ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО 15745-4 — 2012

Системы промышленной автоматизации и интеграция

ПРИКЛАДНАЯ ИНТЕГРАЦИОННАЯ СРЕДА ОТКРЫТЫХ СИСТЕМ

Часть 4

Эталонное описание систем управления на основе стандарта Ethernet

ISO 15745 – 4:2003

Industrial automation systems and integration —

Open systems application integration framework —

Part 4: Reference description for Ethernet-based control systems

(IDT)

Издание официальное

Москва
Стандартинформ
2014

gost.e-ecolog.ru

Предисловие

Цели и принципы стандартизации в Российской Федерации установлены Федеральным законом от 27 декабря 2002 г. № 184-ФЗ «О техническом регулировании», а правила применения национальных стандартов Российской Федерации — ГОСТ Р 1.0 – 2004 «Стандартизация в Российской Федерации. Основные положения»

Сведения о стандарте

1 ПОДГОТОВЛЕН АНО «Международная академия менеджмента и качества бизнеса» на основе собственного аутентичного перевода на русский язык стандарта, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 100 «Стратегический и инновационный менеджмент»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 29 ноября 2012 г. № 1708-ст

4 Настоящий стандарт идентичен международному стандарту ИСО 15745–4:2003 «Системы промышленной автоматизации и интеграция. Прикладная интеграционная среда открытых систем. Часть 4. Эталонное описание систем управления на основе стандарта Ethernet» (ISO 15745-4:2003 «Industrial automation systems and integration — Open systems application integration framework. – Part 4: Reference description for Ethernet-based control systems»)

В настоящий стандарт введено дополнение № 2 к ИСО 15745-4:2003 от 2007.

При применении настоящего стандарта рекомендуется использовать вместо ссылочных международных стандартов соответствующие им национальные стандарты Российской Федерации, сведения о которых приведены в дополнительном приложении ДА

5 Некоторые положения международного стандарта, указанного в пункте 4, могут являться объектами патентных прав. Международная организация по стандартизации (ИСО) не несет ответственности за идентификацию подобных патентных прав

6 Введен взамен ГОСТ Р ИСО 15745-4 – 2010

Правила применения настоящего стандарта установлены в ГОСТ Р 1.0-2012 (раздел 8). Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте национального органа Российской Федерации по стандартизации в сети Интернет (gost.ru).

© Стандартинформ, 2014

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Введение

Прикладная интеграционная среда (AIF) определяет элементы и правила, облегчающие:

– систематическую организацию и представление требований к интеграции прикладных систем с помощью моделей интеграции;

– разработку спецификаций интерфейсов в форме профилей интероперабельности прикладных систем (AIP), что позволяет выбирать как соответствующие ресурсы, так и документацию «встроенных» прикладных систем.

ИСО 15745-1:2003 устанавливает общие элементы и правила описания интегрированных моделей и AIP вместе с их профилями компонентов – профили процессов, профили обмена информацией и профили ресурсов. Содержание комплекса стандартов ИСО 15745 и структурный вид составных частей AIP представлен на рисунке 1 ИСО 15745-1:2003.

Настоящий стандарт расширяет исходную AIF, описанную в ИСО 15745-1:2003, путем определения специальных технологических элементов и правил описания как профилей коммуникационных сетей, так и связанных с коммуникациями аспектов профилей устройств, специфичных для систем управления, основанных на Ethernet[1)], ADS-net[2)], FL-net[®3)] и EtherNet/IP™[4)]. Технология EtherNet/IP™ использует профиль МЭК 61158, который описан в МЭК 61784-1.

В частности, настоящий стандарт описывает специфические для технологии шаблоны профилей для профилей устройств и профилей коммуникационных сетей. В рамках AIP экземпляр профиля устройства или экземпляр профиля коммуникационной сети является частью профиля ресурсов, определенного в ИСО 15745-1. Экземпляры файлов XML профиля устройства и профиля коммуникационной сети включены в экземпляр XML профиля ресурсов с

[1)] Ethernet использован в настоящем стандарте в качестве синонима ИСО/МЭК 8803-3.

[2)] ADS-net является торговой маркой, используемой для описания JIS-TR B0012 (сеть автономной децентрализованной системы). Эта информация предоставлена для удобства пользователей ИСО 15745 и не означает одобрения со стороны ИСО торговой марки или какой-либо продукции. Для соответствия настоящему стандарту не требуется использования торговой марки ADS-net.

[3)] FL-net является торговой маркой для описания JEM 1479. Эта информация предоставлена для удобства пользователей ИСО 15745 и не означает одобрения со стороны ИСО торговой марки или какой-либо продукции. Для соответствия настоящему стандарту не требуется использования торговой марки FL-net.

[4)] EtherNet/IP™ является торговой маркой ControlNet International, Ltd и Ассоциации поставщиков открытой DeviceNet. Эта информация предоставлена для удобства пользователей ИСО 15745 и не означает одобрения со стороны ИСО торговой марки или какой-либо продукции. Для соответствия настоящему стандарту не требуется использования торговой марки ControlNet/IP™. Для применения торговой марки ControlNet/IP™ необходимо получить разрешение либо ControlNet International, Ltd, или Ассоциации поставщиков открытой DeviceNet .

использованием ProfileHandle_DataType согласно определениям, приведенным в ИСО 15745-1:2003, пункт 7.2.5.

Установленные среды (AIF), использующие элементы и правила по ИСО 15745-1, могут быть легко интегрированы с профилями компонентов, определенных с использованием элементов и правил, установленных в настоящем стандарте.

Настоящий стандарт подготовлен техническим комитетом ИСО/ТК 184 «Системы промышленной автоматизации и интеграция», подкомитетом ПК 5 «Архитектура, коммуникации и структуры интеграции».

Комплекс стандартов ИСО 15745 состоит из следующих частей, имеющих общее название «Системы промышленной автоматизации и интеграция. Прикладная среда интегрирования открытых систем»:

- часть 1. Общее эталонное описание;
- часть 2. Эталонное описание систем управления на основе ИСО 11898;
- часть 3. Эталонное описание систем управления на основе МЭК 61158;
- часть 4. Эталонное описание систем управления на основе Ethernet.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Системы промышленной автоматизации и интеграция ПРИКЛАДНАЯ ИНТЕГРАЦИОННАЯ СРЕДА ОТКРЫТЫХ СИСТЕМ Часть 4 Эталонное описание систем управления на основе стандарта Ethernet

Industrial automation systems and integration. Open systems application integration framework. Part 4. Reference description for Ethernet-based control systems

Дата введения – 2014 – 01 – 01

1 Область применения

Настоящий стандарт распространяется на описание технологических спецификаций для элементов и правил как профилей коммуникационной сети, так и связанных с коммуникациями аспектов профилей устройств, относящихся к системам управления на основе Ethernet.

Примечание – Общие элементы и правила описания моделей интеграции и профилей интероперабельности приложений с профилями их компонентов (профили процессов, профили обмена информацией и профили ресурсов) установлены в ИСО 15745-1.

При описании среды интеграции приложений настоящий стандарт следует использовать совместно с ИСО 15745-1.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты, которые необходимо учитывать при использовании настоящего стандарта. В случае ссылок на документы, у которых указана дата утверждения, необходимо пользоваться только указанной редакцией. В случае, когда дата утверждения не приведена, следует пользоваться последней редакцией ссылочных документов, включая любые поправки и изменения к ним:

ИСО 639-2 Коды для представления названий языков. Часть 2. Трехбуквенный код (ISO 639-2, Codes for the representation of names of languages — Part 2: Alpha-3 code)

ИСО 15745-1 Системы промышленной автоматизации и интеграция. Прикладная интегриционная среда открытых систем. Часть 1. Общее эталонное описание (ISO 15745-1, Industrial automation systems and integration – Open systems application integration framework – Part 1: Generic reference description)

ИСО/МЭК 7498-4 Системы обработки информации. Взаимодействие открытых систем. Базовая эталонная модель. Часть 4. Структура управления (ISO/IEC 7498-4, Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 4: Management framework)

ИСО/МЭК 8802-3 Информационные технологии. Телекоммуникации и информационный обмен между системами. Локальные и общегородские сети. Специальные требования. Часть 3. Метод доступа (CSMA/CD) с обнаружением столкновений и спецификации физического уровня (ISO/IEC 8802-3, Information technology – Telecommunications and information exchange between systems– Local and metropolitan area networks – Specific requirements – Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications)

ИСО/МЭК 10646-1 Информационные технологии. Универсальный многооктетный набор кодированных знаков (UCS). Часть 1. Архитектура и основная многоязычная матрица (ISO/IEC 10646-1, Information technology — Universal Coded Character Set (UCS))

МЭК 61158 (все части) Передача цифровых данных для измерения и управления. Полевая шина для систем автоматического регулирования и управления технологическими процессами (IEC 61158 (all parts)), Digital data communications for measurement and control – Fieldbus for use in industrial control systems)

МЭК 61784-1 Цифровые передачи данных для измерения и управления. Часть 1. Профильные наборы для непрерывного и раздельного производства полевых шин для систем автоматического регулирования и управления

технологическими процессами (IEC 61784-1, Digital data communications for measurement and control – Part 1: Profile sets for continuous and discrete manufacturing relative to fieldbus use in industrial control systems)

IEEE Std 754-1985 (R1990) IEEE Standard for Binary Floating Point Arithmetic

JEM 1479 Protocol specification – Factory automation control Link Network (FL-net)

JIS-TR B0012:2000 Autonomous Decentralized System Network (ADS-net)

REC-xml-20001006 Extensible Markup Language (XML) 1.0 Second Edition — W3C Recommendation 6 October 2000

REC-xmlschema-1-20010502 XML Schema Part 1: Structures — W3C Recommendation 02 May 2001

REC-xmlschema-2-20010502 XML Schema Part 2: Datatypes - W3C Recommendation 02 May 2001

RFC 768 User Diagram Protocol — Internet Engineering Task Force (IETF), request for Comments (RFC)

RFC 791 Internet Pritocol — Internet Engineering Task Force (IETF), request for Comments (RFC)

RFC 793 Transmission Control Pritocol — Internet Engineering Task Force (IETF), request for Comments (RFC)

RFC 894 A standard for the Transmission of IP Datagrams over Ethernet Network — Internet Engineering Task Force (IETF), request for Comments (RFC)

RFC 1738 Uniform Resource Locators (URL) - Internet Engineering Task Force (IETF), Request for Comments (RFC)

RFC 1759:1995 Printer MIB - Internet Engineering Task Force (IETF), Request for Comments (RFC)

ИСО 639-1 Коды для представления названий языков. Часть 1. Двухбуквенный код (ISO 639-1, Codes for the representation of names of languages — Part 1: Alpha-2 code)

ИСО 15745-3 Системы промышленной автоматизации и интеграция. Прикладная интеграционная среда открытых систем. Часть 3. Эталонное описание систем управления на основе МЭК 61158 (ISO 15745-3, Industrial

automation systems and integration – Open systems application integration framework – Part 3: Reference description for IEC 61158-based control systems)

ИСО/МЭК 11578 Информационные технологии. Взаимосвязь открытых систем. Вызов удаленных процедур (ISO/IEC 11578, Information technology — Open Systems Interconnection — Remote Procedure Call (RPC))

REC-svg-20030114 Scalable Vector Graphics (SVG) 1.1 Specification - W3C Recommendation 14 January 2003, available at gost.e-ecolog.ru

REC-xpath-1999111 XML Path Language (XPath) Version 1.0- W3C Recommendation 16 November 1999

RFC 1101 DNS encoding of network names and other types — Internet Engineering Task Force (IETF), Request for Comments (RFC)

RFC 2131:1997 Dynamic Host Configuration Protocol — Internet Engineering Task Force (IETF), Request for Comments (RFC)

UML V1.4, OMG — Unified Modeling Language Specification (Version 1.4, September 2001)

ИСО 3166-1 Коды для представления названий стран и единиц их административно-территориального деления. Часть 1. Коды стран (ISO 3166-1, Codes for the representation of names of countries and their subdivisions - Part 1: Country codes)

ИСО 80000-1 Величины и единицы. Часть 1. Общие положения (ISO 80000-1, Quantities and units — Part 1: General)

МЭК/ПАС 62030 Передача цифровых данных измерений и контроля. Полевая шина для промышленных систем управления. Раздел 1. Спецификация V1.1a протокола применения MODBUS®. Раздел 2. Проводная версия 1.0 спецификации протокола общественной подписки в реальном масштабе времени (IEC/PAS 62030, Digital data communications for measurement and control — Fieldbus for use in industrial control systems — Section 1: MODBUS® Application Protocol Specification V1.1a — Section 2: Real-Time Publish-Subscribe (RTPS) Wire Protocol Specification Version 1.0)

МЭК/ПАС 62407 Технология автоматизации управления Ethernet работающей в реальном масштабе времени (EtherCATTM) (IEC/PAS 62407 Real-time Ethernet control automation technology (EtherCATTM))

МЭК/ПАС 62408 Ethernet Powerlink (EPL) в реальном масштабе времени (EtherCATTM) (IEC/PAS 62408, Real-time Ethernet Powerlink (EPL))

RFC 1157 SNMP, Simple Network Management Protocol (SNMP) Management Frameworks.

3 Термины и определения

В настоящем стандарте применены термины по ИСО 15745-1.

Примечание – Терминология UML и обозначения, использованные в настоящем стандарте, описаны в ИСО 15745-1:2003 (приложение А).

4 Сокращения и обозначения

AIF – прикладная интеграционная среда (Application Integration Framework);

AIP – профиль интероперабельности приложений (Application Interoperability Profile);

ASCII – американский стандартный код для обмена информацией (American Standard Code for Information Interchange);

ASE – элемент сервиса приложения (Application Service Element);

CIP™[5)] – общий промышленный протокол (Common Industrial Protocol);

DDXML – расширяемый язык разметки для описания устройств (Duct Design Extensible Markup Language);

EDS – электронный бланк данных (Electronic Data Sheet);

EPL – ETHERNET Powerlink;

[5)] CIP™ является торговой маркой ControlNet International, Ltd. и Open DeviceNet Vendor Association, Inc. Эта информация предоставлена для удобства пользователей ИСО 15745 и не означает подтверждения со стороны ИСО держателя торговой марки или какой-либо из его продукции. Для соответствия настоящему стандарту не требуется использования торговой марки CIP™. Применение торговой марки CIP™ требует разрешения ControlNet International, Ltd. и Open DeviceNet Vendor Association, Inc.

FMMU – блок управления памятью полевой шины (Fieldbus Memory Management Unit);

IAS – системы промышленной автоматизации (Industrial Automation Systems);

ID – идентификатор (Identifier);

IP – Интернет-протокол (см. RFC 791 и RFC 894) (Internet Protocol);

MAC – контроль доступа к носителю информации (Media Access Control);

MCG – группа многоадресной передачи (MultiCast Group);

MIB – база управляющей информации (Management Information Base);

OSI – взаимодействие открытых систем (Open System Interconnection);

SNMP – простой протокол сетевого управления (RFC 1157)

TCD – код транзакции (Transaction CoDe);

TCP – протокол контроля передачи (см. RFC 793) (Transmission Control Protocol);

UDP – протокол датаграммы пользователя (см. RFC 768) (User Datagram Protocol);

UML – унифицированный язык моделирования (см. UML V1.4) (Unified Modeling Language);

XML – расширяемый язык разметки (см. REC-xml-20001006) (eXtensible Markup Language);

DAP – точка доступа к устройству (Device Access Point);

DNS – система имен доменов (Domain Name System (см. RFC 1101));

GSD – общее описание станции (Generic Station Description);

GSDML – язык разметки для общего описания станции (Generic Station Description Markup Language);

ID – идентификация (Identification);

IO – ввод/вывод (Input/Output);

PDU – блок данных протокола (Protocol Data Unit);

SVG – масштабируемая векторная графика (Scalable Vector Graphics (см. REC-svg-20030114));

URL – унифицированный указатель ресурса (Uniform Resource Locator);

UUID – универсальный уникальный идентификатор (Universally Unique Identifier (см. ИСО/МЭК 11578));

W3C – Консорциум всемирной сети (World Wide Web Consortium).

5 Элементы и правила, специфические для технологии

5.1 Модели интеграции и интерфейсы IAS

Разработчик AIP должен подготовить модель интеграции, использующую описанные в ИСО 15745-1 правила, а также гарантировать, что основанные на Ethernet профили устройств и коммуникационных сетей (как соответствующие требованиям интерфейса, так и полученные на основе существующих устройств/коммуникационных сетей) включают в себя необходимые интерфейсы IAS. Интерфейсы IAS, включенные в профиль, должны быть указаны в разделе заголовков (см. ИСО 15745-1:2003, пункт 7.2.2).

Примечание – Интерфесы IAS описаны в ИСО 15745-1:2003, приложение B.

5.2 Шаблоны профилей

5.2.1 Общие положения

Специфические для основанной на Ethernet технологии шаблоны профилей разрабатывают исходя из общих шаблонов профилей,

установленных в ИСО 15745-1:2003, раздел 7.

5.2.2 Содержание и синтаксис

Комплекс стандартов ИСО 15745 устанавливает шаблоны профилей, представляющие собой XML схемы (REC-xmlschema-1-20010502 и REC-xmlschema-2-20010502) и использующие общую родовую структуру. Профили устройств и коммуникационных сетей, основанные на этих шаблонах, как правило, включают в себя следующее:

– информацию, необходимую для идентификации соединяемого устройства;

– описание данных устройства, которые доступны в сети;

– описание коммуникационных возможностей, поддерживаемых устройством;

– дополнительную относящуюся к поставщику информацию.

Однако в некоторых технологиях на основе Ethernet используют ранее принятый специальный синтаксис ASCII. В связи с этим для достижения обратной совместимости определений шаблонов в любых технологиях (см. приложения A - G) используют все или следующие необходимые подмножества:

– шаблонов профилей коммуникационных сетей и устройств согласно определениям, приведенным в ИСО 15745-1;

– шаблона ИСО 15745 для инкапсуляции файлов, содержащих ранее принятый синтаксис ASCII («оболочку»);

– ранее принятого синтаксиса ASCII.

5.2.3 Заголовок

Заголовок шаблона профиля, определенный в ИСО 15745-1:2003, пункт 7.2.2, используют для специфических для технологии Ethernet шаблонов профилей. Каждая технология использует одно или более имен для идентификации технологии или ее частных компонентов (таблица 1). Выбранное имя должно сохраняться в атрибуте ProfileTechnology в разделе заголовка.

Таблица 1 — Имена ProfileTechnology

Имя ProfileTechnology	Технология
ADS-Net	ADS-Net
FL-net	FL-net
EtherNet/IP	EtherNet/IP
CIP	EtherNet/IP
EDS	EtherNet/IP
GSDML	PROFINET
DDXML	Modbus TCP
EtherCAT	EtherCAT
EPL	ETHERNET Powerlink

5.3 Специфические профили

Структура специфического для технологии профиля коммуникационной сети и коммуникационные аспекты структуры профиля устройства, основанные на технологиях Ethernet, описаны в разделе 6. Рассматриваемые технологии включают в себя:

- ADS-net (см. 6.1);
- FL-net (см. 6.2);
- EtherNet/IP (см. 6.3);
- PROFINET (см. 6.4).
- Modbus TCP (см. 6.5);
- EtherCAT (см. 6.6);
- ETHERNET Powerlink (см. 6.7).

Соответствующие определения шаблонов профилей приведены в приложениях A – G.

6 Профили устройств и коммуникационных сетей в системах управления, основанных на Ethernet

6.1 ADS-net

6.1.1 Профиль устройства

6.1.1.1 Общие положения

На рисунке 1 показана диаграмма класса профилей устройства ADS-net.

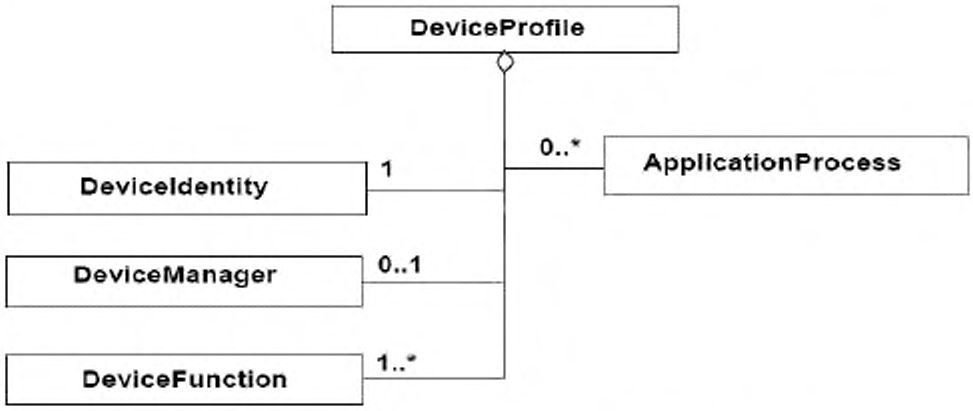


Рисунок 1 — Диаграмма класса профилей устройства ADS-net

Имеющиеся форматы профилей устройств ADS-net описаны в разделе А.2 приложения А.

XML схема, представляющая шаблон профиля устройства ADS-net, определена в А.2.3. Имя файла этой XML схемы — «ADS-net_Device_Profile.xsd».

Примечание — Диаграмма класса профиль устройства ADS-net, представленная на рисунке 1, определяет основные классы. Затем эти классы должны быть представлены в развернутом виде согласно приложению А.

XML схема, представляющая шаблон профиля устройства ADS-net, определена в А.2.

6.1.1.2 DeviceIdentity (идентификация устройства)

Класс DeviceIdentity содержит атрибуты, которые уникальным образом идентифицируют устройство, и поддерживает сервисы, позволяющие извлечь эту информацию из устройства.

Эти атрибуты предоставляют следующие данные:

– идентификацию продавца (VendorName);

– идентификацию устройства (код продукта, версия продукта, имя устройства).

6.1.1.3 DeviceManager (менеджер устройства)

Класс менеджер устройства содержит атрибуты и поддерживает сервисы, используемые для контроля и конфигурирования устройства.

Эти атрибуты предоставляют следующую информацию:

– статус устройства (DeviceState).

6.1.1.4 DeviceFunction (функция устройства)

Класс DeviceFunction содержит атрибуты и поддерживает сервисы, позволяющие управлять функциями устройства, например его конфигурацией.

Эти атрибуты предоставляют следующую информацию:

– номер поля данных (DFNO):

– номер логического узла (LNODENO);

– режим (MODE).

6.1.1.5 ApplicationProcess (прикладной процесс)

Класс ApplicationProcess содержит атрибуты и поддерживает сервисы, позволяющие осуществлять контроль коммуникаций между программами приложений, выполняемых в ADS-net.

Эти атрибуты предоставляют следующеую информацию:

– список TCD производителей (Producer-TCD-List);

– список TCD потребителей (Consumer-TCD-List);

– приоритет (Priority).

6.1.2 Профиль коммуникационной сети

6.1.2.1 Общие положения

На рисунке 2 представлена диаграмма класса профилей коммуникационной сети ADS-net.

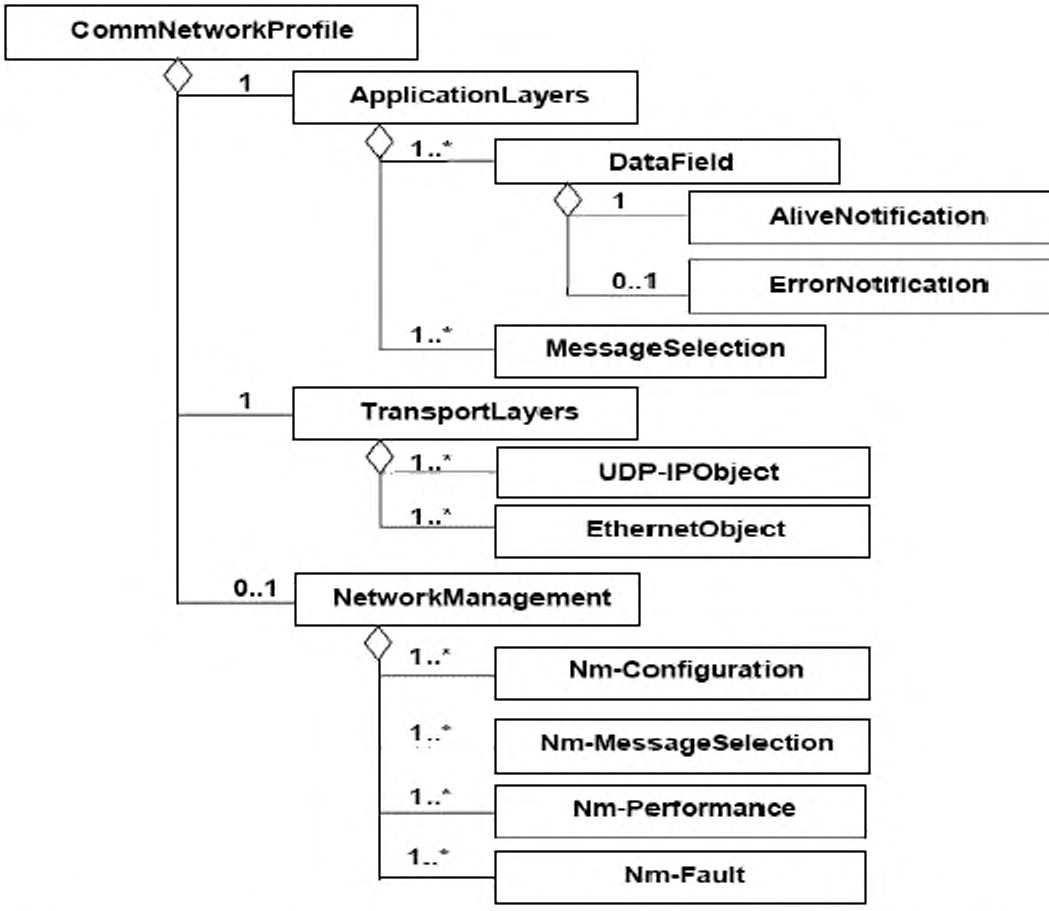


Рисунок 2 — Диаграмма класса профилей коммуникационной сети ADS-net

Существующие форматы профилей коммуникационной сети ADS-net описаны в А.3 приложения А.

XML схема, представляющая шаблон профиля коммуникационной сети ADS-net, определена в А.3.3 приложения А. Имя файла этой XML схемы — «ADS-net_CommNet_Profile.xsd».

6.1.2.2 ApplicationLayers (прикладные уровни)

6.1.2.2.1 Общие положения

Класс ApplicationLayers ADS-net представляет комбинированные профили трех верхних уровней OSI модели интеграции коммуникационной сети ADS-net. Он устанавливает поддерживаемые элементы сервиса приложения и их ассоциированные сервисы.

На рисунке 3 представлена диаграмма класса ApplicationLayers ADS-net.

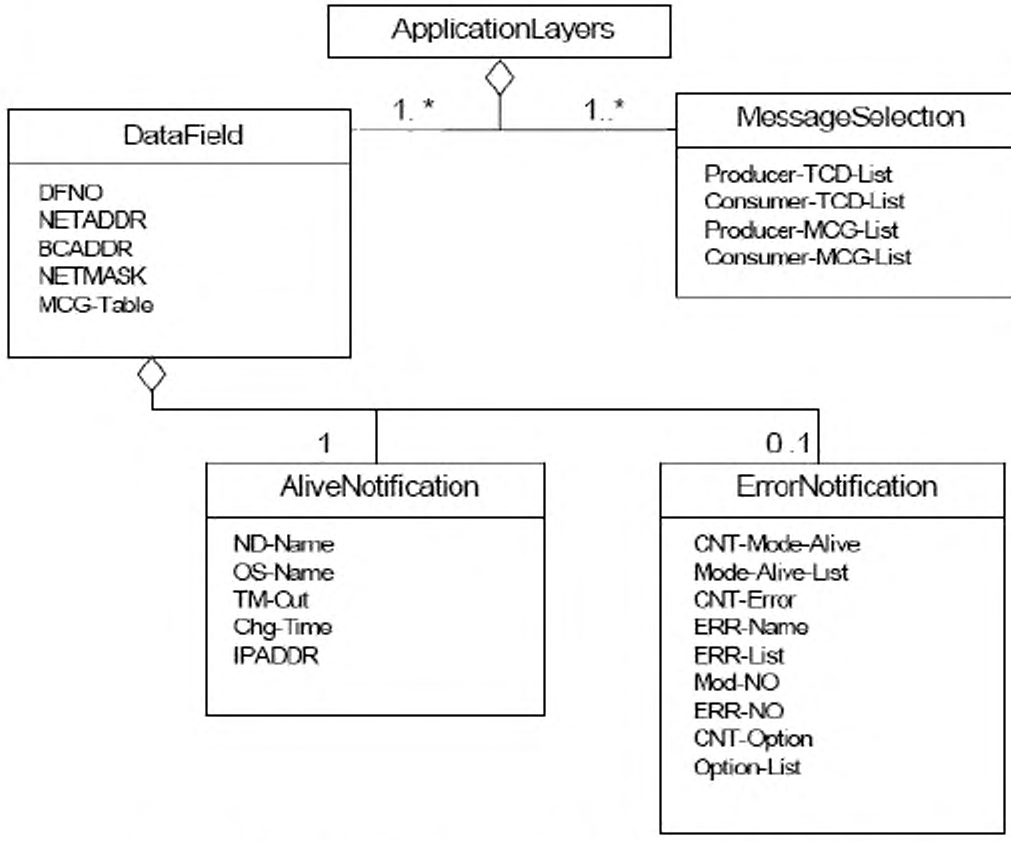


Рисунок 3 — Диаграмма класса ApplicationLayers ADS-net

6.1.2.2.2 DataField (поле данных)

6.1.2.2.2.1 Общие положения

ADS-net определяет домен с именем «Data Field», в котором подсистемы разделяют информацию при пересылке сообщений между равноправными узлами. Каждый узловой компьютер передает сообщения на многоадресной основе в поле данных, к которому он относится; другие узловые компьютеры, принадлежащие к тому же полю данных, могут получать эти данные автономно. Многие компьютеры, относящиеся к какому-либо полю данных, посылают или принимают данные. Многоадресная группа (MCG), т. е. группа узловых компьютеров, относящихся к определенному полю данных, вводится в целях локализации многоадресной передачи.

DataField определяет характеристики, используемые для контроля поля данных. Атрибуты DataField предоставляют, в частности, следующую информацию:

- номер поля данных (DFNO);
- таблицу многоадресной группы (MCG-Table).

6.1.2.2.2.2 AliveNotification (уведомление о рабочем состоянии)

На поле данных периодически передается сообщение «keep alive» (в рабочем состоянии) для уведомления других узловых компьютеров о статусе узлового компьютера-отправителя.

AliveNotification определяет параметры, используемые для проверки статуса узловых компьютеров. Атрибуты AliveNotification предоставляют, в частности, следующую информацию:

- имя узла (ND-Name);
- перерыв работы (TM-Out).

6.1.2.2.2.3 ErrorNotification (уведомление об ошибке)

Когда на узловом компьютере происходит отказ, информация об отказе включается в сообщение «keep alive», передаваемое в поле данных во время цикла передачи сообщения «keep alive». Любой узловой компьютер, относящийся к этому же полю данных, может обнаружить состояние отказа (ошибки).

ErrorNotification определяет параметры, описывающие информацию об отказе. Атрибуты ErrorNotification предоставляют, в частности, следующую информацию:

- имя ошибки (ERR-Name);
- список ошибок (ERR-List).

6.1.2.2.3 MessageSelection (выбор сообщения)

Код транзакции TCD представляет собой идентификатор сообщения, определенный уникальным образом в поле данных. Передающее устройство посылает сообщение с присвоенным TCD на указанное поле данных на многоадресной основе, а каждый узловой компьютер, относящийся к этому

полю данных, автономно выбирает только подходящие сообщения на основе TCD.

MessageSelection определяет параметры, используемые для обмена сообщениями. Атрибуты MessageSelection предоставляют, в частности, следующую информацию:

- список TCD производителей (Producer-TCD-List);
- список TCD потребителей (Consumer-TCD-List);
- список MCG производителей (Producer-MCG-List);
- список MCG потребителей (Consumer-MCG-List).

6.1.2.3 TransportLayers (транспортные уровни)

6.1.2.3.1 Общие положения

Класс ADS-net TransportLayers представляет комбинированные профили для нижних четырех уровней OSI модели интеграции коммуникационной сети. Класс TransportLayers подразделяется на один или более объектов, основанных на Ethernet и объект UDP/IP.

6.1.2.3.2 EthernetObject (объект Ethernet)

EthernetObject определяет параметры Ethernet, используемые для реализации ADS-net. Атрибуты EthernetObject предоставляют следующую информацию:

- тип носителя информации (MediaType);
- скорость коммуникаций (CommRate);
- индикаторы (Indicators);
- адрес MAC (MACAddress);
- журнал ошибок (ErrorLog).

6.1.2.3.3 Объект UDP-IP (UDP-IPObject)

UDP-IPObject определяет параметры UDP/IP, используемые для реализации ADS-net. Атрибуты UDP-IPObject предоставляют следующую информацию:

- адрес IP (IPADDR);
- информацию о многоадресной группе (UDP-IP-MCGs);

– имя хоста (HostName);

– журнал ошибок (ErrorLog).

6.1.2.4 NetworkManagement (управление сетью)

6.1.2.4.1 Общие положения

Класс ADS-net NetworkManagement представляет конфигурацию сети и возможности регулировки характеристик модели интеграции коммуникационной сети ADS-net.

Далее этот класс подразделяется на несколько классов согласно рисунку 2.

6.1.2.4.2 Nm-Configuration (Nm-Конфигурация)

6.1.2.4.2.1 Общие положения

Nm-Configuration определяет параметры конфигурации сети, относящейся к ADS-net. Атрибуты Nm-Configuration предоставляют, в частности, следующую информацию:

– список номеров активных полей данных (ActiveDataFieldNoList);

– список номеров активных узлов (ActiveNodeNoList);

– список номеров активных многоадресных групп (ActiveMulticastGroupNoList).

6.1.2.4.2.2 Nm-MessageSelection (Выбор Nm-сообщения)

Nm-MessageSelection определяет параметры, используемые для управления выбором сообщений. Атрибуты Nm-MessageSelection предоставляют, в частности, следующую информацию:

– поддерживаемый список TCD активных производителей (ActiveProducerTCDSupportedList);

– поддерживаемый список TCD активных потребителей (ActiveConsumerTCDSupportedList).

6.1.2.4.2.3 Nm-Performance (Nm-характеристики)

Nm-Performance определяет параметры, используемые для мониторинга характеристик. Атрибуты Nm-Performance предоставляют следующую информацию:

– число сообщений в процессе обмена (MessageTransaction).

6.1.2.4.3 Nm-Fault (Nm-Отказ)

Nm-Fault определяет параметры, используемые для мониторинга отказов. Атрибуты Nm-Fault предоставляют, в частности, следующую информацию:

– список аварийных узлов (FaultNodeList).

6.2 FL-net

6.2.1 Профиль устройства

6.2.1.1 Общие положения

На рисунке 4 представлена диаграмма класса профилей устройства FL-net.

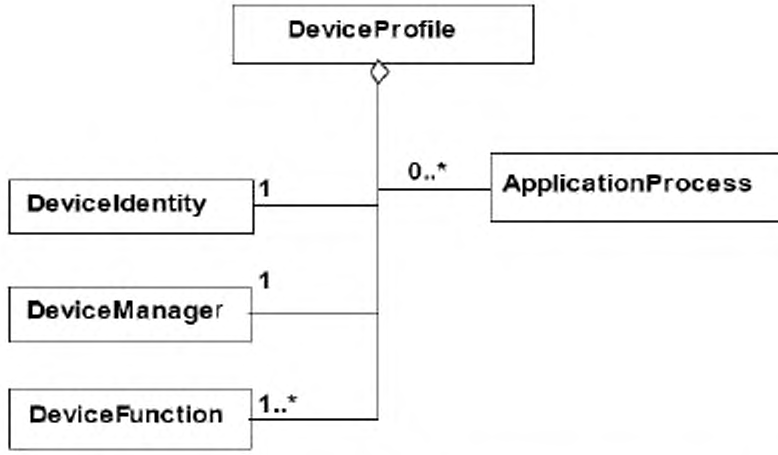


Рисунок 4 — Диаграмма класса профиля устройства FL-net

Существующие форматы профилей устройства FL-net описаны в B.1 приложения B.

XML схема, представляющая шаблон профиля устройства FL-net, определена в B.1.5 приложения B. Имя файла этой XML схемы — «FL-net_Device_Profile.xsd».

Примечание 1 – Диаграмма класса профиля устройства FL-net, представленная на рисунке 4, определяет основные классы. Затем некоторые классы должны быть представлены в развернутом виде согласно приложению В.

Примечание 2 – Эти классы представлены на одной XML схеме, определенной в В.1.5 приложения В.

6.2.1.2 DeviceIdentity (идентификация устройства)

Класс DeviceIdentity определен на рисунке 5, а подклассы приведены в таблице 2.

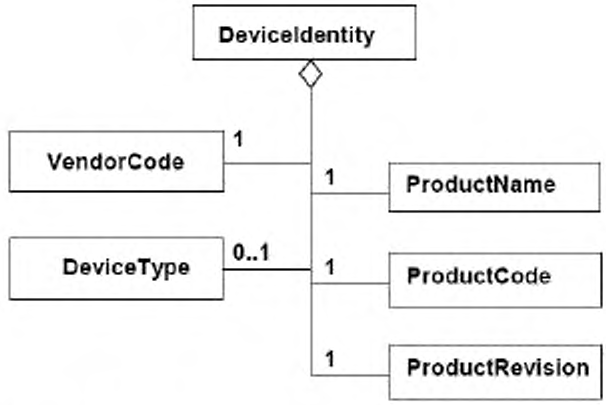


Рисунок 5 — Диаграмма класса FL-net DeviceIdentity

Таблица 2 — Элементы объекта DeviceIdentity

Элементы класса	Описание
VendorCode (код продавца)	Код изготовителя или продавца устройства
ProductName (имя продукта)	Специальное имя продукта продавца
ProductCode (код продукта)	Уникальный идентификатор ID, идентифицирующий продукт. Формат определяется продавцом
ProductRevision (версия продукта)	Номер версии продукта продавца
DeviceType (тип продукта)	Конкретный тип устройства FL-net (см. приложение В, таблица В.1)

6.2.1.3 DeviceManager (менеджер устройства)

6.2.1.3.1 Общие положения

На рисунке 6 представлена структура класса объекта DeviceManager.

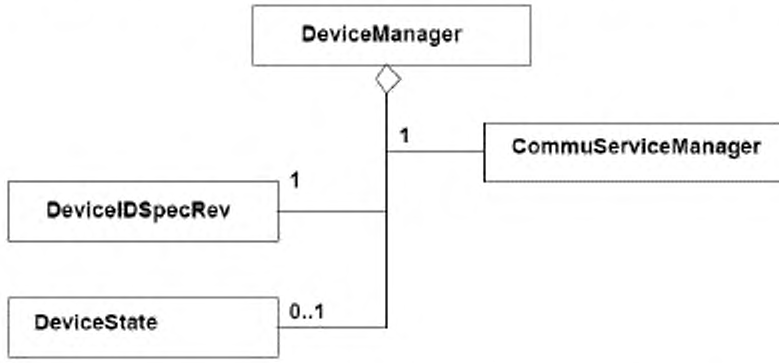


Рисунок 6 — Диаграмма класса FL-net DeviceManager

6.2.1.3.2 DeviceIDSpecRev (версия спецификации идентификатора устройства)

Объект DeviceIDSpecRev должен описывать версию объекта идентификатора FL-net.

6.2.1.3.3 CommuServiceManager (менеджер коммуникационных услуг)

Объект CommuServiceManager должен описывать сервис коммуникаций, который несет ответственность за мониторинг и конфигурацию.

6.2.1.3.4 DeviceState (состояние устройства)

Объект DeviceState должен описывать состояния устройства.

6.2.1.4 DeviceFunction (функция устройства)

Объект DeviceFunction содержит атрибуты и поддерживает сервисы, обеспечивающие менеджмент (например, конфигурацию) функций устройства.

Примечание – Настоящий стандарт не распространяется на класс функций устройства.

6.2.1.5 ApplicationProcess (прикладной процесс)

Объект ApplicationProcess содержит атрибуты и поддерживает сервисы, соответствующие требованиям приложения.

Эти атрибуты предоставляют, в частности, следующую информацию:

– название завода (PlantName).

Для создания конкретного представления процесса приложения могут быть определены дополнительные подклассы и дополнительные атрибуты, описывающие прикладной процесс.

6.2.2 Профиль коммуникационной сети

6.2.2.1 Общие положения

На рисунке 7 представлена диаграмма профилей коммуникации сети FL-net.

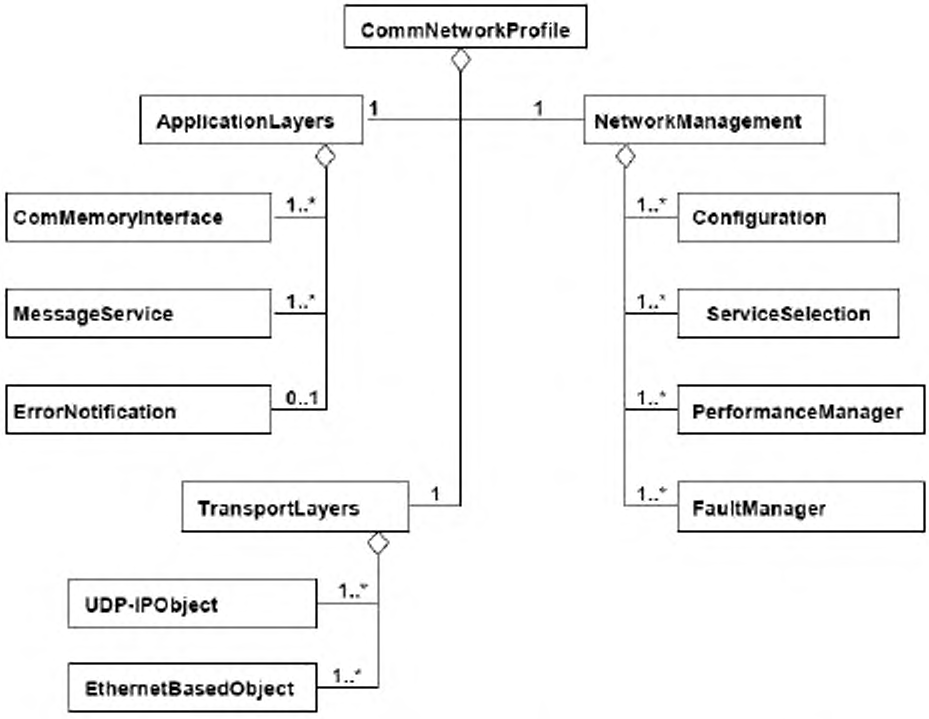


Рисунок 7 — Диаграмма класса профилей коммуникационной сетиFL-net

Существующие форматы профилей коммуникационной сети FL-net описаны в B.2.

XML схема, представляющая шаблон профилей коммуникационной сети FL-net, определена в B.2.4.5 приложения B. Имя файла этой XML схемы — «FL-net_CommNet_Profile.xsd».

6.2.2.2 ApplicationLayers (прикладные уровни)

6.2.2.2.1 Общие положения

Класс FL-net ApplicationLayers представляет комбинированные профили трех верхних уровней OSI модели интеграции коммуникационной сети FL-net. Он устанавливает поддерживаемые элементы сервиса приложения и ассоциированные с ними сервисы.

Далее этот класс подразделяется на несколько классов согласно рисунку 7.

Примечание – Полное определение объекта ApplicationLayers приведено в JEM 1479:2002.

6.2.2.2.2 ComMemoryInterface (интерфейс коммуникативной памяти)

Объект ComMemoryInterface определяет характеристики, связанные с общим интерфейсом памяти. Элементы объекта ComMemoryInterface определены в B.2.2.1 приложения B.

6.2.2.2.3 MessageService (сервис сообщений)

Объект MessageService определяет характеристики, ассоциированные с сервисами сообщений устройства. Элементы объекта MessageService определены в B.2.2.2 приложения B.

6.2.2.2.4 ErrorNotification (уведомление об ошибке)

Объект ErrorNotification определяет характеристики, ассоциированные с видами ошибок, относящимися к сети и устройству. Элементы объекта ErrorNotification определены в B.2.2.3 приложения B.

6.2.2.3 TransportLayers (транспортные уровни)

6.2.2.3.1 Общие положения

Класс FL-net TransportLayers представляет комбинированные профили четырех нижних уровней OSI модели интеграции коммуникационной сети FL-net.

Этот класс далее подразделяется на несколько классов согласно рисунку 7.

6.2.2.3.2 EthernetBasedObject (объект на основе Ethernet)

Объект EthernetBasedObject определяет характеристики, связанные с физическим уровнем FL-net. Элементы объекта EthernetBasedObject определены в B.2.3.1 приложения B.

6.2.2.3.3 UDP-IPObject (объект UDP-IP)

Объект UDP-IPObject определяет характеристики, связанные с конфигурацией и мониторингом канала передачи данных. Элементы объекта UDP-IPObject определены в B.2.3.2 приложения B.

6.2.2.4 NetworkManagement (управление сетью)

6.2.2.4.1 Общие положения

Класс FL-net NetworkManagement представляет средства наладки характеристик и конфигурации сети в модели интеграции коммуникационной сети FL-net.

Далее этот класс подразделяется на несколько классов согласно рисунку 7.

Примечание – Полное определение объекта NetworkManagement приведено в JEM 1479:2002.

6.2.2.4.2 Configuration (конфигурация)

Объект Configuration определяет характеристики, связанные с первоначальной установкой и модификацией конфигурации. Элементы объекта Configuration определены в B.2.4.1 приложения B.

6.2.2.4.3 ServiceSelection (выбор сервиса)

Объект ServiceSelection определяет характеристики, связанные с сервисами коммуникационной сети. Элементы объекта ServiceSelection определены в B.2.4.2 приложения B.

6.2.2.4.4 PerformanceManager (менеджер исполнения)

Объект PerformanceManager определяет параметры, связанные с характеристиками обмена данными в сети. Элементы объекта PerformanceManager определены в B.2.4.3 приложения B.

6.2.2.4.5 FaultManager (менеджер отказов)

Объект FaultManager определяет характеристики, связанные с возможностями отказов в FL-net. Элементы объекта FaultManager определены в В.2.4.4 приложения В.

6.3 EtherNet/IP

6.3.1 Профиль устройства

6.3.1.1 Общие положения

На рисунке 8 представлена диаграмма класса профиля устройства EtherNet/IP.

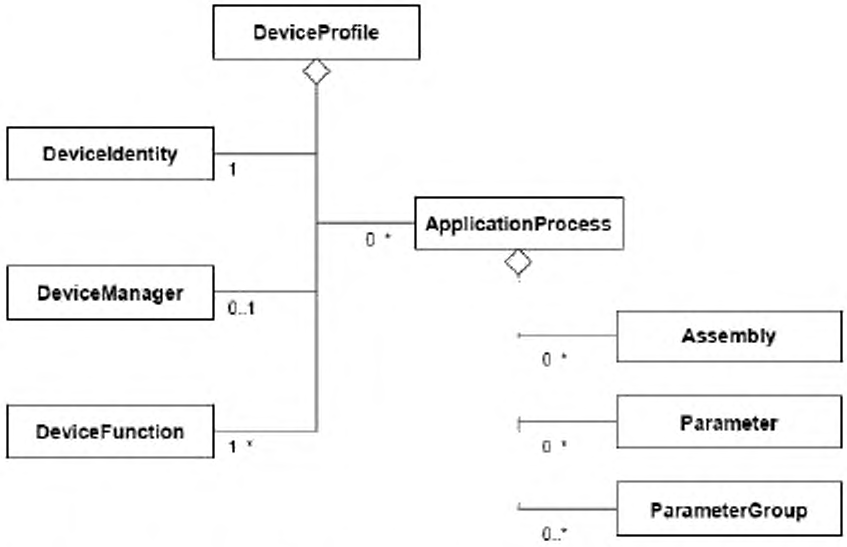


Рисунок 8 — Диаграмма класса профиля устройства EtherNet/IP

Описание существующих форматов профиля устройства EtherNet/IP приведено в С.1 приложения С.

XML схема, представляющая шаблон профиля устройства EtherNet/IP, определена в С.2.1.3.3. Имя файла этой схемы XML — «2CIP_Device_Profile.xsd».

Примечание – Диаграмма класса профиля устройства EtherNet/IP, представленная на рисунке 8, определяет основные классы. Затем некоторые классы должны быть представлены в развернутом виде согласно приложению С.

XML схема, представляющая инкапсуляцию ранее принятого EtherNet/IP EDS в шаблон профиля устройства ИСО 15745, определена в С.2.2.2 приложения С. Имя файла этой XML схемы — «EDS_Device_Profile_wrapper.xsd». Описание синтаксиса ASCII ранее принятого EDS приведено в С.4 приложения С.

6.3.1.2 Device identity (идентификация устройства)

Класс DeviceIdentity содержит атрибуты, которые уникальным образом идентифицируют устройство и поддерживают сервисы, которые позволяют извлекать эту информацию из устройства.

Эти атрибуты предоставляют, в частности, следующую информацию:

– идентификацию изготовителя (имя и код идентификации);

– идентификацию устройства (тип устройства, имя продукта, версия, серийный номер);

– классификацию устройства.

Сервисы позволяют осуществлять:

– перезагрузку устройства;

– получение атрибутов DeviceManager.

6.3.1.3 Device manager (менеджер устройства)

Класс DeviceManager содержит атрибуты и поддерживает сервисы, используемые для мониторинга и конфигурации устройства.

Эти атрибуты предоставляют, в частности, следующую информацию:

– версию объекта идентичности EtherNet/IP;

– информацию о структуре устройства (для устройств, интегрированных в модульную систему).

Сервисы позволяют осуществлять:

– перезагрузку устройства;

– извлечение атрибутов DeviceManager.

6.3.1.4 Device function (функция устройства)

Класс DeviceFunction содержит атрибуты и поддерживает сервисы, позволяющие осуществлять управление функциями устройства (например,

конфигурацией).

Пример – Примерами объектов DeviceFunction являются Overload (перегрузка), Presence Sensing (обнаружение присутствия), Analogue Input (аналоговый ввод), Discrete Output (дискретный вывод).

Примечание – Настоящий стандарт не распространяется на DeviceFunction.

6.3.1.5 Application process (прикладной процесс)

На рисунке 9 представлена структура класса ApplicationProcess.

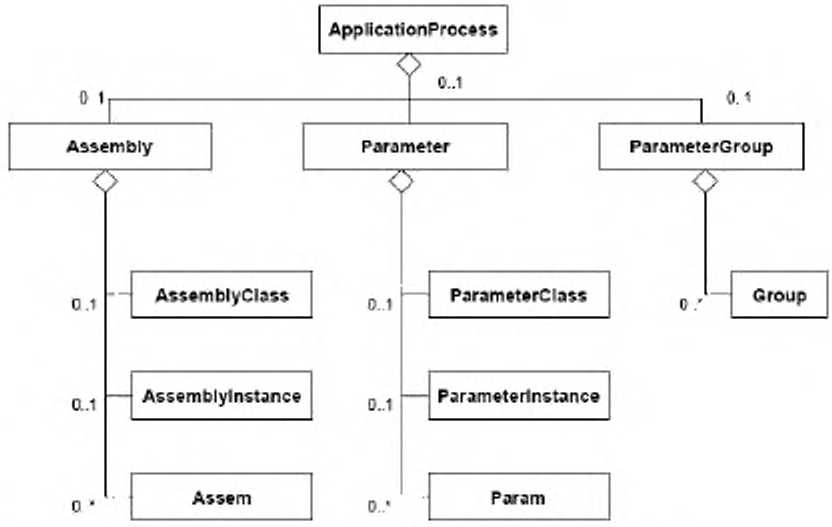


Рисунок 9 — Диаграмма класса EtherNet/IP ApplicationProcess

Класс Assembly объединяет несколько элементов данных процесса приложения в один блок с целью оптимизации коммуникаций. Класс Parameter предоставляет стандартный интерфейс, обеспечивающий доступ к отдельным элементам данных процесса приложения. Класс ParameterGroup определяет группы связанных параметров для специальных целей (например, конфигурации, мониторинга).

Классы Assembly и Parameter поддерживают атрибуты и сервисы на уровнях как класса, так и экземпляра.

Классы Assem, Param и Group определяют отдельные экземпляры основных классов.

Примечание – Классы Assembly и Parameter соответствуют объекту EtherNet/IP Assembly и объектам Parameter. Полное определение объекта Assembly приведено в МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2).

6.3.2 Профиль коммуникационной сети

6.3.2.1 Общие положения

На рисунке 10 представлена диаграмма класса профилей коммуникационной сети EtherNet/IP.

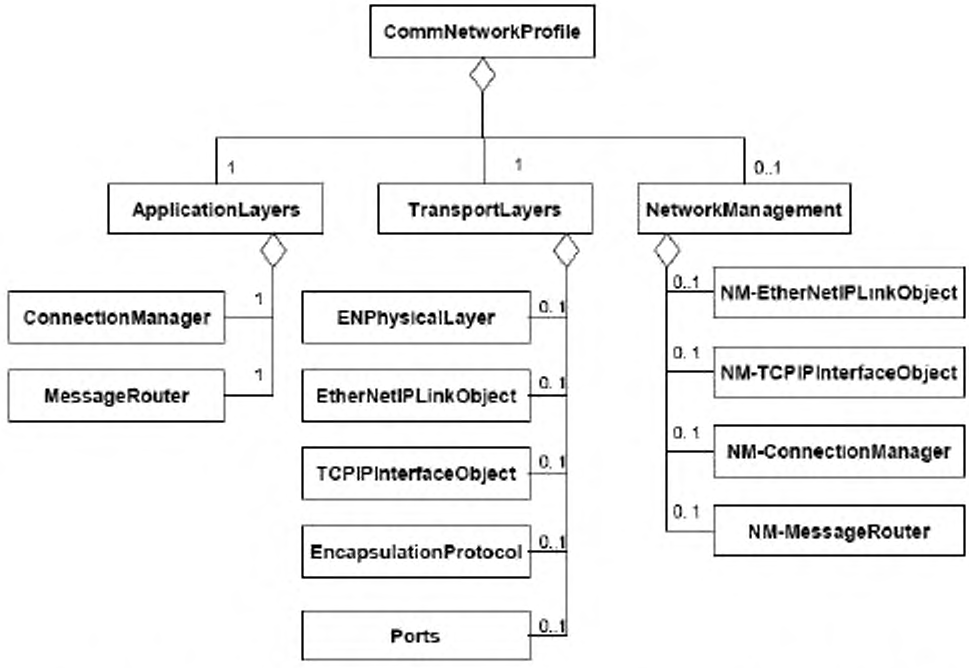


Рисунок 10 — Диаграмма класса профилей коммуникационной сети EtherNet/IP

Существующие форматы профилей коммуникационной сети EtherNet/IP описаны в С.3 приложения С.

XML схема, представляющая шаблон профиля коммуникационной сети EtherNet/IP, определена в С.3.1.3 приложения С. Имя файла этой XML схемы — «ENet_CommNet_Profile.xsd».

XML схема, представляющая инкапсуляцию ранее принятого EtherNet/IP EDS в шаблон профиля коммуникационной сети ИСО 15745, определена в С.3.2.2 приложения С. Имя файла этой XML схемы —

«EDS_CommNet_Profile_wrapper.xsd». Синтаксис ASCII ранее принятого EDS описан в С.4 приложения С.

6.3.2.2 Application Layers (прикладные уровни)

Класс EtherNet/IP ApplicationLayers представляет комбинированные профили трех верхних уровней OSI модели интеграции коммуникационной сети EtherNet/IP.

Далее этот класс подразделяется на несколько классов согласно рисунку 10:

– ConnectionManager (менеджер связи) определяет характеристики, относящиеся к соединениям и управлению соединениями;

– MessageRouter (маршрутизатор сообщений) определяет характеристики, связанные с маршрутизацией внутренних сообщений в устройстве.

Примечание – Полные описания объектов Connection Manager и Message Router приведены в МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2).

6.3.2.3 Transport Layers (транспортные уровни)

Класс EtherNet/IP TransportLayers представляет комбинированные профили для четырех нжних уровней OSI модели интеграции коммуникационной сети EtherNet/IP.

Далее этот класс подразделяется на несколько классов согласно рисунку 10:

– ENPhysicalLayer (физический уровень EN) определяет характеристики физического уровня;

– EtherNetIPLinkObject определяет характеристики, связанные с конфигурацией и мониторингом канала передачи данных;

– TCPIPInterfaceObject определяет характеристики, связанные с конфигурацией и мониторингом TCP/IP;

– EncapsulationProtocol (протокол инкапсуляции) определяет характеристики, связанные с инкапсуляцией сообщений приложения в TCP/IP;

– порты определяют порты устройства, которые могут направлять сообщения из одной связи на другую связь.

Примечание – Полные описания соответствующих объекта связи EtherNet Link и объект интерфейса TCP/IP приведены в МЭК 61158-4:2003 (тип 2), а опции Encapsulation Protocol (протокол инкапсуляции) изложены в МЭК 61158-6:2003 (тип 2).

6.3.2.4 Network management (управление сетью)

Класс EtherNet/IP NetworkManagement представляет средства наладки конфигурации и характеристик сети в модели интеграции коммуникационной сети EtherNet/IP.

Далее этот класс подразделяется на несколько классов согласно рисунку 10:

– NM-EtherNetIPLinkObject, NM-TCPIPInterfaceObject, NM-ConnectionManager и NM-MessageRouter определяют характеристики, связанные с менеджментом класса соответствующих объектов.

6.4 PROFINET

6.4.1 Общие положения

Примечание – Дополнительно к терминологии и нотации UML, приведенным в ИСО 15745-1:2003 (приложение А), используют нотацию кратности (UML V1.4). Кратность атрибута приводят в квадратных скобках.

6.4.2 Профиль устройства

6.4.2.1 Общие положения

На рисунке 11 представлена диаграмма класса профилей устройства GSDML.

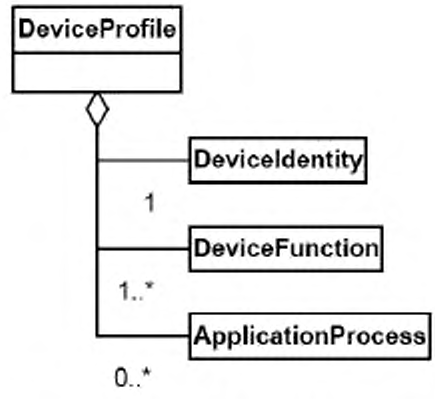


Рисунок 11 — Диаграмма класса профилей устройства GSDML

XML схема, представляющая шаблон профиля устройства GSDML, определена в D.5.2 приложения D. Заголовок профиля для профиля устройства GSDML должен включать в себя:

<ProfileHeader>
<ProfileIdentification>PROFINET Device Profile</ProfileIdentification>
<ProfileRevision>1.00</ProfileRevision>
<ProfileName>Device Profile for PROFINET Devices</ProfileName>
<ProfileSource>PROFIBUS Nutzerorganisation e. V. (PNO)</ProfileSource>
<ProfileClassID>Device</ProfileClassID>
<ISO15745Reference>
<ISO15745Part>4</ISO15745Part>
<ISO15745Edition>1</ISO15745Edition>
<ProfileTechnology>GSDML</ProfileTechnology>
</ISO15745Reference>
</ProfileHeader>

6.4.2.2 Device Identity (идентификация устройства)

На рисунке 12 представлена диаграмма класса DeviceIdentity.

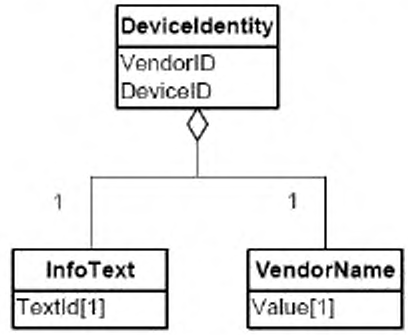


Рисунок 12 — Диаграмма класса DeviceIdentity

Атрибуты и семантика классов определены в D.4.2 приложения D.

6.4.2.3 Device Function (функция устройства)

На рисунке 13 представлена диаграмма класса DeviceFunction.

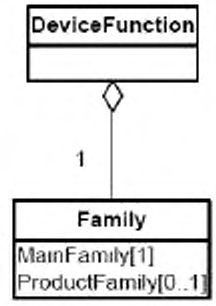


Рисунок 13 — Диаграмма класса DeviceFunction

Атрибуты и семантика классов определены в D.4.3 приложения D.

6.4.2.4 Application Process (прикладной процесс)

6.4.2.4.1 Общие положения

На рисунке 14 представлена структура элемента ApplicationProcess. Подробные определения классов UML без поля атрибутов приведены на отдельной диаграмме. Атрибуты и семантика классов определены в D.4.4 приложения D.

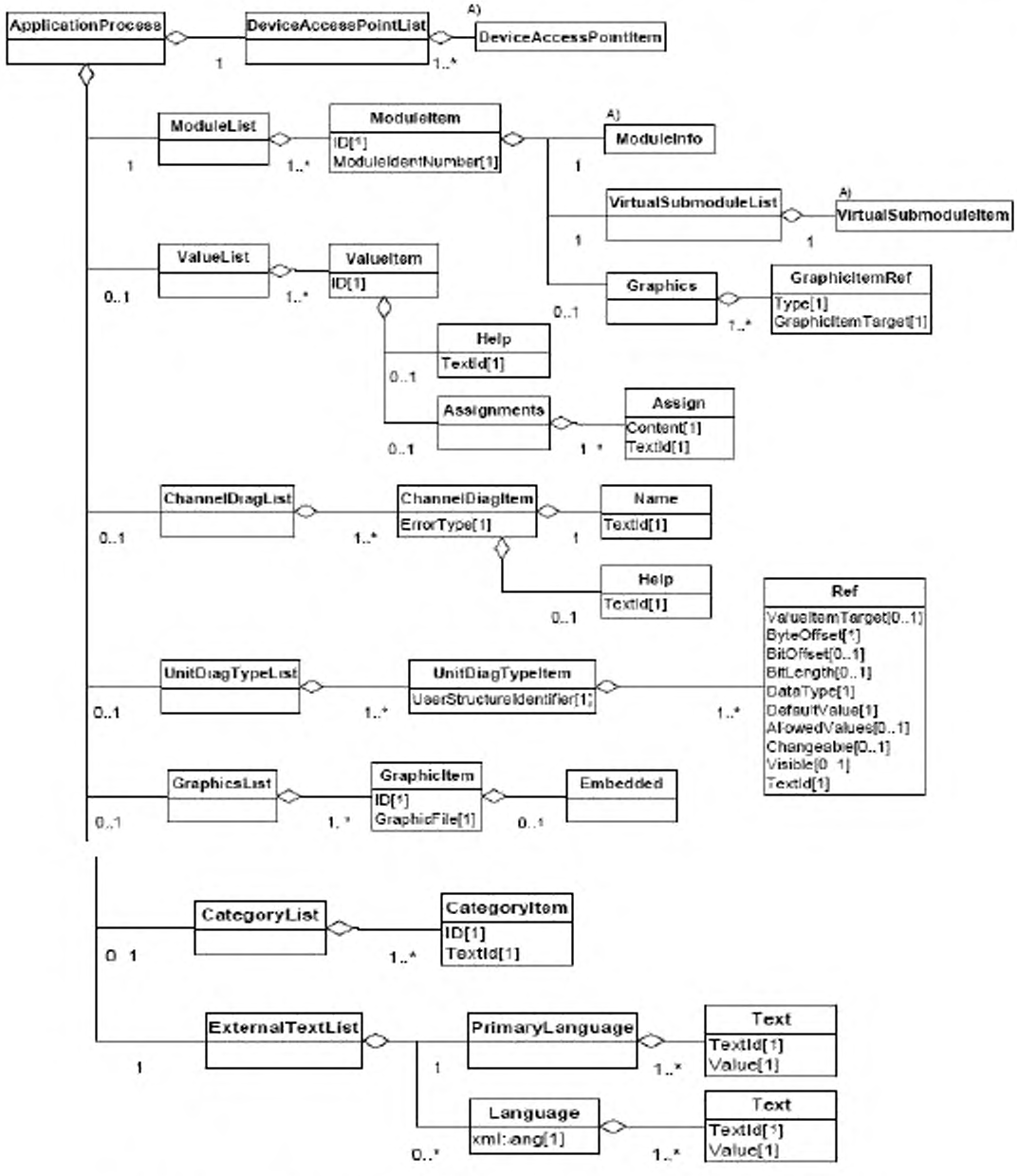


Рисунок 14 — Диаграмма класса ApplicationProcess PROFINET

6.4.2.4.2 DeviceAccessPointItem (точечный элемент доступа к устройству)

На рисунке 15 представлена структура элемента DeviceAccessPointItem. Подробные описания классов UML без полей атрибутов приведены на отдельной диаграмме. Атрибуты и семантика

классов определены в D.4.5 приложения D.

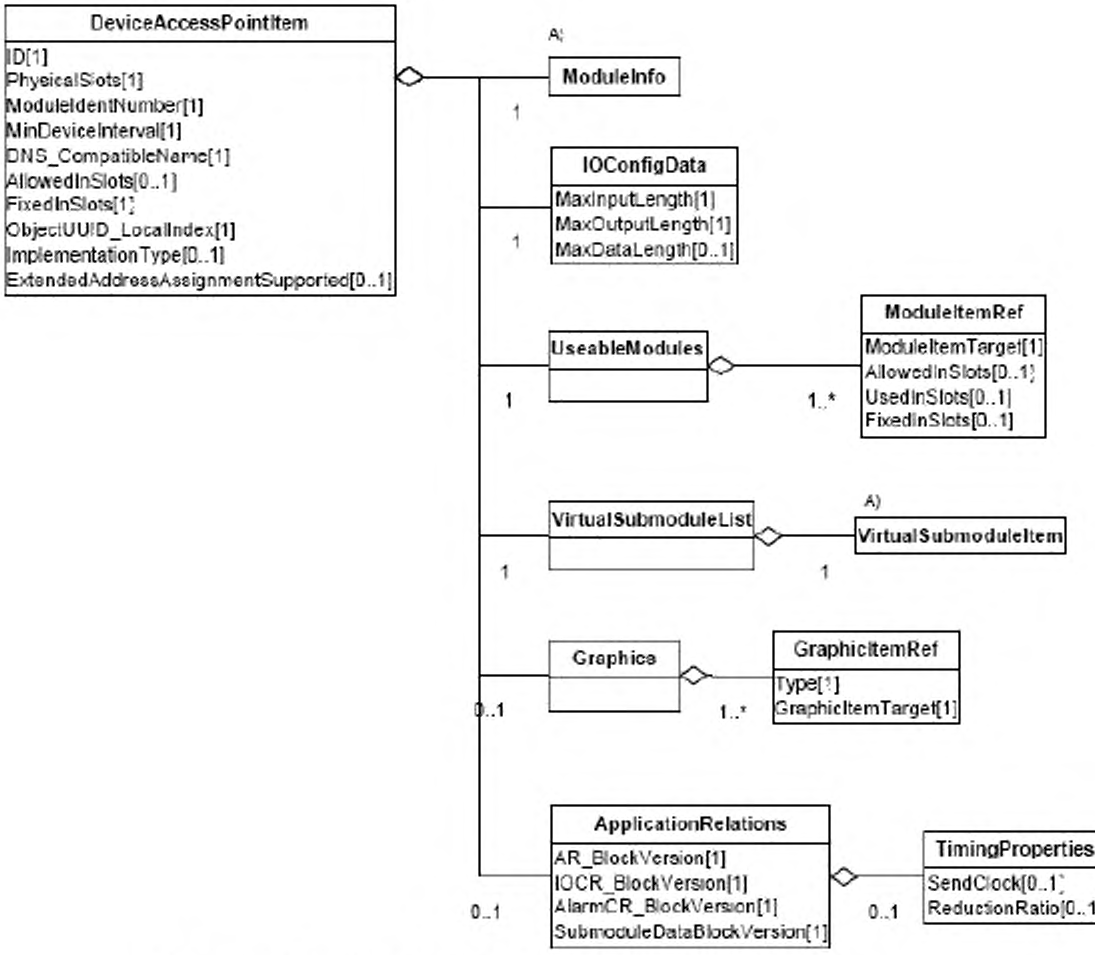


Рисунок 15 — Диаграмма класса DeviceAccessPointItem

6.4.2.4.3 VirtualSubmoduleItem (виртуальный элемент подмодуля)

На рисунке 16 приведено описание элемента VirtualSubmoduleItem. Подробные описания классов UML без полей атрибутов приведены на отдельной диаграмме. Атрибуты и семантика этих классов определены в D.4.6 приложения D.

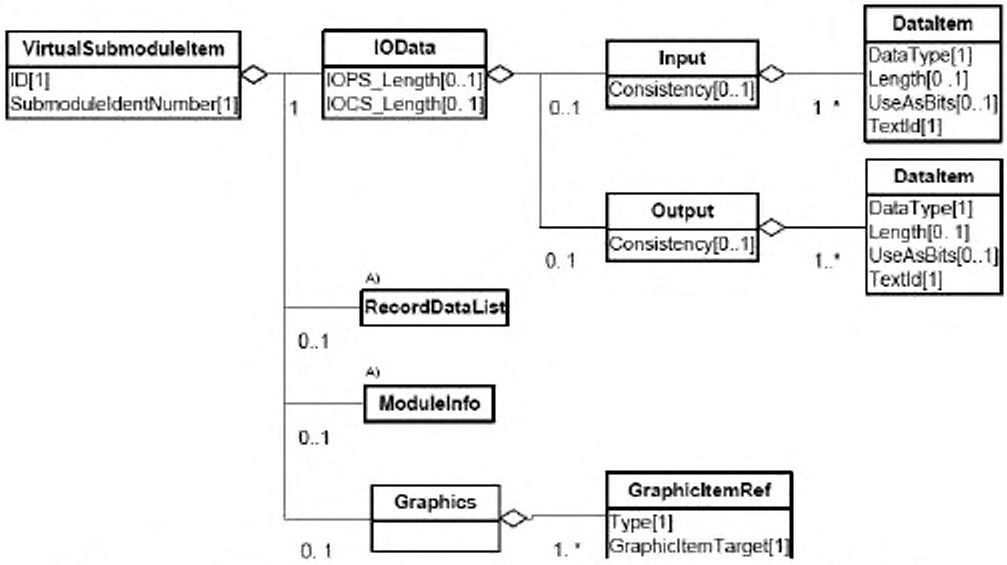


Рисунок 16 — Диаграмма класса PROFINET VirtualSubmoduleItem

6.4.2.4.4 RecordDataList (список записи данных)

На рисунке 17 представлена диаграмма элемента RecordDataList. Атрибуты и семантика классов определены в D.4.7 приложения D.

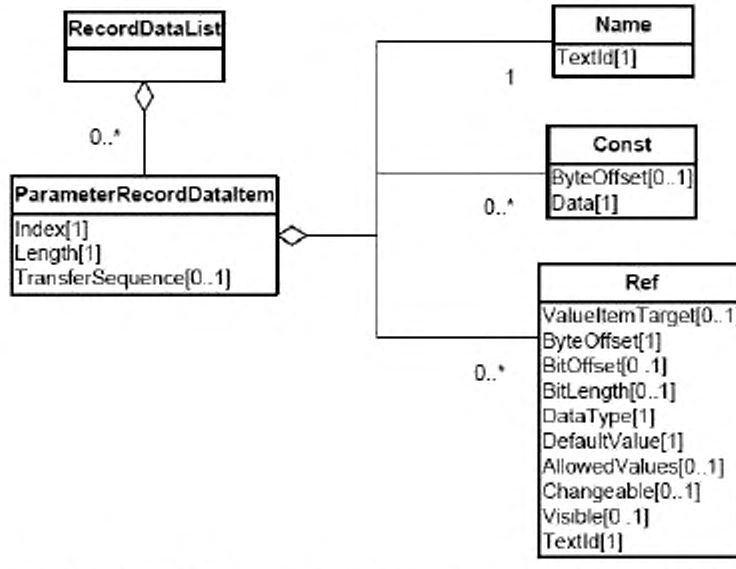


Рисунок 17 — Диаграмма класса PROFINET RecordDataList

6.4.2.4.5 ModuleInfo (информационный модуль)

На рисунке 18 представлена диаграмма элемента ModuleInfo. Атрибуты и семантика этих классов определены в D.4.8 приложения D.

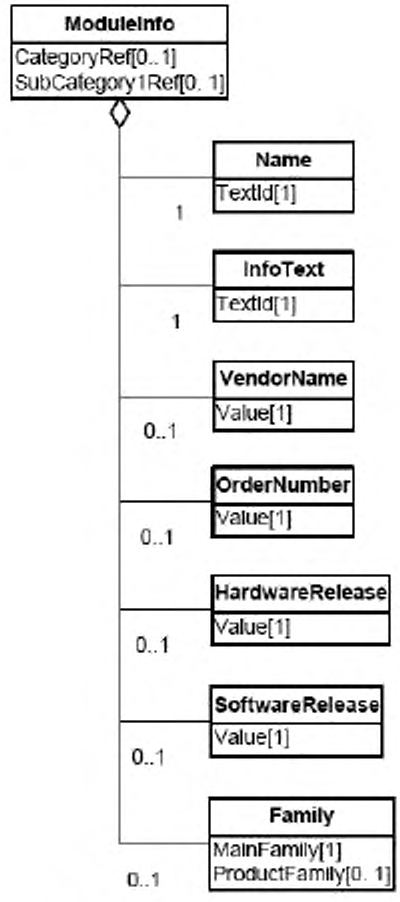


Рисунок 18 — Диаграмма класса PROFINET ModuleInfo

6.4.3 Профиль коммуникационной сети

На рисунке 19 представлена диаграмма класса профиля коммуникационной сети GSDML.

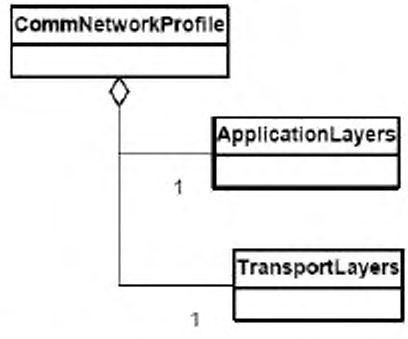


Рисунок 19 — Диаграмма класса профиля коммуникационной сети PROFINET

Примечание – В GSDML классы профиля коммуникационной сети пустые, в связи с тем, что экземпляры коммуникационных характеристик устройства PROFINET не предоставлены.

XML схема, представляющая шаблон профиля коммуникационной сети GSDML, определена в D.5.3 приложения D.

6.5 Modbus TCP

6.5.1 Профиль устройства

6.5.1.1 Общие положения

На рисунке 20 представлена структура класса профиль устройства Modbus TCP.

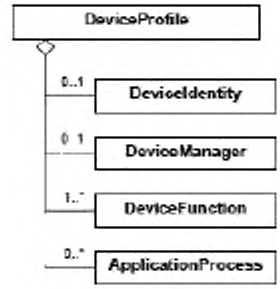


Рисунок 20 — Диаграмма класса профиля устройства Modbus TCP

Примечание – Диаграмма класса профиля устройства Modbus TCP, представленная на рисунке 20, определяет основные классы, подробные описания которых приведены в приложении E.

Схемы XML, представляющие шаблон профиля устройства Modbus TCP, определены в E.4.6. Шаблон состоит из двух частей:

- заголовка профиля DDXML, определенного в E.3 и
- профиля устройства DDXML, определенного в E.4.

6.5.1.2 Идентичность устройства

Класс DeviceIdentity (идентичность устройства) содержит атрибуты, которые не зависят от сети и процесса и уникально идентифицируют устройство.

На рисунке 21 представлена структура класса Modbus TCP DeviceIdentity.

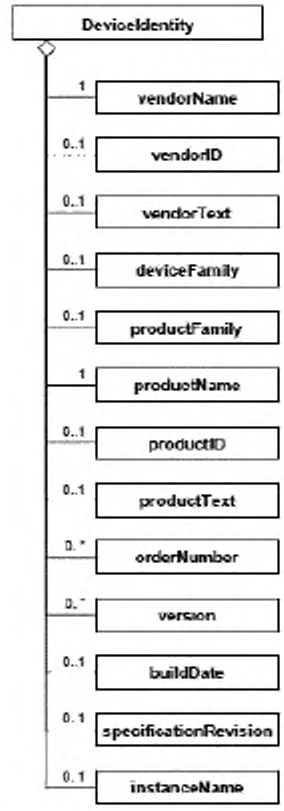


Рисунок 21 — Диаграмма класса Modbus TCP DeviceIdentity

Подробные описания класса шаблонов приведены в E.4.2 приложения E.

6.5.1.3 Менеджер устройства

Класс DeviceManager (менеджер устройства) содержит атрибуты и поддерживает сервисы, позволяющие осуществлять мониторинг устройства. Данные конфигурации и информация отображения, относящиеся к коммуникациям, определены в относящейся к коммуникационной сети части, структура которой соответствует схеме, приведенной в E.5.

На рисунке 22 представлена структура класса Modbus TCP DeviceManager.

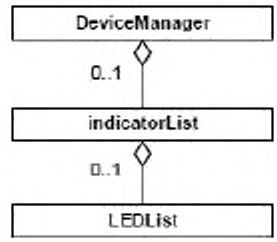


Рисунок 22 — Диаграмма класса Modbus TCP DeviceManager

Подробное описание класса приведено в Е.4.3 приложения Е.

6.5.1.4 Функция устройства

Класс DeviceFunction (функция устройства) описывает собственные функции устройства в терминах его технологии. Он содержит не зависящие от сети описания/определения технологических функциональных характеристик устройства.

На рисунке 23 представлена структура класса Modbus TCP DeviceFunction.

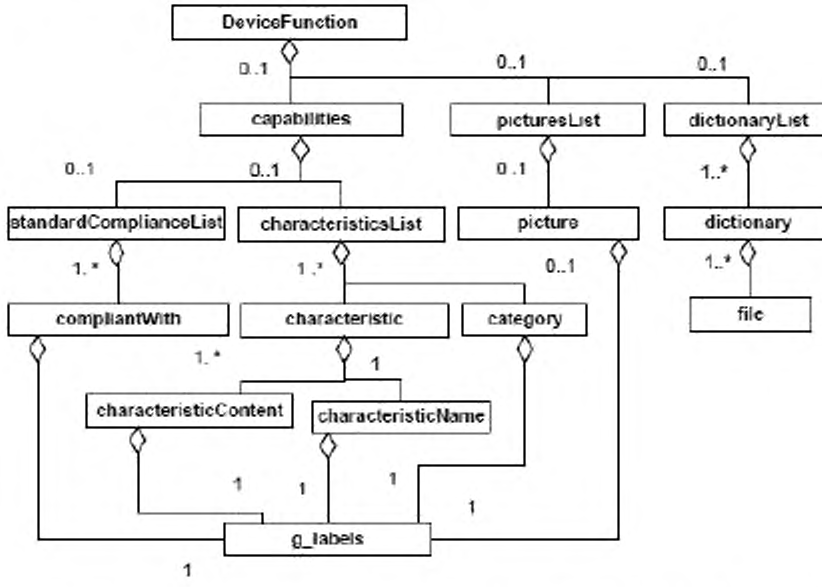


Рисунок 23 — Диаграмма класса Modbus TCP DeviceFunction

Подробное описание класса приведено в Е.4.4 приложения Е.

6.5.1.5 Прикладной процесс

Класс ApplicationProcess (прикладной процесс) представляет набор сервисов и параметров, которые формируют поведение и интерфейсы устройства в терминах приложения, независимо от технологии устройства и лежащих в его основе сетей и протоколов коммуникаций.

На рисунке 24 представлена структура класса Modbus TCP ApplicationProcess.

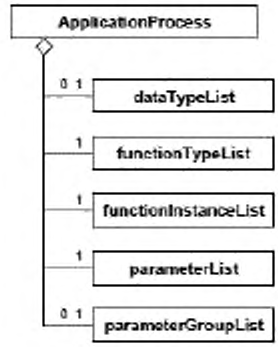


Рисунок 24 — Диаграмма класса Modbus TCP ApplicationProcess

Подробное описание класса приведено в Е.4.5 приложения Е.

6.5.2 Профиль коммуникационной сети

6.5.2.1 Общие положения

На рисунке 25 представлена структура класса профиль коммуникационной сети Modbus TCP. Далее эти классы должны быть представлены в развернутом виде и подробно описаны согласно приложению Е.

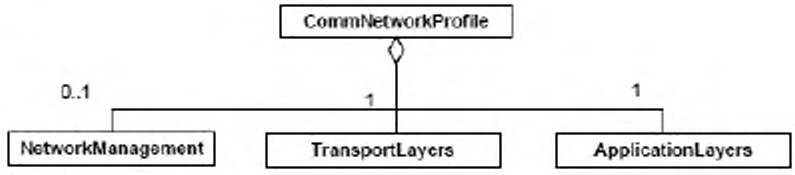


Рисунок 25 — Диаграмма класса профиля коммуникационной сети Modbus TCP

Схемы XML, представляющие шаблон профиля коммуникационной сети Modbus TCP, определены в Е.5.5 приложения Е. Аналогично профилю устройств, шаблон состоит из двух частей:

– заголовка профиля DDXML, определенного в E.3, и

– профиля коммуникационной сети DDXML, определенного в E.5.

6.5.2.2 Прикладные уровни

Класс Modbus TCP ApplicationLayers (прикладные уровни) представляет комбинированные профили для трех верхних уровней OSI модели интеграции коммуникационной сети Modbus TCP.

Подробное описание классов приведено в E.5.2 приложения E.

6.5.2.3 Транспортные уровни

Класс Modbus TCP TransportLayers (транспортные уровни) представляет комбинированные профили для четырех нижних уровней OSI модели интеграции коммуникационной сети Modbus TCP.

Подробные описания классов приведены в E.5.3 приложения E.

6.5.2.4 Менеджмент сети

Класс Modbus TCP NetworkManagement (менеджмент сети) представляет конфигурацию сети и возможности регулирования характеристик модели интеграции коммуникационной сети Modbus TCP.

Подробные описания классов приведены в E.5.4 приложения E.

6.6 EtherCAT

6.6.1 Профиль устройства

6.6.1.1 Общие положения

На рисунке 26 представлена структура класса профиля устройства EtherCAT.

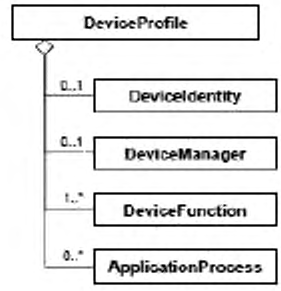


Рисунок 26 — Диаграмма класса профиля устройства EtherCAT

Примечание – Диаграмма класса профиля устройства EtherCAT, изображенная на рисунке 26, определяет только основные классы, описания которых приведены в приложении F.

Схема XML, представляющая шаблон профиля устройства EtherCAT, определена в F.4.6 приложения F.

Шаблон состоит из двух частей:

– заголовка профиля EtherCAT, определенного в F.3, и

– профиля устройства EtherCAT, определенного в F.4.

6.6.1.2 Идентичность устройства

Класс DeviceIdentity (идентичность устройство) содержит атрибуты, не зависящие от сети и от процесса и уникально идентифицирующие устройство.

На рисунке 27 представлена структура класса EtherCAT DeviceIdentity.

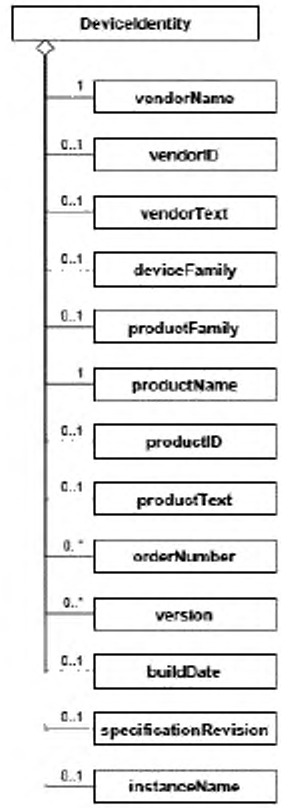


Рисунок 27 — Диаграмма класса EtherCAT DeviceIdentity

Подробное описание класса приведено в F 4.2 приложения F.

6.6.1.3 Менеджер устройства

Класс DeviceManager (менеджер устройства) содержит атрибуты и поддерживает сервисы, позволяющие осуществлять мониторинг устройства. Относящиеся к коммуникациям данные конфигурации и информация отображения определены в относящейся к коммуникационной сети части, структура которой соответствует схеме, приведенной в F.5 приложения F.

На рисунке 28 представлена структура класса EtherCAT DeviceManager.

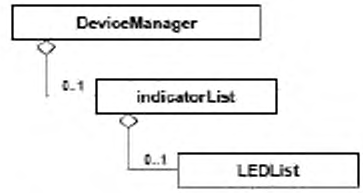


Рисунок 28 — Диаграмма класса EtherCAT DeviceManager

Подробное описание класса приведено в F.4.3 приложения F.

6.6.1.4 Функция устройства

Класс DeviceFunction (функция устройства) описывает собственные функции устройства в терминах технологии и содержит не зависящие от сети описания/определения технологических функциональных характеристик устройства.

На рисунке 29 представлена структура класса EtherCAT DeviceFunction.

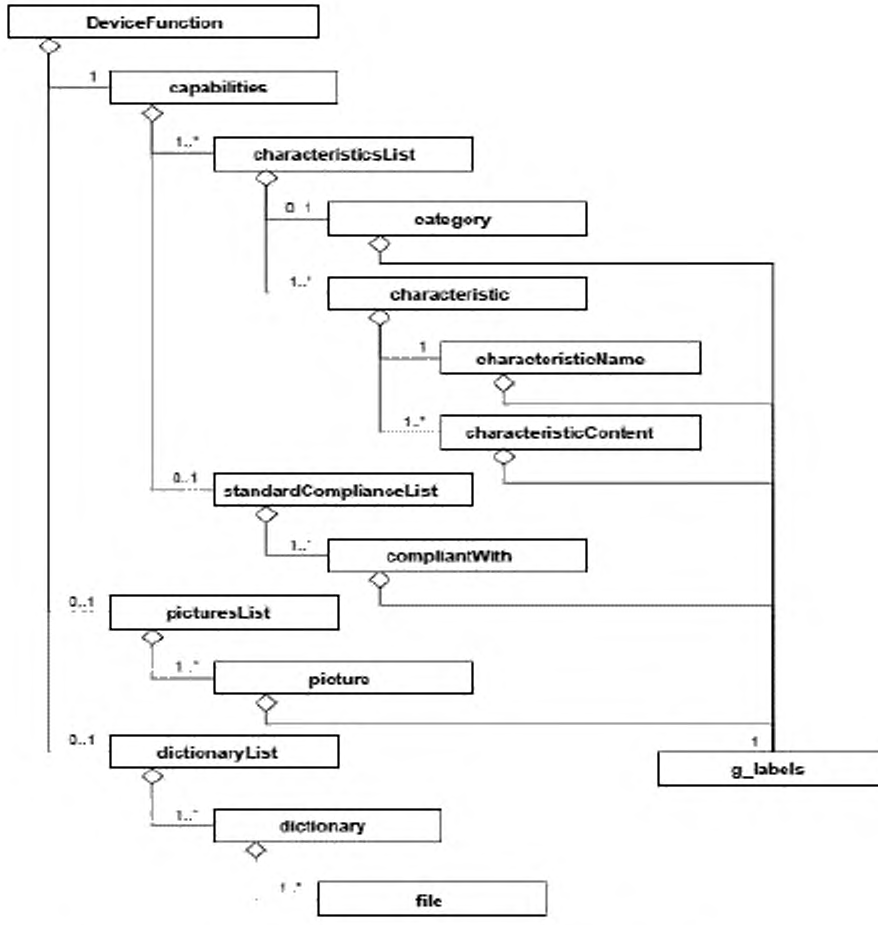


Рисунок 29 — Диаграмма класса EtherCAT DeviceFunction

Подробное описание класса приведено в F.4.4 приложения F.

6.6.1.5 Прикладной процесс

Класс ApplicationProcess (прикладной процесс) представляет набор сервисов и параметров, которые формируют поведение и интерфейсы устройства в терминах приложения, независимо от технологии устройства и лежащих в его основе коммуникационных сетей и протоколов коммуникаций.

На рисунке 30 представлена структура класса EtherCAT ApplicationProcess.

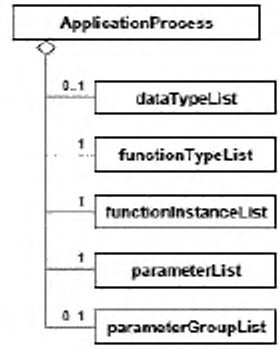


Рисунок 30 — Диаграмма класса EtherCAT Application Process

Подробное описание класса приведено в F.4.5 приложения F.

6.6.2 Профиль коммуникационной сети

6.6.2.1 Общие положения

На рисунке 31 представлена структура класса профиля коммуникационной сети EtherCAT, описание которого приведено в приложении F.

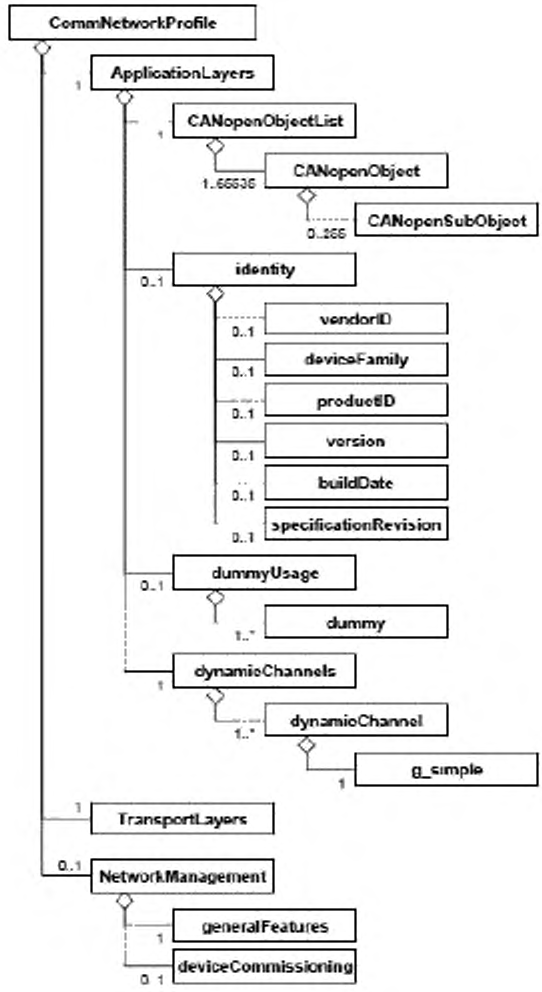


Рисунок 31 — Диаграмма класса профиля коммуникационной сети EtherCAT

Схема XML, представляющая профиль коммуникационной сети EtherCAT, определена в F.5.5 приложения F.

6.6.2.2 Прикладные уровни

Класс EtherCAT ApplicationLayers (прикладные уровни) представляет комбинированные профили для трех верхних уровней OSI модели интеграции коммуникационной сети EtherCAT.

Подробное описание класса приведено в F.5.2 приложения F.

6.6.2.3 Транспортные уровни

Класс EtherCAT TransportLayers (транспортные уровни) представляет

комбинированные профили для четырех нижних уровней OSI модели интеграции коммуникационной сети EtherCAT.

Подробное описание класса приведено в F.5.3 приложения F.

6.6.2.4 Менеджмент сети

Класс EtherCAT NetworkManagement (менеджмент сети) представляет конфигурацию сети и возможности регулирования характеристик модели интеграции коммуникационной сети EtherCAT.

Подробное описание класса приведено в F.5.4 приложения F.

6.7 ETHERNET Powerlink

6.7.1 Профиль устройства

6.7.1.1 Общие положения

На рисунке 32 представлена структура класса профиля устройства ETHERNET Powerlink.

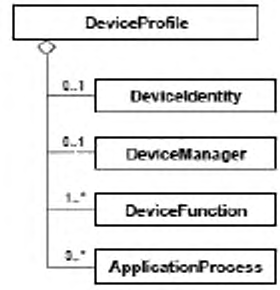


Рисунок 32 — Диаграмма класса профиля устройства ETHERNET Powerlink

Примечание – Диаграмма класса профиля устройства ETHERNET Powerlink, представленная на рисунке 32, определяет только основные классы, подробные определения которых приведены в приложении G.

Схема XML, представляющая шаблон профиля устройства ETHERNET Powerlink, определена в G.4.6 приложения G.

Шаблон состоит из двух частей:

- заголовка профиля EPL, определенного в G.3, и
- профиля устройства EPL, определенного в G.4.

6.7.1.2 Идентичность устройства

Класс DeviceIdentity (идентичность устройства) содержит атрибуты, не зависящие от сети и процесса, и уникально идентифицирующие устройство.

На рисунке 33 представлена структура класса ETHERNET Powerlink DeviceIdentity.

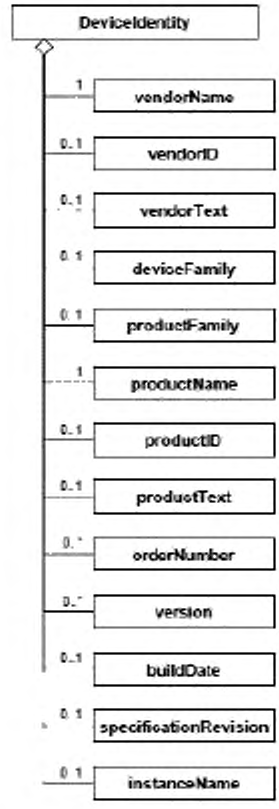


Рисунок 33 — Диаграмма класса ETHERNET Powerlink DeviceIdentity

Подробное описание класса приведено в G.4.2 приложения G.

6.7.1.3 Менеджер устройства

Класс DeviceManager (менеджер устройства) содержит атрибуты и поддерживает сервисы, позволяющие осуществлять мониторинг устройства. Относящиеся к коммуникациям данные конфигурации и информация отображения определены в относящейся к коммуникационной сети части, структура которой соответствует схеме, приведенной в G.5 приложения G.

На рисунке 34 представлены структура класса ETHERNET Powerlink DeviceManager.

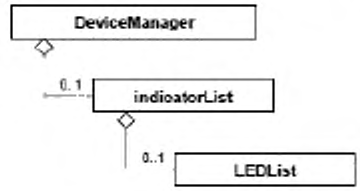


Рисунок 34 — Диаграмма класса ETHERNET Powerlink DeviceManager

Подробное описание класса приведено в G.4.3 приложения G.

6.7.1.4 Функция устройства

Класс DeviceFunction (функция устройства) описывает собственные функции устройства в терминах технологии и содержит не зависящие от сети описания/определения технологических функциональных характеристик устройства.

На рисунке 35 представлена структура класса ETHERNET Powerlink DeviceFunction.

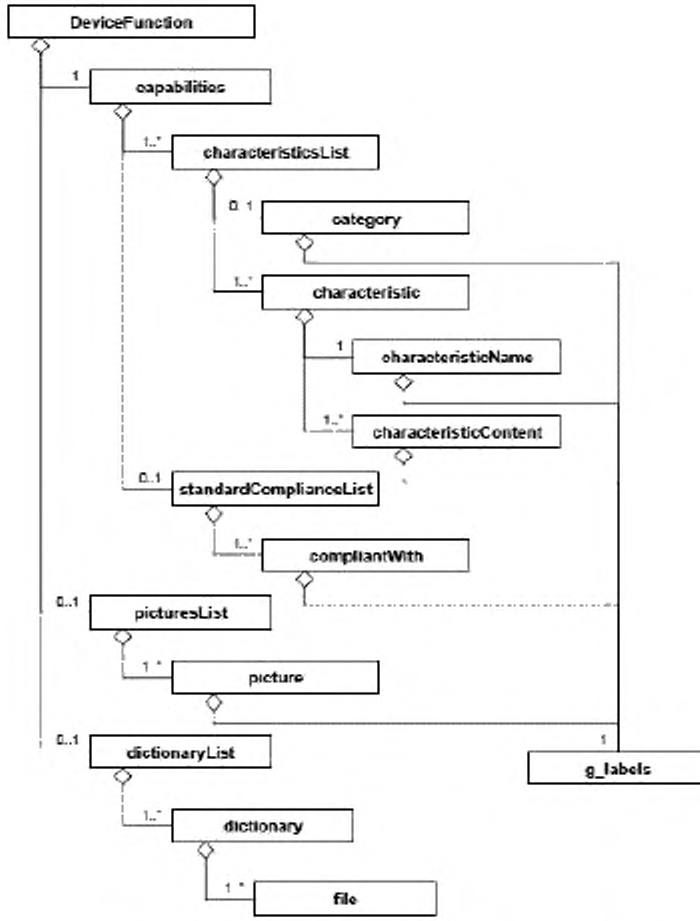


Рисунок 35 — Диаграмма класса ETHERNET Powerlink DeviceFunction

Подробное описание класса приведено в G.4.4 приложения G.

6.7.1.5 Прикладной процесс

Класс ApplicationProcess (прикладной процесс) представляет набор сервисов и параметров, которые формируют поведение и интерфейсы устройства в терминах приложения, независимо от технологии устройства и лежащих в его основе коммуникационных сетей и протоколов коммуникаций.

На рисунке 36 представлена структура класса ETHERNET Powerlink ApplicationProcess.

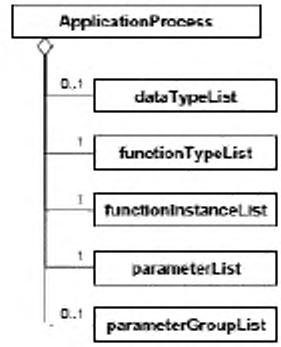


Рисунок 36 — Диаграмма класса ETHERNET Powerlink ApplicationProcess

Подробное описание класса приведено в G.4.5 приложения G.

6.7.2 Профиль коммуникационной сети

6.7.2.1 Общие положения

На рисунке 37 представлена структура класса профиля коммуникационной сети ETHERNET Powerlink, подробное описание которого приведено в приложении G.

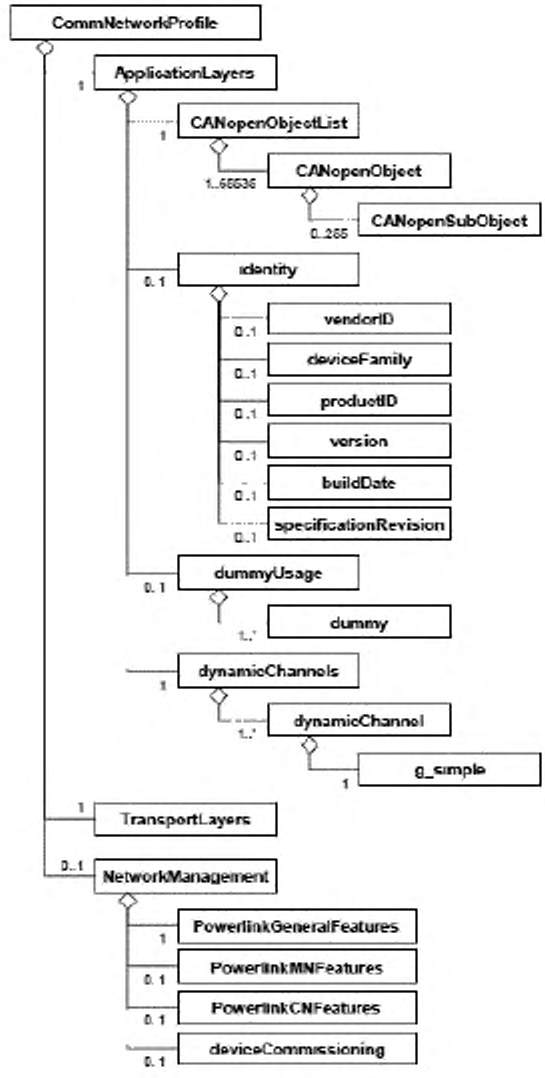


Рисунок 37 — Диаграмма класса профиля коммуникационной сети ETHERNET Powerlink

Схема XML, представляющая профиль коммуникационной сети ETHERNET Powerlink, определена в G.5.5 приложения G.

6.7.2.2 Прикладные уровни

Класс ETHERNET Powerlink ApplicationLayers (прикладные уровни) представляет комбинированные профили для трех верхних уровней OSI модели интеграции коммуникационной сети ETHERNET Powerlink.

Подробное описание класса приведено в G.5.2 приложения G.

6.7.2.3 Транспортные уровни

Класс ETHERNET Powerlink TransportLayers (транспортные уровни) представляет комбинированные профили для четырех нижних уровней OSI модели интеграции коммуникационной сети ETHERNET Powerlink.

Подробное описание класса приведено в G.5.3 приложения G.

6.7.2.4 Менеджмент сети

Класс ETHERNET Powerlink NetworkManagement (менеджмент сети) представляет конфигурацию сети и возможности регулирования характеристик модели интеграции коммуникационной сети ETHERNET Powerlink.

Подробное описание класса приведено в G.5.4 приложения G.

Приложение А
(обязательное)
Шаблоны профиля ADS-net

А.1 Общие положения

Верхние уровни ADS-net основаны на сети автономной децентрализованной системы (ADS-net). Этот протокол моделирует все коммуникации и обмен сообщениями, имеющими место при взаимосвязях производитель – потребитель.

ADS-net предоставляет доступ ко всем данным конфигурации, информации о статусе и параметрам рабочего цикла узла и/или системы.

А.2 Описание шаблона профиля устройства ADS-net

А.2.1 Общие положения

Файлы XML профиля устройства должны соответствовать XML схеме профиля устройства, установленной в А.2.3.

Содержание этой XML схемы выведено исходя из диаграмм класса профиля устройства, показанных в 6.1.1, и расширено дополнительными элементами, позволяющими дать полное описание требований и возможностей коммуникационной сети.

А.2.2 Семантика элементов схемы XML

А.2.2.1 DeviceIdentity (идентичность устройства)

Семантика элемента DeviceIdentity определена в таблице А.1. Эти элементы используются в среде рабочего цикла ADS-net в целях предоставления информации для полной идентификации устройства.

Таблица А.1 — Элементы DeviceIdentity

Элементы	Описание и семантика
VendorName	Имя продавца устройства
ProductCode	Идентификатор устройства в продукте продавца (максимум шесть знаков)
ProductRevision	Номер версии продукта
DeviceName	Имя устройства

Более подробные данные о семантике каждого атрибута см. [4].

A.2.2.2 DeviceManager (менеджер устройства)

Семантика субэлементов DeviceManager-Attributes элемента DeviceManager определена в таблице А.2. Эти субэлементы используются в среде рабочего цикла ADS-net.

Таблица А.2 — Элементы DeviceManager-Attributes

Элементы	Описание и семантика
DeviceState (устройство состояние)	Состояние устройства

A.2.2.3 DeviceFunction (функция устройства)

Семантика субэлементов DeviceFunction-Attributes элемента DeviceFunction определена в таблице А.3. Эти субэлементы используются в среде рабочего цикла ADS-net.

Таблица А.3 — Элементы DeviceFunction-Attributes

Элементы	Описание и семантика
DFNO	Номер поля данных (десятичная нотация: 0 - 255)
LNODENO	Номер логического узла (десятичная нотация: 0 - 4095)
MODE	Режим сообщения (десятичная нотация: 0 - 1): 0 = режим онлайн; 1 = тестовый режим

Более подробные сведения о семантике каждого атрибута см. [4].

A.2.2.4 ApplicationProcess (прикладной процесс)

Семантика субэлементов DeviceProcess-Attributes элемента DeviceProcess определена в таблице А.4.

Таблица А.4 – Элементы DeviceProcess-Attributes

Элементы	Описание и семантика
Producer-TCD-List (Список TCD производителей)	Список TCD (десятичная нотация: 1 – 65534) для посылаемых из процесса приложения сообщений
Consumer-TCD-List (Список TCD потребителей)	Список TCD (десятичная нотация: 1 – 65534) для получаемых процессом приложения сообщений
Приоритет	Приоритет (десятичная нотация: 0 – 7), ассоциированный с сообщением, как для посылаемых, так и получаемых сообщений

Более подробные сведения о семантике каждого атрибута см. [4].

A.2.3 ADS-net_Device_Profile.xsd

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                                    ">
<xsd:annotation>
<xsd:documentation>
Device Profile - ISO 15745 Part 4
</xsd:documentation>
</xsd:annotation>
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP"/>
<xsd:enumeration value="Process"/>
<xsd:enumeration value="InformationExchange"/>
<xsd:enumeration value="Resource"/>
<xsd:enumeration value="Device"/>
<xsd:enumeration value="CommunicationNetwork"/>
<xsd:enumeration value="Equipment"/>
<xsd:enumeration value="Human"/>
<xsd:enumeration value="Material"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="CSI"/>
<xsd:enumeration value="HCI"/>
```

```
<xsd:enumeration value="ISI"/>
<xsd:enumeration value="API"/>
<xsd:enumeration value="CMI"/>
<xsd:enumeration value="ESI"/>
<xsd:enumeration value="FSI"/>
<xsd:enumeration value="MTI"/>
<xsd:enumeration value="SEI"/>
<xsd:enumeration value="USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>
ADS-net Device Profile Template
</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">

<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DeviceIdentity" minOccurs="1" maxOccurs="1"/>
<xsd:element ref="DeviceManager" minOccurs="0" maxOccurs="1"/>
<xsd:element ref="DeviceFunction" minOccurs="1" maxOccurs="unbounded"/>
<xsd:element ref="ApplicationProcess" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="DeviceIdentity">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="VendorName" type="xsd:string"/>
<xsd:element name="ProductCode" type="xsd:string"/>
<xsd:element name="ProductRevision" type="xsd:string"/>
<xsd:element name="DeviceName" minOccurs="0" maxOccurs="1" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DeviceManager-Attributes" minOccurs="0" maxOccurs="1"/>
<xsd:element ref="DeviceManager-Operations" minOccurs="0" maxOccurs="1"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceManager-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DeviceState" minOccurs="1" maxOccurs="1" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceManager-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_Single" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Set_Attribute_Single" minOccurs="0" maxOccurs="1"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceFunction">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DeviceFunction-Attributes" minOccurs="1" maxOccurs="1"/>
<xsd:element ref="DeviceFunction-Operations" minOccurs="0" maxOccurs="1"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceFunction-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DFNO" minOccurs="1" maxOccurs="unbounded"
type="xsd:unsignedByte"/>
<xsd:element name="LNODENO" minOccurs="1" maxOccurs="unbounded"
type="xsd:unsignedShort"/>
<xsd:element name="MODE" minOccurs="1" maxOccurs="1" type="xsd:unsignedShort"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceFunction-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0" maxOccurs="1"/>
```

```
<xsd:element name="Get-Attribute-All" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Set-Attribute-All" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Create" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Delete" minOccurs="0" maxOccurs="1"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationProcess">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ApplicationProcess-Attributes" minOccurs="0" maxOccurs="1"/>
<xsd:element ref="ApplicationProcess-Operations" minOccurs="0" maxOccurs="1"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationProcess-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Producer-TCD-List" minOccurs="1" maxOccurs="unbounded"
type="xsd:string"/>
<xsd:element name="Consumer-TCD-List" minOccurs="1" maxOccurs="unbounded"
type="xsd:string"/>
<xsd:element name="Priority" minOccurs="1" maxOccurs="unbounded"
type="xsd:unsignedShort"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationProcess-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Get-Attribute-All" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Set-Attribute-All" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Create" minOccurs="0" maxOccurs="1"/>
<xsd:element name="Delete" minOccurs="0" maxOccurs="1"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

A.3 Описание шаблона профиля коммуникационной сети ADS-net

A.3.1 Общие положения

Файлы XML профиля коммуникационной сети должны соответствовать XML схеме профиля коммуникационной сети, установленной в A.3.3.

Содержание XML схемы выводится из диаграмм класса профиля коммуникационной сети, показанных в 6.1.2, расширенного дополнительными элементами, позволяющими дать полное описание требований или возможностей коммуникационной сети.

A.3.2 Семантика элементов XML схемы

A.3.2.1 ApplicationLayers (прикладные уровни)

A.3.2.1.1 DataField (поле данных)

A.3.2.1.1.1 Общие положения

Данный элемент устанавливает поддерживаемые атрибуты экземпляра, используемые для контроля поля данных.

DataField соответствует домену, где подсистемы (т. е. узловые компьютеры или программы приложения) совместно используют информацию путем обмена сообщениями между равноправными устройствами. Эти сообщения имеют уникальную идентификацию в поле данных. Узловые компьютеры могут совместно использовать информацию с помощью указания номера поля данных в виде части идентификатора сообщения. Одно поле данных создается для адреса сети или подсети для коммуникаций между узловыми компьютерами или создается в памяти для коммуникаций между программами приложения в узловом компьютере.

Поле данных имеет уникальную идентификацию с помощью относящегося к нему номера поля данных (Data Field Number – DFNO). Уникальное значение DFNO присвоено всем полям данных в системе в диапазоне от 1 до 255. DFNO, равное 0, зарезервировано для коммуникаций внутри текущего узла.

Семантика субэлементов DataField-Attributes элемента DataField определена в таблице А.5.

Таблица А.5 – Элементы DataField-Attributes

Элементы	Описание и семантика
DFNO	Номер поля данных (десятичная нотация: 0 — 255)
NETADDR	Адрес подсети, описанный в стандартном формате с точкой Internet как «XXX.XXX.XXX.XXX»
BCADDR	Широковещательный адрес, описанный в стандартном в формате Internet standard dot format
NETMASK	Маска подсети, описанная в стандартном в формате Internet standard dot format
MCG-Table	Список номеров MCG (десятичная нотация: 0 — 255), использованных в этом поле данных

Более подробные сведения о семантике каждого атрибута см. [4].

A.3.2.1.1.2 AliveNotification (уведомление о существовании)

Данный элемент устанавливает атрибуты, используемые для проверки нормального состояния узлового компьютера.

Семантика элемента AliveNotification определена в таблице А.6.

Таблица А.6 – Элементы AliveNotification

Элементы	Описание и семантика
ND-Name	Имя узла (строка символов ASCII длиной до девяти символов)
OS-Name	Имя устройства поставщика (строка символов ASCII длиной до девяти символов). Рекомендуется придерживаться следующих руководящих указаний: «BN_MN» BN: Идентификатор продавца (2 символа) MN: Определенное продавцом имя устройства или OS
TM-Out	Интервал задержки в секундах после сигнала о рабочем состоянии при мониторинге рабочего состояния. Интервал времени, проходящий перед признанием узла, не работающим после прекращения передачи им сигналов о рабочем состоянии
Chg-Time	Время момента изменения состояния узла. Рекомендуется использовать время по Гринвичу (прошедшее время в секундах от 0 ч. 0 мин. 0 с., 1970). Установите 0, если время по Гринвичу использовать нельзя
IPADDR	Адрес IP LAN, описанный в стандартном в формате Internet standard dot format

Более подробные сведения о семантике каждого атрибута см. [4].

A.3.2.1.1.3 ErrorNotification (уведомление об ошибке)

Данный элемент устанавливает атрибуты, используемые для проверки ошибок узлового компьютера.

Семантика элемента ErrorNotification определена в таблице А.7.

Таблица А.7 — Элементы ErrorNotification

Элементы	Описание и семантика
CNT-Mode-Alive	Число модулей, сообщающих, что они в рабочем состоянии
Mod-Alive-List	Список статуса всех модулей (в рабочем или нерабочем состоянии)
CNT-Error	Число сообщенных ошибок
ERR-Name	Имя ошибки
ERR-List	Список номеров ошибок
Mod-NO	Номер модуля (1 или выше)
ERR-NO	Номер ошибки, которая уникально идентифицирована именем ERR
CNT-Option	Длина опциональной информации
Option-List	Список опциональной информации

Более подробные сведения о семантике каждого атрибута см. [4].

A.3.2.1.2 MessageSelection (выбор сообщений)

Данный элемент устанавливает атрибуты, используемые при обмене сообщениями в ADS-net.

Сообщение идентифицируется уникальным образом кодом транзакции (TCD). Передатчик посылает сообщение, ассоциированное с некоторым TCD, используя многоадресную передачу на группу, принимающую передачу и имеющую установленное поле данных. Узлы в указанной группе приема широковещательной передачи принимают только сообщения, имеющие определенный TCD.

Семантика субэлементов MessageSelection-Attributes элемента MessageSelection определена в таблице А.8.

Таблица А.8 – Элементы MessageSelection-Attributes

Элементы	Описание и семантика
Producer-TCD-List (список TCD производителей)	Список TCD (десятичная нотация: 1 — 65534) для посылаемых сообщений
Consumer-TCD-List (список TCD потребителей)	Список TCD (десятичная нотация: 1 — 65534) для получаемых сообщений
Producer- MCG -List (список MCG производителей)	Список MCG (десятичная нотация: 0 — 255), используемых для посылки сообщений
Producer- MCG -List (список MCG потребителей)	Список MCG (десятичная нотация: 0 — 255), используемых для приема сообщений

Более подробные сведения о семантике каждого описания см. [4].

A.3.2.2 TransportLayers (транспортные уровни)

A.3.2.2.1 EthernetObject (объект Ethernet)

Данный элемент устанавливает атрибуты Ethernet, используемые в ADS-net.

EthernetObject определяет атрибуты, связанные с конфигурацией и мониторингом канала передачи данных.

Семантика субэлементов EthernetObject-Attributes элемента EthernetObject приведена в таблице А.9.

Таблица А.9 – Элементы EthernetObject-Attributes

Элементы	Описание и семантика
MediaType	Тип среды Ethernet (например, 10Base-T, 100Base-TX)
CommRate	Скорость коммуникаций, зависящая от среды Ethernet (например, 100 Mbps – 100 мегабит в секунду)
Индикаторы	Состояние среды Ethernet
MACAddress	MAC адрес
ErrorLog	Список ошибок, возникших на физическом уровне или уровне канала передачи данных

A.3.2.2.2 UDP-IPObject

Данный элемент устанавливает атрибуты UDP/IP, используемые в ADS-net.

Семантика субэлементов UDP-IPObject -Attributes элемента UDP-IPObject определена в таблице A.10.

Таблица A.10 — Элементы UDP-IPObject -Attributes

Элементы	Описание и семантика
IPADDR	IP адрес, описанный в стандартном формате Internet с точкой
UDP-IP-MCGs	Семантика этого элемента установлена в таблице A.11
HostName	Имя хоста устройства, соответствующее адресу IP
ErrorLog	Список ошибок, возникших на транспортном уровне

Таблица A.11 – Элементы UDP-IP-MCG

Элементы	Описание и семантика
Mcg Number	Номер идентификации многоадресной группы (десятичная нотация: 0 — 255)
PortNumber	Номер порта, используемого многоадресной группой

Более подробные сведения о семантике каждого атрибута см. [4].

A.3.2.3 NetworkManagement (управление сетью)

A.3.2.3.1 Nm-Configuration (конфигурация)

Данный элемент устанавливает атрибуты, используемые для конфигурации сети.

Семантика субэлементов Nm-Configuration-Attributes элемента Nm-Configuration определена в таблице A.12.

Таблица А.12 – Элементы Nm-Configuration-Attributes

Элементы	Описание и семантика
ActiveDataFieldNoList	Список номеров полей данных (десятичная нотация: 0 — 255), используемых в целевой системе
ActiveNodeNoList	Список номеров узлов (десятичная нотация: 0 — 4095), работающих в целевой системе. Данный список создан путем сбора информации сообщений о рабочем состоянии
Active Multicast Group NoList List	Список номеров MCG (десятичная нотация: 0 — 255), используемых в целевой системе

Более подробные сведения о семантике каждого описания см. [4].

A.3.2.3.2 Nm-MessageSelection (выбор Nm-Сообщений)

Данный элемент устанавливает атрибуты, используемые при выборе сообщений.

Семантика субэлементов Nm-MessageSelection-Attributes элемента Nm-MessageSelection определена в таблице А.13.

Таблица А.13 – Элементы Nm-MessageSelection-Attributes

Элементы	Описание и семантика
ActiveProducerTCDSupportedList	Список TCD (десятичная нотация: 1 — 65534) для любого посылаемого сообщения
ActiveConsumerTCDSupportedList	Список TCD (десятичная нотация: 1 — 65534) для любого получаемого сообщения
ActiveProducerMulticastGroupSupportedList	Список номеров MCG (десятичная нотация: 0 — 255), используемых для передачи любого сообщения
ActiveConsumerMulticastGroupSupportedList	Список номеров MCG (десятичная нотация: 0 — 255), используемых для приема любого сообщения

Более подробные сведения о семантике каждого описания см. [4].

A.3.2.3.3 Nm-Performance (Nm-исполнение)

Данный элемент устанавливает атрибуты, используемые для мониторинга характеристик.

Семантика субэлементов Nm-Performance-Attributes элемента Nm-Performance определена в таблице А.14.

Таблица А.14 – Элементы Nm-Performance-Attributes

Элементы	Описание и семантика
MessageTransaction	Число сообщений, посланных и полученных устройством

A.3.2.3.4 Nm-Fault

Данный элемент определяет атрибуты, используемые для мониторинга отказов.

Семантика субэлементов Nm-Fault-Attributes элемента Nm-Fault определена в таблице А.15.

Таблица А.15 — Элементы Nm-Fault-Attributes

Элементы	Описание и семантика
FaultNodeList	Список номеров узлов (десятичная нотация: 0 – 4095), диагностика которых показывает отказ
Not if icationOf Fault Prog ram	Идентификатор аварийного модуля, обнаруженного сообщением о его состоянии

Более подробные сведения о семантике каждого описания см. [4].

A.3.3 ADS-net CommNet Profile.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    ">
<xsd:annotation>
<xsd:documentation>
Communication Profile - ISO 15745 Part 4
</xsd:documentation>
</xsd:annotation>
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP"/>
<xsd:enumeration value="Process"/>
<xsd:enumeration value="InformationExchange"/>
<xsd:enumeration value="Resource"/>
<xsd:enumeration value="Device"/>
```

```
<xsd:enumeration value="CommunicationNetwork"/>
<xsd:enumeration value="Equipment"/>
<xsd:enumeration value="Human"/>
<xsd:enumeration value="Material"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="CSI"/>
<xsd:enumeration value="HCI"/>
<xsd:enumeration value="ISI"/>
<xsd:enumeration value="API"/>
<xsd:enumeration value="CMI"/>
<xsd:enumeration value="ESI"/>
<xsd:enumeration value="FSI"/>
<xsd:enumeration value="MTI"/>
<xsd:enumeration value="SEI"/>
<xsd:enumeration value="USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>
ADS-net Communication Network Profile Template
</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"/>
```

```
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ApplicationLayers"/>
<xsd:element ref="TransportLayers"/>
<xsd:element ref="NetworkManagement" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DataField" maxOccurs="unbounded"/>
<xsd:element ref="MessageSelection" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DataField">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DataField-Attributes"/>
<xsd:element ref="DataField-Operations" minOccurs="0"/>
<xsd:element ref="AliveNotification"/>
<xsd:element ref="ErrorNotification" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DataField-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DFNO" type="xsd:unsignedByte"/>
<xsd:element name="NETADDR" type="xsd:unsignedInt"/>
<xsd:element name="BCADDR" type="xsd:unsignedInt"/>
<xsd:element name="NETMASK" type="xsd:unsignedInt"/>
<xsd:element name="MCG-Table" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DataField-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_Single" minOccurs="0"/>
<xsd:element name="Set_Attribute_Single" minOccurs="0"/>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Set_Attribute_All" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="AliveNotification">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ND-Name" type="xsd:string"/>
<xsd:element name="OS-Name" type="xsd:string"/>
<xsd:element name="TM-Out" type="xsd:nonNegativeInteger"/>
<xsd:element name="Chg-Time" type="xsd:nonNegativeInteger"/>
<xsd:element name="IPADDR" type="xsd:unsignedInt"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ErrorNotification">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="CNT-Mode-Alive" type="xsd:string"/>
<xsd:element name="Mod-Alive-List" type="xsd:string" minOccurs="0"/>
<xsd:element name="CNT-Error" type="xsd:string"/>
<xsd:element name="ERR-Name" type="xsd:string" minOccurs="0"/>
<xsd:element name="ERR-List" type="xsd:string" minOccurs="0"/>
<xsd:element name="Mod-NO" type="xsd:string" minOccurs="0"/>
<xsd:element name="ERR-NO" type="xsd:string" minOccurs="0"/>
<xsd:element name="CNT-Option" type="xsd:string"/>
<xsd:element name="Option-List" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="MessageSelection">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="MessageSelection-Attributes"/>
<xsd:element ref="MessageSelection-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="MessageSelection-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Producer-TCD-List" type="xsd:string" minOccurs="0"/>
<xsd:element name="Consumer-TCD-List" type="xsd:string" minOccurs="0"/>
<xsd:element name="Producer-MCG-List" type="xsd:string" minOccurs="0"/>
<xsd:element name="Consumer-MCG-List" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="MessageSelection-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_Single" minOccurs="0"/>
<xsd:element name="Set_Attribute_Single" minOccurs="0"/>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Set_Attribute_All" minOccurs="0"/>
<xsd:element name="Create" minOccurs="0"/>
<xsd:element name="Delete" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TransportLayers">
```

```
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="EthernetObject" maxOccurs="unbounded"/>
<xsd:element ref="UDP-IPObject" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EthernetObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="EthernetObject-Attributes"/>
<xsd:element ref="EthernetObject-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EthernetObject-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MediaType" type="xsd:string"/>
<xsd:element name="CommRate" type="xsd:unsignedInt"/>
<xsd:element name="Indicators" type="xsd:string" minOccurs="0"/>
<xsd:element name="MACAddress" type="xsd:string"/>
<xsd:element name="ErrorLog" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EthernetObject-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Set_Attribute_All" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UDP-IPObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="UDP-IPObject-Attributes"/>
<xsd:element ref="UDP-IPObject-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UDP-IPObject-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="IPADDR" type="xsd:string"/>
<xsd:element ref="UDP-IP-MCGs" maxOccurs="unbounded"/>
<xsd:element name="HostName" type="xsd:string"/>
<xsd:element name="ErrorLog" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UDP-IP-MCGs">
<xsd:complexType>
<xsd:sequence>
```

```
<xsd:element name="McgNumber" type="xsd:unsignedShort"/>
<xsd:element name="PortNumber" type="xsd:unsignedShort"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UDP-IPObject-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Set_Attribute_All" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="NetworkManagement">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="Nm-Configuration" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="Nm-MessageSelection" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="Nm-Performance" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="Nm-Fault" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-Configuration">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="Nm-Configuration-Attributes"/>
<xsd:element ref="Nm-Configuration-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-Configuration-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ActiveDataFieldNoList" type="xsd:string"/>
<xsd:element name="ActiveNodeNoList" type="xsd:string"/>
<xsd:element name="ActiveMulticastGroupNoList" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-Configuration-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Set_Attribute_All" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-MessageSelection">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="Nm-MessageSelection-Attributes"/>
<xsd:element ref="Nm-MessageSelection-Operations" minOccurs="0"/>
```

```
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-MessageSelection-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ActiveProducerTCDSupportedList" type="xsd:string"/>
<xsd:element name="ActiveConsumerTCDSupportedList" type="xsd:string"/>
<xsd:element name="ActiveProducerMulticastGroupSupportedList" type="xsd:string"/>
<xsd:element name="ActiveConsumerMulticastGroupSupportedList" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-MessageSelection-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_Single" minOccurs="0"/>
<xsd:element name="Set_Attribute_Single" minOccurs="0"/>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Set_Attribute_All" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-Performance">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="Nm-Performance-Attributes" minOccurs="0"/>
<xsd:element ref="Nm-Performance-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-Performance-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MessageTransaction" type="xsd:unsignedInt"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-Performance-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-Fault">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="Nm-Fault-Attributes" minOccurs="0"/>
<xsd:element ref="Nm-Fault-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-Fault-Attributes">
<xsd:complexType>
<xsd:sequence>
```

```
<xsd:element name="FaultNodeList" type="xsd:string"/>
<xsd:element name="NotificationOfFaultProgram" type="xsd:string" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Nm-Fault-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
<xsd:element name="Get-Attributes-All" minOccurs="0"/>
<xsd:element name="Set-Attributes-All" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

Приложение В
(обязательное)
Шаблоны профиля FL-net

В.1 Описание шаблона профиля устройства

В.1.1 Общие положения

XML схема шаблона профиля устройства, определенная в В.1.5, содержит отображение диаграмм класса профиль устройства, показанных в 6.2.1. Помимо отображенных классов и атрибутов эта схема включает в себя дополнительные элементы, позволяющие дать более полное описание требований и возможностей устройств.

В.1.2 DeviceIdentity (идентичность устройства)

Данный элемент определяет атрибуты и операции объекта DeviceIdentity совместно с дополнительной информацией для полной идентификации устройства.

В таблице В.1 приведено описание элементов объекта DeviceType (устройство тип).

Таблица В.1 — Элементы объекта DeviceType

Элементы	Описание и семантика
PC	Программируемый контроллер
NC	Цифровой блок управления
RC	Контроллер робота
computer	Персональный компьютер, компьютер пульта, рабочая станция
SP-*xxx*	xxx – имя устройства, установленное поставщиком

В.1.3 DeviceManager (менеджер устройства)

В.1.3.1 Общие положения

Данный элемент определяет атрибуты и операции объекта DeviceManager совместно с дополнительной информацией по управлению устройством.

B.1.3.2 Объект DeviceIDSpecRev (версия спецификации идентификатора устройства)

В таблице B.2 приведено описание элементов объекта DeviceIdSpecRev согласно определению, приведенному в 6.2.1.3.2.

Таблица B.2 — Элементы объекта DeviceIDSpecRev

Элементы	Описание и семантика
COMVER	Версия DeviceIdentity
Date	Дата определения идентичности

B.1.3.3 Объект CommuServiceManager (менеджер коммуникационного сервиса)

В таблице B.3 приведено описание элементов объекта CommuServiceManager согласно определению, приведенному в 6.2.1.3.3.

Таблица B.3 – Элементы объекта CommuServiceManager

Элементы	Описание и семантика
DM-NodeNumber	Номер узла, используемого в FL-net
DM-ComMemory1Datafield	Поле данных общей области памяти 1, используемое в устройстве
DM-ComMemory2Datafield	Поле данных общей области памяти 2, используемое в устройстве
DM-MessageServiceList	Список сервисов сообщений, используемых в устройстве

B.1.3.4 Объект DeviceState (состояние устройства)

В таблице B.4 приведено описание элементов объекта DeviceState согласно определению, приведенному в 6.2.1.3.4.

Таблица В.4 – Элементы объекта DeviceState

Элементы	Описание и семантика
RUN-STOP	Индикация состояния устройства (работа или останов)
Normal	Индикация нормального состояния устройства
Warning	Индикация аварийного состояния устройства – устройство еще работает
Alarm	Индикация аварийного состояния устройства – устройство не работает
InRingState	Индикация состояния коммуникаций устройства

В.1.4 ApplicationProcess (прикладной процесс)

Данный элемент устанавливает атрибуты и операции объекта ApplicationProcess совместно с дополнительными элементами.

В таблице В.5 приведено описание элементов объекта PlantName согласно определению, приведенному в 6.2.1.5.

Таблица В.5 – Элементы объекта PlantName

Элементы	Описание и семантика
PlantName	Имя установки или процесса

В.1.5 Схемы XML шаблона профиля устройства FL-net

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://gost.e-ecolog.ru                    ">
<xsd:annotation>
<xsd:documentation>
Device Profile - ISO 15745 Part 4
</xsd:documentation>
</xsd:annotation>
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP"/>
<xsd:enumeration value="Process"/>
<xsd:enumeration value="InformationExchange"/>
<xsd:enumeration value="Resource"/>
<xsd:enumeration value="Device"/>
<xsd:enumeration value="CommunicationNetwork"/>
<xsd:enumeration value="Equipment"/>
<xsd:enumeration value="Human"/>
<xsd:enumeration value="Material"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
```

```
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="CSI"/>
<xsd:enumeration value="HCI"/>
<xsd:enumeration value="ISI"/>
<xsd:enumeration value="API"/>
<xsd:enumeration value="CMI"/>
<xsd:enumeration value="ESI"/>
<xsd:enumeration value="FSI"/>
<xsd:enumeration value="MTI"/>
<xsd:enumeration value="SEI"/>
<xsd:enumeration value="USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>
FL-net Device Profile Template
</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DeviceIdentity" minOccurs="0"/>
<xsd:element ref="DeviceManager" minOccurs="0"/>
<xsd:element ref="DeviceFunction" maxOccurs="unbounded"/>
<xsd:element ref="ApplicationProcess" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceIdentity">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DeviceIdentity-Attributes"/>
<xsd:element ref="DeviceIdentity-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceIdentity-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="VendorCode" type="xsd:string"/>
<xsd:element name="ProductName" type="xsd:string"/>
<xsd:element name="ProductCode" type="xsd:string"/>
<xsd:element name="ProductRevision" type="xsd:string"/>
<xsd:element name="DeviceType" minOccurs="0">
<xsd:simpleType>
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="PC"/>
<xsd:enumeration value="NC"/>
<xsd:enumeration value="RC"/>
<xsd:enumeration value="computer"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:pattern value="SP-\*.{0,32}\*"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceIdentity-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Get_Attribute_Single" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceManager">
```

```
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DeviceManager-Attributes"/>
<xsd:element ref="DeviceManager-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceManager-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DeviceIDSpecRev">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="COMVER">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="1"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="Date" type="xsd:date"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="CommuServiceManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DM-NodeNumber">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="1"/>
<xsd:maxInclusive value="255"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="DM-ComMemory1Datafield">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="0"/>
<xsd:maxInclusive value="63"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="DM-ComMemory2Datafield">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="0"/>
<xsd:maxInclusive value="8195"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="DM-MessageServiceList" type="xsd:unsignedShort"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceState" minOccurs="0">
<xsd:complexType>
```

```
<xsd:sequence>
<xsd:element name="RUN-STOP" type="xsd:boolean"/>
<xsd:element name="Normal" type="xsd:boolean"/>
<xsd:element name="Warning" type="xsd:boolean"/>
<xsd:element name="Alarm" type="xsd:boolean"/>
<xsd:element name="InRingState" type="xsd:boolean"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceManager-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Get_Attribute_Single" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceFunction">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationProcess">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ApplicationProcess-Attributes"/>
<xsd:element ref="ApplicationProcess-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationProcess-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="PlantName" type="xsd:string"/>
<xsd:choice maxOccurs="unbounded">
<xsd:element ref="externalelement" minOccurs="0" maxOccurs="unbounded"/>
</xsd:choice>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="externalelement">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationProcess-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
```

```
<xsd:element name="Get-Attribute-All" minOccurs="0"/>
<xsd:element name="Set-Attribute-All" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

B.2 Описание шаблона профиля коммуникационной сети FL-net

B.2.1 Общие положения

XML схема шаблона профиля коммуникационной сети, определенная в B.2.4.5, содержит отображение диаграмм класса профиль коммуникационной сети, показанных на рисунке 7. Помимо отображенных классов и атрибутов она содержит дополнительные элементы, позволяющие дать полное описание требований и возможностей коммуникационной сети.

B.2.2 ApplicationLayers (уровени приложения)

B.2.2.1 Объект ComMemoryInterface

В таблице B.6 приведено описание элементов объекта ComMemoryInterface согласно определениям, приведенному в 6.2.2.2.2. Субэлементы ComMemory1AllocationList и ComMemory2AllocationList предоставляют информацию о распределении памяти. Их атрибуты описаны в таблице B.7.

Таблица B.6 – Элементы ComMemoryInterface

Элементы	Описание и семантика
Com Memory1AllocationList	Распределение общей области памяти 1
ComMemory2AllocationList	Распределение общей области памяти 2
NodeNumber	Номер узла
DataAvailability	Информация о наличии данных в общей памяти
DataConsistency	Совместимость данных
Cyclic	Циклический обмен данными

Таблица В.7 – Атрибуты ComMemoryAllocationList

Элементы	Описание и семантика
Size	Размер данных
FieldNumber	Номер поля данных в общей памяти

В.2.2.2 Объект MessageService (сервис сообщений)

В таблице В.8 приведено описание элементов объекта MessageService согласно определению, приведенном у в 6.2.2.2.3.

Таблица В.8 – Элементы объекта MessageService

Элементы	Описание и семантика
NodeNumberList	Список номеров узлов
TCDList	Список TCD

В.2.2.3 Объект ErrorNotification (уведомление об ошибке)

В таблице В.9 приведено описание элементов объекта ErrorNotification согласно определению, приведенному в 6.2.2.2.4.

Таблица В.9 – Элементы объекта ErrorNotification

Элементы	Описание и семантика
ActiveNodeNumberList	Список номеров активных узлов
NodeStatus	Статус узла

В.2.3 TransportLayers (транспортные уровни)

В.2.3.1 Объект EthernetBasedObject

В таблице В.10 приведено описание элементов объекта EthernetBasedObject согласно определению, приведенному в 6.2.2.3.2.

Таблица В.10 – Элементы объекта EthernetBasedObject

Элементы	Описание и семантика
MediaType	Тип среды Ethernet (например, 10Base-5, 10Base-T, 100Base-TX)
CommRate	Скорость коммуникаций, зависящая от среды Ethernet (например, 10 Мегабит/с, 100 Мегабит/с)
Indicators	Состояние среды Ethernet
MACAddress	MAC адрес
ErrorLog	Список ошибок, возникших на физическом уровне и уровне канала передачи данных

В.2.3.2 Объект UDP-IPObject

В таблице В.11 приведено описание элементов объекта UDP-IPObject согласно определению, приведенному в 6.2.2.3.3.

Таблица В.11 – Элементы объекта UDP-IPObject

Элементы	Описание и семантика
IPADDR	Адрес IP
PortNumber	Номер порта, определенный в FL-net
HostName	Имя хоста устройства, соответствующее адресу IP
ErrorLog	Список ошибок, возникших на транспортном уровне

В.2.4 NetworkManagement (управление сетью)

В.2.4.1 Объект Configuration (конфигурация)

В таблице В.12 приведено описание элементов объекта Configuration согласно определению, приведенному в 6.2.2.4.2.

Таблица В.12 – Элементы объекта Configuration

Элементы	Описание и семантика
SpecificationMajorversion	Большая версия спецификации FL-net
SpecificationMinorversion	Малая версия спецификации FL-net
InRingManager	Объект Manager, ассоциированный с участием сети
ConnectionManager	Объект Manager, ассоциированный с соединением
Indicators	Состояние FL-net

В.2.4.2 Объект ServiceSelection (выбор сервиса)

В таблице В.13 приведено описание элементов объекта ServiceSelection согласно определению, приведенному в 6.2.2.4.3.

Таблица В.13 – Элементы объекта ServiceSelection

Элементы	Описание и семантика
NM-MessageTypeList	Список типов сообщений
NM-TCDList	Список TCD для посылаемых сообщений
NM-VendorCodeList	Список кодов продавца

В.2.4.3 Объект PerformanceManager (менеджер исполнения)

В таблице В.14 приведено описание элементов объекта PerformanceManager согласно определению, приведенному в 6.2.2.4.4.

Таблица В.14 – Элементы объекта PerformanceManager

Элементы	Описание и семантика
TokenMonitoringTime	Максимальное время перед тем, как следующий узел воспроизведет маркер
TokenHoldTime	Максимальное время, в течение которого узлу разрешается удерживать маркер
MinimumFrameInterval	Минимальный интервал времени между передачей фреймов
RefreshCycleAvailableTime	Время восстановления цикла
MessageTransmissionSchedule	Расписание передачи сообщений

B.2.4.4 Объект FaultManager (менеджер ошибок)

В таблице B.15 приведено описание элементов объекта FaultManager согласно определению, приведенному в 6.2.2.4.5.

Таблица B.15 – Элементы объекта FaultManager

Элементы	Описание и семантика
FaultNodeList	Список номеров активных узлов
ErrorName	Имя ошибки, которое присвоено каждой ошибке
ErrorLog	Список ошибок, возникших в коммуникациях FL-net
ComMemoryAreaOverlap	Информация о перекрытии распределения общей памяти
NodeNurnberConflict	Информация о конфликтах номеров узлов

B.2.4.5 Схема XML: FL-net_CommNet_Profile.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://gost.e-ecolog.ru                    >
<xsd:annotation>
<xsd:documentation>
Communication Profile - ISO 15745 Part 4
</xsd:documentation>
</xsd:annotation>
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP"/>
<xsd:enumeration value="Process"/>
<xsd:enumeration value="InformationExchange"/>
<xsd:enumeration value="Resource"/>
<xsd:enumeration value="Device"/>
<xsd:enumeration value="CommunicationNetwork"/>
<xsd:enumeration value="Equipment"/>
<xsd:enumeration value="Human"/>
<xsd:enumeration value="Material"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
```

```
<xsd:enumeration value="CSI"/>
<xsd:enumeration value="HCI"/>
<xsd:enumeration value="ISI"/>
<xsd:enumeration value="API"/>
<xsd:enumeration value="CMI"/>
<xsd:enumeration value="ESI"/>
<xsd:enumeration value="FSI"/>
<xsd:enumeration value="MTI"/>
<xsd:enumeration value="SEI"/>
<xsd:enumeration value="USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>
FL-net Communication Network Profile Template
</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ApplicationLayers"/>
<xsd:element ref="TransportLayers"/>
<xsd:element ref="NetworkManagement" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="ApplicationLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ComMemoryInterface" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="MessageService" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="ErrorNotification" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ComMemoryInterface">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ComMemoryInterface-Attributes"/>
<xsd:element ref="ComMemoryInterface-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ComMemoryInterface-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ComMemory1AllocationList" minOccurs="0"
maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="Size">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="0"/>
<xsd:maxInclusive value="64"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="FieldNumber">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="0"/>
<xsd:maxInclusive value="63"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="ComMemory2AllocationList" minOccurs="0"
maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="Size">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="0"/>
<xsd:maxInclusive value="8196"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="FieldNumber">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="0"/>
```

```
<xsd:maxInclusive value="8195"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="NodeNumber" type="xsd:string" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="DataAvailability" type="xsd:boolean" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="DataConsistency" type="xsd:boolean" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="Cyclic" type="xsd:boolean" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ComMemoryInterface-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_Single" minOccurs="0"/>
<xsd:element name="Set_Attribute_Single" minOccurs="0"/>
<xsd:element name="Apply_Attributes" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="MessageService">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="MessageService-Attributes"/>
<xsd:element ref="MessageService-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="MessageService-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="NodeNumberList" type="xsd:unsignedShort" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="TCDList" type="xsd:unsignedInt" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="MessageService-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Get_Attribute_Single" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ErrorNotification">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ActiveNodeNumberList" type="xsd:unsignedInt" minOccurs="0"
maxOccurs="unbounded"/>
```

```
<xsd:element name="NodeStatus" type="xsd:unsignedInt" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TransportLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="EthernetBasedObject" maxOccurs="unbounded"/>
<xsd:element ref="UDP-IPObject" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EthernetBasedObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="EthernetBasedObject-Attributes"/>
<xsd:element ref="EthernetBasedObject-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EthernetBasedObject-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MediaType" type="xsd:string"/>
<xsd:element name="CommRate" type="xsd:unsignedInt"/>
<xsd:element ref="Indicators" type="xsd:string" minOccurs="0"/>
<xsd:element name="MACAddress" type="xsd:string"/>
<xsd:element name="ErrorLog" type="xsd:unsignedInt" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EthernetBasedObject-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
<xsd:element name="Get-Attributes-All" minOccurs="0"/>
<xsd:element name="Clear" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UDP-IPObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="UDP-IPObject-Attributes"/>
<xsd:element ref="UDP-IPObject-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UDP-IPObject-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="IPADDR" type="xsd:string"/>
<xsd:element name="PortNumber" type="xsd:unsignedShort" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="HostName" type="xsd:string"/>
```

```
<xsd:element name="ErrorLog" type="xsd:unsignedInt" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UDP-IPObject-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
<xsd:element name="Get-Attributes-All" minOccurs="0"/>
<xsd:element name="Clear" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="NetworkManagement">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="Configuration" maxOccurs="unbounded"/>
<xsd:element ref="ServiceSelection" maxOccurs="unbounded"/>
<xsd:element ref="PerformanceManager" maxOccurs="unbounded"/>
<xsd:element ref="FaultManager" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Configuration">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="Configuration-Attributes"/>
<xsd:element ref="Configuration-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Configuration-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="SpecificationMajorVersion" minOccurs="0"/>
<xsd:element ref="SpecificationMinorVersion" minOccurs="0"/>
<xsd:element ref="InRingManager" minOccurs="0"/>
<xsd:element ref="ConnectionManager" minOccurs="0"/>
<xsd:element ref="Indicators" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="SpecificationMajorVersion">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="1"/>
<xsd:maxInclusive value="15"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="SpecificationMinorVersion">
<xsd:simpleType>
<xsd:restriction base="xsd:integer">
<xsd:minInclusive value="0"/>
<xsd:maxInclusive value="15"/>
</xsd:restriction>
```

```
</xsd:simpleType>
</xsd:element>
<xsd:element name="InRingState" type="xsd:string"/>
<xsd:element name="ConnectionManager" type="xsd:string"/>
<xsd:element name="Indicators" type="xsd:string"/>
<xsd:element name="Configuration-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
<xsd:element name="Get-Attributes-All" minOccurs="0"/>
<xsd:element name="Set-Attributes-All" minOccurs="0"/>
<xsd:element name="Clear" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ServiceSelection">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ServiceSelection-Attributes"/>
<xsd:element ref="ServiceSelection-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ServiceSelection-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="NM-MessageTypeList" type="xsd:unsignedShort"
maxOccurs="unbounded"/>
<xsd:element name="NM-TCDList" type="xsd:unsignedInt" maxOccurs="unbounded"/>
<xsd:element name="NM-VendorCodeList" type="xsd:string" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ServiceSelection-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All" minOccurs="0"/>
<xsd:element name="Get_Attribute_Single" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="PerformanceManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="PerformanceManager-Attributes"/>
<xsd:element ref="PerformanceManager-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="PerformanceManager-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="TokenMonitoringTime" type="xsd:unsignedShort" minOccurs="0"/>
<xsd:element name="TokenHoldTime" type="xsd:unsignedShort" minOccurs="0"/>
<xsd:element name="MinimumFrameInterval" minOccurs="0"/>
```

```
<xsd:element name="RefreshCycleAvailableTime" type="xsd:unsignedShort" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="MessageTransmissionSchedule" type="xsd:unsignedShort"
minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="PerformanceManager-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
<xsd:element name="Get-Attributes-All" minOccurs="0"/>
<xsd:element name="Set-Attributes-All" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="FaultManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="FaultManager-Attributes"/>
<xsd:element ref="FaultManager-Operations" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="FaultManager-Attributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="FaultNodeList" type="xsd:unsignedShort" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="ErrorName" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element name="ErrorLog" type="xsd:unsignedInt" minOccurs="0"/>
<xsd:element name="ComMemoryAreaOverlap" type="xsd:boolean" minOccurs="0"
maxOccurs="unbounded"/>
<xsd:element name="NodeNumberConflict" type="xsd:boolean" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="FaultManager-Operations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get-Attribute-Single" minOccurs="0"/>
<xsd:element name="Set-Attribute-Single" minOccurs="0"/>
<xsd:element name="Get-Attributes-All" minOccurs="0"/>
<xsd:element name="Set-Attributes-All" minOccurs="0"/>
<xsd:element name="Clear" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

Приложение С
(обязательное)
Шаблоны профиля EtherNet/IP

С.1 Общие положения

Верхние уровни сети EtherNet/IP основаны на общем промышленном протоколе CIP. Этот протокол моделирует все сущности коммуникаций и приложений в виде объектов. Специальные сервисы CIP по запросу сообщений должны выполняться на соответствующих экземплярах объектов (или их атрибутах). Эта схема предоставляет явный доступ ко всем данным параметров конфигурации, состояния и рабочего цикла в узле. В то же время соединения ввода/вывода допускают прямой обмен с базой данных ввода/вывода без промежуточной обработки. В обоих случаях все ссылки на данные внутри устройства указываются с помощью путей CIP, т.е. потока октетной строки, которая определяет экземпляр объекта приложения, атрибут и/или конечную точку соединения.

Для дистанционной конфигурации устройства доступны многие опции с помощью интерфейса коммуникаций CIP, включая следующее:

— сохраненную информацию об устройстве в печатном или электронном формате;

— выделенные объекты параметров (Parameter Objects), которые предоставляют известный общедоступный интерфейс для индивидуальных значений данных конфигурации/параметров и могут также вводить дополнительную информацию о конфигурации, например дескриптивный текст, тип данных, предельные значения данных и значения по умолчанию;

— выделенное объединение конфигураций (Configuration Assembly), что допускает групповое сохранение и загрузку данных конфигурации путем объединения в группы отдельных значений данных по конфигурации/параметрам;

— комбинации указанных выше методов.

Инструменты конфигурации, имеющиеся в настоящее время в основанных на CIP устройствах, используют специально форматированный

файл ASCII, называемый электронным бланком данных (EDS), который предоставляет следующее:

— информацию, необходимую для идентификации присоединенного устройства;

— описание данных устройства, которые могут быть доступны через сеть (например, конфигурируемые параметры);

— описание возможностей коммуникации, поддерживаемых устройством (например, соединения);

— дополнительную информацию поставщика.

EDS допускает применение инструмента конфигурации, автоматически выполняющего процесс конфигурации устройства. Требования EDS обеспечивают открытый, последовательный и совместимый подход к выполнению конфигурации устройства в среде CIP.

Информация EDS в высокой степени аналогична информации, требующейся в профилях как устройств, так и коммуникационных сетей, в связи с чем в следующих подразделах определяется формат для:

— шаблонов профиля коммуникационной сети и устройства согласно определению, приведенному в ИСО 15745-1;

— инкапсуляции ранее принятых файлов EDS в шаблоны комплекса стандартов ИСО 15745 («оболочки»);

— ранее принятого EDS, включая общую семантическую информацию.

Примечание – EtherNet/IP EDS некоторого устройства может быть получен из содержания соответствующих файлов XML профиля устройства и коммуникационной сети, используя бланки подходящего стиля.

C.2 Описание шаблона профиля устройства

C.2.1 Описание шаблона профиля устройства на основе XML

C.2.1.1 Общие положения

Файлы XML профиля устройства должны соответствовать XML схеме профиля устройства, установленной в C.2.1.3.3.

Содержание этой XML схемы выводится из диаграмм класса профиль устройства, показанных в 6.3.1, расширенного с помощью дополнительных

элементов, что позволяет дать полное описание требований и возможностей устройства.

С.2.1.2 Семантика элементов XML схемы

С.2.1.2.1 ProfileBody (тело профиля)

Этот основной элемент ассоциируется с набором атрибутов, которые предоставляют дополнительную информацию относительно файла профиля.

Семантика этих атрибутов установлена в С.4.1.4.2.

С.2.1.2.2 DeviceIdentity (идентичность устройства)

Этот элемент устанавливает поддерживаемые атрибуты и операции сущности Identity Object (см. МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2)) совместно с дополнительной информацией для полной идентификации устройства. Если это целесообразно, он также показывает фактические значения атрибутов сущности.

Семантика субэлементов DeviceIdentity_InstanceAttributes элемента DeviceIdentity определена в таблице С.1.

Таблица С.1 – Элементы DeviceIdentity_InstanceAttributes

Элементы XML схемы	Атрибуты объекта	Семантика
SpecificationConformance	Нет	Строка, определяющая ссылочную версию спецификаций EtherNet/IP
VendCode, ProdType, ProdCode, ProdRevision	Да	См. С.4.1.4.3
VendName, ProdTypeStr, ProdName, Catalog, Icon, ExcludeFromAdapterRackConnection	Нет	См. С.4.1.4.3
Status, SerialNumber	Да	Не применяется
State, ConfigurationConsistencyValue, HeartbeatInterval	Да	Не применяется

Окончание таблицы С.1

Элементы XML схемы	Атрибуты объекта	Семантика
DeviceClassification	Нет	См. С.4.1.4.4 и С.4.2.2.1

С.2.1.2.3 DeviceManager (менеджер устройства)

Данный элемент определяет поддерживаемые атрибуты и операции класса объекта Identity Object (см. МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2)) совместно с дополнительной информацией по управлению устройством. Когда это целесообразно, он также показывает фактические значения атрибутов сущности.

Семантика Модульного (Modular) субэлемента элемента DeviceManager установлена в С.4.1.5.2.

С.2.1.2.4 DeviceFunction (функция устройства)

Содержание этого элемента в настоящем стандарте подробно не рассматривается.

С.2.1.2.5 ApplicationProcess (прикладной процесс)

С.2.1.2.5.1 Assembly (Сборка)

Данный элемент определяет поддерживаемые атрибуты и операции класса и экземпляра объекта Assembly Object (см. МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2)), совместно с описанием отдельных экземпляров.

Семантика субэлементов Assem, ProxyAssem и ProxiedAssem элемента Assembly определена в С.4.1.4.8 и С.4.1.5.3.2.

С.2.1.2.5.2 Parameter (параметр)

Данный элемент устанавливает поддерживаемые атрибуты и операции класса и экземпляра объекта Parameter Object совместно с описанием отдельных экземпляров.

Семантика субэлемента Parameter_ClassAttributes элемента Parameter определена в С.4.1.4.5.

Семантика субэлементов Param, ProxyParam и ProxiedParam элемента Parameter определена в С.4.1.4.6 и С.4.1.5.3.1.

C.2.1.2.5.3 ParameterGroup (группа параметров)

Данный элемент устанавливает группы связанных параметров для специальных целей.

Семантика субэлемента Group элемента ParameterGroup определена в C.4.1.4.7.

C.2.1.3 Схемы XML

C.2.1.3.1 MasterTemplateTypes.xsd

Примечание – Данная XML схема содержит все стили, определенные как часть шаблона ведущего устройства по ИСО 15745-1.

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                                    ">
<!-- Target namespaces are not specified in this master template -->
<xsd:annotation>
<xsd:documentation>* HEADER DATA TYPES *</xsd:documentation>
</xsd:annotation>
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP" />
<xsd:enumeration value="Process" />
<xsd:enumeration value="InformationExchange" />
<xsd:enumeration value="Resource" />
<xsd:enumeration value="Device" />
<xsd:enumeration value="CommunicationNetwork" />
<xsd:enumeration value="Equipment" />
<xsd:enumeration value="Human" />
<xsd:enumeration value="Material" />
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger" />
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger" />
<xsd:element name="ProfileTechnology" type="xsd:string" />
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="CSI" />
<xsd:enumeration value="HCI" />
<xsd:enumeration value="ISI" />
<xsd:enumeration value="API" />
<xsd:enumeration value="CMI" />
<xsd:enumeration value="ESI" />
<xsd:enumeration value="FSI" />
```

```
<xsd:enumeration value="MTI" />
<xsd:enumeration value="SEI" />
<xsd:enumeration value="USI" />
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4" />
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:annotation>
<xsd:documentation>* ISO 15745 DEFINED DATA TYPES *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileHandle_DataType">
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string" />
<xsd:element name="ProfileRevision" type="xsd:string" />
<xsd:element name="ProfileLocation" type="xsd:anyURI" minOccurs="0" maxOccurs="1" />
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

C.2.1.3.2 CIPDataTypes.xsd

Примечание – Данная XML схема определяет пункты XML схемы (например, типы данных, типы элементов, группы атрибутов), используемые в других XML схемах.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                                    ">
<!-- Target namespaces are not specified in this master template -->
<xsd:annotation>
<xsd:documentation>* CIP DATA TYPES *</xsd:documentation>
</xsd:annotation>
<xsd:simpleType name="dt_USINT">
<xsd:restriction base="xsd:unsignedByte"/>
</xsd:simpleType>
<xsd:simpleType name="dt_UINT">
<xsd:restriction base="xsd:unsignedShort"/>
</xsd:simpleType>
<xsd:simpleType name="dt_UDINT">
<xsd:restriction base="xsd:unsignedInt"/>
</xsd:simpleType>
<xsd:simpleType name="dt_ULINT">
<xsd:restriction base="xsd:unsignedLong"/>
</xsd:simpleType>
<xsd:simpleType name="dt_SINT">
<xsd:restriction base="xsd:byte"/>
</xsd:simpleType>
<xsd:simpleType name="dt_INT">
<xsd:restriction base="xsd:short"/>
</xsd:simpleType>
<xsd:simpleType name="dt_DINT">
<xsd:restriction base="xsd:int"/>
</xsd:simpleType>
```

```
<xsd:simpleType name="dt_LINT">
<xsd:restriction base="xsd:long"/>
</xsd:simpleType>
<xsd:simpleType name="dt_BYTE">
<xsd:restriction base="xsd:hexBinary">
<xsd:maxLength value="1"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_WORD">
<xsd:restriction base="xsd:hexBinary">
<xsd:maxLength value="2"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_DWORD">
<xsd:restriction base="xsd:hexBinary">
<xsd:maxLength value="4"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_LWORD">
<xsd:restriction base="xsd:hexBinary">
<xsd:maxLength value="8"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_REAL">
<xsd:restriction base="xsd:float"/>
</xsd:simpleType>
<xsd:simpleType name="dt_LREAL">
<xsd:restriction base="xsd:double"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_Char_Array">
<xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EPATH">
<xsd:list itemType="et_EPATH_item"/>
</xsd:simpleType>
<xsd:simpleType name="dt_STRINGI">
<xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_Date">
<xsd:restriction base="xsd:date"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_Time_Of_Day">
<xsd:restriction base="xsd:time"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_Revision">
<xsd:restriction base="xsd:string">
<xsd:pattern value="[0-9]\.[1-9]|[1-9]\.[0-9]|[1-9]\.[1-9]"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_URL">
<xsd:restriction base="xsd:anyURI">
<xsd:pattern value="http://.*"/>
<xsd:pattern value="ftp://.*"/>
<xsd:pattern value=".*"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="at_AccessType_OptionalGet">
```

```
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="None"/>
<xsd:enumeration value="Get"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="at_AccessType_OptionalSet">
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="None"/>
<xsd:enumeration value="Get"/>
<xsd:enumeration value="Set"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="at_AccessType_Mandatory">
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="Get"/>
<xsd:enumeration value="Set"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="et_VendorSpecificKeyword">
<xsd:restriction base="xsd:string">
<xsd:pattern value="[1-9][0-9]{0,4}_([A-Z]|[a-z]|[0-9])([A-Z]|[a-z]|[0-9]|[_])*"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="et_EPATH_item">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:pattern value="([0-9]|[a-f]|[A-F]){2}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="SLOT"/>
<xsd:enumeration value="SLOT_MINUS_ONE"/>
<xsd:enumeration value="SYMBOL_ANSI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:pattern value="Param[1-9][0-9]{0,4}"/>
<xsd:pattern value="\[Param[1-9][0-9]{0,4}\]"/>
<xsd:pattern value="ProxyParam[1-9][0-9]{0,4}"/>
<xsd:pattern value="\[ProxyParam[1-9][0-9]{0,4}\]"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:simpleType name="et_ParamReference">
<xsd:restriction base="xsd:NMTOKEN">
<xsd:pattern value="Param[1-9][0-9]{0,4}([:][0-9]{1,2})*"/>
<xsd:pattern value="ProxyParam[1-9][0-9]{0,4}([:][0-9]{1,2})*"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="et_AssemReference">
<xsd:restriction base="xsd:NMTOKEN">
<xsd:pattern value="Assem[1-9][0-9]{0,4}"/>
<xsd:pattern value="ProxyAssem[1-9][0-9]{0,4}"/>
```

```
</xsd:restriction>
</xsd:simpleType>
<xsd:attributeGroup name="ag_FileDescription">
<xsd:attribute name="DescText" type="dt_EDS_Char_Array" use="required"/>
<xsd:attribute name="CreateDate" type="dt_EDS_Date" use="required"/>
<xsd:attribute name="CreateTime" type="dt_EDS_Time_Of_Day" use="required"/>
<xsd:attribute name="ModDate" type="dt_EDS_Date" use="optional"/>
<xsd:attribute name="ModTime" type="dt_EDS_Time_Of_Day" use="optional"/>
<xsd:attribute name="Revision" type="dt_EDS_Revision" use="required"/>
<xsd:attribute name="HomeURL" type="dt_EDS_URL" use="optional"/>
<xsd:attribute name="SpecificationConformance" type="dt_EDS_Char_Array"
use="required"/>
</xsd:attributeGroup>
</xsd:schema>
```

C.2.1.3.3 CIP_Device_Profile.xsd

Примечание – Данная XML схема включает в себя файлы «MasterTemplateTypes.xsd» (см. С.2.1.3.1) и «CIPDataTypes.xsd» (см. С.2.1.3.2).

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://gost.e-ecolog.ru                    MLSchema"
elementFormDefault="qualified">
<!-- Target namespaces are not specified in this master template -->
<xsd:redefine schemaLocation="MasterTemplateTypes.xsd">
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:complexContent>
<xsd:restriction base="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string" fixed="CIP"/>
</xsd:sequence>
</xsd:restriction>
</xsd:complexContent>
</xsd:complexType>
</xsd:redefine>
<xsd:include schemaLocation="CIPDataTypes.xsd"/>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* HEADER SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType" fixed="Device"/>
```

```
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* BODY SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DeviceIdentity"/>
<xsd:element ref="DeviceManager" minOccurs="0"/>
<xsd:element ref="DeviceFunction" maxOccurs="unbounded"/>
<xsd:element ref="ApplicationProcess" minOccurs="0"/>
<xsd:element name="ExternalProfileHandle" type="ProfileHandle_DataType" minOccurs="0"/>
</xsd:sequence>
<xsd:attributeGroup ref="ag_FileDescription"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceIdentity">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DeviceIdentity_InstanceAttributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="VendCode">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="dt_UINT">
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="VendName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="dt_EDS_Char_Array"/>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="SpecificationConformance" type="dt_EDS_Char_Array"minOccurs="0"/>
<xsd:element name="ProdType">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="dt_UINT">
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"use="required"
fixed="Get"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="ProdTypeStr">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="dt_EDS_Char_Array"/>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProdCode">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="dt_UINT">
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProdRevision">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MajRev" type="dt_USINT"/>
<xsd:element name="MinRev" type="dt_USINT"/>
</xsd:sequence>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Status" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="SerialNumber" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProdName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="State" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConfigurationConsistencyValue" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"use="required"/>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="HeartbeatInterval" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet"use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Catalog" type="xsd:string" minOccurs="0"/>
<xsd:element name="Icon" type="xsd:string" minOccurs="0"/>
<xsd:element name="ExcludeFromAdapterRackConnection" type="xsd:string"
minOccurs="0"/>
<xsd:element name="DeviceClassification" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Class" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MainClass">
<xsd:simpleType>
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="ControlNet"/>
<xsd:enumeration value="DeviceNet"/>
<xsd:enumeration value="EtherNetIP"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="et_VendorSpecificKeyword"/>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
</xsd:element>
<xsd:element name="SubClass" type="xsd:NMTOKEN" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="Class[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceIdentity_InstanceOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
```

```
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Reset">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DeviceIdentity_ClassAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxInstance">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxIDClassAttributes">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"use="required"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxIDInstanceAttributes">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"use="required"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceIdentity_ClassOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Reset">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Find_Next_Object_Instance">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Modular" minOccurs="0">
<xsd:complexType>
<xsd:choice>
<xsd:element name="Chassis">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DefineSlotsInRack" type="dt_UINT"/>
<xsd:element name="SlotDisplayRule" type="et_ParamReference"
minOccurs="0"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Module">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Width" type="dt_UINT"/>
<xsd:element name="Rack" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="VendCode" type="dt_UINT"/>
<xsd:element name="ProdType" type="dt_UINT"/>
<xsd:element name="ProdCode" type="dt_UINT"/>
<xsd:element name="MajRev" type="dt_USINT"/>
<xsd:element name="MinRev" type="dt_USINT"/>
<xsd:element name="LegalSlot" type="dt_UINT"maxOccurs="unbounded"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
```

```
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="Rack[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="ExternalID" type="dt_EPATH" minOccurs="0"/>
<xsd:element name="GenericID" type="dt_EPATH" minOccurs="0"/>
<xsd:element name="ExternIDExactMatch" minOccurs="0">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="Yes"/>
<xsd:enumeration value="No"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="Query" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Path" type="dt_EPATH"/>
<xsd:element name="Service" type="dt_USINT"/>
<xsd:element name="Size">
<xsd:simpleType>
<xsd:restriction base="dt_USINT">
<xsd:minInclusive value="1"/>
<xsd:maxInclusive value="16"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="ExternalID" type="dt_EPATH"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceFunction">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationProcess">
<xsd:complexType>
<xsd:sequence>
```

```
<xsd:element name="Parameter" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Parameter_Class" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Parameter_ClassAttributes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxInstance">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ParameterClassDescriptor">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConfigurationAssemblyInstance">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="NativeLanguage" minOccurs="0">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalSet" use="required"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Parameter_ClassOperations" minOccurs="0">
<xsd:complexType>
```

```
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Reset">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Set_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Restore">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Save">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Parameter_Instance" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Parameter_InstanceAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ParameterValue">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Set"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="LinkPathSize">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Set"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="LinkPath">
```

```
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Set"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ParamDescriptor">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="DataType">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="DataSize">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ParameterName">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="UnitsString">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="HelpString">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MinimumValue">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaximumValue">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="DefaultValue">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="ScalingMultiplier">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ScalingDivider">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ScalingBase">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ScalingOffset">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MultiplierLink">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="DivisorLink">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="BaseLink">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="OffsetLink">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="DecimalPrecision">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
```

```
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Parameter_InstanceOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Set_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Enum_String">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Param" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_ParamType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="Param[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProxyParam" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_ProxyParamType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="ProxyParam[1-9][0-9]{0,4}"/>
</xsd:restriction>
```

```
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProxiedParam" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_ParamType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="ProxiedParam[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Assembly" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Assembly_Class" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Assembly_ClassAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxInstance">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Assembly_ClassOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Create">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="Delete">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Assembly_Instance" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Assembly_InstanceAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="NumberOfMembers">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="AssemblyMemberList">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalSet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="AssemblyData">
<xsd:complexType>
<xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Set"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Assembly_InstanceOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Delete">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="Set_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Member">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Set_Member">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Insert_Member">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Remove_Member">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Assem" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_AssemType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="Assem[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProxyAssem" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_AssemType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="ProxyAssem[1-9][0-9]{0,4}"/>
```

```
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProxiedAssem" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_AssemType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="ProxiedAssem[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ParameterGroup" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Group" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="NameString" type="dt_EDS_Char_Array"/>
<xsd:element name="NumberOfMembers" type="dt_UINT"/>
<xsd:choice maxOccurs="unbounded">
<xsd:element name="ParameterRef" type="dt_UINT"/>
<xsd:element name="VariantRef" type="xsd:NMTOKEN"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="Group[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:attribute name="SupportedService" use="required">
```

```
<xsd:simpleType>
<xsd:restriction base="xsd:boolean">
<xsd:pattern value="true|false"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:complexType name="et_ParamType">
<xsd:sequence>
<xsd:element name="LinkPathSize" type="dt_USINT" minOccurs="0"/>
<xsd:element name="LinkPath" type="dt_EPATH" minOccurs="0"/>
<xsd:element name="ParamDescriptor" type="dt_WORD"/>
<xsd:element name="DataType">
<xsd:simpleType>
<xsd:union memberTypes="dt_USINT dt_EPATH"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="DataSize" type="dt_USINT"/>
<xsd:element name="ParameterName" type="dt_EDS_Char_Array"/>
<xsd:element name="UnitsString" type="dt_EDS_Char_Array"/>
<xsd:element name="HelpString" type="dt_EDS_Char_Array"/>
<xsd:element name="MinimumValue" minOccurs="0"/>
<xsd:element name="MaximumValue" minOccurs="0"/>
<xsd:element name="DefaultValue" minOccurs="0"/>
<xsd:element name="ScalingMultiplier" type="dt_UINT" minOccurs="0"/>
<xsd:element name="ScalingDivider" type="dt_UINT" minOccurs="0"/>
<xsd:element name="ScalingBase" type="dt_UINT" minOccurs="0"/>
<xsd:element name="ScalingOffset" type="dt_INT" minOccurs="0"/>
<xsd:element name="MultiplierLink" type="dt_UINT" minOccurs="0"/>
<xsd:element name="DivisorLink" type="dt_UINT" minOccurs="0"/>
<xsd:element name="BaseLink" type="dt_UINT" minOccurs="0"/>
<xsd:element name="OffsetLink" type="dt_INT" minOccurs="0"/>
<xsd:element name="DecimalPrecision" type="dt_USINT" minOccurs="0"/>
<xsd:element name="InternationalParameterName" type="dt_EDS_Char_Array"
minOccurs="0"/>
<xsd:element name="InternationalEngineeringUnits" type="dt_EDS_Char_Array"
minOccurs="0"/>
<xsd:element name="InternationalHelpString" type="dt_EDS_Char_Array" minOccurs="0"/>
<xsd:element name="Enum" minOccurs="0">
<xsd:complexType>
<xsd:sequence maxOccurs="unbounded">
<xsd:element name="EnumValue" type="dt_LINT"/>
<xsd:element name="EnumName" type="dt_EDS_Char_Array"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="et_ProxyParamType">
<xsd:complexContent>
<xsd:extension base="et_ParamType">
<xsd:sequence>
<xsd:element name="ProxyParamSizeAdder" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MinimumValue"/>
<xsd:element name="MaximumValue"/>
```

```
<xsd:element name="DefaultValue"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="et_AssemType">
<xsd:sequence>
<xsd:element name="AssemblyName" type="dt_EDS_Char_Array" minOccurs="0"/>
<xsd:element name="AssemblyPath" type="dt_EPATH" minOccurs="0"/>
<xsd:element name="AssemblyDataSize" type="dt_UINT" minOccurs="0"/>
<xsd:element name="AssemblyDescriptor" type="dt_WORD" minOccurs="0"/>
<xsd:element name="AssemblyMember" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:choice>
<xsd:element name="MemberSize" type="dt_UINT"/>
<xsd:element name="MemberReference" type="et_MemberReferenceType"/>
<xsd:element name="VariantReference">
<xsd:complexType/>
</xsd:element>
<xsd:choice>
<xsd:sequence>
<xsd:element name="MemberSize" type="dt_UINT"/>
<xsd:element name="MemberReference" type="et_MemberReferenceType"/>
</xsd:sequence>
<xsd:sequence>
<xsd:element name="MemberSize" type="dt_UINT"/>
<xsd:element name="VariantReference">
<xsd:complexType/>
</xsd:element>
</xsd:sequence>
</xsd:choice>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="et_MemberReferenceType">
<xsd:union memberTypes="et_AssemReference et_ParamReference dt_UDINT dt_EPATH
xsd:NMTOKEN"/>
</xsd:simpleType>
</xsd:schema>
```

С.2.2 Описание шаблона профиля устройства – инкапсуляция XML файлов EDS

С.2.2.1 Общие положения

Файлы XML профиля устройства, используемые для инкапсуляции файлов EDS, должны соответствовать XML схеме профиля устройства, определенной в С.2.2.2.

Семантика субэлементов элемента ExternalProfileHandle, использованных для ссылки на существующий файл EDS, определена в таблице С.2. В зависимости от значения атрибута WrapperReference ссылка на файл EDS будет осуществляться с использованием элементов идентификации либо в самом файле EDS, либо в продукте, описанном этим EDS.

Примечание 1 – Выбор необходимых элементов идентификации будет зависеть от ожидаемого использования файла оболочки.

Таблица С.2 — Элементы ExternalProfileHandle

Элементы XML схемы	WrapperReference = FILEINFO	WrapperReference =DEVICEINFO
ProfileIdentification	Текст описания файла EDS[a]	ID поставщика, Тип устройства, код продукта[b)]
ProfileRevision ProfileLocation	Версия EDS[a)] EDS HomeURL[a)]	Версия продукта[b)] Имя файла в виде иконки[b)]
[a)] Более подробно см. С.4.1.4.2. [b)] Более подробно см. С.4.1.4.3.		

При наличии элементов DeviceIdentity, DeviceManager, DeviceFunction и ApplicationProcess они должны быть совместимы с форматами, определенными в С.2.1.3.3.

Примечание 2 – Это может быть использовано на этапе перехода от ранее принятого формата EDS к полному формату XML.

С.2.2.2 Схема XML : EDS_Device_Profile_wrapper.xsd

Примечание – Эта XML схема включает в себя файл «MasterTemplateTypes.xsd» (см. С.2.1.3.1).

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                                    ">
<!-- Target namespaces are not specified in this master template -->
<xsd:redefine schemaLocation="MasterTemplateTypes.xsd">
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:complexContent>
<xsd:restriction base="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string" fixed="EDS"/>
</xsd:sequence>
</xsd:restriction>
</xsd:complexContent>
</xsd:complexType>
</xsd:redefine>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* HEADER SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType" fixed="Device"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* BODY SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DeviceIdentity" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any"/>
</xsd:sequence>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceManager" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceFunction" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationProcess" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ExternalProfileHandle">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="ProfileHandle_DataType">
<xsd:attribute name="WrapperReference" use="optional" default="FILEINFO">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="FILEINFO"/>
<xsd:enumeration value="DEVICEINFO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema
```

C.3 Описание шаблона профиля коммуникационной сети

C.3.1 Описание шаблона профиля коммуникационной сети на основе XML

C.3.1.1 Общие положения

Файлы XML профиля коммуникационной сети должны соответствовать XML схеме профиля коммуникационной сети, установленной в C.3.1.2.

Содержание этой XML схемы выводится из диаграмм класса профилей коммуникационной сети, показанных в 6.3.2, и расширено с помощью

дополнительных элементов, что позволяет дать полное описание требований и возможностей коммуникационной сети.

С.3.1.2 Семантика элементов XML схемы

С.3.1.2.1 ProfileBody (тело профиля)

Этот основной элемент ассоциируется с набором атрибутов, которые предоставляют дополнительную информацию о файле профиля.

Семантика этих атрибутов установлена в С.4.1.4.2.

С.3.1.2.2 ApplicationLayers (прикладные уровни)

С.3.1.2.2.1 ConnectionManager (менеджер соединений)

Данный элемент устанавливает поддерживаемые атрибуты и операции сущности объекта Connection Manager Object (см МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2)) совместно с описанием отдельных экземпляров соединения.

Семантика субэлементов Connection, ProxyConnect и ProxiedConnect элемента ConnectionDescriptions определена в С.4.1.4.9 и С.4.1.5.3.3.

С.3.1.2.2.2 MessageRouter (маршрутизатор сообщений)

Данный элемент устанавливает поддерживаемые атрибуты и операции сущности объекта Message Router Object (см. МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2)).

С.3.1.2.3 TransportLayers (транспортные уровни)

С.3.1.2.3.1 ENPhysicalLayer (EN физический уровень)

Данный элемент идентифицирует физический уровень. Содержание этого элемента в данном документе подробно не рассмотрено.

С.3.1.2.3.2 EtherNetIPLinkObject

Данный элемент определяет поддерживаемые атрибуты и операции сущности объекта EtherNet/IP Link Object (см. МЭК 61158-4:2003 (тип 2)).

С.3.1.2.3.3 TCPIPInterfaceObject

Данный элемент определяет поддерживаемые атрибуты и операции сущности объекта TCP/IP Interface Object (см. МЭК 61158-4:2003 (тип 2)).

C.3.1.2.3.4 EncapsulationProtocol (протокол инкапсуляции)

Данный элемент определяет поддерживаемые атрибуты и операции, ассоциированные с инкапсуляцией сообщений приложения в TCP/IP (см. МЭК 61158-6:2003 (тип 2)).

C.3.1.2.3.5 Ports (порты)

Данный элемент идентифицирует порты устройства, позволяющие направлять сообщения из одного канала связи в другой канал связи.

Семантика субэлемента Port элемента Ports определена в C.4.1.4.10 и C.4.2.2.2.

C.3.1.2.4 NetworkManagement (управление сетью)

C.3.1.2.4.1 NM-EtherNetIPLinkObject

Данный элемент определяет поддерживаемые атрибуты и операции класса объекта EtherNet/IP Link Object (см. МЭК 61158-4:2003 (тип 2)).

C.3.1.2.4.2 NM-TCPIPInterfaceObject

Данный элемент определяет поддерживаемые атрибуты и операции класса объекта TCP/IP Interface Object (см. МЭК 61158-4:2003 (тип 2)).

C.3.1.2.4.3 NM-ConnectionManager (NM – менеджер соединений)

Данный элемент определяет поддерживаемые атрибуты и операции класса объекта Connection Manager Object (см. МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2)).

C.3.1.2.4.4 NM-MessageRouter (NM – маршрутизатор сообщений)

Данный элемент определяет поддерживаемые атрибуты и операции класса объекта Message Router Object (см. МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2)).

C.3.1.3 Схема XML: ENet_CommNet_Profile.xsd

Примечание – Данная XML схема включает в себя файлы «MasterTemplateTypes.xsd» (см. C.2.1.3.1) и «CIPDataTypes.xsd» (см. C.2.1.3.2).

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    elementFormDefault="qualified">
<!-- Target namespaces are not specified in this master template -->
<xsd:redefine schemaLocation="MasterTemplateTypes.xsd">
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:complexContent>
<xsd:restriction base="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger" fixed="4"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger" fixed="1"/>
<xsd:element name="ProfileTechnology" type="xsd:string" fixed="EtherNet/IP"/>
</xsd:sequence>
</xsd:restriction>
</xsd:complexContent>
</xsd:complexType>
</xsd:redefine>
<xsd:include schemaLocation="CIPDataTypes.xsd"/>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* HEADER SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"
fixed="CommunicationNetwork"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" fixed="CSI"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* BODY SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ApplicationLayers"/>
<xsd:element ref="TransportLayers"/>
<xsd:element ref="NetworkManagement" minOccurs="0"/>
</xsd:sequence>
<xsd:attributeGroup ref="ag_FileDescription"/>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="ApplicationLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ConnectionManager"/>
<xsd:element ref="MessageRouter"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TransportLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ENPhysicalLayer" minOccurs="0"/>
<xsd:element ref="EtherNetIPLinkObject" minOccurs="0"/>
<xsd:element ref="TCPIPInterfaceObject" minOccurs="0"/>
<xsd:element ref="EncapsulationProtocol" minOccurs="0"/>
<xsd:element ref="Ports" minOccurs="0"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="NetworkManagement">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="NM-EtherNetIPLinkObject" minOccurs="0"/>
<xsd:element ref="NM-TCPIPInterfaceObject" minOccurs="0"/>
<xsd:element ref="NM-ConnectionManager" minOccurs="0"/>
<xsd:element ref="NM-MessageRouter" minOccurs="0"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConnectionManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ConnectionManager_InstanceAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="OpenReqs">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="OpenFormatRejects">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="OpenResourceRejects">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="OpenOtherRejects">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet" use="required"/>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="CloseReqs">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="CloseFormatRejects">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="CloseOtherRejects">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConnTimeouts">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="NumConnEntries">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConnOpenBits">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="CpuUtilization">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxBuffSize">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="BufSizeRemaining">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConnectionManager_InstanceOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="Set_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_List">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Set_Attribute_List">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Set_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Forward_Close">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Unconnected_Send">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Forward_Open">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Connection_Data">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Search_Connection_Data">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Ex_Forward_Open">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Object_Owner">
```

```
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element ref="ConnectionDescriptions" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element
<xsd:element name="ConnectionDescriptions">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Connection" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_ConnectionType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="Connection[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProxyConnect" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_ConnectionType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="ProxyConnect[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProxiedConnect" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_ConnectionType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="ProxiedConnect[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
```

```
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:key name="ConnectionName">
<xsd:selector xpath="Connection|ProxyConnect|ProxiedConnect"/>
<xsd:field xpath="Name_String"/>
</xsd:key>
</xsd:element>
<xsd:element name="MessageRouter">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MessageRouter_InstanceAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Object_List">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaximumConnectionSupported">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="dt_UINT">
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="MessageRouter_InstanceOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_List">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
```

```
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ENPhysicalLayer">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EtherNetIPLinkObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="EtherNetIPLinkObject_InstanceAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="InterfaceSpeed">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="InterfaceFlags">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" use="required"
fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="PhysicalAddress">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" use="required"
fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="InterfaceCounters">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MediaCounters">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EtherNetIPLinkObject_InstanceOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
```

```
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_and_Clear">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TCPIPInterfaceObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="TCPIPInterfaceObject_InstanceAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Status">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" use="required"
fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConfigurationCapability">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" use="required"
fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConfigurationControl">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" use="required"
fixed="Set"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="PhysicalLinkObject">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" use="required"
fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="InterfaceConfiguration">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" use="required"
fixed="Set"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="HostName">
```

```
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" use="required"
fixed="Set"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TCPIPInterfaceObject_InstanceOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Set_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Set_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EncapsulationProtocol">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="EncapsulationProtocol_Attributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProtocolVersion" type="dt_USINT"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EncapsulationProtocol_Operations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="NOP">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="ListServices">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ListIdentity">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ListInterfaces">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="RegisterSession">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="UnRegisterSession">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="SendRRData">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="SendUnitData">
<xsd:complexType>
<xsd:attribute ref="SupportedService" fixed="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="IndicateStatus">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Cancel">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Ports">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Port" minOccurs="0" maxOccurs="unbounded">
```

```
<xsd:complexType>
<xsd:sequence>
<xsd:element name="PortTypeName">
<xsd:simpleType>
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="ControlNet"/>
<xsd:enumeration value="ControlNet_Redundant"/>
<xsd:enumeration value="TCP"/>
<xsd:enumeration value="DeviceNet"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="et_VendorSpecificKeyword"/>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
</xsd:element>
<xsd:element name="PortName" type="dt_EDS_Char_Array" minOccurs="0"/>
<xsd:element name="PortObject" type="dt_EPATH" minOccurs="0"/>
<xsd:element name="PortNumber" type="dt_UINT"/>
<xsd:element name="PortSpecific">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="Port[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="NM-EtherNetIPLinkObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="EtherNetIPLinkObject_ClassAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxInstance">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" fixed="Get"/>
```

```
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EtherNetIPLinkObject_ClassOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="NM-TCPIPInterfaceObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="TCPIPInterfaceObject_ClassAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxInstance">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TCPIPInterfaceObject_ClassOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
```

```
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="NM-ConnectionManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ConnectionManager_ClassAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxInstance">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="OptionalAttributeList">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConnectionManager_ClassOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_List">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="NM-MessageRouter">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MessageRouter_ClassAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="OptionalAttributeList">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="OptionalServiceList">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxIDClassAttributes">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxIDInstanceAttributes">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="MessageRouter_ClassOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_All">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_List">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
```

```
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:attribute name="SupportedService" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:boolean">
<xsd:pattern value="true|false"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:complexType name="et_ConnectionType">
<xsd:sequence>
<xsd:element name="Trigger_Transport" type="dt_DWORD"/>
<xsd:element name="Connection_Parameters" type="dt_DWORD"/>
<xsd:element name="O-T_RPI" minOccurs="0">
<xsd:simpleType>
<xsd:union memberTypes="dt_UDINT et_ParamReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="O-T_Size" minOccurs="0">
<xsd:simpleType>
<xsd:union memberTypes="dt_UINT et_ParamReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="O-T_Format" minOccurs="0">
<xsd:simpleType>
<xsd:union memberTypes="et_ParamReference et_AssemReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="T-O_RPI" minOccurs="0">
<xsd:simpleType>
<xsd:union memberTypes="dt_UDINT et_ParamReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="T-O_Size" minOccurs="0">
<xsd:simpleType>
<xsd:union memberTypes="dt_UINT et_ParamReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="T-O_Format" minOccurs="0">
<xsd:simpleType>
<xsd:union memberTypes="et_ParamReference et_AssemReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="Config1_Size" minOccurs="0">
<xsd:simpleType>
<xsd:union memberTypes="dt_UINT et_ParamReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="Config1_Format" minOccurs="0">
<xsd:simpleType>
<xsd:union memberTypes="et_ParamReference et_AssemReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="Config2_Size" minOccurs="0">
```

```
<xsd:simpleType>
<xsd:union memberTypes="dt_UINT et_ParamReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="Config2_Format" minOccurs="0">
<xsd:simpleType>
<xsd:union memberTypes="et_ParamReference et_AssemReference"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="Name_String" type="dt_EDS_Char_Array"/>
<xsd:element name="Help_String" type="dt_EDS_Char_Array"/>
<xsd:element name="Path"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

С.3.2 Описание шаблона профиля коммуникационной сети – инкапсуляция XML файлов EDS

С.3.2.1 Общие положения

Файлы XML профиля коммуникационной сети, используемые для инкапсуляции файлов EDS, должны соответствовать XML схеме профиля коммуникационной сети, определенной в С.3.2.2.

Семантика субэлементов элемента ExternalProfileHandle, используемых для ссылки на существующий файл EDS, определена в таблице С.2. В зависимости от значения атрибута WrapperReference ссылка на файл EDS будет осуществляться с использованием элементов идентификации либо в самом файле EDS, либо в продукте, описанном этим EDS.

Примечание – Выбор необходимых элементов идентификации будет зависеть от ожидаемого использования файла оболочки.

С.3.2.2 Схема XML : EDS_CommNet_Profile_wrapper.xsd

Примечание – Эта XML схема включает в себя файл «MasterTemplateTypes.xsd» (см. С.2.1.3.1).

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                                    ">
<!-- Target namespaces are not specified in this master template -->
<xsd:redefine schemaLocation="MasterTemplateTypes.xsd">
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:complexContent>
<xsd:restriction base="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
```

```
<xsd:element name="ProfileTechnology" type="xsd:string" fixed="EDS"/>
</xsd:sequence>
</xsd:restriction>
</xsd:complexContent>
</xsd:complexType>
</xsd:redefine>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* HEADER SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"
fixed="CommunicationNetwork"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" fixed="CSI"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* BODY SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ExternalProfileHandle">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="ProfileHandle_DataType">
<xsd:attribute name="WrapperReference" use="optional" default="FILEINFO">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="FILEINFO"/>
<xsd:enumeration value="DEVICEINFO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
```

```
</xsd:element>
</xsd:schema>
```

С.4 Электронная информационная таблица (EDS)

С.4.1 Общие требования к CIP EDS

С.4.1.1 Общие положения

В данном подразделе установлены требования к кодировке файла EDS, которые являются общими для всех сетей, основанных на CIP. Требования к кодировке EDS определяют стандартный формат кодировки файла для применения в продуктах CIP независимо от платформы хоста инструмента конфигурации или файловой системы.

В данном разделе термин «файл» относится ко всем распознаваемым форматам файлов, ассоциированным с файловой системой инструмента конфигурации, независимо от среды хранения файлов.

Файл EDS определяется как файл ASCII, который включает в себя представление ASCII объектов в устройстве, к которым имеется доступ из сети (например, Parameter и Assembly), и некоторую дополнительную информацию, требующуюся для поддержки адресации объекта.

С.4.1.2 Содержание EDS

С.4.1.2.1 Структура EDS

Один файл должен содержать полный EDS. EDS должен состоять из секций. В таблице С.3 представлены сводка данных о структуре секций, которые являются общими для нескольких основанных на CIP сетей, соответствующие принятые разграничители секций и порядок этих секций в EDS.

Таблица С.3 – Структура файла CIP EDS

Имя секции	Принятый разграничитель	Размещение	Требуемый/ Опциональный
Описание файла	[File]	1	Требуемый
Описание устройства	[Device]	2	Требуемый

Окончание таблицы С.3

Имя секции	Принятый разграничитель	Размещение	Требуемый/ Опциональный
Классификация устройства	[Device Classification]	a)	Опциональный
Класс параметра	[ParamClass]	a)	Опциональный
Параметры	[Params]	a)	Опциональный
Группы параметров	[Groups]	a)	Опциональный
Объединение	[Assembly]	a)	Опциональный
Характеристики соединения	[Connection Manager]	a)	Опциональный
Порт	[Port]	a)	Опциональный
Модульный	[Modular]	a)	Опциональный
Определяется продавцом	[VendorID_vendorspecifickeyword]	Последний	Опциональный

a) Размещение опциональных групп должно быть только после требуемых.

Содержание EDS должно быть организовано следующим образом:

– все файлы EDS должны включать в себя раздел «Описание файла», который должен быть первой секцией файла EDS и должен использовать принятый разграничитель [File];

– все файлы EDS должны включать в себя секцию и «Описание устройства», которая должна быть расположена после секции «Описание файла» и должна использовать принятый разграничитель [Device];

– опциональные секции, описанные в данной спецификации, могут быть представлены в любом порядке при условии, что в файле EDS отсутствуют прямые ссылки;

– опциональная(ые) секция(и), определяемая(ые) продавцом, должна(ны) использовать принятые разграничители [VendorID_vendorspecifickeyword (ключевое слово поставщика)] согласно

C.4.1.2.2.11 и должна(ны) быть помещена(ны) после всех секций, определенных в данной спецификации.

C.4.1.2.2 Правила форматирования EDS

C.4.1.2.2.1 Общие положения

Файл EDS должен состоять из секций, входов, полей, комментариев и пустых пробелов. Данный подраздел определяет правила, которые должны выполняться при определении EDS.

C.4.1.2.2.2 Пустой пробел в EDS

Пустой пробел может быть использован в файле EDS, но должен быть проигнорирован всеми интерпретаторами EDS, если он располагается вне полей и наборов символов в двойных кавычках.

Интерпретатор EDS должен рассматривать указанные ниже символы, как символы пустых пробелов. Эти символы, прочитанные интерпретатором, но не кодированные как читаемые человеком символы, означают присутствие в файле пустых пробелов:

— символ пробела;

— новая строка;

— возврат каретки;

— переход на новую строку;

— табуляция (вертикальная или горизонтальная);

— подача страницы;

— маркер конца файла;

— комментарии.

C.4.1.2.2.3 Символы ключевых слов

Все ключевые слова в файле EDS должны состоять из символов ASCII, входящих в следующий список:

— прописные буквы от A до Z;

— строчные буквы от a до z;

— цифры от 0 до 9;

— специальный символ подчеркивания «_»;

— символ пробела.

Пробел должен использоваться только в ключевых словах секций. Пробел может располагаться только внутри имени секции, а множественные пробелы являются ошибкой.

C.4.1.2.2.4 Секции

Файл EDS должен быть разделен на требуемые и опциональные секции.

C.4.1.2.2.5 Разграничители секций

Каждая секция EDS должна быть правильно ограничена ключевыми словами в квадратных скобках (являющихся ограничителями). Правильные принятые разграничители должны соответствовать указанным в таблице C.3.

C.4.1.2.2.6 Ключевые слова секций

Ключевое слово секции определяется как текст между начальным ограничителем ключевого слова «[» и конечным ограничителем «]». Символы, предназначенные для использования в ключевых словах секций, определены в C.4.1.2.2.3. Существуют два типа ключевых слов секций – общие и специальные для поставщиков.

C.4.1.2.2.7 Порядок секций

Каждая требуемая секция должна быть помещена в требуемом порядке согласно условиям, приведенным в C.4.1.2. Опциональные секции могут быть пропущены или включены в виде символа-заполнителя без данных. Кроме относящихся к поставщику секций опциональные секции могут быть помещены в любом порядке. Относящиеся к поставщику секции должны быть в файле EDS на последнем месте.

C.4.1.2.2.8 Вход

Каждая секция EDS должна включать в себя один или более входов, начинающихся с ключевого слова входа, за которым следует знак равенства. Значение ключевого слова входа должно иметь общее значение, допуская использование ключевых слов, определенных в одних секциях, в других секциях. Каждый вход должен быть ограничен точкой с запятой. Вход может

распространяться на несколько строк, если поля правильно разграничены запятыми.

C.4.1.2.2.9 Ключевые слова входа

Ключевое слово входа должно состоять из уникальной последовательности символов ключевого слова согласно определениям, приведенным в C.4.1.2.2.3. Существуют два типа ключевых слов входа – общие и относящиеся к поставщику.

C.4.1.2.2.10 Общее ключевое слово

Общее ключевое слово должно быть определено в спецификации CIP ответственными ассоциациями поставщиков. Общее ключевое слово не должно начинаться с цифрового разряда.

C.4.1.2.2.11 Относящиеся к поставщику ключевые слова

Ключевые слова могут относиться к поставщику. Эти ключевые слова должны начинаться с идентификатора поставщика (Vendor ID) компании с последующим дополнением после символа подчеркивания (VendorID_VendorSpecificKeyword). VendorID должен быть представлен в виде десятичных цифр без нулей на первых позициях. Каждый поставщик несет ответственность за поддержание и документальное оформление относящихся к поставщику ключевых слов.

C.4.1.2.2.12 Поля входа

Каждый вход должен включать в себя одно или более полей. Все поля должны быть разграничены запятыми. Значение поля (полей) должно зависеть от контекста секции. Поля входа должны быть либо обязательными, либо опциональными в соответствии с определениями в данной спецификации. Пустой пробел или отсутствие символа между запятыми должны использоваться для непредоставленных опциональных полей. Точка с запятой может быть использована для указания отсутствия последующих опциональных полей. Элемент «Номер поля» должен указывать положение поля на входе. Поля должны быть пронумерованы слева направо (или сверху вниз), начиная с номера 1.

С.4.1.2.2.13 Ключевые слова поля

Ключевое слово поля должно состоять из уникальной последовательности символов ключевого слова согласно определениям, приведенным в С.4.1.2.2.3. Существуют два типа ключевых слов поля – общие и относящиеся к поставщику.

С.4.1.2.2.14 Составные поля данных

Некоторые поля входа должны быть определены с помощью данных, которые не могут быть установлены одним значением между разграничителями в виде запятых. Возможность дальнейшего разграничения поля входа реализуется путем использования одного или более набора соответствующих символов скобок «{» и «}». Содержание между символами скобок должно рассматриваться как один объект или вход. Содержание может быть сгруппировано с помощью нескольких скобок.

С.4.1.2.2.15 Комментарии

Комментарии должны быть разграничены с помощью символа доллара ($) и символа новой строки. Интерпретатор EDS должен рассматривать все символы между разграничителями комментария как пустой пробел. Разграничитель комментария $, появляющийся внутри поля или набора символов в двойных кавычках, не должен рассматриваться как разграничитель комментария.

Пример – Некоторые примеры комментариев приведены ниже:

$ – правильная строка комментария <NL>

1, 2, 3; $ – правильный комментарий <NL>

$ комментарий не может распространяться <NL>

более чем на одну строку <NL> <= – ошибка – нет $

С.4.1.2.2.16 Пример структуры форматирования EDS

На рисунке С.1 приведен пример, поясняющий структуру EDS.

```
[имя секции]
$ комментарий – распространяется до конца
строки
Entry1=Field1, Field2, Field3                    $ весь вход на одной строке
Entry2=Field1, Field2, Field3, Field4            $ весь вход на одной строке

Entry3=                                          $ вход на нескольких строках
Field1,                                          $ поле1
Field2,                                          $ поле 2
Field3                                           $ поле 3

Entry4=                                          $ комбинация
Field1, Field2,                                  $ поля 1 и 2 она одной строке
Field3,                                          $ поле 3
Field4                                           $ поле 4

Entry5=          1,                              $ поле 1 устанавливает значение 1
{1,2,3}                                          $ поле 2 устанавливает набор или
                                                 структуру с тремя значениями

Entry6=          { 44, {22,33,11} }              $ вход 6 устанавливает одно поле
                                                 $ поле содержит два набора данных
                                                 $ первый набор соответствует одному
                                                 значению 44
                                                 $ второй набор содержит три значения

65535_Entry=                                     $ относящийся к поставщику вход для
Field1, Field2                                   $ Vendor_ID 65535 с двумя полями
```

Рисунок С.1 — Пример структуры форматирования EDS (информативный)

С.4.1.2.3 Требования к наименованию файлов

Никаких соглашений по наименованию файлов EDS на дисках не существует, за исключением файлов в среде DOS/Windows: эти файлы должны иметь суффикс «.EDS», добавляемый к имени файла.

С.4.1.3 Требования к кодировке данных EDS

С.4.1.3.1 Общие положения

Данный подраздел устанавливает требования к кодировке данных в файлах EDS.

Информация, содержащаяся в файле EDS, может представлять собой атрибуты сущностей объектов в подлежащем конфигурации устройстве. Все данные в файле EDS должны быть текстом ASCII, тогда как класс объекта и атрибуты экземпляров объекта не обязательно должны быть в виде ASCII (существующие типы данных определены в спецификации CIP). В связи с этим может быть необходимо преобразование данных в файле EDS и атрибутах объекта.

Простейшие типы данных, установленные в спецификации CIP, используются также для других элементов EDS, однако значение преобразуется согласно описанию в пунктах С.4.1.3.3 – С.4.1.3.10.

Некоторые типы данных используются только в файлах EDS (см. С.4.1.3.11 – С.4.1.3.14).

С.4.1.3.2 Соглашение о файлах с символами ASCII

Все данные в EDS должны быть закодированы с использованием 8-битовых символов ASCII, где все ссылки на «символы ASCII» означают 8-битовый формат символов ASCII (согласно определению, приведенному в таблицах 1 и 2, ряд 00 ИСО/МЭК 10646-1:2000). Символы, которые не могут быть представлены на терминале ANSI, не должны использоваться в именах идентификаторов или в представлении данных. Действующие значения символов ASCII должны включать в себя новую строку, табуляцию и десятичные цифры от 32 до 126.

С.4.1.3.3 Соглашение о строке символов – EDS_Char_Array

С.4.1.3.3.1 Общие положения

Все строковые данные в файле EDS должны быть строками символов фиксированной длины, без символов конца строки, и должны быть заключены в двойные кавычки (тип данных EDS_Char_Array).

Существуют две формы преобразования строковых данных. Символы, содержащиеся между двойными кавычками, должны преобразовываться в 8-битовые символы ASCII. Символы, содержащиеся между двойными кавычками, которым предшествует прописная буква L, должны преобразовываться в символы UNICODE (16-битовые).

Пример 1 – «Эти результаты в строке составлены из 8-битовых символов».

Пример 2 – L «Строка символов UNICODE, включая греческий символ Pi \u03C0».

Примечание – Текст \u03C0 определяет единичный 16-битовый символ, значение которого – 03C0. В наборе символов UNICODE он расположен в таблице 9, ряд 3, основной греческий – символ для строчного «Pi». Описание последовательности переключения кода символов приведено в С.4.1.3.3.5.

С.4.1.3.3.2 Обработка недостаточных символов в поле строки

Интепретатор EDS должен использовать выравнивание по правому знаку или разряду символов в поле и заполнять все неуказанные символы начальными пробелами (ASCII 0x20) во всей оставшейся длине строки.

Пример – Если параметр имеет максимальную длину строки 8 и получает строку «123AB», эта строка интепретируется как «~~~123AB», где символы тильды (~) представляют пробелы.

С.4.1.3.3.3 Обработка избыточных символов в поле строки

Если данное поле строки содержит слишком много символов, интепретатор EDS должен обрезать символы слева направо.

Пример – Если параметр имеет максимальную длину строки 8 и получает строку «I23ABCDEFG», строка обрезается и интепретируется как «I23ABCDE».

С.4.1.3.3.4 Сцепление строк

Множественные строки, не содержащие запятых, должны быть сцеплены (соединены).

Пример 1
Строка: «ABC» «123» «XYZ»
интепретируется как «ABC123XYZ»

Строки могут также быть представлены в виде отдельных строк.

Пример 2

Следующие строки:

«ABC» $ это комментарий

«123»

«XYZ»

интепретируются как «ABC123XYZ»

В строке UNICODE (длинная строка) только перед первым знаком двойных кавычек должна быть прописная буква L.

Пример 3 – L «ABC» «123» «XYZ» аналогичны L «ABC123XYZ».

С.4.1.3.3.5 Управляющие последовательности строки

Интерпретатор EDS должен распознавать все управляющие последовательности, перечисленные в таблице С.4. Интепретация зависит от приложения.

Таблица С.4 – Управляющие последовательности строки

Управляющая последовательность	Преобразование
\\	\
\n	Новая строка
\t \v \b \r	Табуляция Вертикальная табуляция Возврат Возврат каретки
\f	Подача страницы
\a	Символ BELL (0x07)
\"	"
\`	`
\xnn	Один байт, содержащий значение «nn» при шестнадцатиричном выражении
\unnnn	Два байта, содержащих значение «nnnn» при шестнадцатиричном выражении. Эта форма управляющей последовательности будет правильной, только в том случае, если результирующая строка данных будет 16-битовой по длине, например «L» форма спецификации строки

Если встречаются последовательности, не указанные выше, интерпретирующее устройство должно отбраковать всю строку и показать

ошибку. Файлы EDS должны содержать только управляющие последовательности, определенные в таблице С.4.

С.4.1.3.4 Соглашение о строке ASCII (STRING, SHORT_STRING, STRING2)

Все типы строковых данных (STRING, SHORT_STRING, STRING2), используемые в атрибутах объектов, должны быть преобразованы в EDS_Char_Array в файле EDS.

С.4.1.3.5 STRINGI

Тип данных CIP International String (STRINGI) кодируется в файле EDS как сложное представление данных. Полное содержание входа STRINGI должно быть заключено в две скобки. За рядом элементов языка, определенных как USINT, должны следовать определения элементов языка, каждое из которых заключено в пару скобок и отделено запятой. Каждый элемент языка входа STRINGI должен быть задан в виде четырех полей. Первое поле (выбор языка) должно быть выражено в виде строки фиксированной длины точно из трех символов, заключенных в маркеры двойных кавычек – код языка согласно определению, приведенному в ИСО 639-2/Т. Тип строковых данных должен быть выражен с использованием кода типа данных согласно определению, приведенному в спецификации CIP для STRING, STRING2, STRINGN или SHORT_STRING. Выбор набора символов должен быть выражен в виде UINT согласно определению, приведенному в IANA MIB принтерных кодах (RFC 1759). Часть содержания строки, относящаяся к элементу языка, должна быть выражена в виде строки или длинной строки.

Пример –

Далее представлен вход STRINGI с тремя языками:

```
Field1 = { 3,
        {«eng», 0xD0, 4, «Это строка ASCII на английском языке»},
        {«spa», 0xD5,1000, L «Españoles palabras»},        $ «испанские слова»
                                                           $ использующие UNICODE
        {«deu», 0xD0, 4, «Spanische Wörter auf Deutsch»}   $ «испанские слова на немецком»
        };
```

C.4.1.3.6 EPATH

Тип данных CIP EPATH, используемый, в частности, для определения строк путей CIP, должен быть закодирован в файлах EDS с использованием базового формата, определенного в ИСО 15745-4 для EDS_Char_Array. Кроме того, содержание строк для путей CIP или других данных EPATH должно состоять из групп, состоящих из двух прилегающих шестнадцатеричных символов, разделенных пробелами. Могут быть использованы символы как верхнего, так и нижнего регистра.

Пример 1 – «20 04 24 01»

Пример 2 – «20 05 24 02 30 04»

C.4.1.3.7 Соглашение о беззнаковых целых числах ASCII (USINT, UINT, UDINT, ULINT)

Типы данных беззнаковых целых чисел представляют собой значения положительных целых чисел. Данные беззнаковых целых чисел должны вводиться либо в десятичной, либо в шестнадцатеричной нотации без пробелов или запятых между символами. Если шестнадцатеричная нотация используется для представления символов беззнаковых целых чисел, перед символами беззнаковых целых чисел должна быть поставлена последовательность из двух символов 0x без пробелов.

Диапазон принятых данных USINT:

десятичная нотация: от 0 до 255

шестнадцатеричная нотация: от 0x0 до 0xFF

Диапазон принятых данных UINT:

десятичная нотация: от 0 до 65535

шестнадцатеричная нотация: от 0x0 до 0xFFFF

Диапазон принятых данных UDINT:

десятичная нотация: от 0 до 4294967295

шестнадцатеричная нотация: от 0x0 до 0xFFFFFFFF

Диапазон принятых данных ULINT:

десятичная нотация: от 0 до 18446744073709551615

шестнадцатеричная нотация: от 0x0 до 0xFFFFFFFFFFFFFFFF

Ведущие нули использовать в десятичной нотации не допускается, но их можно использовать в шестнадцатиричной нотации. В шестнадцатеричной нотации допускается использовать как прописные, так и строчные символы, а полное число символов должно быть ограничено 10 (0x плюс 8) или 18 (0x плюс 16) для типа ULINT.

Пример – Десятичное значение UINT 254 может быть представлено как 254 (десятичное) или как 0xFE (шестнадцатиричное), или как 0x000000FE (шестнадцатеричное), но 0254 (десятичное) и 0x0000000FE (шестнадцатиричное) являются неправильными.

C.4.1.3.8 Соглашение о целых числах ASCII со знаком (SINT, INT, DINT, LINT)

Типы данных SINT, INT, DINT и LINT представляют значения целых чисел со знаком. Данные целых чисел со знаком должны вводиться либо в десятичной, либо в шестнадцатеричной нотации без пробелов или запятых между символами. Если шестнадцатеричная нотация используется для представления символов целых чисел со знаком, перед символами целых значений должна быть поставлена последовательность двух символов 0x без пробелов.

Диапазон принятых данных SINT:

десятичная нотация: от -128 до 127

шестнадцатеричная нотация: от 0x80 до 0x7F

Диапазон принятых данных INT:

десятичная нотация: от -32768 до 32767

шестнадцатеричная нотация: от 0x80 до 0x7FFF

Диапазон принятых данных DINT:

десятичная нотация: от -2147483648 до 2147483647

шестнадцатеричная нотация: от 0x80000000 до 0x7FFFFFFF

Диапазон принятых данных LINT:

десятичная нотация: от -9223372036854775808 до 9223372036854775807

шестнадцатеричная нотация: от 0x8000000000000000 до 0x7FFFFFFFFFFFFFFF

Ведущие нули использовать в десятичной нотации не допускается, но их можно использовать в шестнадцатеричной нотации. В шестнадцатеричной нотации допускается использовать как прописные, так и строчные символы, а полное число символов должно быть ограничено 10 (0x плюс 8) или 18 (0x плюс 16) для типа LINT.

Пример – Десятичное значение INT 254 может быть представлено как 254 (десятичное) или 0xFE (шестнадцатеричное), или 0x000000FE (шестнадцатеричное), но 0254 (десятичное) и 0x0000000FE (шестнадцатеричное) являются неправильными.

C.4.1.3.9 Соглашение о словах ASCII (BYTE, WORD, DWORD, LWORD)

Типы данных BYTE, WORD, DWORD и LWORD представляют величины с побитовой адресацией. Эти величины считаются дискретными значениями позиции двоичного разряда и не предназначены для представления целых величин со знаком или без знака. Однако эти величины следует вводить либо в десятичной, либо в шестнадцатеричной, либо в двоичной нотации без пробелов или запятых между символами. Если для представления символов величин используется шестнадцатеричная (соответственно двоичная) нотация, перед символами значения должна быть поставлена последовательность из двух символов 0x (соответственно 0b) без пробелов.

Диапазон принятых данных BYTE:

десятичная нотация:	от 0 до 255
шестнадцатеричная нотация:	от 0x0 до 0xFF
двоичная нотация:	от 0b00000000 до 0b11111111

Диапазон принятых данных WORD:

десятичная нотация:	от 0 до 65535
шестнадцатеричная нотация:	от 0x0 о 0xFFFF
двоичная нотация:	от 0b0000000000000000 до 0b1111111111111111

Диапазон принятых данных DWORD:

десятичная нотация:	от 0 до 4294967295
шестнадцатеричная нотация:	от 0x0 до 0xFFFFFFFF

двоичная нотация: от 0b00000000000000000000000000000000

до 0b11111111111111111111111111111111

Диапазон принятых данных LWORD:

десятичная нотация:от 0 до 18446744073709551615

шестнадцатеричная нотация: от 0x0 до xFFFFFFFFFFFFFFFF

двоичная нотация: от

0b0000000000000000000000000000000000000000000000000000000000000000

до

0b1111111111111111111111111111111111111111111111111111111111111111

Ведущие нули использовать в десятичной нотации не допускается, но их можно использовать в шестнадцатеричной и двоичной нотациях. В шестнадцатеричной нотации допускается использовать как прописные, так и строчные символы, а полное число символов должно быть ограничено 10 (0x плюс 8) или 18 (0x плюс 16) для типа LWORD.

Пример – Десятичное значение WORD 254 может быть представлено как 254 (десятичное) или 0xFE (шестнадцатиричное), или 0x000000FE (шестнадцатеричное), но 0254 (десятичное) и 0x0000000FE (шестнадцатеричное) являются неправильными.

C.4.1.3.10 Соглашение по ASCII с плавающей точкой (REAL, LREAL)

Типы данных REAL и LREAL представляют двоичные величины с плавающей точкой. Внутреннее представление этих форматов данных описано в IEEE 754. Этот стандарт описывает как числовые величины, так и двоичные последовательности, которые интерпретируются как «нечисловые»(NaN) символьные величины и положительная и отрицательная бесконечности. Величины с плавающей запятой могут вводиться либо как значения целых чисел, либо как величины, основанные на десятичном представлении с плавающей точкой, или величины, вводимые в «научной» нотации с использованием базовой величины и сдвига в экспоненциальной форме. Значения целых чисел аналогичны используемым в типах данных INT, DINT или LINT. Эти величины не могут быть использованы для представления дробных величин. Десятичные с плавающей точкой величины аналогичны величинам, включающим в себя целую и дробную компоненту.

Целочисленная величина и дробные компоненты разделяются десятичной точкой «.» или знаком точка-разделитель. Экспоненциальная (научная) форма нотации величины аналогичны представлению дробной величины, но с добавлением экспоненциальной компоненты. Экспонента представляет собой целую величину со знаком степени десять, умноженную на базовую величину.

Примечание – Максимальная точность величины с плавающей точкой определяется возможностями внутреннего двоичного формата, т. е. числом двоичных разрядов, применяемых для кодирования мантиссы. Следовательно, использование большого числа десятичных разрядов в десятичной нотации (или составляющей мантиссу части научной нотации) величины с плавающей точкой предназначено больше для удобства, чем для повышения точности. EDS определяет произвольные пределы числа десятичных разрядов.

Диапазон принятых данных REAL (единый IEEE, 32-битовый формат) определяют по формуле

значение = $(-1)^{s} \bullet (2)^{e-127} \bullet (m)$,

где s – значение бита знака;

e – 8-битовая экспонента, которая допускает диапазон экспоненты от минус 126 до плюс127;

m – нормализованная 24-битовая мантисса (23 внутренних для хранения плюс один скрытый бит), что допускает использование диапазона значений мантиссы от 0 до 16777215.

Комбинация e и m допускает использование приближенного абсолютного диапазона значений от 0 до $3{,}4028e^{38}$.

EDS использует для данных REAL следующие нотации величин с плавающей точкой:

целочисленная (фиксированная) нотация: от 16777215 до 16777215

десятичная (с плавающей точкой) нотация: от 0.0 до ±9999999999999999

Полное число разрядов не должно превышать 16 (дополнительно к символам десятичной точки и знака). Как символ десятичной точки, так и символ знака могут быть пропущены (подразумевается знак «+», если символ знака пропущен).

Научная нотация: от 0.0 до ±nn.nnnnnnnnnE±xxxx

Полное число разрядов мантиссы не превышает 11 (дополнительно к символу десятичной точки и символу знака), а число разрядов экспоненты не должно превышать 4 (дополнительно к символу «E» и символу знака). Десятичная точка может быть помещена в мантиссе где угодно. Как символ десятичной точки, так и символ знака могут быть опущены в мантиссе (знак «+» подразумевается, если символ знака опущен).

Диапазон допустимых данных LREAL (двойной IEEE, 64-битовый формат) определяют по формуле

величина = $(-1)^{s} \bullet (2)^{e-1023} \bullet (m)$,

где

s - значение бита знака;

e - 11-битовая экспонента, которая допускает диапазон значений от 1022 до +1023;

m - нормализованная 53-битовая мантисса (52 внутренних в памяти плюс один скрытый бит), что допускает использование диапазона значений мантиссы от 0 до 9007199254740991.

Комбинация e и m допускает использвание абсолютной величины в диапазоне от 0 до $1,7976e^{308}$.

EDS использует для данных LREAL следующие нотации величин с плавающей запятой:

— целая (фиксированная) нотация: от 9007199254740991 до 9007199254740991,

— десятичная (с плавающей запятой) нотация: от 0.0 до ±9999999999999999.

Полное число разрядов не превышает 16 в дополнение к символам десятичной точки и знака. Как символ десятичной точки, так и символ знака могут быть опущены (знак «+» подразумевается, если символ знака опущен).

Научная нотация: 0.0 до ±nnnn.nnnnnnnnnnnnE±xxxx.

Полное число разрядов мантиссы не превышает 16 (дополнительно к символу десятичной точки и символу знака), а число разрядов экспоненты не должно превышать 4 (дополнительно к символу «E» и символу знака).

Десятичная точка может быть помещена в мантиссе где угодно. Как символ десятичной точки, так и символ знака могут быть опущены в мантиссе (знак «+» подразумевается, если символ знака опущен).

Дополнительно к указанным выше величинам ввода представление с плавающей запятой допускает использование двух видов «нечислового» или NaN символического ввода и двух форм бесконечности. Существуют два типа NaN: сигнальный (Signaling) NaN и тихий (Quiet) NaN. Формат также допускает представление величин положительной и отрицательной бесконечности. Для этих случаев зарезервированы указанные ниже специальные слова, которые должны использоваться для представления ввода соответствующих символов с плавающей запятой:

— тихий нечисловой: QUIET-NAN

— сигнальный нечисловой: SIGNAL-NAN

— положительная бесконечность: INFINITY (или +INFINITY)

— отрицательная бесконечность: – INFINITY

C.4.1.3.11 EDS_Date

Тип данных EDS_Date должен иметь формат mm-dd-yyyy, где mm — месяц, dd — день месяца и yyyy — год. Значения данных для месяца, дня и года в mm-dd-yyyy задают следующим образом:

- mm — от 01 до 12;
- dd — от 01 до 31 (в зависимости от месяца и года);
- yyyy — от 1996 до 9999.

Может быть использовано двузначное представление года, в этом случае тип данных EDS_Date должен иметь формат mm—dd—yy, где mm — месяц, dd — день месяца и yy — год. Две цифры года подразумевают наличие в начале числа 19, так что yy=96 будет соответствовать 1996. В этом случае значения данных для месяца, дня и года в mm-dd-yyyy задают следующим образом:

- mm — от 01 до 12;
- dd — от 01 до 31 (в зависимости от месяца и года);
- yy — от 96 до 99 (подразумевается наличие в начале числа 19).

Примечание – Использовать двузначное представление года не рекомендуется.

C.4.1.3.12 EDS_Time_Of_Day

Тип данных EDS_Time_Of_Day должен иметь формат hh:mm:ss, где hh — часы, mm — минуты и ss — секунды. Правильные значения часов, минут и секунд должны быть следующими:

- hh — от 00 до 23;
- mm — от 00 до 59;
- ss — от 00 до 59.

C.4.1.3.13 EDS_Revision

Тип данных EDS_Revision должен иметь формат Major_ Revision.Minor_Revision со следующими значениями:

- Major_Revision – от 0 до 9;
- Minor_Revision – от 0 до 9.

EDS_Revision со значением 0.0 является неправильным.

Пример – EDS_Revision со значением 1.4 соответствует большой проверке со значением 1 и малой проверке со значением 4.

C.4.1.3.14 Унифицированный указатель информационного ресурса EDS_URL

Все ссылки на EDS_URL в рамках требований EDS предназначены для получения формализованной информации, необходимой для поиска и получения ресурсов путем использования сети Интернет. EDS_URL должен быть закодирован в файлах EDS с использованием базового формата, определенного в ИСО 15745-4 для EDS_Char_Array. Кроме того, содержание строки для EDS_URL должно быть в формате, определенном Рабочей группой сети Интернет RFC 1738 «Унифицированный указатель информационного ресурса (URL)». В спецификациях файла EDS EDS_URL должно быть ограничено одной из следующих форм:

- http;
- ftp;
- file.

C.4.1.4 Основные требования к файлу EDS

C.4.1.4.1 Обзор

В данном пункте описаны основные секции EDS, которые являются общими для ряда основанных на CIP сетей, и установлены соответствующие требования при использовании.

В таблице С.5 приведено расположение подразделов, содержащих определения этих секций.

Таблица С.5 — Определение основных секций

Секция EDS	Пункт настоящего стандарта
Секция описания файлов	С.4.1.4.2
Секция описания устройств	С.4.1.4.3
Секция классификации устройств	С.4.1.4.4
Секция классов параметров	С.4.1.4.5
Секция параметров	С.4.1.4.6
Секция групп параметров	С.4.1.4.7
Секция объединений	С.4.1.4.8
Секция менеджера соединения	С.4.1.4.9
Секция порта	С.4.1.4.10
Секция модулей	С.4.1.5.2

С.4.1.4.2 Секция описания файлов

Секция описания файлов должна содержать административную информацию о файле EDS. Инструмент конфигурации должен считывать эту информацию, форматировать ее и предоставлять пользователю. Пользователь может также получить доступ в эту секцию для просмотра текста файла и показа неформатированной информации. Эта секция не требует выполнения модификации, если только пользователь не выполняет модификацию файла вручную. Секция описания файла должна содержать входы, указанные в таблице С.6.

Таблица С.6 – Формат описания файла

Имя входа	Ключевое слово входа	Номер поля	Тип данных	Требуемое/ опциональное
Текст описания файла	DescText	1	EDS_Char_Array	Требуемое
Дата создания файла	CreateDate	1	EDS_Date	Требуемое
Время создания файла	CreateTime	1	EDS_Time_Of_Day	Требуемое
Дата последней модификации	ModDate	1	EDS_Date	Условное
Время последней модификации	ModTime	1	EDS_Time_Of_Day	Условное
EDS Версия	Версия	1	EDS_Revision	Требуемое
Домашний URL	HomeURL	1	EDS_URL	Опциональное

Входы в секции описания файла должны предоставлять информацию, указанную в таблице С.7.

Таблица С.7 – Входы описания файлов

Вход	Описание
Текст описания файла	Одна строка текста показывается на дисплее конфигурационным инструментом. Разработчик EDS должен присвоить значащую строку текста этому входу. Все наборы символов должны быть заключены в двойные кавычки
Дата создания файла	Дата создания EDS, присвоенная разработчиком EDS. Предоставленная только для удобства, эта дата может быть использована для получения информации о версии файла. Инструмент конфигурации не должен использовать эту информацию для какого-либо контроля версии, но он может показывать содержимое
Время создания файла	Время создания EDS, присвоенное разработчиком EDS. Предоставленное только для удобства, это время может быть использовано для получения информации о версии файла. Инструмент конфигурации не должен использовать эту информацию для какого-либо контроля версии, но он может показывать содержимое

Окончание таблицы С.7

Входы	Описание
Дата последней модификации	Дата последней модификации EDS. Инструмент конфигурации, который позволяет выполнять модификацию файла EDS, должен обновлять это поле по мере необходимости. Инструмент конфигурации должен показывать содержимое этого входа, если оно существует. Если инструмент конфигурации изменяет EDS, он должен обновлять это поле. Однако если поле EDS модифицировано вручную или с помощью редактора текстов, его также необходимо обновлять. Данный вход требуется, если: – файл EDS модифицирован с помощью программного инструмента; – имеется вход времени последней модификации
Время последней модификации	Время последней модификации EDS. Инструмент конфигурации, который позволяет выполнять модификацию файла EDS, должен обновлять это поле по мере необходимости. Инструмент конфигурации должен показывать содержимое этого входа, если оно существует. Если инструмент конфигурации изменяет EDS, он должен обновлять это поле. Однако если поле EDS модифицировано вручную или с помощью редактора текстов, его также необходимо обновлять
Версия EDS	Версия EDS не должна иметь взаимосвязи с версией продукта, она является версией файла EDS
Домашний URL	Унифицированный указатель информационного ресурса мастера файла EDS, файла иконки и других файлов, относящихся к EDS. Домашний URL должен указывать точный URL для ссылки на мастер версию файла EDS. Кроме того, область ссылки (без указания спецификации имени файла) используется для указания области, где находятся другие файлы, относящиеся к устройству, описанному данным EDS

На рисунке С.2 приведен пример, показывающий типичную секцию [File].

```
[File]
DescText = «Smart Widget EDS File»;
CreateDate = 04-03-94;                    $ создан
CreateTime = 17:51:44;
```

```
ModDate = 04-06-94;                    $ последнее
                                       изменение
ModTime = 22:07:30;
Revision = 2.1;                        $ Проверка EDS
HomeURL = gost.e-ecolog.ru                    ;
```

Рисунок С.2 — Пример секции [File - Файл] (информативный)

A.4.1.4.3 Секция описания устройства

Секция описания устройства должна содержать информацию изготовителя об устройстве, включая некоторые из таких величин, как Identity Object устройства. Секция описания устройства должна включать в себя входы, указанные в таблице С.8.

Таблица С.8 — Формат описания устройства

Имя входа	Ключевое слово входа	Номер поля	Тип данных	Требуемое/ Опциональное
Vendor Id (идентификатор продавца)[a), b)]	VendCode	1	UINT	Требуемое
Vendor Name (имя продавца)	VendName	1	EDS_Char_Array	Требуемое
Device Type (тип устройства)[a), b)]	ProdType	1	UINT	Требуемое
Device Type String (строка типа устройства)	ProdTypeStr	1	EDS_Char_Array	Требуемое
Product Code (код продукта)[a), b)]	ProdCode	1	UINT	Требуемое
Major Revision (большая проверка)[a), b)]	MajRev	1	USINT	Требуемое

Окончание таблицы С.8

Имя входа	Ключевое слово входа	Номер поля	Тип данных	Требуемое/ Опциональное
Minor Revision (малая проверка)[a]	MinRev	1	USINT	Требуемое
Product Name (имя продукта)[c]	ProdName	1	EDS_Char_Array	Требуемое
Catalog Number (номер каталога)	Catalog	1	EDS_Char_Array	Опциональное
Exclude from Adapter Rack Connection (исключить из адаптера соединения стойки)	ExcludeFromAdapterRackConnection	1	EDS_Char_Array	Опциональное
Icon File Name (имя файла иконки)	Icon	1	EDS_Char_Array	Опциональное

[a] Данный вход представляет атрибут Identity Object (объект идентичности).

[b] Данный вход используется для согласования EDS с конкретным продуктом/версией.

[c] Данный вход представляет атрибут Identity Object (объект идентичности), хотя тип данных может незначительно отличаться.

Имя входа для поля описания устройства описывает уникальный номер строки ввода данных.

Инструмент конфигурации должен использовать требуемые входы в секции описания устройства для согласования EDS с конфигурируемым устройством. Входы секции описания устройства должны предоставлять информацию, указанную в таблице С.9.

Таблица С.9 — Входы описания устройства

Вход	Описание
Vendor ID (идентификатор продавца)	Числовой идентификатор продавца, определенный в Identity Object, атрибут 1
Vendor Name (имя продавца)	Текстовое имя продавца. При представлении на дисплее может быть обрезано для соответствия дисплею
Device Type (тип устройства)	Числовой идентификатор устройства, определенный в Identity Object, атрибут 2
DeviceType String (строка устройство-тип)	Текстовое описание типа устройства согласно определению в соответствующем профиле устройства CIP. Индивидуальные поставщики могут выбирать строки для конкретных типов устройств поставщика
Product Code (код продукта)	Присвоенный поставщиком числовой код идентификатора продукта, определенный в Identity Object, атрибут 3. Каждый код продукта должен иметь свой собственный EDS
Major Revision (большая проверка)	Присвоенный поставщиком номер большой проверки, определенный в Identity Object, атрибут 4. Большая проверка продукта может обычно проводиться по частям, когда возникают изменения формы, установки, или функций устройства. Изменения при больших проверках должны быть использованы инструментом конфигурации для приведения устройства в соответствие EDS
Minor Revision (малая проверка)	Присвоенный продавцом номер малой проверки, определен в Identity Object, атрибут 4. Номер малой проверки должен быть использован для идентификации изменений продукта, не оказывающих влияния на выбор конфигурации пользователя (ошибки программно-аппаратных средств, дополнительные LED, внутренние

Продолжение таблицы С.9

Вход	Описание
	изменения аппаратных средств). Изменения при малых проверках не должны использоваться инструментом конфигурации для приведения устройства в соответствие EDS
Product Name (имя продукта)	Текстовое имя продукта, определенное в Identity Object, атрибут 7. При представлении на дисплее может быть обрезано для соответствия дисплею
Catalog Number (номер каталога)	Номер в текстовом каталоге или модели. С конкретным кодом продукта может быть связан один или более номеров каталога. Примечание – При наличии нескольких номеров каталога реокмендуется предоставлять их как можно номеров. Например 1438-BAC7xx, где 'xx' представляет экземпляры номеров каталога, поддерживаемых кодом/EDS данного продукта
ExcludeFromAdapterRackConnection (исключить из адаптера соединения стойки)	Это поле используется для описания того, должно ли быть установленное на стенде устройство исключено из соединения переходной стойки. Если значение поля – строка «Да», этот модуль должен быть исключен из соединения переходной стойки путем переустановки соответствующих битов щелевой маски (ввода, вывода и конфигурации). Если значение поля – строка «Нет» или это необязательное поле пропущено, соответствующие биты щелевой маски могут быть установлены
Icon File Name (имя файла иконки)	Имя файла иконки указывает на файл, содержащий графическое представление устройства. Файл должен иметь формат *.ICO MSWindows и должен, как минимум, содержать иконку 16x16. Файл может также содержать

Окончание таблицы С.9

Входы	Описание

	иконки 32x32, 48x48 и 64x64. Расположение файла иконки представляет собой комбинацию места, указанного ключевым словом HomeURL (без компонента имени файла HomeURL), и имени файла, указанного этим ключевым словом. Это ключевое слово должно быть только в том случае, если имеется ключевое слово HomeURL

На рисунке С.3 приведен пример типичной секции устройства.

```
[Device - Устройство]
VendCode = 65535;
VendName = "Widget-Works, Inc.";
ProdType = 0;
ProdTypeStr = “Generic”;
ProdCode = 42;
MajRev = 1;                    $ большая проверка устройства
MinRev = 1;                    $ малая проверка устройства
ProdName = "Smart-Widget";
Catalog = “1499-DVG”;
Icon = “example.ico”.
```

Рисунок С.3 — Пример секции [Device — Устройство] (информативный)

С.4.1.4.4 Секция Device Classification (классификация устройства)

Секция классификации устройств должна выполнять классификацию устройств, описанных в EDS, по одной или более категориям устройств. Ключевое слово всех классов должно состоять из набора символов «Class» («класс»), скомбинированных с десятичным числом. Числа должны начинаться с 1 для первого класса и увеличиваться для каждого следующего класса.

Число полей каждого входа классификации должно быть переменным для создания возможности древовидной структуры классификации, аналогичной структуре файловой системы каталога. Должны быть зарезервированы подклассы общей классификации. Классификация

поставщика может иметь подклассы по его выбору. Первое поле должно представлять наиболее высокий уровень в древовидной структуре и должно быть одним из следующих ключевых слов поля:

- ControlNet;
- DeviceNet;
- EtherNetIP;
- ключевое слово поля поставщика.

Ключевое слово поля поставщика должно начинаться с идентификатора поставщика Vendor ID компании с дополнением через черту снизу специального поля поставщика VendorID_VendorSpecificField. Идентификатор поставщика VendorID должен быть выражен в виде десятичной цифры, не содержащей нулей на передних разрядах. Каждый поставщик несет ответственность за поддержание и документальное оформление ключевого слова для поля поставщика.

С.4.1.4.5 Секция Parameter Class (класс параметров)

Секция класс параметров должна определять общие атрибуты параметров конфигурации, описанных в EDS, соответствующие подмножеству атрибутов класса Parameter Object согласно описанию в библиотеке объектов CIP.

Секция класса параметров должна содержать входы, указанные в таблице С.10.

Таблица С.10 – Формат класса параметров

Имя входа	Ключевое слово входа	Номер поля	Тип данных	Требуемый/ опциональный
Max Instances (Макс сущности)	MaxInst	1	UINT	Требуемый
Parameter Class Descriptor (дескриптор класса параметров)	Descriptor	1	WORD	Требуемый
Configuration Assembly Instance (сущность объединения конфигураций)	CfgAssembly	1	UINT	Требуемый

Входы секции класс параметров должны предоставлять информацию, указанную в таблице С.11.

Таблица С.11 – Входы класса параметров

Вход	Описание
Max Instances (максимальное значение сущности)	Определяет полное число параметров конфигурации, содержащихся в устройстве, ассоциированном с EDS
Parameter Class Descriptor (дескриптор класса параметров)	Содержит битовые флажки, описывающие поведение объектов параметр устройства
Configuration Assembly Instance (сущность объединения конфигураций)	Определяет номер сущности Assembly Object, содержащей данные о конфигурации устройства

Вход Parameter Class Descriptor должен содержать биты, предназначенные для описания характеристик параметров, согласно определению в таблице С.12. Биты, не определенные в таблице С.12, не должны использоваться и должны быть установлены на нуль (0).

Таблица С.12 – Значения битов дескриптора класса параметров

Бит	Имя	Значение бита и его смысл
0	Поддержка отдельного доступа к параметру	0 = Параметр не может быть индивидуально доступен. Используется только объединение конфигураций; 1 =параметр может быть доступен индивидуально
1	Поддержка всех атрибутов	0 = только текущее значение параметра доступно в устройстве; 1 = все данные конфигурации для параметра доступны внутри устройства
2	Команда постоянного сохранения	0 = Параметры сохраняются автоматически; 1 = Параметры не сохраняются автоматически. Необходима команда для обеспечения постоянного сохранения тех параметров, которые необходимо сохранить в постоянной памяти

Окончание таблицы С.12

Бит	Имя	Значение бита и его смысл

3	Параметры сохраняются в постоянной памяти	0 = параметры не сохраняются в постоянной памяти. 1 = все параметры полностью сохраняются в постоянной памяти

На рисунке С.4 приведен пример типичной секции класса параметров.

```
[ParamClass]
MaxInst = 3;
Descriptor = 0x0E;
CfgAssembly = 3.
```

Рисунок С.4 – Пример секции класса параметров ParamClass (информативный)

С.4.1.4.6 Секция параметров

Секция параметров должна определять параметры конфигурации в устройстве. Ключевое слово входа должно быть одним из следующих наборов символов «Param», «ProxyParam», «ProxiedParam», скомбинированных с номером сущности параметра (десятичным) для устройства, например «Param1». Сущность объекта параметра может (но необязательно) применяться в устройстве, но все сущности объекта параметр должны иметь соответствующий вход «ParamN» в EDS. Однако если сущность объекта параметр существует внутри узла и параметр описан в EDS, то значение «N» в «ParamN» должно соответствовать сущности объекта параметра.

Каждый вход должен содержать форматированные поля, указанные в таблице С.13. Ключевые слова «ProxyParam» и «ProxiedParam» определены в С.4.1.5.3.1 в качестве части требований модульного EDS.

Таблица С.13 – Формат параметров

Имя поля	Номер Поля	Тип данных	Требуемое/ опциональное
Reserved (зарезервировано)	1	USINT	Требуемое
Link Path Size (размер маршрута связи)	2	USINT	Опционально

Продолжение таблицы С.13

Имя поля	Номер	Тип данных	Требуемое/

	Поля		Опциональное
Link Path (маршрут связи)	3	EPATH	Опционально
Descriptor (дескриптор)	4	WORD	Требуемое
Data Type (тип данных)	5	USINT/EPATH	Требуемое
Data Size (размер данных)	6	USINT	Требуемое
Parameter Name (имя параметра)	7	EDS_Char_Array	Требуемое
Units String (единичная строка)	8	EDS_Char_Array	Требуемое
Help String (вспомогательная строка)	9	EDS_Char_Array	Требуемое
Minimum Value (минимальное значение)	10	Тип данных	Условное[a)]
Maximum Value (максимальное значение)	11	Тип данных	Условное[a)]
Default Value (значение по умолчанию)	12	Тип данных	Требуемое
Scaling Multiplier (множитель масштабирования)	13	UINT	Опциональное
Scaling Divider (делитель масштабирования)	14	UINT	Опциональное
Scaling Base (база масштабирования)	15	UINT	Опциональное
Scaling Offset (сдвиг масштабирования)	16	INT	Опциональное
Multiplier Link (множественная связь)	17	UINT	Опциональное
Divisor Link (делительная связь)	18	UINT	Опциональное
Base Link (базовая связь)	19	UINT	Опциональное

Окончание таблицы С.13

Имя поля	Номер Поля	Тип данных	Требуемое/ Опциональное
Offset Link (связь со сдвигом)	20	UINT	Опциональное
Decimal Precision (десятичная точность)	21	USINT	Опциональное
International Parameter Name (международное имя параметра)	22	STRINGI	Опциональное
International Engineering Units (международные технические единицы)	23	STRINGI	Опциональное
International Help String (международная строка помощи)	24	STRINGI	Опциональное
[a] Параметры указаны в таблице С.17.			

Входы в секции параметров должны предоставлять информацию, указанную в таблицах С.14 и С.18.

Поля параметров, перечисленные в таблице С.14, являются общими для всех параметров.

Таблица С.14 – Общие поля параметров

Поле	Описание
Зарезервировано	Это поле должно быть первым и содержать ноль
Link Path Size (размер маршрута связи)	Число байтов, используемых для представления пути. Если размер связи не согласуется с числом байтов в поле «Link Path», то «Link Path» должен быть проигнорирован. Если этот параметр не адресуется из связи, то поле должно быть пустым. Если поле пустое, «Link Size» должен быть равен числу байтов в поле «Link Path»
Link Path (маршрут связи)	Путь CIP к атрибуту объекта из места, где извлечена величина параметра. Путь следует вводить как набор символов, используя нотацию пути, описанную в МЭК 61158-6:2003 (тип 2) с форматом указанным в А.4.1.3.6. Если параметр, описанный этим входом

Продолжение таблицы С.14

Поле	Описание
	ParamN, не адресуется прямо из сети, то поле должно быть пустым. Если это поле содержит нулевую строку "", параметр, описанный входом ParamN, должен быть адресуем как атрибут данных (атрибут сущности 1) N-й сущности объекта Parameter (например, используя путь 20 0F 24 N 30 01)
Descriptor (дескриптор)	Параметр дескриптора. Содержит битовые флажки, описывающие поведение отдельных параметров (см. таблицу С.15)
Data Type (тип данных)	Идентификатор типа данных согласно определению, приведенному в МЭК 61158-6:2003 (тип 2: Data Type Reporting; сообщение типа данных). Этот идентификатор должен быть закодирован либо как USINT, либо как EPATH. Примечание – Старые версии файлов EDS могут использовать идентификаторы типа данных USINT согласно таблице С.16, но в настоящее время не действительно. Данные идентификаторы предоставлены здесь по причинам совместимости
Data Size (размер данных)	Числовая величина размера данных. Для строковых и EPATH типов данных это поле устанавливает число байтов на символ или вход. Следовательно для типов данных STRING и EPATH это значение должно быть равно 1; для типа данных STRING2 — 2; для типа данных STRINGN — «N»
Parameter Name (имя параметра)	Текстовое имя параметра. При необходимости следует сократить текст до максимальной длины набора
Units String (единичная строка)	Текстовое представление набора символов. При необходимости следует сократить текст до максимальной длины набора
Help String (вспомогательная строка)	Текстовый вспомогательный набор символов. При необходимости следует сократить извлеченый текст до максимальной длины набора

Окончание таблицы С.14

Поле	Описание
Minimum Value (минимальное значение)	Значения и требования, основанные на типе данных параметров см. таблицу С.17.
Maximum Value (максимальное значение)	Значения и требования, основанные на типе данных параметров см. таблицу С.17
Default Value (зачение по умолчанию)	Числовые значения по умолчанию, присвоенные величине данных параметра
International Parameter Name (международное имя параметра)	Имя параметра, выраженное в нотации STRINGI
International Engineering Name (международное техническое наименование)	Технические единицы, выраженные в нотации STRINGI
International Help String (международная вспомогательная строка)	Вспомогательная строка, выраженная в нотации STRINGI

Биты поля дескриптора должны соответствовать определениям, приведенным в таблице С.15.

Таблица С.15 — Определение битов полей дескриптора

Бит	Определение	Значение и смысл бита
0	Поддерживает аппаратную установку пути	0 = путь связи не может быть установлен; 1 = путь связи может быть установлен
1	Поддерживают пронумерованные строки	0 = пронумерованные строки не поддерживаются; 1 = пронумерованные строки поддерживаются и могут быть прочитаны
2	Поддерживается масштабирование	0 = масштабирование не поддерживается. 1 = масштабирование поддерживается. Атрибуты масштабирования установлены и значение представлено пользователю в технических единицах.

Окончание таблицы С.15

Бит	Определение	Значение и смысл бита
3	Поддерживаются связи с масштабированием	0 = связи с масштабированием не поддерживаются. 1 = значения атрибутов масштабирования могут быть извлечены из других параметров
4	Только считывание параметра	0 = значение параметра может быть записано (установлено) и считано (получено). 1 = значение параметра может быть только считано (получено), но не установлено
5	Мониторинг параметра	0 = значение параметра не обновляется устройством в реальном времени. 1 = значение параметра обновляется устройством в реальном времени
6	Поддерживается повышенная точность масштабирования	0= повышенная точность масштабирования не поддерживается. 1 = повышенная точность масштабирования должна использоваться и значения представляются пользователю в технических единицах
7	Поддерживается непоследовательная нумерация строк	0 = непоследовательная нумерация строк не поддерживается. 1 = непоследовательная нумерация строк поддерживается
8	Допускаются как нумерация, так и отдельные значения	0 = нумерация и отдельные значения не поддерживаются. 1 = нумерация и отдельные значения поддерживаются
9-15	Зарезервировано	Эти биты зарезервированы и должны быть установлены в 0

Старые версии файлов EDS могут использовать идентификаторы типа данных, указанные в таблице С.16.

Таблица С.16 – Идентификаторы типа данных (устаревшие)

Идентификатор типа данных	Определение	Описание типа данных
1	WORD	16-битовое слово
2	UINT	16-битовое целое число без знака
3	INT	16-битовое целое число со знаком
4	BOOL	Булево значение
5	SINT	Короткое целое число
6	DINT	Двойное целое число
7	LINT	Длинное целое число
8	USINT	Короткое целое число без знака
9	UDINT	Двойное целое число без знака
10	ULINT	Длинное целое число без знака
11	REAL	Формат с одной плавающей точкой (IEEE 754)
12	LREAL	Формат с двойной плавающей точкой (IEEE 754)
13	ITIME	Длительность (короткая)
14	TIME	Длительность
15	FTIME	Длительность (высокое разрешение)
16	LTIME	Длительность (длинная)
17	DATE	Дата
18	TIME_OF_DAY	Время дня
19	DATE_AND_TIME	Дата и время
20	STRING	Строка 8 битов на символ
21	STRING2	Строка 16 битов на символ
22	STRINGN	Строка N байтов на символ
23	SHORT_STRING	Короткая строка N-байтов на символ
24	BYTE	8-битовая строка
25	DWORD	32- битовая строка
26	LWORD	64- битовая строка

В таблице С.17 установлены описание и специальные требования для входов с минимальными и максимальными значениями на основе типов данных параметров.

Таблица С.17 – Семантика входов с минимальным и максимальным значениями

Тип данных	Описание и семантика	Семантика минимальных значений	Семантика максимальных значений	Требуемое/ Опциональное/ Недопустимое
BYTE	Битовая строка – длина 8 битов	Минимальные и максимальные значения этих типов данных не определены и не должны устанавливаться в файле EDS		Недопустимое
WORD	Битовая строка – длина 16 битов			
DWORD	Битовая строка – длина 32 бита			
LWORD	Битовая строка – длина 64 бита			
STRING[a)]	Строка (индикатор длины 2 байта, 1 байт на символ)	Минимальная длина строки	Максимальная длина строки	Требуемое
STRING2[a)]	Строка (индикатор длины 2 байта, 2 байта на символ)	Минимальная длина строки	Максимальная длина строки	Требуемое
STRINGN[a)]	Строка (индикатор длины 2 байта, N байтов на символ)	Минимальная длина строки	Максимальная длина строки	Требуемое
SHORT_STRING[a)]	Символьная строка (индикатор длины 1 байт, символы 1 байт)	Минимальная длина строки	Максимальная длина строки	Требуемое
EPATH[a)]	Пронумерованный путь	Минимальная длина строки	Максимальная длина строки	Опциональное
Другие типы данных		Минимальное числовое значение,	Максимальное числовое значение,	Опциональное[b)]

Окончание таблицы С.17

Тип данных	Описание и семантика	Семантика минимальных значений	Семантика максимальных значений	Требуемое/ Опциональное/ Недопустимое
		которое может быть присвоено величине данных	которое может быть присвоено величине данных	

[a] Типы данных STRING, STRING2, STRINGN, SHORT_STRING и EPATH не имеют спецификации минимального или максимального значения. Поля минимального и максимального значений используются для представления минимальной и максимальной длины строки или пути. В этих случаях параметр Data Size используется для представления числа байтов, требующихся на символ или кодирование входа.

[b] Если минимальное и/или максимальное значения не установлены, то минимальное и/или максимальное значения величины данных параметра определены в МЭК 61158-5:2003 (тип 2) в соответствии с типом данных параметра.

Поля параметра, перечисленные в таблице С.18, являются необязательными и значащими, только в том случае, если они используются со следующими типами данных: SINT, INT, DINT, LINT, USINT, UINT, UDINT, ULINT, REAL и LREAL. Спецификация этих полей с любым другим типом данных не допускается.

Таблица С.18 – Поля параметров, зарезервированные для типов числовых данных

Поля	Описание
Scaling Multiplier (умножитель масштабирования)	Числовое значение множителя, применяемое к текущему значению данных параметра
Scaling Divider (делитель масштабирования)	Числовое значение делителя, применяемое к текущему значению данных параметра
Scaling Base (база масштабирования)	Числовое значение базы, применяемое к текущему значению данных параметра

Окончание таблицы С.18

Поля	Описание
Scaling Offset (сдвиг масштабирования)	Числовое значение сдвига, применяемое к текущему значению данных параметра
Multiplier Link (множительная связь)	Номер параметра, указывающий на сущность Parameter Object или другой атрибут объекта, содержащий числовое значение множителя, применяемое к текущему значению данных параметра
Divisor Link (делительная связь)	Номер параметра, указывающий на сущность Parameter Object или другой атрибут объекта, содержащий числовое значение делителя, применяемое к текущему значению данных параметра
Base Link (базовая связь)	Номер параметра, указывающий на сущность Parameter Object или другой атрибут объекта, содержащий числовое значение базы, применяемое к текущему значению данных параметра
Offset Link (связь со сдвигом)	Номер параметра, указывающий на сущность Parameter Object или другой атрибут объекта, содержащий числовое значение сдвига, применяемое к текущему значению данных параметра
Decimal Precision (десятичная точность)	Числовое значение точности, применяемое к текущему значению данных параметра

Масштабирование должно выполняться не устройством, содержащим параметр, а средствами дисплея. Если масштабирование поддерживается, средства дисплея должны использовать уравнение, приведенное на рисунке С.5, для определения технического значения параметра (т.е величины на дисплее) исходя из реального значения параметра. Если масштабирование не поддерживается, то величина параметра должна быть представлена на дисплее без изменений.

$$EngValue = \frac{(ActualValue + Offset) * Mult * Base}{Div * 10^{\text{Десятичная точность}}}$$

[a] Если расширенное масштабирование не поддерживается, эта формула должна применяться при Десятичной Точности, равной 0.

Рисунок С.5 — Формула масштабирования параметра

В секции [Params] возможно также наличие второго ключевого слова. Это ключевое слово должно быть использовано для предоставления списка нумерации экземпляров параметра пользователю. Ключевое слово входа для всех пронумерованных параметров должно состоять из набора символов, «Enum», объединенного с десятичным числом из соответствующего входа Param. Каждый вход Enum должен состоять из пар целых чисел и строк.

Пример на рисунке С.6 показывает типичную секцию Parameter (Параметров).

```
[Params]
    Param1 = 0, 1,"20 02", 0x0E94, 1, 1,"Preset","V","User Manual p33", 0, 5, 1, 1, 1, 1, 0, 0, 0,
    0, 0, 2;
    Param2 =                                    $ parameter instance
    0,                                          $ First field shall equal 0
    6, "20 04 24 01 30 03",                     $ path size, path
    0x0A94,                                     $ descriptor - in hex format
    1,                                          $ data type
    1,                                          $ data size
    "Trigger",                                  $ name
    "Hz",                                       $ units
    "User Manual p49",                          $ help string
    0, 2, 0,                                    $ min, max, default data values
    1, 1, 1, 0,                                 $ mult, div, base, offset scaling
    , , , ,                                     $ mult, div, base, offset links not used
    2;                                          $ decimal places
    Param3 =                                    $ not addressable from link
    0, , , 0x0082, 8, 1, "speed control", "", "", 3, 12, 3, ,,,,,,,,;
    Enum3 = 3, "stop", 8, "slow", 12, "fast";
```

Рисунок С.6 — Пример секции [Params]

С.4.1.4.7 Секция Parameter Groups

Секция группы параметров должна определять все группы параметров в устройстве. Каждая группа параметров должна содержать список параметров в группе. Ключевое слово входа каждой группы должно состоять из комбинации набора символов «Group» и номера группы параметров (десятичного), например «Group1». Десятичные номера должны начинаться с единицы и увеличиваться на единицу.

Фактический экземпляр объекта Parameter Group может (не необязательно) применяться в устройстве. Не требуется, чтобы все экземпляры объекта Parameter Group имели соответствующий вход «GroupN» в EDS. Однако если экземпляр объекта Parameter Group существует в узле и

если эта Группа Параметров описана в EDS, то значение «N» в «GroupN» должно быть равно экземпляру объекта Parameter Group.

Поля каждого входа должны содержать имя группы, число членов группы, а также номера экземпляров параметров в этой группе. Секция группы параметров должна содержать поля, указанные в таблице С.19.

Таблица С.19 – Формат группы параметров

Имя поля	Номер поля	Тип данных	Требуемое/ Опциональное
Строка имени группы	1	EDS_Char_Array	Требуемое
Число членов	2	UINT	Требуемое
Параметр	3 и более (число членов + 2)	UINT	Требуемое

На рисунке С.7 приведен пример типичной секции Parameter Group.

```
[Groups]
     Group1 = "Setup", 2, 1, 2;          $ group 1
     Group2 = "Monitor", 2, 2, 3;        $ group 2
     Group3 = "Maintenance", 2, 1, 3;    $ group 3
```

Рисунок С.7 — Пример секции [Groups]

С.4.1.4.8 Секция Assembly

Секция Assembly описывает структуру блока данных. Часто этот блок представляет собой атрибут данных объекта Assembly, однако эта секция EDS может быть использована для описания любых сложных структур. Описание этого блока данных является параллельным механизму, который объект Assembly использует для описания списка своих членов.

Ключевое слово входа «Revision» должно иметь одно 16-битовое поле целого числа, которое должно соответствовать версии (атрибут класса 1) объекта Assembly внутри устройства. Если этот опциональный вход пропущен, версия объекта Assembly должна быть 2.

Ключевое слово для всех объединений должно состоять из одного из следующих наборов символов: «Assem», «ProxyAssem», «ProxiedAssem», скомбинированных с номером экземпляра объекта Assembly (десятичным)

для данного устройства, например «Assem1». Если конкретный экземпляр объекта Assembly адресуем из связи, то должна быть парность один к одному между номером Assem в файле EDS и номером экземпляра Assembly в устройстве. Ключевые слова «ProxyAssem» и «ProxiedAssem» определены в С.4.1.5.3.2 как часть требований модульного EDS.

Каждый вход должен содержать форматированные поля, указанные в таблице С.20.

Таблица С.20 – Формат ключевого слова AssemN

Имя поля	Номер поля	Тип данных	Требуемое/ Опциональное
Имя	1	EDS_Char_Array	Опциональное
Путь	2	EDS_Char_Array	Опциональное
Размер	3	UINT	Условное
Описание	4	WORD	Опциональное
Зарезервировано	5, 6	Пустой	
Размер элемента	7, 9, 11 ...	UINT	Условное
Ссылка на элемент	8, 10, 12 ...	AssemN, ProxyAssemN, ParamN, ProxyParamN UDINT или EPATH	Условное

Первое поле «Name», должно быть строкой, устанавливающей имя блоку данных. Это опциональное поле может быть использовано через интерфейс пользователя.

Второе поле «Path» должно быть строкой, определяющей логический путь. Этот путь должен указывать адрес блока данных в устройстве. Если блок, описанный этим входом AssemN, не адресуем прямо из связи, поле должно быть пустым. Если это поле — нулевая строка « », блок данных должен быть адресуем как атрибут данных (атрибут экземпляра 3) N-го экземпляра объекта Assembly.

Третье поле «Size» должно представлять собой размер блока данных в байтах. Если ни это поле, ни поля «Member Size» / «Member Reference» не присутствуют, размер блока данных должен быть равен 0. Оба этих поля

могут присутствовать, однако поскольку они оба устанавливают размер блока, установленные обоими способами размеры должны быть согласованы.

Четвертое поле «Descriptor» должно быть битовым полем, которое описывает некоторые характеристики Assembly. Биты этого поля следует интерпретировать согласно таблице С.21.

Таблица С.21 – Определение бита поля дескриптора Assembly

Бит	Имя	Значение
0	Allow Value Edit (допускается редактировать значение)	Если этот бит установлен (1), содержание полей ссылок члена объединения, определенное в виде значений, можно редактировать.
		Если восстановлен (0), содержание полей ссылок этих членов нельзя редактировать.
		Если это поле пустое, значение по умолчанию должно быть установлено на (0).
		Ссылками на члены, считающимися значениями, являются те, которые устанавливают либо константу UDINT, либо путь, состоящий из сегментов данных Data Segments
1—15		Зарезервированы

Поля 5 и 6 должны быть зарезервированными и пустыми.

Остальные поля должны быть парными (например, поле «Member Size» составляет пару с полем «Member Reference»), что требует четного полного числа полей. Число пар полей на каждом входе должно быть переменным. Эти пары должны соответствовать списку членов объекта Assembly.

Допустимым значением поля «Ссылка Элемента» должно быть одно из следующих:

- ссылка ParamN или ProxyParamN из секции [Params];
- ссылка AssemN или ProxyAssemN из секции [Assembly];
- строка, представляющая путь (EPATH);
- константа UDINT;
- пустое поле;
- дополнительные значения согласно определению для модульного EDS, приведенному в С.4.1.5.3.2.

Если поле «Member Reference» пустое, число битов, установленное в поле «Member Size», должно быть использовано в качестве заполнителя незначащей информацией в Assembly. Поле «Member Reference», содержащее нулевую строку, должно рассматриваться как пустое поле. Поле «Member Reference» и ему соответствующее поле «Member Size» не должны быть оба пустыми. Если поле «Member Reference» указывает EPATH, этот путь должен состоять либо из логических сегментов (путь к объекту внутри устройства), либо из сегментов данных.

Поле «Member Size» должно иметь единицы битов. Если поле «Member Size» пустое, следует использовать заданный размер, соответствующий полю «Member Reference». Заданный размер входа «Param» должен быть приведен в его шестом поле (размер). Заданный размер входа «Assem» должен быть приведен в его третьем поле (размер).

Элементы должны быть помещены в блок данных начиная с младшего бита, аналогично объекту Assembly. Если поле «Member Size» меньше заданного размера соответствующего поля «Member Reference», должны быть использованы младшие биты соответствующего поля «Member Reference». Если поле «Member Size» больше заданного размера соответствующего поля «Member Reference», за полным элементом должно следовать заполнение нулями до расширения элемента на весь «Member Size». Представленный блок данных должен быть целым числом байтов. Сумма размеров всех элементов должна быть равна полю Размер AssemN (в битах).

На рисунке С.8 приведен пример, показывающий типичную секцию Assembly. В этом примере Assem5 имеет длину 1 байт и значение по умолчанию 0x21.

```
[Params]
    Param1 =
        0,                          $ first field shall equal 0
        6, "20 0F 24 01 30 01",     $ path size, path
        0x0000,                     $ descriptor
        2,                          $ data type : 16-bit WORD
        2,                          $ data size in bytes
        "Idle state",               $ name
        "",                         $ units
        "User Manual p48",          $ help string
```

```
        0, 2, 1,                        $ min, max, default data values
        0, 0, 0, 0,                     $ mult, dev, base, offset scaling not used
        0, 0, 0, 0,                     $ mult, dev, base, offset link not used
        0;                              $ decimal places not used

Param2 =
        0, 6, "20 0F 24 02 30 01",      $ path size, path
        0x0000, 2, 2,
        "Fault state", "", "User Manual p49",
        0, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0;
        [Assembly]
        Revision = 2;
        Assem5 = "configuration", "20 04 24 05 30 03",1,,,,
        4, Param1,
        3, Param2,
        1, ;
```

Рисунок С.8 — Пример секции [Assembly]

Примечание – Ключевое слово «Variant», скомбинированное с десятичным числом (например, «Variant1»), зарезервировано для будущего определения новых типов входов в секции Assembly.

С.4.1.4.9 Раздел Connection Manager

С.4.1.4.9.1 Содержание

Раздел менеджера соединений Connection Manager должен содержать информацию, касающуюся числа типов соединений приложений, которые поддерживают устройство. Этот раздел моделируется как Connection Manager Object. Многие использованные здесь термины определены в МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2). Ключевое слово каждого входа должно быть одним из следующего набора символов: «Connection», «ProxyConnect», «ProxiedConnect», объединенных с числом (десятичным), например «Connection1», «ProxyConnect1» или «ProxiedConnect1». Десятичные числа должны начинаться с 1 и увеличиваться для каждого дополнительного входа «ProxyConnect», «ProxiedConnect». Ключевые слова «ProxyConnect» и «ProxiedConnect» определены в С.4.1.5.3.3 в качестве части требований к модульному EDS.

Каждый вход должен содержать форматированные поля, указанные в таблице С.22.

Таблица С.22 – Формат Connection Manager

Имя поля	Номер поля	Тип данных	Требуемое/ Опциональное
Trigger and transport	1	DWORD	Требуемое
Connection parameters	2	DWORD	Требуемое
O=>T RPI	3	UDINT или ParamN, или ProxyParamN	Опциональное
O=>T size	4	UINT или ParamN, или ProxyParamN	Условное
O=>T format	5	ParamN или ProxyParamN, или AssemN, или ProxyAssemN	Условное
T=>O RPI	6	UDINT или ParamN, или ProxyParamN	Опциональное
T=>O size	7	UINT или ParamN, или ProxyParamN	Условное
T=>O format	8	ParamN или ProxyParamN, или AssemN, или ProxyAssemN	Условное
Config #1 size	9	UINT или ParamN, или ProxyParamN	Опциональное
Config #1 format	10	ParamN или ProxyParamN, или AssemN, или ProxyAssemN	Опциональное
Config #2 size	11	UINT или ParamN, или ProxyParamN	Опциональное
Config #2 format	12	ParamN или ProxyParamN, или AssemN, или ProxyAssemN	Опциональное
Connection name string	13	EDS_Char_Array	Требуемое
Help string	14	EDS_Char_Array	Требуемое
Path	15	EDS_Char_Array	Требуемое

С.4.1.4.9.2 Маска переключения и транспортировки

Присвоение битов в маске переключения и транспортировки должно соответствовать указанному в таблице С.23. Бит должен быть установлен на 1 (включено) для каждого режима переключения, который поддерживает

соединение. Все другие биты должны быть установлены на 0 (выключено). Для бита клиент/сервер: 0=клиент, 1=сервер. Только один из типов транспорта должен быть установлен на 1 (включено).

Таблица С.23 – Присвоение битов в маске переключения и транспорта

Бит	Определение бита
0	Класс 0: нуль
1	Класс 1: повторное детектирование
2	Класс 2: подтверждение
3	Класс 3: проверка
4	Класс 4: отсутствие блокировки
5	Класс 5: отсутствие блокировки, фрагментирование
6	Класс 6: групповая адресация, фрагментирование
7-15	Класс: зарезервировано
16	Переключатель: циклический
17	Переключатель: изменение состояния
18	Переключатель: приложение
19-23	Переключатель: зарезервировано
24	Тип транспорта: только ожидание
25	тип транспорта: только ввод
26	Тип транспорта: исключительный владелец
27	Тип транспорта: резервный владелец
28-30	Зарезервировано
31	Клиент = 0 / сервер = 1

С.4.1.4.9.3 Параметры соединения

Присвоение битов для типа соединения и маски приоритетов должно соответствовать указанному в таблице С.24. Бит должен быть установлен на

1 (включено) для каждого типа соединения и приоритета, подерживаемых соединением. Все другие биты должны быть установлены на 0 (выключено).

Таблица С.24 – Присвоение битов параметров соединения

Бит	Определение бита
0	O=>T поддержка фиксированного размера
1	O=>T поддержка переменного размера
2	T=>O поддержка фиксированного размера
3	T=>O поддержка переменного размера
4 – 5	O=>T число байтов на слот в O=>T пакете данных в реальном времени для соединений блока переходников: 0 = 1 байт; 1 = 2 байта; 2 = 4 байта; 3 = 8 байтов
6 – 7	T=>O число байтов на слот в T=>O пакете данных в реальном времени для соединений блока переходников: 0 = 1 байт; 1 = 2 байта; 2 = 4 байта; 3 = 8 байтов
8 – 10	O=>T формат передачи в реальном времени: 0 = соединение для констант и безрежимное; 1 = использование пакета данных нулевой длины для указания холостого режима; 2 = зарезервировано; 3 = тактовый импульс; 4 = 32-битовый прогон/заголовок холостого хода; 5 - 7 зарезервированы
11	Зарезервировано
12 – 14	T=>O формат передачи в реальном времени:

Окончание таблицы С.24

Бит	Определение бита
12 – 14	0 = соединение для констант и безрежимное; 1 = использование пакета данных нулевой длины для указания холостого режима; 2 = зарезервировано; 3 = тактовый импульс; 4 = 32-битовый прогон/заголовок холостого хода; 5 - 7 зарезервированы
15	Зарезервировано
16	O=>T тип соединения: NULL (НЕОПРЕДЕЛЕННЫЙ)
17	O=>T тип соединения: MULTICAST (МНОГОАДРЕСНЫЙ)
18	O=>T тип соединения: POINT2POINT
19	O=>T тип соединения: зарезервировано
20	T=>O тип соединения: NULL (НЕОПРЕДЕЛЕННЫЙ)
21	T=>O тип соединения: MULTICAST (МНОГОАДРЕСНЫЙ)
22	T=>O тип соединения: POINT2POINT
23	T=>O тип соединения: зарезервировано
24	O=>T приоритет: LOW (НИЗКИЙ)
25	O=>T приоритет: HIGH (ВЫСОКИЙ)
26	O=>T приоритет: SCHEDULED (ПЛАНОВЫЙ)
27	O=>T приоритет: зарезервировано
28	T=>O приоритет: LOW (НИЗКИЙ)
29	T=>O приоритет: HIGH (ВЫСОКИЙ)
30	T=>O приоритет: SCHEDULED (ПЛАНОВЫЙ)
31	T=>O приоритет: зарезервировано

C.4.1.4.9.4 O=>T RPI (Requested Packet Interval)

O=>T RPI должно быть числом микросекунд интервала запрашиваемого пакета. O=>T RPI должно быть UDINT, или Param, или ProxyParam входом из секции [Params], который определяется в UDINT. Если это поле пустое, на O=>T RPI никакие ограничения не накладываются.

C.4.1.4.9.5 O=>T size

Размер O=>T должен быть числом байтов, предоставляемых для целевого транспорта. Он не должен включать в себя счет последовательности транспорта. Размер O=>T должен быть UINT или Param, или ProxyParam входом из секции [Params], который определяется в UINT. Если это поле пустое, заданный размер формата O=>T должен использоваться после добавления опционального размера заголовка прогона/холостого хода.

C.4.1.4.9.6 O=>T format

Формат входа O=>T должен определять структуру буфера потребителя для этого соединения. Правильные дескрипторы формата должны быть идентификаторами в файле EDS, включая следующие:

- Param или ProxyParam вход из секции [Params];
- Assem или ProxyAssem вход из секции [Assembly].

Это поле может быть пустым, показывая, что формат потребителя не установлен. Это поле не должно быть пустым, если поле O=>T размер пустое. Формат O=>T не должен включать 32-битовый заголовок, если он присутствует.

C.4.1.4.9.7 T=>O RPI

T=> O RPI должно быть числом микросекунд интервала запрашиваемого пакета. T=> O RPI должно быть UDINT, или Param, или ProxyParam входом из секции [Params], который определяется в UDINT. Если это поле пустое, на T=> O RPI никакие ограничения не накладываются.

C.4.1.4.9.8 T=>O size

Размер T=> O должен быть числом байтов, предоставляемых для целевого транспорта. Он не должен включать в себя счет последовательности транспорта. Размер T=> O должен быть UINT или Param, или ProxyParam входом из секции [Params], который определяется в UINT. Если это поле пустое, заданный размер формата T=> O должен использоваться после добавления опционального заголовка прогона/ холостого хода.

C.4.1.4.9.9 T=>O format

Формат входа T=> O должен определять структуру буфера потребителя для этого соединения. Правильные дескрипторы формата должны быть идентификаторами в файле EDS, включая следующие:

- Param или ProxyParam вход из секции [Params];

- Assem или ProxyAssem вход из секции [Assembly].

Это поле может быть пустым, показывая, что создаваемый формат не установлен. Это поле не должно быть пустым, если поле размер T=> O пустое. Формат должен включать заголовок статуса, если он присутствует.

C.4.1.4.9.10 Configuration (конфигурация)

Размеры Config #1 и Config #2 должны устанавливать размер сегмента опциональных данных, которые присоединяются к пути Forward_Open. Сегмент данных должен быть конкатенацией двух буферов в соответствии с форматами Config #1 и Config #2. Размеры должны измеряться числом байтов и принадлежать входам UINT или Param, или ProxyParam секции [Params], которые определяются в UINT. Если одно из полей Config пустое, должен быть использован по умолчанию размер формата соответствующего поля Config.

Действительные поля, устанавливающие формат конфигурации Config должны иметь идентификаторами в файле EDS, включая следующие:

- Param или ProxyParam вход из секции [Params];

- Assem или ProxyAssem вход из секции [Assembly].

Эти поля могут быть пустыми, показывая, что формат конфигурации не установлен. Если оба поля конфигурации размера и конфигурации формата пустые, сегменты данных к пути Forward_Open не добавляют.

C.4.1.4.9.11 Connection name string (строка имени соединения)

Инструмент может показывать строку имени соединения (набор символов). Строка имени соединения должна быть уникальной среди всех входов Connection в рамках EDS.

C.4.1.4.9.12 Help string (вспомогательная строка)

Инструмент может показывать текстовый вспомогательный набор символов. Если вспомогательная строка не предоставлена, необходимо использовать нулевую строку, определяемую парой двойных кавычек «» при отсутствии символов между метками.

C.4.1.4.9.13 Path (путь)

Путь содержит ссылку на целевой объект. Путь следует вводить как CIP (EPATH) с использованием заполняющей путь нотации, описанной в МЭК 61158-6:2003 (тип 2), в формате, установленном в C.4.1.3.6. Дополнительно к формату, установленному в C.4.1.3.6, поле пути может также содержать приведенные ниже ссылки:

- Param или ProxyParam входы из секции [Params];
- ключевое слово SLOT;
- ключевое слово SYMBOL_ANSI;
- ключевое слово SLOT_MINUS_ONE.

Входы Param/ProxyParam следует обозначать согласно USINT, UINT или UDINT. Значение Param/ProxyParam следует использовать с обратным порядком байтов для вставки в путь. Ссылки Param/ProxyParam в пути могут быть заключены в скобки согласно рисунку C.9. Если значение Param/ProxyParam заключено в скобки, оно используется локально для пути – тот же вход Param/ProxyParam может иметь другое значение где-либо в EDS. Если Param/ProxyParam не заключено в скобки, значение в EDS должно быть одинаковым.

Ключевое слово SLOT должно быть определено в USINT. Значения, подставляемые в ключевое слово SLOT, должны соответствовать позиции модуля в панели.

Ключевое слово SLOT_MINUS_ONE должно быть определено в USINT. Значения, подставляемые в ключевое слово SLOT_MINUS_ONE, должны соответствовать позиции модуля в панели минус 1.

Ключевое слово SYMBOL_ANSI следует определять согласно расширенному символьному сегменту (см. МЭК 61158-6:2003 (тип 2)), введенному через интерфейс пользователя. Расширенный символьный сегмент должен быть расширенным символом ANSI (тип пути CIP = 0x91).

Например, строка «CAB» должна определяться следующим расширенным символьным сегментом (заполненным): 0x91 0x03 0x43 0x41 0x42 0x00.

C.4.1.4.9.14 Пример секции Connection Manager (информативный)

На рисунке C.9 приведен пример, показывающий типичную секцию Connection Manager (Менеджер Соединения).

```
[Params]
    Param1 =                          $ specifies read buffer
      0, , ,                          $ no path means not directly accessible
      0x0004,                         $ descriptor : support scaling
      8, 1,                           $ USINT, 1 byte
      "Read",                         $ name
      "", "",                         $ units & help string
      64, 95, 64,                     $ min, max, default data values
      1, 1, 1, -63,                   $ mult, div, base, offset scaling
      0, 0, 0, 0, 0;                  $ mult, div, base, offset link & decimal
                                      $(not used)

    Param2 = $ specifies write buffer
      0, , ,                          $ no path means not directly accessible
      0x0004,                         $ descriptor : support scaling
      8, 1,                           $ USINT, 1 byte
      "Write",                        $ name
      "", "",                         $ units & help string
      160, 191, 160,                  $ min, max, default data values
      1, 1, 1, -159,                  $ mult, div, base, offset scaling
      0, 0, 0, 0, 0;                  $ mult, div, base, offset link & decimal
                                      $(not used)

    [Connection Manager]
      Connection1 =
      0x04010002,                     $ trigger & transport
                                      $ class 1, cyclic, exclusive-owner
      0x44244401,                     $ point/multicast & priority & realtime format
                                      $ fixed, 32-bit headers, scheduled,
                                      $ O=>T point-to-point, T=>O multicast
      , 16, ,                         $ O=>T RPI, size, format
      , 12, ,                         $ T=>O RPI, size, format
      , ,                             $ config part 1 (not used)
      , ,                             $ config part 2 (not used)
      "read/write",                   $ connection name
      "",                             $ Help string
      "20 04 24 01 2C [Param2] 2C [Param1]";
```

Рисунок C.9 — Пример секции [Connection Manager]

C.4.1.4.10 Секция Port

Секция Port должна описывать порты, имеющие маршруты CIP и доступные внутри устройства. Каждый имеющий CIP маршрутизацию порт должен иметь соответствующий вход в этой секции. Ключевое слово входа для всех портов должно состоять из набора символов «Port», скомбинированного с десятичным числом, соответствующим сущности объекта порт. Например Port1 является сущностью 1 Port Object.

Примечание – Маршрутизируемый согласно CIP порт – это порт, способный обмениваться сообщениями CIP с другим портом CIP, соединенным с другой связью CIP.

Каждый вход должен содержать форматированные поля, указанные в таблице С.25.

Таблица С.25 – Формат входа в порт

Имя поля	Номер поля	Тип данных	Требуемое/ Опциональное
Port Type Name	1	Ключевое слово поля	Требуемое
Port Name	2	EDS_Char_Array	Опциональное
Port Object	3	EDS_Char_Array	Опциональное
Port Number	4	UINT	Требуемое
Reserved	5, 6	Пусто	Не используется
Port Specific	7, 8, …	Специфичное для порта	Специфичное для порта

Первое поле, называемое «Port Type Name», должно быть одним из следующих ключевых слов поля:

- ControlNet;
- ControlNet_Redundant;
- TCP (для указания имеющего возможности EtherNet/IP TCP порта);
- DeviceNet;
- зависящее от поставщика ключевое слово поля, начинающееся с идентификатора поставщика Vendor ID устройства и символа подчеркивания ('65535_').

Опциональное поле «Port Name» должно быть строкой, содержащей имя порта, и может быть использовано в интерфейсе пользователя. Поле «Port

Object» должно быть путем EPATH, который указывает на определенный объект связи сети, ассоциированный с портом.

Порт номер 1 должен соответствовать порту объединительной панели. Устройства с объединительной панелью, которые не могут определять маршрут сообщений CIP, не должны иметь порт номер 1.

На рисунке С.10 приведен пример, показывающий типичную секцию Port.

```
[Port]
  Port1 = DeviceNet,
          "Port A",                    $ name of port
          "20 03 24 01",               $ instance one of the DeviceNet object
          2;                           $ port number 2
  Port2 = 65535_Chassis,
            "Chassis",                 $ name of port
            "20 9A 24 01",             $ vendor specific backplane object
            1;                         $ port number 1
```

Рисунок С.10 — Пример секции [Port]

С.4.1.5 Требования к файлу модульного EDS

С.4.1.5.1 Общие положения

В данном пункте приведено описание концепции и содержания модульного EDS и установлены требования по применению.

С.4.1.5.2 Модульная секция

С.4.1.5.2.1 Содержание

Секция [Modular] должна описывать систему на основе стойки. Существуют два типа модульных устройств:

- стойки;

- модуль.

С.4.1.5.2.2 Устройство стойки

Секция [Modular], описывающая стойку, должна содержать требуемое ключевое слово «DefineSlotsInRack». Единственное поле этого входа должно быть 16-битовым беззнаковым целым числом (UINT), указывающим число слотов в стойке. Даже если электронный ключ определен для этой стойки, она необязательно будет адресуемой из связи. Ключевое слово SLOT,

использованное в определениях пути в секции [Connection Manager], должно иметь диапазон от 0 до числа слотов минус 1.

Ключевое слово «SlotDisplayRule» необязательно. Единственное поле этого входа должно быть параметром из секции [Params] (только ParamN), которое определяет преобразование между внутренним и внешним номером слота.

На рисунке С.11 приведен пример, показывающий EDS для устройства стойки, включая секцию [Modular].

```
[File]
        DescText = "Wonder Chassis EDS file";
        CreateDate = 09-01-1997;
        CreateTime = 17:23:00;
        Revision = 1.1;
[Device]
        VendCode = 65535;
        VendName = "Widget Works, Inc.";
        ProdType = 101;
        ProdTypeStr = "Widget Works Generic";
        ProdCode = 1;
        MajRev = 1;
        MinRev = 1;
        ProdName = "Widget Chassis";
        Catalog = "1234-chassis";
[Params]
        Param1 =
        0, $ first field shall equal 0
        ,, $ path size,path
        0x0004, $ descriptor
        8, $ data type: 32-bit Unsigned Long Integer
        1, $ data size in bytes
        "Slot Naming Convention", $ name
        "", $ units
        "", $ help string
        0,4,0, $ min,max,default data values
        0,0,0,0, $ mult,dev,base,offset scaling
        0,0,0,0, $ mult,dev,base,offset link not used
        0; $ decimal places not used
        Enum1 = 0,"n/a",1,"0",2,"1",3,"2",4,"3";
[Modular]
        DefineSlotsInRack = 5;
        SlotDisplayRule = Param1;
```

Рисунок С.11 — Модульная секция [Modular], описывающая стойки

С.4.1.5.2.3 Модульное устройство (основные входы)

[Modular] секция, описывающая модуль, должна содержать входы «Width (ширина)» и «Rack (блок)».

Требуемый вход с ключевым словом «Width» должен иметь одно поле, показывающее, сколько слотов стойки используется модулем. Это поле должно быть 16-битовым беззнаковым целым числом (UINT).

Ключевое слово входа для всех стоек, в которые модуль может быть установлен, должно состоять из набора символов, «Rack», скомбинированного с десятичным числом. Числа должны начинаться с 1 для первой стойки и увеличиваться для каждой дополнительной стойки. Поля для входов «Rack» должны соответствовать указанным в таблице С.26.

Таблица С.26 – Формат входа Rack

Имя поля	Номер поля	Тип данных	Требуемое/ Опциональное
Vendor ID	1	UINT	Требуемое
Product Type	2	UINT	Требуемое
Product Code	3	UINT	Требуемое
Major Revision	4	USINT	Требуемое
Minor Revision	5	USINT	Требуемое
Reserved	6, 7, 8	пустые	Не используется
Legal Slot	9, 10, 11 ...	UINT	Требуемое

Поля «Vendor ID», «Product Type», «Product Code», «Major Revision» и «Minor Revision» должны идентифицировать электронный ключ стойки, в которые может быть установлен модуль. Резервное поле должно быть пустым. Поля «Legal Slot» должны указывать слоты, в которые может быть установлен модуль. EDS для модуля должен содержать один вход «Rack» для каждой стойки, в которые данный модуль может быть установлен.

На рисунке С.12 приведен пример, показывающий типичную модульную секцию [Modular].

```
[Modular]
   Width = 1;
   Rack1 =                          $ this module can plug into
      65535, 101, 1, 1, 1,,,,       $ slots 1, 2, 3 and 4 of
      1, 2, 3, 4;                   $ this five slot chassis
```

Рисунок С.12 — Пример секции [Modular]

C.4.1.5.2.4 Модульное устройство (дополнительные входы)

Обзор

В EDS определены дополнительные входы для создания возможности идентификации устройства и проверки ключа устройства для модулей в системе на основе стоек, не поддерживающих CIP.

Для этой цели модульные устройства обычно подразделяются на две категории:

— модули, имеющие соединение связи CIP, соответствующий адресуемый из связи объект идентификации, и помещаемые в слот 0 (например, связные адаптеры);

— модули, не имеющие соединение связи CIP или адресуемый объект идентификации и, следовательно, которые не могут быть помещены в слот 0 (например, модули ввода/вывода).

Примечание – CIP предоставляет другие механизмы идентификации устройства и коммутации устройства в случае модулей, поддерживающих объект идентичности, адресуемый в связи CIP.

Входы для модуля, не имеющего адресуемого из связи объекта идентичности

Секция [Modular], описывающая модуль, не имеющий адресуемого из связи объекта идентичности, может содержать ключевое слово «ExternalID (Внешняя идентичность)». Ключевое слово должно иметь одно поле. Это поле должно быть байтовой строкой, идентифицирующей модуль, которая должна иметь кодировку с использованием такого же формата, который установлен для EPATH.

На рисунке C.13 приведен пример, показывающий типичную Модульную секцию [Modular], описывающую модуль, не имеющий адресуемого из связи объекта идентичности.

```
[Modular]
        Width = 1;
        Rack1 =                                 $ this module can plug into
                65535, 101, 1, 1, 1,,,,         $ slots 1, 2, 3 and 4 of
                1, 2, 3, 4;                     $ this five slot chassis
        Rack2 =
```

```
        65535, 101, 2, 1, 1,,,,
        1, 2, 3, 4, 5, 6, 7;
ExternalID = "12 34";
```

Рисунок С.13 — Пример модульной секции [Modular] (модуль, не имеющий адресуемой связи с объектом идентичности)

Входы для модулей, имеющих соединение связи и помещаемых в слот 0

Модульная секция, описывающая модуль, имеющий соединение связи и помещаемый в слот 0, может содержать любое из указанных ниже ключевых слов входа или их комбинацию.

Ключевое слово «GenericID» должно иметь одно поле. Это поле должно быть байтовой строкой, которая должна быть включена в сегмент данных для соединения модуля вместо ExternalID, когда кодирование нежелательно. Эта байтовая строка должна иметь кодировку с использованием такого же формата, который установлен для EPATH.

Ключевое слово «ExternIDExactMatch» должно иметь одно поле со значением «Да» или «Нет». «Да» показывает, что ExternalID устанавливает одно конкретное устройство, «Нет» — что ExternalID устанавливает одно из набора совместимых устройств. Если ключевое слово «ExternIDExactMatch» пропущено, то условие по умолчанию должно быть следующим: ExternalID устанавливает одно конкретное устройство.

Ключевое слово «Query» должно иметь четыре поля. Первое поле должно быть путем, указывающим адресуемый связью атрибут, содержащий набор внешних идентификаторов по одному для каждого слота стойки, за исключением слота 0. Второе поле должно быть сервисом для использования путем запроса (т. е. 1 – получить все атрибуты или 14 – получить один атрибут). Третье поле должно быть целым числом, которое определяет число байтов, используемых для идентификации каждого модуля, и должно быть в диапазоне 1 – 16. Если модуль с двойными слотами имеется в стойке, внешний идентификатор для этого модуля должен появляться дважды в наборе, возвращаемо в ответе на запрос. Запрос должен адресоваться только на модуль в слоте 0. Четвертое поле должно быть ExternalID, возвращаемое,

когда существует пустой слот с кодировкой такого же формата, который установлен для EPATH.

На рисунке С.14 приведен пример, показывающий типичную секцию [Modular], описывающую модуль, имеющий соединение связи, помещенный в слот 0.

```
[Modular]
Width = 1;
Rack1 =                                  $ this module can only plug into
      65535, 101, 1, 1, 1,,,,            $ slot 0 of this five slot chassis
      0;
Rack2 = 65535, 101, 2, 1, 1,,,, 0;
Query = "20 04 24 07 30 03",1,2,"FF FF";
GenericID = "00 00";
ExternalIDExactMatch = No;
```

Рисунок С.14 — Пример модульной секции [Modular] (модуль с соединением связи в слоте 0)

С.4.1.5.3 Модульные дополнения к основным секциям EDS

С.4.1.5.3.1 Дополнения к секции Parameter

Для описания параметров, которые ретранслируются адаптерным устройством EtherNet/IP на другое устройство, которое не поддерживает протокол CIP, необходимо использовать ключевые слова «ProxyParam» и «ProxiedParam». Примером этого является адаптерный модуль EtherNet/IP (устройство, выполняющее функции доступа к соединению) в блоке с многими слотами ввода/вывода для модуля с аналоговыми вводом/выводом (устройство для которого реализуются функции proxy).

«ProxyParam» должен существовать в EDS для устройства, которое выполняет функции proxy.

Ключевое слово «ProxiedParam» должно существовать в EDS для того устройства, для которого выполняются функции proxy.

Информация в модульной секции [Modular] должна быть использована для создания ассоциации файлов EDS, содержащих ключевые слова «ProxyParam», с файлами EDS, содержащими ключевые слова «ProxiedParam». Эта ассоциация должна существовать, когда оба файла EDS указывают соответствующие входы Rack.

Десятичное число, комбинируемое с «ProxyParam» и «ProxiedParam» следует использовать для указания соответствия между «ProxyParam» и «ProxiedParam». Значения поля соответствующих пар «ProxyParam» и «ProxiedParam» должны быть скомбинированы для составления информации, аналогичной значению поля одного из входов «Param». Эта комбинация должна быть выполнена путем использования значения поля из «ProxyParam», если это значение поля не является ключевым словом «Module». Если значением поля, указанным в «ProxyParam», является «Module», следует использовать значение поля, указанное в «ProxiedParam». Необходимо также указывать значения поля для входов «ProxiedParam», в том случае, если поле в «ProxyParam» не принимает значение «Module», однако эти значения не следует использовать, их следует отмечать только для документации.

В секции [Params] может также существовать другое ключевое слово. Это ключевое слово следует использовать для предоставления минимального, максимального значений и значений по умолчанию, которые следует добавлять к минимальным, максимальным значениям и значениям по умолчанию «ProxyParam». Это ключевое слово входа должно быть «ProxyParamSizeAdder», скомбинированное с десятичным числом из соответствующего входа «ProxyParam». Каждый вход «ProxyParam» должен состоять из полей Minimum Value, Maximum Value Default Value (по умолчанию). Определение этих полей соответствует определениям «Param». Ключевое слово «ProxyParamSizeAdder» предоставляет средства для адаптера соединения модуля (например, «ProxyConnect»), позволяющие добавлять данные адаптера к данным модуля и возвращать комбинированные данные по соединению.

В секции [Param] может также существовать другое ключевое слово, которое соответствует «ProxyParam», «ProxyEnum». «ProxyEnum» имеет определение, аналогичное «Enum», за исключением того, что оно ассоциировано с «ProxyParam» вместо «Param». В секции [Param] может также существовать второе ключевое слово, которое соответствует «ProxiedParam», «ProxiedEnum». «ProxiedEnum» имеет определение,

аналогичное «Enum», за исключением того, что оно ассоциировано с «ProxiedParam» вместо «Param».

C.4.1.5.3.2 Дополнения к секции Assembly

Дополнительные ключевые слова входа

Ключевые слова «ProxyAssem» и «ProxiedAssem» должны быть использованы для описания функциональных блоков, которые выполняют функцию proxy с помощью адаптерного устройства CIP для другого устройства, которое не поддерживает протокол CIP. Примером этого является адаптерный модуль EtherNet/IP (устройство для осуществления proxying соединения) в блоке с множественными слотами входа/выхода, соединяющий его с аналоговым модулем входа/выхода (устройство, на соединении с которым выполняется функция proxy).

Ключевое слово «ProxyAssem» должно существовать в EDS для устройства, выполняющего функцию proxy; ключевое слово «ProxiedAssem» должно существовать в EDS для того устройства, для которого выполняется функция proxy.

Информация в Модульной секции [Modular] должна использоваться для создания ассоциации содержащих ключевые слова «ProxyAssem» файлов EDS с файлами EDS, содержащими ключевые слова «ProxiedAssem». Такая ассоциация должна существовать, в том случае, если оба файла EDS указывают соответствующий вход Rack.

Десятичное число (которое комбинируется с «ProxyAssem» и «ProxiedAssem») следует использовать для указания соответствия между «ProxyAssem» и «ProxiedAssem». Значения поля соответствующих пар «ProxyAssem» и «ProxiedAssem» должны быть скомбинированы для составления информации значения поля, аналогичной одному из входов «Assem». Эта комбинация должна быть выполнена путем использования значения поля из «ProxyAssem», если это значение поля не является одним из ключевых слов «Module» или «ModuleMemberList». Если значением поля, указанным в «ProxyAssem», является «Module», следует использовать значение поля, указанное в «ProxiedAssem». Значение поля «Module» не следует использовать для полей «Member Size» или «Member Reference».

«ModuleMemberList» следует использовать вместо пары полей «Member Size» и «Member Reference». Если значением поля, установленным в «ProxyAssem», является «ModuleMemberList», должны быть использованы все поля «Member Size» и «Member Reference», указанные в «ProxiedAssem». Следует указывать значения поля для входов «ProxiedAssem», чьи значения поля в «ProxyAssem» не равны «Module», однако эти значения поля не следует использовать, их следует указывать только в документации.

Дополнительные ключевые слова поля

Адаптерное соединение блока представляет собой соединение с основанным на блоке адаптерным устройством, которое включает в себя данные из модулей в блоке. Такое соединение может также быть использовано для посылки данных конфигурации и ключей для модулей блока (например, при установке соединения).

Указанные ниже ключевые слова являются дополнительными значениями, допустимыми для поля «Member Reference» в секции Assembly, которые указывают специальные цели, предусмотренные при использовании данных, определенных элементом объединения:

— ExternalID;

— InputSlotMask0 или InputSlotMask1;

— OutputSlotMask0 или OutputSlotMask1;

— ConfigSlotMask0 или ConfigSlotMask1.

Ключевое слово «ExternalID» указывает, что этот член объединения должен содержать либо значение «ExternalID» модульного устройства, если желательно наличие ключа устройства, либо значение «GenericID», определенное в EDS адаптера, если ключ нежелателен.

Ключевое слово «ExternalID», скомбинированное с десятичным числом (например ExternalID2), следует использовать для разрешения применения ключа отдельного устройства для соединений с адаптерным блоком. Десятичное (положительное) число N в «ExternalIDN» указывает слот N в блоке. Ключевое слово «ExternalIDN» указывает, что этот элемент объединения должен содержать либо значение «ExternalID» модульного устройства для слота N, если желательно применение ключа устройства на

данном слоте, либо значение «GenericID», определенное в EDS адаптера, если применение ключа модуля на данном слоте нежелательно.

Примечание – Ключ для слота 0 отсутствует.

Ключевые слова «InputSlotMask0» или «InputSlotMask1» должны показывать расположение входной маски слота в объединении. Входная маска слота представляет собой набор битов, представляющих включение или исключение целевых данных создателем модуля в соединении адаптерного блока. Если используется ключевое слово «InputSlotMask0», бит 0 представляет слот 0, бит 1 — слот 1 и т. д. Если используется ключевое слово «InputSlotMask1», бит 0 представляет слот 1, бит 1 — слот 2 и т. д. «InputSlotMask0» и «InputSlotMask1» не должны быть оба использованы в одном и том же объединении. В этом случае требуется предшествующее поле “Member size”.

Ключевые слова «OutputSlotMask0» или «OutputSlotMask1» должны указывать расположение маски слота вывода в объединении. Выходная маска слота представляет собой набор битов, представляющих включение или исключение целевых данных создателем модуля в соединении адаптерного блока. Если используется ключевое слово «OutputSlotMask0», бит 0 представляет слот 0, бит 1 — слот 1 и т. д. Если используется ключевое слово «OutputSlotMask1», бит 0 представляет слот 1, бит 1 — слот 2 и т. д. «OutputSlotMask0» и «OutputSlotMask1» не должны быть оба использованы в одном и том же объединении. В этом случае требуется предшествующее поле «Member size».

Ключевые слова «ConfigSlotMask0» или «ConfigSlotMask1» должны указывать расположение маски слота конфигурации в объединении. Маска слота конфигурации представляет собой набор битов, представляющих включение или исключение данных конфигурации модуля при определении сервиса соединения с помощью адаптерного блока. Если используется ключевое слово «ConfigSlotMask0», бит 0 в этом наборе представляет слот 0, бит 1 представляет слот 1 и т. д. Если используется ключевое слово «ConfigSlotMask1», бит 0 представляет слот 1, бит 1 — слот 2 и т. д.

«ConfigSlotMask0» и «ConfigSlotMask1» не должны быть оба использованы в одном и том же объединении. В этом случае требуется предшествующее поле «Member size (размер элемента)».

C.4.1.5.3.3 Дополнения к секции Connection Manager

Ключевые слова «ProxyConnect» и «ProxiedConnect» должны быть использованы для описания соединений, на которых выполняется функция proxy с помощью адаптерного устройства CIP, с другим устройством, которое не поддерживает протокол CIP. Примером этого является адаптерный модуль EtherNet/IP (устройство для осуществления proxy соединения) в блоке с множественными слотами входа/выхода, соединяющий его с аналоговым модулем входа/выхода (устройство, на соединении с которым выполняется функция proxy).

Ключевое слово «ProxyConnect» должно существовать в EDS для устройства, выполняющего функцию proxy. В примере, приведенном выше, этим устройством является адаптерный модуль EtherNet/IP.

Ключевое слово «ProxiedConnect» должно существовать в EDS для устройства, для которого выполняется функция proxy. В примере, приведенном выше, этим устройством является аналоговый модуль входа/выхода.

Информация в секции [Modular] должна использоваться для создания ассоциации содержащих ключевые слова «ProxyConnect» файлов EDS с файлами EDS, содержащими ключевые слова «ProxiedConnect». Такая ассоциация должна существовать, если оба файла EDS указывают соответствующий вход Rack.

Десятичное число (комбинируемое с «ProxyConnect» и «ProxiedConnect») должно быть использовано для указания соответствия между «ProxyConnect» и «ProxiedConnect». Значения поля соответствующих пар «ProxyConnect» и «ProxiedConnect» должны быть скомбинированы для составления такой же информации значения поля, которая существует в одном входе «Connection». Эта комбинация должна быть выполнена путем использования значений поля из «ProxyConnect», за исключением тех полей, значения которых равны ключевому слову «Module». В этих случаях следует

использовать значение поля, установленное в ассоциированном «ProxiedConnect». Следует указывать значения поля для входов «ProxiedConnect», чьи значения поля в «ProxyConnect» не равны «Module», однако эти значения поля не следует использовать, их следует указывать только в документации. Значение для поля «ProxyConnect» «строка имени соединения» не должно быть «Module»; «ProxyConnect» должен всегда указывать «строку имени соединения».

C.4.1.5.3.4 Примеры расширенной секции EDS (информативные)

На рисунках C.15 и C.16 приведены примеры, показывающие использование модульных расширений EDS для секций [Parameter], [Assembly] и [Connection Manager].

```
[Params]
    Param1 =                    0,,,0x0010,2,2," Target Error Codes",
                                "","",0,0xFFFF,0,0,0,0,0,0,0,0,0,0;
    ProxyParam1 =               0,,,0x0000,2,2,"input size",
                                "","",Module,Module,Module,0,0,0,0,,,,,0;
    ProxyParamSizeAdder1 = 4,4,4;
[Assembly]
    Assem1 = "connection input format",,,,,,
        32,Param1,
        ,ProxyAssem1,
        ,ProxyAssem2;
    ProxyAssem1 = "real time input format","20 7D 24 SLOT 30 0A",,,,,
    ModuleMemberList;
        ProxyAssem2 = "real time status format","20 7D 24 SLOT 30 0B",,,,,
        ModuleMemberList;
[Connection Manager]
    ProxyConnect1 = 0x010100002, 0x44244401,
        2, 0, , 2, ProxyParam1, Assem1, , , , , "Listen Only", "",
        "01 SLOT_MINUS_ONE 20 04 24 03 2C 04 2C 02";
```

Рисунок C.15 — Пример входов ProxyParam и ProxyAssem

```
[Params]
    ProxiedParam1 = ,,,,,,"input size","","",0,2,2,,,,,,,,,;
[Assembly]
    ProxiedAssem1 = "real time input format",,,,,;
    ProxiedAssem2 = "real time status format",,,,,,16,;
[Connection Manager]
    ProxiedConnect1 = ,,,0,,,,,,,,,,;
```

Рисунок C.16 — Пример соответствующих входов ProxiedParam и ProxiedAssem

С.4.2 Специальные требования к EDS в EtherNet/IP

С.4.2.1 Содержание EDS в EtherNet/IP

В данном подразделе устанавлены требования к кодировке в Электронном бланке данных (EDS) в сетях EtherNet/IP.

В таблице С.27 в общем виде приведена структура секций, которые могут быть представлены в EtherNet/IP EDS, соответствующие принятые разграничители секций и порядок секций в EDS. Некоторые из секций являются общими для ряда основанных на CIP сетей и их специальное применение в EtherNet/IP указано в С.4.2.2, если это необходимо. Другие секции, специфические для EtherNet/IP, рассмотрены в С.4.2.4.

Таблица С.27 – Структура файла EDS в EtherNet/IP

Имя секции	Принятый разграничитель	Размещение	Требуемый/ Опциональный
File Description (описание файла)	[File]	1	Требуемый
Device Description (описание устройства)	[Device]	2	Требуемый
Device Classification (классификация устройства)	[Device Classification]	a)	Требуемый
Parameter Class (класс параметра)	[ParamClass]	a)	Опциональный
Parameters (параметры)	[Params]	a)	Опциональный
Parameter Groups (группы параметров)	[Groups]	a)	Опциональный
Assembly (объединение)	[Assembly]	a)	Опциональный
Connection Characteristics (характеристики соединения)	[Connection Manager]	a)	Условный [b)]

Окончание таблицы С.27

Имя секции	Принятый разграничитель	Размещение	Требуемый/ Опциональный
Port (порт)	[Port]	a)	Опциональный
Modular (модульный)	[Modular]	a)	Опциональный
Vendor Specific (относящийся к поставщику)	[VendorID_vendorspecificke yword]	Последний	Опциональный
a) Размещение групп должно быть только после секций описания устройства и классификации устройства.			
b) Секции необходимы, если применяются соответствующие функциональные характеристики, иначе они могут быть пропущены.			

Затем содержание EtherNet/IP EDS должно быть организовано следующим образом:

— все файлы EtherNet/IP EDS должны включать в себя секцию Device Classification, в которой должен использоваться принятый разграничитель [Device Classification] и которая может быть помещена где-либо после секции File Description;

— опциональные и условные секции, описанные в данной спецификации, могут быть представлены в любом порядке при условии, что опережающие ссылки в файле EDS отсутствуют.

С.4.2.2 Выполнение общих требований CIP

С.4.2.2.1 Секция Device Classification

Для любого соответствующего EtherNet/IP устройства, секция классификации устройства в относящемся к нему файле EDS должна включать в себя хотя бы один вход с ключевым словом ClassN, где первое поле установлено на EtherNet/IP. Дальнейшая подклассификация классификации EtherNet/IP должна быть зарезервирована.

C.4.2.2.2 Секция Port

В секции Port файла EDS, вход PortN, соответствующий применяемому в EtherNet/IP порту, должен быть установлен следующим образом:

— поле «Port Type Name» должно иметь значение «TCP»;

— опциональное поле «Port Object» должно быть установлено на путь объекта интерфейс TCP/IP для этого порта;

— никакие дополнительные требования, кроме указанных в общем подразделе CIP (см. C.4.1.4.10), не помещают в поля «Port Name» и «Port Number».

Примечание – EDS для устройства EtherNet/IP не должно содержать прямую ссылку на объект связи для порта EtherNet/IP (например, объект Link EtherNet/IP), поскольку ссылка может осуществляться через объект TCP/IP Interface для этого порта.

C.4.2.3 Дополнительные требования к кодировке данных

Дополнительных требований к кодировке данных файлов EtherNet/IP EDS не представляют.

C.4.2.4 Дополнительные требования к файлам

Дополнительных требований к файлам для файлов EtherNet/IP EDS не представляют.

Приложение D

(справочное)

Шаблоны профиля PROFINET

D.1 Обзор

PROFINET представляет собой основанную на Ethernet сеть, соответствующую МЭК 61784-1 (издание 1) CP 3/3.

Сеть PROFINET использует описание профиля, соответствующее требованияма ИСО 15745-1. Имя технологии профиля – GSDML (Generic Station Description Markup Language; язык разметки для общего описания станции).

Формат GSDML не обеспечивает описание технологических функций или графического интерфейса пользователя в устройстве. Для этой цели уже установлены рекомендуемые концепции (например, язык описания электронных устройств Electronic Device Description Language (EDDL) в соответствии с МЭК 61804-2).

Путем использования GSDML создается файл GSD (Generic Station Description; общее описание станции). В целях указания отличий от формата PROFIBUS[2)] GSD, описанного в ИСО 15745-3:2003 (приложение B), в данном документе использован термин «основанный на GSDML файл».

Основанный на GSDML файл может содержать более чем одну точку доступа в устройство Device Access Points (DAP). DAP представляет собой специальный модуль, соединяющий устройство с сетью. Это позволяет построить один файл для семейства устройств, совместно использующих одни и те же модули (см. D.4.4.1 и D.4.5).

D.2 Типы данных

В таблице D.1 приведено описание типов данных, используемых в GSDML. Используемые регулярные выражения определены в REC-xml-20001006.

[2)] PROFIBUS – торговая марка PROFIBUS International (PI). Эта информация предоставлена для удобства пользователей комплекса стандартов ИСО 15745 и не означает подтверждения со стороны ИСО торговой марки или какой-либо продукции. Для соответствия настоящему профилю не требуется использования торговой марки PROFIBUS. Для применения торговой марки PROFIBUS требуется разрешение PROFIBUS International.

Таблица D.1 – Типы данных

Имя	Схема определения	Описание
Boolean	gost.e-ecolog.ru	См. REC-xmlschema-2-20010502
Enumeration	GSDML - DeviceProfile-v1.0.xsd	Строка с определенным содержанием
IdT	GSDML - Примитив-v1.0.xsd	Используется для идентификации объекта. Список значений должен соответствовать следующему регурярному выражению (\S(.)*\S)|\S
NormalizedString	gost.e-ecolog.ru	См. REC-xmlschema-2-20010502
Token	gost.e-ecolog.ru	См. REC-xmlschema-2-20010502
RefIdT	GSDML - Примитивы -v1.0.xsd	Используется для ссылки на объект
Unsigned8	GSDML - Примитивы -v1.0.xsd	Беззнаковое целое число в диапазоне 0..255
Unsigned16	GSDML - Примитивы -v1.0.xsd	Беззнаковое целое число в диапазоне 0..65535
Unsigned32	GSDML - Примитивы -v1.0.xsd	Беззнаковое целое число в диапазоне 0.. 4294967295
ValueList	GSDML - Примитивы -v1.0.xsd	Базовый тип для списка, включающего в себя диапазоны величин без знака. Список величин должен соответствовать следующему регулярному выражению: ((\-?\d+\.\.\-?\d+)|(\-?\d+))((\-?\d+\.\.\-?\d+)|(\-?\d+))*

Окончание таблицы D.1

Имя	Схема определения	Описание
		Если диапазон определен, значение слева из «..» должно быть меньше значения справа из «..» (см. пример 1)
SignedValueList	GSDML- Примитивы -v1.0.xsd	Базовый тип для списка, включающего диапазоны величин со знаком. Список величин должен соответствовать следующему регулярному выражению: ((\-?\d+\.\.\-?\d+)|(\-?\d+))((\-?\d+\.\.\-?\d+)|(\-?\d+))* Если диапазон определен, значение слева из «..» должно быть меньше значения справа из «..» (см. пример 2)
Пример 1 – Правильный ValueList содержит: ***1 2 3 4*** ***5 12211*** ***0..6*** ***5..12*** ***0..34 36 38*** ***Пример 2 – Правильный SignedValueList содержит:*** ***1 2 3 4*** ***-12 5 12211*** ***0..6*** ***-5..12*** ***0..34 36 38***		

D.3 Общие правила

D.3.1 Контроль версии

Если файл на основе GSDML уже выпущен, важно сохранять неизменной идентификацию объектов. Следовательно содержание атрибутов, соответствующих указанным ниже выражениям для XPath (см.

REC-xpath-19991116), нельзя изменять в новой версии файла основанного на GSDML:

//DeviceAccessPointItem/@ID

//ModuleList/ModuleItem/@ID

//VirtualSubmoduleItem/@ID//ValueItem/@ID

//GraphicItem/@ID

//CategoryItem/@ID

D.3.2 Правила для имени файлов, основанных на GSDML

Имя файла, основанного на GSDML, должно быть составлено из шести указанных ниже полей в следующем порядке:

— GSDML»;

— ID версии в формате Vx.y, где x и y – беззнаковые целые числа. ID версии относится к ID использованной схемы GSDML;

— имя поставщика;

— имя семейства устройств;

— дата выпуска файла, основанного на GSDML, в формате yyyymmdd;

— «.xml» (расширение файла).

В качестве разграничителей между полями следует использовать символ тире «—» (ASCII 45 десятичное).

Пример – «GSDML-V1.0-Lieferant-ET200X-20030818.xml»

Уже выпущенные файлы нельзя изменять без изменения имени файла. При построении новой версии файла, основанного на GSDML, дата выпуска должна быть изменена.

При установке более чем одной версии файла, основанного на GSDML, техническая система может использовать дату выпуска для определения последней версии.

D.3.3 Расположение схемы в файле, основанном на GSDML

Для системы проверки допустимости XML схемы необходима информация о расположении выбранного файла схемы. Следовательно,

должен быть указан атрибут xsi:schemaLocation корневого элемента профиля ИСО 15745.

Для использования одного и того же расположения для всех файлов, основанных на GSDML, необходимо использовать для файлов схемы относительный путь «..\xsd».

D.3.4 Идентификация объектов

Некоторые элементы GSDML схемы могут быть адресованы с помощью идентификатора. Этот идентификатор является атрибутом с именем «ID». Диапазона этого атрибута указан в D.3.2.

Идентификация объектов должна поддерживаться уникальной для всех элементов одной и той же категории согласно описанию в приведенной в таблице Д.2 (например, вся идентификация объектов для объектов типа «ModuleItem» должна быть уникальной.)

Уникальность идентификаторов ID по всему документу необязательна.

В таблице D.2 указаны адресуемые элементы. В правой графе указаны те объекты, на которые ссылаются элементы в левой графе. Эти ссылки используют соответствующие идентификаторы ID в качестве средства адресации.

Таблица D.2 – Идентификация объекта

Тип	Ссылочный элемент
DeviceAccessPointItem	Зарезервировано для будущего использования
ModuleItem	Атрибут ModuleItemTarget элемента UseableModules/ModuleItemRef (см. D.4.5.5)
VirtualSubmoduleItem	Зарезервировано для будущего использования
ValueItem CategoryItem	Атрибут ValueItemTarget всех элементов Ref Атрибуты CategoryRef и SubCategory1Ref элемента ModuleInfo (см D.4.8.1)
GraphicItem	Производится ссылка из элемента Graphics/GraphicItemRef (см. D.4.8.10)

D.3.5 Поддержка языка

Поддержка языка основана на концепциях XML. Зависящие от языка строки могут поддерживаться в файлах, основанных на GSDML, или

выбранных строках в других файлах. Обе стратегии могут комбинироваться.

Другие строки помещают внутри GSDML в виде словарей. Каждый зависящий от языка текст должен иметь атрибут «TextId», имеющий ссылку на вход в словаре.

Пример 1 –

```
<ChannelDiagItem ErrorType="19">
<Text TextId="ID_COMM_ERROR"/>
</ChannelDiagItem>
<ExternalTextList>
<PrimaryLanguage>
< Text TextId="ID_COMM_ERROR" Value = "Communication error"/>
</PrimaryLanguage>
<Language xml:lang="de">
< Text TextId="ID_COMM_ERROR" Value = "Kommunikationsfehler"/>
</Language>
<Language xml:lang="fr">
< Text TextId="ID_COMM_ERROR" Value = "Erreur de communication"/>
</Language>
</ ExternalTextList>
```

ExternalTextList должен иметь элемент PrimaryLanguage. Используют текстовые строки, определенные в элементе PrimaryLanguage, если текстовая строка в выбранном словаре утеряна. В файлах, основанных на GSDML, первичный язык должен быть английским.

Элементы Language в ExternalTextList должны иметь атрибут «xml:lang» для идентификации выбранного языка. Код для представления имени языка должен соответствовать ИСО 639-1:2002.

Кроме того, строки могут присутствовать во внешних файлах – изменения не являются необходимыми в файлах, основанных на GSDML, для поддержки нового языка. Имя внешнего файла должно быть построено на имени соответствующего основанного на GSDML файла путем присоединения строки «-Text-» и соответствующего ИСО 639-1:2002 двухбуквенного кода.

Пример 2 – «GSDML-V1.0-Lieferant-ET200X-20030818-Text-fr.xml»

Внешние файлы должны быть расположены в подкаталоге по отношению к файлу, основанному на GSDML Имя подкаталога должно быть построено из двухбуквенного кода языка внешнего файла (ИСО 639-1:2002).

Кодировка XML файла (например, UTF-8) не определяется в GSDML. Допускается любая кодировка, соответствующая правилам спецификации XML.

D.3.6 Нотация элементов и атрибутов для расширений схемы

В расширениях GSDML схемы имена элементов и атрибутов должны быть составлены следующим образом:

— первый символ должен быть либо десятичным числом («0» — «9»), либо прописной буквой в диапазоне от «A» до «Z»;

— последующие символы должны быть из диапазонов от «0» до «9» или от «a» до «z». Прописные буквы в диапазоне от «A» до «Z» также могут быть использованы для улучшения читаемости;

— аббревиатуры следует использовать в виде прописных букв, за которыми следует символ «_», если последующими будут один или несколько символов;

— первый символ после символа «_» должен быть или десятичным числом («0» — «9») или прописной буквой в диапазоне от «A» до «Z».

Примечание – В GSD до версии 5 символ «_» использовался для разделения частей ключевых слов с целью приведения ключевых слов к более удобному для чтения виду. Иногда для этой же цели использовались буквы верхнего и нижнего регистров.

D.4 Описание элемента

D.4.1 Общие положения

Используют регулярные выражения согласно определению REC-xml-20001006.

D.4.2 Относящиеся к идентичности устройства элементы

D.4.2.1 DeviceIdentity

Содержит общую информацию об устройстве.

Каждый элемент должен включать в себя атрибуты, указанные в таблице D.3.

Таблица D.3 – Атрибуты элемента DeviceIdentity

Имя	Тип	Применение	Описание
VendorID	NormalizedString	Требуемое	Содержит определяемую поставщиком часть DeviceIdentNumber. Эта величина должна соответствовать следующему регулярному выражению: 0x[0-9,a-f,A-F]{1,4}. VendorID присваивает ассоциация поставщиков. Изготовители устройств должны подать заявление для получения VendorID в соответствующую ассоциацию поставщиков. В случае PROFINET ответственной ассоциацией, присваивающей уникальный VendorID, является владелец торговой марки.
DeviceID	NormalizedString	Требуемое	Содержит зависящую от устройства часть DeviceIdentNumber. DeviceID является уникальным ID для всех устройств поставщика. Поставщик должен поддерживать уникальность данной ID. Значение должно соответствовать следующему регулярному выражению: 0x[0-9,a-f,A-F]{1,4}

D.4.2.2 DeviceIdentity / InfoText

Содержит предназначенную для чтения дополнительную текстовую информацию об устройстве.

Применение: Требуется.

Каждый элемент должен включать в себя не менее одиного атрибута, указанного в таблице D.4.

Таблица D.4 – Атрибуты элемента InfoText

Имя	Тип	Применение	Описание
TextId	Token	Требуемое	Содержит ID текста как ссылку на ExternalTextList (список внешнего текста) (см. D.4.4.24)

D.4.2.3 DeviceIdentity / VendorName

Содержит имя поставщика устройства.

Применение: Требуется.

Каждый элемент должен содержать атрибут указанный в таблице D.5.

Таблица D.5 – Атрибут элемента VendorName

Имя	Тип	Применение	Описание
Value	Token	Требуемое	Содержит имя поставщика устройства

D.4.3 Относящиеся к функции устройства элементы

D.4.3.1 DeviceFunction

Элемент DeviceFunction должен содержать элемент «Family».

Применение: Требуется.

Атрибуты: Нет.

D.4.3.2 DeviceFunction / Family

Устройству должен быть присвоен класс функции. Помимо основного семейства устройству может быть присвоено зависящее от поставщика семейство продуктов.

Каждый элемент должен включать в себя не менее одиного атрибута, указанного в таблице D.6.

Таблица D.6 — Атрибуты элемента Family

Имя	Тип	Применение	Описание
MainFamily	Enumeration	Требуемое	Содержит назначение класса функции. Допускается одно из следующих значений: 0: «Общее»; 1: «Приводы» ; 2: «Переключающие устройства»; 3: «Ввод/вывод»; 4: «Вентили»; 5: «Контроллеры» ; 6: «HMI (интерфейс человек – машина)»; 7: «Кодировщики»; 8: «NC/RC (сетевой компьютер/Дистанционный концентратор)»; 9: «Шлюз»; 10: «Программируемые логические контроллеры»; 11: «Системы идентификации»; 12: «PROFIBUS PA Profile»
ProductFamily	xsd:string	Опциональное	Содержит установленное поставщиком прнадлежность устройства семейству продуктов

D.4.4 Относящиеся к процессу приложения элементы

D.4.4.1 DeviceAccessPointList

GSDML должен содержать информацию относительно одной или более различных точек доступа в одном семействе. Этот элемент содержит список установленных DAP.

Применение: Требуется.

Атрибуты: Нет.

D.4.4.2 ModuleList

Данный список содержит все модули, описанные в основанном на GSDML файле.

Применение: Требуется.

Атрибуты: Нет.

D.4.4.3 ModuleList / ModuleItem

Данный элемент содержит субэлементы для описания характеристик модуля.

Применение: Один или более.

Каждый элемент должен содержать атрибуты, указанные в таблице D.7.

Таблица D.7 — Атрибуты элемента ModuleItem

Имя	Тип	Применение	Описание
ID	IdT	Требуемое	Содержит Уникальный ID для ссылки из элемента UseableModules/ModuleItemRef (см. D.4.5.5). Должен быть совместимым вверх с GSD (см. ИСО 15745-3:2003, приложение B). Значение ID должно соответствовать значению параметра Module_Reference ключевого слова Module в GSD (см. ИСО 15745-3:2003, приложение B)
ModuleIdentNumber	NormalizedString	Требуемое	Содержит номер идентификации модуля. Значение должно соответствовать следующему регулярному выражению 0x[0-9,a-f,A-F]{1,8}

D.4.4.4 ModuleList / ModuleItem / ModuleInfo

См. D.4.7.

D.4.4.5 ModuleList / ModuleItem / VirtualSubmoduleList

См. D.4.6.

D.4.4.6 ModuleList / ModuleItem / Graphics

См. D.4.7.

D.4.4.7 ValueList

Элемент ValueList содержит элементы для присвоения значений текстовым строкам.

Пример: ValueList см. в D.4.7.4.

Применение: Опциональное.

Атрибуты: Нет.

D.4.4.8 ValueList / ValueItem

Элемент ValueItem группирует все объекты значений и на него допускается производить ссылку из элемента «UserDataItem/Data».

Применение: Один или более.

Каждый элемент должен содержать атрибуты, указанные в таблице D.8.

Таблица D.8 — Атрибуты элемента ValueItem

Имя	Тип	Применение	Описание
ID	IdT	Требуемое	Содержит ID элемента ValueItem, на который производится ссылка из атрибута TextTarget элемента Ref

D.4.4.9 ValueList / ValueItem / Help

Элемент Help содержит дополнительную вспомогательную информацию о параметре ValueItem.

Применение: Опциональное.

Каждый элемент должен содержать не менее одного атрибута, указанного в таблице D.9

Таблица D.9 — Атрибут элемента Help

Имя	Тип	Применение	Описание
TextId	Token	Требуемое	Содержит ID текста в качестве ссылки на ExternalTextList (см. D.4.4.24)

D.4.4.10 ValueList / ValueItem / Assignments

Данный элемент содержит неограниченное число элементов "Assign".

Применение: Опциональное.

Атрибуты: Нет.

D.4.4.11 ValueList / ValueItem / Assignments / Assign

Элемент Assign содержит присваивание от содержания параметра до текстового представления.

Применение: Один или более.

Каждый элемент должен содержать атрибуты, указанные в таблице D.10.

Таблица D.10 – Атрибуты элемента Assign

Имя	Тип	Применение	Описание
Content	NormalizedString	Требуемое	Содержит значение, присваиваемое текстовой ссылке. Значение должно соответствовать следующему регулярному выражению: \-?[\d+]{1,20}. Внутри элемента Assignments атрибут Content всех дочерних элементов Assign должен быть уникальным
TextId	Token	Требуемое	Содержит ID зависящего от языка текста в ExternalTextList (см. D.4.4.24). Внутри элемента Assignments атрибут TextId всех дочерних элементов Assign должен быть уникальным

D.4.4.12 ChannelDiagList

Устанавливает список специфических для канала текстов ошибок.

Примечание — Используется для вспомогательной информации.

Применение: Опциональное.

Атрибуты: Нет.

D.4.4.13 ChannelDiagList / ChannelDiagItem

ChannelDiagItem содержит атрибуты для определения типа ошибок конкретного канала.

Применение: Один или более.

Каждый элемент должен содержать атрибуты, указанные в таблице

D.11.

Таблица D.11 — Атрибуты элемента ChannelDiagItem

Имя	Тип	Применение	Описание
ErrorType	Unsigned16	Требуемое	Содержит тип ошибки, там где могут быть присвоены определения диагностического текста

D.4.4.14 ChannelDiagList / ChannelDiagItem / Name

Содержит зависящую от языка текстовую информацию.

Применение: Требуемое.

Каждый элемент должен содержать атрибуты, указанные в таблице D.12.

Таблица D.12 — Атрибуты элемента Name

Имя	Тип	Применение	Описание
TextId	Token	Требуемое	Содержит ID текста в качестве ссылки на ExternalTextList (см. D.4.4.24).

D.4.4.15 ChannelDiagList / ChannelDiagItem / Help

Содержит зависящую от языка вспомогательную информацию.

Применение: Опциональное.

Каждый элемент должен содержать атрибуты, указанные в таблице D.13.

Таблица D.13 — Атрибуты элемента Help

Имя	Тип	Применение	Описание
TextId	Token	Требуемое	Содержит ID текста в качестве ссылки на ExternalTextList (см. D.4.4.24)

D.4.4.16 UnitDiagTypeList

UnitDiagTypeList присваивает диагностические значения специальным сообщениям изготовителя о статусе и ошибках.

Применение: Опциональное.

Атрибуты: Нет.

D.4.4.17 UnitDiagTypeList / UnitDiagTypeItem

Применение: Один или более.

Каждый элемент должен содержать атрибуты, указанные в таблице D.14.

Таблица D.14 – Атрибуты элемента UnitDiagTypeItem

Имя	Тип	Применение	Описание
UserStructureIdentifier	Unsigned16	Требуемое	Описывает UserStructureIdentifier блока запроса тревоги. UserStructureIdentifier должен быть в диапазоне от 0 до 32767. В UnitDiagTypeList (см. D.4.4.16) UserStructureIdentifier должен быть уникальным

D.4.4.18 UnitDiagTypeList / UnitDiagTypeItem / Ref

Элемент Ref содержит информацию об элементе диагностических данных в объекте данные тревоги.

Этот элемент должен иметь атрибуты, определенные в D.4.7.4.

Атрибут «ByteOffset» этого элемента ссылается на блок «additional alarm info» в PDU запроса тревоги – информация заголовка не включена.

D.4.4.19 GraphicsList

Данный элемент содержит список GraphicItems (см. D.4.4.20).

Применение: Опциональное.

Атрибуты: Нет.

D.4.4.20 GraphicsList / GraphicItem

GraphicItem содержит информацию по символическому представлению устройства, модуля или субмодуля.

Применение: Один или более.

Каждый элемент должен содержать атрибуты, указанные в таблице D.15.

Таблица D.15 — Атрибуты элемента GraphicItem

Имя	Тип	Применение	Описание
ID	IdT	Требуемое	Содержит ID элемента GraphicItem, на который производится ссылка из элемента Graphics/GraphicItemRef (см. D.4.8.10)
GraphicFile	String	Требуемое	Содержит имя файла без расширения. Формат зависит от типа GraphicItem, который определяется атрибутом Type элемента GraphicItemRef

D.4.4.21 GraphicsList / GraphicItem / Embedded

Данный элемент используется для описания графической информации внутри файла, основанного на GSDML, в формате SVG (см. REC-svg-20030114).

Примечание – Масштабируемая векторная графика (SVG) представляет собой язык для описания двумерной векторной и смешанной векторно-растровой графики в XML.

Применение: Опциональное

Атрибуты: Нет

D.4.4.22 CategoryList

Данный элемент содержит список элементов CategoryItem (см. D.4.4.23).

Примечание 1 – GSDML допускает построение категорий модулей и субмодулей. Эти категории могут быть использованы для группировки модулей и субмодулей внутри каталога инженерного инструмента. Например, все модули аналогового ввода могут быть помещены в одну секцию каталога. Это упрощает поиск требуемых модулей пользователем или инженерным инструментом.

Примечание 2 – Присвоение категории модуля не влияет на характеристики времени выполнения модуля или субмодуля.

Применение: Опциональное.

Атрибуты: Нет.

D.4.4.23 CategoryList / CategoryItem

CategoryItem определяет информацию внутри одной категории.

Применение: Один или более.

Каждый элемент должен содержать атрибуты согласно таблице D.16.

Таблица D.16 — Атрибуты элемента CategoryItem

Имя	Тип	Применение	Описание
ID	IdT	Требуемое	Содержит ID элемента CategoryItem, на который производится ссылка из атрибута CategoryRef или SubCategory1 Ref элемента ModuleInfo (см. D.4.8.1)
TextId	Token	Требуемое	Содержит ID текста в качестве ссылки на ExternalTextList (см. D.4.4.24)

D.4.4.24 ExternalTextList

ExternalTextList содержит текстовые строки, зависящие от языка.

Применение: Требуемое.

Атрибуты: Нет.

D.4.4.25 ExternalTextList / PrimaryLanguage

Элемент PrimaryLanguage содержит текстовые определения первичного языка, который должен использоваться, если текст на выбранном языке недоступен. Для GSDML первичным языком является английский.

Применение: Требуемое.

Атрибуты: Нет.

D.4.4.26 ExternalTextList / PrimaryLanguage / Text

Элемент PrimaryLanguage.

Применение: Требуемое.

Атрибуты: см. таблицу D.17.

Таблица D.17 – Атрибуты элемента Text

Имя	Тип	Применение	Описание
TextId	xsd:string	Требуемое	Содержит уникальный идентификатор ID для ссылки на текст. TextId должен поддерживаться уникальным внутри родительского элемента PrimaryLanguage или Language
Value	xsd:string	Требуемое	Содержит зависящий от языка текст

D.4.4.27 ExternalTextList / Language

Элемент Language содержит текстовое определение указанного языка.

Применение: Один для каждого языка.

Атрибуты: см. таблицу D.18.

Таблица D.18 — Атрибуты элемента Language

Имя	Тип	Применение	Описание
xml:lang	xsd:language	Требуемое	Определяет язык текста в субэлементах Text/@Value

D.4.4.28 ExternalTextList/ Language / Text

Элемент Language.

Применение: Требуемое.

Атрибуты: См. таблицу D.17.

D.4.5 Элементы, относящиеся к DeviceAccessPoint

D.4.5.1 DeviceAccessPointItem

Данный элемент описывает характеристики DAP.

Применение: Один для каждого DAP.

Каждый элемент должен содержать атрибуты, указанные в таблице D.19.

Таблица D.19 – Атрибуты элемента DeviceAccessPointItem

Имя	Тип	Применение	Описание
ID	ID	Требуемое	Содержит ID элемента
PhysicalSlots	ValueList	Требуемое	Данный список описывает, какие слоты поддерживаются DAP. Слотномер DAP должен быть частью списка
ModuleIdentNumber	NormalizedString	Требуемое	Содержит Module Ident Number модуля. Значение должно соответствовать следующему регулярному выражению: 0x[0-9,a-f,A-F]{1,8}

Продолжение таблицы D.19

Имя	Тип	Применение	Описание
MinDeviceInterval	Unsigned16	Требуемое	Данное время устанавливает минимальный интервал для посылки циклических данных ввода-вывода (IO). Базовый такт часов равен 31,25 мкс. Значение этого элемента содержит множитель базового такта часов (см. пример в конце таблицы)
ImplementationType	NormalizedString	Опциональное	Содержит описание стандартного выполнения в DAP, например решения в виде стандартного программного обеспечения, контроллера или ASIC (Специальной интегральной схемы приложения)
DNS_CompatibleName	NormalizedString	Требуемое	Описывает имя по умолчанию устройства в соответствии с правилами для Системы имен доменов (DNS) согласно RFC 1101 (см. примечание)
ExtendedAddressAssignmentSupported	Boolean	Опциональное	Для PROFINET IO каждое устройство IO должно выполнять Discovery и Configuration ппротоколы (DCP) при назначении адресов IP. Если DAP поддерживает другой способ назначения адресов IP типа Dynamic Host Configuration Protocol (протокол динамической конфигурации хоста) (DHCP)

Окончание таблицы D.19

Имя	Тип	Применение	Описание
			согласно RFC 2131, этот атрибут должен быть установлен на «true». По умолчанию: «fals»
AllowedInSlots	ValueList	Опциональное	Определяет, на какие слоты могут быть поставлены DAP. Если не определено, DAP не может быть поставлено на другие слоты, кроме указанных в атрибуте «FixedInSlots»
FixedInSlots	ValueList	Требуемое	Определяет номер слота в DAP, когда устройство конфигурируется в инженерном инструменте. Для нерезервных DAP только один номер слота допускается указывать в списке. Рекомендуется использовать «0» как слотномер. Если атрибут «AllowedInSlots» используется, определенные в «FixedInSlots» слоты должны быть подмножеством слотов, определенных атрибутом «AllowedInSlots»
ObjectUUID_LocalIndex	Unsigned16	Требуемое	Устанавливает поле Instance объекта UUID

Пример – MinDeviceInterval=32 означает, что устройство способно посылать /получать данные каждую миллисекунду.

Примечание – Атрибут DNS_CompatibleName может быть использован инженерной системой для построения уникального имени станции.

D.4.5.2 ModuleInfo

См. D.4.8.1.

D.4.5.3 IOConfigData

Данный элемент содержит информацию о количестве данных IO.

Применение: Требуемое.

Каждый элемент должен содержать атрибуты, указанные в таблице D.20.

Таблица D.20 – Атрибуты элемента IOConfigData

Имя	Тип	Применение	Описание
MaxInputLength	Unsigned16	Требуемый	Содержит максимальную длину данных в октетах, которая может быть передана из Устройства IO в Контроллер IO. Эта длину определяют как сумму входных данных всех используемых субмодулей соответствующего статуса IO производителя и статуса IO потребителя используемых выходных субмодулей
MaxOutputLength	Unsigned16	Требуемый	Содержит максимальную длину данных в октетах, которая может быть передана из контроллера IO в устройство IO. Эта длину определяют как сумму выходных данных всех используемых субмодулей соответствующего статуса IO производителя и статуса IO потребителя используемых входных субмодулей
MaxDataLength	Unsigned16	Опциональный	Содержит максимальную длину выходных и входных данных в октетах. MaxDataLength должна быть не меньше наибольшей величины MaxInputLength или MaxOutputLength. Она не должна быть больше суммы MaxInputLength и MaxOutputLength.

Окончание таблицы D.20

Имя	Тип	Применение	Описание
			Если ключевое слово не предоставлено, максимальная длина должна быть равна сумме MaxInputLength и MaxOutputLength

D.4.5.4 UseableModules

Элемент UseableModules содержит список модульных ссылок на модули элемента ModuleList. Только модули из этого списка совместимы с DAP.

Инженерный инструмент не должен производить конфигурацию других модулей для данного DAP.

Применение: Требуемое.

Атрибуты: Нет.

D.4.5.5 UseableModules / ModuleItemRef

Данный элемент ссылается на модуль в ModuleList, совместимый с DAP.

Применение: Один или более.

Каждый элемент должен содержать атрибуты согласно таблице D.21.

Таблица D.21 – Атрибуты элемента ModuleItemRef

Имя	Тип	Применение	Описание
ModuleItemTarget	RefIdT	Требуемое	Ссылки на элемент ModuleList/ModuleItem (см. D.4.4.3)
AllowedInSlots	ValueList	Опциональное	Определяет, в какие слоты модуль может быть помещен. Если этот атрибут не определен, модуль может быть помещен в любой слот
UsedInSlots	ValueList	Опциональное	Определяет, в какие слоты модуль помещают по умолчанию. Встроенные модули могут быть удалены с помощью инженерного инструмента. Если используется атрибут AllowedInSlots, слоты, определенные в UsedInSlots, должны быть подмножеством слотов, определенных атрибутом AllowedInSlots

Окончание таблицы D.21

Имя	Тип	Применение	Описание
FixedInSlots	ValueList	Опциональное	Определяет, в какие слоты модуль помещают по умолчанию. Встроенные модули не могут быть удалены с помощью инженерного инструмента. Если используется атрибут AllowedInSlots, слоты, определенные в FixedInSlots, должны быть подмножеством слотов, определенных атрибутом AllowedInSlots

D.4.5.6 VirtualSubmoduleList

Данный элемент содержит список элементов VirtualSubmoduleItem (см. D.4.6.1).

Применение: Требуемое.

Атрибуты: Нет.

D.4.5.7 VirtualSubmoduleList / VirtualSubmoduleItem

См. D.4.6.1.

D.4.5.8 Graphics

См. D.4.7.

D.4.5.9 Graphics / GraphicItemRef

См. D.4.7.

D.4.5.10 ApplicationRelations

Данный элемент содержит информацию относительно отношений приложений, выполняемых с помощью Устройства IO.

Атрибуты VersionInformation необходимы для проверки, соответствует ли структура соединения PDU (iPNIO_D_Connect-REQ-PDU) функциональным характеристикам устройства IO. Инженерный инструмент должен заполнить информацию о версии по соединению PDU с помощью этого атрибута.

Применение: Опциональное.

Каждый элемент должен содержать атрибуты, указанные в таблице D.22.

Таблица D.22 — Атрибуты элемента ApplicationRelations

Имя	Тип	Применение	Описание
AR_BlockVersion	Unsigned16	Требуемый	Содержит информацию о версии блока AR
IOCR_BlockVersion	Unsigned16	Требуемый	Содержит информацию о версии блока IO взаимосвязей коммуникаций (CR), называемых блоком CR
AlarmCR_BlockVersion	Unsigned16	Требуемый	Содержит информацию о версии блока тревожной сигнализации
SubmoduleDataBlockVersion	Unsigned16	Требуемый	Содержит информацию о версии блока данных субмодуля

D.4.5.11 ApplicationRelations / TimingProperties

Данный элемент описывает временное поведение при отправке циклических данных IO.

Применение: Опциональное.

Каждый элемент должен содержать атрибуты, указанные в таблице D.23.

Таблица D.23 — Атрибуты элементов TimingProperties

Имя	Тип	Применение	Описание
SendClock	ValueList	Опциональное	Поддерживает список времени циклов, поддерживаемых DAP при отправке циклических данных. Базовый такт часов равен 31,25 мкс. Каждое значение этого элемента содержит множитель базового такта часов и может быть в диапазоне от 1 до 128 (см. пример в конце таблицы). По умолчанию — 32
ReductionRatio	ValueList	Опциональное	Интервал часов при посылке может быть уменьшен путем сжатия. Этот атрибут описывает поддерживаемые значения коэффициентов сжатия DAP.
			Каждое значение этого элемента должно быть в диапазоне от 1 до 16384
Пример – SendClock= «16, 32, 64» означает, что DAP способен посылать циклические данные с интервалом 500 мкс, 1 мс или 2 мс.			

D.4.6 Элементы подмодулей

D.4.6.1 VirtualSubmoduleItem

Данный элемент описывает характеристики подмодуля в качестве части модуля.

Применение: Требуемое.

Каждый элемент должен содержать атрибуты, указанные в таблице D.24.

Таблица D.24 – Атрибуты элемента VirtualSubmoduleItem

Имя	Тип	Применение	Описание
ID	IdT	Требуемое	Содержит ID элемента
SubmoduleIdentNumber	NormalizedString	Требуемое	Содержит SubmoduleIdentNumber для целей идентификации. При выпуске GSDML значение должно быть установлено равным 0x00000000

D.4.6.2 VirtualSubmoduleItem / IOData

Данный элемент определяет характеристики данных IO субмодуля.

Применение: Требуемое.

Каждый элемент должен содержать атрибуты, указанные в таблице D.25.

Таблица D.25 – Атрибуты элекмента IOData

Имя	Тип	Применение	Описание
IOPS_Length	Unsigned8	Опциональное	Содержит длину IO статуса продьюсера в октетах в качестве части объекта IO данные. По умолчанию: 1 октет
IOCS_Length	Unsigned8	Опциональное	Содержит длину IO статуса потребителя в октетах в качестве части объекта IO данные. По умолчанию: 1 октет

D.4.6.3 VirtualSubmoduleItem / IOData / Input

Определяет входные характеристики субмодуля. Если входные данные имеются, этот элемент имеет элементы DataItem (см. D.4.6.4).

Применение: Опциональное.

Каждый элемент должен содержать атрибуты, указанныеи в таблице D.26.

Таблица D.26 — Атрибуты элемента Input

Имя	Тип	Применение	Описание
Consistency	Enumeration	Опционально	Если данный атрибут установлен на совместимость Item (элемента), то совместимость устанавливается только в DataItem. Если данный атрибут установлен на совместимость всех элементов, то субмодуль обеспечивает совместимость для всех элементов данных. По умолчанию: совместимость элемента.

D.4.6.4 VirtualSubmoduleItem / IOData / Input / DataItem

Элемент DataItem содержит информацию об одном конкретном DataItem.

Применение: Один для каждого DataItem.

Каждый элемент должен содержать атрибуты, указанные в таблице D.27.

Таблица D.27 — Атрибуты элемента DataItem

Имя	Тип	Применение	Описание
DataType	Enumeration	Требуемое	Определяет тип данных элемента данных. Допускается использовать одно из следующих значений: Integer8; Integer16; Integer32; Integer64; Unsigned8; Unsigned16; Unsigned32; Unsigned64; Float32; Float64; Date; TimeOfDay with date indication; TimeOfDay without date indication; TimeDifference with date indication; TimeDifference without date indication; NetworkTime; NetworkTimeDifference; VisibleString; OctetString
Length	Unsigned16	Опциональное	Этот атрибут должен быть использован, если атрибут DataType установлен в VisibleString или OctetString. Этот атрибут устанавливает длину элемента DataItem в октетах[a)]. Если этот атрибут использован для типов данных с заданием

Окончание таблицы D.27

Имя	Тип	Применение	Описание
			фиксированной длины, то длина должна соответствовать определению, приведенному в МЭК 61158-5
UseAsBits	Boolean	Опциональное	Если атрибут установлен на true, инженерная система должна отобразить элемент данных в битовом представлении. По умолчанию fals
TextId	Token	Требуемое	Содержит ID текста в качестве ссылки на ExternalTextList (см. D.4.4.24)
[a] Необходимо, поскольку эти типы данных имеют переменную длину.			

D.4.6.5 VirtualSubmoduleItem / IOData / Output

Элемент вывода устанавливают характеристики вывода подмодуля. При наличии данных вывода, этот элемент содержит элементы DataItem (см. D.4.6.6).

Применение: Опциональное.

Атрибуты: см. D.4.6.3.

D.4.6.6 VirtualSubmoduleItem / IOData / Output / DataItem

Элемент DataItem содержит информацию об одном конкретном DataItem.

Применение: Один для каждого DataItem.

Атрибуты: см. D.4.6.4.

D.4.6.7 VirtualSubmoduleItem / RecordDataList

Данный элемент содержит список ParameterRecordDataItem (см. D.4.7.1).

Атрибуты: Нет.

D.4.6.8 VirtualSubmoduleItem / ModuleInfo

См. D.4.8.1.

D.4.6.9 VirtualSubmoduleItem / Graphics

См. D.4.8.9.

D.4.7 Элементы, относящиеся к RecordDataList

D.4.7.1 ParameterRecordDataItem

Элемент ParameterRecordDataItem описывает структуру данных объекта данных регистрации параметра.

Примечание – Все параметры в ParameterRecordDataItems будут переданы в субмодуль в ходе процедуры запуска устройства IO.

Применение: Один или более.

Каждый элемент должен содержать атрибуты, указанные в таблице D.28.

Таблица D.28 – Атрибуты элемента ParameterRecordDataItem

Имя	Тип	Применение	Описание
Index	Unsigned16	Требуемый	Содержит уникальный ID объекта регистрация данных, который используется для адресации объекта с помощью PROFINET IO Дистанционного вызова процедур (RPC)
Length	Unsigned32	Требуемый	Содержит длину объекта регистрации данных в октетах
TransferSequence	Unsigned16	Опциональный	Данный атрибут контролирует последовательность передачи объектов ParameterRecordData в течение запуска. Атрибут TransferSequence должен быть уникальным в субмодуле. Первый передаваемый объект ParameterRecordData должен иметь установку этого атрибута на 1, следующие объекты должны иметь эти атрибуты, увеличивающиеся в прямой последовательности (без

Окончание таблицы D.28

Имя	Тип	Применение	Описание
			пропусков). Если данный атрибут установлен на 0 (по умолчанию), то последовательность передачи не определена
Пример 1 – Правильные конфигурации: ***TransferSequence (передача последовательности)*** ***Первый объект: 1*** ***Второй объект: 2*** ***Третий объект: 3*** ***Пример 2 – Неправильные конфигурации:*** ***2,3,4: Не начинается с 1*** ***1,2,4: Число «3» пропущено*** ***1,2,2: Многократное использование числа 2.***			

D.4.7.2 RecordDataItem / Name

Элемент Name устанвливает имя для объекта регистрации данных, предназначенное для чтения человеком.

Примечание – Это дает возможность средствам разработки выполнить группировку объектов данных объекта регистрации данных таким образом, чтобы, например, при диалоге можно было использовать это имя в качестве названия диалога.

Применение: Требуемое.

Каждый элемент должен содержать атрибуты, указанные в таблице D.29.

Таблица D.29 – Атрибуты элемента Name

Имя	Тип	Применение	Описание
TextId	Token	Требуемое	Содержит ID текста в качестве ссылки на ExternalTextList (см. D.4.4.24)

D.4.7.3 RecordDataItem / Const

Элемент Const используется для инициализации содержания объекта данных регистрации. Если определение Const не описывает полное содержание объекта данных регистрации, неопределенные поля должны быть установлены на нуль.

Если элемент Const пропущен, объект данных регистрации инициализируется с помощью октетов, установленных на нуль.

Если определен более чем один элемент Const, перекрытие между определениями не допускается.

Применение: Нуль или более.

Каждый элемент должен содержать атрибуты, указанные в таблице D.30.

Таблица D.30 — Атрибуты элемента Const

Имя	Тип	Применение	Описание
ByteOffset	Unsigned32	Опциональное	Содержит сдвиг в октетах объекта ссылочных данных от начала объекта данных регистрации. По умолчанию: 0. Длина определения Const должна быть менее или равна атрибуту Length элемента RecordDataItem
Data	NormalizedString	Требуемое	Содержит контент данных объекта данных регистрации, начиная от ByteOffset (см. пример). Строка разделена на шестнадцатеричные лексемы и должна выполнять правила следующего регулярного выражения (0x[0-9,a-f,A-F][0-9,a-f,A-F],?){[1],}
Пример – 0x10,0xff, 0x4A			

D.4.7.4 VirtualSubmoduleItem / RecordDataList / RecordDataItem / Ref

Данный элемент ссылается на объект данных в блоке данных регистрации.

Так как этот элемент может описывать тот же самый объект, как элемент const, может возникнуть перекрытие. В этом случае необходимо использовать определение Ref.

Применение: Ноль или более.

Каждый элемент должен содержать атрибуты, указанные в таблице D.31.

Таблица D.31 – Атрибуты элемента Ref

Имя	Тип	Применение	Описание
ValueItemTarget	RefIdT	Опциональное	Ссылки на элемент ValueDataItem, содержащий имя и диапазон значений объекта данных
ByteOffset	Unsigned32	Требуемое	Содержит сдвиг в октетах ссылочного объекта данных от начала объекта данных регистрации
BitOffset	NormalizedString	Опциональное	Содержит сдвиг в битах ссылочного объекта данных от начала ссылочного октета. Правильный диапазон: 0..7. По умолчанию: 0. Может быть использован только вместе с типом данных Bit или BitArea
BitLength	Unsigned8	Опциональное	Если тип этого объекта данных BitArea, этот атрибут устанавливает длину данного поля. Атрибут игнорируется, если используется другой тип данных. По умолчанию: 1. BitLength должен быть в диапазоне от 1 до 15
DataType	Enumeration	Требуемое	Устанавливает тип объекта данных. Используется одно из следующих значений:

Продолжение таблицы D.31

Имя	Тип	Применение	Описание
			Bit; BitArea (см. примечание); Integer8; Integer16; Integer32; Signed64; Unsigned8; Unsigned16; Unsigned32; Unsigned64
DefaultValue	NormalizedString	Требуемое	Содержит значение по умолчанию для параметра. Это значение должно соответствовать следующему регулярному выражению \-?[\d+]{1,20}
AllowedValues	SignedValueList	Опциональное	Устанавливает возможные значения этого параметра. Если этот атрибут не используется, должны применяться следующие правила: — если используется атрибут ValueItemTarget, диапазон значений параметра определяется атрибутом Content элемента Assignments ссылочного ValueItem; — если атрибут ValueItemTarget не используется или ссылочный ValueItem не содержит элементы Assignments, диапазон значений параметра определяется типом данных, определенным атрибутом DataType; — если атрибуты AllowedValues и ValueItemTarget используются

Окончание таблицы D.31

Имя	Тип	Применение	Описание
			оба, атрибут AllowedValues должен содержать только значения, определенные как атрибут Content элементов Assignments ссылочного ValueItem; – если ValueItem не содержит элементы Assignments, правильными являются все значения в диапазоне значений типа данных для AllowedValues
Changeable	Boolean	Опциональное	– Если данный Attribute установлен на fals, изменения данного параметра не допускаются. По умолчанию = true
Visible	Boolean	Опциональное	Если данный Attribute установлен на fals, этот параметр не будет отображаться в инженерном инструменте. Установка fals не имеет смысла в комбинации с Changeable = true. Схема не способна воспринимать взаимное исключение. По умолчанию = true
TextId	Token	Требуемое	Содержит ID текста в качестве ссылки на ExternalTextList (см.D.4.4.24)

Примечание – Тип данных BitArea может быть использован для уменьшения размера объекта данных регистрации, поскольку он имеет переменную битовую длину. Например, если параметр может иметь только четыре различных значения, BitArea с длиной 2 бита достаточно. Параметр BitArea может содержать кодировки или отдельные биты.

В следующем примере показано применение различных списков для описания модуля с объектом регистрации данных одного параметра и двух параметров:

Пример –

```
<RecordDataList>
<!--The module has one 20 octet record data object with index “1”. The description of this
record data object is “General Parameter” in the english language.-->
 <ParameterDataRecordItem Index="1" Length="20">
  <Name TextId="General Parameter"/>
  <Ref ValueItemTarget="AlarmEnabling" DataType="Bit" ByteOffset="8" BitOffset="0"
   DefaultValue="0" AllowedValues="0..1" TextId="ID_TEXT1"/>
  <Ref ValueItemTarget="IF Supp" DataType="BitArea" ByteOffset="10" BitOffset="0"
   BitLength="2" DefaultValue="0" AllowedValues="0..2" TextId="ID_TEXT2"/>
 </ParameterDataRecordItem>
</RecordDataList>
 <ValueList>
 <ValueItem ID="AlarmEnabling">
  <NameTextId="ID_ENABLE_ALARMS"/>
                <Assignments>
                 <Assign Content="0" TextId="No"/>
                 <Assign Content="1" TextId="yes"/>
                </Assignments>
</ValueItem>
<ValueItem ID=" IF Supp">
    <NameTextId="ID_IF_SUPP"/>
                <Assignments>
                 <Assign Content="0" TextId="50 Hz"/>
                 <Assign Content="1" TextId="60 Hz"/>
                 <Assign Content="2" TextId="400 Hz"/>
                </Assignments>
 </ValueItem>
</ValueList>
```

D.4.8 Глобально используемые элементы

D.4.8.1 ModuleInfo

Данный элемент содержит общую информацию о модуле или субмодуле.

Каждый элемент должен содержать атрибуты, указанные в таблице D.32.

Таблица D.32 — Атрибуты элемента ModuleInfo

Имя	Тип	Применение	Описание
CategoryRef	RefIdT	Опциональное	Содержит категорию модуля. Значение CategoryRef должно содержать ID элемента CategoryList/CategoryItem (см. D.4.4.23)
SubCategory1Ref	RefIdT	Опциональное	Внутри категории могут быть созданы субкатегории. Значение CategoryRef должно содержать элемент CategoryList/CategoryItem (см. D.4.4.23)

D.4.8.2 ModuleInfo / Name

Элемент Name содержит зависящее от языка имя модуля или субмодуля.

Применение: Требуемое.

Каждый элемент должен содержать атрибуты, указанные в таблице D.33.

Таблица D.33 – Атрибуты элемента Name

Имя	Тип	Применение	Описание
TextId	Token	Требуемое	Содержит ID текста в качестве ссылки на ExternalTextList (см. D.4.4.24)

D.4.8.3 ModuleInfo / InfoText

Элемент InfoText содержит читаемую человеком текстовую информацию о модуле или субмодуле.

Применение: Требуемое.

Каждый элемент должен содержать атрибуты, указанные в таблице D.34.

Таблица D.34 – Атрибуты элемента InfoText

Имя	Тип	Применение	Описание
TextId	Token	Требуемое	Содержит ID текста в качестве ссылки на ExternalTextList (см. D.4.4.24)

D.4.8.4 ModuleInfo / VendorName

Элемент VendorName содержит имя продавца устройства. Если этот элемент отсутствует, следует использовать имя поставщика в элементе «DeviceInfo/Vendorname».

Применение: Опциональное.

Каждый элемент должен содержать атрибуты, указанные в таблице D.35.

Таблица D.35 — Атрибуты элемента VendorName

Имя	Тип	Применение	Описание
Значение	Token	Требуемое	Содержит имя продавца устройства

D.4.8.5 ModuleInfo / OrderNumber

Элемент OrderNumber содержит номер заказа модуля или субмодуля.

Применение: Опциональное.

Каждый элемент должен содержать атрибуты, указанные в таблице D.36.

Таблица D.36 — Атрибуты элемента OrderNumber Номер Заказа)

Имя	Тип	Применение	Описание
Значение	Token	Требуемое	Содержит номер заказа модуля или субмодуля

D.4.8.6 ModuleInfo / HardwareRelease

Элемент HardwareRelease содержит выпуск аппаратуры модуля или субмодуля.

Применение: Опциональное.

Каждый элемент должен содержать атрибуты, указанные в таблице D.37.

Таблица D.37 – Атрибуты элемента HardwareRelease

Имя	Тип	Применение	Описание
Значение	Token	Требуемое	Содержит выпуск аппаратуры модуля или субмодуля.

D.4.8.7 ModuleInfo / SoftwareRelease

Содержит выпуск программного обеспечения модуля/субмодуля.

Применение: Опциональное.

Каждый элемент должен содержать атрибуты, указанные в таблице D.38.

Таблица D.38 – Атрибуты элемента SoftwareRelease

Имя	Тип	Применение	Описание
Значение	Token	Требуемое	Содержит выпуск программного обеспечения модуля или субмодуля

D.4.8.8 ModuleInfo / Family

См. D.4.3.2.

D.4.8.9 Graphics

Элемент Graphics содержит список GraphicItemRef (см. D.4.8.10).

Применение: Опциональное.

Атрибуты: Нет.

D.4.8.10 Graphics / GraphicItemRef

Элемент GraphicItemRef дает ссылку на графическую информацию о модуле или субмодуле устройства.

Применение: Один или более.

Каждый элемент должен содержать атрибуты, указанные в таблице D.39.

Таблица D.39 – Атрибуты элемента GraphicItemRef

Имя	Тип	Применение	Описание
Тип	Enumeration	Требуемое	Описывает тип графического элемента. Может быть использовано одно из следующих значений: DeviceSymbol или DeviceIcon. Внутри одного элемента Graphics должна использоваться только одна ссылка GraphicItemRef одного типа. Если используется тип DeviceSymbol, присвоенный элемент GraphicItem должен быть растровым файлом в формате DIB (70*40 пикселей (ширина*высота) 16 цветов).

Окончание таблицы D.39

Имя	Тип	Применение	Описание
			В случае DeviceAccessPointItem элемент GraphicItem должен представлять устройство. Для модулей и субмодулей элемент GraphicItem должен представлять модуль или субмодуль. Если используется тип DeviceIcon, присвоенный элемент GraphicItem должен быть файлом в формате иконки
GraphicItem Target	RefIdT	Требуемое	Ссылка на элемент GraphicsList/GraphicItem (см. D.4.4.20).

D.5 Схемы GSDML шаблона профиля устройства

D.5.1 Общие положения

Примечание – Приведенные ниже определения схемы используют файл xml.xsd. Этот файл схемы предоставлен World Wide Web Consortium. W3C предлагает загрузить этот файл, используя идентификатор пространства имен в качестве URL.

D.5.2 Схема GSDML профиля устройства (GSDML-DeviceProfile-v1.0.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema targetNamespace="gost.e-ecolog.ru                    /DeviceProfile"
xmlns="gost.e-ecolog.ru                    /DeviceProfile"
xmlns:xsd="gost.e-ecolog.ru                    "
xmlns:base="gost.e-ecolog.ru                    "
elementFormDefault="qualified"
attributeFormDefault="unqualified" version="1.0">
<xsd:import namespace="gost.e-ecolog.ru
schemaLocation="xml.xsd"/>
<xsd:import namespace="gost.e-ecolog.ru                    /Primitives"
schemaLocation="GSDMLPrimitives-
v1.0.xsd"/>
<!--                                                                    -->
<!-- *** ISO 15745 Profile definition ***-->
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
<xsd:key name="ExternalText_ID">
<xsd:selector xpath=".//*/PrimaryLanguage/Text"/>
<xsd:field xpath="@TextId"/>
</xsd:key>
```

```
<xsd:key name="DeviceAccessPointItem_ID">
<xsd:selector xpath=".//*/DeviceAccessPointList/DeviceAccessPointItem"/>
<xsd:field xpath="@ID"/>
</xsd:key>
<xsd:key name="ModuleItem_ID">
<xsd:selector xpath=".//*/ModuleList/ModuleItem"/>
<xsd:field xpath="@ID"/>
</xsd:key>
<xsd:key name="SubModuleItem_ID">
<xsd:selector xpath=".//*/VirtualSubmoduleList/VirtualSubmoduleItem"/>
<xsd:field xpath="@ID"/>
</xsd:key>
<xsd:key name="ValueItem_ID">
<xsd:selector xpath=".//*/ValueList/ValueItem"/>
<xsd:field xpath="@ID"/>
</xsd:key>
<xsd:key name="GraphicItem_ID">
<xsd:selector xpath=".//*/GraphicsList/GraphicItem"/>
<xsd:field xpath="@ID"/>
</xsd:key>
<xsd:key name="CategoryItem_ID">
<xsd:selector xpath=".//*/CategoryList/CategoryItem"/>
<xsd:field xpath="@ID"/>
</xsd:key>
<xsd:keyref name="UseableModuleItemRef" refer="ModuleItem_ID">
<xsd:selector xpath=".//*/ModuleItemRef"/>
<xsd:field xpath="@ModuleItemTarget"/>
</xsd:keyref>
<xsd:keyref name="GraphicsRef" refer="GraphicItem_ID">
<xsd:selector xpath=".//*/GraphicItemRef"/>
<xsd:field xpath="@GraphicItemTarget"/>
</xsd:keyref>
<xsd:keyref name="CategoryRef" refer="CategoryItem_ID">
<xsd:selector xpath=".//*/ModuleInfo"/>
<xsd:field xpath="@CategoryRef"/>
</xsd:keyref>
<xsd:keyref name="ValueItemRef" refer="ValueItem_ID">
<xsd:selector xpath=".//*/Ref"/>
<xsd:field xpath="@ValueItemTarget"/>
</xsd:keyref>
<xsd:keyref name="TextRef" refer="ExternalText_ID">
<xsd:selector xpath=".//DeviceIdentity/*"/>
<xsd:field xpath="@TextId"/>
</xsd:keyref>
<xsd:keyref name="TextRef1" refer="ExternalText_ID">
<xsd:selector xpath=".//DeviceFunction/*"/>
<xsd:field xpath="@TextId"/>
</xsd:keyref>
<xsd:keyref name="TextRef2" refer="ExternalText_ID">
<xsd:selector
xpath=".//DeviceAccessPointList/*|.//ModuleList/*|.//ValueList/*|.//ChannelDiagList/*|.//UnitD
iagType
List/*|.//GraphicsList/*|.//CategoryList/*"/>
<xsd:field xpath="@TextId"/>
</xsd:keyref>
</xsd:element>
<!--_____-->
```

```
<!-- *** ProfileHeader ***-->
<xsd:element name="ProfileHeader">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--_____________________________________________________________________-->
<!-- *** ProfileBody ***-->
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DeviceIdentity" minOccurs="0"/>
<xsd:element ref="DeviceManager" minOccurs="0"/>
<xsd:element ref="DeviceFunction" maxOccurs="unbounded"/>
<xsd:element ref="ApplicationProcess" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element name="ExternalProfileHandle" type="ProfileHandle_DataType"
minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--_____________________________________________________________________-->
<!-- *** DeviceIdentity related ***-->
<xsd:element name="DeviceIdentity">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="InfoText" type="base:LocalizableTextParameterT"/>
<xsd:element name="VendorName" type="base:TokenParameterT"/>
</xsd:sequence>
<xsd:attribute name="VendorID" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
<xsd:pattern value="0x[0-9,a-f,A-F]{1,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="DeviceID" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
<xsd:pattern value="0x[0-9,a-f,A-F]{1,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<!--_____________________________________________________________________-->
<!-- *** DeviceManager related ***-->
```

```
<xsd:element name="DeviceManager"/>
<!--_____-->
<!-- *** DeviceFunction related ***-->
<xsd:element name="DeviceFunction">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Family" type="base:FamilyT"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--_____-->
<!-- *** ApplicationProcess related ***-->
<xsd:element name="ApplicationProcess">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="DeviceAccessPointList"/>
<xsd:element ref="ModuleList"/>
<xsd:element ref="ValueList" minOccurs="0"/>
<xsd:element ref="ChannelDiagList" minOccurs="0"/>
<xsd:element ref="UnitDiagTypeList" minOccurs="0"/>
<xsd:element ref="GraphicsList" minOccurs="0"/>
<xsd:element ref="CategoryList" minOccurs="0"/>
<xsd:element ref="ExternalTextList"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceAccessPointList">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Defines an AccessPoint list of a device.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element ref="DeviceAccessPointItem" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ModuleList">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Defines a module list of a device.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element ref="ModuleItem" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ValueList">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ValueItem" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="base:ObjectT">
<xsd:sequence>
<xsd:element name="Help" type="base:LocalizableTextParameterT" minOccurs="0"/>
<xsd:element name="Assignments" minOccurs="0">
<xsd:complexType>
```

```
<xsd:sequence>
<xsd:element name="Assign" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="base:LocalizableTextParameterT">
<xsd:attribute name="Content" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
<xsd:pattern value="\-?[\d+]{1,20}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ChannelDiagList">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Specifies a list of channel type specific error text with help
information.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="ChannelDiagItem" maxOccurs="unbounded">
<xsd:complexType mixed="true">
<xsd:annotation>
<xsd:documentation>Defines a channel type specific error text with help
information.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="Name" type="base:LocalizableTextParameterT"/>
<xsd:element name="Help" type="base:LocalizableTextParameterT" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="ErrorType" type="base:unsigned16T" use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UnitDiagTypeList">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="UnitDiagTypeItem" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Ref" type="ValueItemReferenceT" maxOccurs="unbounded"/>
```

```
</xsd:sequence>
<xsd:attribute name="UserStructureIdentifier" type="base:unsigned16T" use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="GraphicsList">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Contains a list of graphic items, which can contain either external
references to graphic files or embedded graphic information.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="GraphicItem" maxOccurs="unbounded">
<xsd:complexType mixed="true">
<xsd:annotation>
<xsd:documentation>Contains information about a graphic. An external reference
to a graphics file and optionally embedded graphics information can be
given.</xsd:documentation>
</xsd:annotation>
<xsd:sequence minOccurs="0">
<xsd:element name="Embedded">
<xsd:annotation>
<xsd:documentation>Contains embedded graphics information in SVG
format.</xsd:documentation>
</xsd:annotation>
<xsd:complexType mixed="true">
<xsd:annotation>
<xsd:documentation>This parameter enables embedding graphic
information into the XML document.</xsd:documentation>
</xsd:annotation>
<xsd:complexContent mixed="true">
<xsd:restriction base="xsd:anyType">
<xsd:sequence>
<xsd:any namespace="##any" processContents="lax" minOccurs="0"
maxOccurs="unbounded">
<xsd:annotation>
<xsd:documentation>This element contains graphics
information in SVG (Scalable Vector Graphics) format.</xsd:documentation>
</xsd:annotation>
</xsd:any>
</xsd:sequence>
</xsd:restriction>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="ID" type="base:IdT" use="required"/>
<xsd:attribute name="GraphicFile" type="xsd:string" use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="CategoryList">
<xsd:complexType>
```

```
<xsd:annotation>
<xsd:documentation>Defines a list of text definitions for catalog categories for modules
and submodules.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="CategoryItem" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="base:ObjectT">
<xsd:attribute name="TextId" type="xsd:token" use="required"/>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ExternalTextList">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="PrimaryLanguage">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Text" type="ExternalTextT" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Language" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Text" type="ExternalTextT" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute ref="xml:lang"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--_____________________________________________________________________-->
<!-- *** Globally defined elements ***-->
<xsd:element name="DeviceAccessPointItem">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Represents the Device Access Point for PROFINET IO
Devices.</xsd:documentation>
</xsd:annotation>
<xsd:complexContent>
<xsd:extension base="base:ObjectT">
<xsd:sequence>
<xsd:element name="ModuleInfo" type="ModuleInfoT"/>
<xsd:element name="IOConfigData">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Contains general device specific IO data
definitions.</xsd:documentation>
</xsd:annotation>
<xsd:attribute name="MaxInputLength" type="base:unsigned16T" use="required"/>
```

```
<xsd:attribute name="MaxOutputLength" type="base:unsigned16T" use="required"/>
<xsd:attribute name="MaxDataLength" type="base:unsigned16T" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="UseableModules">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Contains a list of module references which can be used
with this access point.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="ModuleItemRef" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="ModuleItemTarget" type="xsd:string" use="required"/>
<xsd:attribute name="AllowedInSlots" type="base:ValueListT" use="optional"/>
<xsd:attribute name="UsedInSlots" type="base:ValueListT" use="optional"/>
<xsd:attribute name="FixedInSlots" type="base:ValueListT" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="VirtualSubmoduleList" type="VirtualSubmoduleListT"/>
<xsd:element name="Graphics" type="GraphicsReferenceT" minOccurs="0"/>
<xsd:element ref="ApplicationRelations" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="PhysicalSlots" type="base:ValueListT" use="required"/>
<xsd:attribute name="ModuleIdentNumber" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
<xsd:pattern value="0x[0-9,a-f,A-F]{1,8}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="MinDeviceInterval" type="base:unsigned16T" use="required"/>
<xsd:attribute name="ImplementationType" type="xsd:normalizedString" use="optional"/>
<xsd:attribute name="DNS_CompatibleName" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
<xsd:pattern value="([0-9,a-z,A-Z]|[0-9,a-z,A-Z]-)*[0-9,a-z,A-Z]"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="ExtendedAddressAssignmentSupported" type="xsd:boolean"
use="optional" default="false"/>
<xsd:attribute name="AllowedInSlots" type="base:ValueListT" use="optional">
<xsd:annotation>
<xsd:documentation>If the Device AccessPoint can be placed into more than one
slot, this parameter specifies the possible slots.</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
<xsd:attribute name="FixedInSlots" type="base:ValueListT" use="required"/>
<xsd:attribute name="ObjectUUID_LocalIndex" type="base:unsigned16T"
use="required"/>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="ApplicationRelations">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="TimingProperties" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="SendClock" type="base:ValueListT" use="optional"
default="32">
<xsd:annotation>
<xsd:documentation>Defines the minimal clock for sending cyclic data. Basic
clock is 31,25 mircoseconds. The value of this element contains the factor of the basic
clock.</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
<xsd:attribute name="ReductionRatio" type="base:ValueListT" use="optional">
<xsd:annotation>
<xsd:documentation>Contains a list of Values, describing the supported
reduction ratios of an access point </xsd:documentation>
</xsd:annotation>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="AR_BlockVersion" type="base:unsigned16T" use="required"/>
<xsd:attribute name="IOCR_BlockVersion" type="base:unsigned16T" use="required"/>
<xsd:attribute name="AlarmCR_BlockVersion" type="base:unsigned16T" use="required"/>
<xsd:attribute name="SubmoduleDataBlockVersion" type="base:unsigned16T"
use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="ModuleItem">
<xsd:annotation>
<xsd:documentation>Defines the contents of a module in GSDML Device
Description.</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="base:ObjectT">
<xsd:sequence>
<xsd:element name="ModuleInfo">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="ModuleInfoT"/>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="VirtualSubmoduleList" type="VirtualSubmoduleListT">
<xsd:annotation>
<xsd:documentation>Only contains virtual submodules.</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="Graphics" type="GraphicsReferenceT" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="ModuleIdentNumber" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
<xsd:pattern value="0x[0-9,a-f,A-F]{1,8}"/>
```

```
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="VirtualSubmoduleItem">
<xsd:annotation>
<xsd:documentation>Defines the contents of a submodule in GSDML Device
Description.</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="base:ObjectT">
<xsd:sequence>
<xsd:element name="IOData">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Defines the input and output data items for a
submodule.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="Input" minOccurs="0">
<xsd:complexType mixed="true">
<xsd:annotation>
<xsd:documentation>Contains the DataItems used to describe the
input data.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="DataItem" maxOccurs="unbounded">
<xsd:complexType mixed="true">
<xsd:complexContent mixed="true">
<xsd:extension base="DataItemT"/>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="Consistency"
type="base:IODataConsistencyEnumT" use="optional" default="Item consistency"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Output" minOccurs="0">
<xsd:complexType mixed="true">
<xsd:annotation>
<xsd:documentation>Contains the DataItems used to describe the
output data.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="DataItem" maxOccurs="unbounded">
<xsd:complexType mixed="true">
<xsd:complexContent mixed="true">
<xsd:extension base="DataItemT"/>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
```

```
<xsd:attribute name="Consistency"
type="base:IODataConsistencyEnumT" use="optional" default="Item consistency"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="IOPS_Length" type="base:unsigned16T" use="optional"
default="1">
<xsd:annotation>
<xsd:documentation>Length of the IO producer status within an io data
object.</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
<xsd:attribute name="IOCS_Length" type="base:unsigned16T" use="optional"
default="1">
<xsd:annotation>
<xsd:documentation>Length of the IO consumer status within an io data
object.</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="VirtualSubmoduleItem">
<xsd:annotation>
<xsd:documentation>Defines the contents of a submodule in GSDML Device
Description.</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="base:ObjectT">
<xsd:sequence>
<xsd:element name="IOData">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Defines the input and output data items for a
submodule.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="Input" minOccurs="0">
<xsd:complexType mixed="true">
<xsd:annotation>
<xsd:documentation>Contains the DataItems used to describe the
input data.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="DataItem" maxOccurs="unbounded">
<xsd:complexType mixed="true">
<xsd:complexContent mixed="true">
<xsd:extension base="DataItemT"/>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
```

```
<xsd:attribute name="Consistency"
type="base:IODataConsistencyEnumT" use="optional" default="Item consistency"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Output" minOccurs="0">
<xsd:complexType mixed="true">
<xsd:annotation>
<xsd:documentation>Contains the DataItems used to describe the
output data.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="DataItem" maxOccurs="unbounded">
<xsd:complexType mixed="true">
<xsd:complexContent mixed="true">
<xsd:extension base="DataItemT"/>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="Consistency"
type="base:IODataConsistencyEnumT" use="optional" default="Item consistency"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="IOPS_Length" type="base:unsigned16T" use="optional"
default="1">
<xsd:annotation>
<xsd:documentation>Length of the IO producer status within an io data
object.</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
<xsd:attribute name="IOCS_Length" type="base:unsigned16T" use="optional"
default="1">
<xsd:annotation>
<xsd:documentation>Length of the IO consumer status within an io data
object.</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="RecordDataList" minOccurs="0">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>Defines a list of Data Records in a
submodule.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element ref="ParameterRecordDataItem" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ModuleInfo" type="ModuleInfoT" minOccurs="0"/>
<xsd:element name="Graphics" type="GraphicsReferenceT" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="SubmoduleIdentNumber" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
```

```
<xsd:pattern value="0x[0-9,a-f,A-F]{1,8}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<!--_____-->
<!-- *** RecordData related ***-->
<xsd:complexType name="ParameterRecordDataT" mixed="true">
<xsd:sequence>
<xsd:element name="Name" type="base:LocalizableTextParameterT"/>
<xsd:element name="Const" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType mixed="true">
<xsd:attribute name="ByteOffset" type="base:unsigned32T" use="optional" default="0"/>
<xsd:attribute name="Data" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
<xsd:pattern value="(0x[0-9,a-f,A-F][0-9,a-f,A-F],?){1,}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="Ref" type="ValueItemReferenceT" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="Index" type="base:unsigned16T" use="required"/>
<xsd:attribute name="Length" type="base:unsigned32T" use="required"/>
<xsd:attribute name="TransferSequence" type="base:unsigned16T" use="optional"
default="0"/>
</xsd:complexType>
<xsd:element name="ParameterRecordDataItem" type="ParameterRecordDataT">
<xsd:annotation>
<xsd:documentation>This defines a ParameterRecordData element.</xsd:documentation>
</xsd:annotation>
</xsd:element>
<!--_____-->
<!-- *** Multiply used types ***-->
<xsd:complexType name="ValueItemReferenceT">
<xsd:attribute name="ValueItemTarget" type="xsd:string" use="optional"/>
<xsd:attribute name="ByteOffset" type="base:unsigned32T" use="required"/>
<xsd:attribute name="BitOffset" use="optional" default="0">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
<xsd:pattern value="[0-7]"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="BitLength" type="base:unsigned8T" use="optional" default="1"/>
<xsd:attribute name="DataType" type="base:DataTypeEnumT" use="required"/>
<xsd:attribute name="AllowedValues" type="base:SignedValueListT" use="optional"/>
<xsd:attribute name="DefaultValue" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:normalizedString">
<xsd:pattern value="\-?[\d+]{1,20}"/>
```

```
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="Changeable" type="xsd:boolean" use="optional" default="true"/>
<xsd:attribute name="Visible" type="xsd:boolean" use="optional" default="true"/>
<xsd:attribute name="TextId" type="xsd:token" use="required"/>
</xsd:complexType>
<xsd:complexType name="GraphicsReferenceT">
<xsd:annotation>
<xsd:documentation>This type is used for as a reference to one or more items of the
global graphics list.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="GraphicItemRef" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="Type" type="base:GraphicsTypeEnumT" use="required"/>
<xsd:attribute name="GraphicItemTarget" type="xsd:string" use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ModuleInfoT">
<xsd:annotation>
<xsd:documentation>Contains general information about a Module.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="Name" type="base:LocalizableTextParameterT"/>
<xsd:element name="InfoText" type="base:LocalizableTextParameterT"/>
<xsd:element name="Family" type="base:FamilyT" minOccurs="0"/>
<xsd:element name="VendorName" type="base:TokenParameterT" minOccurs="0"/>
<xsd:element name="OrderNumber" type="base:TokenParameterT" minOccurs="0"/>
<xsd:element name="HardwareRelease" type="base:TokenParameterT" minOccurs="0"/>
<xsd:element name="SoftwareRelease" type="base:TokenParameterT" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="CategoryRef" type="xsd:string" use="optional"/>
<xsd:attribute name="SubCategory1Ref" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="VirtualSubmoduleListT">
<xsd:annotation>
<xsd:documentation>Defines a submodule list used in the module.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element ref="VirtualSubmoduleItem"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DataItemT" mixed="true">
<xsd:annotation>
<xsd:documentation>Represents the DataItem used to define the input or output data of a
submodule.</xsd:documentation>
</xsd:annotation>
<xsd:attribute name="DataType" type="base:DataItemTypeEnumT" use="required"/>
<xsd:attribute name="Length" type="base:unsigned16T" use="optional"/>
<xsd:attribute name="UseAsBits" type="xsd:boolean" use="optional" default="false"/>
<xsd:attribute name="TextId" type="xsd:token" use="required"/>
</xsd:complexType>
<xsd:complexType name="ExternalTextT">
<xsd:attribute name="TextId" type="xsd:string" use="required"/>
```

```
<xsd:attribute name="Value" type="xsd:string" use="required"/>
</xsd:complexType>
<!--_____-->
<!-- *** Profile Header Data Types ***-->
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP"/>
<xsd:enumeration value="Process"/>
<xsd:enumeration value="InformationExchange"/>
<xsd:enumeration value="Resource"/>
<xsd:enumeration value="Device"/>
<xsd:enumeration value="CommunicationNetwork"/>
<xsd:enumeration value="Equipment"/>
<xsd:enumeration value="Human"/>
<xsd:enumeration value="Material"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="CSI"/>
<xsd:enumeration value="HCI"/>
<xsd:enumeration value="ISI"/>
<xsd:enumeration value="API"/>
<xsd:enumeration value="CMI"/>
<xsd:enumeration value="ESI"/>
<xsd:enumeration value="FSI"/>
<xsd:enumeration value="MTI"/>
<xsd:enumeration value="SEI"/>
<xsd:enumeration value="USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<!--_____-->
<!-- *** DeviceProfile Data Types ***-->
<xsd:complexType name="ProfileHandle_DataType">
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileLocation" type="xsd:anyURI" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

D.5.3 Схема GSDML профиля коммуникационной сети GSDML-CommNetwork-v1.0.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema
targetNamespace="gost.e-ecolog.ru                                                                    "
xmlns:xsd="gost.e-ecolog.ru                                    "
xmlns="http://www.profibus.com/GSDML/2003/11/CommNetworkProfile" version="1.0">
<!--_____________________________________________________________-->
<!-- *** ISO 15745 Profile definition ***-->
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ProfileHeader"/>
<xsd:element ref="ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--_____________________________________________________________-->
<!-- *** ProfileHeader ***-->
<xsd:element name="ProfileHeader">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--_____________________________________________________________-->
<!-- *** ProfileBody ***-->
<xsd:element name="ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ApplicationLayers"/>
<xsd:element ref="TransportLayers"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--_____________________________________________________________-->
<!-- *** Application Layer related *** -->
<xsd:element name="ApplicationLayers"/>
<!--_____________________________________________________________-->
<!-- ***Transport Layer related *** -->
<xsd:element name="TransportLayers"/>
<!--_____________________________________________________________-->
<!-- *** Profile Header Data Types ***-->
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP"/>
```

```
<xsd:enumeration value="Process"/>
<xsd:enumeration value="InformationExchange"/>
<xsd:enumeration value="Resource"/>
<xsd:enumeration value="Device"/>
<xsd:enumeration value="CommunicationNetwork"/>
<xsd:enumeration value="Equipment"/>
<xsd:enumeration value="Human"/>
<xsd:enumeration value="Material"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="CSI"/>
<xsd:enumeration value="HCI"/>
<xsd:enumeration value="ISI"/>
<xsd:enumeration value="API"/>
<xsd:enumeration value="CMI"/>
<xsd:enumeration value="ESI"/>
<xsd:enumeration value="FSI"/>
<xsd:enumeration value="MTI"/>
<xsd:enumeration value="SEI"/>
<xsd:enumeration value="USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>

</xsd:schema>
```

D.5.4 Схема GSDML примитивов (GSDML-Primitives-v1.0.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema targetNamespace="http://gost.e-ecolog.ru                    "
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="gost.e-ecolog.ru                              /Primitives"
version="1.0">
<xsd:import namespace="gost.e-ecolog.ru
schemaLocation="xml.xsd"/>
<!--_____-->
<!-- *** Base Data Types for GSDML Device Description ***-->
<!--_____-->
<!-- *** Object definition for GSDML ***-->
```

```
<xsd:complexType name="ObjectT">
<xsd:annotation>
<xsd:documentation>Base definition for objects which can be
referred.</xsd:documentation>
</xsd:annotation>
<xsd:attribute name="ID" type="IdT" use="required"/>
</xsd:complexType>
<xsd:simpleType name="IdT">
<xsd:annotation>
<xsd:documentation> Base type for object IDs. Any string without whitespaces at the
beginning and end is allowed.</xsd:documentation>
</xsd:annotation>
<xsd:restriction base="xsd:string">
<xsd:pattern value="(\S(.)*\S)|\S"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="ValueListT">
<xsd:annotation>
<xsd:documentation>Base type for a list including ranges of unsigned
values</xsd:documentation>
</xsd:annotation>
<xsd:restriction base="xsd:string">
<xsd:pattern value="((\d+\.\.\d+)|(\d+))(( \d+\.\.\d+)|( \d+))*"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="SignedValueListT">
<xsd:annotation>
<xsd:documentation>Base type for a list including ranges of signed
values</xsd:documentation>
</xsd:annotation>
<xsd:restriction base="xsd:string">
<xsd:pattern value="((\-?\d+\.\.\-?\d+)|(\-?\d+))(( \-?\d+\.\.\-?\d+)|( \-?\d+))*"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="TokenParameterT">
<xsd:annotation>
<xsd:documentation/>
</xsd:annotation>
<xsd:attribute name="Value" type="xsd:token" use="required"/>
</xsd:complexType>
<!--_____________________________________________________________________-->
<!-- *** Specialized Data Types from Primitives schema ***-->
<xsd:complexType name="LocalizableTextParameterT">
<xsd:attribute name="TextId" type="xsd:token" use="required"/>
</xsd:complexType>
<!--_____________________________________________________________________-->
<!-- *** Data Type aliases to map former GSD Datatypes ***-->
<xsd:simpleType name="unsigned8T">
<xsd:restriction base="xsd:unsignedByte"/>
</xsd:simpleType>
<xsd:simpleType name="unsigned16T">
<xsd:restriction base="xsd:unsignedShort"/>
</xsd:simpleType>
<xsd:simpleType name="unsigned32T">
<xsd:restriction base="xsd:unsignedInt"/>
</xsd:simpleType>
<!--_____________________________________________________________________-->
```

```
<!-- *** Data Type for enumerations ***-->
<xsd:complexType name="FamilyT">
<xsd:annotation>
<xsd:documentation>Sets the Device family for identification
purpose.</xsd:documentation>
</xsd:annotation>
<xsd:attribute name="MainFamily" type="FamilyEnumT" use="required"/>
<xsd:attribute name="ProductFamily" type="xsd:normalizedString" use="optional"/>
</xsd:complexType>
<xsd:simpleType name="FamilyEnumT">
<xsd:annotation>
<xsd:documentation>Defines the possible settings for
Family/MainFamily.</xsd:documentation>
</xsd:annotation>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="General"/>
<xsd:enumeration value="Drives"/>
<xsd:enumeration value="Switching Devices"/>
<xsd:enumeration value="I/O"/>
<xsd:enumeration value="Valves"/>
<xsd:enumeration value="Controllers"/>
<xsd:enumeration value="HMI"/>
<xsd:enumeration value="Encoders"/>
<xsd:enumeration value="NC/RC"/>
<xsd:enumeration value="Gateway"/>
<xsd:enumeration value="PLCs"/>
<xsd:enumeration value="Ident Systems"/>
<xsd:enumeration value="PA Profiles"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="GraphicsTypeEnumT">
<xsd:annotation>
<xsd:documentation>This is a list of possible types of a graphic representation of a
module or submodule.</xsd:documentation>
</xsd:annotation>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="DeviceSymbol"/>
<xsd:enumeration value="DeviceIcon"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="DataItemTypeEnumT">
<xsd:annotation>
<xsd:documentation>Defines the possible Datatypes for
DataItems</xsd:documentation>
</xsd:annotation>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="Integer8"/>
<xsd:enumeration value="Integer16"/>
<xsd:enumeration value="Integer32"/>
<xsd:enumeration value="Integer64"/>
<xsd:enumeration value="Unsigned8"/>
<xsd:enumeration value="Unsigned16"/>
<xsd:enumeration value="Unsigned32"/>
<xsd:enumeration value="Unsigned64"/>
<xsd:enumeration value="Float32"/>
<xsd:enumeration value="Float64"/>
<xsd:enumeration value="Date"/>
```

```
<xsd:enumeration value="TimeOfDay with date indication"/>
<xsd:enumeration value="TimeOfDay without date indication"/>
<xsd:enumeration value="TimeDifference with date indication"/>
<xsd:enumeration value="TimeDifference without date indication"/>
<xsd:enumeration value="NetworkTime"/>
<xsd:enumeration value="NetworkTimeDifference"/>
<xsd:enumeration value="VisibleString"/>
<xsd:enumeration value="OctetString"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="IODataConsistencyEnumT">
<xsd:annotation>
<xsd:documentation>Specifies the consistency behaviour </xsd:documentation>
</xsd:annotation>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="Item consistency"/>
<xsd:enumeration value="All items consistency"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="DataTypeEnumT">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="Bit"/>
<xsd:enumeration value="BitArea"/>
<xsd:enumeration value="Integer8"/>
<xsd:enumeration value="Integer16"/>
<xsd:enumeration value="Integer32"/>
<xsd:enumeration value="Integer64"/>
<xsd:enumeration value="Unsigned8"/>
<xsd:enumeration value="Unsigned16"/>
<xsd:enumeration value="Unsigned32"/>
<xsd:enumeration value="Unsigned64"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:schema
```

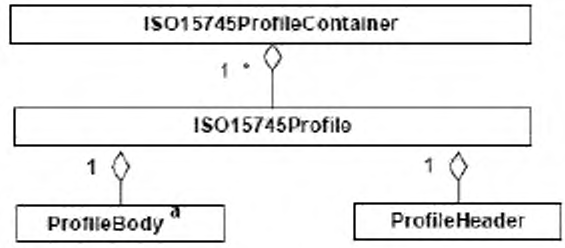
Приложение Е
(обязательное)
Шаблоны профиля Modbus TCP

Е.1 Обзор

Modbus TCP представляет собой основанную на Ethernet систему коммуникаций, описанную в МЭК/ПАС 62030.

В Modbus TCP используется концепция мультипрофильного контейнера, установленная в Изменении 1 к ИСО 15745-4:2003 для XML файлов профиля. Следовательно, шаблоны профиля Modbus TCP основаны на переменном шаблоне ISO15745ProfileContainer, описанном в изменении 1[6] ИСО 15745-1.

На рисунке Е.1 показана структура профиля XML Modbus TCP.



Примечание – Использованы два типа ProfileBody: ProfileBody_Device_ModbusTCP или ProfileBody_CommunicationNetwork_ModbusTCP. Название технологии ProfileTechnology – DDXML (расширяемый язык разметки для описания устройств).

Рисунок Е.1 — Шаблон профиля Modbus TCP

Е.2 Общие правила

Е.2.1 Использование уникальных идентификаторов (ID)

Элемент может иметь атрибут uniqueID (уникальный идентификатор) типа xsd:ID, который должен быть уникальным во всем XML файле. Элемент, ссылающийся на уникальный идентификатор, содержит атрибут с именем типа xsd:IDREF.

Уникальные идентификаторы могут быть созданы двумя способами, один из которых заключается в построении строки из имени элемента и

подсчитываемого номера, а второй – в конкатенации строк родительских элементов. Оба метода обеспечивают уникальность строки.

E.2.2 Поддержка языка

E.2.2.1 Общие положения

Профили устройства, соответствующие XML схеме, описанной в данном приложении, должны поддерживать различные языки, поскольку в этом случае инструменты будут способны использовать имена из файла XML для их отображения на дисплее интерфейса пользователя. Например в средствах интерфейса инструмента пользователя могут быть представлены параметры коммуникации.

Поддержка языка осуществляется с помощью группы меток g_labels. Каждое имя элемента, которое может быть отображено на дисплее, и, следовательно, является зависимым от языка, содержится внутри схемы как элемент g_labels. В качестве опции к элементу метки может быть добавлен в виде атрибута URI.

Пример —

Для данного имени параметра:

- ***немецкий: Baudrate;***
- ***английский: Baud rate;***
- ***французский: Vitesse de transmission.***

E.2.2.2 Элемент g_labels

Группа g_labels (метки) поддерживает введение метки (имени) и описания в контексте родительского элемента (см. рисунок E.2).

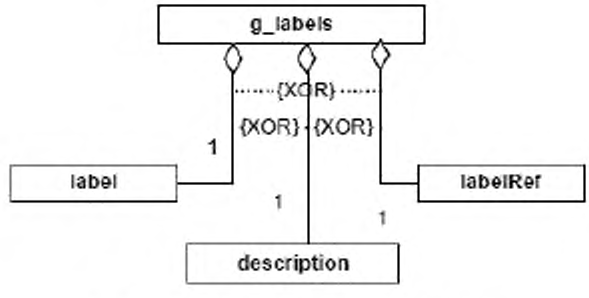


Рисунок E.2 — Группа g_labels

Каждый элемент, для которого необходимо имя или описание, должен выбирать только один из трех элементов, предназначенных для выполнения этой задачи: метку, описание или элемент labelRef (ссылочная метка).

1 Элемент метка позволяет хранить идентифицирующее имя и текст описания внутри самого файла XML. Элемент метка должен иметь атрибуты, указанные в таблице Е.1.

Таблица Е.1 — Атрибуты элемента label (метка)

Атрибут	Тип данных	Применение	Описание
lang	xsd:language	Требуемый	Язык, используемый в имени или описании
URI	xsd:anyURI	Опциональный	Опциональная связь для получения дальнейшей информации описания

Элемент может появляться несколько раз, по одному – для каждого языка. Для идентификации языка используется атрибут lang.

2 Элемент description (описание) обеспечивает возможность хранения текстовых описаний внутри самого файла XML. Элемент может появиться несколько раз, по одному – для каждого языка. Элемент описания имеет те же атрибуты, что и элемент метки.

3 Элемент labelRef (ссылочная метка) допускает возможность хранения ссылочных текстов описания внутри файла – источника внешних текстов.

Элемент labelRef предоставляет указатель с помощью атрибутов dictID (идентификатор словаря) и textID (идентификатор текста) на текстовые входы в отдельном файле-источнике текстов. На эти файлы – источники текста имеется ссылка в субэлементах словаря элемента DeviceFunction (функция устройства). Файлы источники текста могут быть любыми файлами, содержащими последовательности символов и другую информацию, например рисунки.

Элемент labelRef также может появляться несколько раз, позволяя осуществлять ссылки на несколько входов словаря, которые содержат связи с файлами на различных языках. Соответствующие языки определены в атрибуте lang элемента словаря.

Элемент labelRef содержит атрибуты, указанные в таблице Е.2.

Таблица Е.2 — Атрибуты элемента labelRef (ссылочная метка)

Атрибут	Тип данных	Применение	Описание
dictID	xsd:IDREF	требуемое	Производится ссылка на один элемент словаря внутри элемента dictionaryList (список словарей); элемент словаря содержит связь с файлом источником внешнего текста
textID	xsd:string	опциональное	Производится ссылка на последовательность символов внутри файла источника внешнего текста с помощью сравнения с образцом

Е.2.2.3 Идентификатор языка

Для реализации мультиязыковой поддержки каждая метка получает атрибут, содержащий код языка. Код языка соответствует содержанию элемента метки.

С целью проверки, какие языки поддерживаются в файле XML, в атрибуте supportedLanguages (поддерживаемые языки) элемента ProfileBody (тело профиля) содержится список поддерживаемых языков.

Е.2.2.4 Атрибут lang

Идентификатор языка lang состоит из комбинации кода языка (согласно ИСО 639-1), необязательного символа тире, а также необязательного кода страны согласно ИСО 3166-1. Атрибут lang является атрибутом элемента метки.

Некоторые значения lang приведены в таблице Е.3.

Таблица Е.3 — Значения атрибута lang

Язык	значение
Английский (США)	en-us
Немецкий (Стандартный)	de
Французский (Стандартный)	fr
Испанский (Стандартный)	es
Итальянский (Стандартный)	It
Португальский (Бразилия)	pt-br

Е.2.2.5 Атрибут SupportedLanguages (поддерживаемые языки)

Атрибут supportedLanguages определяет поддерживаемые языки и состоит из списка кодов языков, а также необязательных кодов страны.

Пример — supportedLanguages="en-us de fr es"

Е.2.2.6 URI

Общий механизм допускает описание URI в контексте элемента метки. URI реализуется с помощью опционального атрибута URI.

Пример — Используется в контексте метки поставщика, метки параметра, или метки сервиса.

Е.3 Описание ProfileHeader (заголовок профиля)

В целях облегчения идентификации профиля заголовока профиля устройства, а также заголовока профиля коммуникационной сети должны соответствовать представленной на рисунке Е.3 диаграмме по ИСО 15745-1.

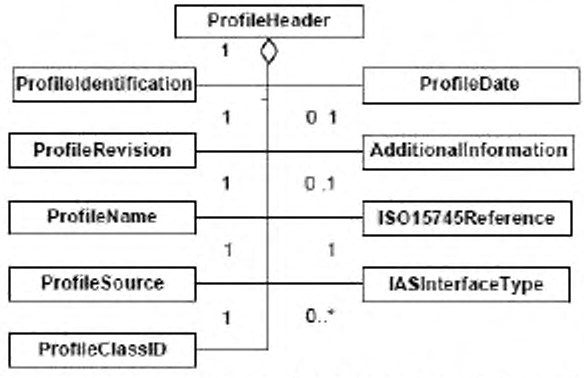


Рисунок Е.3 — Диаграмма класса Profile header (заголовок профиля)

Элемент ProfileHeader состоит из следующих элементов:

— элемента ProfileIdentification (идентификация профиля), идентифицирующего текущий профиль;

— элемента ProfileRevision (версия профиля), идентифицируего текущую версию профиля;

— элемента ProfileName (имя профиля), содержащего описательное имя текущего профиля на английском языке. При наличии в профиле устройства более чем одного элемента ProfileBody, считают, что значение элемента ProfileName (имя профиля) должно быть конкатенацией значений элементов ProductName (имя продукта) внутри соответствующих элементов DeviceIdentity (идентичность устройства);

— элемента ProfileSource (источник профиля), идентифицирующего средство проверки допустимости текущего профиля;

— элемента ProfileClassID (идентификатор класса профиля), идентифицирующего класс текущего профиля согласно ИСО 15745-1;

— элемента ISO 15745 Reference (ссылка ИСО 15745), указывающего на издание ИСО 15745, а также технологию, которым соответствует описание.

E.4 Описание шаблона профиля устройства

E.4.1 ProfileBody_Device_ModbusTCP

В настоящем стандарте определен профиль устройства Modbus TCP.

ProfileBody_Device_ModbusTCP содержит элементы DeviceIdentity (идентичность устройства), DeviceManager (менеджер устройства), DeviceFunction (функция устройства) и ApplicationProcess (прикладной процесс), представленные на рисунке 20.

Элемент ProfileBody содержит описание:

— единичного устройства (например, датчика ближней локации или электромеханического концевого выключателя) или более сложного устройства (например прерывателя цепи с количеством параметров до 2500 и с более чем 100 функциями), или

— части устройства, называемой также «модуль» в области программируемых логических контроллеров (например, части контроллера ввода/вывода или блока электрической защиты).

Элемент ProfileBody содержит атрибуты, указанные в таблице E.4.

Таблица Е.4 — Атрибуты элемента ProfileBody (тело профиля)

Атрибут	Тип данных	Применение	Описание
formatName	xsd:string	фиксир	Идентификатор формата
formatVersion	xsd:string	фиксир	Идентификатор версии формата
filename	xsd:string	требуемое	Имя файла с расширением без пути
fileCreator	xsd:string	требуемое	Лицо, разработавшее файл
fileCreationDate	xsd:date	требуемое	Дата создания файла
fileCreationTime	xsd:time	опциональное	Время создания файла
fileModifiedBy	xsd:string	опциональное	Лицо, изменившее файл
fileModificationDate	xsd:date	опциональное	Дата последнего изменения файла
fileModificationTime	xsd:time	опциональное	Время последнего изменения файла
fileVersion	xsd:string	требуемое	Установленная поставщиком версия файла
supportedLanguages	xsd:NMTOKENS	опциональное	Список поддерживаемых языков

E.4.2 DeviceIdentity (идентичность устройства)

E.4.2.1 Общие положения

Класс DeviceIdentity (см. рисунок 21) содержит элементы, не зависящие от сети и процесса, и описывает идентичность одного устройства или группы устройств.

Таблица Е.5 устанавливает атрибут readOnly (только чтение), который присоединяется к элементам vendorName (имя поставщика), vendorID (идентификатор поставщика), vendorText (текст поставщика), deviceFamily (семейство устройства), productFamily (семейство продуктов), productName (имя продукта), productID (идентификатор продукта), productText (текст продукта), orderNumber (номер заказа), version (версия), specificationRevision (спецификация версии) и instanceName (имя экземпляра).

Таблица Е.5 — Атрибут элемента vendorName (имя поставщика)

Атрибут	Тип данных	Применение	Описание
readOnly	xsd:Boolean	По умолчанию	Указывает, является ли значение для пользователя атрибута только чтение-ложь, истина (по умолчанию)

E.4.2.2 Элемент vendorName (имя поставщика)

Элемент vendorName определяет имя или торговую марку поставщика устройства.

E.4.2.3 Элемент vendorID (идентификатор поставщика)

Элемент vendorID идентифицирует поставщика. Эта информация должна быть заполнена при признании описанного продукта и подтверждении консорциумом.

Примечание — Зависящие от консорциума семейства продуктов и идентификаторы поставщика связаны.

E.4.2.4 Элемент vendorText (текст поставщика)

Элемент vendorText предназначен для предоставления поставщиком дополнительной информации о компании, например адреса или номера горячей линии связи. Группа g_labels предлагает возможность включения URI поставщика в элемент vendorText.

E.4.2.5 Элемент deviceFamily (семейство устройства)

Элемент deviceFamily определяет семейство устройства.

Пример —

Примерами семейства устройства являются следующие:

- ***привод с переменной скоростью;***
- ***прерыватель цепи;***
- ***датчик давления.***

E.4.2.6 Элемент productFamily (семейство продуктов)

Элемент productFamily устанавливает определенную поставщиком принадлежность типа устройства к некоторому набору устройств в семействе. Список действительных значений productFamily - система, инструмент или относящийся к консорциуму.

Примечание — Определяемые консорциумом семейства продуктов и идентификаторы поставщика взаимосвязаны.

E.4.2.7 Элемент productName (имя продукта)

Элемент productName устанавливает определенное поставщиком обозначение или наименование типа устройства.

E.4.2.8 Элемент productID (идентификатор продукта)

Элемент productID устанавливает определенную поставщиком уникальную идентификацию описываемого типа устройства.

E.4.2.9 Элемент productText (текст продукта)

Элемент productText создает для поставщика возможность предоставить короткое текстовое описание типа устройства.

E.4.2.10 Элемент orderNumber (номер заказа)

Элемент orderNumber используется для сохранения единичного номера заказа данного продукта или набора номеров различных заказов продуктов в семействе продуктов в зависимости от того, относится профиль устройства к одному продукту или семейству продуктов.

E.4.2.11 Элемент version (версия)

Элемент version используется для хранения различных типов информации о версиях. Возможны элементы для нескольких версий.

Элемент version имеет атрибуты, указанные в таблице E.6.

Таблица E.6 — Атрибуты элемента version (версия)

Атрибут	Тип данных	Применение	Описание
versionType	xsd:NMToken	требуемое	Тип версии: — SW – программная; — FW - программно-аппаратная; — HW - аппаратная
readOnly	xsd:Boolean	По умолчанию	Указывает, является ли значение для пользователя только для чтения: ложь, истина (по умолчанию)

E.4.2.12 Элемент buildDate (создание даты)

Элемент buildDate определяет дату создания программного блока.

E.4.2.13 Элемент specificationRevision (спецификация версии)

Элемент specificationRevision содержит версию спецификации, которой соответствует устройство.

E.4.2.14 Элемент instanceName (имя экземпляра)

Данный элемент содержит имя экземпляра устройства.

E.4.3 DeviceManager (менеджер устройства)

E.4.3.1 Общие положения

Элемент DeviceManager определяет список индикаторов, предоставленных типом устройства, если они существуют.

E.4.3.2 Элементы indicatorList (список индикаторов) / LEDList (LED список)

E.4.3.2.1 Общие положения

На рисунке E.4 показано число и типы индикаторов, предоставленных в типе устройства.

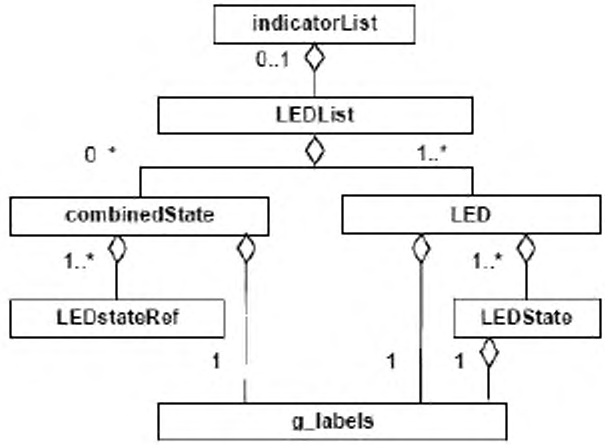


Рисунок E.4 — indicatorList (список индикаторов) / LEDList (LED список)

E.4.3.2.2 LED

Элемент LED описывает характеристики одного LED в данном типе устройства. Подробное описание характеристик может быть представлено в группе g_labels.

Дополнительные характеристики LED представлены в виде атрибутов элемента LED, указанных в таблице Е.7.

Таблица Е.7 — Атрибуты элемента LED

Атрибут	Тип данных	Применение	Описание
LEDcolors	xsd:string	Требуемое	Для цветов LED; правильными значениями являются монохромные и двухцветные
LEDtype	xsd:string	Опциональное	При грубой классификации контролируемых объектов или функциональных параметров; правильные значения - IO, устройство и коммуникации

Дополнительно к приведенному выше элемент LED содержит один или более элементов LEDstate (LED состояние), которые определяют состояния устройства, обеспечиваемые с помощью LED, и визуальные характеристики, используемые для сигнализации о состояниях.

Визуальные параметры, используемые для сигнализации о состояниях, закодированы в виде значений атрибутов элемента LEDstate, указанных в таблице Е.8. Дополнительно состояниям LED присвоены уникальные идентификаторы ID.

Таблица Е.8 — Атрибуты элемента LEDstate (состояние LED)

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный ID для состояния LED; может быть объектом ссылки из элемента LEDstateRef
State	xsd:string	Требуемое	Состояние LED; возможные значения атрибута: включено, выключено, мигание
LEDcolor	xsd:string	Требуемое	Цвет состояния LED; правильные значения: зеленый, янтарный, красный
flashingPeriod	xsd:unsignedInt	Опциональное	Если состояние – мигание, то период мигания LED в миллисекундах

impulsWidth	xsd:unsignedByte	По умолчанию	Ширина импульса мигания в процентах периода мигания; если атрибут impulsWidth пропущен, значение по умолчанию равно 50 %
numberOfImpulses	xsd:unsignedByte	По умолчанию	Число импульсов в случае более чем одного импульса мигания за период мигания; если этот атрибут присутствует, атрибут impulsWidth должен быть представлен также, если атрибут numberOfImpulses пропущен, значение по умолчанию - 1

E.4.3.2.3 Элемент combinedState (комбинированное состояние)

Элемент combinedState разрешает индикацию состояний устройства, которые реализуются с помощью сигналов более чем одного LED.

Описание комбинированного состояния предоставлено в группе g_labels.

Ссылки на состояния LED, участвующие в сигнализации комбинированного состояния, должна осуществляться с помощью не менее чем двух субэлементов LEDstateRef (ссылка на состояние LED) элемента combinedState (комбинированное состояние).

Ссылка на элемент LEDstate кодируется как значение атрибута единичного атрибута элемента LEDstateRef (см. таблицу E.9).

Таблица E.9 — Атрибуты элемента LEDstateRef (ссылка на состояние LED)

Атрибут	Тип данных	Применение	Описание
stateIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного элемента LEDstate

E.4.4 DeviceFunction (функция устройства)

E.4.4.1 Общие положения

Элемент DeviceFunction, показанный на рисунке 23, определяет вид устройства в каталоге, представленный в виде набора capabilities(возможностей), перечисляющих характеристики устройства и указывающих его соответствие различным стандартам.

E.4.4.2 Элемент capabilities (возможности)

E.4.4.2.1 Общие положения

Обязательный элемент capabilities описывает все функциональные параметры, их характеристики, и другие основные параметры устройства, которые должны определяться с помощью инструментов, использующих профиль устройства для выбора изделий с такими же или аналогичными свойствами.

Элемент capabilities (возможности) описывает характеристики устройства в текстовом формате. Он содержит последовательность из одного или более элементов characteristicsList (список характеристик) и опциональный элемент standardComplianceList (стандартный список соответствия).

E.4.4.2.2 Элемент characteristicsList (список характеристик)

E.4.4.2.2.1 Общие положения

Элемент characteristicsList (список характеристик) представляет собой собрание характеристик. Этот элемент должен содержать не менее одного субэлемента characteristic (характеристика). Характеристики в списке могут быть ассоциированы с категорией, которая может быть выражена в виде текстового содержания субэлемента g_labels опционального субэлемента category (категория) элемента characteristicsList.

E.4.4.2.2.2 Элемент characteristic (характеристика)

Элемент characteristic описывает единичную характеристику устройства. Он содержит обязательный элемент characteristicName (имя характеристики) и один или несколько элементов characteristicContent (содержание характеристики).

E.4.4.2.2.3 Элемент characteristicName (имя характеристики)

Обязательный элемент characteristicName обозначает основную техническую характеристику устройства. Для имен характеристик рекомендуется использовать словарь в перечне данных изделия.

Пример —

«Максимальное рабочее напряжение», «Защита от перегрузки», «Электрическая прочность».

E.4.4.2.2.4 Элемент characteristicContent (содержание характеристики)

Этот обязательный элемент содержит значение элемента характеристики. Множественные значения могут быть выражены с помощью использования нескольких элементов characteristicContent.

Пример —

Пример единичного значения для «Максимальное рабочее напряжение» – 680 В.

E.4.4.2.3 Элемент standardComplianceList (стандартный список соответствия)

Элемент standardComplianceList представляет собой собрание элементов compliantWith (соответствие с). Этот элемент отдельно является опциональным; если он существует, то содержит не менее одного субэлемента compliantWith.

Субэлемент compliantWith имеет атрибуты, которые устанавливают соответствие устройства международным стандартам или внутренним стандартам компании. Содержание типа g_labels этого элемента может включать в себя замечания, касающиеся этих стандартов.

Имя или номер стандарта предоставляется с помощью требуемого атрибута name (имя) элемента compliantWith. Второй атрибут — значение по умолчанию range (диапазон) элемента compliantWith определяет диапазон применимости стандарта согласно таблице E.10.

Таблица E.10 — Атрибуты элемента compliantWith (соответствие с)

Атрибут	Тип данных	Применение	Описание
Name	xsd:строка	требуемое	Имя или номер стандарта
Range	xsd:NMToken	По умолчанию	Два возможных пронумерованных значения атрибута – международный (по умолчанию) или внутренний

E.4.4.3 Элемент picturesList (список изображений)

Элемент picturesList предлагает возможность связать изображения с профилем устройства. Он содержит один или более субэлементов с изображениями, доступ к которым обеспечивается через субэлемент g_labels.

В таблице Е.11 определены атрибуты субэлемента изображение: опциональное число изображений, обязательная связь с внешними источниками, содержащими графическую информацию.

Таблица Е.11 — Атрибуты элемента picture (изображение)

Атрибут	Тип данных	Применение	Описание
URI	xsd:anyURI	Требуемое	Связь с внешним источником
Number	xsd:unsignedInt	Опциональное	Номер изображения

Е.4.4.4 Элемент dictionaryList (список словарей)

Опциональный элемент dictionaryList создает возможность включить в профиль устройства связи с внешними файлами-источниками текстов. Он содержит один или более элементов словарей, каждый из которых включает в себя один или более субэлементов файлов. Несколько файлов необходимы при наличии различных форматов файлов в словаре.

Обязательный атрибут lang типа xsd:language определяет языки, используемые в файлах, которые связаны с элементом словарь (см. таблицу Е.12). Обязательный атрибут uniqueID (уникальный идентификатор) типа xsd:ID поддерживает уникальную идентификацию входа в словарь, на который имеется ссылка из атрибута dictID элемента labelRef согласно таблице Е.2.

Таблица Е.12 — Атрибуты элемента dictionary (словарь)

Атрибут	Тип данных	Применение	Описание
Lang	xsd:language	Требуемый	Язык, используемый для файлов, относящихся к входу в словарь
uniqueID	xsd:ID	Требуемый	Уникальный ID входа в словарь

Субэлемент file (файл) содержит один обязательный атрибут, указанный в таблице Е.13.

Таблица Е.13 — Атрибуты элемента file (файл)

Атрибут	Тип данных	Применение	Описание
URI	xsd:anyURI	Требуемый	Связь с соответствующим файлом

E.4.5 ApplicationProcess (прикладной процесс)

E.4.5.1 Общие положения

Элемент ApplicationProcess представляет набор сервисов и параметров, формирующих поведение и интерфейсы устройства в терминах приложения, независимо от технологии устройства и лежащих в его основе коммуникационных сетей и протоколов коммуникаций.

Субэлементы элемента ApplicationProcess на рисунке 24 предоставляют общего подхода к описанию функций устройства, имеющих произвольную, одноуровневую или иерархическую структуру.

Функции моделируются как типы функций, которые реализуются внутри устройства или, если необходима иерархическая структура, внутри типов функций. Параметры интерфейса этих реализаций функций, которые могут соответствовать простому или сложному типу данных, ассоциированы с параметрами устройства путем построения ссылки из параметра на соответствующий параметр интерфейса реализации функции как в одноуровневых, так и в иерархических структурах.

Элемент ApplicationProcess содержит до пяти списков объектов (см. рисунок 24):

— два списка, определяющих типы данных (опционально) и типы функций (обязательно);

— один обязательный список, определяющий реализации функций на уровне устройства (возможно включая соединения между реализациями);

— один обязательный список, определяющий параметры устройства, и;

— один опциональный список, определяющий группы параметров (комбинации параметров для специальных целей).

E.4.5.2 Элемент dataTypeList (список типов данных)

E.4.5.2.1 Общие положения

Опциональный элемент dataTypeList присутствует, если внутри деклараций параметров в профиле устройства необходимы сложные типы данных типа таблиц или структур.

При наличии, этот элемент dataTypeList, представленный на рисунке Е.5, содержит последовательность одного или нескольких элементов из следующего набора:

— элемента array (таблица);

— элемента struct (структура);

— элемента enum (нумерация) или

— производного элемента.

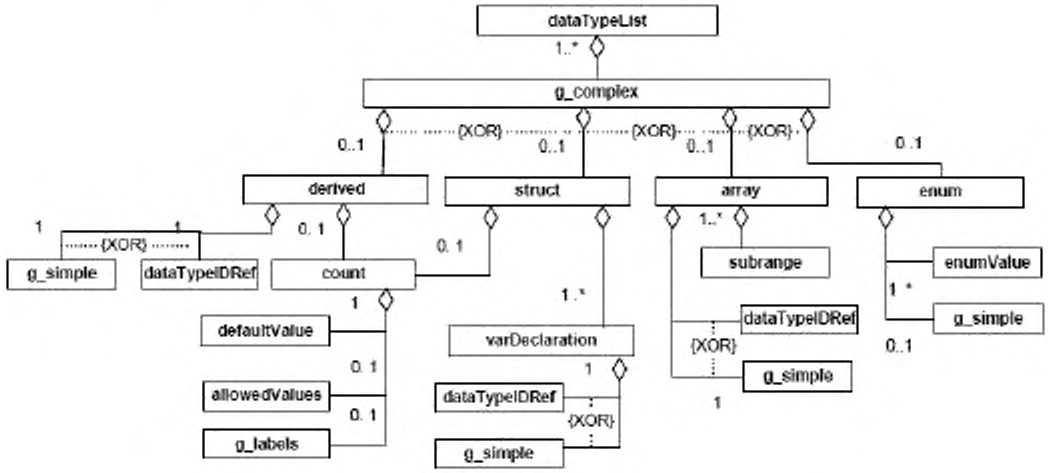


Рисунок Е.5 — dataTypeList (список типов данных)

Е.4.5.2.2 Общие элементы

Е.4.5.2.2.1 Группа g_simple

Группа g_simple содержит набор элементов, имена которых представляют имена всех простых типов данных, допускаемых при определении параметров внутри профиля устройства. Простые типы данных соответствуют элементарным типам данных, определенным в МЭК 61131-3, к которым добавляют типы данных BITSTRING и CHAR (=STRING[1]).

Эти элементы вводят в группы, допуская их размещение непосредственно как субэлементов элемента таблицы (или элемента varDeclaration, см. Е.4.5.4.3.2).

Е.4.5.2.2.2 Элемент count (счет)

Элемент count определяет число используемых блоков базового типа производного типа. Многоязычные имена и/или описания для элемента count

предоставляют с помощью группы g_labels. Описание группы g_labels приведено в Е.2.2.2.

Элемент count описывают следующими способами:

— его атрибутами;

— обязательным субэлементом defaultValue (значение по умолчанию) и, возможно, пустым набором субэлементов g_labels и allowedValues (допустимые значения).

Число блоков определяют значением атрибута defaultValue элемента count. Атрибут allowedValue определяет диапазон значений для значения по умолчанию.

Субэлементы defaultValue и allowedValues описаны в Е.4.5.6.2.5 и Е.4.5.6.2.7. Элемент count должен иметь атрибуты, указанные в таблице Е.14.

Таблица Е.14 — Атрибуты элемента count (счет)

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный идентификатор count
Access	xsd:NMToken	По умолчанию	Определяет, какие операции правильны для count: — чтение – доступ только для чтения (значение по умолчанию); — запись – доступ только для записи; — чтение/запись – доступ для чтения и записи; — нет Доступа – доступ не разрешен

Е.4.5.2.3 Элемент array (таблица)

Е.4.5.2.3.1 Общие положения

Элемент array (таблица) предназначен для описания типа данных array (массив), ссылка на который может выполняться из параметров интерфейса типа функций, из других определений типа таблицы, или из параметра компоненты внутри определения структурированного типа данных.

Элемент array содержит не менее одного элемента subrange (поддиапазон) и либо один элемент, описывающий простой тип данных из группы g_simple, либо элемент dataTypeIDRef (ссылка идентификатора типа

данных), имеющий ссылку на один из определенных сложных типов данных внутри элемента dataTypeList (список типов данных).

В случае многомерных таблиц будут присутствовать несколько элементов subrange. В этом случае первый элемент поддиапазона в последовательности определяет поддиапазон для первого слева индекса таблицы, а последний элемент поддиапазона в последовательности определяет поддиапазон для крайнего правого индекса таблицы.

Элемент array (массив) содержит атрибуты, указанные в таблице Е.15.

Таблица Е.15 — Атрибуты элемента array (массив)

Атрибут	Тип данных	Применение	Описание
Name	xsd:строка	Требуемое	Имя типа данных (тип таблица)
uniqueID	xsd:ID	Требуемое	Уникальный ID (тип таблица)
Description	xsd:строка	Опциональное	Опциональное текстовое описание (тип таблица)

Е.4.5.2.3.2 Элемент subrange (поддиапазон)

Элемент subrange определяет нижний и верхний предельные значения индекса таблицы для одного измерения таблицы. Этот элемент не имеет субэлементов.

Предельные значения типа xsd:long содержатся в двух атрибутах элемента поддиапазона, указанных в таблице Е.16.

Таблица Е.16 — Атрибуты элемента subRange (поддиапазон)

Атрибут	Тип данных	Применение	Описание
lowerLimit	xsd:long	Требуемое	Нижний предел поддиапазона
upperLimit	xsd:long	Требуемое	Верхний предел поддиапазона

Е.4.5.2.4 Элемент struct (структура)

Е.4.5.2.4.1 Общие положения

Элемент struct предназначен для описания типа структурированных данных, на который может выполняться ссылка из параметра интерфейса типа функции, из определения типа таблицы или из параметра компонента внутри определения другого типа структурированных данных.

Элемент struct содержит последовательность из одного или более элементов varDeclaration, которые определяют компоненты типа структурированных данных.

Элемент struct должен содержать атрибуты, указанные в таблице Е.17.

Таблица Е.17 — Атрибуты элемента struct (структура)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя типа данных типа структурированных данных
uniqueID	xsd:ID	Требуемое	Уникальный ID типа структурированных данных
Description	xsd:string	Опциональное	Опциональное текстовое описание типа структурированных данных

E.4.5.2.4.2 Элемент varDeclaration

В контексте определения типа структурированных данных элемент varDeclaration описывает единичный параметр компонента (составляющею) структуры.

В контексте определения интерфейса функции элемент varDeclaration описывает единичный параметр интерфейса типа функции.

Тип данных параметра компонента или параметра интерфейса определяется либо с помощью элемента, описывающего простой тип данных группы g_simple, либо с помощью элемента dataTypeIDRef (ссылка идентификатора типа данных), который выполняет ссылку на один из определенных сложных типов данных в элементе dataTypeList (список типов данных).

Последующие характеристики параметров содержатся в атрибутах элемента varDeclaration, указанных в таблице Е. 18.

Таблица Е.18 — Атрибуты элемента varDeclaration

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя параметра интерфейса или компонента структуры
uniqueID	xsd:ID	Требуемое	Уникальный ID параметра интерфейса или компонента структуры (см. примечание 1)
Size	xsd:string	Опциональное	Число элементов, если параметр интерфейса или компонент структуры анонимного типа ARRAY, BITSTRING, STRING или WSTRING (см. примечание 2)
initialValue	xsd:string	Опциональное	Начальное значение параметра интерфейса или компонента структуры (см. примечание 3)
Description	xsd:string	Опциональное	Опциональное текстовое описание параметра интерфейса или компоненты структуры

Примечание 1 — При создании уникальных ID для параметров важно соблюдать уникальность всех ID внутри файла – источника XML. Для создания возможности одинаковых имен для параметров компонентов различных структур данных и одинаковых имен для параметров интерфейса типов функций ID параметра должен обычно соединять типовое имя типа структурированных данных или типовое имя типа функций с именем параметра для гарантии уникальности.

Примечание 2 — Анонимные типы определяют размер таблицы, битовой строки или строки непосредственно в декларации параметра, а не путем ссылки на тип сложных данных с именем. Для таблицы тип данных единичного элемента таблицы дается типом данных параметра. Для битовой строки, единичный элемент является одним битом.

Примечание 3 — При наличии данного атрибута он определяет начальное (по умолчанию) значение параметра интерфейса типа функции и представляется заданным значением по умолчанию параметра, ассоциированного с параметром интерфейса экземпляра функции.

E.4.5.2.5 Элемент enum (нумерация)

E.4.5.2.5.1 Общие положения

Элемент enum предназначен для описания типа пронумерованных данных, на которые могут быть ссылки из параметра интерфейса типа функции, из определения типа таблицы или из параметра компонента внутри определения типа структурированных данных.

В соответствии с рисунком E.5 этот элемент содержит последовательность из одного или более элементов enumValue (значение нумерации), которые определяют константы нумерации типа пронумерованных данных. Тип данных констант нумерации опционально определяется с помощью элемента, описывающего тип простых данных из группы g_simple.

Элемент enum содержит атрибуты, указанные в таблице E.19.

Таблица E.19 — Атрибуты элемента enum (нумерация)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Типовое имя типа пронумерованных данных
uniqueID	xsd:ID	Требуемое	Уникальный ID типа пронумерованных данных
Size	xsd:string	Опциональное	Опциональное число пронумерованных значений типа пронумерованных данных
Description	xsd:string	Опциональное	Опциональное текстовое описание типа пронумерованных данных

E.4.5.2.5.2 Элемент enumValue (значение нумерации)

Элемент enumValue определяет имя (имена) и опционально числовое значение единичной константы нумерации. Имя (имена) устанавливается в группе g_labels, а значение содержится в атрибуте единичного значения элемента enumValue как указано в таблице E.20.

Таблица Е.20 — Атрибуты элемента enumValue(значение нумерации)

Атрибут	Тип данных	Применение	Описание
Value	xsd:string	Опциональное	Опциональный атрибут: фиксированное числовое значение для константы нумерации, представленное в виде строки символов

Е.4.5.2.6 Элемент derived (производный)

Элемент derived предназначен для вывода нового типа данных из данного базового типа.

Элемент derived содержит опциональный элемент count (счет) и либо элемент, описывающий тип простых данных из группы g_simple, либо элемент dataTypeIDRef, который ссылается на один из определенных сложных типов данных внутри элемента dataTypeList (список типов данных).

Если элемент count (счет) пропущен, определение производного типа вводит имя нового типа для соответствующего базового типа. Если элемент count присутствует, он определяет число блоков соответствующего базового типа, используемых для построения производного типа (например базовый тип BITSTRING, count = 4 определяет производный тип размером 4 бита).

Элемент derived содержит атрибуты, указанные в таблице Е.21.

Таблица Е.21 — Атрибуты элемента derived

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя типа данных производного типа
uniqueID	xsd:ID	Требуемое	Уникальный ID производного типа
Description	xsd:string	Опциональное	Опциональное текстовое описание производного типа

E.4.5.3 Элемент functionTypeList (список типов функций)

Если опциональный элемент ApplicationProcess (прикладной процесс) присутствует в профиле устройства, он содержит обязательный элемент functionTypeList, указанный на рисунке E.6.

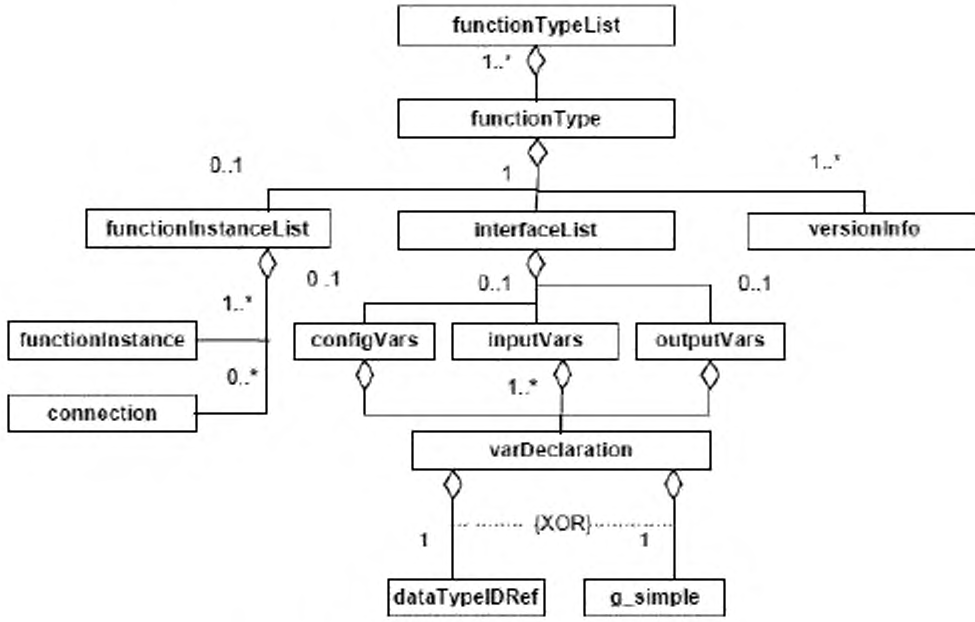


Рисунок E.6 — functionTypeList (список типов функций)

Элемент functionTypeList представляет последовательность из одного или многих элементов functionType (тип функции).

Каждый из элементов functionType представляет описание типа функции устройства, на которую имеется ссылка из не менее чем одного экземпляра типа функции в элементе functionInstanceList (список экземпляров функции). Ссылки из более чем одного экземпляра того же типа функций также возможны.

Описание типа функции содержит объекты и данные, которые являются общими для всех экземпляров данного типа функций.

Пример 1 — Примерами являются параметры (или параметры функций) – объекты, формирующие интерфейс функции (тип экземпляра).

Пример 2 — Другими примерами являются сущности, содержащиеся внутри тела функции в иерархически структурированном функциональном описании. Эти сущности, которые расположены внутри элемента functionInstanceList (список экземпляров функции) внутри типа функции, ссылаются на другие типы функции в списке типов функций.

E.4.5.4 Элемент functionType (тип функции)

E.4.5.4.1 Общие положения

Элемент functionType содержит один или несколько элементов versionInfo (информация о версии), обязательный элемент interfaceList (список интерфейсов) и опциональный элемент functionInstanceList (список экземпляров функции). Элемент functionInstanceList присутствует в элементе functionType только в том случае, если функция иерархически структурирована.

Дополнительно элемент functionType должен содержать атрибуты, указанные в таблице E.22.

Таблица E.22 — Атрибуты элемента functionType (тип функции)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Типовое имя типа функции
uniqueID	xsd:ID	Требуемое	Уникальный ID типа функции
Description	xsd:string	Опциональное	Опциональное текстовое описание типа функции
Package	xsd:string	Опциональное	Опциональная текстовая ассоциация типа функции с «пакетом» или аналогичной схемой классификации; использование этого атрибута осуществляет валидатор профиля

E.4.5.4.2 Элемент versionInfo (информация о версии)

Обязательный элемент versionInfo, входящий в элемент functionType (тип функции), предоставляет информацию об истории версий типа функций, относящуюся к определению интерфейса.

Для прослеживания истории версий элемент versionInfo может быть введен несколько раз. Множественные входы должны быть организованы в элементе functionType в следующей последовательности:

a) первый вход предоставляет наиболее раннюю версию;

b) второй вход предоставляет непосредственно предшествующую версию;

c) последний вход представляет первую выпущенную версию.

Данный элемент будет представлен один раз при создании описания типа функции. Новые элементы будут добавляться только при введении модификации типа функции, которая приводит к модифицированной версии профиля устройства.

Элемент versionInfo должен содержать атрибуты, указанные в таблице E.23.

Таблица E.23 — Атрибуты элемента versionInfo (информация о версии)

Атрибут	Тип данных	Применение	Описание
Organization	xsd:string	Требуемое	Имя организации, поддерживающей тип функции
Version	xsd:string	Требуемое	Обозначение версии в истории версий; предлагаемый формат: "xx.yy" (xx,yy = 0..255)
Author	xsd:string	Требуемое	Имя лица, поддерживающего тип функции
Date	xsd:date	Требуемое	Дата данной версии
Remarks	xsd:string	Опциональное	Описательная информация, относящаяся к специальному этапу истории версий

E.4.5.4.3 Элемент interfaceList (список интерфейсов)

E.4.5.4.3.1 Общие положения

Обязательный элемент interfaceList, входящий в элемент functionType(тип функции), предоставляет определение интерфейса типа функции. Элементы интерфейса следующие:

— входные параметры и/или

— выходные параметры, и/или

— параметры конфигурации типа функции.

Следовательно элемент interfaceList содержит последовательность из трех элементов, где каждый элемент представляет списки от одной до нескольких деклараций параметров, закодированных как элементы varDeclaration:

— один опциональный элемент inputVars (входные параметры),

— один опциональный элемент outputVars (выходные параметры), и

— один опциональный элемент configVars (конфигурационные параметры).

Ни один из элементов interfaceList, inputVars, outputVars или configVars не имеет каких-либо атрибутов.

E.4.5.4.3.2 Элемент varDeclaration

В контексте определения структурированного типа данных элемент varDeclaration описывает единичный параметр компонента (элемента) структуры.

В контексте определения интерфейса типа функции элемент varDeclaration описывает единичный параметр интерфейса типа функции.

Тип данных параметра компоненты или параметра интерфейса определяется либо элементом, описывающим простой тип данных из группы g_simple, либо элементом dataTypeIDRef (ссылочный идентификатор типа данных), который ссылается на один из определенных сложных типов данных элемента dataTypeList (список типов данных).

В E.4.5.2.2.1 Приведено описание группы g_simple, а в E.4.5.4.3.3 – элементов dataTypeIDRef.

Все дополнительные характеристики параметра содержатся в атрибутах элемента varDeclaration, указанных в таблице E.24.

Таблица Е.24 — Атрибуты элемента varDeclaration

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя параметра интерфейса или компонента структуры
uniqueID	xsd:ID	Требуемое	Уникальный ID параметра интерфейса или компонента структуры
Size	xsd:string	Опциональное	Число элементов, если параметр интерфейса или компонент структуры имеют анонимный тип ARRAY, BITSTRING, STRING или WSTRING
initialValue	xsd:string	Опциональное	Начальное значение параметра интерфейса или компонента структуры
Description	xsd:string	Опциональное	Опциональное текстовое описание параметра интерфейса или компонента структуры

Е.4.5.4.3.3 Элемент dataTypeIDRef (ссылка идентификатора типа данных)

Элемент dataTypeIDRef предназначен для ссылки на сложный тип данных элемента dataTypeList (список типов данных) (см. Е.4.5.2) либо из параметра интерфейса типа функции, либо из определения типа таблицы, либо из параметра компоненты внутри определения структурированного типа данных.

Ссылка типа xsd:IDREF предоставляется в виде атрибута элемента dataTypeIDRef как указано в таблице Е.25.

Таблица Е.25 — Атрибуты элемента dataTypeIDRef (ссылка идентификатора типа данных)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного типа данных

E.4.5.5 Элемент functionInstanceList (список экземпляров функции)

E.4.5.5.1 Общие положения

Если опциональный элемент ApplicationProcess (прикладной процесс) присутствует в профиле устройства, он содержит обязательный элемент functionInstanceList, который включает в себя последовательность из одного или нескольких элементов functionInstance (экземпляр функции), а также от нуля до нескольких элементов connection (соединение).

На уровне процесса приложения элементы functionInstance предоставляют допустимые функции приложения типа устройства, не зависящие от типа или протокола сети. Элементы connection представляют соединения (если они существуют) между конкретными выходными и входными параметрами этих экземпляров функции.

Элемент functionInstanceList также определяется как опциональный субэлемент элемента functionType (см. E.4.5.4). Аналогично этому на уровне процесса приложения элемент functionInstanceList содержит последовательность из одного или нескольких элементов functionInstance и от нуля до нескольких элементов connection.

Элемент functionInstanceList присутствует в элементе functionType только в том случае, если функция иерархически структурирована. В этом случае элементы functionInstance предоставляют внутренние функции, содержащиеся в типе функций, а элементы connection – опциональные внутренние соединения. Эти функции и их опциональные соединения могут быть реализованы совместно с реализацией содержащего их типа функций.

Элемент functionInstanceList не имеет каких-либо атрибутов.

E.4.5.5.2 Элемент functionInstance (экземпляр функции)

Обязательный элемент functionInstance не содержит субэлементов.

Элемент functionInstance должен содержать атрибуты, указанные в таблице E.26.

Таблица Е.26 — Атрибуты элемента functionInstance (экземпляр функции)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя экземпляра функции
uniqueID	xsd:ID	Требуемое	Уникальный ID экземпляра функции (см. примечание)
typeIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного типа функции
Description	xsd:string	Опциональное	Опциональное текстовое описание экземпляра функции
Примечание — При создании уникального ID для экземпляра функции важно, чтобы ID был уникальным во всех создаваемых ID внутри файла источника XML. Для создания возможности одинаковых имен для экземпляров функции при различных типах функций ID экземпляра функции должен обычно образовываться путем соединения имени типа функции с именем экземпляра функции для обеспечения уникальности.			

Е.4.5.5.3 Элемент connection (соединение)

Опциональный элемент connection определяет соединение между выходным параметром экземпляра функции и входным параметром другого экземпляра функции. Внутри типов функции соединение также может быть проведено между входным параметром типа функции и входным параметром содержащегося в типе экземпляра функции или между выходным параметром содержащегося в типе экземпляра функции и выходным параметром типа функции. Элемент connection может появляться от ни одного до нескольких раз.

Элемент connection содержит атрибуты, указанные в таблице Е.27.

Таблица Е.27 — Атрибуты элемента connection (соединение)

Атрибут	Тип данных	Применение	Описание
Source	xsd:string	Требуемое	Начальная точка соединения
Destination	xsd:string	Требуемое	Конечная точка соединения
Description	xsd:string	Опциональное	Опциональное текстовое описание соединения

Пример — Значения атрибутов source (источник) и destination (назначение) могут быть использованы для кодирования начальной и конечной точек соединения с помощью синтаксиса <function_instance_name>'.'<variable_name>; пример для значения атрибута источник: 'PowerMeasures.Frequency' ('Мощность измерения. Частота'). Соединения с параметрами интерфейса типа функции используют только имена параметров интерфейса.

E.4.5.6 Элемент parameterList (список параметров)

E.4.5.6.1 Общие положения

Если в профиле устройства присутствует опциональный элемент ApplicationProcess (прикладной процесс), он содержит обязательный элемент parameterList (список параметров), указанный на рисунке E.7, который представляет последовательность из одного или нескольких элементов parameter (параметр).

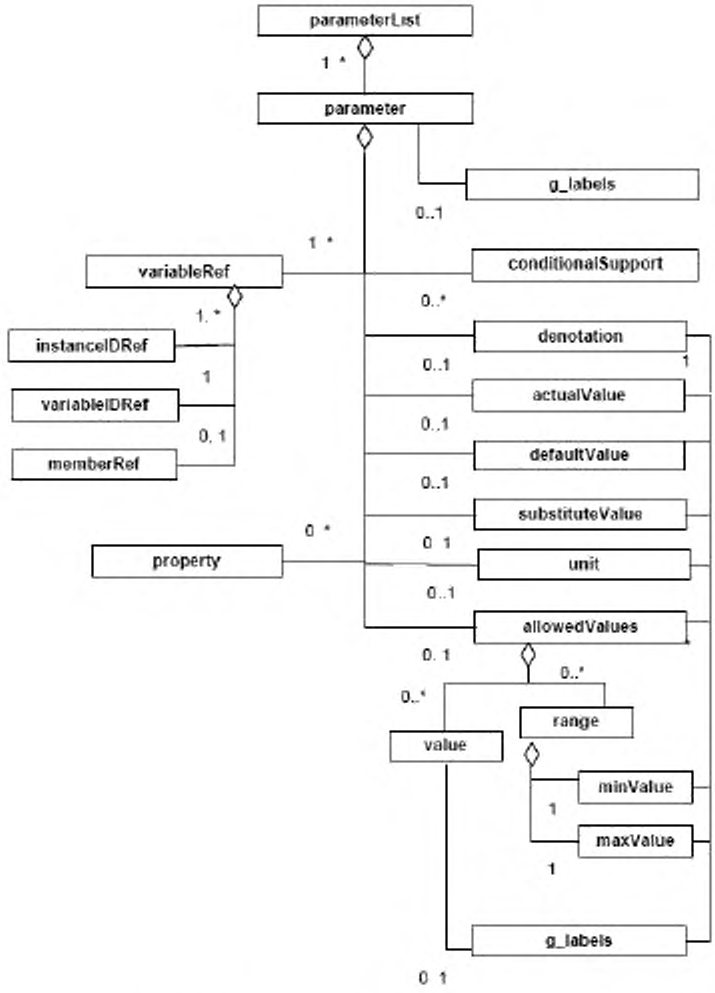


Рисунок E.7 — parameterList (список параметров)

Каждый из элементов parameter представляет параметр профиля устройства. Многоязычные имена и/или описания параметров предоставляются с помощью группы g_labels. Группа g_labels описана в E.2.2.2.

Параметр описывается следующими способами:

— его атрибутами;

— путем ссылки на один (или более) параметр(ов) интерфейса одного (или более) экземпляра(ов) функции (обязательный элемент variableRef (ссылка на переменную));

— пустым набором субэлементов (conditionalSupport (условная

поддержка), denotation (обозначение), actualValue (фактическое значение), defaultValue (значение по умолчанию), substituteValue (подстановочное значение), allowedValues (допустимые значения), блок, свойство и g_labels).

Примечание — Ссылки на многие параметры представляют собой специальный случай: специальные параметры могут одновременно иметь ссылку на выходной параметр одного экземпляра функции и на входной параметр другого экземпляра функции. В этом случае типы данных двух параметров должны быть одинаковыми. Программа синтаксического анализа XML не может проверять эквивалентность типов данных. Это может быть проверено только с помощью вспомогательного инструмента.

E.4.5.6.2 Элемент parameter (параметр)

E.4.5.6.2.1 Общие положения

Элемент parameter должен содержать атрибуты, указанные в таблице E.28.

Таблица E.28 — Атрибуты элемента parameter (параметр)

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный ID параметра
Access	xsd:NMToken	По умолчанию	Определяет, какая операция является правильной для параметра: — чтение – доступ только для чтения (значение по умолчанию); — запись - доступ только для записи; — чтение/Запись – доступ для чтения и записи; — нетДоступа – доступ не разрешается

Продолжение таблицы Е.28

Атрибут	Тип данных	Применение	Описание
Support	xsd:NMToken	Опционально	Определяет, должен или нет параметр использоваться в устройстве; правильные значения: — обязательно – использование параметра требуется; — опционально – использование параметра возможно, но не обязательно; — условно – использование параметра требуется, если один или более из других необязательных параметров используются; эти параметры определяются с помощью субэлемента conditionalSupport(условная поддержка)
Persistence	xsd:boolean	По умолчанию	Определяет поведение после отказа питания; правильное значение - ложь (по умолчанию) или истина
Offset	xsd:string	Опционально	Сдвиг, который добавляется к фактическому значению для образования масштабированной величины: Техническое значение = (Значение параметра + сдвиг) * множитель; при его отсутствии, принимается сдвиг = 0
Multiplier	xsd:string	Опционально	Коэффициент масштабирования, на который умножается фактическое значение для образования масштабированной величины: Техническое значение = (ПараметрЗначение + сдвиг) * множитель; при его отсутствии, принимается множитель = 1

Е.4.5.6.2.2 Элемент conditionalSupport (условная поддержка)

Один или более элементов conditionalSupport присутствуют только в том случае, если значением атрибута support (поддержка) элемента parameter (параметр) является conditional (условный). Каждый элемент имеет ссылку на

единичный опциональный параметр. Если хотя бы один из этих опциональных параметров используется, условный параметр также должен использоваться.

Элемент conditionalSupport должен содержать единичный атрибут, указанный в таблице Е.29.

Таблица Е.29 — Атрибуты элемента conditionalSupport(условная поддержка)

Атрибут	Тип данных	Применение	Описание
paramIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного опционального параметра

Е.4.5.6.2.3 Элемент denotation (обозначение)

Элемент denotation предназначен для поддержки зависящих от приложения многоязычных имен параметра. Имена предоставляются с помощью обязательного субэлемента g_labels. Допускается добавлять многоязычную описательную информацию. Элемент denotation не имеет атрибутов.

Е.4.5.6.2.4 Элемент actualValue (фактическое значение)

Элемент actualValue предназначен для поддержки фактического значения параметра. Опциональный субэлемент g_labels может предоставить многоязычную описательную информацию для этой величины, которая определяется в атрибуте value (значение) элемента actualValue. Могут быть также указаны сдвиг и множитель.

Атрибуты элемента actualValue должны соответствовать указанным в таблице Е.30.

Таблица Е.30 — Атрибуты элемента actualValue (фактическое значение)

Атрибут	Тип данных	Применение	Описание
Value	xsd:строка	Требуемое	Фактическое значение
Offset	xsd: строка	Опциональное	Сдвиг, который добавляется к фактическому значению для образования масштабированной величины: ТехническоеЗначение = (значение + сдвиг) * множитель; при его отсутствии, должно быть использовано соответствующее значение элемента параметра
Multiplier	xsd: строка	Опциональное	Коэффициент масштабирования, на который умножается фактическое значение для образования масштабированной величины: ТехническоеЗначение = (значение + сдвиг) * множитель; при его отсутствии, должно быть использовано соответствующее значение элемента параметра

Е.4.5.6.2.5 Элемент defaultValue (значение по умолчанию)

Элемент defaultValue предназначен для поддержки значения по умолчанию параметра. Это значение копирует начальное значение параметра интерфейса типа функции, ассоциированной с параметром.

Опциональный субэлемент g_labels может предоставить многоязычные имена и/или обязательную (дескриптивную) информацию для этой величины. Сама величина предоставляется атрибутом value (значение) элемента defaultValue. Могут быть также установлены сдвиг и множитель.

Атрибуты элемента defaultValue должны соответствовать указанным в таблице Е.30.

E.4.5.6.2.6 Элемент substituteValue (подстановочное значение)

Элемент substituteValue определяет специальное значение параметра, которое предоставляется для приложения устройства в некоторых рабочих состояниях устройства (например, в случае отказа устройства).

Опциональный субэлемент g_labels может предоставить многоязычные имена и/или дескриптивную информацию для этой величины. Сама эта величина предоставляется атрибутом value (значение) элемента substituteValue. Могут быть также установлены сдвиг и множитель.

Атрибуты элемента substituteValue должны соответствовать указанным в таблице E.30.

E.4.5.6.2.7 Элемент allowedValues (допустимые значения)

Элемент allowedValues определяет список поддерживаемых значений и/или один или несколько диапазонов поддерживаемых значений для параметра.

Список поддерживаемых значений представлен в виде некоторого количества от нуля до нескольких субэлементов value элемента allowedValues, а диапазоны также представлены в виде некоторого количества от нуля до нескольких субэлементов range (диапазон) элемента allowedValues.

Значение субэлемента устанавливает одно допустимое значение параметра. Опциональный субэлемент g_labels может предоставить многоязычные имена и/или дескриптивную информацию для этой величины. Сама эта величина предоставляется атрибутом value (значение) элемента value. Могут быть также установлены сдвиг и множитель.

Атрибуты элемента value должны соответствовать указанным в таблице E.30.

Субэлемент range содержит два требуемых субэлемента, а именно элемент minValue (минимальное значение) и элемент maxValue (максимальное значение), которые определяют пределы диапазона допустимых значений. Элементы minValue и maxValue имеют такую же структуру и атрибуты, как субэлемент value элемента allowedValues.

Следовательно, описание субэлемента value и таблица Е.30 действительны также для этих субэлементов.

Е.4.5.6.2.8 Элемент unit (единица)

Элемент unit определяет технические единицы параметра (например время, температуру, давление, поток, ускорение, ток, энергию), согласно ИСО 80000-1. Опциональный элемент g_labels может предоставить многоязычные имена и/или описательную информацию для технических единиц.

Атрибуты элемента unit должны соответствовать указанным в таблице Е.31.

Таблица Е.31 — Атрибуты элемента unit(единица)

Атрибут	Тип данных	Применение	Описание
Multiplier	xsd:string	Требуемое	Множитель для технических единиц аналоговых параметров
unitURI	xsd:anyURI	Опциональное	Связь с соответствующим определением единицы в файле, содержащем технические единицы (например, время, температуру, давление, поток, ускорение, ток, энергию) согласно ИСО 80000-1

Е.4.5.6.2.9 Элемент variableRef (ссылка на переменную)

Элемент variableRef создает ссылку на параметр интерфейса экземпляра функции или, если параметр представляет собой таблицу или структуру, возможна ссылка на элемент параметра (элемент таблицы или компонент структуры).

В иерархически структурированном элементе ApplicationProcess (прикладной процесс) экземпляры функции могут быть расположены внутри экземпляров функции других типов функций. Следовательно, конкретный экземпляр на дереве функций может быть доступен только путем последовательных шагов по дереву, т. е. конкретный экземпляр должен получить адрес путем конкатенации имен экземпляров. Для отображения этой

конкатенации и создания возможности ссылки на элемент структуры variableRef (ссылка на переменную) содержит следующее:

— последовательность из одного или нескольких элементов instanceIDRef (ссылка на идентификатор экземпляра) за которой следует;

— один обязательный элемент variableIDRef (ссылка на идентификатор переменной) и

— опциональный элемент memberRef (ссылка на элемент).

Элемент variableRef содержит атрибут, указанный в таблице Е.33.

Таблица Е.32 — Атрибут элемента variableRef (ссылка на переменную)

Атрибут	Тип данных	Применение	Описание
Position	xsd:unsignedByte	По умолчанию	Определяет последовательность нескольких отображенных объектов данных в одном объекте параметра; позиция = 1 означает начало отображения в самом низком положении бита; число битов определяется типом данных объекта данных; последовательные объекты данных упаковываются без зазоров; значение по умолчанию: 1 (см. примечание)
Примечание — В случае одного отображенного объекта данных атрибут может быть пропущен.			

Е.4.5.6.2.10 Элемент instanceIDRef (ссылка на идентификатор экземпляра)

Элемент instanceIDRef предназначен для ссылки на экземпляр функции внутри элемента functionInstanceList (список экземпляров функции), который может находиться либо на уровне элемента ApplicationProcess (прикладной процесс), либо на уровне элемента functionType (тип функции).

Ссылка типа xsd:IDREF предоставлена в виде атрибута элемента instanceIDRef, указанного в таблице Е.33.

Таблица Е.33 — Атрибуты элемента instanceIDRef (ссылка на идентификатор экземпляра)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного экземпляра функции

Е.4.5.6.2.11 Элемент variableIDRef (ссылка на идентификатор переменной)

Элемент variableIDRef предназначен для ссылки на параметр интерфейса типа функции внутри элемента functionTypeList (список типов функций).

В данном элементе variableRef (ссылка на переменную) экземпляр этого типа функции определяется элементом functionInstance (экземпляр функции), на который производится ссылка элементом instanceIDRef (ссылка на идентификатор экземпляра), который непосредственно предшествует элементу variableIDRef.

Ссылка типа xsd:IDREF предоставляется в виде атрибута элемента variableIDRef, указанного в таблице Е.34.

Таблица Е.34 — Атрибуты элемента variableIDRef (ссылка на идентификатор переменной)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного параметра интерфейса типа функции

Е.4.5.6.2.12 Элемент memberRef (ссылка на элемент)

Опциональный элемент memberRef ссылается либо на соответствующий компонент параметра интерфейса типа структурированных данных (используется атрибут uniqueIDRef (ссылка на уникальный идентификатор)), либо на соответствующий элемент таблицы параметра интерфейса типа табличных данных (используется атрибут index (индекс)). Один из этих двух атрибутов должен присутствовать, если имеется элемент memberRef.

Элемент memberRef должен содержать атрибуты, указанные в таблице E.35.

Таблица E.35 — Атрибуты элемента memberRef (ссылка на элемент)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Опционально	Уникальный ID ссылочного компонента типа структурированных данных
Index	xsd:long	Опционально	Индекс ссылочного элемента таблицы

E.4.5.6.3 Элемент property (свойство)

Элемент property вводится в качестве общего элемента для создания возможности включения значений для дополнительных специальных свойств в описание параметра.

Элемент property должен содержать атрибуты, указанные в таблице E.36.

Таблица E.36 — Атрибуты элемента property (свойство)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя свойства
Value	xsd:string	Требуемое	Значение свойства

E.4.5.7 Элемент parameterGroupList (список группы параметров)

E.4.5.7.1 Общие положения

Опциональный элемент parameterGroupList, при его наличии, содержит последовательность от одного до нескольких элементов parameterGroup (группа параметров) соглано рисунку E.8. Многоязычные имена и/или описания групп параметров предоставляются с помощью группы g_labels. Описание группы g_labels приведено в E.2.2.2.

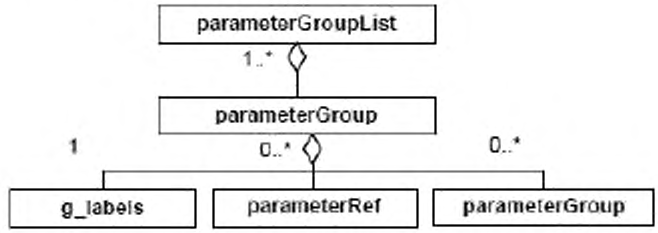


Рисунок E.8 — parameterGroupList (список группы параметров)

E.4.5.7.2 Элемент parameterGroup (группа параметров)

Каждый из элементов parameterGroup объединяет набор параметров из элемента parameterList (список параметров) для построения группы параметров, предназначенных для специальной цели, например для подготовки видов HMI. Эта цель указывается значением атрибута kindOfAccess (вид доступа) элемента parameterGroup. Возможно определение иерархии групп параметров.

На соответствующие параметры в наборе производится ссылка с помощью соответствующего номера элементов parameterRef (ссылка на переменную).

Элемент parameterGroup содержит атрибуты, указанные в таблице E.37.

Таблица E.37 — Атрибуты элемента parameterGroup (группа параметров)

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный ID группы параметров
kindOfAccess	xsd:string	Опционально	Классифицирует параметры группы параметров

E.4.5.7.3 Элемент parameterRef (ссылка на переменную)

Элемент parameterRef предназначен для ссылки на элемент parameter (параметр) внутри элемента parameterList (список параметров), относящегося к элементу ApplicationProcess (прикладной процесс).

Ссылка типа xsd:IDREF предоставляется в качестве атрибута элемента parameterRef (ссылка на переменную) указанного в таблице E.38.

Таблица E.38 — Атрибуты элемента parameterRef (ссылка на переменную)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного параметра

E.4.6 Схемы DDXML шаблона профиля устройства

E.4.6.1 XML Схема: ISO15745ProfileContainer.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="
```

gost.e-ecolog.ru

```
>
```

```
<xsd:element name="ISO15745ProfileContainer">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ISO15745Profile" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileHeader" type="ProfileHeader_DataType"/>
<xsd:element name="ProfileBody" type="ProfileBody_DataType"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* HEADER SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileHeader_DataType">
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0" maxOccurs="1"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"
maxOccurs="1"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:annotation>
<xsd:documentation>* BODY SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileBody_DataType" abstract="true">
</xsd:complexType>
<xsd:annotation>
<xsd:documentation>* HEADER DATA TYPES *</xsd:documentation>
</xsd:annotation>
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP"/>
<xsd:enumeration value="Process"/>
<xsd:enumeration value="InformationExchange"/>
<xsd:enumeration value="Resource"/>
<xsd:enumeration value="Device"/>
<xsd:enumeration value="CommunicationNetwork"/>
<xsd:enumeration value="Equipment"/>
<xsd:enumeration value="Human"/>
<xsd:enumeration value="Material"/>
```

```
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="CSI"/>
<xsd:enumeration value="HCI"/>
<xsd:enumeration value="ISI"/>
<xsd:enumeration value="API"/>
<xsd:enumeration value="CMI"/>
<xsd:enumeration value="ESI"/>
<xsd:enumeration value="FSI"/>
<xsd:enumeration value="MTI"/>
<xsd:enumeration value="SEI"/>
<xsd:enumeration value="USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:annotation>
<xsd:documentation>* ISO 15745 DEFINED DATA TYPES *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileHandle_DataType">
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileLocation" type="xsd:anyURI" minOccurs="0"
maxOccurs="1"/>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

E.4.6.2 XML Схема: CommonElements.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    ">
<!--################ common attribute group-->
<xsd:attributeGroup name="ag_formatAndFile">
<xsd:attribute name="formatName" type="xsd:string" fixed="DDXML" form="unqualified"/>
```

```
<xsd:attribute name="formatVersion" type="xsd:string" fixed="2.0" form="unqualified"/>
<xsd:attribute name="fileName" type="xsd:string" use="required" form="unqualified"/>
<xsd:attribute name="fileCreator" type="xsd:string" use="required" form="unqualified"/>
<xsd:attribute name="fileCreationDate" type="xsd:date" use="required"
form="unqualified"/>
<xsd:attribute name="fileCreationTime" type="xsd:time" use="optional"/>
<xsd:attribute name="fileModificationDate" type="xsd:date" use="optional"
form="unqualified"/>
<xsd:attribute name="fileModificationTime" type="xsd:time" use="optional"/>
<xsd:attribute name="fileModifiedBy" type="xsd:string" use="optional"/>
<xsd:attribute name="fileVersion" type="xsd:string" use="required" form="unqualified"/>
</xsd:attributeGroup>
<!--################ common groups-->
<xsd:group name="g_labels">
<xsd:sequence>
<xsd:choice maxOccurs="unbounded">
<xsd:element name="label">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="lang" type="xsd:language" use="required"/>
<xsd:attribute name="URI" type="xsd:anyURI" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="description">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="lang" type="xsd:language" use="required"/>
<xsd:attribute name="URI" type="xsd:anyURI" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="labelRef">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:anyURI">
<xsd:attribute name="dictID" type="xsd:IDREF" use="required"/>
<xsd:attribute name="textID" type="xsd:string" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
</xsd:choice>
</xsd:sequence>
</xsd:group>
<xsd:group name="g_simple">
<xsd:choice>
```

```
<xsd:element name="BOOL"/>
<xsd:element name="BITSTRING"/>
<xsd:element name="BYTE"/>
<xsd:element name="CHAR"/>
<xsd:element name="WORD"/>
<xsd:element name="DWORD"/>
<xsd:element name="LWORD"/>
<xsd:element name="SINT"/>
<xsd:element name="INT"/>
<xsd:element name="DINT"/>
<xsd:element name="LINT"/>
<xsd:element name="USINT"/>
<xsd:element name="UINT"/>
<xsd:element name="UDINT"/>
<xsd:element name="ULINT"/>
<xsd:element name="REAL"/>
<xsd:element name="LREAL"/>
<xsd:element name="TIME"/>
<xsd:element name="DATE"/>
<xsd:element name="DT"/>
<xsd:element name="TOD"/>
<xsd:element name="STRING"/>
<xsd:element name="WSTRING"/>
</xsd:choice>
</xsd:group>
<!--################ common elements-->
<xsd:element name="vendorID">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="deviceFamily">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="productID">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="version">
```

```
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="versionType" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="SW"/>
<xsd:enumeration value="FW"/>
<xsd:enumeration value="HW"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="buildDate" type="xsd:date"/>
<xsd:element name="specificationRevision">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

E.4.6.3 XML схема: ProfileBody_Device_ModbusTCP.xsd

XML Схема ProfileBody_Device_ModbusTCP.xsd включает схему ISO15745ProfileContainer.xsd в E.4.6.1 и схему CommonElements.xsd в E.4.6.2.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                                    ">
<xsd:include schemaLocation="ISO15745ProfileContainer.xsd"/>
<xsd:include schemaLocation="CommonElements.xsd"/>
<!--################ profile body device -->
<xsd:complexType name="ProfileBody_Device_ModbusTCP">
<xsd:complexContent>
<xsd:extension base="ProfileBody_DataType">
<xsd:sequence>
<xsd:element ref="DeviceIdentity" minOccurs="0"/>
<xsd:element ref="DeviceManager" minOccurs="0"/>
<xsd:element ref="DeviceFunction" maxOccurs="unbounded"/>
<xsd:element ref="ApplicationProcess" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element name="ExternalProfileHandle" type="ProfileHandle_DataType"
minOccurs="0"
maxOccurs="unbounded"/>
```

```
</xsd:sequence>
<xsd:attributeGroup ref="ag_formatAndFile"/>
<xsd:attribute name="supportedLanguages" type="xsd:NMTOKENS" use="optional"/>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<!--################ device identity elements -->
<xsd:element name="DeviceIdentity">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="vendorName"/>
<xsd:element ref="vendorID" minOccurs="0"/>
<xsd:element ref="vendorText" minOccurs="0"/>
<xsd:element ref="deviceFamily" minOccurs="0"/>
<xsd:element ref="productFamily" minOccurs="0"/>
<xsd:element ref="productName"/>
<xsd:element ref="productID" minOccurs="0"/>
<xsd:element ref="productText" minOccurs="0"/>
<xsd:element ref="orderNumber" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="version" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="buildDate" minOccurs="0"/>
<xsd:element ref="specificationRevision" minOccurs="0"/>
<xsd:element ref="instanceName" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="productFamily">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="instanceName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="false"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="orderNumber">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="productName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="productText">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="vendorName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="vendorText">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:complexType>
</xsd:element>
<!--################ device manager elements -->
<xsd:element name="DeviceManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="indicatorList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="indicatorList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="LEDList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="LEDList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="LED" maxOccurs="unbounded"/>
<xsd:element ref="combinedState" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="LED">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element ref="LEDstate" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="LEDcolors" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="monocolor"/>
<xsd:enumeration value="bicolor"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="LEDtype" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="IO"/>
<xsd:enumeration value="device"/>
<xsd:enumeration value="communication"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="LEDstate">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="state" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="on"/>
<xsd:enumeration value="off"/>
<xsd:enumeration value="flashing"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="LEDcolor" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="green"/>
<xsd:enumeration value="amber"/>
<xsd:enumeration value="red"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="flashingPeriod" type="xsd:unsignedInt" use="optional"/>
<xsd:attribute name="impulsWidth" type="xsd:unsignedByte" default="50"/>
<xsd:attribute name="numberOfImpulses" type="xsd:unsignedByte" default="1"/>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="combinedState">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element name="LEDstateRef" minOccurs="2" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="stateIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--################ device function elements -->
<xsd:element name="DeviceFunction">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="capabilities"/>
<xsd:element ref="picturesList" minOccurs="0"/>
<xsd:element ref="dictionaryList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="capabilities">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="characteristicsList" maxOccurs="unbounded"/>
<xsd:element ref="standardComplianceList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristicsList">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="category" minOccurs="0">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element ref="characteristic" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristic">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="characteristicName"/>
<xsd:element ref="characteristicContent" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="characteristicContent">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristicName">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="standardComplianceList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="compliantWith" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="compliantWith">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="range" default="international">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="international"/>
<xsd:enumeration value="internal"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="picturesList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="picture" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="picture">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="URI" type="xsd:anyURI" use="required"/>
<xsd:attribute name="number" type="xsd:unsignedInt" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="dictionaryList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="dictionary" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="dictionary">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="file" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="lang" type="xsd:language" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="file">
<xsd:complexType>
<xsd:attribute name="URI" type="xsd:anyURI" use="required"/>
</xsd:complexType>
</xsd:element>
<!--################ application process elements -->
<xsd:element name="ApplicationProcess">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="dataTypeList" minOccurs="0"/>
<xsd:element ref="functionTypeList"/>
<xsd:element ref="functionInstanceList"/>
<xsd:element ref="parameterList"/>
<xsd:element ref="parameterGroupList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dataTypeList">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_complex" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionTypeList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="functionType" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionType">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="versionInfo" maxOccurs="unbounded"/>
<xsd:element ref="interfaceList"/>
<xsd:element ref="functionInstanceList" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
<xsd:attribute name="package" type="xsd:string" use="optional"/>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="versionInfo">
<xsd:complexType>
<xsd:attribute name="organization" type="xsd:string" use="required"/>
<xsd:attribute name="version" type="xsd:string" use="required"/>
<xsd:attribute name="author" type="xsd:string" use="required"/>
<xsd:attribute name="date" type="xsd:date" use="required"/>
<xsd:attribute name="remarks" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="interfaceList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="inputVars" minOccurs="0"/>
<xsd:element ref="outputVars" minOccurs="0"/>
<xsd:element ref="configVars" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="inputVars">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="outputVars">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="configVars">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="varDeclaration">
<xsd:complexType>
<xsd:choice>
<xsd:group ref="g_simple"/>
<xsd:element ref="dataTypeIDRef"/>
</xsd:choice>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="size" type="xsd:string" use="optional"/>
<xsd:attribute name="initialValue" type="xsd:string" use="optional"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="dataTypeIDRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionInstanceList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="functionInstance" maxOccurs="unbounded"/>
<xsd:element ref="connection" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionInstance">
<xsd:complexType>
<xsd:attribute name="name" type="xsd:string" use="required"/>

<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="typeIDRef" type="xsd:IDREF" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="connection">
<xsd:complexType>
<xsd:attribute name="source" type="xsd:string" use="required"/>
<xsd:attribute name="destination" type="xsd:string" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="parameter" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameter">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element ref="variableRef" maxOccurs="unbounded"/>
<xsd:element ref="conditionalSupport" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="denotation" minOccurs="0"/>
<xsd:element ref="actualValue" minOccurs="0"/>
<xsd:element ref="defaultValue" minOccurs="0"/>
<xsd:element ref="substituteValue" minOccurs="0"/>
<xsd:element ref="allowedValues" minOccurs="0"/>
<xsd:element ref="unit" minOccurs="0"/>
<xsd:element ref="property" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attributeGroup ref="ag_parameter"/>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="variableRef">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="instanceIDRef" maxOccurs="unbounded"/>
<xsd:element ref="variableIDRef"/>
<xsd:element ref="memberRef" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="position" type="xsd:unsignedByte" default="1"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="instanceIDRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="variableIDRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="memberRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="optional"/>
<xsd:attribute name="index" type="xsd:long" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="actualValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="conditionalSupport">
<xsd:complexType>
<xsd:attribute name="paramIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="denotation">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="defaultValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="substituteValue">
<xsd:complexType>
```

```
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="allowedValues">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="value" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="range" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="value">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="range">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="minValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="maxValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="unit">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="multiplier" type="xsd:string" use="required"/>
<xsd:attribute name="unitURI" type="xsd:anyURI" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="property">
<xsd:complexType>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="value" type="xsd:string" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterGroupList">
<xsd:complexType>
<xsd:sequence>
```

```
<xsd:element ref="parameterGroup" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterGroup">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element ref="parameterGroup" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="parameterRef" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="kindOfAccess" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<!--################ complex types -->
<xsd:element name="array">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="subrange" maxOccurs="unbounded"/>
<xsd:choice>
<xsd:group ref="g_simple"/>
<xsd:element ref="dataTypeIDRef"/>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="subrange">
<xsd:complexType>
<xsd:attribute name="lowerLimit" type="xsd:long" use="required"/>
<xsd:attribute name="upperLimit" type="xsd:long" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="struct">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="count" minOccurs="0"/>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="enum">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="enumValue" maxOccurs="unbounded"/>
<xsd:group ref="g_simple" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="size" type="xsd:string" use="optional"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="enumValue">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="value" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="derived">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="count" minOccurs="0"/>
<xsd:choice>
<xsd:group ref="g_simple"/>
<xsd:element ref="dataTypeIDRef"/>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="count">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:element ref="defaultValue"/>
<xsd:element ref="allowedValues" minOccurs="0"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="access" default="read">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="read"/>
<xsd:enumeration value="write"/>
<xsd:enumeration value="readWrite"/>
<xsd:enumeration value="noAccess"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<!--################ group-->
```

```
<xsd:group name="g_complex">
<xsd:choice>
<xsd:element ref="array"/>
<xsd:element ref="struct"/>
<xsd:element ref="enum"/>
<xsd:element ref="derived"/>
</xsd:choice>
</xsd:group>
<!--################ attribute groups-->
<xsd:attributeGroup name="ag_parameter">
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="access" default="read">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="read"/>
<xsd:enumeration value="write"/>
<xsd:enumeration value="readWrite"/>
<xsd:enumeration value="noAccess"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="support" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="mandatory"/>
<xsd:enumeration value="optional"/>
<xsd:enumeration value="conditional"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="persistence" type="xsd:boolean" default="false"/>
<xsd:attribute name="offset" type="xsd:string" use="optional"/>
<xsd:attribute name="multiplier" type="xsd:string" use="optional"/>
</xsd:attributeGroup>
<xsd:attributeGroup name="ag_value">
<xsd:attribute name="value" type="xsd:string" use="required"/>
<xsd:attribute name="offset" type="xsd:string" use="optional"/>
<xsd:attribute name="multiplier" type="xsd:string" use="optional"/>
</xsd:attributeGroup>
</xsd:schema>
```

E.5 Описание шаблона профиля коммуникационной сети

E.5.1 ProfileBody_CommunicationNetwork_ModbusTCP

Настоящий стандарт определяет профиль коммуникационной сети Modbus TCP.

ProfileBody_CommunicationNetwork_ModbusTCP содержит ApplicationLayers (уровни приложения), TransportLayers (транспортные уровни) и элементы NetworkManagement (менеджмент сети), указанные на рисунке 25.

E.5.2 ApplicationLayers (уровни приложения)

E.5.2.1 Общие положения

На рисунке E.9 представлена структура класса Modbus TCP ApplicationLayers.

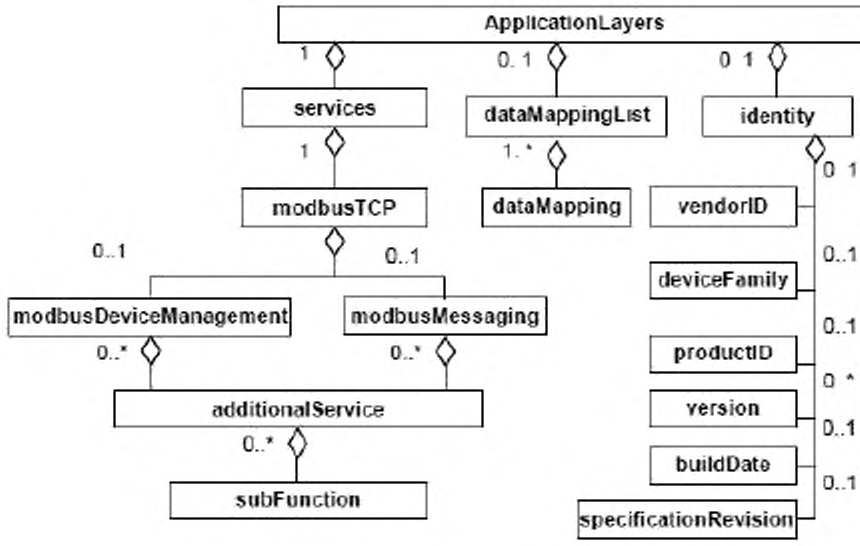


Рисунок E.9 — Класс Modbus TCP ApplicationLayers (уровни приложения)

Элемент ApplicationLayers содержит атрибуты, указанные в таблице E.39.

Таблица Е.39 — Атрибуты элемента ApplicationLayers (уровни приложения)

Атрибут	Тип данных	Применение	Описание
conformanceClass	xsd:string	Требуемое	Класс соответствие типа тип устройства (см. примечание)
communicationEntityType	xsd:NMTokens	По умолчанию	Тип сущности коммуникаций – если поддерживаются несколько типов, их имена должны быть разделены символами пробела, например «ведущий ведомый»; имена типов должны быть выбраны из следующего перечисления имен: — ведомый; — ведущий; — сервер; — клиент; — взаимосвязь (например, взаимосвязью является функция шлюза); — сущность равноценных коммуникаций, которые действуют как клиент и сервер
Примечание — Используя этот атрибут возможно выполнить классификацию устройства согласно поддерживаемым сервисам протокола коммуникаций			

Е.5.2.2 Элемент identity (идентичность)

Поскольку различные профили коммуникаций могут требовать различную информацию об идентичности, может быть использован опциональный субэлемент локальной идентичности элемента ApplicationLayers (уровни приложения). Элемент identity может содержать подмножество субэлементов элемента DeviceIdentity (идентичность устройства), описанного в 6.5.1.2. Все описания субэлементов, приведенные в вышеуказанном пункте применяются также к субэлементам данного элемента identity.

E.5.2.3 Элементы services / ModbusTCP

E.5.2.3.1 Общие положения

Единичный элемент services описывает специальные характеристики профиля коммуникаций типа устройства. Данное приложение только указывает элемент modbusTCP, соответствующий сервису Modbus TCP.

Элемент modbusTCP описывает дополнительные сервисы Modbus, поддерживаемые устройством, которые не определены с помощью атрибута conformanceClass (класс соответствия) элемента ApplicationLayers. Он имеет опциональные субэлементы modbusMessaging (modbus сообщения) и modbusDeviceManagement (менеджмент устройства modbus), как показано на рисунке E.9.

Элемент modbusTCP содержит атрибуты, указанные в таблице E.40.

Таблица E.40 — Атрибуты элемента modbusTCP

Атрибут	Тип данных	Применение	Описание
Type	xsd:NMTokens	Требуемое	Тип конечной точки сообщения – если поддерживается комбинация типов, их имена должны быть разделены символом пробела, например «сервер шлюз»; правильные имена типов: — клиент — сервер — шлюз
accessControl	xsd:Boolean	По умолчанию	Определяет, поддерживает или нет тип устройства контроль доступа для удаленных адресов IP: ложь (по умолчанию), истина
maxNbOfTotalConnection	xsd:unsignedInt	Опциональное	Максимальное число конкурирующих соединений Modbus TCP, поддерживаемых типом устройства; требования: ПолнСоед = СерверСоед + КлиентСоед

Окончание таблицы Е.40

Атрибут	Тип данных	Применение	Описание
maxNbOfServerConnection	xsd:unsignedInt	Опциональное	Максимальное число конкурирующих серверных соединений Modbus TCP, поддерживаемых типом устройства
maxNbOfClientConnection	xsd:unsignedInt	Опциональное	Максимальное число конкурирующих клиентских соединений Modbus TCP, поддерживаемых типом устройства
maxSizeOfGatewayRouting Table	xsd:unsignedInt	Опциональное	Если применимо, максимальный размер таблицы маршрутов шлюза, поддерживаемый типом устройства

Е.5.2.3.2 Элемент modbusMessaging (modbus сообщения)

Если он присутствует, опциональный элемент modbusMessaging содержит последовательность одного или нескольких субэлементов additionalService (дополнительный сервис), которые создают нумерацию дополнительных сервисов Modbus, обеспечиваемых устройством. Элемент modbusMessaging не имеет атрибутов.

Е.5.2.3.3 Элемент modbusDeviceManagement (менеджмент устройства modbus)

Если он присутствует, опциональный элемент modbusDeviceMessaging содержит последовательность одного или нескольких субэлементов additionalService (дополнительный сервис), которые создают нумерацию дополнительных сервисов менеджмента устройств Modbus, обеспечиваемых устройством. Элемент modbusDeviceMessaging не имеет атрибутов.

Е.5.2.3.4 Элемент additionalService (дополнительный сервис)

Субэлемент additionalService элемента modbusMessaging, представленного на рисунке Е.9, создает возможность определения дополнительных сервисов Modbus, поддерживаемых типом устройства, и содержит атрибуты, указанные в таблице Е.41.

Таблица Е.41 — Атрибуты элемента additionalService (дополнительный сервис)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя дополнительного поддерживаемого сервиса Modbus
functionCode	xsd:unsignedByte	Требуемое	Код функции дополнительного поддерживаемого сервиса Modbus
subFunctionCode	xsd:unsignedByte	Опциональное	Код субфункции дополнительного поддерживаемого сервиса Modbus, если имеется (см. примечание)
Description	xsd:string	Опциональное	Опциональное текстовое описание дополнительного поддерживаемого сервиса Modbus
Примечание — Данный атрибут пропускают, если не поддерживается код субфункции или если более чем один код субфункции поддерживается сервисом. Если поддерживается более чем один код субфункции, то используют субэлемент subFunction (подфункция).			

E.5.2.3.5 Элемент subFunction (подфункция)

Данный элемент используется в случаях, когда несколько кодов субфункции ассоциированы с сервисом коммуникаций Modbus.

Элемент subFunction содержит атрибуты, указанные в таблице Е.42.

Таблица Е.42 — Атрибуты элемента subFunction (подфункция)

Атрибут	Тип данных	Применение	Описание
Code	xsd:unsignedByte	Требуемое	Код субфункции поддерживаемого сервиса Modbus
Description	xsd:string	Опциональное	Текстовое описание кода субфункции

E.5.2.4 Элемент dataMappingList (список отображения данных)

Элемент dataMappingList, представленный на рисунке Е.9, содержит последовательность из одного или нескольких субэлементов dataMapping

(отображение данных), которые определяют ассоциации между конкретными адресами протокола сообщаемых данных и соответствующими элементами параметров внутри элемента ApplicationProcess (прикладной процесс).

Элемент dataMappingList содержит атрибут, указанный в таблице Е.43.

Таблица Е.43 — Атрибуты элемента dataMappingList (список отображения данных)

Атрибут	Тип данных	Применение	Описание
Description	xsd:string	Опциональное	Опциональное текстовое описание списка отображения данных

Элемент dataMapping (отображение данных) имеет часть с пустым контентом и содержит атрибут, указанные в таблице Е.44.

Таблица Е.44 — Атрибуты элемента dataMapping (отображение данных)

Атрибут	Тип данных	Применение	Описание
parameterIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного элемента parameter
accessPath	xsd:string	Требуемое	Значение конкретного адреса протокола
Services	xsd:NMTokens	Опциональное	Имена сервисов, допустимых для данного параметра

Е.5.3 TransportLayers (транспортные уровни)

Е.5.3.1 Общие положения

Элемент TransportLayers содержит элемент PhysicalLayer (физический уровень) и элементы MacLinkLayer (управление доступом к среде канального уровня), указанные на рисунке Е.10.

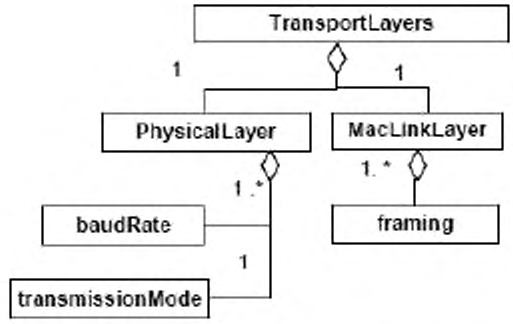


Рисунок Е.10 — Класс Modbus TCP TransportLayers (транспортные уровни)

Е.5.3.2 PhysicalLayer (физический уровень)

Е.5.3.2.1 Общие положения

Элемент PhysicalLayer, представленный на рисунке Е.10, содержит последовательность обязательных субэлементов, определяющих свойства физических уровней коммуникационной сущности. Субэлемент baudRate (скорость в бодах) должен появляться от одного до нескольких раз в том случае, если имеется один обязательный элемент transmissionMode (режим передачи). Все субэлементы определяют правильные значения для их различных атрибутов.

Е.5.3.2.2 Элемент baudRate (скорость в бодах)

Элемент baudRate предоставляет одно значение скорости в бодах, поддерживаемое соединением Ethernet. Если поддерживаются несколько значений скорости в бодах, этот элемент появляется несколько раз. Элемент baudRate имеет пустой контент и атрибут, указанный в таблице Е.45.

Таблица Е.45 — Атрибут элемента baudRate (скорость в бодах)

Атрибут	Тип данных	Применение	Описание
Value	xsd:string	Требуемое	Поддерживаемое значение скорости в бодах; правильные значения - 10 и 100 Мб/с

Е.5.3.2.3 Элемент transmissionMode (режим передачи)

Элемент transmissionMode предоставляет режим передачи, поддерживаемый соединением Ethernet. Элемент transmissionMode имеет пустой контент и атрибут, указанный в таблице Е.46.

Таблица Е.46 — Атрибут элемента transmissionMode (режим передачи)

Атрибут	Тип данных	Применение	Описание
Mode	xsd:string	Требуемое	Поддерживаемый режим передачи; правильные значения – полудуплексные, дуплексные и автоматически согласованные

Е.5.3.3 MacLinkLayer (управление доступом к среде канального уровня) / framing (формирование кадров)

Элемент MacLinkLayer содержит только элемент framing (формирование кадра), указанный на рисунке Е.10.

Элемент framing предоставляет один тип формирования кадра, поддерживаемый соединением Ethernet. Если поддерживаются несколько типов формирования кадра, этот элемент появляется несколько раз. Элемент framing имеет пустой контент и атрибут, указанный в таблице Е.47.

Таблица Е.47 — Атрибут элемента framing(формирование кадра)

Атрибут	Тип данных	Применение	Описание
Type	xsd:string	Требуемое	Поддерживаемый тип формирования кадра; правильные значения - Ethernet II, IEEE 802.3 передатчик и IEEE 802.3 приемник

Е.5.4 NetworkManagement (менеджмент сети)

Е.5.4.1 Общие положения

Элемент NetworkManagement (менеджмент сети) содержит субэлемент SNMP, указанный на рисунке Е.11.

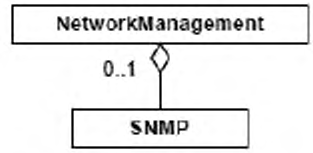


Рисунок Е.11 — Класс Modbus TCP NetworkManagement (менеджмент сети)

E.5.4.2 SNMP

Элемент SNMP, представленный на рисунке E.11 показывает, что сервис «Simple Network Management Protocol (простой протокол управления сетью)» (см. RFC 1157) поддерживается данным типом устройства.

Элемент SNMP содержит атрибут, указанный в таблице E.48.

Таблица E.48 — Атрибут элемента SNMP

Атрибут	Тип данных	Применение	Описание
Version	xsd:unsignedByte	По умолчанию	Версия поддерживаемого протокола управления сетью; правильные значения - snmpV1, snmpV2 и snmpV3

E.5.5 Схемы DDXML шаблона профиля коммуникационной сети

XML схема ProfileBody_CommunicationNetwork_ModbusTCP.xsd включает в себя схему ISO15745ProfileContainer.xsd, приведенную в E.4.6.1 и схему CommonElements.xsd, приведенную в E.4.6.2.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    >
<xsd:include schemaLocation="ISO15745ProfileContainer.xsd"/>
<xsd:include schemaLocation="CommonElements.xsd"/>
<!--################ profile body communication network -->
<xsd:complexType name="ProfileBody_CommunicationNetwork_ModbusTCP">
<xsd:complexContent>
<xsd:extension base="ProfileBody_DataType">
<xsd:choice>
<xsd:sequence>
<!--################ application layers elements -->
<xsd:element name="ApplicationLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="identity" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="vendorID" minOccurs="0"/>
<xsd:element ref="deviceFamily" minOccurs="0"/>
<xsd:element ref="productID" minOccurs="0"/>
<xsd:element ref="version" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="buildDate" minOccurs="0"/>
<xsd:element ref="specificationRevision" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dataMappingList" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
```

```
<xsd:element name="dataMapping" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="parameterIDRef" type="xsd:IDREF" use="required"/>
<xsd:attribute name="accessPath" type="xsd:string" use="required"/>
<xsd:attribute name="services" type="xsd:NMTOKENS" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="services">
<xsd:annotation>
<xsd:documentation>Ethernet TCP/IP services</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element name="modbusTCP">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="modbusMessaging" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="additionalService" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence minOccurs="0">
<xsd:element name="subFunction" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="code" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="functionCode" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="subFunctionCode" type="xsd:unsignedByte" use="optional"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="modbusDeviceManagement" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="additionalService" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence minOccurs="0">
<xsd:element name="subFunction" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="code" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
```

```
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="functionCode" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="subFunctionCode" type="xsd:unsignedByte" use="optional"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="type" type="xsd:NMTOKENS" use="required"/>
<xsd:attribute name="accessControl" type="xsd:boolean" default="false"/>
<xsd:attribute name="maxNbOfTotalConnection" type="xsd:unsignedInt" use="optional"/>
<xsd:attribute name="maxNbOfServerConnection" type="xsd:unsignedInt" use="optional"/>
<xsd:attribute name="maxNbOfClientConnection" type="xsd:unsignedInt" use="optional"/>
<xsd:attribute name="maxSizeOfGatewayRoutingTable" type="xsd:unsignedInt" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="conformanceClass" type="xsd:string" use="required"/>
<xsd:attribute name="communicationEntityType" type="xsd:NMTOKENS" default="slave">
<xsd:annotation>
<xsd:documentation>Defines the entity type: slave (default), master, client, server,
interconnection (example: gateway), peer (acts as client and server)</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<!--################ transport layers elements -->
<xsd:element name="TransportLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MacLinkLayer">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="framing" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="type" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="Ethernet II"/>
<xsd:enumeration value="IEEE 802.3 sender"/>
<xsd:enumeration value="IEEE 802.3 receiver"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
```

```
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="PhysicalLayer">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="baudRate" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="value" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="10 Mbps"/>
<xsd:enumeration value="100 Mbps"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="transmissionMode">
<xsd:complexType>
<xsd:attribute name="mode" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="half-duplex"/>
<xsd:enumeration value="full-duplex"/>
<xsd:enumeration value="auto-negotiation"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--################ network management elements -->
<xsd:element name="NetworkManagement" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="SNMP" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="version" default="snmpV1">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="snmpV1"/>
<xsd:enumeration value="snmpV2"/>
<xsd:enumeration value="snmpV3"/>
</xsd:restriction>
```

```
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:element name="ExternalProfileHandle" type="ProfileHandle_DataType"/>
</xsd:choice>
<xsd:attributeGroup ref="ag_formatAndFile"/>
<xsd:attribute name="supportedLanguages" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKENS"/>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:schema>
```

Приложение F

(обязательное)

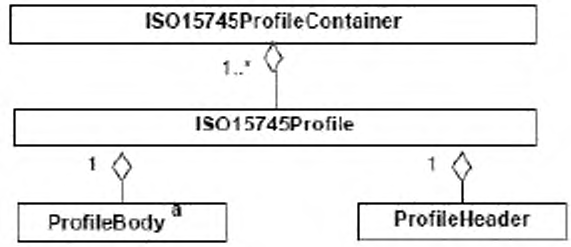
Шаблоны профиля EtherCAT

F.1 Обзор

EtherCAT представляет собой основанную на Ethernet систему коммуникаций, спецификация которой содержится в МЭК/ПАС 62407.

В EtherCAT используется концепция мультипрофильного контейнера, описанная в изменении 1 ИСО 15745-1:2003 для XML файлов профилей. Следовательно, шаблоны профилей EtherCAT основаны на переменном шаблоне мастер профиля ISO15745ProfileContainer, установленном в изменении ИСО 15745-1:2003.

На рисунке F.1 представлена структура профиля XML EtherCAT.



[a] Используют два типа ProfileBody (тело профиля): ProfileBody_Device_EtherCAT и ProfileBody_CommunicationNetwork_EtherCAT

Рисунок F.1 — Шаблон профиля EtherCAT

Имя ProfileTechnology (технология профиля) - EtherCAT.

F.2 Общие правила

F.2.1 Использование уникальных идентификаторов

Элемент может иметь атрибут uniqueID (уникальный идентификатор) типа xsd:ID. Следовательно, идентификатор должен быть уникальным во всем файле XML. Элемент, ссылающийся на уникальный идентификатор, содержит named attribute (атрибут с именем) типа xsd:IDREF.

Уникальные идентификаторы могут быть созданы двумя способами. Одна возможность заключается в построении строки из имени элемента и вычисляемого путем увеличения номера числа. Второй способ заключается в конкатенации строк родительских элементов. Оба метода гарантируют уникальность строки.

F.2.2 Поддержка языка

F.2.2.1 Общие положения

Профили устройства, соответствующие схеме XML, описанной в данном приложении, должны поддерживать различные языки, так как в этом случае инструменты будут способны использовать имена из файла XML для их отображения на дисплее интерфейса пользователя. Например в средствах интерфейса инструмента пользователя могут быть представлены параметры коммуникации.

Поддержка языка осуществляется с помощью группы меток g_labels. Каждое имя элемента, которое может быть отображено на дисплее, и, следовательно, является зависимым от языка, содержится внутри схемы как элемент g_labels. В качестве опции к элементу метки может быть добавлен в виде атрибута URI.

Пример —

Для данного имени параметра:

- ***немецкий: Baudrate;***
- ***английский: Baud rate;***
- ***французский: Vitesse de transmission***

F.2.2.2 Элемент g_labels

Группа g_labels (метки) поддерживает введение label (имени) и описание в контексте родительского элемента (см. рисунок F.2).

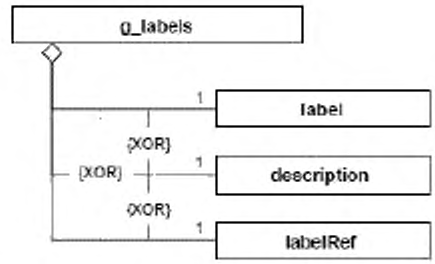


Рисунок F.2 — Группа g_labels

Каждый элемент, для которого необходимо имя или описание, должен выбрать только один из элементов, предназначенных для выполнения этой задачи: метку, описание или ссылочную метку.

1. Элемент label (метка) позволяет хранить идентифицирующее имя и текст описания внутри самого файла XML. Элемент label должен иметь атрибуты, указанные в таблице F.1.

Таблица F.1 — Атрибуты элемента label (метка)

Атрибут	Тип данных	Применение	Описание
lang	xsd:language	Требуемое	Язык, используемый в имени или описании
URI	xsd:anyURI	Опциональное	Опциональная связь для получения дальнейшей информации описания

Элемент может появляться n раз, по одиному разу для каждого языка. Для идентификации языка используется атрибут lang.

2 Элемент description (описание) обеспечивает возможность хранения текстовых описаний внутри файла XML. Элемент может появиться несколько раз, по одному разу для каждого языка. Элемент описания имеет те же атрибуты, что и элемент метки.

3 Элемент labelRef (ссылочная метка) допускает возможность ссылки на тексты описания внутри файла - источника внешних текстов.

Элемент labelRef предоставляет указатель с помощью атрибутов dictID (идентификатор словаря) и textID (идентификатор текста) на текстовые входы в отдельном файле-источнике текстов. На файлы источники текста имеется ссылка в субэлементах словаря элемента DeviceFunction (функция устройства). Файлы источники текста могут быть любыми файлами,

содержащими последовательности символов и другую информацию, например рисунки.

Элемент labelRef также может появляться n раз, позволяя осуществлять ссылки на несколько входов словаря, которые содержат связи с файлами на различных языках. Соответствующие языки определены в атрибуте lang элемента словаря.

Элемент labelRef содержит атрибуты, указанные в таблице F.2.

Таблица F.2 — Атрибуты элемента labelRef(ссылочная метка)

Атрибут	Тип данных	Применение	Описание
dictID	xsd:IDREF	Требуемое	Производится ссылка на один элемент словаря элемента dictionaryList (список словарей); элемент словаря содержит связь с внешним файлом – источником текста
textID	xsd:string	Опциональное	Производится ссылка на последовательность символов внешнего файла – источника текста с помощью сравнения с образцом

F.2.2.3 Идентификатор языка

Для реализации мультиязыковой поддержки каждая метка получает атрибут, содержащий код языка. Код языка соответствует содержанию элемента метки.

С целью проверки, какие языки поддерживаются в файле XML, в атрибуте supportedLanguages (поддерживаемые языки) элемента ProfileBody (тело профиля) содержится список поддерживаемых языков.

F.2.2.4 Атрибут lang

Идентификатор языка lang состоит из комбинации кода языка (согласно определению, приведенному в ИСО 639-1), и необязательного символа «тире», и необязательного кода страны (согласно определению, приведенному в ИСО 3166-1). Атрибут lang является атрибутом элемента метки.

Некоторые значения lang приведены в таблице F.3.

Таблица F.3 — Значения атрибута lang

Язык	Значение lang
Английский (США)	en-us
Немецкий (стандартный) Французский (стандартный)	de fr
Испанский (стандартный)	es
Итальянский (стандартный)	It
Португальский (Бразилия)	pt-br

F.2.2.5 Атрибут SupportedLanguages (поддерживаемые языки)

Атрибут supportedLanguages определяет поддерживаемые языки и состоит из списка кодов языков плюс необязательные коды страны.

Пример — supportedLanguages="en-us de fr es"

F.2.2.6 URI

Общий механизм допускает описание URI в контексте элемента метки. URI реализуется с помощью опционального атрибута URI.

Пример — Это используется, например, в контексте метки поставщика, метки параметра, или метки сервиса.

F.3 ProfileHeader (заголовок профиля)

В целях облегчения идентификации профиля, заголовок профиля устройства должен соответствовать представленном на рисунке F.3 модели, соответствующей ИСО 15745-1.

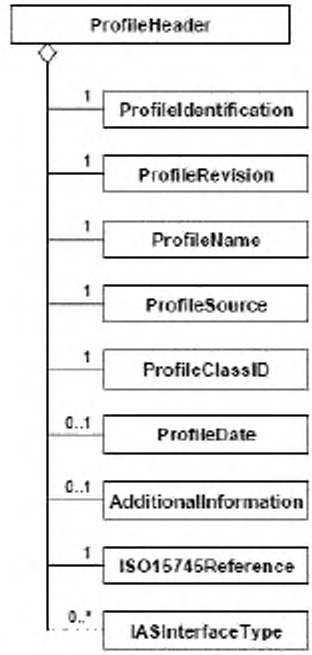


Рисунок F.3 — Диаграмма класса Profile header (заголовок профиля)

Элемент ProfileHeader состоит из следующих элементов:

— элемента ProfileIdentification (идентификация профиля), идентифицирующего текущий профиль;

— элемента ProfileRevision (версия профиля), идентифицирующего текущую версию профиля;

— элемента ProfileName (имя профиля), содержащего описательное имя текущего профиля на английском языке. При наличии в профиле устройства более одного элемента ProfileBody считают, что значение элемента ProfileName (имя профиля) должно быть конкатенацией значений элементов productName (имя продукта) соответствующих элементах DeviceIdentity (идентичность устройства);

— элемента ProfileSource (источник профиля), идентифицирующего средство проверки допустимости текущего профиля;

— элемента ProfileClassID (идентификатор класса профиля), идентифицирующего класс текущего профиля согласно ИСО 15745-1;

— элемента ISO15745Reference, указывающего на часть ИСО 15745, издание и технологию, которым соответствует описание.

F.4 Описание шаблона профиля устройства

F.4.1 ProfileBody_Device_EtherCat

В данном разделе определен профиль устройства EtherCat.

ProfileBody_Device_EtherCat содержит элементы DeviceIdentity (идентичность устройства), DeviceManager (менеджер устройства), DeviceFunction (функция устройства) и ApplicationProcess (прикладной процесс), представленные на рисунке 26.

Элемент ProfileBody содержит описания:

— единичного устройства (например, датчика ближней локации или электромеханического концевого выключателя) или более сложного устройства (например, прерывателя цепи с количеством параметров до 2500 и с более чем 100 функциями) или

— части устройства, называемой также «модуль» в области программируемых логических контроллеров (например, части контроллера ввода/вывода или блока электрической защиты).

Элемент ProfileBody содержит атрибуты, указанные в таблице F.4.

Таблица F.4 — Атрибуты элемента ProfileBody (тело профиля)

Атрибут	Тип данных	Применение	Описание
formatName	xsd:string	Фиксированное	Идентификатор формата
formatVersion	xsd:string	Фиксированное	Идентификатор версии формата
fileName	xsd:string	Требуемое	Имя файла с расширением без пути
fileCreator	xsd:string	Требуемое	Лицо, разработавшее файл
fileCreationDate	xsd:date	Требуемое	Дата создания файла
fileCreationTime	xsd:time	Опциональное	Время создания файла
fileModifiedBy	xsd:string	Опциональное	Лицо, изменившее файл
fileModificationDate	xsd.date	Опциональное	Дата последнего изменения файла
fileModificationTime	xsd:time	Опциональное	Время последнего изменения файла
fileVersion	xsd:string	Требуемое	Установленная поставщиком версия файла
supportedLanguages	xsd:NMTOKENS	Опциональное	Список поддерживаемых языков

F.4.2 DeviceIdentity (идентичность устройства)

F.4.2.1 Общие положения

Класс DeviceIdentity (см. рисунок 27) содержит элементы, не зависящие от сети и процесса, и описывает идентичность одного устройства или группы устройств.

Таблица F.5 устанавливает атрибут readOnly (только чтение), который присоединяется к элементам vendorName (имя поставщика), vendorID (идентификатор поставщика), vendorText (текст поставщика), deviceFamily (семейство устройства), productFamily (семейство продуктов), productName (имя продукта), productID (идентификатор продукта), productText (текст продукта), orderNumber (номер заказа), version (версия), specificationRevision (спецификация версии) и instanceName (имя экземпляра).

Таблица F.5 — Атрибут элемента vendorName (имя поставщика)

Атрибут	Тип данных	Применение	Описание
readOnly	xsd:Boolean	По умолчанию	Указывает является ли для пользователя значение атрибута только чтение: ложь, истина (по умолчанию)

F.4.2.2 Элемент vendorName (имя поставщика)

Элемент vendorName определяет имя или торговую марку поставщика устройства.

F.4.2.3 Элемент vendorID (идентификатор поставщика)

Элемент vendorID идентифицирует поставщика. Эта информация должна быть заполнена при признании описанного продукта и подтверждении консорциумом.

Примечание — Зависящие от консорциума семейства продуктов и идентификаторы поставщика взаимосвязаны.

F.4.2.4 Элемент vendorText (текст поставщика)

Элемент vendorText предназначен для предоставления поставщиком дополнительной информации о компании, например адреса или номера

горячей линии связи. Группа g_labels обеспечивает возможность включения URI поставщика в элемент vendorText.

F.4.2.5 Элемент deviceFamily (семейство устройства)

Элемент deviceFamily определяет семейство устройства.

Пример —

Примерами семейства устройства являются следующие:

- ***привод с переменной скоростью;***
- ***прерыватель цепи;***
- ***датчик давления***

F.4.2.6 Элемент productFamily (семейство продуктов)

Элемент productFamily устанавливает определенную поставщиком принадлежность типа устройства к некоторому набору устройств в семействе. Действительные значения productFamily – система, инструмент, отношение к консорциуму.

Примечание — Определяемые консорциумом семейства продуктов и идентификаторы поставщика взаимосвязаны.

F.4.2.7 Элемент productName (имя продукта)

Элемент productName устанавливает определенное поставщиком обозначение или наименование типа устройства.

F.4.2.8 Элемент productID (идентификатор продукта)

Элемент productID устанавливает определенную поставщиком уникальную идентификацию описываемого типа устройства.

F.4.2.9 Элемент productText (текст продукта)

Элемент productText обеспечивает для поставщика возможность предоставить короткое текстовое описание типа устройства.

F.4.2.10 Элемент orderNumber (номер заказа)

Элемент orderNumber используют для сохранения единичного номера заказа данного продукта или набора номеров различных заказов продуктов в семействе продуктов в зависимости от того, относится профиль устройства к одному продукту или семейству продуктов.

F.4.2.11 Элемент version (версия)

Элемент version используют для хранения различных типов информации о версиях. Возможны элементы для нескольких версий.

Элемент version имеет атрибуты, указанные в таблице F.6.

Таблица F.6 — Атрибуты элемента version (версия)

Атрибут	Тип данных	Применение	Описание
versionType	Xsd:NMToken	Требуемое	Тип версии: — SW - программная — FW - программно-аппаратная — HW - аппаратная
readOnly	Xsd:Boolean	По умолчанию	Указывает, является ли значение для пользователя только для чтения: ложь, истина (по умолчанию)

F.4.2.12 Элемент buildDate (создание даты)

Элемент buildDate определяет дату создания программного блока.

F.4.2.13 Элемент specificationRevision (спецификация версии)

Элемент specificationRevision содержит версию спецификации, которая соответствует устройству.

F.4.2.14 Элемент instanceName (имя экземпляра)

Данный элемент содержит имя экземпляра устройства.

F.4.3 DeviceManager (менеджер устройства)

F.4.3.1 Общие положения

Элемент DeviceManager определяет список индикаторов, предоставленных типом устройства, если они существуют.

F.4.3.2 LEDList (LED список)

F.4.3.2.1 Общие положения

Элемент LEDList (LED список), представленные на рисунке F.4, определяют число и тип индикаторов, содержащихся в типе устройства.

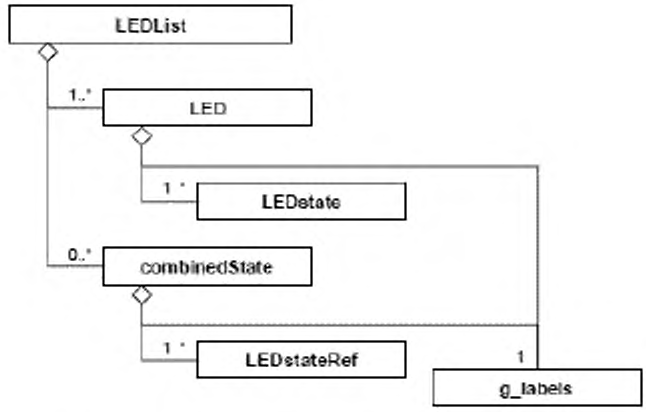


Рисунок F.4 — LEDList (LED список)

F.4.3.2.2 LED

Элемент LED описывает характеристики одного LED в данном типе устройства. Подробное описание характеристик может быть предоставлено в группе g_labels.

Дополнительные характеристики LED представлены в виде атрибутов элемента LED, указанных в таблице F.7.

Таблица F.7 — Атрибуты элемента LED

Атрибут	Тип данных	Применение	Описание
LEDcolors	xsd:string	Требуемое	Для цветов LED; правильные значения − монохромные и двухцветные
LEDtype	xsd:string	Опциональное	При грубой классификации контролируемых объектов или функциональных параметров; правильные значения – IO, устройство и коммуникации

Дополнительно элемент LED содержит один или более элементов LEDstate (LED состояние), которые определяют состояния устройства, сигнализируемые с помощью LED, и визуальные характеристики, используемые для сигнализации о состояниях.

Визуальные параметры, используемые для сигнализации о состояниях, закодированы в виде значений атрибутов элемента LEDstate согласно таблице F.8. Дополнительно состояниям LED присвоены уникальные идентификаторы.

Таблица F.8 — Атрибуты элемента LEDstate

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный ID для состояния LED; может быть объектом ссылки из элемента LEDstateRef
State	xsd:string	Требуемое	Состояние LED; возможные значения атрибута – включено, выключено, мигание
LEDcolor	xsd:string	Требуемое	Цвет состояния LED; правильные значения – зеленый, янтарный, красный
flashingPeriod	xsd:unsignedByte	Опциональное	Если состояние – мигание, то период мигания LED измеряют в миллисекундах
impulsWidth	xsd:unsignedByte	По умолчанию	Ширина импульса мигания в процентах периода мигания; если атрибут impulsWidth пропущен, то значение по умолчанию равно 50 %
numberOfImpulses	xsd:unsignedByte	По умолчанию	Число импульсов в случае более чем одного импульса мигания за период мигания; если атрибут присутствует, то атрибут impulsWidth должен быть представлен также, если атрибут numberOfImpulses пропущен, значение по умолчанию - 1

F.4.3.2.3 Элемент combinedState (комбинированное состояние)

Элемент combinedState разрешает индикацию состояний устройства, которые сигнализируются с помощью более одного LED.

Описание комбинированного состояния предоставлено в группе g_labels.

Ссылка на состояния LED, участвующие в сигнализации комбинированного состояния, должна осуществляться с помощью не менее двух субэлементов LEDstateRef (Ссылка на состояние LED) элемента combinedState (комбинированное состояние).

Ссылка на элемент LEDstate кодируется как значение атрибута единичного атрибута элемента LEDstateRef (см. таблицу F.9).

Таблица F.9 — Атрибуты элемента LEDstateRef (ссылка на состояние LED)

Атрибут	Тип данных	Применение	Описание
stateIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного элемента LEDstate

F.4.4 DeviceFunction (функция устройства)

F.4.4.1 Общие положения

Элемент DeviceFunction, показанный на рисунке 29, определяет вид устройства в каталоге, представленный в виде набора возможностей (capabilities), перечисляющих характеристики устройства и указывающих его соответствие различным стандартам.

F.4.4.2 Элемент capabilities (возможности)

F.4.4.2.1 Общие положения

Обязательный элемент capabilities описывает все функциональные параметры, их характеристики и другие основные параметры устройства, которые должны определяться с помощью инструментов, использующих профиль устройства для выбора изделий с такими же или аналогичными свойствами.

Элемент capabilities (возможности) описывает характеристики устройства в виде текста и содержит последовательность одного или более элементов characteristicsList (список характеристик) и опциональный элемент standardComplianceList (стандартный список соответствия).

F.4.4.2.2 Элемент characteristicsList (список характеристик)

F.4.4.2.2.1 Общие положения

Элемент characteristicsList (список характеристик) представляет собой совокупность характеристик. Этот элемент должен содержать не менее одного субэлемента characteristic (характеристика). Характеристики в списке могут быть ассоциированы с категорией, которая может быть выражена в виде текстового содержания субэлемента g_labels опционального субэлемента category (категория) элемента characteristicsList.

F.4.4.2.2.2 Элемент characteristic (характеристика)

Элемент characteristic описывает единичную характеристику устройства, и содержит обязательный элемент characteristicName (имя характеристики), а также один или несколько элементов characteristicContent (содержание характеристики).

F.4.4.2.2.3 Элемент characteristicName (имя характеристики)

Обязательный элемент characteristicName обозначает основную техническую характеристику устройства. Для наименований характеристик рекомендуется использовать словарь в перечне данных изделия.

Пример — «Максимальное рабочее напряжение», «Защита от перегрузки», «Электрическая прочность».

F.4.4.2.2.4 Элемент characteristicContent (содержание характеристики)

Этот обязательный элемент содержит значение элемента характеристики. Множественные значения могут быть выражены с помощью использования нескольких элементов characteristicContent.

Пример — Пример единичного значения для «Максимальное рабочее напряжение» – 680 В.

F.4.4.2.3 Элемент standardComplianceList (стандартный список соответствия)

Элемент standardComplianceList представляет собой собрание элементов compliantWith (соответствие с). Этот элемент является опциональным; если он присутствует, то содержит не менее одного субэлемента compliantWith.

Субэлемент compliantWith имеет атрибуты, которые устанавливают соответствие устройства международным стандартам или внутренним стандартам компании. Содержание типа g_labels этого элемента может включать в себя замечания, касающиеся этих стандартов.

Наименование или обозначение стандарта предоставляют с помощью требуемого атрибута name (имя) элемента compliantWith. Второй атрибут — значение по умолчанию range (диапазона) элемента compliantWith – определяет диапазон применимости стандарта согласно указаниям таблицы F.10.

Таблица F.10 — Атрибуты элемента compliantWith (соответствие с)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Наименование или обозначение стандарта
Range	xsd:NMToken	По умолчанию	Два возможных пронумерованных значения атрибута - международный (по умолчанию) или внутренний

F.4.4.3 Элемент picturesList (список изображений)

Элемент picturesList обеспечивает возможность связать изображения с профилем устройства и содержит один или более субэлементов с изображениями, доступ к которым обеспечивается через субэлемент g_labels.

В таблице F.11 определены атрибуты субэлемента изображения: опциональное число изображений, обязательная связь с внешними источниками, содержащими графическую информацию.

Таблица F.11 — Атрибуты элемента picture (изображение)

Атрибут	Тип данных	Применение	Описание
URI	xsd:anyURI	Требуемое	Связь с внешним источником
Номер	xsd:unsignedInt	Опциональное	Номер изображения

F.4.4.4 Элемент dictionaryList (список словарей)

Опциональный элемент dictionaryList обеспечивает возможность включения в профиль устройства связей с внешними файлами - источниками

текстов и содержит один или более элементов словарей, каждый из которых включает в себя один или более субэлементов файлов. Несколько файлов необходимо при наличии различных форматов файлов в словаре.

Обязательный атрибут lang типа xsd:language определяет языки, используемые в файлах, которые связаны с элементом словарь (см. таблицу F.12). Обязательный атрибут uniqueID (уникальный идентификатор) типа xsd:ID поддерживает уникальную идентификацию входа в словарь, на который имеется ссылка из атрибута dictID элемента labelRef согласно таблице F.2.

Таблица F.12 — Атрибуты элемента dictionary (словарь)

Атрибут	Тип данных	Применение	Описание
Lang	xsd:language	Требуемый	Язык, используемый для файлов, относящихся к входу в словарь
uniqueID	xsd:ID	Требуемый	Уникальный ID входа в словарь

Субэлемент file (файл) содержит обязательный атрибут, указанный в таблице F.13.

Таблица F.13 — Атрибуты элемента file (файл)

Атрибут	Тип данных	Применение	Описание
URI	xsd:anyURI	Требуемый	Связь с соответствующим файлом

F.4.5 ApplicationProcess (прикладной процесс)

F.4.5.1 Общие положения

Элемент ApplicationProcess представляет набор сервисов и параметров, формирующих поведение и интерфейсы устройства в терминах приложения, независимо от технологии устройства, и лежащих в его основе коммуникационных сетей и протоколов коммуникаций.

Субэлементы элемента ApplicationProcess, представленный на рисунке 30 обеспечивают общий подход к описанию функций устройства, имеющих произвольную, одноуровневую или иерархическую структуру.

Функции моделируются как типы функций, которые реализуются внутри устройства или, если необходима иерархическая структура, внутри типов функций. Параметры интерфейса этих реализаций функций, которые могут

соответствовать простому или сложному типу данных, ассоциированы с параметрами устройства путем построения ссылки из параметра на соответствующий параметр интерфейса реализации функции как в одноуровневых, так и в иерархических структурах.

Элемент ApplicationProcess содержит до пяти списков объектов (см. рисунок 30):

— два списка, определяющих типы данных (опционально) и типы функций (обязательно);

— один обязательный список, определяющий реализации функций на уровне устройства (возможно включая соединения между реализациями);

— один обязательный список, определяющий параметры устройства, и

— один опциональный список, определяющий группы параметров (комбинации параметров для специальных целей).

F.4.5.2 Элемент dataTypeList (список типов данных)

F.4.5.2.1 Общие положения

Опциональный элемент dataTypeList присутствует, если внутри деклараций параметров в профиле устройства необходимы сложные типы данных типа таблиц или структур.

Элемент dataTypeList, представленный на рисунке F.5, содержит последовательность одного или нескольких элементов из следующего набора:

— элемент array (таблица);

— элемент struct (структура);

— элемент enum (нумерация) или

— производный элемент.

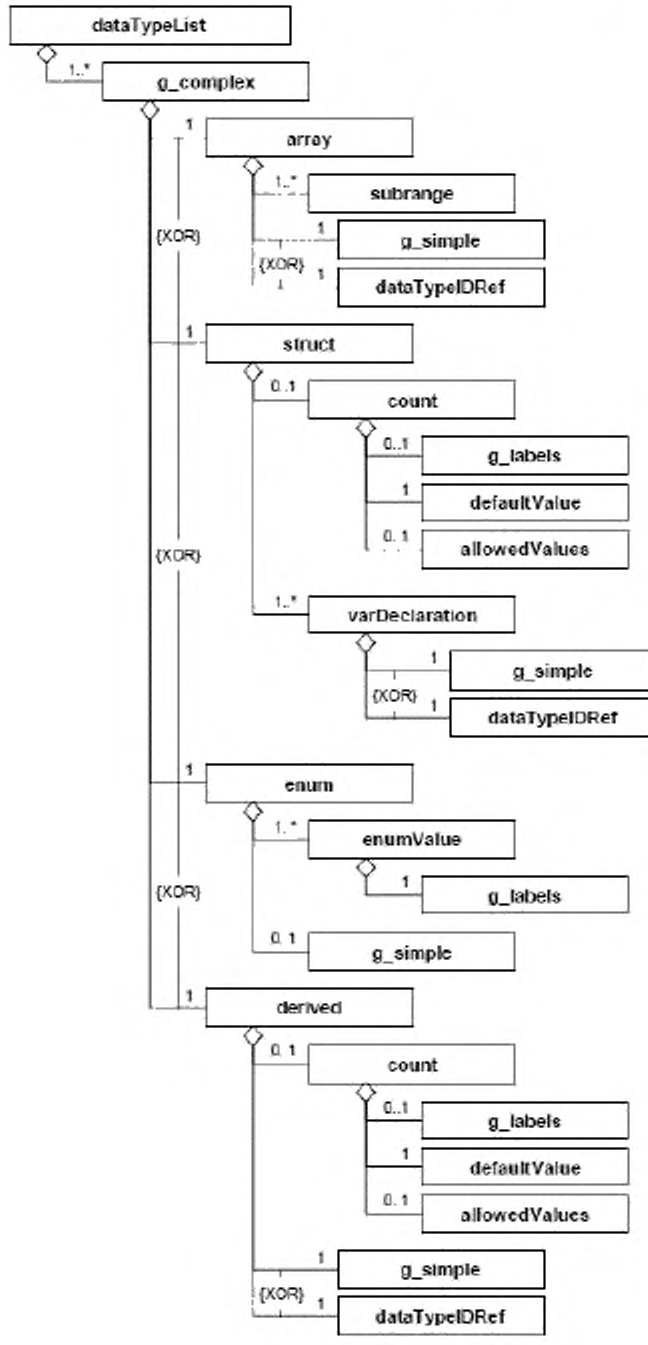


Рисунок F.5 — dataTypeList (список типов данных)

F.4.5.2.2 Общие элементы

F.4.5.2.2.1 Группа g_simple

Группа g_simple содержит набор элементов, имена которых представляют имена всех простых типов данных, допускаемых при определении параметров внутри профиля устройства. Простые типы данных соответствуют элементарным типам данных, определенным в МЭК 61131-3; к ним добавляют типы данных BITSTRING и CHAR (=STRING[1]).

Эти элементы вводят внутри группы, допуская их размещение непосредственно как субэлементов элемента таблицы (или элемента varDeclaration по F.4.5.4.3.2.

F.4.5.2.2.2 Элемент count (счет)

Элемент count определяет число используемых блоков базового типа производного типа. Многоязычные имена и/или описания для элемента count предоставляют с помощью группы g_labels. По вопросу описания группы g_labels см. F.2.2.2.

Элемент count описывают следующими способами:

— его атрибутами;

— обязательным субэлементом defaultValue (значение по умолчанию) и возможно пустым набором субэлементов g_labels и allowedValues (допустимые значения).

Число блоков выражают как значение атрибута defaultValue элемента count. Атрибут allowedValue определяет диапазон значений по умолчанию.

Субэлементы defaultValue и allowedValues описаны в F.4.5.6.2.5 и F.4.5.6.2.7.

Элемент count должен содержать атрибуты, указанные в таблице F.14.

Таблица F.14 — Атрибуты элемента count (счет)

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный идентификатор count
Access	xsd:NMToken	По умолчанию	Определяет, какие операции являются правильными для count: – чтение – доступ только для чтения (значение по умолчанию); – запись – доступ только для записи; – тениеЗапись – доступ для чтения и записи; – нетДоступа – доступ не разрешается

F.4.5.2.3 Элемент array (массив)

F.4.5.2.3.1 Общие положения

Элемент array (массив) предназначен для описания типа данных array (массив), ссылка на которые может выполняться из параметров интерфейса типа функций, из других определений типа таблицы, или из параметра компонента определения структурированного типа данных.

Элемент array содержит не менее одного элемента subrange (поддиапазон) и либо элемент, описывающий простой тип данных из группы g_simple, либо элемент dataTypeIDRef (ссылка идентификатора типа данных), имеющий ссылки на один из определенных сложных типов данных элемента dataTypeList (список типов данных).

В случае многомерных таблиц должны будут присутствовать несколько элементов subrange. В этом случае первый элемент поддиапазона в последовательности определяет поддиапазон для первого слева индекса таблицы, а последний элемент поддиапазона в последовательности определяет поддиапазон для крайнего правого индекса таблицы.

Элемент array (массив) содержит атрибуты, указанные в таблице F.15.

Таблица F.15 — Атрибуты элемента array (массив)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя типа данных (тип таблица)
uniqueID	xsd:ID	Требуемое	Уникальный ID (тип таблица)
Description	xsd:string	опциональное	Опциональное текстовое описание (тип таблица)

F.4.5.2.3.2 Элемент subrange (поддиапазон)

Элемент subrange определяет нижний и верхний предельные значения индекса таблицы для одного измерения таблицы. Этот элемент не имеет субэлементов.

Предельные значения типа xsd:long содержатся в двух атрибутах элемента поддиапазона, указанных в таблице F.16.

Таблица F.16 — Атрибуты элемента subRange (поддиапазон)

Атрибут	Тип данных	Применение	Описание
lowerLimit	xsd:long	Требуемое	Нижнее значение поддиапазона
upperLimit	xsd:long	Требуемое	Верхнее значение поддиапазона

F.4.5.2.4 Элемент struct (структура)

F.4.5.2.4.1 Общие положения

Элемент struct предназначен для описания типа структурированных данных, на который может выполняться ссылка из параметра интерфейса типа функции, из определения типа таблицы или из параметра компонента, входящего в определение другого типа структурированных данных.

Элемент struct содержит последовательность одного или более элементов varDeclaration (объявление переменной), которые определяют компоненты типа структурированных данных.

Элемент struct должен содержать атрибуты, указанные в таблице F.17.

Таблица F.17 — Атрибуты элемента struct (структура)

Атрибут	Тип данных	Применение	Описание
Nam	xsd:string	Требуемое	Имя типа данных (тип структурированных данных)
uniqueID	xsd:ID	Требуемое	Уникальный ID (тип структурированных данных)
Description	xsd:string	Опциональное	Опциональное текстовое описание (тип структурированных данных)

F.4.5.2.4.2 Элемент varDeclaration

В контексте определения типа структурированных данных элемент varDeclaration описывает единичный параметр компонента структуры.

В контексте определения интерфейса функции элемент varDeclaration описывает единичный параметр интерфейса типа функции.

Тип данных параметра компонента или параметра интерфейса определяют либо с помощью элемента, описывающего простой тип данных группы g_simple, либо с помощью элемента dataTypeIDRef (ссылка идентификатора типа данных), который выполняет ссылку на один из определенных сложных типов данных в элементе dataTypeList (список типов данных).

Остальные характеристики параметров содержатся в атрибутах элемента varDeclaration, указанных в таблице F. 18.

Таблица F.18 — Атрибуты элемента varDeclaration

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя параметра интерфейса или компонента структуры
uniqueID	xsd:ID	Требуемое	Уникальный ID параметра интерфейса или компонента структуры (см. примечание 1)

Окончание таблицы F.18

Атрибут	Тип данных	Применение	Описание
Size	xsd:string	Опциональное	Число элементов, если параметр интерфейса или компонента структуры анонимного типа ARRAY, BITSTRING, STRING или WSTRING (см. примечание 2)
InitialValue	xsd:string	Опциональное	Начальное значение параметра интерфейса или компоненты структуры (см. примечание 3)
Description	xsd:string	Опциональное	Опциональное текстовое описание параметра интерфейса или компонента структуры
Примечание 1 — При создании уникальных ID для параметров важно соблюдать уникальность всех ID файла – источника XML. Для создания возможности одинаковых имен для параметров компонентов различных структур данных и одинаковых имен для параметров интерфейса типов функций ID параметра должен обычно соединять типовое имя типа структурированных данных или типовое имя типа функций с именем параметра для гарантии уникальности. Примечание 2 — Анонимные типы определяют размер таблицы, битовой строки или строки непосредственно в декларации параметра, а не путем ссылки на тип сложных данных с именем. Для таблицы тип данных единичного элемента таблицы определяется типом данных параметра. Для битовой строки единичный элемент таблицы является одним битом.			
Примечание 3 — Данный атрибут определяет начальное (по умолчанию) значение параметра интерфейса типа функции. Он определяется заданным значением по умолчанию параметра, ассоциированного с параметром интерфейса экземпляра функции.			

F.4.5.2.5 Элемент enum (нумерация)

F.4.5.2.5.1 Общие положения

Элемент enum предназначен для описания типа пронумерованных данных, на которые могут быть ссылки из параметра интерфейса типа функции, из определения типа таблицы или из параметра компонента определения типа структурированных данных.

В соответствии с рисунком F.5 этот элемент содержит последовательность из одного или более элементов enumValue (значение нумерации), которые определяют константы нумерации типа пронумерованных данных. Тип данных констант нумерации опционально определяется с помощью элемента, описывающего тип простых данных группы g_simple.

Элемент enum содержит атрибуты, указанные в таблице F.19.

Таблица F.19 — Атрибуты элемента enum (нумерация)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Типовое имя типа пронумерованных данных
uniqueID	xsd:ID	Требуемое	Уникальный ID типа пронумерованных данных
Size	xsd:string	Опциональное	Опциональное число пронумерованных значений типа пронумерованных данных
Description	xsd:string	Опциональное	Опциональное текстовое описание типа пронумерованных данных

F.4.5.2.5.2 Элемент enumValue (значение нумерации)

Элемент enumValue определяет имя (имена) и опционально числовое значение единичной константы нумерации. Имя (имена) устанавливаются в группе g_labels, а значение содержится в атрибуте единичного значения элемента enumValue согласно в таблице F.20.

Таблица F.20 — Атрибуты элемента enumValue (значение нумерации)

Атрибут	Тип данных	Применение	Описание
Value	xsd:string	Опционально	Опциональный атрибут: фиксированное числовое значение для константы нумерации, представленное в виде строки символов

F.4.5.2.6 Элемент derived (производный)

Элемент derived предназначен для вывода нового типа данных из данного базового типа.

Элемент derived содержит опциональный элемент count (счет) и либо элемент, описывающий тип простых данных из группы g_simple, либо элемент dataTypeIDRef, который ссылается на один из определенных сложных типов данных внутри элемента dataTypeList (список типов данных).

Если элемент count (счет) пропущен, определение производного типа вводит только имя нового типа для соответствующего базового типа. Если элемент count присутствует, он определяет число блоков соответствующего базового типа, используемых для построения производного типа (например, базовый тип BITSTRING count = 4 определяет производный тип размером 4 бита).

Элемент derived содержит атрибуты, указанные в таблице F.21.

Таблица F.21 — Атрибуты элемента derived (производный)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя типа данных производного типа
uniqueID	xsd:ID	Требуемое	Уникальный ID производного типа
Description	xsd:string	Опциональное	Опциональное текстовое описание производного типа

F.4.5.3 Элемент functionTypeList (список типов функций)

Если опциональный элемент ApplicationProcess (прикладной процесс) присутствует в профиле устройства, он содержит обязательный элемент functionTypeList, представленный на рисунке F.6.

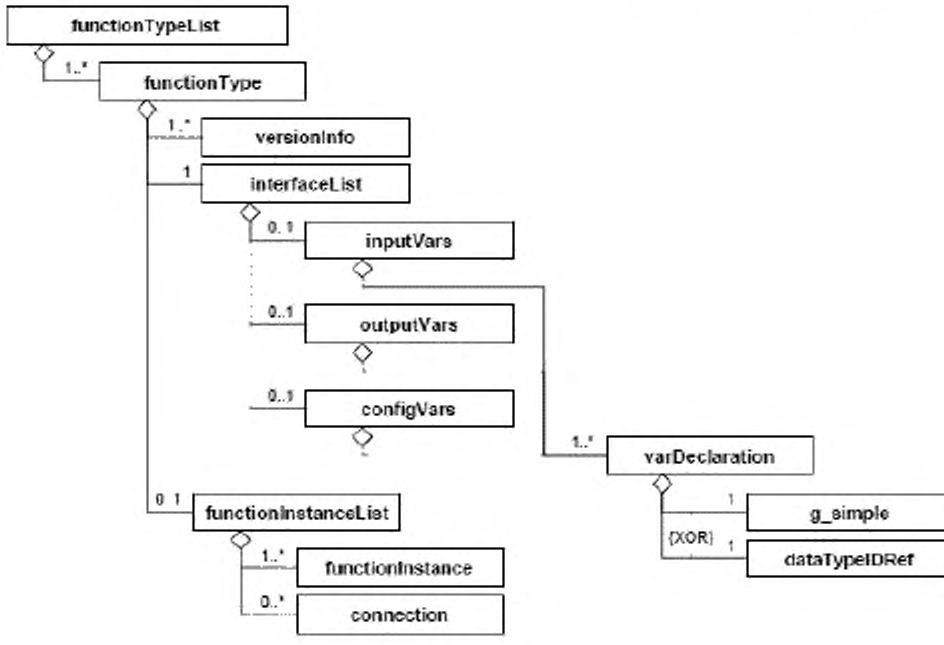


Рисунок F.6 — functionTypeList (список типов функций)

Элемент functionTypeList представляет последовательность одного или многих элементов functionType (тип функции).

Каждый из элементов functionType представляет описание типа функции устройства, на которую имеется ссылка из не менее чем одного экземпляра типа функции в элементе functionInstanceList (список экземпляров функции). Ссылки из более чем одного экземпляра того же типа функций также допускаются.

Описание типа функции содержит все объекты и данные, которые являются общими для всех экземпляров данного типа функций.

Пример 1 — Примерами являются параметры – или параметры функций – объекты, формирующие интерфейс функции (тип соответствующего экземпляр).

Пример 2 — Другими примерами являются сущности, содержащиеся внутри тела функции в иерархически структурированном функциональном описании. Эти сущности, которые расположены в элементе functionInstanceList (список экземпляров функции) типа функции, ссылаются на другие типы функции в списке типов функций.

F.4.5.4 Элемент functionType (тип функции)

F.4.5.4.1 Общие положения

Элемент functionType содержит один или несколько элементов versionInfo (информация о версии), обязательный элемент interfaceList (список интерфейсов) и опциональный элемент functionInstanceList (список экземпляров функции). Элемент functionInstanceList присутствует в элементе functionType только в том случае, если функция иерархически структурирована.

Дополнительно элемент functionType должен содержать атрибуты, указанные в таблице F.22.

Таблица F.22 — Атрибуты элемента functionType (тип функции)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Типовое имя типа функции
uniqueID	xsd:ID	Требуемое	Уникальный ID типа функции
Description	xsd:string	Опциональное	Опциональное текстовое описание типа функции
Package	xsd:string	Опциональное	Опциональная текстовая ассоциация типа функции с «пакетом» или аналогичной схемой классификации – использование этого атрибута обеспечивает валидатор профиля

F.4.5.4.2 Элемент versionInfo (информация о версии)

Обязательный элемент versionInfo элемента functionType (тип функции) предоставляет информацию по истории версий типа функций (относящуюся к определению интерфейса).

Для прослеживания истории версий элемент versionInfo может быть введен несколько раз. Множественные входы должны быть организованы в элементе functionType в следующей последовательности:

a) первый вход представляет наиболее раннюю версию;

b) второй вход представляет непосредственно предшествующую версию;

c) последний вход представляет первую выпущенную версию.

Данный элемент будет предоставлен один раз при создании описания

типа функции. Новые элементы будут добавляться только при введении модификации типа функции, которая приводит к модифицированной версии профиля устройства.

Элемент versionInfo должен содержать атрибуты, указанные в таблице F.23.

Таблица F.23 — Атрибуты элемента versionInfo (информация о версии)

Атрибут	Тип данных	Применение	Описание
Organization	xsd:string	Требуемое	Имя организации, поддерживающей тип функции
Version	xsd:string	Требуемое	Обозначение версии в истории версий; предлагаемый формат: "xx.yy" (xx,yy = 0..255)
Author	xsd:string	Требуемое	Имя лица, поддерживающего тип функции
Date	xsd:date	Требуемое	Дата данной версии
Remarks	xsd:string	Опциональное	Описательная информация, относящаяся к специальному этапу истории версий

F.4.5.4.3 Элемент interfaceList (список интерфейсов)

F.4.5.4.3.1 Общие положения

Обязательный элемент interfaceList элемента functionType (тип функции) предоставляет определение интерфейса типа функции. Элементами интерфейса являются следующие:

— входные параметры и/или

— выходные параметры, и/или

— параметры конфигурации типа функции.

Следовательно элемент interfaceList содержит последовательность трех элементов, где каждый элемент представляет списки от одной до нескольких деклараций параметров, закодированных как элементы varDeclaration:

— один опциональный элемент inputVars (входные параметры);

— один опциональный элемент outputVars (выходные параметры), и

— один опциональный элемент configVars (конфигурационные параметры).

Ни один из элементов interfaceList, inputVars, outputVars или configVars не имеет каких-либо атрибутов.

F.4.5.4.3.2 Элемент varDeclaration (объявление переменной)

В контексте определения структурированного типа данных элемент varDeclaration описывает единичный параметр компоненту (элемент) структуры.

В контексте определения интерфейса типа функции элемент varDeclaration описывает единичный параметр интерфейса типа функции.

Тип данных параметра компоненты или параметра интерфейса определяется либо элементом, описывающим простой тип данных группы g_simple либо элементом dataTypeIDRef (ссылочный идентификатор типа данных), который ссылается на один из определенных сложных типов данных элемента dataTypeList (список типов данных).

F.4.5.2.2.1 описывает группу g_simple и F.4.5.4.3.3 описывает элемент dataTypeIDRef.

Дополнительные характеристики параметра содержатся в атрибутах элемента varDeclaration, указанных в таблице F.24.

Таблица F.24 — Атрибуты элемента varDeclaration (объявление переменной)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	требуемое	Имя параметра интерфейса или компонента структуры
uniqueID	xsd:ID	Требуемое	Уникальный ID параметра интерфейса или компонента структуры
Size	xsd:string	Опциональное	Число элементов, если параметр интерфейса или компонента структуры имеют анонимный тип ARRAY, BITSTRING, STRING или WSTRING
initialValue	xsd:string	Опциональное	Начальное значение параметра интерфейса или компонента структуры
Description	xsd:string	Опциональное	Опциональное текстовое описание параметра интерфейса или компонента структуры

F.4.5.4.3.3 Элемент dataTypeIDRef (ссылка идентификатора типа данных)

Элемент dataTypeIDRef предназначен для ссылки на сложный тип данных элемента dataTypeList (список типов данных) (см. F.4.5.2) либо из параметра интерфейса типа функции, либо из определения типа таблицы, либо из параметра компоненты внутри определения структурированного типа данных.

Ссылка типа xsd:IDREF предоставляется в виде атрибута элемента dataTypeIDRef согласно в таблице F.25.

Таблица F.25 — Атрибуты элемента dataTypeIDRef (ссылка идентификатора типа данных)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного типа данных

F.4.5.5 Элемент functionInstanceList (список экземпляров функции)

F.4.5.5.1 Общие положения

Если опциональный элемент ApplicationProcess (прикладной процесс) присутствует в профиле устройства, он содержит обязательный элемент functionInstanceList, который включает в себя последовательность одного или нескольких элементов functionInstance (экземпляр функции) и от нуля до нескольких элементов connection (соединение).

На уровне процесса приложения элементы functionInstance представляют допустимые функции приложения типа устройства, не зависящие от типа или протокола сети. Элементы connection представляют соединения (если они существуют) между конкретными выходными и входными параметрами этих экземпляров функции.

Элемент functionInstanceList также появляется как опциональный субэлемент элемента functionType (см. F.4.5.4). Аналогично этому на уровне процесса приложения элемент functionInstanceList содержит последовательность одного или нескольких элементов functionInstance и от нуля до нескольких элементов connection.

Элемент functionInstanceList присутствует в элементе functionType только в том случае, если функция иерархически структурирована. В этом случае элементы functionInstance представляют внутренние функции, содержащиеся в типе функций, а элементы connection – опциональные внутренние соединения. Эти функции и их опциональные соединения могут быть реализованы совместно с реализацией содержащего их типа функций.

Элемент functionInstanceList не имеет каких-либо атрибутов.

F.4.5.5.2 Элемент functionInstance (экземпляр функции)

Обязательный элемент functionInstance не содержит субэлементов.

Элемент functionInstance должен содержать атрибуты, указанные в таблице F.26.

Таблица F.26 — Атрибуты элемента functionInstance (экземпляр функции)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя экземпляра функции
uniqueID	xsd:ID	Требуемое	Уникальный ID экземпляра функции (см. примечание)
typeIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного типа функции
Description	xsd:string	Опциональное	Опциональное текстовое описание экземпляра функции
Примечание — При создании уникального ID для экземпляра функции важно, чтобы этот ID был уникальным во всех создаваемых ID файла - источника XML. Для создания возможности применения одинаковых имен для экземпляров функции при различных типах функций ID экземпляра функции должен образовываться путем соединения имени типа функции с именем экземпляра функции, для обеспечения уникальности.			

F.4.5.5.3 Элемент connection (соединение)

Опциональный элемент connection определяет соединение между выходным параметром экземпляра функции и входным параметром другого экземпляра функции. Внутри типов функции соединение также может быть проведено между входным параметром типа функции и входным параметром содержащегося в типе экземпляра функции или между выходным параметром содержащегося в типе экземпляра функции и выходным параметром типа функции. Элемент connection может появляться от ни одного до нескольких раз.

Элемент connection содержит атрибуты, указанные в таблице F.27.

Таблица F.27 — Атрибуты элемента connection (соединение)

Атрибут	Тип данных	Применение	Описание
Source	xsd:string	Требуемое	Начальная точка соединения
Destination	xsd:string	Требуемое	Конечная точка соединения
Description	xsd:string	Опциональное	Опциональное текстовое описание соединения

Пример — Значения атрибутов source (источник) и destination (назначение) могут быть использованы для кодирования начальной и конечной точки соединения с помощью синтаксиса <function_instance_name>'.'<variable_name>; пример для значения атрибута источник: 'PowerMeasures. Frequency '(Мощность измерения. Частота). Соединения с параметрами интерфейса типа функции используют только имена параметров интерфейса.

F.4.5.6 Элемент parameterList (список параметров)

F.4.5.6.1 Общие положения

Если в профиле устройства присутствует опциональный элемент ApplicationProcess (прикладной процесс), он содержит обязательный элемент parameterList (список параметров), представленный на рисунке F.7, который обеспечивает последовательность одного или нескольких элементов parameter (параметр).

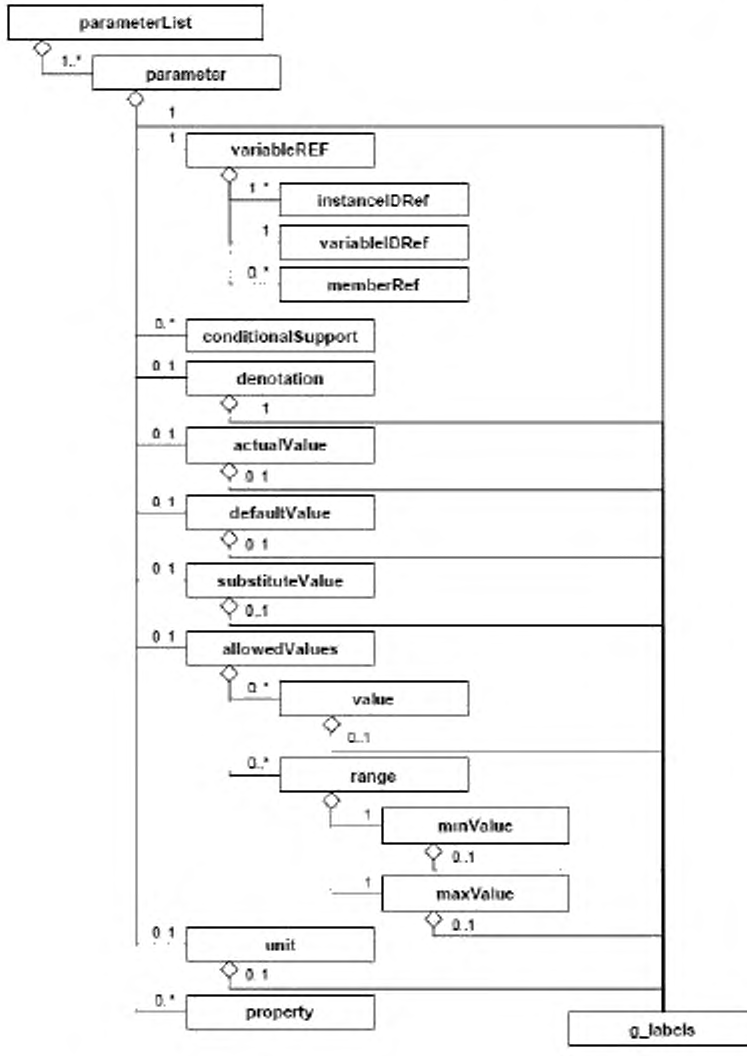


Рисунок F.7 — parameterList (список параметров)

Каждый элемент parameter представляет параметр профиля устройства. Многоязычные имена и/или описания параметров предоставляются с помощью группы g_labels. Группа g_labels описана в F.2.2.2.

Параметр описывают следующими способами:

— его атрибутами;

— путем ссылки на один (или более) параметр(ов) интерфейса одного (или более) экземпляра(ов) функции (обязательный элемент variableRef (ссылка на переменную));

— возможно, пустым набором субэлементов (conditionalSupport (условная поддержка), denotation (обозначение), actualValue (фактическое значение), defaultValue (значение по умолчанию), substituteValue (подстановочное значение), allowedValues (допустимые значения), блок, свойство и g_labels).

Примечание — Ссылки на многие параметры представляют собой специальный случай: специальные параметры могут одновременно иметь ссылку на выходной параметр одного экземпляра функции и на входной параметр другого экземпляра функции. В этом случае типы данных двух параметров должны быть одинаковыми. Программа синтаксического анализа XML не может проверять эквивалентность типов данных. Это может быть проверено только с помощью вспомогательного инструмента.

F.4.5.6.2 Элемент parameter (параметр)

F.4.5.6.2.1 Общие положения

Элемент parameter должен содержать атрибуты, указанные в таблице F.28.

Таблица F.28 — Атрибуты элемента parameter (параметр)

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный ID параметра
Access	xsd:NMToken	По умолчанию	Определяет, какая операция является правильной для параметра: — чтение - доступ только для чтения (значение по умолчанию) — запись - доступ только для записи — чтение/запись - доступ для чтения и записи — нет доступа - доступ не допускается

Окончание таблицы F.28

Атрибут	Тип данных	Применение	Описание
Support	xsd:NMToken	Опциональное	Определяет, должен или нет параметр использоваться в устройстве; правильные значения: — обязательное – использование параметра требуется — опциональное – использование параметра возможно, но не обязательно — условное – использование параметра требуется, если один или более из других необязательных параметров используются; эти параметры определяются с помощью субэлемента conditionalSupport (условная поддержка)
Persistence	xsd:boolean	По умолчанию	Определяет поведение после отказа питания; правильное значение - ложь (по умолчанию) и истина
Offset	xsd:string	Опциональное	Сдвиг, который добавляется к фактическому значению для образования масштабированной величины: ТехническоеЗначение = (ПараметрЗначение + сдвиг) * множитель; если не присутствует, принимается сдвиг = 0
Multiplier	xsd:string	Опциональное	Коэффициент масштабирования, на который умножается фактическое значение для образования масштабированной величины: ТехническоеЗначение = (ПараметрЗначение + сдвиг) * множитель; если не присутствует, принимается множитель = 1

F.4.5.6.2.2 Элемент conditionalSupport (условная поддержка)

Один или более элементов conditionalSupport присутствуют, только в том случае, если значение атрибута support (поддержка) элемента parameter (параметр) является условным (conditional). Каждый элемент имеет ссылку на

единичный опциональный параметр. Если хотя бы один из опциональных параметров используется, условный параметр также должен использоваться.

Элемент conditionalSupport должен содержать единичный атрибут, указанный в таблице F.29.

Таблица F.29 — Атрибуты элемента conditionalSupport (условная поддержка)

Атрибут	Тип данных	Применение	Описание
ParamIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного опционального параметра

F.4.5.6.2.3 Элемент denotation (обозначение)

Элемент denotation предназначен для поддержки зависящих от приложения многоязычных имен параметра. Имена предоставляются с помощью обязательного субэлемента g_labels. Допускается дополнять многоязычную описательную информацию. Элемент denotation не имеет атрибутов.

F.4.5.6.2.4 Элемент actualValue (фактическое значение)

Элемент actualValue предназначен для поддержки фактического значения параметра. Опциональный субэлемент g_labels может предоставить многоязычную описательную информацию для этой величины, которая определяется в атрибуте value (значение) элемента actualValue. Могут быть также указаны сдвиг и множитель.

Атрибуты элемента actualValue должны соответствовать указанным в таблице F.30.

Таблица F.30 — Атрибуты элемента actualValue (фактическое значение)

Атрибут	Тип данных	Применение	Описание
Value	xsd:string	Требуемое	Фактическое значение
Offset	xsd:string	Опциональное	Сдвиг, добавляемый к фактическому значению для образования масштабированной величины: ТехническоеЗначение = (значение + сдвиг) * множитель; если не присутствует, должно быть использовано соответствующее значение элемента параметра
Multiplier	xsd:string	Опциональное	Коэффициент масштабирования, на который умножается фактическое значение для образования масштабированной величины: ТехническоеЗначение = (значение + сдвиг) * множитель; если не присутствует, должно быть использовано соответствующее значение элемента параметра

F.4.5.6.2.5 Элемент defaultValue (значение по умолчанию)

Элемент defaultValue предназначен для поддержки значения по умолчанию параметра. Это значение переписывает начальное значение параметра интерфейса типа функции, ассоциированной с параметром.

Опциональный субэлемент g_labels может предоставить многоязычные имена и/или дескриптивную информацию для этой величины, предоставляемой атрибутом value (значение) элемента defaultValue. Могут быть также установлены сдвиг и множитель.

Атрибуты элемента defaultValue должны соответствовать указанным в таблице F.30.

F.4.5.6.2.6 Элемент substituteValue (подстановочное значение)

Элемент substituteValue определяет специальное значение параметра, предоставляемого для приложения устройства в некоторых рабочих состояниях устройства (например, в случае отказа устройства).

Опциональный субэлемент g_labels может предоставить многоязычные имена и/или дескриптивную информацию для этой величины, предоставляемой атрибутом value (значение) элемента substituteValue. Могут быть также установлены сдвиг и множитель.

Атрибуты элемента substituteValue должны соответствовать указанным в таблице F.30.

F.4.5.6.2.7 Элемент allowedValues (допустимые значения)

Элемент allowedValues определяет список поддерживаемых значений и/или один диапазон или несколько диапазонов поддерживаемых значений параметра.

Список поддерживаемых значений представлен в виде некоторого количества от ни одного до нескольких субэлементов value элемента allowedValues, а диапазоны также представлены в виде некоторого количества ни одного до нескольких субэлементов range (диапазон) элемента allowedValues.

Значение субэлемента value устанавливает одно допустимое значение параметра. Опциональный субэлемент g_labels может предоставить многоязычные имена и/или дескриптивную информацию для этой величины, предоставляемой атрибутом value (значение) элемента value. Могут быть также установлены сдвиг и множитель.

Атрибуты элемента value должны соответствовать указанным в таблице F.30.

Субэлемент range содержит два требуемых субэлемента, а именно элемент minValue (минимальное значение) и элемент maxValue (максимальное значение), которые определяют предельные допустимые значения диапазона. Элементы minValue и maxValue имеют такую же структуру и атрибуты, как субэлемент value элемента allowedValues.

Следовательно, описание субэлемента value и таблица F.30 действительны также для этих субэлементов.

F.4.5.6.2.8 Элемент unit (единица)

Элемент unit определяет технические единицы параметра (например время, температуру, давление, поток, ускорение, ток, энергию), согласно ИСО 1000. Опциональный элемент g_labels может предоставить многоязычные имена и/или описательную информацию для технических единиц.

Атрибуты элемента unit должны соответствовать указанным в таблице F.31.

Таблица F.31 — Атрибуты элемента unit (единица)

Атрибут	Тип данных	Применение	Описание
Multiplier	xsd:string	Требуемое	Множитель для технических единиц аналоговых параметров
unitURI	xsd:anyURI	Опциональное	Связь с соответствующим определением единицы в файле, содержащем технические единицы (например время, температуру, давление, поток, ускорение, ток, энергию) согласно ИСО 1000

F.4.5.6.2.9 Элемент variableRef (ссылка на переменную)

Элемент variableRef создает ссылку на параметр интерфейса экземпляра функции, или, если параметр представляет собой таблицу или структуру – ссылку на элемент параметра (элемент таблицы или компонент структуры).

В иерархически структурированном элементе ApplicationProcess (прикладной процесс) экземпляры функции могут быть расположены внутри экземпляров функции других типов функций. Следовательно, конкретный экземпляр на дереве функций может быть доступен только путем последовательных шагов по дереву, т. е. конкретный экземпляр должен получить адрес путем конкатенации имен экземпляров. Для отображения этой

конкатенации и создания возможности ссылки на элемент структуры variableRef (ссылка на переменную) содержит следующее:

— последовательность из одного или нескольких элементов instanceIDRef (ссылка на идентификатор экземпляра), за которой следует

— один обязательный элемент variableIDRef (ссылка на идентификатор переменной) и

— опциональный элемент memberRef (ссылка на элемент).

Элемент variableRef имеет атрибут, указанный в таблице F.32.

Таблица F.32 — Атрибут элемента variableRef (ссылка на переменную)

Атрибут	Тип данных	Применение	Описание
Position	xsd:unsignedByte	По умолчанию	Определяет последовательность нескольких отображенных объектов данных в одном объекте параметра; позиция = 1 означает начало отображения в самом низком положении бита; число битов определяется типом данных объекта данных; последовательные объекты данных упаковываются без зазоров; значение по умолчанию: 1 (см. примечание)
Примечание — В случае одного отображенного объекта данных атрибут может быть пропущен.			

F.4.5.6.2.10 Элемент instanceIDRef (ссылка на идентификатор экземпляра)

Элемент instanceIDRef предназначен для ссылки на экземпляр функции внутри элемента functionInstanceList (список экземпляров функции), который может находиться либо на уровне элемента ApplicationProcess (прикладной процесс), либо на уровне элемента functionType (тип функции).

Ссылка типа xsd:IDREF предоставлена в виде атрибута элемента instanceIDRef согласно таблице F.33.

Таблица F.33 — Атрибуты элемента instanceIDRef (ссылка на идентификатор экземпляра)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	Xsd:IDREF	Требуемое	Уникальный ID ссылочного экземпляра функции

F.4.5.6.2.11 Элемент variableIDRef (ссылка на идентификатор переменной)

Элемент variableIDRef предназначен для ссылки на параметр интерфейса типа функции внутри элемента functionTypeList (список типов функций).

В данном элементе variableRef (ссылка на переменную) экземпляр этого типа функции определяется элементом functionInstance (экземпляр функции), на который производится ссылка элементом instanceIDRef (экземплярIDCсылки), который непосредственно предшествует элементу variableIDRef.

Ссылка типа xsd:IDREF предоставляется в виде атрибута элемента variableIDRef, согласно таблице F.34.

Таблица F.34 — Атрибуты элемента variableIDRef (ссылка на идентификатор переменной)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	Xsd:IDREF	требуемое	Уникальный ID ссылочного параметра интерфейса типа функции

F.4.5.6.2.12 Элемент memberRef (ссылка на элемент)

Опциональный элемент memberRef ссылается либо на соответствующий компонент параметра интерфейса типа структурированных данных (используется атрибут uniqueIDRef (ссылка на уникальный идентификатор)), либо на соответствующий элемент таблицы параметра интерфейса типа табличных данных (используется атрибут index). Один из этих двух атрибутов должен присутствовать, если имеется элемент memberRef.

Элемент memberRef должен содержать атрибуты, указанные в таблице F.35.

Таблица F.35 — Атрибуты элемента memberRef (ссылка на элемент)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Опционально	Уникальный ID ссылочного компонента типа структурированных данных
Index	xsd:long	Опционально	Индекс ссылочного элемента таблицы

F.4.5.6.3 Элемент property (свойство)

Элемент property вводится в качестве общего элемента для создания возможности включения значений для дополнительных специальных свойств в описание параметра.

Элемент property должен содержать атрибуты, указанные в таблице F.36.

Таблица F.36 — Атрибуты элемента property (свойство)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя свойства
Value	xsd:string	Требуемое	Значение свойства

F.4.5.7 Элемент parameterGroupList (список группы параметров)

F.4.5.7.1 Общие положения

Опциональный элемент parameterGroupList содержит последовательность от одного до нескольких элементов parameterGroup (группа параметров) согласно рисунку F.8. Многоязычные имена и/или описания групп параметров предоставляются с помощью группы g_labels. Описание группы g_labels приведено в F.2.2.2.

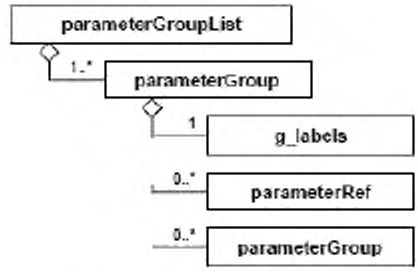


Рисунок F.8 — parameterGroupList (список группы параметров)

F.4.5.7.2 Элемент parameterGroup (группа параметров)

Каждый элемент parameterGroup объединяет набор параметров элемента parameterList (список параметров) для построения группы параметров, предназначенных для специальной цели, например, для подготовки видов HMI. Эта цель указывается значением атрибута kindOfAccess (вид доступа) элемента parameterGroup. Возможно определение иерархии групп параметров.

На соответствующие параметры в наборе производится ссылка с помощью соответствующего номера элементов parameterRef (ссылка на переменную).

Элемент parameterGroup содержит атрибуты, указанные в таблице F.37.

Таблица F.37 — Атрибуты элемента parameterGroup (группа параметров)

Атрибут	Тип данных	Применение	Описание
uniqueID	Xsd:ID	Требуемое	Уникальный ID группы параметров
kindOfAccess	Xsd:string	Опциональное	Классифицирует параметры группы параметров

F.4.5.7.3 Элемент parameterRef (ссылка на переменную)

Элемент parameterRef предназначен для ссылки на элемент parameter (параметр) элемента parameterList (список параметров), относящегося к элементу ApplicationProcess (прикладной процесс).

Ссылка типа xsd:IDREF предоставляется в качестве атрибута элемента parameterRef (ссылка на переменную) согласно таблице F.38.

Таблица F.38 — Атрибуты элемента parameterRef (ссылка на переменную)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	Xsd:IDREF	Требуемое	Уникальный ID ссылочного параметра

F.4.6 Схемы шаблона профиля устройства EtherCAT

F.4.6.1 XML Схема: ISO15745ProfileContainer.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    >
<xsd:element name="ISO15745ProfileContainer">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ISO15745Profile" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileHeader" type="ProfileHeader_DataType"/>
<xsd:element name="ProfileBody" type="ProfileBody_DataType"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* HEADER SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileHeader_DataType">
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:annotation>
<xsd:documentation>* BODY SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileBody_DataType" abstract="true"/>
<xsd:annotation>
<xsd:documentation>* HEADER DATA TYPES *</xsd:documentation>
```

```
</xsd:annotation>
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP"/>
<xsd:enumeration value="Process"/>
<xsd:enumeration value="InformationExchange"/>
<xsd:enumeration value="Resource"/>
<xsd:enumeration value="Device"/>
<xsd:enumeration value="CommunicationNetwork"/>
<xsd:enumeration value="Equipment"/>
<xsd:enumeration value="Human"/>
<xsd:enumeration value="Material"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="CSI"/>
<xsd:enumeration value="HCI"/>
<xsd:enumeration value="ISI"/>
<xsd:enumeration value="API"/>
<xsd:enumeration value="CMI"/>
<xsd:enumeration value="ESI"/>
<xsd:enumeration value="FSI"/>
<xsd:enumeration value="MTI"/>
<xsd:enumeration value="SEI"/>
<xsd:enumeration value="USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:annotation>
<xsd:documentation>* ISO 15745 DEFINED DATA TYPES *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileHandle_DataType">
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileLocation" type="xsd:anyURI" minOccurs="0"/>
```

```
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

F.4.6.2 XML Схема: CommonElements.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                                    ">
<!--################ common attribute group-->
<xsd:attributeGroup name="ag_formatAndFile">
<xsd:attribute name="formatName" type="xsd:string" fixed="DDXML" form="unqualified"/>
<xsd:attribute name="formatVersion" type="xsd:string" fixed="2.0" form="unqualified"/>
<xsd:attribute name="fileName" type="xsd:string" use="required" form="unqualified"/>
<xsd:attribute name="fileCreator" type="xsd:string" use="required" form="unqualified"/>
<xsd:attribute name="fileCreationDate" type="xsd:date" use="required"
form="unqualified"/>
<xsd:attribute name="fileCreationTime" type="xsd:time" use="optional"/>
<xsd:attribute name="fileModificationDate" type="xsd:date" use="optional"
form="unqualified"/>
<xsd:attribute name="fileModificationTime" type="xsd:time" use="optional"/>
<xsd:attribute name="fileModifiedBy" type="xsd:string" use="optional"/>
<xsd:attribute name="fileVersion" type="xsd:string" use="required" form="unqualified"/>
</xsd:attributeGroup>
<!--################ common groups-->
<xsd:group name="g_labels">
<xsd:sequence>
<xsd:choice maxOccurs="unbounded">
<xsd:element name="label">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="lang" type="xsd:language" use="required"/>
<xsd:attribute name="URI" type="xsd:anyURI" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="description">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="lang" type="xsd:language" use="required"/>
<xsd:attribute name="URI" type="xsd:anyURI" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="labelRef">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:anyURI">
<xsd:attribute name="dictID" type="xsd:IDREF" use="required"/>
```

```
<xsd:attribute name="textID" type="xsd:string" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
</xsd:choice>
</xsd:sequence>
</xsd:group>
<xsd:group name="g_simple">
<xsd:choice>
<xsd:element name="BOOL"/>
<xsd:element name="BITSTRING"/>
<xsd:element name="BYTE"/>
<xsd:element name="CHAR"/>
<xsd:element name="WORD"/>
<xsd:element name="DWORD"/>
<xsd:element name="LWORD"/>
<xsd:element name="SINT"/>
<xsd:element name="INT"/>
<xsd:element name="DINT"/>
<xsd:element name="LINT"/>
<xsd:element name="USINT"/>
<xsd:element name="UINT"/>
<xsd:element name="UDINT"/>
<xsd:element name="ULINT"/>
<xsd:element name="REAL"/>
<xsd:element name="LREAL"/>
<xsd:element name="TIME"/>
<xsd:element name="DATE"/>
<xsd:element name="DT"/>
<xsd:element name="TOD"/>
<xsd:element name="STRING"/>
<xsd:element name="WSTRING"/>
</xsd:choice>
</xsd:group>
<!--################ common elements-->
<xsd:element name="vendorID">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="deviceFamily">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="productID">
```

```
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="version">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="versionType" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="SW"/>
<xsd:enumeration value="FW"/>
<xsd:enumeration value="HW"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="buildDate" type="xsd:date"/>
<xsd:element name="specificationRevision">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

F.4.6.3 XML Схема: ProfileBody_Device_EtherCAT.xsd

Схема XML ProfileBody_Device_EtherCAT.xsd включает схему ISO15745ProfileContainer.xsd в F.4.6.1 и схему CommonElements.xsd в F.4.6.2.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    ">
<xsd:include schemaLocation="ISO15745ProfileContainer.xsd"/>
<xsd:include schemaLocation="CommonElements.xsd"/>
<!--################ profile body device -->
<xsd:complexType name="ProfileBody_Device_EtherCAT">
<xsd:complexContent>
<xsd:extension base="ProfileBody_DataType">
<xsd:sequence>
<xsd:element ref="DeviceIdentity" minOccurs="0"/>
```

```
<xsd:element ref="DeviceManager" minOccurs="0"/>
<xsd:element ref="DeviceFunction" maxOccurs="unbounded"/>
<xsd:element ref="ApplicationProcess" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element name="ExternalProfileHandle" type="ProfileHandle_DataType"
minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attributeGroup ref="ag_formatAndFile"/>
<xsd:attribute name="supportedLanguages" type="xsd:NMTOKENS" use="optional"/>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<!--################ device identity elements -->
<xsd:element name="DeviceIdentity">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="vendorName"/>
<xsd:element ref="vendorID" minOccurs="0"/>
<xsd:element ref="vendorText" minOccurs="0"/>
<xsd:element ref="deviceFamily" minOccurs="0"/>
<xsd:element ref="productFamily" minOccurs="0"/>
<xsd:element ref="productName"/>
<xsd:element ref="productID" minOccurs="0"/>
<xsd:element ref="productText" minOccurs="0"/>
<xsd:element ref="orderNumber" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="version" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="buildDate" minOccurs="0"/>
<xsd:element ref="specificationRevision" minOccurs="0"/>
<xsd:element ref="instanceName" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="productFamily">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="instanceName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="false"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="orderNumber">
<xsd:complexType>
<xsd:simpleContent>
```

```
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="productName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="productText">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="vendorName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="vendorText">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:complexType>
</xsd:element>
<!--################ device manager elements -->
<xsd:element name="DeviceManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="indicatorList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="indicatorList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="LEDList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="LEDList">
```

```
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="LED" maxOccurs="unbounded"/>
<xsd:element ref="combinedState" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="LED">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element ref="LEDstate" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="LEDcolors" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="monocolor"/>
<xsd:enumeration value="bicolor"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="LEDtype" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="IO"/>
<xsd:enumeration value="device"/>
<xsd:enumeration value="communication"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="LEDstate">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="state" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="on"/>
<xsd:enumeration value="off"/>
<xsd:enumeration value="flashing"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="LEDcolor" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="green"/>
<xsd:enumeration value="amber"/>
<xsd:enumeration value="red"/>
</xsd:restriction>
```

```
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="flashingPeriod" type="xsd:unsignedInt" use="optional"/>
<xsd:attribute name="impulsWidth" type="xsd:unsignedByte" default="50"/>
<xsd:attribute name="numberOfImpulses" type="xsd:unsignedByte" default="1"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="combinedState">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element name="LEDstateRef" minOccurs="2" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="stateIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--################ device function elements -->
<xsd:element name="DeviceFunction">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="capabilities"/>
<xsd:element ref="picturesList" minOccurs="0"/>
<xsd:element ref="dictionaryList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="capabilities">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="characteristicsList" maxOccurs="unbounded"/>
<xsd:element ref="standardComplianceList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristicsList">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="category" minOccurs="0">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element ref="characteristic" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristic">
<xsd:complexType>
<xsd:sequence>
```

```
<xsd:element ref="characteristicName"/>
<xsd:element ref="characteristicContent" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristicContent">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristicName">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="standardComplianceList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="compliantWith" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="compliantWith">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="range" default="international">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="international"/>
<xsd:enumeration value="internal"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="picturesList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="picture" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="picture">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="URI" type="xsd:anyURI" use="required"/>
<xsd:attribute name="number" type="xsd:unsignedInt" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="dictionaryList">
<xsd:complexType>
```

```
<xsd:sequence>
<xsd:element ref="dictionary" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dictionary">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="file" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="lang" type="xsd:language" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="file">
<xsd:complexType>
<xsd:attribute name="URI" type="xsd:anyURI" use="required"/>
</xsd:complexType>
</xsd:element>
<!--################ application process elements -->
<xsd:element name="ApplicationProcess">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="dataTypeList" minOccurs="0"/>
<xsd:element ref="functionTypeList"/>
<xsd:element ref="functionInstanceList"/>
<xsd:element ref="parameterList"/>
<xsd:element ref="parameterGroupList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dataTypeList">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_complex" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionTypeList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="functionType" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionType">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="versionInfo" maxOccurs="unbounded"/>
<xsd:element ref="interfaceList"/>
<xsd:element ref="functionInstanceList" minOccurs="0"/>
</xsd:sequence>
```

```
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
<xsd:attribute name="package" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="versionInfo">
<xsd:complexType>
<xsd:attribute name="organization" type="xsd:string" use="required"/>
<xsd:attribute name="version" type="xsd:string" use="required"/>
<xsd:attribute name="author" type="xsd:string" use="required"/>
<xsd:attribute name="date" type="xsd:date" use="required"/>
<xsd:attribute name="remarks" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="interfaceList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="inputVars" minOccurs="0"/>
<xsd:element ref="outputVars" minOccurs="0"/>
<xsd:element ref="configVars" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="inputVars">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="outputVars">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="configVars">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="varDeclaration">
<xsd:complexType>
<xsd:choice>
<xsd:group ref="g_simple"/>
<xsd:element ref="dataTypeIDRef"/>
</xsd:choice>
<xsd:attribute name="name" type="xsd:string" use="required"/>
```

```
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="size" type="xsd:string" use="optional"/>
<xsd:attribute name="initialValue" type="xsd:string" use="optional"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="dataTypeIDRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionInstanceList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="functionInstance" maxOccurs="unbounded"/>
<xsd:element ref="connection" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionInstance">
<xsd:complexType>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="typeIDRef" type="xsd:IDREF" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="connection">
<xsd:complexType>
<xsd:attribute name="source" type="xsd:string" use="required"/>
<xsd:attribute name="destination" type="xsd:string" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="parameter" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameter">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element ref="variableRef" maxOccurs="unbounded"/>
<xsd:element ref="conditionalSupport" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="denotation" minOccurs="0"/>
<xsd:element ref="actualValue" minOccurs="0"/>
<xsd:element ref="defaultValue" minOccurs="0"/>
<xsd:element ref="substituteValue" minOccurs="0"/>
<xsd:element ref="allowedValues" minOccurs="0"/>
```

```
<xsd:element ref="unit" minOccurs="0"/>
<xsd:element ref="property" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attributeGroup ref="ag_parameter"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="variableRef">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="instanceIDRef" maxOccurs="unbounded"/>
<xsd:element ref="variableIDRef"/>
<xsd:element ref="memberRef" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="position" type="xsd:unsignedByte" default="1"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="instanceIDRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="variableIDRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="memberRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="optional"/>
<xsd:attribute name="index" type="xsd:long" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="actualValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="conditionalSupport">
<xsd:complexType>
<xsd:attribute name="paramIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="denotation">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="defaultValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="substituteValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="allowedValues">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="value" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="range" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="value">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="range">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="minValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="maxValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="unit">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="multiplier" type="xsd:string" use="required"/>
<xsd:attribute name="unitURI" type="xsd:anyURI" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="property">
<xsd:complexType>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="value" type="xsd:string" use="required"/>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="parameterGroupList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="parameterGroup" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterGroup">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element ref="parameterGroup" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="parameterRef" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="kindOfAccess" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<!--################ complex types -->
<xsd:element name="array">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="subrange" maxOccurs="unbounded"/>
<xsd:choice>
<xsd:group ref="g_simple"/>
<xsd:element ref="dataTypeIDRef"/>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="subrange">
<xsd:complexType>
<xsd:attribute name="lowerLimit" type="xsd:long" use="required"/>
<xsd:attribute name="upperLimit" type="xsd:long" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="struct">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="count" minOccurs="0"/>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
```

```
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="enum">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="enumValue" maxOccurs="unbounded"/>
<xsd:group ref="g_simple" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="size" type="xsd:string" use="optional"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="enumValue">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="value" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="derived">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="count" minOccurs="0"/>
<xsd:choice>
<xsd:group ref="g_simple"/>
<xsd:element ref="dataTypeIDRef"/>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="count">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:element ref="defaultValue"/>
<xsd:element ref="allowedValues" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="access" default="read">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="read"/>
<xsd:enumeration value="write"/>
<xsd:enumeration value="readWrite"/>
<xsd:enumeration value="noAccess"/>
</xsd:restriction>
```

```
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<!--################ group-->
<xsd:group name="g_complex">
<xsd:choice>
<xsd:element ref="array"/>
<xsd:element ref="struct"/>
<xsd:element ref="enum"/>
<xsd:element ref="derived"/>
</xsd:choice>
</xsd:group>
<!--################ attribute groups-->
<xsd:attributeGroup name="ag_parameter">
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="access" default="read">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="read"/>
<xsd:enumeration value="write"/>
<xsd:enumeration value="readWrite"/>
<xsd:enumeration value="noAccess"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="support" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="mandatory"/>
<xsd:enumeration value="optional"/>
<xsd:enumeration value="conditional"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="persistence" type="xsd:boolean" default="false"/>
<xsd:attribute name="offset" type="xsd:string" use="optional"/>
<xsd:attribute name="multiplier" type="xsd:string" use="optional"/>
</xsd:attributeGroup>
<xsd:attributeGroup name="ag_value">
<xsd:attribute name="value" type="xsd:string" use="required"/>
<xsd:attribute name="offset" type="xsd:string" use="optional"/>
<xsd:attribute name="multiplier" type="xsd:string" use="optional"/>
</xsd:attributeGroup>
</xsd:schema>
```

F.5 Описание шаблона профиля коммуникационной сети

F.5.1 ProfileBody (тело профиля)

Профиль коммуникационной сети ProfileBody содержит ApplicationLayers (уровни приложения), TransportLayers (транспортные уровни) и элементы NetworkManagement (сетевой менеджмент), показанные на рисунке 31.

Элемент ProfileBody содержит атрибуты, указанные в таблице F.39.

Таблица F.39 — Атрибуты элемента ProfileBody (тело профиля)

Атрибут	Тип данных	Применение	Описание
formatName	xsd:string	Фиксированное	Идентификатор формата
formatVersion	xsd:string	Фиксированное	Идентификатор версии формата
filename	xsd:string	Требуемое	Имя файла с расширением без пути
fileCreator	xsd:string	Требуемое	Лицо, разработавшее файл
fileCreationDate	xsd:date	Требуемое	Дата создания файла
fileCreationTime	xsd:time	Опциональное	Время создания файла
fileModifiedBy	xsd:string	Опциональное	Лицо, изменившее файл
fileModificationDate	xsd.date	Опциональное	Дата последнего изменения файла
fileModificationTime	xsd:time	Опциональное	Время последнего изменения файла
fileVersion	xsd:string	Требуемое	Установленная поставщиком версия файла
supportedLanguages	xsd:NMTOKENS	Опциональное	Список поддерживаемых языков

F.5.2 ApplicationLayers (уровни приложения)

F.5.2.1 Общие положения

На рисунке 31 представлена структура класса CANopenObjectList для EtherCAT ApplicationLayers.

F.5.2.1.1 Общие положения

На рисунке 31 показана структура элемента CANopenObjectList. Этот элемент содержит от одного до нескольких элементов CANopenObject.

Примечание — В EtherCAT принята терминология и структура объекта CANopen по EN 50325-4 (CANopen поверх EtherCAT) и используются термины CANopen.

F.5.2.1.2 CANopenObject

F.5.2.1.2.1 Общие положения

Элемент CANopenObject содержит от ни одного до нескольких элементов CANopenSubObject. Элементы CANopenObject и CANopenSubObject отображают функциональную часть профиля устройства EtherCAT в профиле коммуникационной сети CANopen поверх EtherCAT.

Элемент CANopenObject содержит атрибуты, указанные в таблице F.40.

Таблица F.40 — Атрибуты элемента CANopenObject

Атрибут	Тип данных	Применение	Описание
Index	xsd:hexBinary	Требуемое	Индекс объекта (четыре hex[6] разряда)
Name	xsd:string	Требуемое	Имя объекта
ObjectType	xsd:unsignedByte	Требуемое	Тип объекта CANopen
DataType	xsd:hexBinary	Опциональное	Тип данных CANopen (два hex[6] разряда)
LowLimit	xsd:string	Опциональное	Нижнее предельное значение параметра
HighLimit	xsd:string	Опциональное	Верхнее предельное значение параметра
AccessType	xsd:string	Опциональное	Тип доступа объекта; правильные значения: - ro – доступ только для чтения - wo - доступ только для записи - rw – доступ для записи и чтения - rwr - доступ для записи и чтения, при этом предпочтительным является доступ для чтения; - rww - доступ для записи и чтения, при этом предпочтительным является доступ для записи - const - доступ только для чтения; значения не изменяются

[6] Приставка hex означает шестнадацитиричную систему счисления

Окончание таблицы F.40

Атрибут	Тип данных	Применение	Описание
DefaultValue	xsd:string	Опциональное	Значение объекта по умолчанию
PdoMapping	xsd:boolean	Опциональное	PDO отображение объекта; правильные значения: - истина – отображено - ложь – не отображено
ObjFlags	xsd:hexBinary	Опциональное	Контроль поведения инструментов (четыре hex разряда)
UniqueIDRef	xsd:IDREF	Опциональное	Уникальный ID соответствующего элемента в части процесса приложения, на который имеется ссылка из этого объекта.
SubNumber	xsd:unsignedByte	Опциональное	Число субобъектов объекта

F.5.2.1.2.2 CANopenSubObject

Элемент CANopenSubObject имеет пустой контент.

Элемент CANopenSubObject содержит атрибуты, указанные в таблице F.41.

Таблица F.41 — Атрибуты элемента CANopenSubObject

Атрибут	Тип данных	Применение	Описание
Index	xsd:hexBinary	Требуемое	Индекс объекта (четыре hex разряда)
Name	xsd:string	Требуемое	Имя объекта
ObjectType DataType	xsd:unsignedByte xsd:hexBinary	Требуемое Опциональное	Тип объекта CANopen Тип данных CANopen (два hex разряда)
LowLimit	xsd:string	Опциональное	Нижнее предельное значение параметра
HighLimit	xsd:string	Опциональное	Верхнее предельное значение параметра

Продолжение таблицы F.41

Атрибут	Тип данных	Применение	Описание
AccessType	xsd:string	Опциональное	Тип доступа объекта, действительные значения: - ro – доступ только для чтения - wo - доступ только для записи - rw – доступ для записи и чтения - rwr - доступ для записи и чтения, при этом предпочтительным является доступ для чтения; - rww - доступ для записи и чтения, при этом предпочтительным является доступ для записи; - const – доступ только для чтения; значения не изменяются
DefaultValue	xsd:string	Опциональное	Значение объекта по умолчанию
PdoMapping	xsd:boolean	Опциональное	PDO отображение объекта; правильные значения: - истина – отображено - ложь – не отображено
ObjFlags	xsd:hexBinary	Опциональное	Контроль поведения инструментов(четыре hex разряда)
UniqueIDRef	xsd:IDREF	Опциональное	Уникальный ID соответствующего элемента процесса приложения, на который имеется ссылка из этого объекта. Если атрибуту присвоены атрибуты dataType, lowLimit, highLimit, accessType, то defaultValue определяется ссылочным элементом процесса приложения

F.5.2.2 Элемент identity (идентичность)

Поскольку различные профили коммуникаций могут требовать различную информацию об идентичности, может быть использован опциональный элемент локальной идентичности внутри элемента ApplicationLayers (уровни приложения). Элемент identity может содержать подмножество субэлементов элемента DeviceIdentity (идентичность устройства), описанного в F.4.2. Все описания субэлементов, приведенные в F.4.2, применяются также к субэлементам данного элемента identity.

F.5.2.3 Элемент dummyUsage (использование макета)

F.5.2.3.1 Общие положения

На рисунке 31 представлена структура элемента dummyUsage, содержащего от одного до нескольких элементов dummy (макет).

F.5.2.3.2 Элемент dummy (макет)

Элемент dummy не имеет контента. Этот элемент используется для обеспечения возможности включения и выключения некоторых входов макета для отображения макета.

Элемент dummy содержит атрибуты, указанные в таблице F.42.

Таблица F.42 — Атрибуты элемента dummy (макет)

Атрибут	Тип данных	Применение	Описание
Entry	xsd:string	Требуемое	Строка строится с использованием имени объекта макет, за которым следует знак равенства, а затем значение либо 0 для отключения отображения, либо 1 для включения отображения. Допустимыми значениями являются следующие: - Dummy0001=0; - Dummy 0002=0; - Dummy 0003=0; - Dummy 0004=0; - Dummy 0005=0; - Dummy 0006=0; - Dummy 0007=0; - Dummy 0001 = 1; - Dummy 0002=1; - Dummy 0003=1; - Dummy 0004=1; - Dummy 0005=1; - Dummy 0006=1; - Dummy 0007=1

F.5.2.4 Элемент dynamicChannels (динамические каналы)

F.5.2.4.1 Общие положения

На рисунке 31 представлена структура элемента dynamicChannels, содержащего один или несколько элементов dynamicChannel (динамический канал).

F.5.2.4.2 Элемент dynamicChannel (динамический канал)

Элемент dynamicChannel содержит элемент, описывающий простой тип данных группы g_simple. Этот элемент используется для маркировки доступных каналов, которые могут быть использованы для создания связи

между передаваемыми по сети EtherCAT данными и программой приложения, выполняемой в устройстве.

Элемент dynamicChannel содержит атрибуты, указанные в таблице F.43.

Таблица F.43 — Атрибуты элемента dynamicChannel (динамический канал)

Атрибут	Тип данных	Применение	Описание
AccessType	xsd:NMToken	Требуемое	Тип доступа объекта; правильные значения: - только чтение – доступ только для чтения - только запись - доступ только для записи - чтение/запись/запись – доступ для чтения и записи, при этом предпочтительным являеися доступ для записи
StartIndex	xsd:hexBinary	Требуемое	Начальный индекс объекта
EndIndex	xsd:hexBinary	Требуемое	Конечный индекс объекта
MaxNumber	xsd:unsignedInt	Требуемое	Максимальное число связей с программой приложения
AddressOffset	xsd:hexBinary	Требуемое	Сдвиг адреса в памяти программы приложения
BitAlignment	xsd:unsignedByte	Опциональное	Выравнивание битов данных внутри объекта, счет которых начинается с младшего бита.
ManufacturerSpecific	xsd:string	Опциональное	Предоставляется для специального использования изготовителем

F.5.3 TransportLayers (транспортные уровни)

Элемент TransportLayers не имеет контента.

F.5.4 NetworkManagement (сетевой менеджмент)

F.5.4.1 Общие положения

На рисунке 31 представлена структура класса EtherCAT NetworkManagement (Менеджмент сети).

F.5.4.2 Элемент generalFeatures (общие характеристики)

Элемент generalFeatures имеет пустой контент.

Элемент generalFeatures содержит атрибуты, указанные в таблице Table F.44

Таблица F.44 — Атрибуты элемента generalFeatures (общие характеристики)

Атрибут	Тип данных	Применение	Описание
DynamicChannels	xsd:unsignedByte	Требуемое	Устанавливает поддержку генерации параметра динамической сети. Значение 0 означает, что динамические каналы не поддерживаются. Любое отличающееся от 0 значение означает, что поддерживается соответствующее число каналов
Granularity	xsd:unsignedByte	Требуемое	Устанавливает уровень модульности, поддерживаемый типом устройства. Значение не должно быть равно от 0 и определяет битовый размер наименьшего блока данных, который поддерживается устройством
FMMUChannels	xsd:byte	Требуемое	Число поддерживаемых каналов FMMU
FMMUBitOperationsNot Supported	xsd:Boolean	Требуемое	FMMU поддерживает битовую операцию – ложь – битовая операция поддерживается – истина - битовая операция не поддерживается
SyncManagerChannels	xsd:byte	Требуемое	Число поддерживаеимых Sync менеджер каналов

Окончание таблицы F.44

Атрибут	Тип данных	Применение	Описание
RAMSize	xsd:byte	Требуемое	Размер RAM, доступный для приложения, выражаемый в кбайтах (1 - 64)

F.5.4.3 Элемент deviceCommissioning (ввод в эксплуатацию устройства)

Элемент deviceCommissioning имеет пустой контент.

Элемент deviceCommissioning содержит атрибуты, указанные в таблице F.45.

Таблица F.45 — Атрибуты элемента deviceCommissioning (ввод в эксплуатацию устройства)

Атрибут	Тип данных	Применение	Описание
NodeID	xsd:unsignedByte	Требуемое	Уникальный ID устройства.
NodeName	xsd:string	Требуемое	Имя устройства.
NetworkNumber	xsd:unsignedLong	Требуемое	Уникальный номер сегмента сети, к которому присоединено устройство.
NetworkName	xsd:string	Требуемое	Имя сегмента сети, к которому присоединено устройство.

F.5.5 Схема шаблона профиля коммуникационной сети EtherCAT

XML Схема ProfileBody_CommunicationNetwork_EtherCAT.xsd включает в себя схему ISO15745ProfileContainer.xsd, приведенную в F.4.6.1 и схему CommonElements.xsd, приведенную в F.4.6.2.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="                                    gost.e-ecolog.ru                                    >
<xsd:include schemaLocation="ISO15745ProfileContainer.xsd"/>
<xsd:include schemaLocation="CommonElements.xsd"/>
<!-- profile body -->
<xsd:complexType name="ProfileBody_CommunicationNetwork_EtherCAT">
<xsd:complexContent>
<xsd:extension base="ProfileBody_DataType">
<xsd:choice>
<xsd:sequence>
<xsd:element name="ApplicationLayers">
<xsd:complexType>
<xsd:sequence>
```

```
<xsd:element ref="CANopenObjectList"/>
<xsd:element name="identity" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="vendorID" minOccurs="0"/>
<xsd:element ref="deviceFamily" minOccurs="0"/>
<xsd:element ref="productID" minOccurs="0"/>
<xsd:element ref="version" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="buildDate" minOccurs="0"/>
<xsd:element ref="specificationRevision" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dummyUsage" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="dummy" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="entry" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="Dummy0001=0"/>
<xsd:enumeration value="Dummy0002=0"/>
<xsd:enumeration value="Dummy0003=0"/>
<xsd:enumeration value="Dummy0004=0"/>
<xsd:enumeration value="Dummy0005=0"/>
<xsd:enumeration value="Dummy0006=0"/>
<xsd:enumeration value="Dummy0007=0"/>
<xsd:enumeration value="Dummy0001=1"/>
<xsd:enumeration value="Dummy0002=1"/>
<xsd:enumeration value="Dummy0003=1"/>
<xsd:enumeration value="Dummy0004=1"/>
<xsd:enumeration value="Dummy0005=1"/>
<xsd:enumeration value="Dummy0006=1"/>
<xsd:enumeration value="Dummy0007=1"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dynamicChannels" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="dynamicChannel" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_simple"/>
</xsd:sequence>
<xsd:attribute name="accessType" use="required">
```

```
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="readOnly"/>
<xsd:enumeration value="writeOnly"/>
<xsd:enumeration value="readWriteWrite"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="startIndex" type="xsd:hexBinary" use="required"/>
<xsd:attribute name="endIndex" type="xsd:hexBinary" use="required"/>
<xsd:attribute name="maxNumber" type="xsd:unsignedInt" use="required"/>
<xsd:attribute name="addressOffset" type="xsd:hexBinary" use="required"/>
<xsd:attribute name="bitAlignment"
type="xsd:unsignedByte" use="optional"/>
<xsd:attribute name="manufacturerSpecific"
type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TransportLayers">
<xsd:complexType/>
</xsd:element>
<xsd:element name="NetworkManagement" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="generalFeatures">
<xsd:complexType>
<xsd:attribute name="dynamicChannels" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="granularity" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="FMMUChannels" type="xsd:byte" use="required"/>
<xsd:attribute name="FMMUBitOperationNotSupported" type="xsd:boolean"
use="required"/>
<xsd:attribute name="SyncManagerChannels" type="xsd:byte" use="required"/>
<xsd:attribute name="RAMSize" type="xsd:byte" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="deviceCommissioning" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="nodeID" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="nodeName" type="xsd:string" use="required"/>
<xsd:attribute name="networkNumber" type="xsd:unsignedLong" use="required"/>
<xsd:attribute name="networkName" type="xsd:string" use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
```

```
</xsd:sequence>
<xsd:element name="ExternalProfileHandle" type="ProfileHandle_DataType"/>
</xsd:choice>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<!-- CANopen object dictionary-->
<xsd:element name="CANopenObjectList">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="CANopenObject" maxOccurs="65535">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="CANopenSubObject" minOccurs="0" maxOccurs="255">
<xsd:complexType>
<xsd:attribute name="subIndex" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="objectType" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="dataType" type="xsd:hexBinary" use="optional"/>
<xsd:attribute name="lowLimit" type="xsd:string" use="optional"/>
<xsd:attribute name="highLimit" type="xsd:string" use="optional"/>
<xsd:attribute name="accessType" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="ro"/>
<xsd:enumeration value="wo"/>
<xsd:enumeration value="rw"/>
<xsd:enumeration value="rwr"/>
<xsd:enumeration value="rww"/>
<xsd:enumeration value="const"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="defaultValue" type="xsd:string" use="optional"/>
<xsd:attribute name="actualValue" type="xsd:string" use="optional"/>
<xsd:attribute name="denotation" type="xsd:string" use="optional"/>
<xsd:attribute name="PDOmapping" type="xsd:boolean" use="optional"/>
<xsd:attribute name="objFlags" type="xsd:unsignedInt" use="optional"/>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="index" type="xsd:hexBinary" use="required"/>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="objectType" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="dataType" type="xsd:hexBinary" use="optional"/>
<xsd:attribute name="lowLimit" type="xsd:string" use="optional"/>
<xsd:attribute name="highLimit" type="xsd:string" use="optional"/>
<xsd:attribute name="accessType" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="ro"/>
```

```
<xsd:enumeration value="wo"/>
<xsd:enumeration value="rw"/>
<xsd:enumeration value="rwr"/>
<xsd:enumeration value="rww"/>
<xsd:enumeration value="const"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="defaultValue" type="xsd:string" use="optional"/>
<xsd:attribute name="actualValue" type="xsd:string" use="optional"/>
<xsd:attribute name="denotation" type="xsd:string" use="optional"/>
<xsd:attribute name="PDOmapping" type="xsd:boolean" use="optional"/>
<xsd:attribute name="objFlags" type="xsd:hexBinary" use="optional"/>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="optional"/>
<xsd:attribute name="subNumber" type="xsd:unsignedByte" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

Приложение G

(обязательное)

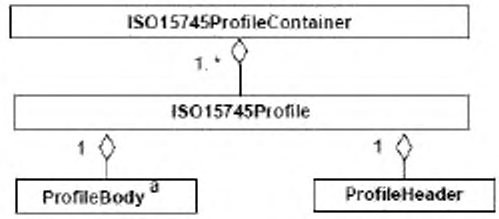
Шаблоны профиля ETHERNET Powerlink

G.1 Обзор

Ethernet Powerlink представляет собой основанную на Ethernet систему коммуникаций, описанную в МЭК/ПАС 62030.

В Ethernet Powerlink используется концепция мультипрофильного контейнера, установленная в изменении 1 к ИСО 15745-4:2003 для XML файлов профиля. Следовательно, шаблоны профиля Ethernet Powerlink основаны на переменном мастер шаблоне профиля ISO15745ProfileContainer (профильный контейнер по ИСО 15745), описанном в изменении 1 к ИСО 15745-1.

На рисунке G.1 представлена структура профиля XML Ethernet Powerlink.



[a] Использованы два типа ProfileBody (тело профиля): ProfileBody_Device_EPL или ProfileBody_CommunicationNetwork_EPL

Рисунок G.1 — Шаблон профиля Ethernet Powerlink

Имя ProfileTechnology (профильная технология) - EPL.

G.2 Общие правила

G.2.1 Использование уникальных идентификаторов (ID)

Элемент может иметь атрибут uniqueID (уникальный идентификатор) типа xsd:ID. Этот идентификатор должен быть уникальным во всем файле XML. Элемент, ссылающийся на уникальный идентификатор, содержит атрибут с именем типа xsd:IDREF.

Уникальные идентификаторы могут быть созданы двумя способами. Одна возможность заключается в построении строки из имени элемента и

вычисляемого номера. Второй способ заключается в конкатенации строк родительских элементов. Оба метода гарантируют уникальность строки.

G.2.2 Поддержка языка

G.2.2.1 Общие положения

Профили устройства, соответствующие схеме XML, описанной в данном приложении, должны поддерживать различные языки, так как в этом случае инструменты будут способны использовать имена из файла XML для их отображения на дисплее интерфейса пользователя. Например в средствах интерфейса инструмента пользователя могут быть представлены параметры коммуникации.

Поддержка языка осуществляется с помощью группы меток g_labels. Каждое имя элемента, которое может быть отображено на дисплее, и, следовательно, является зависимым от языка, содержится внутри схемы как элемент g_labels. В качестве опции к элементу метки может быть добавлен в виде атрибута URI.

Пример

Для данного имени параметра:

- ***немецкий: Baudrate;***
- ***английский: Baud rate;***
- ***французский: Vitesse de transmission.***

G.2.2.2 Элемент g_labels

Группа g_labels (метки) поддерживает введение метки (имени) и описание в контексте родительского элемента (см. рисунок G.2).

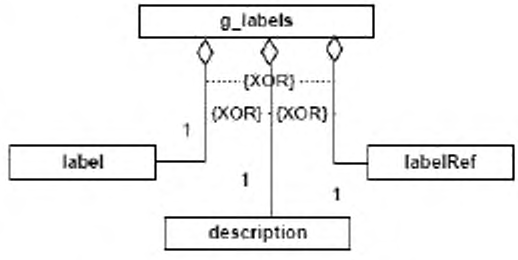


Рисунок G.2 — Группа g_labels

Каждый элемент, для которого необходимо имя или описание, должен содержать только один из элементов, предназначенных для выполнения этой задачи: метку, описание или элемент labelRef (ссылочная метка).

1 Элемент label (метка) позволяет хранить идентифицирующее имя и текст описания внутри самого файла XML. Элемент label должен иметь атрибуты, указанные в таблице G.11.

Таблица G.1 — Атрибуты элемента label (метка)

Атрибут	Тип данных	Применение	Описание
lang	xsd:language	Требуемый	Язык, используемый в имени или описании
URI	xsd:anyURI	Опциональный	Опциональная связь для получения дальнейшей информации описания

Элемент может появляться n раз, по одному разу для каждого языка. Для идентификации языка используется атрибут lang.

2 Элемент description (описание) обеспечивает возможность хранения текстовых описаний внутри файла XML. Элемент может появиться несколько раз, по одному разу для каждого языка. Элемент описания имеет те же атрибуты, что и элемент метки.

3 Элемент labelRef (ссылочная метка) допускает возможность хранения ссылочных текстов описания внутри файла - источника внешних текстов.

Элемент labelRef предоставляет указатель с помощью своих атрибутов dictID (идентификатор словаря) и textID (идентификатор текста) на текстовые входы в отдельном файле-источнике текстов. На файлы – источники текста имеется ссылка в субэлементах словаря элемента DeviceFunction (функция устройства). Файлы источники текста могут быть любыми файлами, содержащими последовательности символов и другую информацию, например рисунки.

Элемент labelRef также может появляться n раз, позволяя осуществлять ссылки на несколько входов словаря, которые содержат связи с файлами на различных языках. Соответствующие языки определены в атрибуте lang элемента словаря.

Элемент labelRef содержит атрибуты, указанные в таблице G.2.

Таблица G.2 — Атрибуты элемента labelRef (ссылочная метка)

Атрибут	Тип данных	Применение	Описание
dictID	xsd:IDREF	Требуемое	Производится ссылка на один элемент словаря элемента dictionaryList (список словарей); элемент словаря содержит связь с файлом – источником внешнего текста
textID	xsd:string	Опциональное	Производится ссылка на последовательность символов файла – источника внешнего текста с помощью сравнения с образцом

G.2.2.3 Идентификатор языка

Для реализации мультиязыковой поддержки каждая метка получает атрибут, содержащий код языка. Код языка соответствует содержанию элемента метки.

С целью проверки, какие языки поддерживаются в файле XML, в атрибуте supportedLanguages (поддерживаемые языки) элемента ProfileBody (тело профиля) содержится список поддерживаемых языков.

G.2.2.4 Атрибут lang

Идентификатор языка lang состоит из комбинации кода языка (согласно определению, приведенному в ИСО 639-1), необязательного символа «тире» необязательного кода страны (согласно определению, приведенному в ИСО 3166-1). Атрибут lang является атрибутом элемента метки.

Некоторые значения lang приведены в таблице G.3.

Таблица G.3 — Значения атрибута lang

Язык	Значение lang
Английский (США)	en-us
Немецкий (стандартный) Французский (стандартный)	de fr
Испанский (стандартный)	es
Итальянский (стандартный)	It
Португальский (Бразилия)	pt-br

G.2.2.5 Атрибут SupportedLanguages (поддерживаемые языки)

Атрибут supportedLanguages определяет поддерживаемые языки и состоит из списка кодов языков плюс необязательные коды страны.

Пример — supportedLanguages="en-us de fr es"

G.2.2.6 URI

Общий механизм допускает описание URI в контексте элемента метки. URI реализуется с помощью опционального атрибута URI.

Пример — Это используется в контексте метки поставщика, метки параметра, или метки сервиса.

G.3 ProfileHeader (заголовок профиля)

В целях облегчения идентификации файла ETHERNET Powerlink заголовок файла ETHERNET Powerlink должен соответствовать представленной на рисунке G.3 модели, соотвестсвующей ИСО 15745-1.

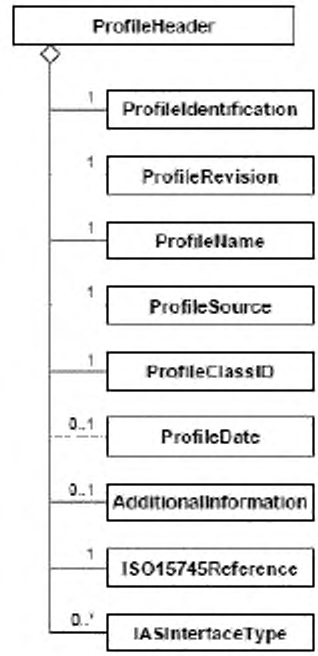


Рисунок G.3 — Диаграмма класса Profile header (заголовок профиля)

Элемент ProfileHeader состоит из следующих элементов:

— элемента ProfileIdentification (идентификация профиля), идентифицирующего текущий профиль;

— элемента ProfileRevision (версия профиля), идентифицирующего текущую версию профиля;

— элемента ProfileName (имя профиля), содержащего описательное имя текущего профиля на английском языке. При наличии в профиле устройства более одного элемента ProfileBody считают, что значение элемента ProfileName (имя профиля) должно быть конкатенацией значений элементов productName (имя продукта) соответствующих элементов DeviceIdentity (идентичность устройства);

— элемента ProfileSource (источник профиля), идентифицирующего средство проверки допустимости текущего профиля;

— элемента ProfileClassID (идентификатор класса профиля), идентифицирующего класс текущего профиля согласно ИСО 15745-1;

— элемента ISO15745Reference, указывающего часть ИСО 15745, издание и технологию, которым соответствует описание.

G.4 Описание шаблона профиля устройства

G.4.1 ProfileBody_Device_EPL

В настоящем разделе определен профиль устройства ETHERNET Powerlink.

ProfileBody_Device_ModbusTCP содержит элементы DeviceIdentity (идентичность устройства), DeviceManager (менеджер устройства), DeviceFunction (функция устройства) и ApplicationProcess (прикладной процесс), представленные на рисунке 32.

Элемент ProfileBody содержит описания:

— единичного устройства (например, датчика ближней локации или электромеханического концевого выключателя) или более сложного устройства (например, прерывателя цепи с количеством параметров до 2500 и с более чем 100 функциями), или

— части устройства, называемой также «модуль» в области программируемых логических контроллеров (например, части контроллера ввода/вывода или блока электрической защиты).

Элемент ProfileBody содержит атрибуты, указанные в таблице G.4.

Таблица G.4 — Атрибуты элемента ProfileBody (тело профиля)

Атрибут	Тип данных	Применение	Описание
formatName	Xsd:string	Фиксированное	Идентификатор формата
formatVersion	Xsd:string	Фиксированное	Идентификатор версии формата
filename	Xsd:string	Требуемое	Имя файла с расширением без пути
fileCreator	Xsd:string	Требуемое	Лицо, разработавшее файл
fileCreationDate	Xsd:date	Требуемое	Дата создания файла
fileCreationTime	Xsd:time	Опциональное	Время создания файла
fileModifiedBy	Xsd:string	Опциональное	Лицо, изменившее файл
fileModificationDate	Xsd.date	Опциональное	Дата последнего изменения файла
fileModificationTime	Xsd:time	Опциональное	Время последнего изменения файла
fileVersion	Xsd:string	Требуемое	Установленная поставщиком версия файла
supportedLanguages	Xsd:NMTOKENS	Опциональное	Список поддерживаемых языков

G.4.2 DeviceIdentity (идентичность устройства)

G.4.2.1 Общие положения

Класс DeviceIdentity (см. рисунок 33) содержит элементы, не зависящие от сети и процесса и описывает идентичность одного устройства или группы устройств.

Таблица G.5 устанавливает атрибут readOnly (только чтение), который присоединяется к элементам vendorName (имя поставщика), vendorID (идентификатор поставщика), vendorText (текст поставщика), deviceFamily (семейство устройства), productFamily (семейство продуктов), productName (имя продукта), productID (идентификатор продукта), productText (текст продукта), orderNumber (номер заказа), version (версия), specificationRevision (спецификация версии) и instanceName (имя экземпляра).

Таблица G.5 — Атрибут элемента vendorName (имя поставщика)

Атрибут	Тип данных	Применение	Описание
readOnly	xsd:boolean	По умолчанию	Указывает является ли для пользователя значение атрибута чтение-только: ложь, истина (по умолчанию)

G.4.2.2 Элемент vendorName (имя поставщика)

Элемент vendorName определяет имя или торговую марку поставщика устройства.

G.4.2.3 Элемент vendorID (идентификатор поставщика)

Элемент vendorID идентифицирует поставщика. Эта информация должна быть заполнена при признании описанного продукта и подтверждении консорциумом.

Примечание — Зависящие от консорциума семейства продуктов и идентификаторы поставщика взаимосвязаны.

G.4.2.4 Элемент vendorText (текст поставщика)

Элемент vendorText предназначен для предоставления поставщиком дополнительной информации о компании, например адреса или номера горячей линии связи. Группа g_labels обеспечивает возможность включения URI поставщика в элемент vendorText.

G.4.2.5 Элемент deviceFamily (семейство устройства)

Элемент deviceFamily определяет семейство устройства.

Пример — Примерами семейства устройства являются следующие:

- ***привод с переменной скоростью;***
- ***прерыватель цепи;***
- ***датчик давления.***

G.4.2.6 Элемент productFamily (семейство продуктов)

Элемент productFamily устанавливает определенную поставщиком принадлежность типа устройства к некоторому набору устройств в семействе. Действительные значения productFamily - система, инструмент или отношение к консорциуму.

Примечание — Определяемые консорциумом семейства продуктов и идентификаторы поставщика взаимосвязаны.

G.4.2.7 Элемент productName (имя продукта)

Элемент productName устанавливает определенное поставщиком обозначение или наименование типа устройства.

G.4.2.8 Элемент productID (идентификатор продукта)

Элемент productID устанавливает определенную поставщиком уникальную идентификацию описываемого типа устройства.

G.4.2.9 Элемент productText (текст продукта)

Элемент productText обеспечивает для поставщика возможность предоставить короткое текстовое описание типа устройства.

G.4.2.10 Элемент orderNumber (номер заказа)

Элемент orderNumber используют для сохранения единичного номера заказа данного продукта или набора номеров различных заказов продуктов в

семействе продуктов, в зависимости от того, относится профиль устройства к одному продукту или семейству продуктов.

G.4.2.11 Элемент version (версия)

Элемент version используют для хранения различных типов информации о версиях. Возможны элементы для нескольких версий.

Элемент version имеет атрибуты, указанные в таблице G.6.

Таблица G.6 — Атрибуты элемента version (версия)

Атрибут	Тип данных	Применение	Описание
versionType	Xsd:NMToken	Требуемое	Тип версии: — SW- программная; — FW – программно-аппаратная; — HW- аппаратная
readOnly	Xsd:Boolean	По умолчанию	Указывает, является ли значение для пользователя только для чтения: ложь, истина (по умолчанию)

G.4.2.12 Элемент buildDate (создание даты)

Элемент buildDate определяет дату создания программного блока.

G.4.2.13 Элемент specificationRevision (спецификация версии)

Элемент specificationRevision содержит версию спецификации, которой соответствует устройство.

G.4.2.14 Элемент instanceName (имя экземпляра)

Данный элемент содержит имя экземпляра устройства.

G.4.3 DeviceManager (менеджер устройства)

G.4.3.1 Общие положения

Элемент DeviceManager определяет список индикаторов, предоставленных типом устройства, если они существуют.

G.4.3.2 LEDList (LED список)

G.4.3.2.1 Общие положения

Элементы LEDList, представленные на рисунке G.4, устанавливают

число и тип индикаторов, содержащихся в типе устройства.

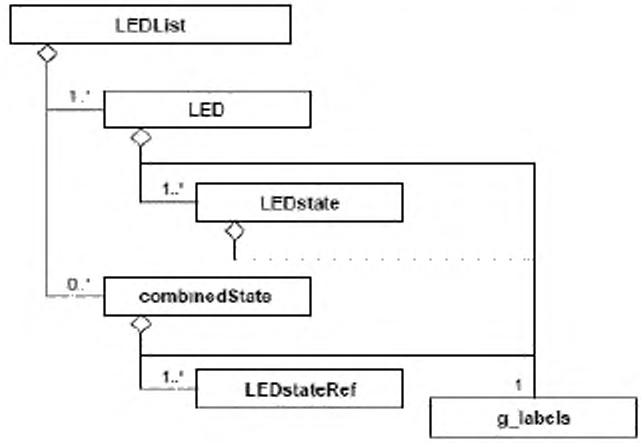


Рисунок G.4 —LEDList (LED список)

G.4.3.2.2 LED

Элемент LED описывает характеристики одного LED в данном типе устройства. Подробное описание характеристик может быть предоставлено в группе g_labels.

Дополнительные характеристики LED представлены в виде атрибутов элемента LED, указанных в таблице G.7.

Таблица G.7 — Атрибуты элемента LED

Атрибут	Тип данных	Применение	Описание
LEDcolors	xsd:string	Требуемое	Для цветов LED; правильные значения − монохромные и двухцветные
LEDtype	xsd:string	Опциональное	При грубой классификации контролируемых объектов или функциональных параметров; правильные значения - IO, устройство и коммуникации

Дополнительно элемент LED содержит один или более элементов LEDstate (LED состояние), которые определяют состояния устройства, сигнализируемые с помощью LED, и визуальные характеристики, используемые для сигнализации о состояниях.

Визуальные параметры, используемые для сигнализации о состояниях, закодированы в виде значений атрибутов элемента LEDstate согласно таблице G.18. Дополнительно состояниям LED присвоены уникальные идентификаторы.

Таблица G.8 — Атрибуты элемента LEDstate (LED состояние)

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный ID для состояния LED; может быть объектом ссылки из элемента LEDstateRef
State	xsd:string	Требуемое	Состояние LED; возможные значения атрибута: включено, выключено, мигание
LEDcolor	xsd:string	Требуемое	Цвет состояния LED; правильные значения: зеленый, янтарный, красный
flashingPeriod	xsd:unsignedInt	Опциональное	Если состояние – мигание, то период мигания LED измеряют в миллисекундах
impulsWidth	xsd:unsignedByte	По умолчанию	Ширина импульса мигания в процентах периода мигания; если атрибут impulsWidth пропущен, то значение по умолчанию равно 50
numberOfImpulses	xsd:unsignedByte	По умолчанию	Число импульсов в случае более одного импульса мигания за период мигания; если атрибут присутствует, атрибут impulsWidth должен быть представлен также, если атрибут numberOfImpulses пропущен, значение по умолчанию - 1

G.4.3.2.3 Элемент combinedState (комбинированное состояние)

Элемент combinedState разрешает индикацию состояний устройства, которые сигнализируются с помощью более одного LED.

Описание комбинированного состояния предоставлено в группе g_labels.

Ссылка на состояния LED, участвующие в сигнализации комбинированного состояния, должна осуществляться с помощью не менее

двух субэлементов LEDstateRef (ссылка на состояние LED) элемента combinedState (комбинированное состояние).

Ссылка на элемент LEDstate кодируется как значение атрибута единичного атрибута элемента LEDstateRef (см. таблицу G.9).

Таблица G.9 — Атрибуты элемента LEDstateRef (ссылка на состояние LED)

Атрибут	Тип данных	Применение	Описание
stateIDRef	Xsd:IDREF	Требуемое	Уникальный ID ссылочного элемента LEDstate

G.4.4 DeviceFunction (функция устройства)

G.4.4.1 Общие положения

Элемент DeviceFunction, указанный на рисунке 35, определяет вид устройства в каталоге, представленный в виде набора capabilities (возможности), перечисляющих характеристики устройства и указывающих его соответствие различным стандартам.

G.4.4.2 Элемент capabilities (возможности)

G.4.4.2.1 Общие положения

Обязательный элемент capabilities описывает все функциональные параметры, их характеристики и другие основные параметры устройства, которые должны определяться с помощью инструментов, использующих профиль устройства для выбора изделий с такими же или аналогичными свойствами.

Элемент capabilities (возможности) описывает характеристики устройства в виде текста и содержит последовательность одного или более элементов characteristicsList (список характеристик) и опциональный элемент standardComplianceList (стандартный список соответствия).

G.4.4.2.2 Элемент characteristicsList (список характеристик)

G.4.4.2.2.1 Общие положения

Элемент characteristicsList (список характеристик) представляет собой совокупность характеристик. Этот элемент должен содержать не менее

одного субэлемента characteristic (характеристика). Характеристики в списке могут быть ассоциированы с категорией, которая может быть выражена в виде текстового содержания субэлемента g_labels опционального субэлемента category (категория) элемента characteristicsList.

G.4.4.2.2.2 Элемент characteristic (характеристика)

Элемент characteristic описывает единичную характеристику устройства и содержит обязательный элемент characteristicName (имя характеристики), а также один или несколько элементов characteristicContent (содержание характеристики).

G.4.4.2.2.3 Элемент characteristicName (имя характеристики)

Обязательный элемент characteristicName обозначает основную техническую характеристику устройства. Для наименований характеристик рекомендуется использовать словарь в перечне данных изделия.

Пример — «Максимальное рабочее напряжение», «Защита от перегрузки», «Электрическая прочность».

G.4.4.2.2.4 Элемент characteristicContent (содержание характеристики)

Этот обязательный элемент содержит значение элемента характеристики. Множественные значения могут быть выражены с помощью использования нескольких элементов characteristicContent.

Пример — Пример единичного значения для «Максимальное рабочее напряжение» – 680 В.

G.4.4.2.3 Элемент standardComplianceList (стандартный список соответствия)

Элемент standardComplianceList представляет собой собрание элементов compliantWith (соответствие с). Этот элемент является опциональным; если он присутствуют, то содержит не менее одного субэлемента compliantWith.

Субэлемент compliantWith имеет атрибуты, которые устанавливают соответствие устройства международным стандартам или внутренним

стандартам компании. Содержание типа g_labels этого элемента может включать в себя замечания, касающиеся этих стандартов.

Наименование или обозначения стандарта предоставляют с помощью требуемого атрибута name (имя) элемента compliantWith. Второй атрибут – величина по умолчанию range (диапазона) элемента compliantWith – определяет диапазон применимости стандарта согласно указаниям таблицы G.10.

Таблица G.10 — Атрибуты элемента compliantWith (соответствие с)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Наименование или обозначения стандарта
Range	xsd:NMToken	По умолчанию	Два возможных пронумерованных значения атрибута - международный (по умолчанию) или внутренний

G.4.4.3 Элемент picturesList (список изображений)

Элемент picturesList обеспечивает возможность связать изображения с профилем устройства и содержит один или более субэлементов с изображениями, доступ к которым обеспечивается через субэлемент g_labels.

В таблице G.11 определены атрибуты субэлемента изображения: опциональное число изображений, обязательная связь с внешними источниками, содержащими графическую информацию.

Таблица G.11 — Атрибуты элемента picture (изображение)

Атрибут	Тип данных	Применение	Описание
URI	xsd:anyURI	Требуемое	Связь с внешним источником
number	xsd:unsignedInt	Опциональное	Номер изображения

G.4.4.4 Элемент dictionaryList (список словарей)

Опциональный элемент dictionaryList обеспечивает возможность включения в профиль устройства связей с внешними файлами - источниками текстов и содержит один или более элементов словарей, каждый из которых включает в себя один или более субэлементов файлов. Несколько файлов необходимо при наличии различных форматов файлов в словаре.

Обязательный атрибут lang типа xsd:language определяет языки, используемые в файлах, которые связаны с элементом словарь (см. таблицу G.12). Обязательный атрибут uniqueID (уникальный идентификатор) типа xsd:ID поддерживает уникальную идентификацию входа в словарь, на который имеется ссылка из атрибута dictID элемента labelRef согласно таблице G.2.

Таблица G.12 — Атрибуты элемента dictionary (словарь)

Атрибут	Тип данных	Применение	Описание
Lang	xsd:language	Требуемый	Язык, используемый для файлов, относящихся к входу в словарь
uniqueID	xsd:ID	Требуемый	Уникальный ID входа в словарь

Субэлемент file (файл) содержит обязательный атрибут, указанный в таблице G.13.

Таблица G.13 — Атрибуты элемента file (файл)

Атрибут	Тип данных	Применение	Описание
URI	xsd:anyURI	Требуемый	Связь с соответствующим файлом

G.4.5 ApplicationProcess (прикладной процесс)

G.4.5.1 Общие положения

Элемент ApplicationProcess представляет набор сервисов и параметров, формирующих поведение и интерфейсы устройства в терминах приложения, независимо от технологии устройства и лежащих в его основе коммуникационных сетей и протоколов коммуникаций.

Субэлементы элемента ApplicationProcess, представленные на рисунке 36, обеспечивают общий подход к описанию функций устройства, имеющих произвольную, одноуровневую или иерархическую структуру.

Функции моделируются как типы функций, которые реализуются внутри устройства или, если необходима иерархическая структура, внутри типов функций. Параметры интерфейса этих реализаций функций, которые могут соответствовать простому или сложному типу данных, ассоциированы с параметрами устройства путем построения ссылки из параметра на соответствующий параметр интерфейса реализации функции как в

одноуровневых, так и в иерархических структурах.

Элемент ApplicationProcess содержит до пяти списков объектов (см. рисунок 36):

— два списка, определяющих типы данных (опционально) и типы функций (обязательно);

— один обязательный список, определяющий реализации функций на уровне устройства (включая соединения между реализациями);

— один обязательный список, определяющий параметры устройства, и

— один опциональный список, определяющий группы параметров (комбинации параметров для специальных целей).

G.4.5.2 Элемент dataTypeList (список типов данных)

G.4.5.2.1 Общие положения

Опциональный элемент dataTypeList присутствует, если внутри деклараций параметров в профиле устройства необходимы сложные типы данных типа таблиц или структур.

Элемент dataTypeList, представленный на рисунке G.5, содержит последовательность одного или нескольких элементов из следующего набора:

- элемент array (таблица);
- элемент struct (структура);
- элемент enum (нумерация) или
- производный элемент.

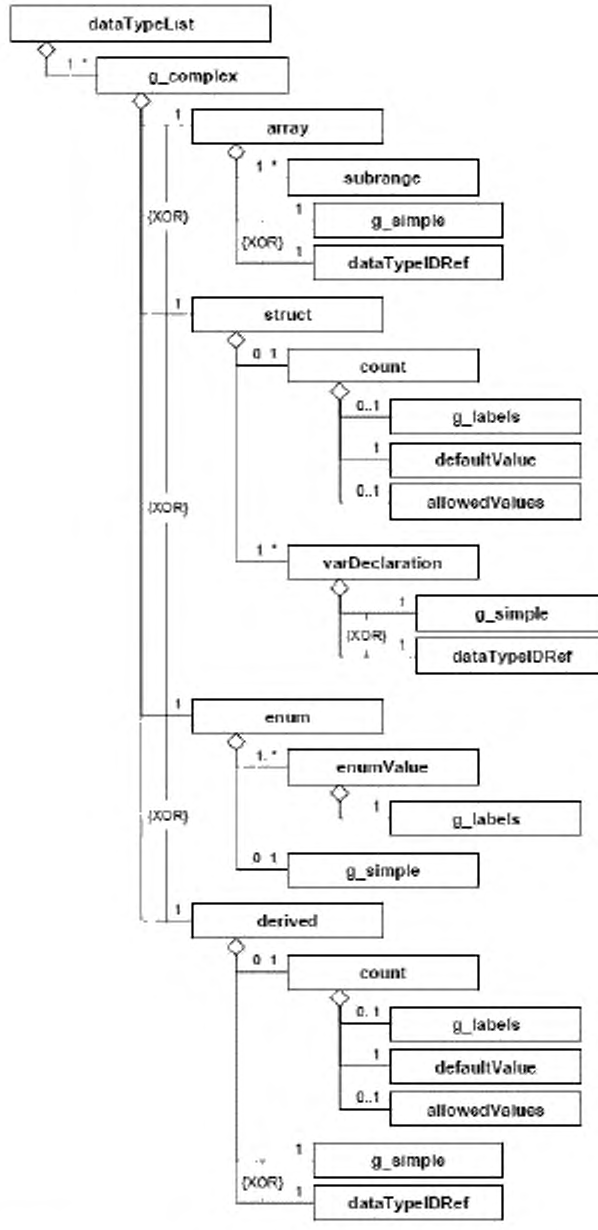


Рисунок G.5 — dataTypeList (список типов данных)

G.4.5.2.2 Общие элементы

G.4.5.2.2.1 Группа g_simple (простой)

Группа g_simple содержит набор элементов, имена которых представляют имена всех простых типов данных, допускаемых при определении параметров внутри профиля устройства. Простые типы данных соответствуют элементарным типам данных, определенным в МЭК 61131-3; к

ним добавляют типы данных BITSTRING и CHAR (=STRING[1]).

Эти элементы вводят внутри группы, допуская их размещение непосредственно как субэлементов элемента таблицы (или элемента varDeclaration (объявление переменной) по G.4.5.4.3.2).

G.4.5.2.2.2 Элемент count (счет)

Элемент count определяет число используемых блоков базового типа производного типа. Многоязычные имена и/или описания для элемента count предоставляют с помощью группы g_labels. По вопросу описания группы g_labels см. G.2.2.2.

Элемент count описывают следующими способами:

– его атрибутами;

– обязательным субэлементом defaultValue (значение по умолчанию) и, возможно, пустым набором субэлементов g_labels и allowedValues (допустимые значения).

Число блоков выражают как значение атрибута defaultValue элемента count. Атрибут allowedValue определяет диапазон значений по умолчанию.

Субэлементы defaultValue и allowedValues описаны в G.4.5.6.2.5 и G.4.5.6.2.7.

Элемент count должен иметь атрибуты, указанные в таблице G.14.

Таблица G.14 — Атрибуты элемента count (счет)

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный идентификатор count
Access	xsd:NMToken	По умолчанию	Определяет, какие операции являются правильными для count: – чтение – доступ только для чтения (значение по умолчанию) – запись – доступ только для записи – чтение/запись – доступ для чтения и записи – нет Доступа – доступ не разрешается

G.4.5.2.3 Элемент array (массив)

G.4.5.2.3.1 Общие положения

Элемент array (массив) предназначен для описания типа данных array (массив), ссылка на которые может выполняться из параметров интерфейса типа функций, из других определений типа таблицы или из параметра компонента определения структурированного типа данных.

Элемент array содержит не менее одного элемента subrange(поддиапазон) и либо элемент, описывающий простой тип данных из группы g_simple, либо элемент dataTypeIDRef (ссылка идентификатора типа данных), имеющий ссылки на один из определенных сложных типов данных элемента dataTypeList (список типов данных).

В случае многомерных таблиц должны присутствовать несколько элементов subrange. В этом случае первый элемент поддиапазона в последовательности определяет поддиапазон для первого слева индекса таблицы, а последний элемент поддиапазона в последовательности определяет поддиапазон для крайнего правого индекса таблицы.

Элемент array (массив) содержит атрибуты, указанные в таблице G.15.

Таблица G.15 — Атрибуты элемента array (массив)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя типа данных (тип таблица)
uniqueID	xsd:ID	Требуемое	Уникальный ID (тип таблица)
Description	xsd:string	Опциональное	Опциональное текстовое описание (тип таблица)

G.4.5.2.3.2 Элемент subrange (поддиапазон)

Элемент subrange определяет нижний и верхний предельные значения индекса таблицы для одного измерения таблицы. Этот элемент не имеет субэлементов.

Предельные значения типа xsd:long содержатся в двух атрибутах элемента поддиапазона, указанных в таблице G.16.

Таблица G.16 — Атрибуты элемента subRange (поддиапазон)

Атрибут	Тип данных	Применение	Описание
lowerLimit	xsd:long	Требуемое	Нижнее предельное значение поддиапазона
UpperLimit	xsd:long	Требуемое	Верхнее предельное значение поддиапазона

G.4.5.2.4 Элемент struct (структура)

G.4.5.2.4.1 Общие положения

Элемент struct предназначен для описания типа структурированных данных, на который может выполняться ссылка из параметра интерфейса типа функции, из определения типа таблицы или из параметра компонента, входящего в определение другого типа структурированных данных.

Элемент struct содержит последовательность одного или более элементов varDeclaration (объявление переменной), которые определяют компоненты типа структурированных данных.

Элемент struct должен содержать атрибуты, указанные в таблице G.17.

Таблица G.17 — Атрибуты элемента struct (структура)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя типа данных (тип структурированных данных)
uniqueID	xsd:ID	Требуемое	Уникальный ID (тип структурированных данных)
Description	xsd:string	Опциональное	Опциональное текстовое описание (тип структурированных данных)

G.4.5.2.4.2 Элемент varDeclaration (объявление переменной)

В контексте определения типа структурированных данных элемент varDeclaration описывает единичный параметр компоненты (составляющую) структуры.

В контексте определения интерфейса функции элемент varDeclaration описывает единичный параметр интерфейса типа функции.

Тип данных параметра компоненты или параметра интерфейса

определяется либо с помощью элемента, описывающего простой тип данных группы g_simple, либо с помощью элемента dataTypeIDRef (ссылка идентификатора типа данных), который выполняет ссылку на один из определенных сложных типов данных в элементе dataTypeList (список типов данных).

Все дальнейшие характеристики параметров содержатся в атрибутах элемента varDeclaration, указанных в таблице G. 18.

Таблица G.18 — Атрибуты элемента varDeclaration (объявление переменной)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя параметра интерфейса или компонента структуры
uniqueID	xsd:ID	Требуемое	Уникальный ID параметра интерфейса или компонента структуры (см. примечание 1)
Size	xsd:string	Опциональное	Число элементов, если параметр интерфейса или компонента структуры анонимного типа ARRAY, BITSTRING, STRING или WSTRING (см. примечание 2)
InitialValue	xsd:string	Опциональное	Начальное значение параметра интерфейса или компонента структуры (см. примечание 3)
Description	xsd:string	Опциональное	Опциональное текстовое описание параметра интерфейса или компонента структуры

Примечание 1 — При создании уникальных ID для параметров важно соблюдать уникальность всех ID файла источника XML. Для создания возможности одинаковых имен для параметров компонентов различных структур данных и одинаковых имен для параметров интерфейса типов функций ID параметра должен обычно соединять типовое имя типа структурированных данных или типовое имя типа функций с именем параметра, для гарантии уникальности.

Примечание 2 — Анонимные типы определяют размер таблицы, битовой строки или строки непосредственно в декларации параметра, а не путем ссылки на тип сложных данных с именем. Для таблицы тип данных единичного элемента таблицы определяют типом данных параметра. Для битовой строки, единичный элемент таблицы является одним битом.

Примечание 3 — Данный атрибут определяет начальное (по умолчанию) значение параметра интерфейса типа функции. Он копирует заданное значение по умолчанию параметра, ассоциированного с параметром интерфейса экземпляра функции.

G.4.5.2.5 Элемент enum (нумерация)

G.4.5.2.5.1 Общие положения

Элемент enum предназначен для описания типа пронумерованных данных, на которые могут быть ссылки из параметра интерфейса типа функции, из определения типа таблицы или из параметра компонента определения типа структурированных данных.

В соответствии с рисунком G.5 этот элемент содержит последовательность одного или более элементов enumValue (значение нумерации), которые определяют константы нумерации типа пронумерованных данных. Тип данных констант нумерации опционально определяется с помощью элемента, описывающего тип простых данных из группы g_simple.

Элемент enum содержит атрибуты, указанные в таблице G.19.

Таблица G.19 — Атрибуты элемента enum (нумерация)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Типовое имя типа пронумерованных данных
uniqueID	xsd:ID	Требуемое	Уникальный ID типа пронумерованных данных
Size	xsd:string	Опциональное	Опциональное число пронумерованных значений типа пронумерованных данных
Description	xsd:string	Опциональное	Опциональное текстовое описание типа пронумерованных данных

G.4.5.2.5.2 Элемент enumValue (значение нумерации)

Элемент enumValue определяет имя (имена) и опционально числовое значение единичной константы нумерации. Имя (имена) устанавливаются в группе g_labels, а значение содержится в атрибуте единичного значения элемента enumValue согалсно таблице G.20.

Таблица G.20 — Атрибуты элемента enumValue (значение нумерации)

Атрибут	Тип данных	Применение	Описание
Value	xsd:string	Опциональное	Опциональный атрибут: фиксированное числовое значение константы нумерации, представленное в виде строки символов

G.4.5.2.6 Элемент derived (производный)

Элемент derived предназначен для вывода нового типа данных из данного базового типа.

Элемент derived содержит опциональный элемент count (счет) и либо элемент, описывающий тип простых данных из группы g_simple, либо элемент dataTypeIDRef, который ссылается на один из определенных сложных типов данных элемента dataTypeList (список типов данных).

Если элемент count (счет) пропущен, определение производного типа вводит только имя нового типа для соответствующего базового типа. Если элемент count присутствует, он определяет число блоков соответствующего базового типа, используемых для построения производного типа (например, базовый тип BITSTRING, count = 4 определяет производный тип размером 4 бита).

Элемент derived содержит атрибуты, указанные в таблице G.21.

Таблица G.21 — Атрибуты элемента derived (производный)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя типа данных производного типа
uniqueID	xsd:ID	Требуемое	Уникальный ID производного типа
Description	xsd:string	Опциональное	Опциональное текстовое описание производного типа

G.4.5.3 Элемент functionTypeList (список типов функций)

Если опциональный элемент ApplicationProcess (прикладной процесс) присутствует в профиле устройства, он содержит обязательный элемент functionTypeList, представленный на рисунке G.6.

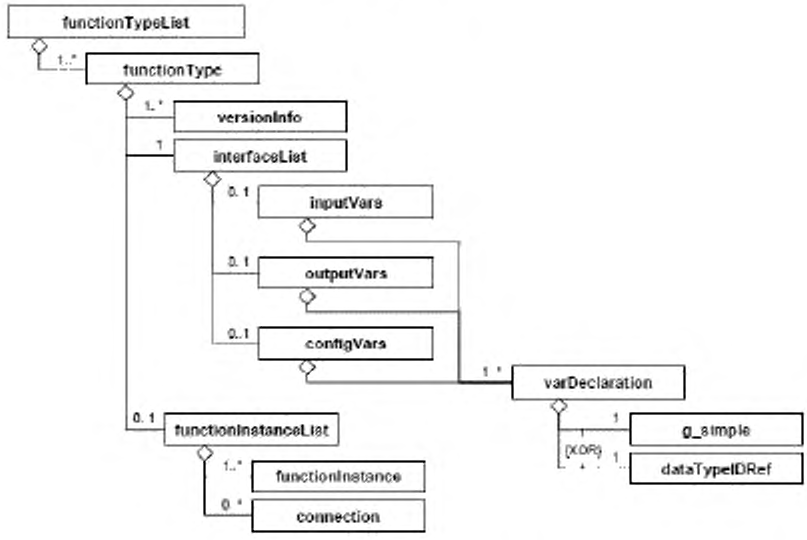


Рисунок G.6 — Диаграмма класса functionTypeList (список типов функций)

Элемент functionTypeList представляет последовательность из одного или многих элементов functionType (тип функции).

Каждый из элементов functionType представляет описание типа функции устройства, на которую имеется ссылка из не менее чем одного экземпляра типа функции в элементе functionInstanceList (список экземпляров функции). Ссылки из более чем одного экземпляра того же типа функций также возможны.

Описание типа функции содержит все те объекты и данные, которые являются общими для всех экземпляров данного типа функций.

Пример 1 — Примерами являются параметры или параметры функций, объекты, формирующие интерфейс функции (тип соответствующего экземпляр).

Пример 2 — Другими примерами являются сущности, содержащиеся внутри тела функции в иерархически структурированном функциональном описании. Эти сущности, расположенные внутри элемента functionInstanceList (список экземпляров функции) внутри типа функции, ссылаются на другие типы функции в списке типов функций.

G.4.5.4 Элемент functionType (тип функции)

G.4.5.4.1 Общие положения

Элемент functionType содержит один или несколько элементов versionInfo (информация о версии), обязательный элемент interfaceList (список интерфейсов) и опциональный элемент functionInstanceList (список экземпляров функции). Элемент functionInstanceList присутствует в элементе functionType только в том случае, если функция иерархически структурирована.

Дополнительно элемент functionType должен содержать атрибуты, указанные в таблице G.22.

Таблица G.22 — Атрибуты элемента functionType (тип функции)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Типовое имя типа функции
uniqueID	xsd:ID	Требуемое	Уникальный ID типа функции
Description	xsd:string	Опциональное	Опциональное текстовое описание типа функции
Package	xsd:string	Опциональное	Опциональная текстовая ассоциация типа функции с «пакетом» или аналогичной схемой классификации – использование этого атрибута обеспечивает валидатор профиля

G.4.5.4.2 Элемент versionInfo (информация о версии)

Обязательный элемент versionInfo внутри элемента functionType (тип функции) предоставляет информацию о истории версий типа функций (относящуюся к определению интерфейса).

Для прослеживания истории версий элемент versionInfo может быть введен несколько раз. Множественные входы должны быть организованы внутри элемента functionType в следующей последовательности:

a) первый вход представляет наиболее раннюю версию;

b) второй вход представляет непосредственно предшествующую версию;

c) последний вход представляет первую выпущенную версию.

Данный элемент будет предоставлен один раз при создании описания типа функции. Новые элементы будут добавляться при введении модификации типа функции, которая приводит к модифицированной версии профиля устройства.

Элемент versionInfo должен содержать атрибуты, указанные в таблице G.23.

Таблица G.23 — Атрибуты элемента versionInfo (информация о версии)

Атрибут	Тип данных	Применение	Описание
Organization	xsd:string	Требуемое	Имя организации, поддерживающей тип функции
Version	xsd:string	Требуемое	Обозначение версии в истории версий; предлагаемый формат: "xx.yy" (xx,yy = 0..255)
Author	xsd:string	Требуемое	Имя лица, поддерживающего тип функции
Date	xsd:date	Требуемое	Дата данной версии
Remarks	xsd:string	Опциональное	Описательная информация, относящаяся к специальному этапу истории версий

G.4.5.4.3 Элемент interfaceList (список интерфейсов)

G.4.5.4.3.1 Общие положения

Обязательный элемент interfaceList элемента functionType (тип функции) предоставляет определение интерфейса типа функции. Элементами интерфейса являются следующие:

- входные параметры и/или
- выходные параметры, и/или
- параметры конфигурации типа функции.

Следовательно элемент interfaceList содержит последовательность трех элементов, где каждый элемент представляет списки от одной до нескольких деклараций параметров, закодированных как элементы varDeclaration:

– один опциональный элемент inputVars (входные переменные);

– один опциональный элемент outputVars (выходные переменные) и

– один опциональный элемент configVars (конфигурационные переменные).

Ни один из элементов interfaceList, inputVars, outputVars или configVars не содержит каких-либо атрибутов.

G.4.5.4.3.2 Элемент varDeclaration (объявление переменной)

В контексте определения структурированного типа данных элемент varDeclaration описывает единичный параметр компоненту (элемент) структуры.

В контексте определения интерфейса типа функции элемент varDeclaration описывает единичный параметр интерфейса типа функции.

Тип данных параметра компонента или параметра интерфейса определяется либо элементом, описывающим простой тип данных из группы g_simple, либо элементом dataTypeIDRef (ссылочный идентификатор типа данных), который ссылается на один из определенных сложных типов данных внутри элемента dataTypeList (список типов данных).

В G.4.5.2.2.1 приведено описание группы, а в g_simple и G.4.5.4.3.3 – элемента dataTypeIDRef.

Все дополнительные характеристики параметра содержатся в атрибутах элемента varDeclaration, указанных в таблице G.24.

Таблица G.24 — Атрибуты элемента varDeclaration (объявление переменной)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя параметра интерфейса или компонента структуры
UniqueID	xsd:ID	Требуемое	Уникальный ID параметра интерфейса или компонента структуры
Size	xsd:string	Опциональное	Число элементов, если параметр интерфейса или компонент структуры имеют тип анонимный ARRAY, BITSTRING, STRING или WSTRING
InitialValue	xsd:string	Опциональное	Начальное значение параметра интерфейса или компонента структуры
Description	xsd:string	Опциональное	Опциональное текстовое описание параметра интерфейса или компонента структуры

G.4.5.4.3.3 Элемент dataTypeIDRef (ссылка идентификатора типа данных)

Элемент dataTypeIDRef предназначен для ссылки на сложный тип данных элемента dataTypeList (список типов данных) (см. G.4.5.2) либо из параметра интерфейса типа функции, либо из определения типа таблицы, либо из параметра компонента определения структурированного типа данных.

Ссылка типа xsd:IDREF предоставляется в виде атрибута элемента dataTypeIDRef согласно таблице G.25.

Таблица G.25 — Атрибуты элемента dataTypeIDRef (ссылка идентификатора типа данных)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef (ссылка на уникальный идентификатор)	xsd:IDREF	Требуемое	Уникальный ID ссылочного типа данных

G.4.5.5 Элемент functionInstanceList (список экземпляров функции)

G.4.5.5.1 Общие положения

Если опциональный элемент ApplicationProcess (прикладной процесс) присутствует в профиле устройства, он содержит обязательный элемент functionInstanceList, который включает в себя последовательность одного или нескольких элементов functionInstance (экземпляр функции) и от ни одного до нескольких элементов connection (соединение).

На уровне процесса приложения элементы functionInstance представляют допустимые функции приложения типа устройства, не зависящие от типа или протокола сети. Элементы connection представляют соединения (если они существуют) между конкретными выходными и входными параметрами этих экземпляров функции.

Элемент functionInstanceList также появляется как опциональный субэлемент элемента functionType (см. G.4.5.4). Аналогично на уровне процесса приложения элемент functionInstanceList в этом случае содержит последовательность одного или нескольких элементов functionInstance и от ни одного до нескольких элементов connection.

Элемент functionInstanceList присутствует в элементе functionType если функция иерархически структурирована. В этом случае элементы functionInstance представляют внутренние функции, содержащиеся в типе функций, а элементы connection – опциональные внутренние соединения. Эти функции и их опциональные соединения могут быть реализованы совместно с реализацией содержащего их типа функций.

Элемент functionInstanceList не содержит каких-либо атрибутов.

G.4.5.5.2 Элемент functionInstance (экземпляр функции)

Обязательный элемент functionInstance не содержит субэлементов.

Элемент functionInstance должен содержать атрибуты, указанные в таблице G.26.

Таблица G.26 — Атрибуты элемента functionInstance (экземпляр функции)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя экземпляра функции
uniqueID	xsd:ID	Требуемое	Уникальный ID экземпляра функции (см. примечание)
typeIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного типа функции
Description	xsd:string	Опциональное	Опциональное текстовое описание экземпляра функции
Примечание — При создании уникального ID для экземпляра функции важно, чтобы этот ID был уникальным во всем создаваемых ID внутри файла - источника XML. Для создания возможности одинаковых имен для экземпляров функции при различных типах функций ID экземпляра функции должен обычно образовываться путем соединения имени типа функции с именем экземпляра функции для обеспечения уникальности.			

G.4.5.5.3 Элемент connection (соединение)

Опциональный элемент connection определяет соединение между выходным параметром экземпляра функции и входным параметром другого экземпляра функции. Внутри типов функции соединение также может быть проведено между входным параметром типа функции и входным параметром содержащегося в типе экземпляра функции или между выходным параметром содержащегося в типе экземпляра функции и выходным параметром типа функции. Элемент connection может появляться от ни одного до нескольких раз.

Элемент connection содержит атрибуты, указанные в таблице G.27.

Таблица G.27 — Атрибуты элемента connection (соединение)

Атрибут	Тип данных	Применение	Описание
Source	xsd:string	Требуемое	Начальная точка соединения
Destination	xsd:string	Требуемое	Конечная точка соединения
Description	xsd:string	Опциональное	Опциональное текстовое описание соединения

Пример — Значения атрибутов source (источник) и destination (назначение) могут быть использованы для кодирования начальной и конечной точек соединения с помощью синтаксиса <function_instance_name>'.'<variable_name>; пример для значения атрибута источник: 'PowerMeasures. Frequency' ('Мощность измерения. Частота'). Соединения с параметрами интерфейса типа функции используют только имена параметров интерфейса.

G.4.5.6 Элемент parameterList (список параметров)

G.4.5.6.1 Общие положения

Если в профиле устройства присутствует опциональный элемент ApplicationProcess (прикладной процесс), он содержит обязательный элемент parameterList (список параметров), указанный на рисунке G.7, который представляет последовательность одного или нескольких элементов parameter (параметр).

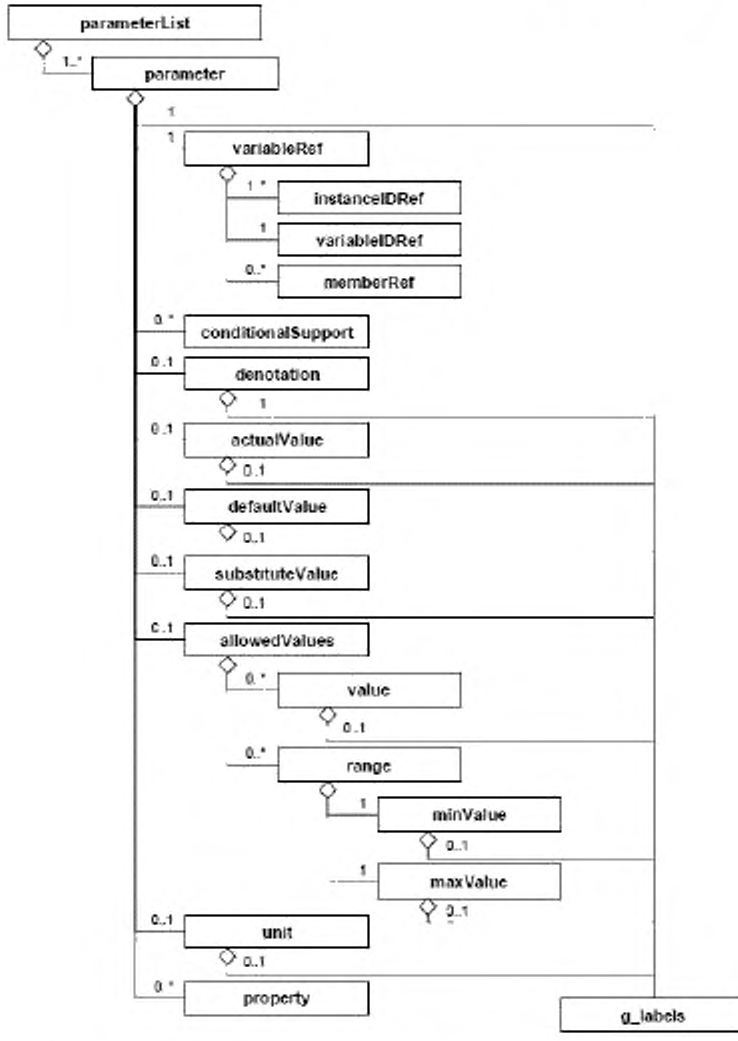


Рисунок G.7 — parameterList (список параметров)

Каждый из элементов parameter представляет параметр профиля устройства. Многоязычные имена и/или описания параметров предоставляются с помощью группы g_labels. Группа g_labels описана в G.2.2.2.

Параметр описывается следующими способами:

– его атрибутами;

– путем ссылки на один или более параметров интерфейса одного или более экземпляров функции (обязательный элемент variableRef (ссылка на переменную));

– возможно, пустым набором субэлементов (conditionalSupport (условная поддержка), denotation (обозначение), actualValue (фактическое значение), defaultValue (значение по умолчанию), substituteValue (подстановочное значение), allowedValues (допустимые значения), блок, свойство и g_labels).

Примечание — Ссылки на многие параметры представляют собой особый случай: специальные параметры могут одновременно иметь ссылку на выходной параметр одного экземпляра функции и на входной параметр другого экземпляра функции. В этом случае типы данных двух параметров должны быть одинаковыми. Программа синтаксического анализа XML не может обеспечить проверку эквивалентности типов данных. Это может быть проверено только с помощью вспомогательного инструмента.

G.4.5.6.2 Элемент parameter (параметр)

G.4.5.6.2.1 Общие положения

Элемент parameter должен содержать атрибуты, указанные в таблице G.28.

Таблица G.28 — Атрибуты элемента parameter (параметр)

Атрибут	Тип данных	Применение	Описание
UniqueID	xsd:ID	Требуемое	Уникальный ID параметра
Access	xsd:NMToken	По умолчанию	Определяет, какая операция является правильной для параметра: — чтение – доступ только для чтения (значение по умолчанию); — запись – доступ только для записи; — чтениеЗапись – доступ для чтения и записи; — нетДоступа – доступ не допускается

Окончание таблицы G.28

Атрибут	Тип данных	Применение	Описание
Support	xsd:NMToken	Опциональное	Определяет, должен или нет параметр использоваться в устройстве; правильными значениями являются следующие: — обязательно – использование параметра требуется; — опционально – использование параметра возможно, но не обязательно; — условно – использование параметра требуется, в том случае если один или более других необязательных параметров используются; эти параметры определяются с помощью субэлемента conditionalSupport (условная поддержка)
Persistence	xsd:Boolean	По умолчанию	Определяет поведение после отказа питания; правильные значения - ложь (по умолчанию) и истина
Offset	xsd:string	Опциональное	Сдвиг, который добавляется к фактическому значению для образования масштабированной величины: ТехническоеЗначение = (значение параметра + сдвиг) * множитель; если не присутствует, принимается сдвиг = 0
Multiplier	xsd:string	Опциональное	Коэффициент масштабирования, на который умножается фактическое значение для образования масштабированной величины: ТехническоеЗначение = (значение параметра + сдвиг) * множитель; если не присутствует, принимается множитель = 1

G.4.5.6.2.2 Элемент conditionalSupport (условная поддержка)

Один или более элементов conditionalSupport присутствуют только в том случае, если значение атрибута support (поддержка) элемента parameter (параметр) conditional (условный). Каждый элемент имеет ссылку на

единичный опциональный параметр. Если хотя бы один из этих опциональных параметров используется, условный параметр также должен использоваться.

Элемент conditionalSupport должен содержать единичный атрибут, указанный в таблице G.29.

Таблица G.29 — Атрибуты элемента conditionalSupport (условная поддержка)

Атрибут	Тип данных	Применение	Описание
paramIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного опционального параметра

G.4.5.6.2.3 Элемент denotation (обозначение)

Элемент denotation предназначен для поддержки зависящих от приложения альтернативных имен параметра на различных языках. Многоязычные альтернативные имена (и, при необходимости, дополнительные описания) параметра предоставляются с помощью группы g_labels. Элемент denotation не имеет атрибутов.

G.4.5.6.2.4 Элемент actualValue (фактическое значение)

Элемент actualValue предназначается для поддержки фактического значения параметра. Опциональный субэлемент g_labels может предоставлять многоязычную описательную информацию для этой величины. Эта величина определяется в атрибуте value (значение) элемента actualValue. Могут быть также указаны сдвиг и множитель.

Атрибуты элемента actualValue должны соответствовать указанным в таблице G.30.

Таблица G.30 — Атрибуты элемента actualValue(фактическое значение)

Атрибут	Тип данных	Применение	Описание
Value	xsd:string	Требуемое	Фактическое значение
Offset	xsd:string	Опциональное	Сдвиг, который добавляется к фактическому значению для образования масштабированной величины: Техническое значение = (значение + сдвиг) * множитель; если не присутствует, должно быть использовано соответствующее значение элемента параметр
Multiplier	xsd:string	Опциональное	Коэффициент масштабирования, на который умножается фактическое значение для образования масштабированной величины: Техническое значение = (значение + сдвиг) * множитель; если не присутствует,должно быть использовано соответствующее значение элемента параметра

G.4.5.6.2.5 Элемент defaultValue (значение по умолчанию)

Элемент defaultValue предназначен для поддержки значения по умолчанию параметра. Это значение устанавливает начальное значение параметра интерфейса типа функции, ассоциированной с параметром.

Опциональный субэлемент g_labels может предоставить многоязычные имена и/или дескриптивную информацию для этой величины. Сама величина предоставляется атрибутом value (значение) элемента defaultValue. Могут быть также установлены сдвиг и множитель.

Атрибуты элемента defaultValue должны соответствовать указанным в таблице G.30.

G.4.5.6.2.6 Элемент substituteValue (подстановочное значение)

Элемент substituteValue определяет специальное значение параметра, которое предоставляется для приложения устройства в некоторых рабочих состояниях устройства (например, в случае отказа устройства).

Опциональный субэлемент g_labels может предоставить многоязычные имена и/или дескриптивную информацию для этой величины, которая предоставляется атрибутом value (значение) элемента substituteValue. Могут быть также установлены сдвиг и множитель.

Атрибуты элемента substituteValue должны соответствовать указанным в таблице G.30.

G.4.5.6.2.7 Элемент allowedValues (допустимые значения)

Элемент allowedValues определяет список поддерживаемых значений и/или один или несколько диапазонов поддерживаемых значений для параметра.

Список поддерживаемых значений представлен в виде некоторого количества от ни одного до нескольких субэлементов value элемента allowedValues, а диапазоны также представлены в виде некоторого количества от ни одного до нескольких субэлементов range (диапазон) элемента allowedValues.

Значение субэлемента устанавливает одно допустимое значение параметра. Опциональный субэлемент g_labels может предоставить многоязычные имена и/или дескриптивную информацию для этой величины, которая предоставляется атрибутом value (значение) элемента value. Могут быть также установлены сдвиг и множитель.

Атрибуты элемента value должны соответствовать указанным в таблице G.30.

Субэлемент range содержит два требуемых субэлемента, а именно элемент minValue (минимальное значение) и элемент maxValue (максимальное значение), которые определяют пределы диапазона допустимых значений. Элементы minValue и maxValue имеют такую же структуру и атрибуты, как субэлемент value элемента allowedValues. Следовательно описание субэлемента value и таблица G.30 действительны

также для этих субэлементов.

G.4.5.6.2.8 Элемент unit (единица)

Элемент unit определяет технические единицы параметра (например время, температуру, давление, поток, ускорение, ток, энергию), согласно ИСО 1000. Опциональный элемент g_labels может предоставить многоязычные имена и/или описательную информацию для технических единиц.

Атрибуты элемента unit должны соответствовать указанным в таблице G.31.

Таблица G.31 — Атрибуты элемента unit (единица)

Атрибут	Тип данных	Применение	Описание
Multiplier	xsd:string	Требуемое	Множитель для технических единиц аналоговых параметров
unitURI	xsd:anyURI	Опциональное	Связь с соответствующим определением единицы в файле, содержащем технические единицы (например время, температуру, давление, поток, ускорение, ток, энергию) согласно ИСО 1000

G.4.5.6.2.9 Элемент variableRef (ссылка на переменную)

Элемент variableRef создает ссылку на параметр интерфейса экземпляра функции или, если параметр представляет собой таблицу или структуру, возможна ссылка на элемент параметра (элемент таблицы или компонент структуры).

В иерархически структурированном элементе ApplicationProcess (прикладной процесс) экземпляры функции могут быть расположены внутри экземпляров функции других типов функций. Следовательно конкретный экземпляр на дереве функций может быть доступен только путем последовательных шагов по дереву, т. е. конкретный экземпляр должен получить адрес путем конкатенации имен экземпляров. Для отображения этой конкатенации и создания возможности ссылки на элемент структуры

variableRef (ссылка на переменную) содержит следующее:

– последовательность одного или нескольких элементов instanceIDRef (ссылка на идентификатор экземпляра), за которой следует

– один обязательный элемент variableIDRef (ссылка на идентификатор переменной) и

– опциональный элемент memberRef (ссылка на элемент).

Элемент variableRef имеет атрибут, указанный в таблице G.32.

Таблица G.32 — Атрибут элемента variableRef (ссылка на переменную)

Атрибут	Тип данных	Применение	Описание
Position	xsd: unsignedByte	По умолчанию	Определяет последовательность нескольких отображенных объектов данных в одном объекте параметра; позиция = 1 означает начало отображения в самом низком положении бита; число битов определяется типом данных объекта данных; последовательные объекты данных упаковываются без зазоров; значение по умолчанию – 1 (см. примечание)
Примечание — В случае одного отображенного объекта данных атрибут может быть пропущен.			

G.4.5.6.2.10 Элемент instanceIDRef (ссылка на идентификатор экземпляра)

Элемент instanceIDRef предназначен для ссылки на экземпляр функции внутри элемента functionInstanceList (список экземпляров функции), который может находиться либо на уровне элемента ApplicationProcess (прикладной процесс), либо на уровне элемента functionType (тип функции).

Ссылка типа xsd:IDREF предоставлена в виде атрибута элемента instanceIDRef согласно таблице G.33.

Таблица G.33 — Атрибуты элемента instanceIDRef (ссылка на идентификатор экземпляра)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного экземпляра функции

G.4.5.6.2.11 Элемент variableIDRef (ссылка на идентификатор переменной)

Элемент variableIDRef предназначен для ссылки на параметр интерфейса типа функции внутри элемента functionTypeList (список типов функций).

В данном элементе variableRef (ссылка на переменную) экземпляр этого типа функции определяется элементом functionInstance (экземпляр функции), на который производится ссылка элементом instanceIDRef (ссылка на идентификатор экземпляра), который непосредственно предшествует элементу variableIDRef.

Ссылка типа xsd:IDREF предоставляется в виде атрибута элемента variableIDRef, согласно таблице G.34.

Таблица G.34 — Атрибуты элемента variableIDRef (ссылка на идентификатор переменной)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного параметра интерфейса типа функции

G.4.5.6.2.12 Элемент memberRef (ссылка на элемент)

Опциональный элемент memberRef ссылается либо на соответствующую компоненту параметра интерфейса типа структурированных данных (используется атрибут uniqueIDRef (ссылка на уникальный идентификатор)), либо на соответствующий элемент таблицы параметра интерфейса типа табличных данных (используется атрибут index (индекс)). Один из этих атрибутов должен присутствовать, если имеется элемент memberRef.

Элемент memberRef должен содержать атрибуты, указанные в таблице G.35.

Таблица G.35 — Атрибуты элемента memberRef (ссылка на элемент)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Опциональное	Уникальный ID ссылочного компонента типа структурированных данных
Index	xsd:long	Опциональное	Индекс ссылочного элемента таблицы

G.4.5.6.3 Элемент property (свойство)

Элемент property вводится в качестве общего элемента для создания возможности включения значений для дополнительных специальных свойств в описание параметра.

Элемент property должен содержать атрибуты, указанные в таблице G.36.

Таблица G.36 — Атрибуты элемента property (свойство)

Атрибут	Тип данных	Применение	Описание
Name	xsd:string	Требуемое	Имя свойства
Value	xsd:string	Требуемое	Значение свойства

G.4.5.7 Элемент parameterGroupList (список группы параметров)

G.4.5.7.1 Общие положения

Опциональный элемент parameterGroupList, если он присутствует, содержит последовательность одного или нескольких элементов parameterGroup (группа параметров) согласно рисунку G.8. Многоязычные имена и/или описания групп параметров предоставляются с помощью группы g_labels. Описание группы g_labels приведено в G.2.2.2.

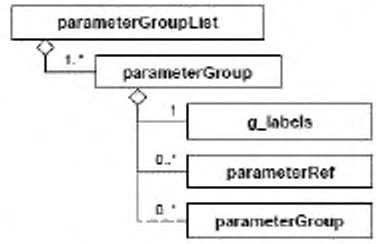


Рисунок G.8 — parameterGroupList (список группы параметров)

G.4.5.7.2 Элемент parameterGroup (группа параметров)

Каждый из элементов parameterGroup объединяет набор параметров элемента parameterList (список параметров) для построения группы параметров, предназначенных для специальной цели, например подготовки видов HMI. Эта цель указывается значением атрибута kindOfAccess (вид доступа) элемента parameterGroup. Возможно определение иерархии групп параметров.

На соответствующие параметры в наборе производится ссылка с помощью соответствующего номера элементов parameterRef (ссылка на переменную).

Элемент parameterGroup содержит атрибуты, указанные в таблице G.37.

Таблица G.37 — Атрибуты элемента parameterGroup (группа параметров)

Атрибут	Тип данных	Применение	Описание
uniqueID	xsd:ID	Требуемое	Уникальный ID группы параметров
kindOfAccess	xsd:string	Опциональное	Классифицирует параметры группы параметров

G.4.5.7.3 Элемент parameterRef (ссылка на переменную)

Элемент parameterRef предназначен для ссылки на элемент parameter (параметр) внутри элемента parameterList (список параметров), относящегося к элементу ApplicationProcess (прикладной процесс).

Ссылка типа xsd:IDREF предоставляется в качестве атрибута элемента parameterRef (ссылка на переменную) согласно таблице G.38.

Таблица G.38 — Атрибуты элемента parameterRef (ссылка на переменную)

Атрибут	Тип данных	Применение	Описание
uniqueIDRef	xsd:IDREF	Требуемое	Уникальный ID ссылочного параметра

G.4.6 Схемы шаблона профиля устройства EPL

G.4.6.1 XML схема: ISO15745ProfileContainer.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    >
<xsd:element name="ISO15745ProfileContainer">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="ISO15745Profile" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ISO15745Profile">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ProfileHeader" type="ProfileHeader_DataType"/>
<xsd:element name="ProfileBody" type="ProfileBody_DataType"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:annotation>
<xsd:documentation>* HEADER SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileHeader_DataType">
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
<xsd:element name="ProfileName" type="xsd:string"/>
<xsd:element name="ProfileSource" type="xsd:string"/>
<xsd:element name="ProfileClassID" type="ProfileClassID_DataType"/>
<xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
<xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
<xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
<xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:annotation>
<xsd:documentation>* BODY SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileBody_DataType" abstract="true"/>
<xsd:annotation>
```

```
<xsd:documentation>* HEADER DATA TYPES *</xsd:documentation>
</xsd:annotation>
<xsd:simpleType name="ProfileClassID_DataType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="AIP"/>
<xsd:enumeration value="Process"/>
<xsd:enumeration value="InformationExchange"/>
<xsd:enumeration value="Resource"/>
<xsd:enumeration value="Device"/>
<xsd:enumeration value="CommunicationNetwork"/>
<xsd:enumeration value="Equipment"/>
<xsd:enumeration value="Human"/>
<xsd:enumeration value="Material"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ISO15745Reference_DataType">
<xsd:sequence>
<xsd:element name="ISO15745Part" type="xsd:positiveInteger"/>
<xsd:element name="ISO15745Edition" type="xsd:positiveInteger"/>
<xsd:element name="ProfileTechnology" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="IASInterface_DataType">
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="CSI"/>
<xsd:enumeration value="HCI"/>
<xsd:enumeration value="ISI"/>
<xsd:enumeration value="API"/>
<xsd:enumeration value="CMI"/>
<xsd:enumeration value="ESI"/>
<xsd:enumeration value="FSI"/>
<xsd:enumeration value="MTI"/>
<xsd:enumeration value="SEI"/>
<xsd:enumeration value="USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:length value="4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:annotation>
<xsd:documentation>* ISO 15745 DEFINED DATA TYPES *</xsd:documentation>
</xsd:annotation>
<xsd:complexType name="ProfileHandle_DataType">
<xsd:sequence>
<xsd:element name="ProfileIdentification" type="xsd:string"/>
<xsd:element name="ProfileRevision" type="xsd:string"/>
```

```
<xsd:element name="ProfileLocation" type="xsd:anyURI" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

G.4.6.2 XML схема: CommonElements.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                                    ">
<!--################ common attribute group-->
<xsd:attributeGroup name="ag_formatAndFile">
<xsd:attribute name="formatName" type="xsd:string" fixed="DDXML" form="unqualified"/>
<xsd:attribute name="formatVersion" type="xsd:string" fixed="2.0" form="unqualified"/>
<xsd:attribute name="fileName" type="xsd:string" use="required" form="unqualified"/>
<xsd:attribute name="fileCreator" type="xsd:string" use="required" form="unqualified"/>
<xsd:attribute name="fileCreationDate" type="xsd:date" use="required"
form="unqualified"/>
<xsd:attribute name="fileCreationTime" type="xsd:time" use="optional"/>
<xsd:attribute name="fileModificationDate" type="xsd:date" use="optional"
form="unqualified"/>
<xsd:attribute name="fileModificationTime" type="xsd:time" use="optional"/>
<xsd:attribute name="fileModifiedBy" type="xsd:string" use="optional"/>
<xsd:attribute name="fileVersion" type="xsd:string" use="required" form="unqualified"/>
</xsd:attributeGroup>
<!--################ common groups-->
<xsd:group name="g_labels">
<xsd:sequence>
<xsd:choice maxOccurs="unbounded">
<xsd:element name="label">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="lang" type="xsd:language" use="required"/>
<xsd:attribute name="URI" type="xsd:anyURI" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="description">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="lang" type="xsd:language" use="required"/>
<xsd:attribute name="URI" type="xsd:anyURI" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="labelRef">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:anyURI">
<xsd:attribute name="dictID" type="xsd:IDREF" use="required"/>
```

```
<xsd:attribute name="textID" type="xsd:string" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
</xsd:choice>
</xsd:sequence>
</xsd:group>
<xsd:group name="g_simple">
<xsd:choice>
<xsd:element name="BOOL"/>
<xsd:element name="BITSTRING"/>
<xsd:element name="BYTE"/>
<xsd:element name="CHAR"/>
<xsd:element name="WORD"/>
<xsd:element name="DWORD"/>
<xsd:element name="LWORD"/>
<xsd:element name="SINT"/>
<xsd:element name="INT"/>
<xsd:element name="DINT"/>
<xsd:element name="LINT"/>
<xsd:element name="USINT"/>
<xsd:element name="UINT"/>
<xsd:element name="UDINT"/>
<xsd:element name="ULINT"/>
<xsd:element name="REAL"/>
<xsd:element name="LREAL"/>
<xsd:element name="TIME"/>
<xsd:element name="DATE"/>
<xsd:element name="DT"/>
<xsd:element name="TOD"/>
<xsd:element name="STRING"/>
<xsd:element name="WSTRING"/>
</xsd:choice>
</xsd:group>
<!--################ common elements-->
<xsd:element name="vendorID">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="deviceFamily">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="productID">
```

```
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="version">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="versionType" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="SW"/>
<xsd:enumeration value="FW"/>
<xsd:enumeration value="HW"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="buildDate" type="xsd:date"/>
<xsd:element name="specificationRevision">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

G.4.6.3 XML схема: ProfileBody_Device_EPL.xsd

XML схема ProfileBody_Device_EPL.xsd включает схему ISO15745ProfileContainer.xsd в G.4.6.1 и схему CommonElements.xsd in G.4.6.2.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    ">
<xsd:include schemaLocation="ISO15745ProfileContainer.xsd"/>
<xsd:include schemaLocation="CommonElements.xsd"/>
<!--################ profile body device -->
<xsd:complexType name="ProfileBody_Device_EPL">
<xsd:complexContent>
<xsd:extension base="ProfileBody_DataType">
<xsd:sequence>
```

```
<xsd:element ref="DeviceIdentity" minOccurs="0"/>
<xsd:element ref="DeviceManager" minOccurs="0"/>
<xsd:element ref="DeviceFunction" maxOccurs="unbounded"/>
<xsd:element ref="ApplicationProcess" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element name="ExternalProfileHandle" type="ProfileHandle_DataType"
minOccurs="0"
maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attributeGroup ref="ag_formatAndFile"/>
<xsd:attribute name="supportedLanguages" type="xsd:NMTOKENS" use="optional"/>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<!--################ device identity elements -->
<xsd:element name="DeviceIdentity">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="vendorName"/>
<xsd:element ref="vendorID" minOccurs="0"/>
<xsd:element ref="vendorText" minOccurs="0"/>
<xsd:element ref="deviceFamily" minOccurs="0"/>
<xsd:element ref="productFamily" minOccurs="0"/>
<xsd:element ref="productName"/>
<xsd:element ref="productID" minOccurs="0"/>
<xsd:element ref="productText" minOccurs="0"/>
<xsd:element ref="orderNumber" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="version" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="buildDate" minOccurs="0"/>
<xsd:element ref="specificationRevision" minOccurs="0"/>
<xsd:element ref="instanceName" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="productFamily">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="instanceName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="false"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="orderNumber">
```

```
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="productName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="productText">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="vendorName">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="vendorText">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="readOnly" type="xsd:boolean" default="true"/>
</xsd:complexType>
</xsd:element>
<!--################ device manager elements -->
<xsd:element name="DeviceManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="indicatorList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="indicatorList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="LEDList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="LEDList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="LED" maxOccurs="unbounded"/>
<xsd:element ref="combinedState" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="LED">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element ref="LEDstate" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="LEDcolors" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="monocolor"/>
<xsd:enumeration value="bicolor"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="LEDtype" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="IO"/>
<xsd:enumeration value="device"/>
<xsd:enumeration value="communication"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="LEDstate">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="state" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="on"/>
<xsd:enumeration value="off"/>
<xsd:enumeration value="flashing"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="LEDcolor" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="green"/>
<xsd:enumeration value="amber"/>
```

```
<xsd:enumeration value="red"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="flashingPeriod" type="xsd:unsignedInt" use="optional"/>
<xsd:attribute name="impulsWidth" type="xsd:unsignedByte" default="50"/>
<xsd:attribute name="numberOfImpulses" type="xsd:unsignedByte" default="1"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="combinedState">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element name="LEDstateRef" minOccurs="2" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="stateIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!--################ device function elements -->
<xsd:element name="DeviceFunction">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="capabilities"/>
<xsd:element ref="picturesList" minOccurs="0"/>
<xsd:element ref="dictionaryList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="capabilities">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="characteristicsList" maxOccurs="unbounded"/>
<xsd:element ref="standardComplianceList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristicsList">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="category" minOccurs="0">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element ref="characteristic" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristic">
```

```
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="characteristicName"/>
<xsd:element ref="characteristicContent" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristicContent">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="characteristicName">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="standardComplianceList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="compliantWith" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="compliantWith">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="range" default="international">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="international"/>
<xsd:enumeration value="internal"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="picturesList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="picture" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="picture">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="URI" type="xsd:anyURI" use="required"/>
<xsd:attribute name="number" type="xsd:unsignedInt" use="optional"/>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="dictionaryList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="dictionary" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dictionary">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="file" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="lang" type="xsd:language" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="file">
<xsd:complexType>
<xsd:attribute name="URI" type="xsd:anyURI" use="required"/>
</xsd:complexType>
</xsd:element>
<!--################ application process elements -->
<xsd:element name="ApplicationProcess">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="dataTypeList" minOccurs="0"/>
<xsd:element ref="functionTypeList"/>
<xsd:element ref="functionInstanceList"/>
<xsd:element ref="parameterList"/>
<xsd:element ref="parameterGroupList" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dataTypeList">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_complex" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionTypeList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="functionType" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionType">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="versionInfo" maxOccurs="unbounded"/>
<xsd:element ref="interfaceList"/>
```

```
<xsd:element ref="functionInstanceList" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
<xsd:attribute name="package" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="versionInfo">
<xsd:complexType>
<xsd:attribute name="organization" type="xsd:string" use="required"/>
<xsd:attribute name="version" type="xsd:string" use="required"/>
<xsd:attribute name="author" type="xsd:string" use="required"/>
<xsd:attribute name="date" type="xsd:date" use="required"/>
<xsd:attribute name="remarks" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="interfaceList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="inputVars" minOccurs="0"/>
<xsd:element ref="outputVars" minOccurs="0"/>
<xsd:element ref="configVars" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="inputVars">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="outputVars">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="configVars">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="varDeclaration">
<xsd:complexType>
<xsd:choice>
<xsd:group ref="g_simple"/>
<xsd:element ref="dataTypeIDRef"/>
```

```
</xsd:choice>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="size" type="xsd:string" use="optional"/>
<xsd:attribute name="initialValue" type="xsd:string" use="optional"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="dataTypeIDRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionInstanceList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="functionInstance" maxOccurs="unbounded"/>
<xsd:element ref="connection" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionInstance">
<xsd:complexType>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="typeIDRef" type="xsd:IDREF" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="connection">
<xsd:complexType>
<xsd:attribute name="source" type="xsd:string" use="required"/>
<xsd:attribute name="destination" type="xsd:string" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="parameter" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameter">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element ref="variableRef" maxOccurs="unbounded"/>
<xsd:element ref="conditionalSupport" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="denotation" minOccurs="0"/>
<xsd:element ref="actualValue" minOccurs="0"/>
<xsd:element ref="defaultValue" minOccurs="0"/>
```

```
<xsd:element ref="substituteValue" minOccurs="0"/>
<xsd:element ref="allowedValues" minOccurs="0"/>
<xsd:element ref="unit" minOccurs="0"/>
<xsd:element ref="property" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attributeGroup ref="ag_parameter"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="variableRef">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="instanceIDRef" maxOccurs="unbounded"/>
<xsd:element ref="variableIDRef"/>
<xsd:element ref="memberRef" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="position" type="xsd:unsignedByte" default="1"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="instanceIDRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="variableIDRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="memberRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="optional"/>
<xsd:attribute name="index" type="xsd:long" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="actualValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="conditionalSupport">
<xsd:complexType>
<xsd:attribute name="paramIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="denotation">
<xsd:complexType>
<xsd:group ref="g_labels"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="defaultValue">
<xsd:complexType>
```

```
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="substituteValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="allowedValues">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="value" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="range" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="value">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="range">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="minValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="maxValue">
<xsd:complexType>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:attributeGroup ref="ag_value"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="unit">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="multiplier" type="xsd:string" use="required"/>
<xsd:attribute name="unitURI" type="xsd:anyURI" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="property">
<xsd:complexType>
<xsd:attribute name="name" type="xsd:string" use="required"/>
```

```
<xsd:attribute name="value" type="xsd:string" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterGroupList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="parameterGroup" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterGroup">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels"/>
<xsd:element ref="parameterGroup" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="parameterRef" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="kindOfAccess" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterRef">
<xsd:complexType>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<!--################ complex types -->
<xsd:element name="array">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="subrange" maxOccurs="unbounded"/>
<xsd:choice>
<xsd:group ref="g_simple"/>
<xsd:element ref="dataTypeIDRef"/>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="subrange">
<xsd:complexType>
<xsd:attribute name="lowerLimit" type="xsd:long" use="required"/>
<xsd:attribute name="upperLimit" type="xsd:long" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="struct">
<xsd:complexType>
<xsd:sequence>
 <xsd:element ref="count" minOccurs="0"/>
<xsd:element ref="varDeclaration" maxOccurs="unbounded"/>
```

```
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="enum">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="enumValue" maxOccurs="unbounded"/>
<xsd:group ref="g_simple" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="size" type="xsd:string" use="optional"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="enumValue">
<xsd:complexType>
<xsd:group ref="g_labels"/>
<xsd:attribute name="value" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="derived">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="count" minOccurs="0"/>
<xsd:choice>
<xsd:group ref="g_simple"/>
<xsd:element ref="dataTypeIDRef"/>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="description" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="count">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_labels" minOccurs="0"/>
<xsd:element ref="defaultValue"/>
<xsd:element ref="allowedValues" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="access" default="read">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="read"/>
<xsd:enumeration value="write"/>
<xsd:enumeration value="readWrite"/>
```

```
<xsd:enumeration value="noAccess"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<!--################ group-->
<xsd:group name="g_complex">
<xsd:choice>
<xsd:element ref="array"/>
<xsd:element ref="struct"/>
<xsd:element ref="enum"/>
<xsd:element ref="derived"/>
</xsd:choice>
</xsd:group>
<!--################ attribute groups-->
<xsd:attributeGroup name="ag_parameter">
<xsd:attribute name="uniqueID" type="xsd:ID" use="required"/>
<xsd:attribute name="access" default="read">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="read"/>
 <xsd:enumeration value="write"/>
<xsd:enumeration value="readWrite"/>
<xsd:enumeration value="noAccess"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="support" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="mandatory"/>
<xsd:enumeration value="optional"/>
<xsd:enumeration value="conditional"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="persistence" type="xsd:boolean" default="false"/>
<xsd:attribute name="offset" type="xsd:string" use="optional"/>
<xsd:attribute name="multiplier" type="xsd:string" use="optional"/>
</xsd:attributeGroup>
<xsd:attributeGroup name="ag_value">
<xsd:attribute name="value" type="xsd:string" use="required"/>
<xsd:attribute name="offset" type="xsd:string" use="optional"/>
<xsd:attribute name="multiplier" type="xsd:string" use="optional"/>
</xsd:attributeGroup>
 </xsd:schema>
```

G.5 Описание шаблона профиля коммуникационной сети

G.5.1 ProfileBody (тело профиля)

Профиль коммуникационной сети ProfileBody содержит ApplicationLayers (уровни приложения), TransportLayers (транспортные уровни) и элементы NetworkManagement (сетевой менеджмент), представленные на рисунке 37.

Элемент ProfileBody содержит атрибуты, указанные в таблице G.39.

Таблица G.39 — Атрибуты элемента ProfileBody (тело профиля)

Атрибут	Тип данных	Применение	Описание
formatName	xsd:string	Фиксированное	Идентификатор формата
formatVersion	xsd:string	Фиксированное	Идентификатор версии формата
fileName	xsd:string	Требуемое	Имя файла с расширением без пути
fileCreator	xsd:string	Требуемое	Лицо, разработавшее файл
fileCreationDate	xsd:date	Требуемое	Дата создания файла
fileCreationTime	xsd:time	Опциональное	Время создания файла
fileModifiedBy fileModificationDate	xsd:string xsd.date	Опциональное Опциональное	Лицо, изменившее файл Дата последнего изменения файла
fileModificationTime	xsd:time	Опциональное	Время последнего изменения файла
fileVersion	xsd:string	Требуемое	Установленная поставщиком версия файла
supportedLanguages	xsd:NMTOKENS	Опциональное	Список поддерживаемых языков

G.5.2 ApplicationLayers (уровни приложения)

G.5.2.1 Общие положения

На рисунке 37 представлена структура класса ETHERNET Powerlink ApplicationLayers.

G.5.2.2 CANopenObjectList

G.5.2.2.1 Общие положения

На рисунке 37 представлена структура элемента CANopenObjectList, который содержит от одного до нескольких элементов CANopenObject.

Примечание — В ETHERNET Powerlink принята терминология и структура объекта CANopen (EN 50325-4) (CANopen поверх ETHERNET Powerlink). Следовательно здесь используются термины CANopen.

G.5.2.2.2 CANopenObject

G.5.2.2.2.1 Общие положения

На рисунке 37 представлена структура элемента CANopenObject, который содержит от нуля до нескольких элементов CANopenSubObject. Элементы CANopenObject и CANopenSubObject отображают функциональную часть профиля устройства ETHERNET Powerlink в профиле коммуникационной сети CANopen поверх ETHERNET Powerlink.

Элемент CANopenObject содержит атрибуты, указанные в таблице G.40.

Таблица G.40 — Атрибуты элемента CANopenObject

Атрибут	Тип данных	Применение	Описание
Index	xsd:hexBinary	Требуемое	Индекс объекта (четыре hex разряда)
Name	xsd:string	Требуемое	Имя объекта
ObjectType	xsd:unsignedByte	Требуемое	Тип объекта CANopen
DataType	xsd:hexBinary	Опциональное	Тип данных CANopen (два hex разряда)
LowLimit	xsd:string	Опциональное	Нижнее предельное значение параметра
HighLimit	xsd:string	Опциональное	Верхнее предельное значение параметра
AccessType	xsd:string	Опциональное	Тип доступа объекта; правильными значениями являются следующие: - ro – доступ только для чтения; - wo - доступ только для записи; - rw – доступ для записи и чтения; - rwr - доступ для записи и чтения; предпочтительным является доступ для чтения; - rww - доступ для записи и чтения; предпочтительным является доступ для записи; - const - доступ только для чтения; значения не изменяются
DefaultValue	xsd:string	Опциональное	Значение объекта по умолчанию

Окончание таблицы G.40

Атрибут	Тип данных	Применение	Описание
PdoMapping	xsd:boolean	Опциональное	PDO отображение объекта; правильными значениями являются следующие: - истина – отображено; - ложь – не отображено
ObjFlags	xsd:hexBinary	Опциональное	Контроль поведения инструментов (четыре hex разряда)
UniqueIDRef	xsd:IDREF	Опциональное	Уникальный ID соответствующего элемента в части процесса приложения, на который имеется ссылка из этого объекта. Если атрибуту приданы атрибуты dataType, lowLimit, highLimit, accessType, a defaultValue устанавливается ссылочным элементом из части процесса приложения
SubNumber	xsd:unsigned Byte	Опциональное	Число субобъектов объекта

G.5.2.2.2.2 CANopenSubObject

Элемент CANopenSubObject имеет пустой контент и содержит атрибуты, указанные в таблице G.41.

Таблица G.41 — Атрибуты элемента CANopenSubObject

Атрибут	Тип данных	Применение	Описание
subIndex	xsd:hexBinary	Требуемое	Субиндекс объекта (два hex разряда)
Name	xsd:string	Требуемое	Имя объекта
ObjectType	xsd:unsignedByte	Требуемое	Тип объекта CANopen
DataType	xsd:hexBinary	Опциональное	Тип данных CANopen (два hex разряда)

Окончание таблицы G.41

Атрибут	Тип данных	Применение	Описание
LowLimit	xsd:string	Опциональное	Нижнее предельное значение параметра
HighLimit	xsd:string	Опциональное	Верхнее предельное значение параметра
AccessType	xsd:string	Опциональное	Тип доступа объекта; правильными значенияями являются следующие: - ro – доступ только для чтения; - wo - доступ только для записи; - rw – доступ для записи и чтения; - rwr - доступ для записи и чтения; предпочтительнум является доступ для чтения; - rww - доступ для записи и чтения; предпочтительным является доступ для записи; - const – доступ только для чтения; значения не изменяются
DefaultValue	xsd:string	Опциональное	Значение объекта по умолчанию
PdoMapping	xsd:boolean	Опциональное	PDO отображение объекта; правильными значениями являются следующие: - истина – отображено; - ложь – не отображено
ObjFlags	xsd:hexBinary	Опциональное	Контроль поведения инструментов (четыре hex разряда)
UniqueIDRef	xsd:IDREF	Опциональное	Уникальный ID соответствующего элемента в части процесса приложения, на которую имеется ссылка из этого объекта. Если атрибуту приданы атрибуты dataType, lowLimit, highLimit, accessType, а defaultValue устанавливается ссылочным элементом процесса приложения

G.5.2.3 Identity (идентичность)

Поскольку различные профили коммуникаций могут требовать различную информацию об идентичности, может быть использован опциональный элемент локальной идентичности внутри элемента ApplicationLayers (уровни приложения). Этот элемент identity может содержать подмножество субэлементов элемента DeviceIdentity (идентичность устройства), описанного в G.4.2. Все описания субэлементов, приведенные в данном подразделе применяются также к субэлементам данного элемента identity.

G.5.2.4 Элемент dummyUsage (использование макета)

G.5.2.4.1 Общие положения

На рисунке 37 представлена структура элемента dummyUsage, который содержит от ни одного до нескольких элементов dummy (макет).

G.5.2.4.2 Элемент dummy (макет)

Элемент dummy не имеет контента. Этот элемент используется для обеспечения возможности включения и выключения некоторых входов макета для построения отображения макета.

Элемент dummy содержит атрибуты, указанные в таблице G.42.

Таблица G.42 — Атрибуты элемента dummy (макет)

Атрибут	Тип данных	Применение	Описание
Entry	xsd:string	Требуемое	Строка строится с использованием имени объекта макета, за которым следует знак равенства и затем – значение либо 0 для отключения отображения или 1 для включения отображения. Допустимыми значениями являются следующие: - dummy0001=0; - dummy0002=0; - dummy0003=0; - dummy0004=0; - dummy0005=0; - dummy0006=0; - dummy0007=0; - dummy0001 = 1; - dummy0002=1; - dummy0003=1; - dummy0004=1; - dummy0005=1; - dummy0006=1; - dummy0007=1

G.5.2.5 Элемент dynamicChannels (динамические каналы)

G.5.2.5.1 Общие положения

На рисунке 37 представлена структура элемента dynamicChannels. Этот элемент содержит один или несколько элементов dynamicChannel (динамический канал).

G.5.2.5.2 Элемент dynamicChannel (динамический канал)

Элемент dynamicChannel содержит элемент, описывающий простой тип данных группы g_simple. Этот элемент используется для маркировки

доступных каналов, которые могут быть использованы для создания связи между передаваемыми по сети ETHERNET Powerlink данными и программой приложения, выполняемой в устройстве.

Элемент dynamicChannel содержит атрибуты, указанные в таблице G.43.

Таблица G.43 — Атрибуты элемента dynamicChannel (динамический канал)

Атрибут	Тип данных	Применение	Описание
AccessType	xsd:NMToken	Требуемое	Тип доступа объекта; правильными значениями являются следующие: - только чтение – доступ только для чтения - только запись - доступ только для записи - чтение запись запись – доступ для чтения и записи; предпочтительным является доступ для записи
StartIndex	xsd:hexBinary	Требуемое	Начальный индекс объекта
EndIndex	xsd:hexBinary	Требуемое	Конечный индекс объекта
MaxNumber	xsd:unsignedByte	Требуемое	Максимальное число связей с программой приложения
AddressOffset	xsd:hexBinary	Требуемое	Сдвиг адреса в памяти программы приложения
BitAlignment	xsd:unsignedByte	Опционально	Выравнивание битов данных внутри объекта, счет которого начинают с младшего бита.
ManufacturerSpecific	xsd:string	Опционально	Предоставляется для специального использования изготовителем

G.5.3 TransportLayers (транспортные уровни)

Элемент TransportLayers не имеет контента.

G.5.4 NetworkManagement (сетевой менеджмент)

G.5.4.1 Общие положения

На рисунке 37 представлена структура класса ETHERNET Powerlink NetworkManagement (сетевой менеджмент).

G.5.4.2 Элемент PowerlinkGeneralFeatures (общие характеристики Powerlink)

Элемент PowerlinkgeneralFeatures имеет пустой контент и содержит атрибуты, указанные в таблице G.44

Таблица G.44 — Атрибуты элемента PowerlinkGeneralFeatures (общие характеристики Powerlink)

Атрибут	Тип данных	Применение	Описание
SdoCommunication	xsd:NMTokens	Требуемое	SDO роль при коммуникациях. Допустимые значения: — sdo_client(клиент); — sdo_server(сервер)
SdoCommands	xsd:NMTokens	Требуемое	SDO команды. Допустимые значения: – writeByIndex (запись по индексу); – readByIndex (чтение по индексу); – writeAllByIndex (запись все по индексу); – readAllByIndex (чтение все по индексу); – writeByName (запись по имени); – readByName (чтение по имени); – fileWrite (запись файла); – fileRead (чтение файла); – writeMultipleParam (запись множественных параметров); – readMultipleParam (чтение множественных параметров); – linkName (имя связи)
SdoClientTimeout	xsd:double	Опционально	Таймаут SDO клиента, значение в [наносекундах]. При отсутствии атрибута значение по умолчанию - 1 000 000.

Продолжение таблицы G.44

Атрибут	Тип данных	Применение	Описание
SdoServerTimeout	xsd:double	Опционально	Таймаут SDO сервера, значение в [наносекундах]. При отсутствии атрибута значение по умолчанию - 1 000 000.
SdoMaxParallelConnections	xsd:unsignedInt	Требуемое	Максимальное число параллельных соединений SDO. Минимальное значение - 1.
PdoMapGranularity	xsd:unsignedByte	Опционально	Уровень модульности отображения PDO, значение в [байтах]. При отсутствии атрибута значение по умолчанию - 8.
emergencyStatusEntries	xsd:unsignedByte	Требуемое	Число входов ошибок/событий сигнализируются Узлом по сети. Минимальное значение - 2 и максимальное значение - 14.
EmergencyQueueSize	xsd:unsignedByte	Опционально	Число входов ошибок/событий в аварийной очереди Узла. При отсутствии атрибута значение по умолчанию – 0
DCFsupport	xsd:NMTokens	Опционально	Способность управляющего узла поддерживать обработку файла конфигурации устройства (DCF). Допустимые значения: — поддержка краткого DCF — поддержка DCF — нет поддержки DCF
RoutingCapability	xsd:Boolean	Опционально	Способность выполнять функции маршрутизации. При отсутствии атрибута значение по умолчанию – ложь
MultiplexedCycles	xsd:Boolean	Опционально	Способность выполнять режим уплотненного доступа. При отсутствии атрибута значение по умолчанию – истина.

G.5.4.3 PowerlinkMNFeatures (характеристики PowerlinkMN)

Элемент PowerlinkMNFeatures имеет пустой контент.

Элемент PowerlinkMNFeatures содержит атрибуты, указанные в таблице G.45.

Таблица G.45 — Атрибуты элемента PowerlinkMNFeatures (характеристики PowerlinkMN)

Атрибут	Тип данных	Применение	Описание
BootupMode	xsd:NMToken	Опциональное	Методология загрузки системы. Допустимые значения: — загрузка системы bootup_simple (простая) без проверки CN и задержанной загрузки опциональных CNs (компьютерных сетей) — загрузка системы bootup_individual (индивидуальная) с проверкой CN и задержанной загрузкой опциональных CNs (компьютерных сетей) При отсутствии атрибута значение по умолчанию - bootup_simple.
ConfigurationManager	xsd:Boolean	Опциональное	Способность выполнять функции Configuration Manager (менеджера конфигурации). При отсутствии атрибута значение по умолчанию - ложь.

Окончание таблицы G.45

Атрибут	Тип данных	Применение	Описание
SendPRes	xsd: boolean	Опциональное	Способность управляющих узлов передавать многоадресные фреймы PR. При отсутствии атрибута значение по умолчанию – ложь.
PdoMaxTxChannels	unsignedByte	Требуемое	Максимальное число каналов TX PDO. Значение по умолчанию - 1.
PdoMaxRxChannels	unsignedByte	Требуемое	Максимальное число каналов RX PDO Значение по умолчанию - 1.
SdoTransferTypePdo	xsd:boolean	Опциональное	Способность передавать данные SDO интегрированные в фреймы PDO. При отсутствии атрибута значение по умолчанию – ложь.

G.5.4.4 PowerlinkCNFeatures (характеристики PowerlinkCN)

Элемент PowerlinkCNFeatures имеет пустой контент.

Элемент PowerlinkCNFeatures содержит атрибуты, указанные в таблице G.46.

Таблица G.46 — Атрибуты элемента PowerlinkCNFeatures (характеристики PowerlinkCN)

Атрибут	Тип данных	Применение	Описание
BasicEthernetMode	xsd:boolean	Опциональное	Поддержка базового Ethernet режима. При отсутствии атрибута значение по умолчанию – истина
IsoCommFeature	xsd:boolean	Требуемое	Поддержка изохронного траффика сети. Значение по умолчанию – истина.
SdoTransferType	xsd:NMTokens	Требуемое	Поддержка методологии передачи SDO. Допустимые значения — sdo_udp SDO хост – во фреймах UDP/IP; — sdo_asend SDO хост – во фреймах EPL ASnd ; — sdo_pdo SDO хост – в контейнере данных, интегрированном в PDO
PdoMaxRxChannels	xsd:unsignedByte	Опциональное	Максимальное число каналов RX PDO При отсутствии атрибута значение по умолчанию - 0

G.5.4.5 Элемент deviceCommissioning (ввод в эксплуатацию устройства)

Элемент deviceCommissioning имеет пустой контент и содержит атрибуты, указанные в таблице G.47.

Таблица G.47 — Атрибуты элемента deviceCommissioning (ввод в эксплуатацию устройства)

Атрибут	Тип данных	Применение	Описание
NodeID	xsd: unsignedByte	Требуемое	ID узла (адрес). Правильные идентификаторы узлов для CN: 1 - 239 и 253 - 254. Правильный идентификатор узла для MN - 240.
NodeType	xsd:NMTOKENS	Требуемое	Способность выполнять MN соответствует функциям CN. Допустимые значения: – CN; – MN; – резервный; – MN. Допускается любая комбинация этих значений.
PdoMaxMapBytes	xsd:unsignedInt	Опциональное	Максимальный размер отображения PDO. Значение предоставляется в байтах. Минимальное значение равно 0 максимальное значение равно 1489. При отсутствии атрибута по умолчанию - 1489.
PdoMaxMapObjects	xsd:byte	Опциональное	Максимальное число отображаемых объектов на канале PDO. При отсутствии атрибута по умолчанию - 255.

Продолжение таблицы G.47

Атрибут	Тип данных	Применение	Описание
MaxSupportedNodes	xsd:unsignedByte	Опционально	Максимальное число узлов в сегменте сети Powerlink. При отсутствии атрибута по умолчанию - 240.
UsedNetworkInterface	xsd:unsignedByte	Опционально	Число активных интерфейсов сети. При отсутствии атрибута по умолчанию - 0.
MaxHeartbeatNodes	xsd:unsignedByte	Опционально	Число узлов, контролируемых по такту. При отсутствии атрибута по умолчанию - 240. Этот параметр действителен только в том случае, если устройство поддерживает тактовый механизм в целом
CycleTimingMinSupport	xsd:double	Требуемое	Минимальное время цикла. Значение указывается в [микросекундах]. По умолчанию - 0
CycleTimingMaxSupport	xsd:double	Требуемое	Максимальное время цикла. Значение указывается в [микросекундах]. Все значения большее или равные cycleTimingMinSupport поддерживаются.
CycleTimingGranularity	xsd:double	Опционально	Уровень модульности времени цикла.
			Значение указывается в [микросекундах]. При отсутствии атрибута по умолчанию - 1 мкс.
MaxDomainSize	xsd:double	Требуемое	Значение по умолчанию отсутствует.

G.5.5 Схема шаблона профиля коммуникационной сети EPL

XML схема ProfileBody_CommunicationNetwork_EPL.xsd включает схему ISO15745ProfileContainer.xsd в G.4.6.1 и схему CommonElements.xsd в G.4.6.2.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                                    ">
<xsd:include schemaLocation="ISO15745ProfileContainer.xsd"/>
<xsd:include schemaLocation="CommonElements.xsd"/>
<!-- profile body -->
<xsd:complexType name="ProfileBody_CommunicationNetwork_EPL">
<xsd:complexContent>
<xsd:extension base="ProfileBody_DataType">
<xsd:choice>
<xsd:sequence>
<xsd:element name="ApplicationLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="CANopenObjectList"/>
<xsd:element name="identity" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="vendorID" minOccurs="0"/>
<xsd:element ref="deviceFamily" minOccurs="0"/>
<xsd:element ref="productID" minOccurs="0"/>
<xsd:element ref="version" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="buildDate" minOccurs="0"/>
<xsd:element ref="specificationRevision" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dummyUsage" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="dummy" maxOccurs="unbounded">
<xsd:complexType>
<xsd:attribute name="entry" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="Dummy0001=0"/>
<xsd:enumeration value="Dummy0002=0"/>
<xsd:enumeration value="Dummy0003=0"/>
<xsd:enumeration value="Dummy0004=0"/>
<xsd:enumeration value="Dummy0005=0"/>
<xsd:enumeration value="Dummy0006=0"/>
<xsd:enumeration value="Dummy0007=0"/>
<xsd:enumeration value="Dummy0001=1"/>
<xsd:enumeration value="Dummy0002=1"/>
<xsd:enumeration value="Dummy0003=1"/>
<xsd:enumeration value="Dummy0004=1"/>
<xsd:enumeration value="Dummy0005=1"/>
<xsd:enumeration value="Dummy0006=1"/>
```

```
<xsd:enumeration value="Dummy0007=1"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="dynamicChannels" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="dynamicChannel" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref="g_simple"/>
</xsd:sequence>
<xsd:attribute name="accessType" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="readOnly"/>
<xsd:enumeration value="writeOnly"/>
<xsd:enumeration value="readWriteWrite"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="startIndex" type="xsd:hexBinary" use="required"/>
<xsd:attribute name="endIndex" type="xsd:hexBinary" use="required"/>
<xsd:attribute name="maxNumber" type="xsd:unsignedInt" use="required"/>
<xsd:attribute name="addressOffset" type="xsd:hexBinary" use="required"/>
<xsd:attribute name="bitAlignment" type="xsd:unsignedByte" use="optional"/>
<xsd:attribute name="manufacturerSpecific" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TransportLayers">
<xsd:complexType/>
</xsd:element>
<xsd:element name="NetworkManagement" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="PowerlinkGeneralFeatures">
<xsd:complexType>
<xsd:attribute name="sdoCommunication" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKENS">
<xsd:enumeration value="sdo_client"/>
```

```
<xsd:enumeration value="sdo_server"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="sdoCommands" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKENS">
<xsd:enumeration value="writeByIndex"/>
<xsd:enumeration value="readByIndex"/>
<xsd:enumeration value="writeAllByIndex"/>
<xsd:enumeration value="readAllByIndex"/>
<xsd:enumeration value="writeByName"/>
<xsd:enumeration value="readByName"/>
<xsd:enumeration value="fileWrite"/>
<xsd:enumeration value="fileRead"/>
<xsd:enumeration value="writeMultipleParam"/>
<xsd:enumeration value="readMultipleParam"/>
<xsd:enumeration value="linkName"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="sdoClientTimeout" type="xsd:double" use="optional"
default="1000000"/>
<xsd:attribute name="sdoServerTimeout" type="xsd:double" use="optional"
default="1000000"/>
<xsd:attribute name="sdoMaxParallelConnections" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedInt">
<xsd:minInclusive value="1"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="pdoMapGranularity" type="xsd:unsignedByte" use="optional"
default="8"/>
<xsd:attribute name="parameterStorage" type="xsd:boolean" use="optional"
default="false"/>
<xsd:attribute name="emergencyStatusEntries" default="2">
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedByte">
<xsd:minInclusive value="2"/>
<xsd:maxInclusive value="14"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="emergencyQueueSize" type="xsd:unsignedByte" use="optional"/>
<xsd:attribute name="DCFsupport" use="optional" default="supportConciseDCF">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKENS">
<xsd:enumeration value="supportConciseDCF"/>
<xsd:enumeration value="supportDCF"/>
</xsd:restriction>
</xsd:simpleType>
```

```
</xsd:attribute>
<xsd:attribute name="routingCapability" type="xsd:boolean" use="optional"
default="false"/>
<xsd:attribute name="multiplexedCycles" type="xsd:boolean" use="optional"
default="true"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="PowerlinkMNFeatures" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="bootupMode" use="optional" default="bootup_simple">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="bootup_simple"/>
<xsd:enumeration value="bootup_individual"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="configurationManager" type="xsd:boolean" use="optional"
default="false"/>
<xsd:attribute name="sendPRes" type="xsd:boolean" use="optional" default="false"/>
<xsd:attribute name="pdoMaxTxChannels" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="pdoMaxRxChannels" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedByte">
<xsd:minInclusive value="1"/>
<xsd:maxInclusive value="252"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="sdoTransferTypePDO" type="xsd:boolean" use="optional"
default="false"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="PowerlinkCNFeatures" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="basicEthernetMode" type="xsd:boolean" use="optional"
default="true"/>
<xsd:attribute name="isoCommFeature" type="xsd:boolean" use="required"/>
<xsd:attribute name="sdoTransferType" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKENS">
<xsd:enumeration value="sdo_udp"/>
<xsd:enumeration value="sdo_asend"/>
<xsd:enumeration value="sdo_pdo"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="pdoMaxRxChannels" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedByte">
<xsd:minExclusive value="0"/>
<xsd:maxExclusive value="253"/>
```

```
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="deviceCommissioning">
<xsd:complexType>
<xsd:attribute name="nodeID" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="nodeType" type="xsd:NMTOKENS" use="required"/>
<xsd:attribute name="pdoMaxMapBytes" use="optional" default="1489">
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedInt">
<xsd:minInclusive value="0"/>
<xsd:maxInclusive value="1489"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="pdoMaxMapObjects" type="xsd:unsignedByte" use="optional"
default="255"/>
<xsd:attribute name="maxSupportedNodes" type="xsd:unsignedByte" use="optional"
default="240"/>
<xsd:attribute name="usedNetworkInterface" type="xsd:unsignedByte" use="optional"
default="0"/>
<xsd:attribute name="maxHeartbeatNodes" type="xsd:unsignedByte" use="optional"
default="240"/>
<xsd:attribute name="cycleTimingMinSupport" type="xsd:double" use="required"/>
<xsd:attribute name="cycleTimingMaxSupport" type="xsd:double" use="required"/>
<xsd:attribute name="cycleTimingGranularity" type="xsd:double" use="optional"
default="1"/>
<xsd:attribute name="maxDomainSize" type="xsd:double" use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:element name="ExternalProfileHandle" type="ProfileHandle_DataType"/>
</xsd:choice>
<xsd:attributeGroup ref="ag_formatAndFile"/>
<xsd:attribute name="supportedLanguages" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKENS"/>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<!-- CANopen object dictionary-->
<xsd:element name="CANopenObjectList">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="CANopenObject" maxOccurs="65535">
```

```
<xsd:complexType>
<xsd:sequence>
<xsd:element name="CANopenSubObject" minOccurs="0" maxOccurs="255">
<xsd:complexType>
<xsd:attribute name="subIndex" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="objectType" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="dataType" type="xsd:hexBinary" use="optional"/>
<xsd:attribute name="lowLimit" type="xsd:string" use="optional"/>
<xsd:attribute name="highLimit" type="xsd:string" use="optional"/>
<xsd:attribute name="accessType" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="ro"/>
<xsd:enumeration value="wo"/>
<xsd:enumeration value="rw"/>
<xsd:enumeration value="rwr"/>
<xsd:enumeration value="rww"/>
<xsd:enumeration value="const"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="defaultValue" type="xsd:string" use="optional"/>
<xsd:attribute name="actualValue" type="xsd:string" use="optional"/>
<xsd:attribute name="denotation" type="xsd:string" use="optional"/>
<xsd:attribute name="PDOmapping" type="xsd:boolean" use="optional"/>
<xsd:attribute name="objFlags" type="xsd:unsignedInt" use="optional"/>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="index" type="xsd:hexBinary" use="required"/>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="objectType" type="xsd:unsignedByte" use="required"/>
<xsd:attribute name="dataType" type="xsd:hexBinary" use="optional"/>
<xsd:attribute name="lowLimit" type="xsd:string" use="optional"/>
<xsd:attribute name="highLimit" type="xsd:string" use="optional"/>
<xsd:attribute name="accessType" use="optional">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="ro"/>
<xsd:enumeration value="wo"/>
<xsd:enumeration value="rw"/>
<xsd:enumeration value="rwr"/>
<xsd:enumeration value="rww"/>
<xsd:enumeration value="const"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="defaultValue" type="xsd:string" use="optional"/>
<xsd:attribute name="actualValue" type="xsd:string" use="optional"/>
<xsd:attribute name="denotation" type="xsd:string" use="optional"/>
```

```
<xsd:attribute name="PDOmapping" type="xsd:boolean" use="optional"/>
<xsd:attribute name="objFlags" type="xsd:hexBinary" use="optional"/>
<xsd:attribute name="uniqueIDRef" type="xsd:IDREF" use="optional"/>
<xsd:attribute name="subNumber" type="xsd:unsignedByte" use="optional"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

Приложение ДА
(справочное)

Сведения о соответствии ссылочных международных стандартов ссылочным национальным стандартам Российской Федерации

Таблица ДА.1

Обозначение ссылочного международного стандарта	Степень соответствия	Обозначение и наименование соответствующего национального стандарта
ИСО 639-1:2002		*
ИСО 639-2:1998		*
ИСО 15745-1:2003		*
ИСО 15745-3:2003		*
ИСО/МЭК 7984-4:1989		*
ИСО/МЭК 8802-3:2000		*
ИСО/МЭК 10646:2012		*
МЭК 61158 (все части)		*
МЭК 61784-1:2010		*
ИСО 3166-1:2006		*
ИСО/МЭК 11578:1996		*
МЭК/ПАС 62030:2004		*
МЭК/ПАС 62407:2005		*
МЭК/ПАС 62408:2005		*

* Соответствующий национальный стандарт отсутствует (в разработке). До его утверждения рекомендуется использовать перевод на русский язык данного международного стандарта. Перевод данного международного стандарта находится в Федеральном информационном фонде технических регламентов и стандартов.

Библиография

[1] МЭК/ТО 13283:1998 Промышленная автоматизация. Критичные по времени архитектуры сообщений. Требования потребителей и сетевое управление для систем сообщений критичных по времени

(ISO/TR 13283:1998) (Industrial automation. Time-critical communications architectures. User requirements and network management for time-critical communications systems)

[2] МЭК/PAS 61499-1:2000 Блоки функциональные для систем измерения и управления производственными процессами. Часть 1. Архитектура

(IEC/PAS 61499-1:2000) (Function blocks for industrial-process measurement and control systems. Part 1. Architecture)

[3] МЭК ТС 61915:2003 Комплектные распределительные устройства низковольтные. Принципы разработки приборных профилей для сетевых промышленных устройств

(IEC TS 61915:2003) (Function blocks for industrial-process measurement and control systems. Part 1. Architecture)

[4] MSTC/JOP 1101:1999 Specification Autonimous Decentralised Protocol R3.0

[5] ODVA/CI EtherNet/IP:2001 EtherNet/IP Specification (Release 1/0)

[6] МЭК 61131-8:2003 Контроллеры программируемые. Часть 8. Руководящие указания по применению и реализации языков программирования

(IEC 61131-8:2003) (Programmable controllers – Part 8: Guidelines for the application and implementation of programming languages)

[7] МЭК/PAS 61804-2:2002 Блоки функциональные (FB) для управления процессом. Часть 2. Спецификация концепции FB и языка описания электронного устройства (EDDL)

(IEC/PAS 61804-2:2002) (Function blocks (FB) for process control - Part 2: Specification of FB concept and Electronic Device Description Language (EDDL))

[8] ИСО 2382 (все части) Информационные технологии – Словарь

(ISO 2382 (all parts) (Information technology – Vocabulary)

[9] ISO/AFNOR Dictionary of Computer Science (1997)

УДК 658.52.011.56 ОКС 25.040.40 Т 58

Ключевые слова: автоматизированные промышленные системы, интеграция, жизненный цикл систем, управление производством

Подписано в печать 30.04.2014. Формат 60x84 1/8.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

ФГУП «СТАНДАРТИНФОРМ»

123995 Москва, Гранатный пер., 4.

www.gostinfo.ru gost.e-ecolog.ru

gost.e-ecolog.ru