

ПРОЕКТИРОВАНИЕ, ИЗГОТОВЛЕНИЕ И ВВОД
В ЭКСПЛУАТАЦИЮ ОБОРУДОВАНИЯ
ОБЪЕКТОВ АТОМНОЙ ОТРАСЛИ
DESIGN, MANUFACTURE AND COMMISSIONING
COMMISSIONING OF EQUIPMENT
NUCLEAR INDUSTRY FACILITIES

УДК 004.75

DOI 10.26583/gns-2023-03-03

EDN FYOEP0



Метод описания модульной информационно-измерительной системы
на основе протокола CANopen с учетом межмодульных
информационных связей

Д.А. Плотников , А.С. Муженко , В.И. Лачин

Южно-Российский государственный политехнический университет (НПИ)

им. М.И. Платова, г. Новочеркасск, Россия

muzhenko97@mail.ru

Аннотация. Современные информационно-измерительные и управляющие системы – ИИУС – обычно имеют модульную структуру. Удобными средствами организации межмодульного взаимодействия являются интерфейсы CAN (CAN FD) и протокол CANopen. Стандартные средства, предусмотренные спецификацией CANopen, ориентированы на описание отдельных модулей и не обеспечивают целостность и непротиворечивость конфигурации ИИУС в целом. В данной статье рассматриваются причины возможных ошибок конфигурирования и предлагается метод описания ИИУС на базе протокола CANopen, обеспечивающий согласованную настройку модулей системы. Цель достигается за счёт дополнения стандартного описания модулей явным описанием сообщений, передаваемых модулями друг другу, с указанием источника, приёмников и перечня параметров. Разработана схема информационных потоков в модульной ИИУС, предложена общая структура описания системы, включающая параметры модулей и параметры сообщений. Подробно рассмотрены структуры данных, описывающие модули и сообщения, особое внимание уделено обеспечению целостности межмодульных информационных связей. Показано, что предлагаемый метод позволяет обнаружить или исключить такие ошибки конфигурирования, как сообщения без источника или без получателей, дублирование передачи параметров, попытка передачи несуществующего параметра, отсутствие передачи некоторых параметров, отсутствие источника у принимаемого параметра. Предложено использование языка XML для реализации метода описания ИИУС, обеспечивающее автоматическую проверку конфигурации системы стандартными средствами языка на основе схемы документа. Отмечено, что использование метода в программе моделирования и конфигурирования модульной системы вибромониторинга обеспечило выявление ошибок несогласованной или неполной настройки отдельных модулей на самых ранних этапах проектирования.

Ключевые слова: модульная система, межмодульное взаимодействие, протокол CANopen, сетевое взаимодействие, интерфейс CAN, обмен данными, информационные объекты, классы сообщений, объектный словарь, параметры модуля, конфигурирование модульных систем.

Для цитирования: Плотников Д.А., Муженко А.С., Лачин В.И. Метод описания модульной информационно-измерительной системы на основе протокола CANopen с учетом межмодульных информационных связей. *Глобальная ядерная безопасность*. 2023;13(3):26–36. <https://doi.org/10.26583/gns-2023-03-03>

A method of describing a modular information and measurement system based on the
CANopen protocol, taking into account intermodule information links

Dmitriy A. Plotnikov , Alexander S. Muzhenko , Vyacheslav I. Lachin

South Russian State Polytechnic University (NPI), Novocherkassk, Rostov region, Russia

muzhenko97@mail.ru

Abstract. Modern information and measuring and control systems (IMCS) usually have a modular structure. Convenient means of organizing intermodule interaction are the CAN interfaces (CAN FD) and the CANopen protocol. The standard tools provided by the CANopen specification are focused on the description of individual modules and do not ensure the integrity and consistency of the IMCS configuration as a whole. This article discusses the causes of possible configuration errors and suggests a method for describing the IMCS based on the CANopen protocol, which ensures consistent configuration of the system modules. The goal is achieved by supplementing the standard description of modules with an explicit description of the messages transmitted by modules to each other, indicating the source, receivers and a list of parameters. A scheme of information flows in a modular IMCS is developed, a general structure of the system description is proposed, including module parameters and message parameters. Data structures describing modules and messages are considered in detail, special attention is paid to ensuring the integrity of intermodule information links. It is shown that the proposed method makes it possible to detect or exclude configuration errors such as messages without a source or without recipients, duplication of parameter transmission, an attempt to transmit a non-existent parameter, the absence of transmission of some parameters, the absence of a source for the received parameter. The use of the XML language for the implementation of the IMCS description method is proposed, which provides automatic verification of the system configuration by standard language means based on the document schema. It is noted that the use of the method in the modeling and configuration program of the modular vibration monitoring system provided the identification of errors of inconsistent or incomplete configuration of individual modules at the earliest stages of design.

Keywords: modular system, intermodule interaction, CANopen protocol, network interaction, CAN interface, data exchange, information objects, message classes, object dictionary, module parameters, configuration of modular systems.

For citation: Plotnikov D.A., Muzhenko A.S., Lachin V.I. A method of describing a modular information and measurement system based on the CANopen protocol, taking into account intermodule information links. *Global nuclear safety*. 2023;13(3):26–36 (In Russ.) <https://doi.org/10.26583/gns-2023-03-03>

Введение

Современные информационно-измерительные и управляющие системы (ИИУС) – такие, например, как системы вибромониторинга турбоагрегатов или контроллеры АСУ ТП электрических подстанций – обычно имеют модульную структуру. Такой подход позволяет с использованием сравнительно небольшой номенклатуры унифицированных модулей скомпоновать систему, обеспечивающую наиболее эффективное решение конкретной прикладной задачи для конкретного объекта (рис. 1).

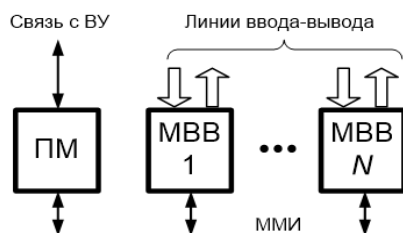


Рисунок 1. Типовая схема модульной ИИУС: ВУ – верхний уровень; МВВ – модули ввода-вывода; ММИ – межмодульный интерфейс; ПМ – процессорный модуль

Figure 1. Typical scheme of a modular IMCS: ВУ – upper level; МВВ – input/output modules; ММИ – inter-module interface; ПМ – processor module

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 ее 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 ее состав, которое в разных реализациях ИИУС может изменяться в весьма широких пределах. Следовательно, для обеспечения и оценки качества функционирования модульных ИИУС на как можно более ранних

этапах проектирования требуются методы и средства описания, моделирования и анализа сетевого межмодульного взаимодействия, учитывающие особенности решаемых задач.

С точки зрения авторов одними из наиболее перспективных интерфейсов межмодульного взаимодействия в ИИУС являются интерфейсы *CAN*¹ и его улучшенная версия *CAN FD*². Они предназначены для создания систем, работающих в режиме реального времени, обеспечивают высокую достоверность обмена данными, имеют развитый механизм автоматического обнаружения ошибок. Кроме того, стоимость их аппаратной реализации невысока, а производители электронных компонентов и программного обеспечения (ПО) предоставляют разработчикам всестороннюю поддержку.

Упомянутые интерфейсы исследованы достаточно хорошо: этому посвящены работы [1-3], ряд методов и моделей для оценки и повышения эффективности межмодульного взаимодействия разработан авторами [4-7]. Однако, как *CAN*, так и *CAN FD* реализуют только физический и канальный уровни модели *OSI*, чего обычно достаточно для решения лишь простейших задач межмодульного взаимодействия. Однако если в состав ИИУС входят десятки модулей, которые обмениваются значениями сотен параметров, возникает необходимость обеспечения адресации и синхронизации модулей, коррекции ошибок взаимодействия, конфигурирования (настройки) как отдельных модулей, так и системы в целом. Для решения этих задач необходимо использование протоколов более высокого уровня. На основе интерфейсов *CAN* и *CAN FD* разработано несколько стандартных высокоуровневых протоколов³, в частности *CANopen*⁴. Этот протокол общего назначения предназначен для использования в распределенных модульных ИИУС реального времени и обеспечивает пять верхних уровней модели *OSI*.

Основным элементом логической структуры модуля (узла) *CANopen* является объектный словарь. Его можно рассматривать как таблицу, в которой хранятся параметры настройки и сведения о текущем состоянии устройства или процесса. Обращения к записям словаря (объектам) выполняются с использованием 16-разрядного индекса. Каждый объект может содержать до 256 подобъектов, которые адресуются с помощью восьмиразрядного субиндекса. Взаимодействие устройств реализуется в виде обращений к объектным словарям друг друга. Текущие и накопленные данные можно извлечь из устройства путем чтения соответствующих объектов его словаря. И наоборот: запись в какой-либо объект может быть интерпретирована устройством, как сигнал, инициирующий выполнение некоторых действий.

Постановка задачи исследования

Для получения доступа к объектам словаря в рабочем режиме ИИУС обычно используется служба *Process Data Objects (PDO)*⁵. Каждое сообщение этой службы содержит одно или несколько значений величин из объектного словаря передающего модуля системы, измеренных или вычисленных этим модулем. Службой *PDO*

¹ ISO 11898-1:2015 Road vehicles – Controller area network (CAN) – Part 1: Data link layer and physical signaling [Электронный ресурс]. – URL: <https://www.iso.org/standard/63648.html> (дата обращения: 22.07.2023)

² CAN with Flexible Data-Rate. Version 1.1. Aug.2011 [Электронный ресурс]. Robert Bosch GmbH. – URL: https://www.semiconductors.bosch.de/media/pdf/.../canliteratur/can_fd.pdf (дата обращения: 22.07.2023).

³ Standardized higher-layer protocols [Электронный ресурс]. © 2021 CAN in Automation. – URL: <https://www.can-cia.org/can-knowledge/hlp/standardized-higher-layer-protocols/> (дата обращения: 30.07.2023).

⁴ CiA 301. CANopen application layer and communication profile. Version: 4.2.0 [Электронный ресурс]. © CAN in Automation (CiA) 2011. – URL: <https://www.can-cia.org> (дата обращения: 30.07.2023)

⁵ Там же.

предусмотрена гибкая настройка режимов передачи: возможна отправка сообщений по изменению данных, по событию синхронизации, периодическая, допустимы также некоторые комбинации этих режимов. При необходимости модуль может формировать сообщения *PDO* разных типов, отличающихся содержанием и режимами передачи. Например, в *PDO* первого типа с заданной периодичностью передаются текущие измеренные значения параметров, *PDO* второго типа содержит признаки состояния модуля и передается только при их изменении. Содержание каждого *PDO* задается параметрами его разметки (*PDO Mapping*, рис. 2) при настройке модулей. 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 рисунке 3 показан пример передачи параметров вибрации, измеренных модулем 1, в контроллер системы вибромониторинга (модуль *N*) с использованием службы *PDO*.

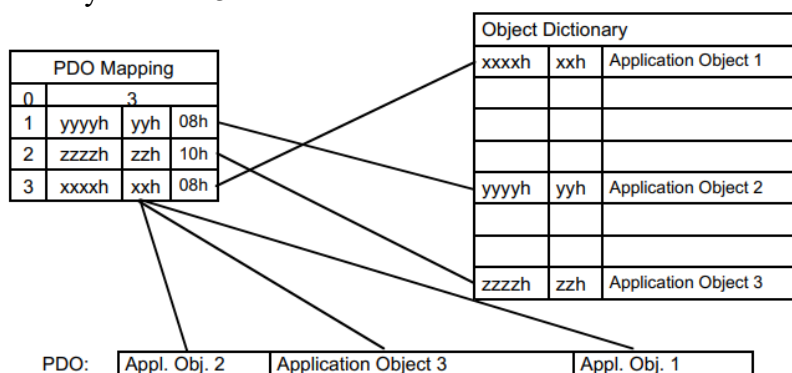


Рисунок 2. Разметка *PDO* (*PDO Mapping*), см. CiA 301. CANopen application (см. постр. сн. 4)

Figure 2. *PDO Mapping*, see CiA 301. CANopen application (page footnote 4)

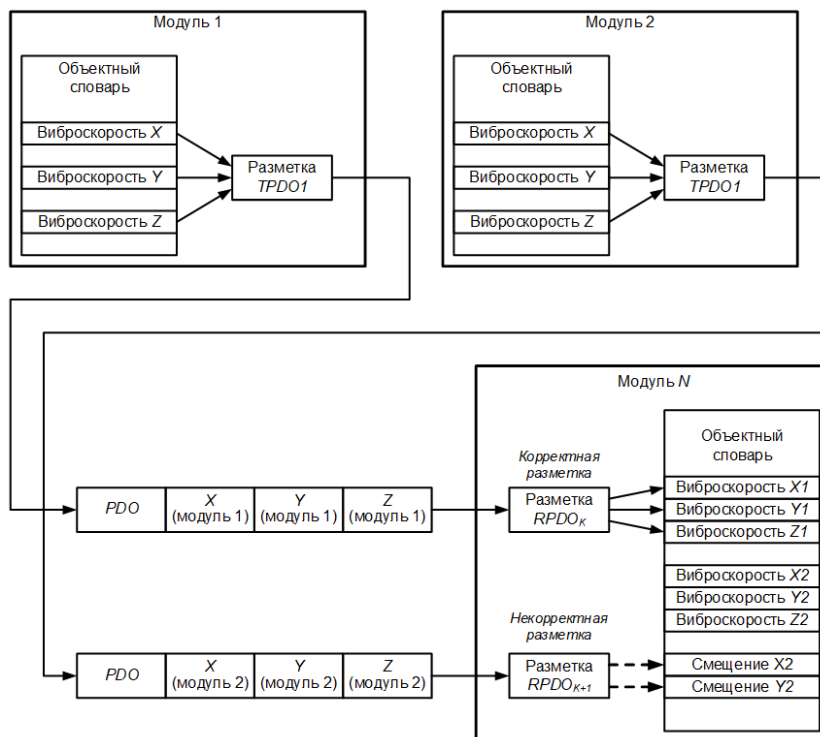


Рисунок 3. Пример организации межмодульного взаимодействия в системе вибромониторинга с использованием службы *PDO*

Figure 3. Example of organization of inter-module interaction in vibration monitoring system using *PDO* service

Описанный механизм обмена данными обладает чрезвычайной гибкостью, но при проектировании ИИУС в процессе ее адаптации к конкретному объекту управления он требует настройки с учетом 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 – приемниками, то есть задать информационные связи между объектными словарями модулей. В процессе настройки этих связей перед вводом ИИУС в эксплуатацию возможно возникновение трудно выявляемых ошибок. Дело в том, что в соответствии со стандартом⁶ разметки *TPDO* и *RPDO* взаимодействующих модулей настраиваются независимо в процессе создания отдельных *DCF*-файлов конфигурации для каждого модуля. Это может привести к ситуации, изображенной на рис. 3 в модуле *N*. Пусть, например, *TPDO1* модуля 1 настроено для передачи текущих значений ортогональных составляющих вибрации *X*, *Y*, *Z*, измеренных в некоторой точке объекта, 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 вибрации в соответствующие объекты словаря. Если же при настройке *RPDO* допущена ошибка (*RPDO_{K+1}*), то часть параметров вибрации, измеренных модулем 2, запишется в объекты словаря, предназначенные для хранения других величин, а часть и вовсе будет потеряна. Некорректное указание размеров данных при настройке разметок *TPDO* и *RPDO* приводит к возникновению еще более сложно выявляемых проблем. При настройке ИИУС, модули которых обмениваются значениями сотен параметров, вероятность появления описанных ошибок весьма велика.

Таким образом, возникает задача разработки методов и средств описания ИИУС, исключающих возможность появления подобных ошибок.

Разработка метода описания ИИУС

Сформулированная задача решена авторами путем дополнения описания ИИУС, предусматривающего независимое создание конфигураций отдельных модулей и регламентированного стандартом⁷, явным описанием межмодульного взаимодействия с размещением всех элементов описания в единственном файле конфигурации системы. С этой целью к описанию системы добавлены информационные объекты, соответствующие *PDO* и включающие в себя всю необходимую информацию не только о содержании, но и о модуле-источнике и модулях-приемниках сообщений.

Для формирования структуры описания модульной ИИУС, основанной на протоколе *CANopen*, разработана схема информационных потоков (рис. 4). В соответствии с ней ИИУС 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* передает 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

⁶ 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: 30.07.2023).

⁷ Там же.

тем выше, чем меньше номер класса. Сообщения с высшим приоритетом передаются и, соответственно, доставляются в первую очередь.

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-приемников, 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 стандартом⁸ параметры PDO , которые не имеют прямого отношения к разработанному методу и, в связи с этим, на рисунке 4 не показаны.

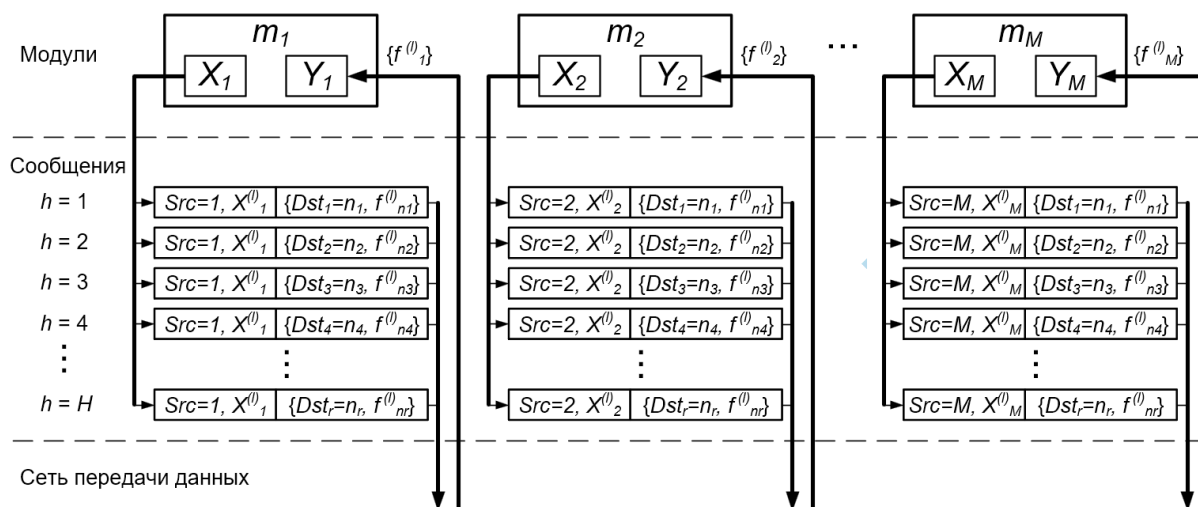


Рисунок 4. Схема информационных потоков в модульной ИИУС

Figure 4. Scheme of information flows in a modular IMCS

На основе разработанной схемы создан метод описания ИИУС, рассматривающий систему как совокупность модулей, взаимодействующих путем обмена сообщениями. На верхнем уровне описания определены две группы параметров (рис. 5): параметры модулей (*Nodes*) и параметры сообщений (*Messages*).



Рисунок 5. Общая структура описания ИИУС

Figure 5. General structure of the IMCS description

Список модулей *Nodes* в соответствии с CiA 301. CANopen application layer and communication profile. Version: 4.2.0 – 2011 (см. постр. сноски 4) 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

⁸ См. CiA 306. CANopen electronic data sheet... – постр. сноска 6

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 рисунке 6. 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 повлечет за собой необходимость выявления и коррекции множества ссылок. Оставшиеся общие параметры являются вспомогательными, их перечень при необходимости может дополняться и изменяться.

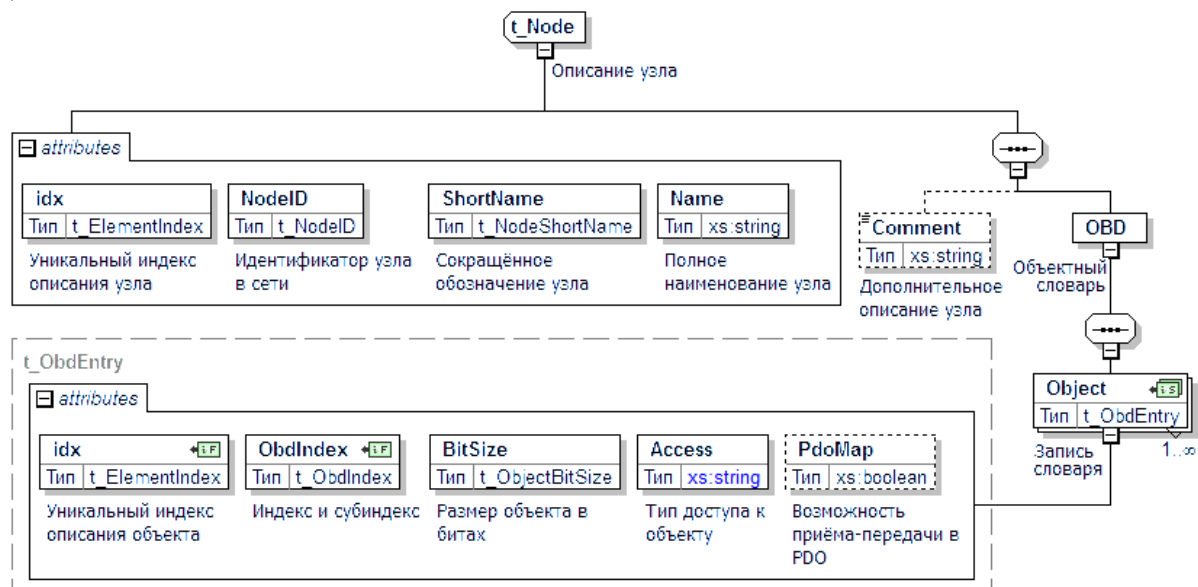


Рисунок 6. Описание модуля ИИУС

Figure 6. IMCS module description

Список *OBD* содержит описания отдельных объектов словаря *Object* типа *t_ObdEntry*. Для каждого объекта задается уникальный индекс *idx*, необходимый для ссылок на этот объект из других частей описания, а также параметры, в соответствии с CiA 301. CANopen application layer and communication profile. Version: 4.2.0 – 2011 (см. постр. сноску 4): индекс и субиндекс (*ObdIndex*), разрядность данных в битах (*BitSize*), режим доступа (*Access*), признак возможности приема 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.

Для описания сообщений ИИУС используется структура данных, упрощенная схема которой показана на рисунке 7.

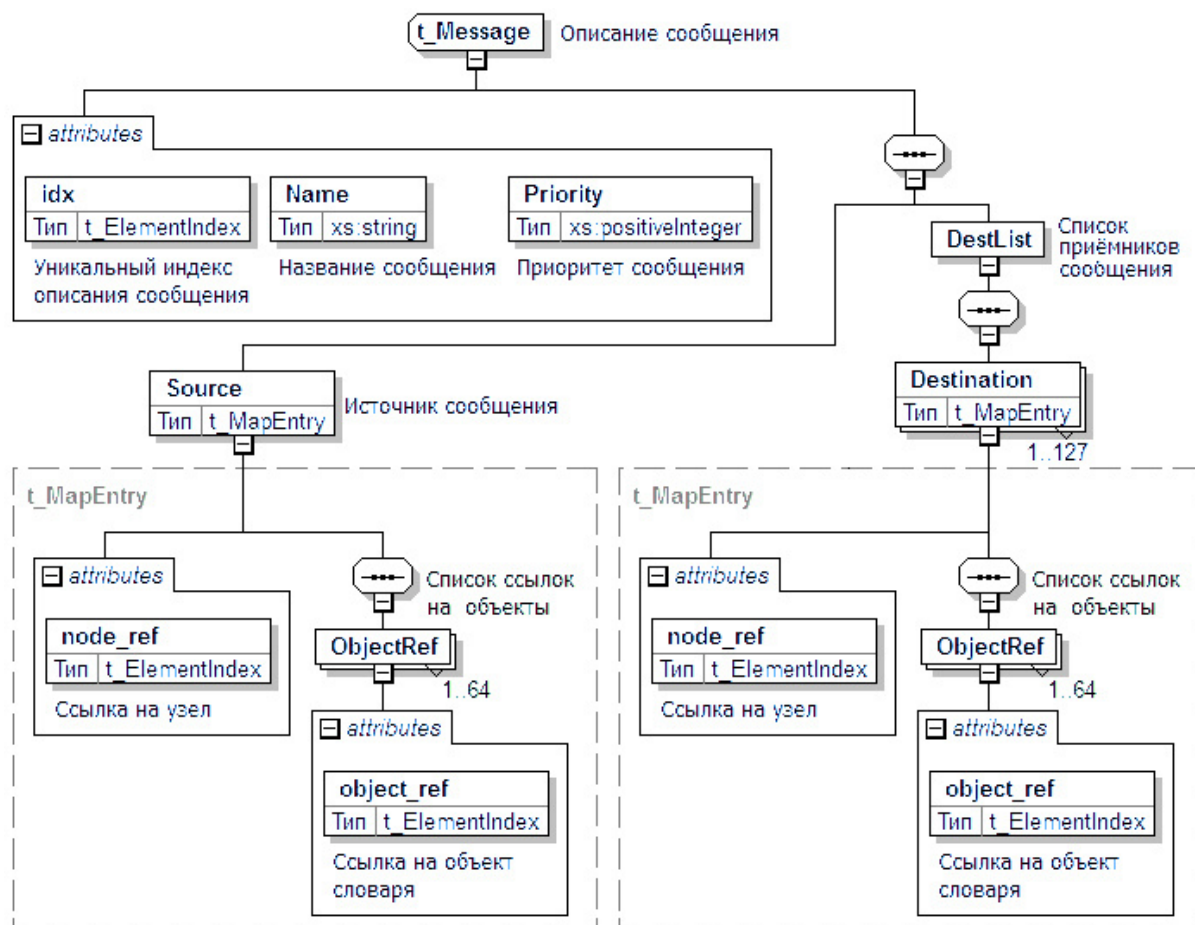


Рисунок 7. Описание сообщения

Figure 7. Message descriptions

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

Источник *Source* и список приемников *DestList* сообщения описываются одинаковыми структурами типа $t_MapEntry$. Различие заключается лишь в том, что любое сообщение всегда имеет только один источник, а приемников 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 $f_{nr}^{(l)}: X_k^{(l)} \rightarrow Y_{nr}$ задается порядком перечисления индексов объектов в списках *ObjectRef* источника и приемников.

Некоторые параметры объектов словаря, оговоренные в CiA 301. CANopen application layer and communication profile. Version: 4.2.0 – 2011 (см. постр. сноску 4), но не относящиеся непосредственно к разработанному методу – такие, как режимы передачи, значения по умолчанию и ряд других – в приведенном описании для простоты не упомянуты, однако при практической реализации метода указываются в соответствующих структурах данных.

Реализация метода с использованием языка XML

Разработанный метод описания ИИУС реализован авторами с использованием языка *XML*⁹, что обеспечивает ему следующие преимущества:

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

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

3. Универсальность. Поскольку язык *XML* стандартизирован и широко распространен, для него существует множество средств разработки и поддержки, что позволяет использовать для создания описаний ИИУС не только специальное, но и универсальное ПО (например, JSON and XML Editor¹⁰);

4. Целостность. Структуру *XML*-документа можно описать и впоследствии автоматически проверить на соответствие этому описанию (схеме документа) при помощи стандартных средств.

Для обеспечения последнего преимущества авторами дополнительно разработана схема описания ИИУС на языке *XSD*¹¹, обеспечивающая контроль полноты и непротиворечивости описания каждой конкретной системы. С использованием этой схемы автоматически выявляются 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.

Выводы

Разработанный метод описания ИИУС имеет ряд преимуществ по сравнению со стандартным (см. постр. сноску 6), задающим межмодульное взаимодействие путем создания независимых конфигураций для каждого модуля системы:

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

⁹ Extensible Markup Language (XML) [Электронный ресурс]. 2016. – URL: <https://www.w3.org/XML/> (дата обращения: 10.08.2023).

¹⁰ JSON and XML Editor [Электронный ресурс]. 2021. – URL: <https://www.altova.com/xmlspy-xml-editor> (дата обращения: 10.08.2023).

¹¹ W3C XML Schema Definition Language (XSD) 1.1 Part 1 [Электронный ресурс]. 2012. – URL: <https://www.w3.org/TR/xmlschema11-1/> (дата обращения: 10.08.2023).

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

3. Исключаются ситуации, когда приемники 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 приемников 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 $f^{(l)}_{nr}: X^{(l)}_k \rightarrow Y_{nr}$ проверяются на совместимость передаваемых и принимаемых данных.

6. На основе единого описания ИИУС возможно автоматизированное создание согласованных друг с другом стандартных файлов конфигурации *DCF* (см. построению 6) для отдельных модулей, что при необходимости позволяет использовать для настройки системы ПО сторонних производителей. Возможно решение и обратной задачи: автоматическое формирование *XML*-описания ИИУС на основе имеющихся отдельных *DCF*-файлов с проверкой его полноты и согласованности.

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

Элементы предложенного метода описания ИИУС реализованы в программе конфигурирования и моделирования модульной системы вибромониторинга турбоагрегатов [8]. Опыт применения программы 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 еще на этапе подготовки конфигурации системы в условиях предприятия-изготовителя, а не в процессе пусконаладочных работ, как это было ранее до внедрения разработанного метода.

СПИСОК ЛИТЕРАТУРЫ / REFERENCES

1. Iehira K., Inoue H., Ishida K. Spoofing attack using bus-off attacks against a specific ECU of the CAN bus. *15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. Las Vegas, NV, USA, 2018. P. 1–4. <https://doi.org/10.1109/CCNC.2018.8319180>
2. Liu Yu, Yin Y. Statistical analysis of improved bit-stuffing method based on CAN bus. *Computer science and engineering*. 2018. <https://doi.org/10.12783/dtcse/ceic2018/24562>
3. Alaei R., Moallem P., Bohlooli A. statistical based algorithm for reducing bit stuffing in the controller area networks. *Microelectronics Journal*. 2020;101:104794. <https://doi.org/10.1016/j.mejo.2020.104794>
4. Плотников Д.А. Оценка времени отклика элементов в модульных информационно-измерительных и управляющих системах, использующих интерфейс CAN. *Известия вузов. Северо-Кавказский регион. Технические науки*. 2017;1:13–18. <http://dx.doi.org/10.17213/0321-2653-2017-1-13-18>
5. Плотников Д.А. Вероятностный метод для оценки параметров межмодульного взаимодействия в системе управления защитой турбоагрегата по вибрации. *Глобальная ядерная безопасность*. 2021;11(3):37–51. <https://doi.org/10.26583/gns-2021-03-04>
6. Лачин В.И., Плотников Д.А., Муженко А.С., Дьяченко В.Б., Рарова Н.В. Разработка имитационной модели модульной информационно-измерительной системы на базе интерфейса CAN FD. *Д.А. Безопасность ядерной энергетики: тезисы докладов 19-й Международной научно-практической конференции, 06-07 июня 2023 г. НИЯУ МИФИ*. Волгоград: ВИТИ НИЯУ МИФИ, 2023. С. 133-136. URL: <https://elibrary.ru/item.asp?id=54313273&pff=1>
7. Плотников Д.А., Лачин В.И., Муженко А.С., Рарова Н.В. Оценка времени передачи сообщений в информационно-измерительных и управляющих системах, использующих интерфейс CAN FD. *Известия вузов. Северо-Кавказский регион. Технические науки*. 2023;2:5–10. <http://dx.doi.org/10.17213/1560-3644-2023-2-5-10>

8. Плотников Д.А., Алджиязна В.К.М. Моделирование межмодульного взаимодействия в системах на базе протокола CANopen (CanOpenAnalyzer). – Свид. о гос. рег. прогр. для ЭВМ 2020610500 РФ. Роспатент. № 2019667012; заявлено 19.12.19, зарег. 15.01.20.

ВКЛАД АВТОРОВ:

Плотников Д.А. – разработка теории, планирование и проведение экспериментов, расчетов, выполнение анализа результатов экспериментов, разработка программного обеспечения;

Муженко А.С. – создание рукописи статьи, выполнение анализа результатов экспериментов и расчетов;

Лачин В.И. – выполнение анализа результатов расчетов, руководство научно-исследовательской работой и проверка ее результатов.

ИСТОЧНИКИ ФИНАНСИРОВАНИЯ:

Статья написана без привлечения внешних источников финансирования.

КОНФЛИКТ ИНТЕРЕСОВ:

Авторы заявляют об отсутствии конфликта интересов.

ИНФОРМАЦИЯ ОБ АВТОРАХ:

Дмитрий Александрович Плотников, д-р техн. наук, профессор, кафедра «Автоматика и телемеханика», Южно-Российский государственный политехнический университет (НПИ) им. М.И. Платова, г. Новочеркасск, Ростовская обл., Российская Федерация.

ORCID: <https://orcid.org/0000-0001-7551-1364>

e-mail: dpl68@mail.ru

Александр Сергеевич Муженко, ассистент, кафедра «Автоматика и телемеханика», Южно-Российский государственный политехнический университет (НПИ) им. М.И. Платова, г. Новочеркасск, Ростовская обл., Российская Федерация.

ORCID: <https://orcid.org/0009-0008-3620-9945>

e-mail: muzhenko97@mail.ru

Вячеслав Иванович Лачин, д-р техн. наук, профессор, кафедра «Автоматика и телемеханика», Южно-Российский государственный политехнический университет (НПИ) им. М.И. Платова, г. Новочеркасск, Ростовская обл., Российская Федерация.

ORCID: <https://orcid.org/0000-0002-7337-229X>

e-mail: lachinv@mail.ru

Поступила в редакцию 07.07.2023

После доработки 28.08.2023

Принята к публикации 05.09.2023

AUTHORS' CONTRIBUTION:

Plotnikov D.A. – development of theory, planning and conducting experiments, calculations, analysis of experimental results, software development;

Muzhenko A.S. – creation of the manuscript, analysis of the results of experiments and calculations;

Lachin V.I. – analysis of the results of calculations, management of research work and verification of its results.

FUNDING:

The article was written without external sources of funding.

CONFLICT OF INTEREST:

The author declares no relevant conflicts of interest.

INFORMATION ABOUT THE AUTHORS:

Dmitriy A. Plotnikov, Doctor of Technical Sciences, Professor, Department «Automation and Telemechanics», Platov South-Russia State Polytechnic University (NPI), Novocherkassk, Rostov region, Russian Federation.

ORCID: <https://orcid.org/0000-0001-7551-1364>

e-mail: dpl68@mail.ru

Alexander S. Muzhenko, assistant lecturer, Department «Automation and Telemechanics», Platov South-Russia State Polytechnic University (NPI), Novocherkassk, Rostov region, Russian Federation.

ORCID: <https://orcid.org/0009-0008-3620-9945>

e-mail: muzhenko97@mail.ru

Vyacheslav I. Lachin, Doctor of Technical Sciences, Professor, Department «Automation and Telemechanics», Platov South-Russia State Polytechnic University (NPI), Novocherkassk, Rostov region, Russian Federation.

ORCID: <https://orcid.org/0000-0002-7337-229X>

e-mail: lachinv@mail.ru

Received 07.07.2023

Revision 28.08.2023

Accepted 05.09.2023