарт по информационному интерфейсу для аппаратуры релейной защиты. - введ. 28.12.2005. – М.: Стандартинформ. – 2006. – 86 с.

47. Modbus application protocol specification V1.1b3, Schneider Automation Inc. [Электронный ресурс] / April 26, 2012, Schneider Automation Inc. – URL: <https://>

modbus.org/ docs/ Modbus_Application_Protocol_V1_1b3.pdf (дата обращения: 26.11.2024).

48. CANopen application layer and communication profile. Version: 4.2.0 21 February 2011 [Электронный ресурс] / CAN in Automation e. V. – 2011. – 158 p. – URL: <http://www.can-cia.org> (дата обращения: 26.11.2024).

49. OMG Unified Modeling Language (OMG UML). Version 2.5.1. [Электронный ресурс] / – 2017. – 796 p. – URL: <https://www.omg.org/spec/UML/2.5.1/PDF> (дата обращения: 25.01.2025).

50. Бабина О.И. Сравнительный анализ имитационных и аналитических моделей [Электронный ресурс] / О.И. Бабина // ИММОД-2009. Имитационное моделирование. Теория и практика: материалы 4-й всеросс. науч.-практ. конф. – СПб., 21-23 октября 2009 г. – URL: <http://gpss.ru/immod09/doklad/09.pdf> (дата обращения: 27.11.2024).

51. ГОСТ Р МЭК 60297-3-101-2006 Конструкции несущие базовые радиоэлектронных средств. Блочные каркасы и связанные с ними вставные блоки. Размеры конструкции серии 482,6 (19 дюймов). - введ. 27.12.2006. – М.: Стандартиформ. – 2007. – 24 с.

52. МЭК 62439-3-2016 Промышленные сети связи. Сети с высокой готовностью к автоматической обработке. Часть 3. Протокол параллельного резервирования (PRP) и бесшовное резервирование среды высокой готовности (HSR). – введ. 29.03.2019. - М.: Стандартиформ. – 2019. – 96 с.

53. UM10204. I2C-bus specification and user manual. Rev. 5. [Электронный ресурс] / 2012, NXP Semiconductor. – URL: https://web.archive.org/web/20130511150526/http://www.nxp.com/documents/user_manual/UM10204.pdf (дата обращения: 02.01.2025).

54. Varnavas, K. Serial Back-Plane Technologies in Advanced Avionics Architectures / K. Varnavas // 24th Digital Avionics Systems Conference. – 2005. - № 1. – с. 2-10. doi: 10.1109/dasc.2005.1563416.

55. ГОСТ Р ИСО/МЭК 7498-1-99 Информационная технология. Взаимосвязь открытых систем. Базовая эталонная модель. Часть 1. Базовая модель. – введ. 1.01.2000. – М.: Изд-во стандартов, 1999. – 58 с.

56. CiA 1311-1 CANopen FD electronic device description – Part 1: XML schema definition for CiA 1301. Version: 1.0.1 August 2022 [Электронный ресурс] / CAN in Automation e. V. – 2022. – 131 p. – URL: <http://www.can-cia.org> (дата обращения: 05.01.2025).

57. The Basics of CANopen. © 2021 National Instruments Corp. – URL: <http://www.ni.com/white-paper/14162/en/#toc2> (дата обращения: 05.01.2025).

58. CANopen – The standardized embedded network. – URL: <https://www.can-cia.org/canopen/> (дата обращения: 05.01.2025).

59. CiA 306. CANopen electronic data sheet (EDS). Version 1.3.0. January 01, 2005 [Электронный ресурс] / © CAN in Automation (CiA) 2005. – URL: <https://www.can-cia.org> (дата обращения: 07.01.2025).

60. Муженко, А.С. Метод описания модульной информационно-измерительной системы на основе протокола CANopen с учётом межмодульных информационных связей / А.С. Муженко, Д.А. Плотников, В.И. Лачин // Глобальная ядерная безопасность. – 2023. – № 48 (3). – С. 26-36.

61. Extensible Markup Language (XML) [Электронный ресурс] / – 2016. – URL: <https://www.w3.org/XML/> (дата обращения: 10.01.2025).

62. JSON and XML Editor [Электронный ресурс] / – 2021. – URL: <https://www.altova.com/xmlspy-xml-editor> (дата обращения: 10.01.2025).

63. Плотников Д.А., Муженко А.С. Проверка XML-описания модульных систем на основе протокола *CANopen FD* / Д.А. Плотников, А.С. Муженко // Свидетельство о регистрации программы для ЭВМ. – 24.12.2024 – № RU 2024691739.

64. W3C XML Schema Definition Language (XSD) 1.1 Part 1 [Электронный ресурс] / – 2012. – URL: <https://www.w3.org/TR/xmlschema11-1/> (дата обращения: 12.01.2025).

65. Плотников, Д.А. Оценка времени отклика элементов в модульных информационно-измерительных и управляющих системах, использующих интерфейс CAN / Д.А. Плотников // Изв. вузов. Сев.-Кавк. регион. Техн. науки. – 2017. – № 1. – С. 13-18. doi: 10.17213/0321-2653-2017-1-13-18

66. Никифоров, В.В. Оценка времени доставки сообщений в распределенных системах реального времени с CAN-интерфейсом / В.В. Никифоров, А.И. Шкиртиль // Известия высших учебных заведений. Приборостроение. – 2010. – № 7. – С. 33-39.

67. Tindell, Hansson, Wellings. Analysing real-time communications: controller area network (CAN). 1994 Proceedings Real-Time Systems Symposium, San Juan, PR, USA, 1994, pp. 259-263, doi: 10.1109/REAL.1994.342710.

68. R.De Andrade, K.N.Hodel, J.F.Justo, A.M.Laganá, M.M.Santos and Z.Gu. Analytical and Experimental Performance Evaluations of CAN-FD Bus // IEEE Access, vol. 6, pp. 21287-21295, 2018, doi: 10.1109/ACCESS.2018.2826522.

69. ISO 11898-1:2015 Road vehicles – Controller area network (CAN) – Part 1: Data link layer and physical signalling. URL: <https://www.iso.org/ru/standard/63648.html> (дата обращения: 15.01.2025).

70. Arthur Mutter. CAN FD and the CRC issue / – 2015. – URL: <https://can-newsletter.org/uploads/media/raw/604de101b0ecaed387518831d32b044e.pdf> (дата обращения: 15.01.2025).

71. Муженко, А.С. Оценка максимального и минимального времени передачи сообщений в информационно-измерительных и управляющих системах, использующих интерфейс CAN FD / А.С. Муженко, В.И. Лачин, Д.А. Плотников // Безопасность ядерной энергетики: тез. докл. 20-й Междунар. науч.-практ. конф., 19-20 сентября 2024 г. / НИЯУ МИФИ [и др.]. – Волгодонск: ВИТИ НИЯУ МИФИ, 2024. – С. 183-186.

72. N. Sußmann, A. Meroth. Model based development and verification of CANopen components 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2017, pp. 1-5, doi: 10.1109/ETFA.2017.8247745.

73. Плотников, Д.А., Лачин В.И., Алджиязна В.К.М., Соломенцев К.Ю. Разработка модели межмодульного взаимодействия в системах контроля вибрации на базе протокола CANopen // Известия вузов. Электромеханика. – 2020. – Т. 63, No 2-3. – С. 68-75. – DOI: 10.17213/0136-3360-2020-2-3-68-75.

74. Плотников, Д.А. Вероятностный метод и модель для оценки параметров межмодульного взаимодействия в системе управления защитой турбоагрегата по вибрации // Глобальная ядерная безопасность. – 2021. – № 3(40). – С. 37-51. doi: 10.26583/gns-2021-03-04.

75. Муженко, А.С. Усовершенствованные вероятностные метод и модель для оценки параметров межмодульного взаимодействия в системах сбора и обработки информации. Часть 1 // Известия высших учебных заведений. Северо-Кавказский регион. Технические науки. – 2024. – № 4. – С. 55–60.

76. Лачин В.И., Плотников Д.А., Муженко, А.С., Рарова Н.В. Оценка времени передачи сообщений в информационно-измерительных и управляющих системах, использующих интерфейс CAN FD // Известия высших учебных заведений. Северо-Кавказский регион. Технические науки. – 2023. – № 2. – С. 5-10.

77. Alaei Reza, Moallem Payman, Bohlooli Ali. Statistical based algorithm for reducing bit stuffing in the Controller Area Networks. Microelectronics Journal. 101. 104794. – 2020. doi: 10.1016/j.mejo.2020.104794.

78. Nolte Thomas, Hansson Hans, Norström Christer, Punnekkat Sasikumar. Using bit-stuffing distributions in CAN analysis. IEEE Real-Time Embedded Systems Workshop. – 2009. URL: [https:// www.researchgate.net/ publication/ 255672829_Using_bitstuffing_distributions_in_CAN_analysis](https://www.researchgate.net/publication/255672829_Using_bitstuffing_distributions_in_CAN_analysis) (дата обращения: 17.01.2025).

79. HMS. Basics of the CANopen and CANopen FD protocol. Mechanism, Functions, Parameters and Implementation. URL: [https:// media.hms-networks.com/ image/ upload/ v1701953901/Documents/Whitepapers/Ixxat_CANopen-Protocol-Whitepaper_EN.pdf](https://media.hms-networks.com/image/upload/v1701953901/Documents/Whitepapers/Ixxat_CANopen-Protocol-Whitepaper_EN.pdf) (дата обращения: 18.01.2025).

80. Плотников Д.А., Муженко А.С. Вероятностная модель модульной системы на базе протокола CANopen FD / Д.А. Плотников, А.С. Муженко // Свидетельство о регистрации программы для ЭВМ. – 19.12.2024 – № RU 2024691024.

81. Муженко, А.С. Разработка имитационной модели модульной информационно-измерительной системы (ИИУС) на базе интерфейса CAN FD / А.С. Муженко, В.И. Лачин, Д.А. Плотников, В.Б. Дьяченко, Н.В. Рарова // Безопасность ядерной энергетики: тез. докл. 19-й Междунар. науч.-практ. конф., 06-07 июня 2023 г. / НИЯУ МИФИ [и др.]. – Волгодонск: ВИТИ НИЯУ МИФИ, 2023. – С. 130-133.

82. Горяинов, В.Б. Математическая статистика: Учеб. Для вузов / В.Б. Горяинов, И.В. Павлов, Г.М. Цветков и др. Под ред. В.С. Зарубина, А.П. Крищенко. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2001. – 424 с.

83. Хамханов, К.М. Основы планирования эксперимента : Методическое пособие – Улан-Удэ, 2001. – 50 с.

84. Арнольд, В.И. Математическое понимание природы. М.: МЦНМО, 2011.

85. Боровков, А.А. Математическая статистика. М.: ФИЗМАТЛИТ, 2007.

86. Иванищев, Ю.Г. Варианты оценки адекватности нормализованной модели / Ю.Г. Иванищев, В.М. Давыдов // Информационные технологии XXI века. Сборник научных трудов. – 2021. – с. 328–335.

87. Шестерикова, Я.В. Анализ математической модели оценки комплексного показателя качества // Наука и бизнес: пути развития. – 2019. – с. 91–94.

88. Муженко, А.С. Результаты оценки параметров межмодульного взаимодействия в контроллерах присоединения электроэнергетических подстанций, работающих на базе интерфейса CAN FD и протокола CANopen FD с использованием разработанной вероятностной модели / А.С. Муженко, Д.А. Плотников / Кибернетика энергетических систем: тез. докл. 46-й международной научно-технической конференции / ЮРГПУ (НПИ). – Новочеркасск: ЮРГПУ (НПИ), 2024. С. 100-110.

89. Плотников, Д.А. Метод повышения помехоустойчивости устройства управления дугогасящим реактором / Д.А. Плотников, В.И. Лачин, К.Ю.

Соломенцев, А.С. Муженко // Известия высших учебных заведений. Электромеханика. – 2023. - № 66 (4). – с. 112-117. doi: 10.17213/0136-3360-2023-4-112-117.

90. Осипов Э.Р., Обабков В.К. Сравнительный анализ способов заземления нейтрали в задаче подавления дуговых замыканий на землю // Изв. вузов. Горный журнал, 1988. – № 3. – С. 94-97.

91. Осипов П.В., Фастий Г.П., Карпов А.С., Ярошевич В.В. Анализ режимов работы нейтрали сети 35 кВ с целью повышения электробезопасности для населения // Труды Кольского научного центра РАН. 2016. №5-13 (39). URL: <https://cyberleninka.ru/article/n/analiz-rezhimov-raboty-neutrali-seti-35-kv-s-tselyu-povysheniya-elektrobezopasnosti-dlya-naseleniya> (дата обращения: 17.01.2025).

92. Zhao, Xiang & Liu, Feng & Tian, Ye. (2014). Adjusting the Degree of Compensation of Small Current Grounding System Petersen Coil. Applied Mechanics and Materials. 620. 394-397. doi: 10.4028/www.scientific.net/AMM.620.394.

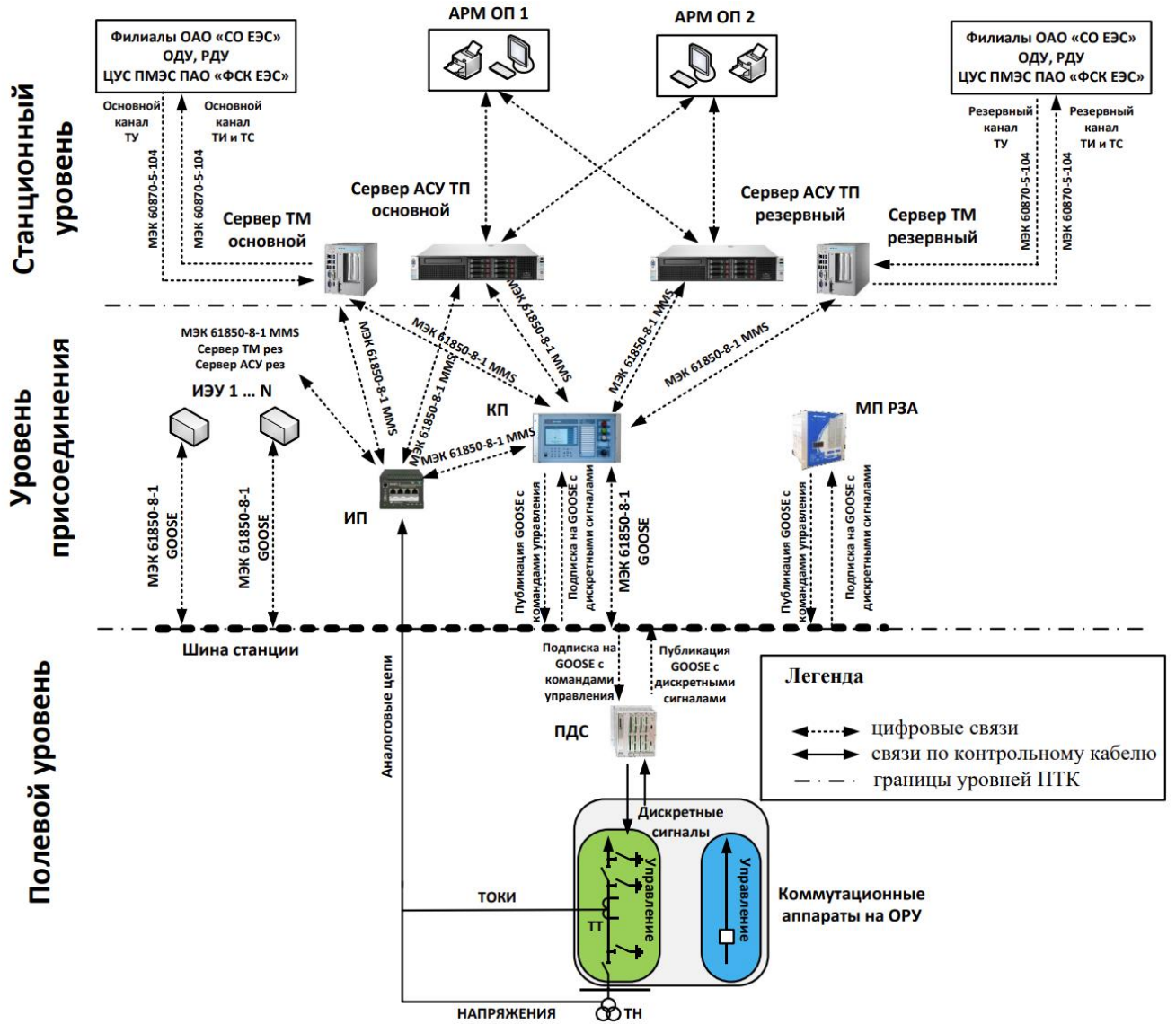
93. Гайдулин, Ф.Л. Перенапряжения в сетях 6-35кВ / Ф.Л. Гайдулин, В.Г. Гольдштейн, А.Л. Дульзон, Ф.Х. Халилов. М.: Энергоатомиздат, 1989. 234 с.

94. Ильин, В.Ф. Определение параметров контура нулевой последовательности сети по его переходной характеристике / В.Ф. Ильин, Е.М. Петров, М.И. Петров И.В. Соловьев // Научные и производственно-технические проблемы энергетики и электротехники: сб. науч. тр. – Чебоксары. – 2005. – С. 74-81.

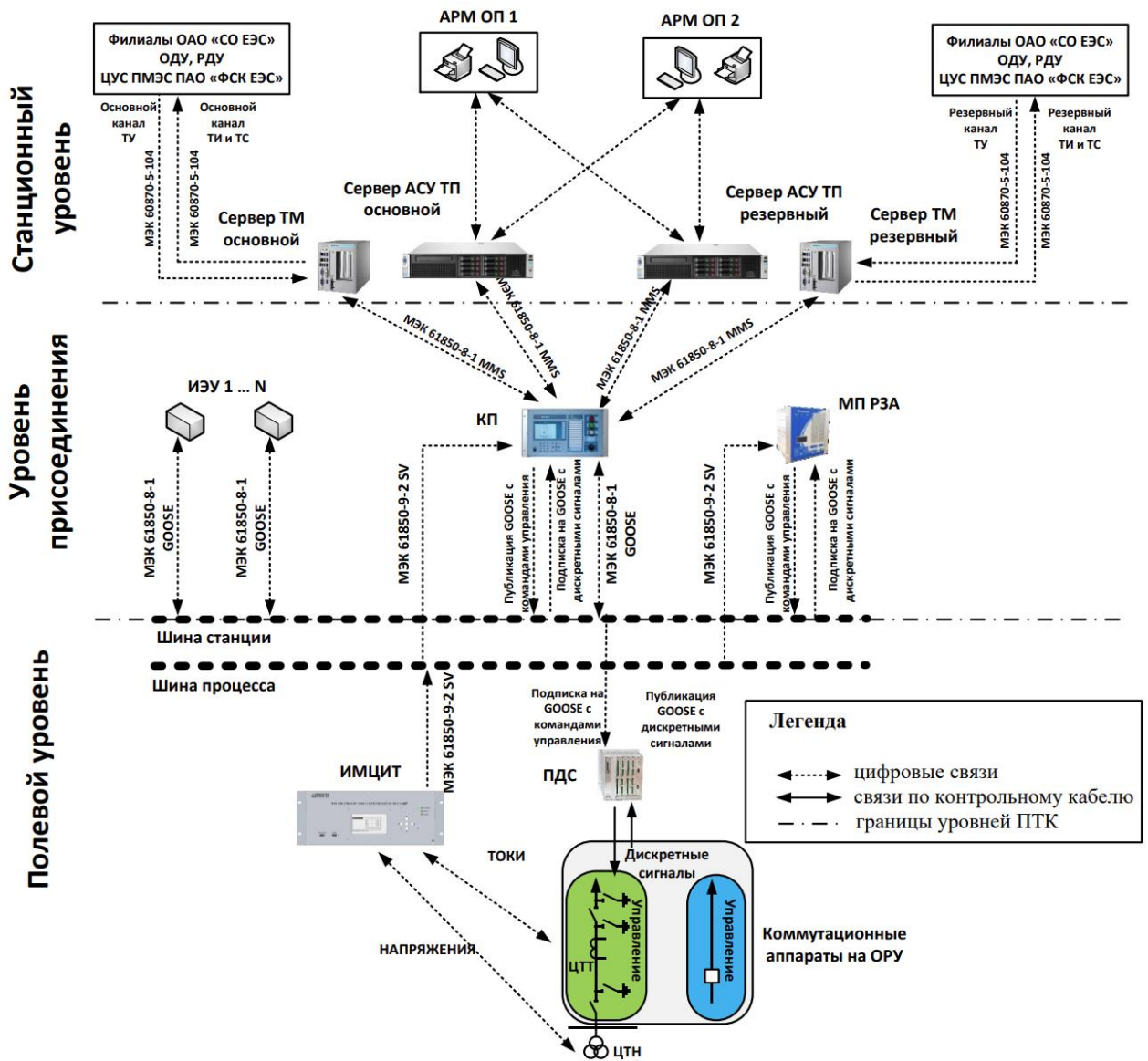
95. Обабков, В.К. Точность автоподстройки частоты свободных колебаний в симметричных сетях с компенсированной нейтралью / В.К. Обабков, А.П. Никифоров // Электричество.–1996. – № 12.– С. 8-16.

Структурные схемы информационных связей,
компоновки элементов ПТК АСУ ТП ПС

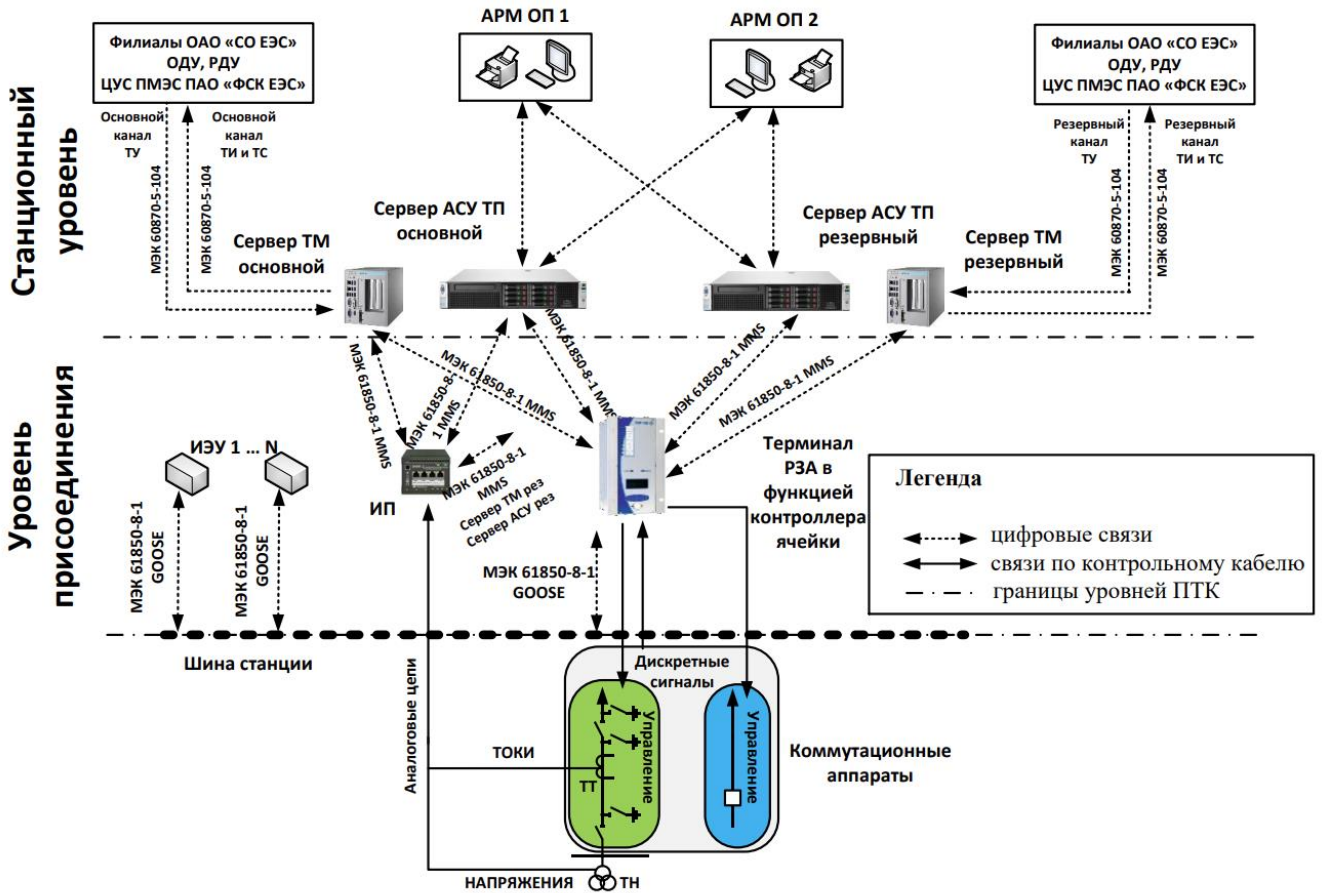
Структурная схема информационных связей ПТК для распределительных устройств напряжением 110 кВ и выше (архитектура II)



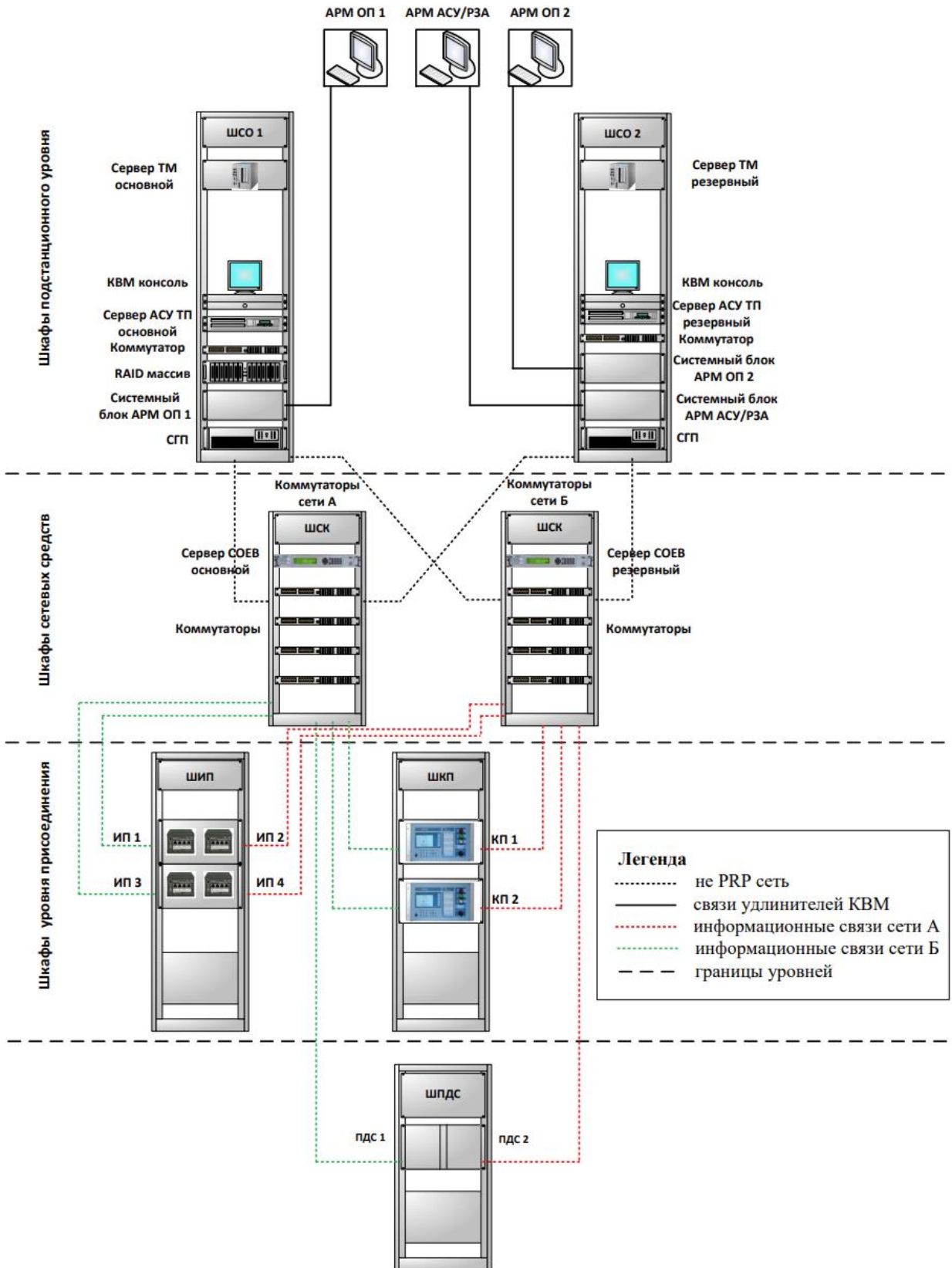
Структурная схема информационных связей ПТК для распределителей напряжением 110 кВ и выше (архитектура III)



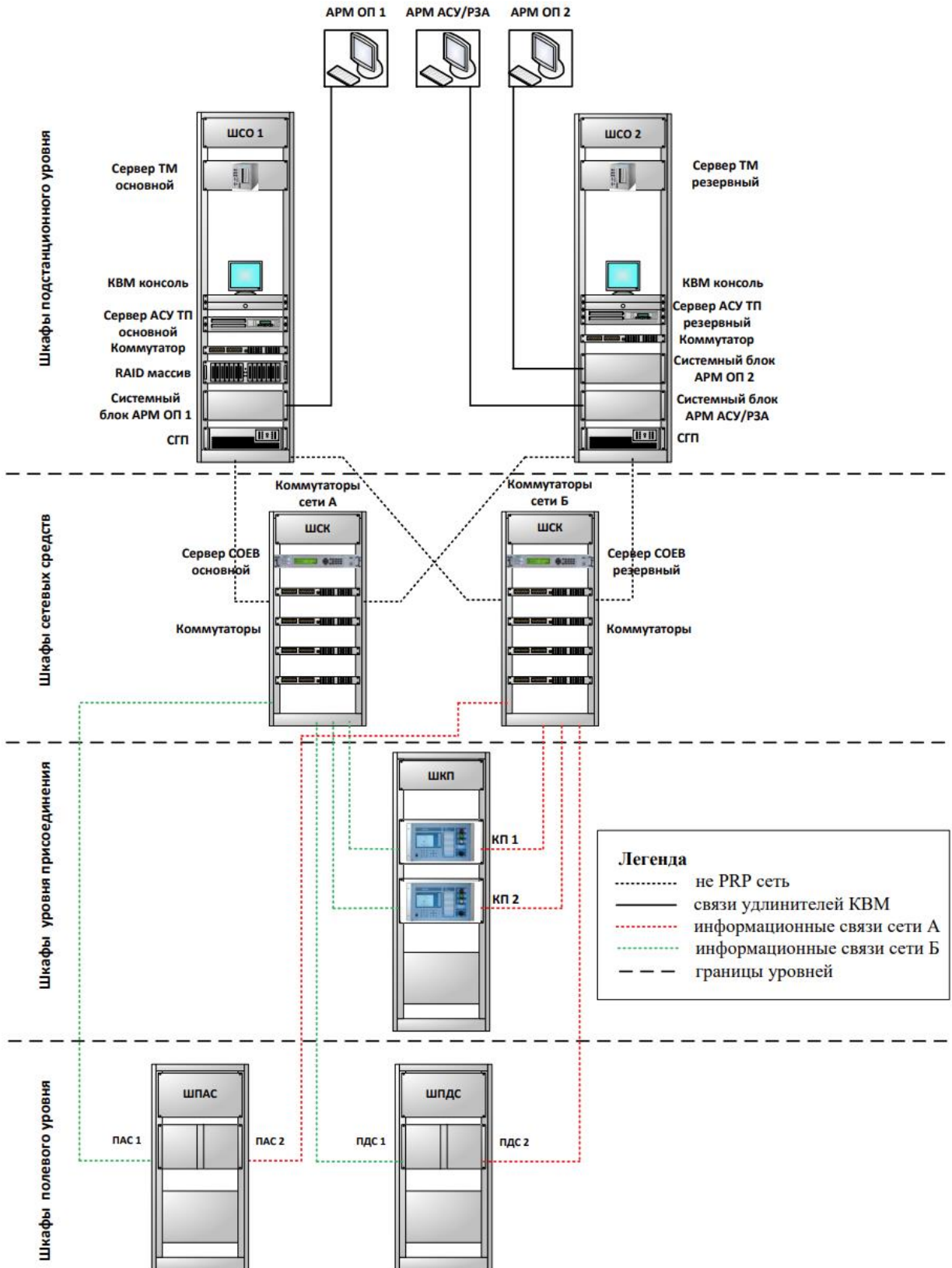
Структурная схема информационных связей ПТК для комплектных распределительных устройств 6 ÷ 35 кВ (архитектура II и III)



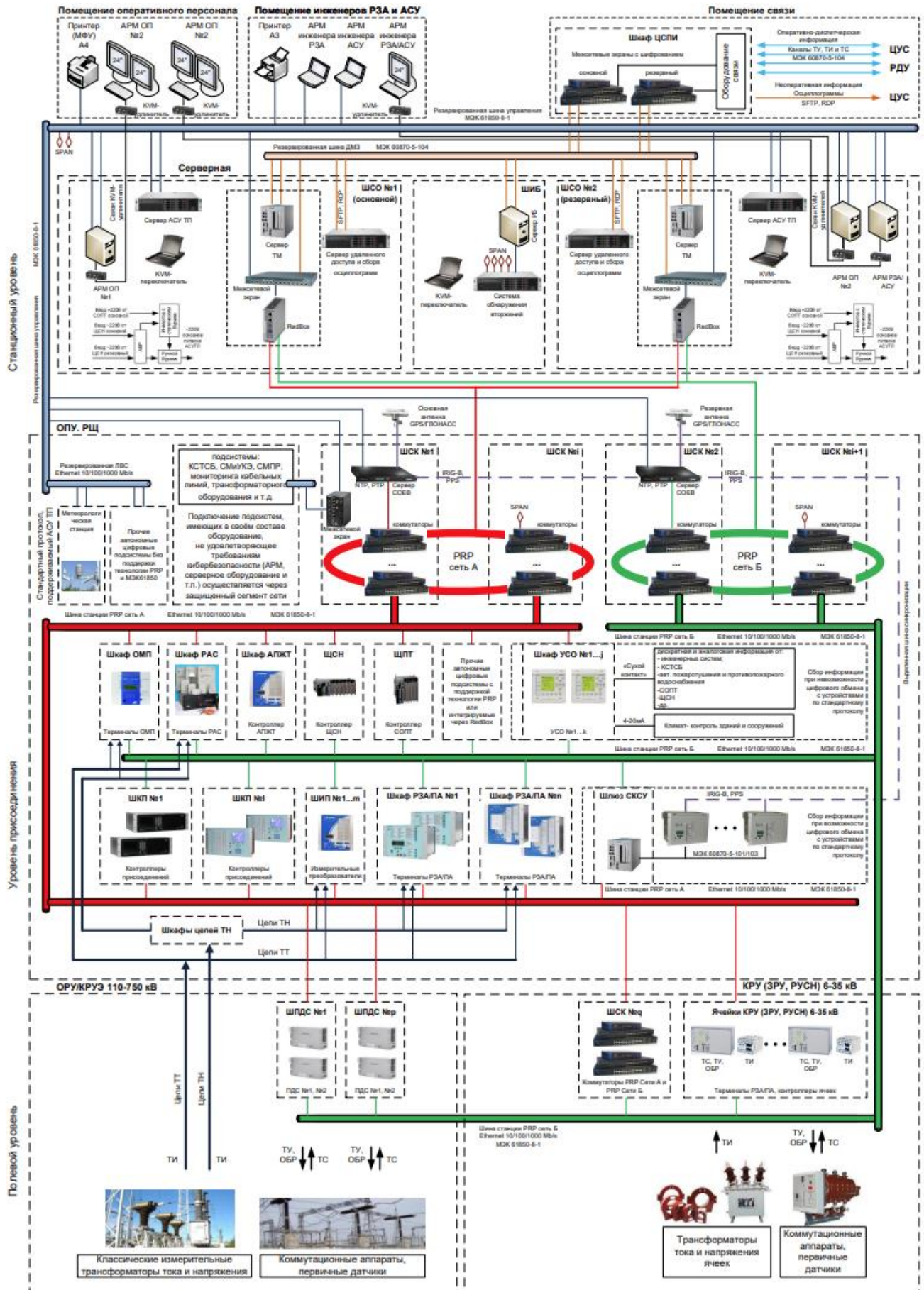
Структурная схема компоновки элементов ПТК (архитектура II)



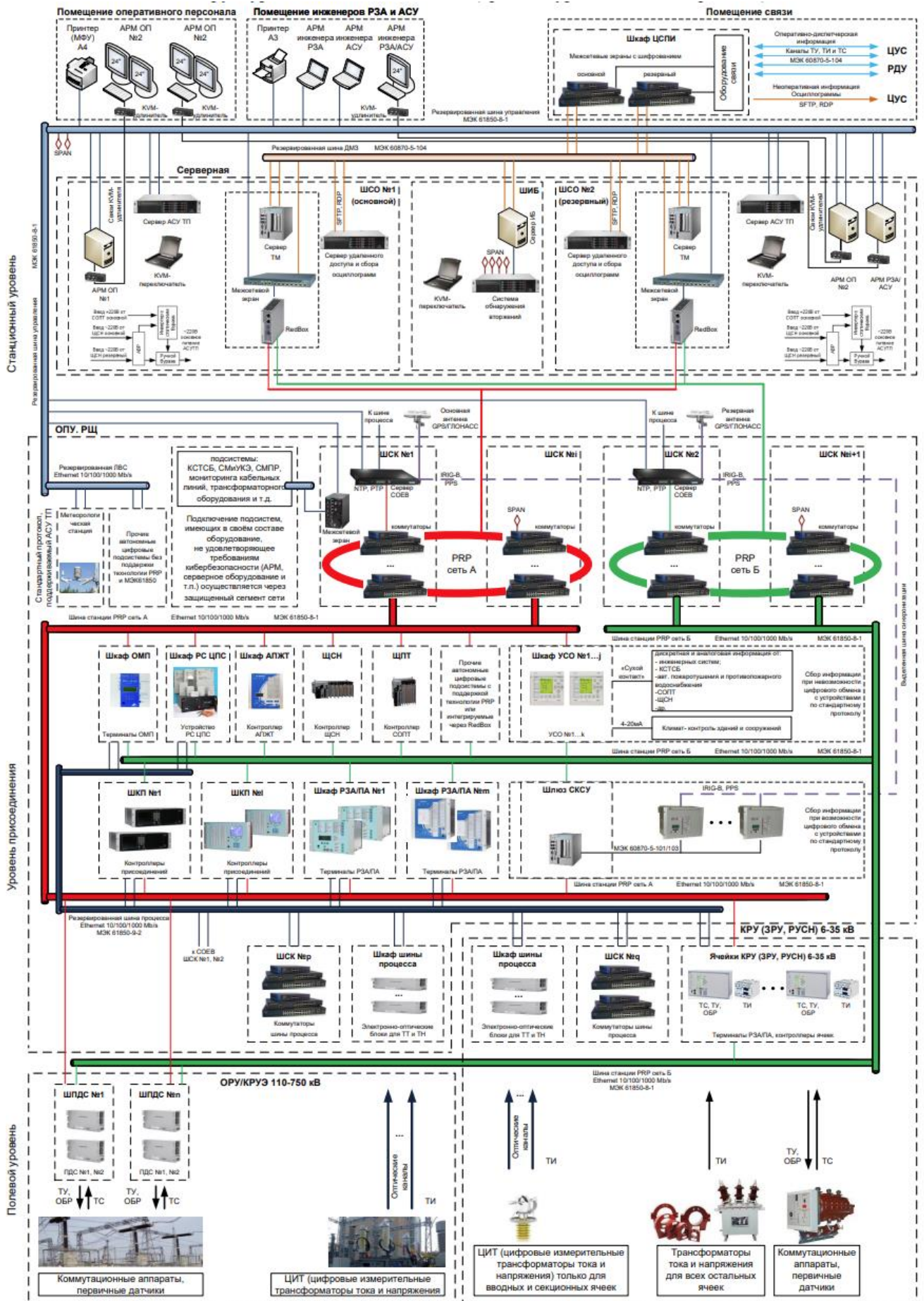
Структурная схема компоновки элементов ПТК (архитектура III)



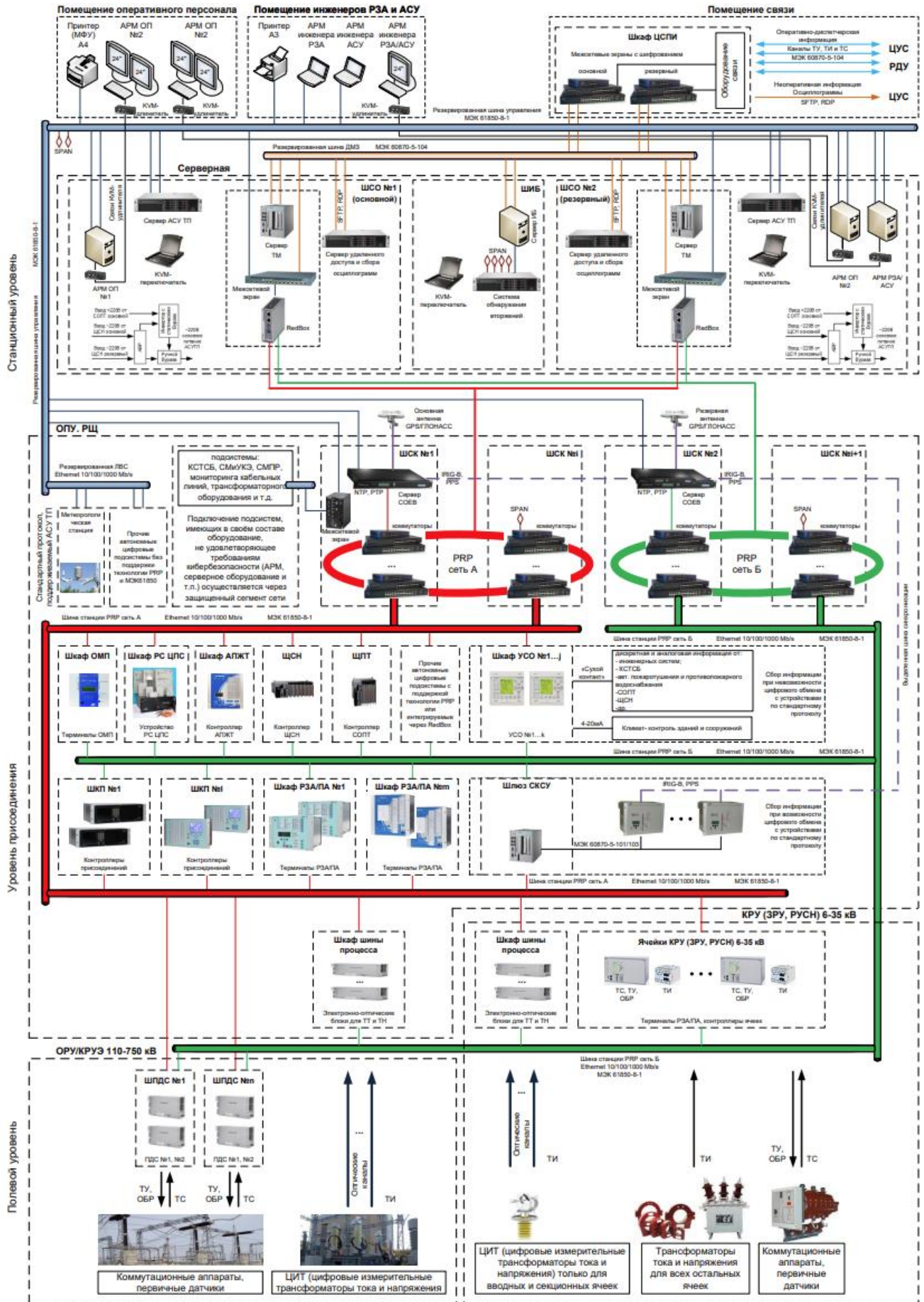
Структурная схема ПТК АСУ ТП (архитектура II)



Структурная схема ПТК АСУ ТП (архитектура III с шиной процесса)



Структурная схема ПТК АСУ ТП (архитектура III с мульти шиной)



Исходные тексты основных модулей
разработанного программного обеспечения

Б.1 Вероятностная модель для исследования параметров межмодульного взаимодействия в системе

MainProg

```
% Основная программа для расчета

BitRate_N = 1000000; % скорость передачи битов полей арбитража на номинальной скорости,
бит/с
BitRate_D = 8000000; % скорость передачи битов полей данных на альтернативной скорости
(скорости данных), бит/с
TSync = 0.1; % длительность цикла синхронизации, с
TypPeriod = 1; % интервал между заявками, с (типовое значение)
kbs = 0.491; % коэффициент добавления дополнительных битов
IDE = 0; % признак формата сообщения (0 - стандартный, 1 - расширенный)
BMs = 127; % BM - Bulk Messages - количество сообщений в блоке
Apr = 0.01; % апертура сигнала для PDO
Nsdo = 4000; % длина осциллограммы (10 периодов)
FlagsMode = 0; % режим вычисления интенсивности PDO с дискретными флагами (0 -
минимальная, !0 - максимальная)
StressMode = 0; % режим максимальной нагрузки (все сообщения в одном цикле
синхронизации)
SimpleMode = 0; % режим упрощенного вычисления вероятности досрочной передачи PDO

% Инициализируем исходные данные
[H, B, M, DLC, V, UseProb] = Init(TSync, TypPeriod, StressMode);

MTypesCnt = length(M); % число типов модулей
ClassesCnt = length(H); % число классов заявок

LambdaS = zeros(MTypesCnt, ClassesCnt);

% Исследование изменений
ITER = 20; % число итераций
x = zeros(1, ITER); % метки оси абсцисс
L_ = zeros(1, ITER); % общая интенсивность потока заявок
b_ = zeros(1, ITER); % среднее время обслуживания заявки
y_ = zeros(1, ITER); % нагрузка системы
q_ = zeros(1, ITER); % скорость передачи SDO

% цикл по варьируемому параметру
for z=1:ITER

    % Время передачи бита
    tN = 1/BitRate_N;
    tD = 1/BitRate_D;

    % Перебираем все типы модулей
    for k=1:MTypesCnt
        % Перебираем все классы заявок для каждого типа модуля
        for h=1:ClassesCnt
            % Длительность сообщения каждого класса для каждого модуля
            if DLC(k, h) >= 0
                Tmsg(k, h) = MsgLength_N(IDE, kbs)*tN ...
                    + MsgLength_D(DLC(k, h), kbs)*tD;
            else
                Tmsg(k, h) = 0;
            end
        end
        % Вычисляем интенсивности потоков заявок каждого модуля
    end
end
```

```

if UseProb(k,h)==0      % только детерминированный поток
    Lambda(k,h) = 1/(TSync * B(k,h));
elseif UseProb(k,h)==1 % имеется случайная составляющая
    if SimpleMode==1    % в упрощённом режиме используем общий параметр а
        Lambda(k,h) = LambdaPDO(SignalAlpha(Apr)*V(k,h),...
            (TSync * B(k,h)), 1/(TSync * B(k,h)), 1/TSync);
    else                % в нормальном режиме используем отдельные параметры а
        Lambda(k,h) = LambdaPDO(SignalAlpha(Apr,k,h),...
            (TSync * B(k,h)), 1/(TSync * B(k,h)), 1/TSync);
    end
elseif UseProb(k,h)==2 % есть флаги
    if FlagsMode == 0
        Lambda(k,h) = 1/(TSync * B(k,h));
    else
        Lambda(k,h) = 1/TSync;
    end
end
LambdaS(k,h) = Lambda(k,h) * M(k);
end
end
% заменяем Inf на 0 (получается после деления на 0)
Lambda(isinf(Lambda))=0;
LambdaS(isinf(LambdaS))=0;
LambdaAll = sum(sum(LambdaS,1));

% Вычисляем среднюю интенсивность обслуживания mu и время обслуживания b
Tmp = LambdaS .* Tmsg;
Tmp = sum(sum(Tmp,2));
b = Tmp / LambdaAll;
mu = 1/b;

% Вычисляем показатели системы
y = LambdaAll / mu;      % нагрузка системы
p = 1-y;                % коэффициент простоя системы

Qh = USdoSpeed(Nsdo, p, kbs, VMs, tN, tD, IDE); % скорость передачи осциллограмм

L_(z) = LambdaAll;      % общая интенсивность потока
b_(z) = b;              % время передачи сообщения
y_(z) = y;              % нагрузка системы
q_(z) = Qh;             % скорость передачи USDO

% Изменяем какой-то параметр модели перед очередной итерацией
x(z) = BitRate_D;
BitRate_D = BitRate_D + 350000;
% M(2) = M(2)+1;
%Apr = Apr + 0.2;

```

end

AvgInterval

```

% Вычисление средней длительности интервала между заявками в заданном
% интервале длительностей и при заданном параметре экспоненциального
% распределения длительностей.

```

```

% Alpha - параметр экспоненциального распределения длительностей
% t - правая граница интервала длительностей (левая - 0)

```

```

function Res = AvgInterval(Alpha, t)
    % Res = (1/a - (a*t + 1)/(a*exp(a*t)))/(1-exp(-a*t));
    Res = (1-(1+Alpha*t).*exp(-Alpha*t))./(Alpha.*(1-exp(-Alpha*t)));
end

```

```

}
}

```

ExpProb

```
% Вычисление вероятности при экспоненциальном распределении
% Alpha - параметр распределения
% t - значение случайной величины
```

```
function Res = ExpProb(Alpha, t)
    Res = 1-exp(-Alpha*t);
end
```

Init

```
% Формирование описания ССОИ
function [H, B, M, DLC, V, UseProb] = Init(TSync, TypPeriod, StressMode)

% классы заявок (1 - SYNC, 2..5 - PDO, 6 - HBEAT, 7 - USDO)
H = 1:7;

% количества модулей (СК, ИЛТН, АВВ, ДВВ, ДВыв)
M = [1, 2, 10, 8, 2]';

tc = TypPeriod / TSync; % число циклов между сообщениями (типовое)
tm = 50000000;

% матрица периодичности отправки заявок B(h,k)
%      S   P1  2   3   4   HB  USDO
B = [ 1   0   0   0   0   tc  0; % СК
      0   tm  tm  tm  tc  tc  0; % ИЛТН
      0   tm  tm  tm  tc  tc  0; % АВВ
      0   tc  0   0   0   tc  0; % ДВВ
      0   tc  0   0   0   tc  0; % ДВыв
    ];

if StressMode ~= 0
    B(B ~= 0) = 1; % в режиме максимальной нагрузки все сообщения отправляются в
каждом цикле
end

% матрица режима вычисления интенсивности:
% 0 - с заданной периодичностью
% 1 - с добавлением случайной составляющей (эксп. распределение)
% 2 - для флагов
%      S   P1  2   3   4   HB  USDO
UseProb = [
    0   0   0   0   0   0   0; % СК
    0   1   1   2   0   0   0; % ИЛТН
    0   1   1   2   0   0   0; % АВВ
    0   2   0   0   0   0   0; % ДВВ
    0   0   0   0   0   0   0; % ДВыв
    ];

% матрица значений DLC (h,k)
%      S   P1  2   3   4   HB  USDO
DLC = [ 0  -1  -1  -1  -1   1  -1; % СК
        -1  12  12   8   14   1  -1; % ИЛТН
        -1  12  12   4   14   1  -1; % АВВ
        -1   2  -1  -1  -1   1  -1; % ДВВ
        -1   2  -1  -1  -1   1  -1; % ДВыв
    ];

% матрица количества передаваемых значений V(h,k)
%      S   P1  2   3   4   HB  USDO
V = [ 0   0   0   0   0   0   0; % СК
      0  11  11   1  22   0   0; % ИЛТН
      0  12  12   1  24   0   0; % АВВ
      0  16   0   0   0   0   0; % ДВВ
      0  16   0   0   0   0   0; % ДВыв
    ];

end
```

LambdaPDO

```

% Вычисление средней интенсивности потока PDO одного класса от одного
% модуля при заданном параметре экспоненциального
% распределения длительностей и периоде синхронизации

% Alpha - параметр экспоненциального распределения длительностей
% t - период синхронизации
% Lmin - минимальная интенсивность (определяется периодичностью отправки PDO)
% Lmax - максимальная интенсивность (в каждом цикле синхронизации)

function Res = LambdaPDO(Alpha, t, Lmin, Lmax)
    Res = ExpProb(Alpha,t)./AvgInterval(Alpha,t) + (1-ExpProb(Alpha,t))/t;
    if Res>Lmax
        Res = Lmax;
    elseif Res<Lmin
        Res = Lmin;
    end
end

```

MsgLength

```

% Вычисление длины сообщения в битах
% Extended != 0 - расширенный формат сообщений
% DLC - поле DLC сообщения (DLC<0 - не считаем)
% Kbs - коэффициент вставки битов [0,1]

function Res = MsgLength(Extended, DLC, Kbs)

    if DLC < 0
        Res = 0;
    else

        if Extended ~= 0
            IDE = 1;
        else
            IDE = 0;
        end

        L = 44 + IDE * 20 + DLC * 8;          % минимальная длина сообщения

        if Extended ~= 0
            if DLC == 3
                a = 1;
            else
                a = 0;
            end
            AddBits = 10 + fix(8 * DLC / 5) + a;
        else
            if (DLC < 2) || (DLC == 3) || (DLC == 8)
                a = 1;
            else
                a = 0;
            end
            AddBits = 5 + fix(8 * DLC / 5) + a;
        end

        % увеличиваем с учетом коэффициента дополнительных бит
        Res = L + AddBits * Kbs;
    end
end

```

MsgLength_D

```

% Вычисление количества битов сообщении передаваемых на скорости данных
% DLC - длина поля данных (!= длина поля данных в байтах)

```

```

% Kbs - коэффициент вставки битов [0,1]
function Res = MsgLength_D( DLC, kbs )
% определение длины поля данных, в зависимости от значения DLC
if DLC <= 8
    N_Data = DLC*8;
elseif DLC <= 12
    N_Data = DLC*32-192;
else
    N_Data = DLC*128-1408;
end

% определение длины поля контрольной суммы, в зависимости от значения DLC
if DLC <= 10
    N_CRC = 28;
else
    N_CRC = 33;
end

% определяем параметр A, зависящий от значения DLC
switch DLC
    case {3, 8, 14, 15}
        A = 1;
    otherwise
        A = 0;
end

% Вычисляем количество битов определяем параметр A, зависящий от значения DLC
Res = 5+N_Data+N_CRC+(N_Data/5+A)*kbs;
end

```

MsgLength_N

```

% Вычисление количества битов сообщения передаваемых на номинальной скорости
% IDE != 0 - расширенный формат сообщений
% Kbs - коэффициент вставки битов [0,1]
function Res = MsgLength_N( IDE, kbs )
if IDE ~=0
    IDE=1;
end
Res = 29+19*IDE+(2+4*IDE)*kbs;
end

```

N2DLC

```

%Преобразование кода DLC в количество битов
function DLC = N2DLC( Bytes )
if Bytes <=8
    DLC = Bytes;
elseif Bytes <= 12
    DLC = 9;
elseif Bytes <= 16
    DLC = 10;
elseif Bytes <= 20
    DLC = 11;
elseif Bytes <= 24
    DLC = 12;
elseif Bytes <= 32
    DLC = 13;
elseif Bytes <= 48
    DLC = 14;
else
    DLC = 15;
end
end

```

SignalAlpha

```
% Вычисление параметра экспоненциального распределения времени
% выхода значения сигнала из заданного диапазона
% Aperture - диапазон (отклонение) в процентах
```

```
function Res = SignalAlpha(Aperture)
    % числовые значения получены по результатам аппроксимации
    % экспериментальной зависимости

    Res = 3.616 * exp(-1.898 * Aperture) + 0.0609;
end
```

USDOSpeed

```
% Вычисление скорости передачи USDO (байт/сек)
% Bytes - размер пакетов данных
% Ro - к-нт простоя
% kbs - к-нт дополнительных битов
% BMs - кол-во сообщений в блоке

function Res = UsdoSpeed(Bytes, Ro, kbs, BMs, T_N, T_D, IDE)
    % Вычисляем суммарное количество битов передающихся на номинальной скорости
    N0_Sum = (2+fix(Bytes/60) ...
        + (ceil(Bytes/60)-fix(Bytes/60)) ...
        + ceil(ceil(Bytes/60)/BMs)) ...
        * MsgLength_N(IDE, kbs);

    % Вычисляем суммарное количество битов передающихся на скорости данных
    N1_Sum = MsgLength_D (N2DLC(12), kbs) ...
        +MsgLength_D (N2DLC(8), kbs) ...
        +MsgLength_D (N2DLC(64), kbs)*fix(Bytes/60) ...
        +MsgLength_D (N2DLC(mod(Bytes,60)), kbs) ...
        +MsgLength_D (N2DLC(3), kbs)*ceil(ceil(Bytes/60)/BMs);

    % Вычисляем суммарное время передачи сообщения USDO для битов передающихся на
    номинальной скорости и на скорости данных (T_N и T_D необходимо сюда передавать)
    T_Sum = N0_Sum*T_N+N1_Sum*T_D;

    % Вычисляем фактическое время передачи сообщения USDO в свободное от высокоприоритетной
    информации время
    T_Sum_F = T_Sum/Ro;

    % Вычисляем скорость передачи USDO
    %Res = (Bytes*32000)/T_Sum_F;
    Res = Bytes/T_Sum_F;
end
```

Б.2 Имитационная модель для исследования параметров межмодульного взаимодействия в системе

```

namespace CanAnalyzer
{
    public class CCanMessage : INotifyPropertyChanged
    {
        protected string _name;           // наименование сообщения
        protected double _time_offs;     // смещение времени попытки передачи сообщения
относительно базовой точки
        protected bool _extended;        // признак расширенного формата сообщения
        protected int _id;               // идентификатор сообщения
        protected int _dlc;              // число байтов данных
        protected int[] _data = new int[8]; // байты данных
        protected int _bit_length;       // число битов в сообщении
        protected TBitStuffMode _bit_stuff; // режим вычисления кол-ва доп. битов

        /// <summary>Режим вычисления количества дополнительных битов в сообщении</summary>
        public enum TBitStuffMode : int
        {
            Unknown = 0,                 // режим не задан
            None,                         // доп. биты не учитываются
            Real,                         // вычисляется реальное кол-во доп. битов
            Max                           // учитывается максимально возможное кол-во доп. битов
        };

        public TBitStuffMode BitStuffMode {
            get { return _bit_stuff; }
            set { _bit_length = 0; _bit_stuff = value; }
        }

        /// <summary>Наименование сообщения</summary>
        public string Name
        {
            get { return _name; }
            set { _name = value; OnPropertyChanged(); }
        }

        /// <summary>Смещение времени попытки передачи сообщения относительно базовой
точки</summary>
        public double TimeOffsUs {
            get { return _time_offs; }
            set { _time_offs = value; OnPropertyChanged(); }
        }

        /// <summary>Признак расширенного формата</summary>
        public bool Extended
        {
            get { return _extended; }
            set { _bit_length = 0; _extended = value; OnPropertyChanged(); }
        }

        /// <summary>Идентификатор сообщения</summary>
        public int Id
        {
            get { return _id; }
            set { _bit_length = 0; _id = value; OnPropertyChanged(); }
        }

        /// <summary>Число байтов данных</summary>
        public int DLC
        {
            get { return _dlc; }
        }
    }
}

```

```

    set { _bit_length = 0; _dlc = value; OnPropertyChanged(); }
}

/// <summary>Число битов в сообщении</summary>
public int BitLength
{
    get { if (_bit_length <= 0) _bit_length = CalcBitLength(); return _bit_length; }
    set { _dlc = value; OnPropertyChanged(); }
}

/// <summary>Байты данных</summary>
public int this[int Idx]
{
    get { return _data[Idx]; }
    set { _bit_length = 0; _data[Idx] = value; OnPropertyChanged(); }
}

/// <summary>Массив данных</summary>
public int[] Data { get { return _data; } }

/// <summary>Вычисление количества битов в сообщении</summary>
protected int CalcBitLength()
{
    if ((BitStuffMode == TBitStuffMode.Unknown) || (BitStuffMode ==
TBitStuffMode.Real))
        throw new SystemException("Заданный режим вычисления количества дополнительных
битов не поддерживается");

    // Вычисляем количество основных битов сообщения
    int N = 44 + 8*DLC + (Extended ? 20 : 0);

    // Вычисляем количество дополнительных битов
    int Naux = 0;
    if (BitStuffMode == TBitStuffMode.Max) {
        if (Extended) {
            Naux = 10 + 8 * DLC / 5;
            if (DLC == 3)
                Naux++;
        } else {
            Naux = 5 + 8 * DLC / 5;
            if ((DLC == 0) || (DLC == 1) || (DLC == 3) || (DLC == 8))
                Naux++;
        }
    }
    return N + Naux;
}

// Обработка изменения свойств
public event PropertyChangedEventHandler PropertyChanged;
public void OnPropertyChanged([CallerMemberName]string prop = "")
{
    if (PropertyChanged != null)
        PropertyChanged(this, new PropertyChangedEventArgs(prop));
}
}
}

```

Класс, представляющий модуль (узел) *CAN FD*

```

/// <summary>
/// Класс, представляющий модуль (узел) CAN FD
/// </summary>
public class CCanUnit : INotifyPropertyChanged
{
    protected string _name;
    protected List<CCanMessage> _msg_list;
    /// <summary>Информативное имя модуля</summary>

```

```

public string Name {
    get { return _name; }
    set { _name = value; OnPropertyChanged(); }
}
public List<CCanMessage> MsgList { get { return _msg_list; } }
public CCanUnit() : base() {
    _msg_list = new List<CCanMessage>();
    _name = "Модуль";
}
// Обработка изменения свойств
public event PropertyChangedEventHandler PropertyChanged;
public void OnPropertyChanged([CallerMemberName]string prop = "")
{
    if (PropertyChanged != null)
        PropertyChanged(this, new PropertyChangedEventArgs(prop));
}
}
}

```

Формирование отчета

```

namespace CanFDAnalyzer
{
    public class CMsgTime
    {
        protected double _bit_time_us; // время передачи одного бита

        /// <summary>Сообщение, которому соответствует запись</summary>
        protected CCanMessage _message;

        public int MsgId { get { return _message.Id; } }

        /// <summary>Название модуля</summary>
        public string UnitName { get { return _unit.Name; } }

        /// <summary>Время постановки сообщения в очередь на передачу</summary>
        public double TimeEnqueuedUs { get { return _message.TimeOffsUs; } }

        /// <summary>Время начала передачи сообщения</summary>
        public double TimeStartTxUs { get; set; }

        /// <summary>Время передачи сообщения (с момента начала до момента
        окончания)</summary>
        public double DeliveredTimeUs { get { return TimeStartTxUs + TimeTxUs; } }

        /// <summary>Длительность доставки сообщения (с момента постановки в очередь до
        момента окончания передачи)</summary>
        public double DeliveryTimeUs { get { return DeliveredTimeUs - TimeEnqueuedUs; } }

        public CMsgTime(double BitTimeUs, CCanUnit AUnit, CCanMessage AMsg)
        {
            _bit_time_us = BitTimeUs;
            _unit = AUnit;
            _message = AMsg;
        }
    }
}

```

Приложение В.
Выборки результатов моделирования
с использованием вероятностной и имитационной моделей

Таблица В.1 – Значения нагрузки на СПД при скорости работы интерфейса $CAN FD = 500$ кбит/с

| № эксперимента | Нагрузка СПД, % | | |
|----------------|---------------------------------------|--|--|
| | Реальная ССОИ ($Y_{\text{эсп}}$) | Имитационная модель ($Y_{\text{им}}$) | Вероятностная модель ($Y_{\text{вер}}$) |
| 1 | 2 | 3 | 4 |
| 1 | 24,08 | 24,04 | 23,74 |
| 2 | 24,77 | 24,79 | 23,74 |
| 3 | 22,84 | 22,85 | 23,74 |
| 4 | 23,5 | 23,41 | 23,74 |
| 5 | 23,9 | 23,90 | 23,74 |
| 6 | 23,69 | 23,78 | 23,74 |
| 7 | 23,36 | 23,42 | 23,74 |
| 8 | 25,63 | 25,57 | 23,74 |
| 9 | 23,57 | 23,61 | 23,74 |
| 10 | 24,16 | 24,17 | 23,74 |
| 11 | 23,54 | 23,53 | 23,74 |
| 12 | 22,39 | 22,44 | 23,74 |
| 13 | 23,82 | 23,81 | 23,74 |
| 14 | 22,98 | 23,01 | 23,74 |
| 15 | 23,9 | 23,88 | 23,74 |
| 16 | 22,6 | 22,60 | 23,74 |
| 17 | 24,62 | 24,63 | 23,74 |
| 18 | 24,28 | 24,31 | 23,74 |
| 19 | 22,65 | 22,53 | 23,74 |
| 20 | 24,64 | 24,64 | 23,74 |
| 21 | 22,92 | 22,95 | 23,74 |
| 22 | 24,53 | 24,49 | 23,74 |
| 23 | 25,15 | 25,06 | 23,74 |
| 24 | 25,45 | 25,36 | 23,74 |
| 25 | 24,14 | 24,13 | 23,74 |
| 26 | 24,11 | 24,09 | 23,74 |
| 27 | 22,85 | 22,86 | 23,74 |
| 28 | 23,8 | 23,73 | 23,74 |
| 29 | 23,74 | 23,77 | 23,74 |
| 30 | 22,62 | 22,57 | 23,74 |

Таблица В.2 – Значения скорости передачи *USDO* при скорости работы интерфейса *CAN FD = 500* кбит/с

| № эксперимента | Скорость передачи, байт/с | | |
|----------------|--|--|--|
| | Реальная ССОИ ($Y_{\text{эксп}}$) | Имитационная модель ($Y_{\text{им}}$) | Вероятностная модель ($Y_{\text{вер}}$) |
| 1 | 2 | 3 | 4 |
| 1 | 27138 | 27065 | 27530 |
| 2 | 27205 | 27225 | 27530 |
| 3 | 27938 | 27754 | 27530 |
| 4 | 27712 | 27773 | 27530 |
| 5 | 27234 | 27133 | 27530 |
| 6 | 27203 | 27173 | 27530 |
| 7 | 28104 | 28084 | 27530 |
| 8 | 26974 | 27008 | 27530 |
| 9 | 26764 | 26816 | 27530 |
| 10 | 27240 | 27328 | 27530 |
| 11 | 27129 | 27160 | 27530 |
| 12 | 26770 | 26864 | 27530 |
| 13 | 27892 | 27867 | 27530 |
| 14 | 27028 | 27028 | 27530 |
| 15 | 28246 | 28306 | 27530 |
| 16 | 27062 | 27089 | 27530 |
| 17 | 27315 | 27261 | 27530 |
| 18 | 28631 | 28731 | 27530 |
| 19 | 27249 | 27247 | 27530 |
| 20 | 26942 | 27013 | 27530 |
| 21 | 27334 | 27291 | 27530 |
| 22 | 27811 | 27743 | 27530 |
| 23 | 26903 | 26970 | 27530 |
| 24 | 27824 | 27763 | 27530 |
| 25 | 28446 | 28476 | 27530 |
| 26 | 27151 | 27176 | 27530 |
| 27 | 27964 | 27914 | 27530 |
| 28 | 27759 | 27729 | 27530 |
| 29 | 28079 | 27940 | 27530 |
| 30 | 27995 | 27897 | 27530 |

Таблица В.3 – Значения нагрузки на СПД при скорости работы интерфейса $CAN FD = 200$ кбит/с

| № эксперимента | Нагрузка СПД, % | | |
|----------------|---------------------------------------|--|--|
| | Реальная ССОИ ($Y_{\text{эсп}}$) | Имитационная модель ($Y_{\text{им}}$) | Вероятностная модель ($Y_{\text{вер}}$) |
| 1 | 2 | 3 | 4 |
| 1 | 49,90 | 49,65 | 48,28 |
| 2 | 45,23 | 45,09 | 48,28 |
| 3 | 46,79 | 47,04 | 48,28 |
| 4 | 47,83 | 47,58 | 48,28 |
| 5 | 46,49 | 46,40 | 48,28 |
| 6 | 47,06 | 47,31 | 48,28 |
| 7 | 46,94 | 46,98 | 48,28 |
| 8 | 49,33 | 49,27 | 48,28 |
| 9 | 49,96 | 50,09 | 48,28 |
| 10 | 47,05 | 46,76 | 48,28 |
| 11 | 48,16 | 48,03 | 48,28 |
| 12 | 49,13 | 49,04 | 48,28 |
| 13 | 48,98 | 49,04 | 48,28 |
| 14 | 47,15 | 47,24 | 48,28 |
| 15 | 47,54 | 47,97 | 48,28 |
| 16 | 48,15 | 48,06 | 48,28 |
| 17 | 48,09 | 48,13 | 48,28 |
| 18 | 47,96 | 47,92 | 48,28 |
| 19 | 47,35 | 47,04 | 48,28 |
| 20 | 50,17 | 49,54 | 48,28 |
| 21 | 48,92 | 48,54 | 48,28 |
| 22 | 47,56 | 47,42 | 48,28 |
| 23 | 48,19 | 48,39 | 48,28 |
| 24 | 49,53 | 49,44 | 48,28 |
| 25 | 48,00 | 48,09 | 48,28 |
| 26 | 47,76 | 47,83 | 48,28 |
| 27 | 47,45 | 47,85 | 48,28 |
| 28 | 48,10 | 47,82 | 48,28 |
| 29 | 49,57 | 49,48 | 48,28 |
| 30 | 48,16 | 47,94 | 48,28 |

Таблица В.4 – Значения скорости передачи *USDO* при скорости работы интерфейса *CAN FD* = 200 кбит/с

| № эксперимента | Скорость передачи, байт/с | | |
|----------------|--|--|--|
| | Реальная ССОИ ($Y_{\text{эксп}}$) | Имитационная модель ($Y_{\text{им}}$) | Вероятностная модель ($Y_{\text{вер}}$) |
| 1 | 2 | 3 | 4 |
| 1 | 8063 | 8063 | 8130 |
| 2 | 8152 | 8122 | 8130 |
| 3 | 8084 | 8050 | 8130 |
| 4 | 8137 | 8106 | 8130 |
| 5 | 8396 | 8409 | 8130 |
| 6 | 8109 | 8088 | 8130 |
| 7 | 8267 | 8259 | 8130 |
| 8 | 8065 | 8048 | 8130 |
| 9 | 7831 | 7811 | 8130 |
| 10 | 8152 | 8123 | 8130 |
| 11 | 7744 | 7723 | 8130 |
| 12 | 8124 | 8125 | 8130 |
| 13 | 8135 | 8079 | 8130 |
| 14 | 8397 | 8372 | 8130 |
| 15 | 8051 | 8024 | 8130 |
| 16 | 8345 | 8326 | 8130 |
| 17 | 7782 | 7786 | 8130 |
| 18 | 8229 | 8259 | 8130 |
| 19 | 8066 | 8088 | 8130 |
| 20 | 8188 | 8165 | 8130 |
| 21 | 8099 | 8100 | 8130 |
| 22 | 8129 | 8131 | 8130 |
| 23 | 8190 | 8210 | 8130 |
| 24 | 8117 | 8091 | 8130 |
| 25 | 8051 | 8061 | 8130 |
| 26 | 8273 | 8297 | 8130 |
| 27 | 7884 | 7870 | 8130 |
| 28 | 7756 | 7711 | 8130 |
| 29 | 8035 | 8035 | 8130 |
| 30 | 7836 | 7823 | 8130 |

Таблица В.5 – Значения нагрузки на СПД при скорости работы интерфейса $CAN FD = 100$ кбит/с

| № эксперимента | Нагрузка СПД, % | | |
|----------------|---------------------------------------|--|--|
| | Реальная ССОИ ($Y_{\text{экс}}$) | Имитационная модель ($Y_{\text{им}}$) | Вероятностная модель ($Y_{\text{вер}}$) |
| 1 | 2 | 3 | 4 |
| 1 | 78,28 | 78,19 | 75,92 |
| 2 | 74,80 | 75,16 | 75,92 |
| 3 | 75,46 | 75,60 | 75,92 |
| 4 | 78,15 | 78,43 | 75,92 |
| 5 | 74,40 | 74,58 | 75,92 |
| 6 | 75,11 | 75,29 | 75,92 |
| 7 | 76,34 | 75,60 | 75,92 |
| 8 | 78,38 | 78,28 | 75,92 |
| 9 | 77,39 | 77,35 | 75,92 |
| 10 | 75,40 | 75,51 | 75,92 |
| 11 | 76,20 | 76,41 | 75,92 |
| 12 | 75,71 | 75,31 | 75,92 |
| 13 | 76,62 | 77,03 | 75,92 |
| 14 | 77,53 | 77,43 | 75,92 |
| 15 | 76,88 | 77,07 | 75,92 |
| 16 | 71,90 | 71,72 | 75,92 |
| 17 | 76,46 | 76,29 | 75,92 |
| 18 | 76,63 | 76,82 | 75,92 |
| 19 | 75,94 | 76,11 | 75,92 |
| 20 | 75,33 | 74,86 | 75,92 |
| 21 | 78,28 | 78,27 | 75,92 |
| 22 | 76,13 | 76,01 | 75,92 |
| 23 | 75,91 | 75,95 | 75,92 |
| 24 | 76,02 | 75,92 | 75,92 |
| 25 | 77,13 | 77,27 | 75,92 |
| 26 | 75,55 | 75,78 | 75,92 |
| 27 | 78,64 | 78,53 | 75,92 |
| 28 | 76,82 | 76,67 | 75,92 |
| 29 | 77,49 | 77,28 | 75,92 |
| 30 | 75,52 | 75,73 | 75,92 |

Таблица В.6 – Значения скорости передачи *USDO* при скорости работы интерфейса *CAN FD* = 100 кбит/с

| № эксперимента | Скорость передачи, байт/с | | |
|----------------|--|--|--|
| | Реальная ССОИ ($Y_{\text{эксп}}$) | Имитационная модель ($Y_{\text{им}}$) | Вероятностная модель ($Y_{\text{вер}}$) |
| 1 | 2 | 3 | 4 |
| 1 | 2266 | 2254 | 2269 |
| 2 | 2258 | 2270 | 2269 |
| 3 | 2248 | 2274 | 2269 |
| 4 | 2157 | 2156 | 2269 |
| 5 | 2320 | 2334 | 2269 |
| 6 | 2217 | 2213 | 2269 |
| 7 | 2312 | 2305 | 2269 |
| 8 | 2312 | 2285 | 2269 |
| 9 | 2378 | 2370 | 2269 |
| 10 | 2149 | 2147 | 2269 |
| 11 | 2148 | 2161 | 2269 |
| 12 | 2260 | 2265 | 2269 |
| 13 | 2186 | 2189 | 2269 |
| 14 | 2371 | 2367 | 2269 |
| 15 | 2172 | 2168 | 2269 |
| 16 | 2250 | 2256 | 2269 |
| 17 | 2303 | 2303 | 2269 |
| 18 | 2312 | 2331 | 2269 |
| 19 | 2334 | 2333 | 2269 |
| 20 | 2246 | 2237 | 2269 |
| 21 | 2141 | 2121 | 2269 |
| 22 | 2319 | 2314 | 2269 |
| 23 | 2314 | 2295 | 2269 |
| 24 | 2380 | 2381 | 2269 |
| 25 | 2294 | 2282 | 2269 |
| 26 | 2279 | 2298 | 2269 |
| 27 | 2265 | 2240 | 2269 |
| 28 | 2376 | 2368 | 2269 |
| 29 | 2279 | 2255 | 2269 |
| 30 | 2255 | 2257 | 2269 |

**Приложение Г.
Документы о внедрении
результатов работы**



ООО «СКБ Электронного Приборостроения»
 Адрес места нахождения: 142432, Московская обл.,
 г.о. Черноголовка, г. Черноголовка, Проезд 1-й, зд. 2А
 Адрес для переписки: а/я 1005, 142432,
 Московская обл., г. Черноголовка
 Тел./факс: +7 (49652) 49-385, 49-446, 49-535,
 49-486, 43-301, 43-314, 43-316
skb@skbchg.ru | www.skbchg.ru
 ОКПО 53936919, ОГРН 1035006114571
 ИНН/КПП 5031044021/503101001

13.02.2025г № 42

На № _____ от _____

Муженко А.С.

СПРАВКА О ВНЕДРЕНИИ

Дана Муженко Александру Сергеевичу в том, что применение разработанных в его кандидатской диссертации методов и моделей, а именно метода описания модульной системы сбора и обработки информации (ССОИ), метода оценки времени передачи сообщений интерфейса *CAN FD*, вероятностного метода оценки параметров межмодульного взаимодействия в ССОИ, а также использующих перечисленные методы вероятностной и имитационной моделей модульной ССОИ на основе протокола *CANopen FD*, способствует повышению эффективности функционирования и улучшению технических и эксплуатационных характеристик ССОИ за счёт уменьшения числа ошибок на ранних стадиях проектирования и оптимизации процессов внутрисистемного взаимодействия.

Программное обеспечение, созданное с участием Муженко А.С. и реализующее указанные методы и модели, использовано в процессе проектирования модульного контроллера присоединения, разрабатываемого нашей организацией и планируемого к серийному выпуску для использования на объектах электроэнергетики с учётом исполнения требований по обеспечению технологической независимости и безопасности критической информационной инфраструктуры Российской Федерации.

Директор

ООО «СКБ Электронного Приборостроения»



С.А. Михеев

Приложение Д.
Свидетельства о регистрации программ
для ЭВМ по теме исследования

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2024691739

Проверка XML-описания модульных систем на основе
протокола CANopen FDПравообладатель: *Плотников Дмитрий Александрович (RU)*Авторы: *Плотников Дмитрий Александрович (RU),
Муженко Александр Сергеевич (RU)*

Заявка № 2024690987

Дата поступления 12 декабря 2024 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 24 декабря 2024 г.

*Руководитель Федеральной службы
по интеллектуальной собственности*ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат: 0692e761a6300c15412601670bca2026
Владелец: **Зубов Юрий Сергеевич**
Действителен с 10.07.2024 по 03.10.2025*Ю.С. Зубов*

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2024691042

**Вероятностная модель модульной системы на базе
протокола CANopen FD**

Правообладатель: *Плотников Дмитрий Александрович (RU)*

Авторы: *Плотников Дмитрий Александрович (RU),
Музенко Александр Сергеевич (RU)*

Заявка № 2024689351

Дата поступления **05 декабря 2024 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **19 декабря 2024 г.**

*Руководитель Федеральной службы
по интеллектуальной собственности*

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат 0692e761ae7000b5412401670bca2024
Владелец **Зубов Юрий Сергеевич**
Действителен с 10.11.2024 по 03.10.2025

Ю.С. Зубов

