ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

ПРЕДВАРИТЕЛЬНЫЙ НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПНСТ 422— 2020 (ИСО/МЭК 30128:2014)

Информационные технологии

СЕТИ СЕНСОРНЫЕ

Сетевой интерфейс прикладного программирования датчика

(ISO/IEC 30128:2014, MOD)

Издание официальное

Москва
Стандартинформ
2020

gost.e-ecolog.ru

Предисловие

1 ПОДГОТОВЛЕН Акционерным обществом «Всероссийский научно-исследовательский институт сертификации» (АО «ВНИИС») и Акционерным обществом «Российская венчурная компания» (АО «РВК») на основе собственного перевода на русский язык англоязычной версии стандарта, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 194 «Кибер-физические системы»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 23 июля 2020 г. № 31-пнст

4 Настоящий стандарт является модифицированным по отношению к международному стандарту ИСО/МЭК 30128:2014 «Информационные технологии. Сенсорные сети. Сетевой интерфейс прикладного программирования датчика» (ISO/IEC 30128:2014 «Information technology — Sensor networks — Generic Sensor Network Application Interface», MOD) путем:

изменения отдельных фраз (слов, значений показателей, ссылок), которые выделены в тексте курсивом;

исключения структурного элемента.

Внесение указанных технических отклонений направлено на учет потребностей национальной экономики Российской Федерации.

Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте, приведены в дополнительном приложении ДА.

Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта приведено в дополнительном приложении ДБ

5 Некоторые элементы настоящего стандарта могут быть объектами патентных прав. Международная организация по стандартизации (ИСО) и Международная электротехническая комиссия (МЭК) не несут ответственности за установление подлинности каких-либо или всех таких патентных прав

Правила применения настоящего стандарта и проведения его мониторинга установлены в ГОСТ Р 1.16—2011 (разделы 5 и 6).

Федеральное агентство по техническому регулированию и метрологии собирает сведения о практическом применении настоящего стандарта. Данные сведения, а также замечания и предложения по содержанию стандарта можно направить не позднее чем за 4 мес до истечения срока его действия разработчику настоящего стандарта по адресу: 121205 Москва, Инновационный центр Сколково, ул. Нобеля, д. 1, e-mail: gost.e-ecolog.ru *в Федеральное агентство по техническому регулированию и метрологии: 109074 Москва, Китайгородский проезд, д. 7, стр. 1.*

*В случае отмены настоящего стандарта соответствующая информация будет опубликована в ежемесячном информационном указателе «Национальные стандарты» и также будет размещена на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (*gost.e-ecolog.ru

© ISO, 2014 — Все права сохраняются
© Стандартинформ, оформление, 2020

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

Содержание

1 Область применения 1
2 Нормативные ссылки 1
3 Термины и определения 1
4 Сокращения 2
5 Обзор приложений сенсорных сетей 2
5.1 Модель связи 2
5.2 Операции клиента сенсорной сети 4
6 Возможности сенсорной сети 8
7 Защищенность 8
8 Модель сенсорных данных и метаданных 8
9 Спецификация общего интерфейса приложения универсальной сенсорной сети 9
9.1 Обзор общего интерфейса приложения сенсорной сети 9
9.2 Обязательные операции 11
9.3 Дополнительные операции 16
Приложение А (справочное) Описание сенсорной сети (пример) 24
Приложение В (справочное) Тип датчика и единица измерения (пример) 27
Приложение ДА (справочное) Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте 29
Приложение ДБ (справочное) Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта 30
Библиография 31

Введение

Сенсорная сеть — это технология, позволяющая создавать интеллектуальные среды, сервисы мониторинга здоровья, сервисы перекрестной реальности и т.д. Однако на практике большое количество реализаций сенсорной сети не может взаимодействовать.

Сенсорные сети разрабатываются в соответствии с требованиями к приложениям сенсорной сети в рамках аппаратных ограничений этой сети. Требования касаются того, как приложения сенсорной сети используют сенсорные сети. Требования к приложениям сенсорной сети включают в себя требования к транспортному уровню, требования к аппаратному обеспечению сенсорных сетей, эксплуатационные требования к приложениям и т. д. Эксплуатационные требования к приложениям могут быть обобщены и использоваться для получения стандартных интерфейсов уровня приложений между сенсорными сетями и поставщиками служб сенсорной сети.

Настоящий стандарт определяет интерфейсы приложений сенсорной сети на основе обобщенных эксплуатационных требований к приложениям сенсорной сети с учетом аппаратных ограничений сенсорной сети.

ПНСТ 422—2020
(ИСО/МЭК 30128:2014)

ПРЕДВАРИТЕЛЬНЫЙ НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационные технологии

СЕТИ СЕНСОРНЫЕ

Сетевой интерфейс прикладного программирования датчика

Information technology. Sensor networks. Generic sensor network application interface

Срок действия — с 2021—01—01
до 2024—01—01

1 Область применения

Настоящий стандарт определяет интерфейсы между уровнем приложения поставщиков служб и шлюзами сенсорной сети.

Настоящий стандарт содержит:

- описание эксплуатационных требований к приложениям общей сенсорной сети;
- описание возможностей сенсорной сети;
- определение обязательных и дополнительных интерфейсов между уровнем приложения поставщиков служб и шлюзами сенсорной сети.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты:

ГОСТ Р 56947/ISO/IEC/IEEE 21450:2010 Информационные технологии. Интерфейс интеллектуального преобразователя для датчиков и исполнительных устройств. Общие функции, протоколы взаимодействия и форматы электронной таблицы данных преобразователя (ЭТДП)

ГОСТ Р 57668 (ИСО 19115-1:2014) Пространственные данные. Метаданные. Часть 1. Основные положения

Примечание — При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет или по ежегодному информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячного информационного указателя «Национальные стандарты» за текущий год. Если заменен ссылочный стандарт, на который дана недатированная ссылка, то рекомендуется использовать действующую версию этого стандарта с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт, на который дана датированная ссылка, то рекомендуется использовать версию этого стандарта с указанным выше годом утверждения (принятия). Если после утверждения настоящего стандарта в ссылочный стандарт, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение рекомендуется применять без учета данного изменения. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, рекомендуется применять в части, не затрагивающей эту ссылку.

3 Термины и определения

В настоящем стандарте применены следующие термины с соответствующими определениями (см. также [*1*]):

Издание официальное

3.1 **непрерывный режим** (continuous mode): Режим запроса сенсорных данных со значением длительности *d* и значением интервала *t*.

3.2 **режим событий** (event mode): Режим запроса сенсорных данных с условиями события, при котором сенсорные сети проводят сбор и передачу сенсорных данных только при соблюдении условий события.

3.3 **мгновенный режим** (instant mode): Режим запроса сенсорных данных для немедленного однократного ответа от сенсорной сети.

3.4 **режим включения** (onTime mode): Режим запроса сенсорных данных в определенный промежуток времени.

3.5 **координатор PAN** (PAN coordinator): Устройство, которое отвечает за формирование и обслуживание PAN.

3.6 **режим pull** (pull mode): Режим доставки сенсорных данных при наличии запроса на доставку.

3.7 **режим push** (push mode): Режим доставки сенсорных данных при отсутствии запроса на доставку.

Примечание — Возможна реализация, когда доставка сенсорных данных в режиме push инициируется и останавливается запросом.

3.8 **клиент сенсорной сети;** SNC (sensor network client; SNC): Программное обеспечение приложения, которое использует информацию, предоставленную сенсорной сетью.

3.9 **ресурс сенсорной сети** (sensor network resource): Сущность сенсорной сети, которая может быть шлюзом сенсорной сети, координатором PAN (при наличии), сенсорным узлом или преобразователем.

3.10 **преобразователь** (transducer): Устройство преобразования энергии одного типа в другой, откалиброванное для минимизации ошибок в процессе преобразования, которое может быть датчиком или исполнительным устройством.

4 Сокращения

В настоящем стандарте применены следующие сокращения:

АСН.1 — абстрактная синтаксическая нотация версии 1 (Abstract Syntax Notation One, ASN.1);

ЭДТП — электронная таблица данных преобразователя (Transducer Electronic Data Sheet, TEDS):

CoAP — ограниченный протокол уровня приложений (Constrained Application Protocol);

DNS — система доменных имен (Domain Name System);

NAK — отсутствие подтверждения приема (Negative Acknowledge);

PAN — персональная сеть (Personal Area Network);

SensorML — язык сенсорной модели (Sensor Model Language);

SNC — клиент сенсорной сети (Sensor Network Client);

SWE — реализация возможностей сенсора через Интернет (Sensor Web Enablement).

5 Обзор приложений сенсорных сетей

5.1 Модель связи

Приложения сенсорной сети предоставляют пользователям службы, связанные с датчиками или исполнительными устройствами, взаимодействуя по крайней мере с одной сенсорной сетью. Поставщикам служб сенсорной сети необходимо устанавливать связь с сенсорными сетями различных типов для предоставления интегрированных служб для пользователей. В другом варианте сенсорные сети предоставляют службы сенсорной сети нескольким поставщикам служб. Модель связи между поставщиками служб сенсорной сети и сенсорными сетями обобщает два варианта.

В одном варианте интерфейс определяется сенсорной сетью, которая предоставляет службы сенсорной сети. Поставщики служб сенсорной сети реализуют интерфейс, определенный сенсорной сетью, для связи с сенсорной сетью (*см. рисунок 1*). Если стандартный интерфейс между сенсорными сетями и поставщиками служб отсутствует, то поставщики служб сенсорной сети должны реализовать разные интерфейсы для связи с разными сенсорными сетями. На рисунке 1 поставщик служб сенсорной сети должен реализовать интерфейс А, интерфейс В и интерфейс С для предоставления интегрированных служб для пользователей. Реализация всех интерфейсов различных сенсорных сетей является трудоемким и неоптимизированным процессом.



Рисунок 1 — Модель связи: множество сенсорных сетей и один поставщик служб сенсорной сети

В другом варианте поставщик служб сенсорной сети определяет собственный интерфейс. Таким образом, сенсорные сети должны реализовывать интерфейс для связи с поставщиком служб сенсорной сети (*см. рисунок 2*). При наличии разных поставщиков служб сенсорной сети с разными интерфейсами сенсорные сети должны реализовать каждый интерфейс. С точки зрения поставщиков сенсорной сети, необходимо или разработать единую сенсорную сеть, которая может реализовывать все интерфейсы, или разработать разные сенсорные сети для разных поставщиков служб сенсорной сети для предоставления одних и тех же служб сенсорной сети. Обе ситуации являются неоптимизированными и увеличивают стоимость и время разработки сенсорной сети.

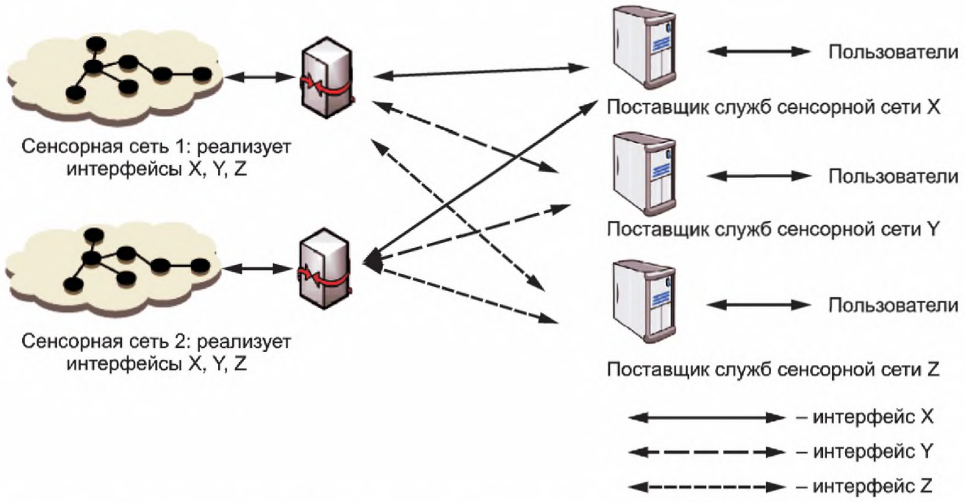


Рисунок 2 — Модель связи: одна сенсорная сеть и множество поставщиков служб сенсорной сети

В связи с вышеприведенным становится актуальным наличие стандартной спецификации интерфейса между сенсорными сетями и поставщиками служб сенсорной сети. Стандартный интерфейс позволяет реализовать реализацию сенсорной сети без учета требований отдельных приложений сенсорной сети, что обеспечивает экономичную массовую разработку сенсорных сетей.

Роль шлюзов сенсорной сети крайне важна. Шлюз сенсорной сети является внешним интерфейсом сенсорной сети по отношению к поставщикам служб сенсорной сети. Следовательно, если определен стандартный интерфейс между сенсорными сетями и поставщиками служб сенсорной сети, то он будет находиться между шлюзом и сенсорной сетью. В случае, когда устройство играет роль автономного сенсорного узла без шлюза, то оно может рассматриваться как шлюз и сенсорный узел одновременно.

Шлюз сенсорной сети преобразует сообщения поставщиков служб сенсорной сети в сообщения сенсорных сетей (например, сообщения ZigBee, Bluetooth, CoAP и т. д.) и наоборот. В некоторых случаях

шлюз сенсорной сети должен проводить синхронизацию времени для доставки сообщений на сенсорные узлы. Это связано с тем, что, когда поставщик служб сенсорной сети отправляет сообщение на сенсорные узлы, сенсорные узлы могут находиться в режиме сна для экономии энергии. В этом случае шлюз сенсорной сети для сохранения и пересылки сообщений на сенсорные узлы должен знать жизненный цикл каждого сенсорного узла. При регистрации сенсорного узла или преобразователя у поставщика служб сенсорной сети, шлюз может потребовать дополнительную информацию от сенсорного узла в сообщении регистрации для того, чтобы внести информацию в иерархическую топологию. Если шлюз является высокопроизводительным, на нем могут проводиться сложные операции от менее производительных сенсорных узлов. В некоторых случаях вычислительные возможности сенсорной сети определяются вычислительными возможностями шлюзов. Таким образом, шлюзы сенсорных сетей играют очень важную роль в сенсорных сетях и отличаются от шлюзов в других областях применения.

5.2 Операции клиента сенсорной сети

5.2.1 Общие положения

Существуют различные виды приложений сенсорных сетей: приложения для логистики и управления цепочками поставок, приложения для распределения энергии и коммунальных услуг, приложения для мониторинга и контроля промышленного производства, здравоохранение, медицинские приложения и т. д.

Приложения сенсорной сети обычно используют сенсорные данные, а некоторые приложения управляют соответствующими исполнительными устройствами для обработки наблюдаемой ситуации. Другие приложения сенсорных сетей контролируют состояние сенсорных сетей и сами сенсорные сети. В настоящем стандарте операции приложений сенсорной сети включают три категории: оперирование сенсорными данными, оперирование исполнительными устройствами, мониторинг и контроль сенсорной сети.

Оперирование сенсорными данными включает в себя несколько режимов сбора сенсорных данных. Режим сбора данных классифицируется на режим push и режим pull. Режим pull включает в себя мгновенный режим, непрерывный режим, режим событий и режим включения.

Оперирование исполнительными устройствами включает в себя работу с различными типами исполнительных устройств: исполнительное устройство включения/выключения, исполнительное устройство с отображением текста/изображения, многофункциональное комплексное исполнительное устройство и т. д. Все операции с исполнительными устройствами выполняются путем выдачи запросов действий с соответствующими параметрами. После завершения действия исполнительным устройством сенсорные сети должны ответить соответствующим статусом действия.

Мониторинг и контроль сенсорной сети заключается в том, что клиент сенсорной сети должен знать текущее состояние сенсорных сетей и контролировать сенсорные сети в соответствии с политикой клиента. Контроль сенсорной сети может осуществляться независимо от процесса мониторинга. Клиент сенсорной сети выдает запрос контроля сенсорной сети для административных целей или для работы с результатами мониторинга. Контроль сенсорной сети может включать в себя несколько операций, таких как перезагрузка, отключение и реконфигурация.

В настоящем стандарте понятие клиента сенсорной сети (SNC) используется для обозначения программного обеспечения приложения, находящегося у поставщика служб сенсорной сети.

5.2.2 Оперирование сенсорными данными

5.2.2.1 Общие положения

Существуют различные способы сбора сенсорных данных в сенсорных сетях. Передача собранных сенсорных данных проводится двумя способами. Одним способом является отправка собранных сенсорных данных без учета конкретных запросов от клиентов сенсорной сети, что называется режимом push. Другим способом является отправка сенсорных данных в соответствии с конкретными запросами клиентов сенсорной сети, что называется режимом pull. Режим pull далее подразделяется на мгновенный режим, непрерывный режим и режим событий в зависимости от характеристик запроса от клиентов сенсорной сети. Клиенты сенсорной сети могут запрашивать сенсорные данные только один раз (мгновенный режим), в режиме повторения (непрерывный режим), только при некоторых событиях (режим событий) или в определенный промежуток времени (режим включения).

5.2.2.2 Режим push

На рисунке 3 показано оперирование сенсорными данными в режиме push.



Рисунок 3 — Сбор сенсорных данных: режим push

Сенсорная сеть собирает и передает сенсорные данные на основе своих настроек сбора и передачи. Оперирование сенсорными данными в режиме push может быть реализовано двумя способами. В первом способе сенсорная сеть передает сенсорные данные при подключении к клиенту сенсорной сети. При втором способе сенсорная сеть начинает отправлять сенсорные данные при получении команды «старт» и прекращает отправлять при получении команды «стоп» от клиента сенсорной сети. В этом случае команды «старт» и «стоп» выдаются клиентом сенсорной сети, но это не меняет настройки сенсорной сети по сбору и передаче сенсорных данных.

Режим push используется в простых приложениях мониторинга сенсорной сети. Ожидаемая вычислительная сложность сенсорных сетей в режиме push является очень низкой. Сенсорные сети передают сенсорные данные клиенту сенсорной сети, и обработка данных проводится клиентом сенсорной сети.

5.2.2.3 Режим pull: мгновенный режим

На рисунке 4 изображен мгновенный режим оперирования сенсорными данными в режиме pull.



Рисунок 4 — Сбор сенсорных данных: режим pull — мгновенный режим

Клиент сенсорной сети отправляет запрос на сбор сенсорных данных в реальном времени. При получении запроса сенсорная сеть проводит сбор сенсорных данных с сенсорных узлов. После завершения сбора сенсорных данных сенсорная сеть передает собранные сенсорные данные датчика клиенту. Восприятие и передача выполняются один раз в режиме реального времени.

Мгновенный режим является самой базовой операцией режима pull. Мгновенный запрос сенсорных данных может включать в себя условия. Например, клиент сенсорной сети делает запрос с условием измерить значение температуры места № 1, только если значение температуры превышает 30 °С. Тогда сенсорная сеть отправляет значение температуры, только если значение температуры превышает 30 °С. Если запрос включает условия, сенсорная сеть проверяет условия перед отправкой собранных сенсорных данных. Возможность проверки условий является дополнительной функцией сенсорных сетей. Поэтому сенсорная сеть должна заранее информировать клиентов сенсорной сети о возможности проверки условий.

По сравнению с режимом push, ожидаемая вычислительная сложность сенсорных сетей для выполнения мгновенного режима работы выше. Сенсорная сеть должна обрабатывать запросы сенсорных данных и отвечать.

5.2.2.4 Режим pull: непрерывный режим

На рисунке 5 изображен непрерывный режим оперирования сенсорными данными в режиме pull.

Клиент сенсорной сети отправляет запрос на сбор сенсорных данных. После получения запроса сенсорная сеть проводит сбор сенсорных данных при каждом значении интервала *t* в течение значения *d* длительности и периодически передает собранные сенсорные данные клиенту.



Рисунок 5 — Сбор сенсорных данных: режим pull – непрерывный режим

Непрерывный режим используется приложениями сенсорной сети, которым требуются сенсорные данные с определенным значением интервала и определенным значением продолжительности. Запрос сенсорных данных в непрерывном режиме должен включать в себя условия, в этом случае сенсорная сеть должна поддерживать возможность проверки условий. Поэтому сенсорная сеть должна заранее информировать клиентов сенсорной сети о возможности проверки условий. Если необходимо остановить работу в непрерывном режиме, клиент сенсорной сети отправляет запрос на остановку обработки в сенсорную сеть. Остановка обработки является дополнительной функцией сенсорной сети, поэтому она должна быть заранее известна клиенту сенсорной сети.

По сравнению с мгновенным режимом ожидаемая вычислительная сложность сенсорных сетей для выполнения непрерывного режима работы выше.

5.2.2.5 Режим pull: режим событий

На рисунке 6 изображено оперирование сенсорными данными в режиме pull — режиме событий.

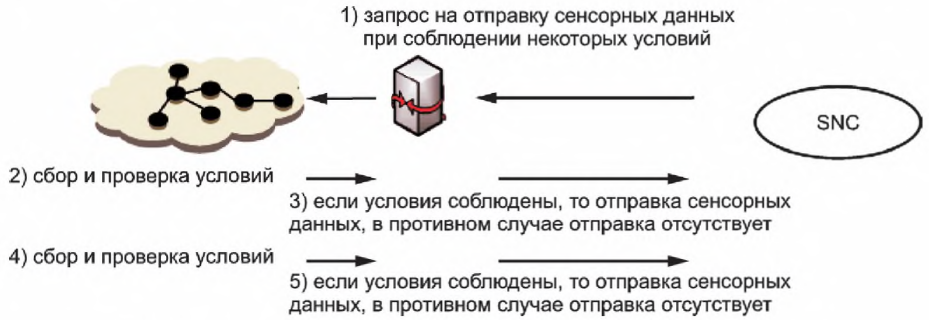


Рисунок 6 — Сбор сенсорных данных: режим pull – режим событий

Клиент сенсорной сети отправляет запрос на сбор и отправку сенсорных данных только при соблюдении указанных условий. Условия задаются клиентом сенсорной сети. Различие между режимом событий и другими режимами pull с условиями заключается в том, что условия в режиме событий являются обязательными. Запросы сенсорных данных в режиме событий обрабатываются сенсорной сетью непрерывно до тех пор, пока клиент сенсорной сети не сделает запрос на остановку обработки. Сенсорная сеть продолжает сбор сенсорных данных на основе настроек сбора, как в режиме push, и проверяет условия клиента. Например, клиент сенсорной сети запрашивает данные о температуре в сенсорной сети № 1, только когда наблюдаемые значения температуры превышают определенный порог, чтобы включить систему охлаждения. Тогда клиент отправляет запрос «отправка данных о температуре, собранных в сенсорной сети № 1, только если значение температуры выше 30 °С».

Режим событий эффективно используется в приложениях мониторинга стихийных бедствий. Приложения мониторинга стихийных бедствий должны проверять аномальные условия в областях мониторинга. Использование режима событий сокращает ненужный трафик между сенсорными сетями и клиентами сенсорных сетей, что приводит к эффективной обработке приложений и продлению срока службы аккумуляторов сенсорной сети за счет уменьшения передачи сигнала.

5.2.2.6 Режим pull: режим включения

На рисунке 7 изображено оперирование сенсорными данными в режиме pull — режиме включения.



Рисунок 7 — Сбор сенсорных данных: режим pull — режим включения

Клиент сенсорной сети отправляет запрос на сбор сенсорных данных в определенный промежуток времени, и сенсорная сеть проводит сбор сенсорных данных в указанное время. Режим включения используется совместно с другими режимами: запросом на сбор сенсорных данных в мгновенном режиме, непрерывном режиме или режиме событий или оперированием исполнительным устройством. Например, если запрос сенсорных данных в мгновенном режиме включает в себя поле промежутка времени, то сенсорная сеть должна обработать запрос в мгновенном режиме в указанное время. При обработке в режиме включения очень важна синхронизация времени между клиентами сенсорной сети и сенсорными сетями.

По сравнению с другими режимами pull, работа в режиме включения требует более высокой вычислительной сложности. Сенсорная сеть должна обеспечивать сложность обработки времени и обработки операции в режиме pull.

***5.2.3* Оперирование исполнительными устройствами**

На рисунке 8 изображено оперирование исполнительными устройствами.



Рисунок 8 — Оперирование исполнительными устройствами

Клиент сенсорной сети управляет исполнительным устройством путем задания параметров исполнения. Исполнительное устройства отвечает клиенту результатом исполнения. Клиент сенсорной сети должен указать идентификатор целевого исполнительного устройства, тип исполнения, параметры исполнения и опционально время исполнения.

Обработка времени исполнения является дополнительной функцией сенсорных сетей, поэтому сенсорные сети должны заранее информировать клиентов сенсорной сети о возможности обработки времени исполнения.

Сложность оперирования исполнительным устройством пропорциональна сложности параметров исполнения. Детальная конструкция исполнительных устройств не рассматривается в настоящем стандарте ввиду разнообразия типов и параметров исполнительных устройств.

***5.2.4* Мониторинг и контроль сенсорной сети**

На рисунке 9 представлены мониторинг и контроль сенсорной сети.



Рисунок 9 — Мониторинг и контроль сенсорной сети

Клиентам сенсорной сети может потребоваться информация о состоянии сенсорной сети, такая как топология, оставшееся время работы каждого сенсорного узла от аккумулятора и доступность каждого сенсорного узла. Информация о состоянии собирается из сенсорных сетей в процессе мониторинга. Мониторинг сенсорной сети осуществляется двумя способами. Некоторые сенсорные сети собирают и отправляют информацию о статусе автономно. Другие сети собирают и отправляют информацию о состоянии только при наличии запроса от клиента сенсорной сети. Сенсорная сеть должна заранее информировать о своем режиме мониторинга.

Клиенты сенсорных сетей могут контролировать сенсорные сети путем перезагрузки, реконфигурации и т. д. Функции контроля реализуются сенсорными сетями выборочно на основе вычислительных ограничений сенсорных сетей и политики поставщика сенсорных сетей.

6 Возможности сенсорной сети

Сенсорные сети имеют разные вычислительные возможности. В настоящее время многие сенсорные сети поддерживают обработку сенсорных данных только в режиме push. Однако для некоторых специализированных служб сенсорных сетей, таких как военные службы или медицинские службы, требуются мощные сенсорные узлы. По мере развития технологий устройств и технологий связи появляются небольшие и интеллектуальные устройства с высокоскоростными вычислительными и коммуникационными возможностями и увеличивающимся объемом памяти. Использование вычислительных возможностей сенсорных сетей является целесообразным для клиентов сенсорной сети, поскольку это снижает трафик и нагрузку на клиентов сенсорной сети.

Некоторые сенсорные сети поддерживают такие вычисления, как запросы сенсорных данных в мгновенном режиме. Другие сенсорные сети поддерживают еще более сложные вычисления, такие как запросы сенсорных данных в непрерывном режиме и режиме событий. Некоторые сенсорные сети могут даже обрабатывать несколько запросов сенсорных данных одновременно. Возможности сенсорных сетей определяются ограничениями и политиками поставщиков сенсорных сетей, требованиями и политиками клиентов сенсорных сетей, ограничениями оборудования сенсорных сетей и операционной системы и т. д. Необходимо, чтобы сенсорные сети заранее информировали о своих возможностях, что позволит клиентам сенсорной сети использовать оптимальные функции среди предоставляемых возможностей. Информация о возможностях сенсорной сети должна быть включена в метаданные сенсорной сети. Метаданные сенсорной сети должны включать идентификационную информацию сенсорной сети, информацию о возможностях и т. д. Метаданные сенсорной сети доставляются клиентам сенсорной сети посредством процедуры регистрации.

Возможности сенсорной сети могут предоставляться преобразователями, сенсорными узлами или шлюзами.

7 Защищенность

Сенсорные данные показывают текущее реальное состояние конкретной среды, личного здоровья, конкретного производства, конкретного завода и т. д. Владельцы сенсорных данных нуждаются в надлежащих мерах защиты (аутентификация для пользователей и защищенный обмен данными). С другой стороны, пользователям сенсорных данных требуются соответствующие уровни доверия к сенсорным данным и поставщикам сенсорных данных. Поэтому поставщики сенсорных данных также должны предоставить подтверждение своей подлинности пользователям сенсорных данных.

При проектировании интерфейса между сенсорными сетями и клиентами сенсорных сетей должны быть приняты во внимание соответствующие меры аутентификации и сохранения конфиденциальности. Сенсорная сеть может иметь авторизацию для предоставления разного уровня служб различным клиентам сенсорной сети.

Вопросы защищенности не рассматриваются в настоящем стандарте и являются объектом стандартизации профильных национальных технических комитетов.

8 Модель сенсорных данных и метаданных

Настоящий стандарт не определяет конкретную модель сенсорных данных и метаданных сенсорной сети. Для сенсорных данных может применяться структура спецификации, представленная в *9.2.3*. Для метаданных сенсорной сети могут применяться SensorML SWE и ЭДТП, определенные в *ГОСТ Р 56947* и в приложении А (см. также [*2*], [*3*]).

Примечания
1 Схема реализации покрытия определена в [4].
2 Метаданные для географической информации определены в *ГОСТ Р 57668*.

9 Спецификация общего интерфейса приложения универсальной сенсорной сети

9.1 Обзор общего интерфейса приложения сенсорной сети

9.1.1 Общие положения

Общий интерфейс приложения сенсорной сети включает в себя обязательный и дополнительный интерфейсы. Обязательный интерфейс должен быть реализован сенсорными сетями и клиентами сенсорных сетей. Обязательный интерфейс представлен интерфейсом регистрации, интерфейсом отмены регистрации и интерфейсом передачи сенсорных данных. Обязательный интерфейс представлен на рисунке 10. В настоящем стандарте интерфейс рассматривается в виде сообщения.

Обычно оперирование с сенсорными данными и исполнительными устройствами занимает некоторое время. Целесообразной является асинхронная обработка таких операций, что является вопросом реализации.

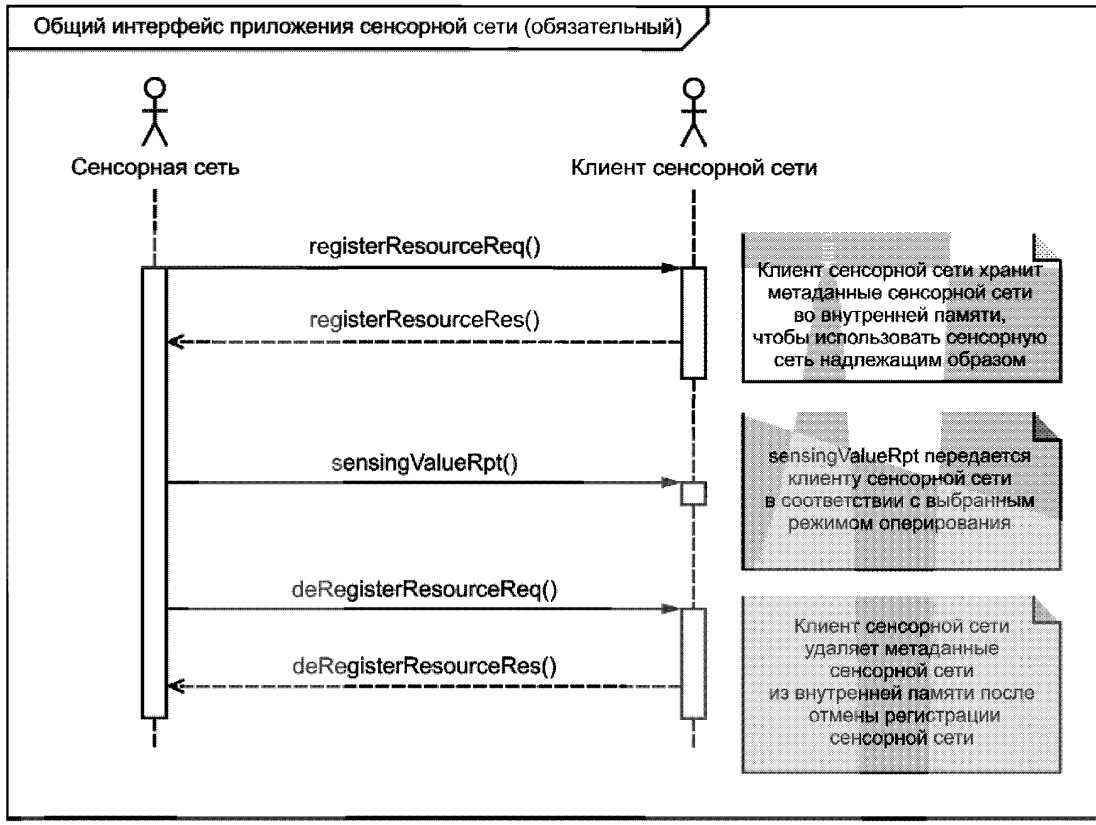


Рисунок 10 — Обязательный интерфейс приложения сенсорной сети

Дополнительный интерфейс может быть реализован сенсорными сетями и клиентами сенсорных сетей. В зависимости от ограничений и политики сенсорных сетей дополнительный интерфейс может быть реализован выборочно. Дополнительный интерфейс показан на рисунке 11. Дополнительный интерфейс включает в себя интерфейс оперирования сенсорными данными в режиме pull, интерфейс оперирования исполнительными устройствами, интерфейс мониторинга и контроля сенсорной сети, интерфейс масштабируемости и интерфейс обработки сообщений. Реализация дополнительного интерфейса должна соответствовать настоящему стандарту.

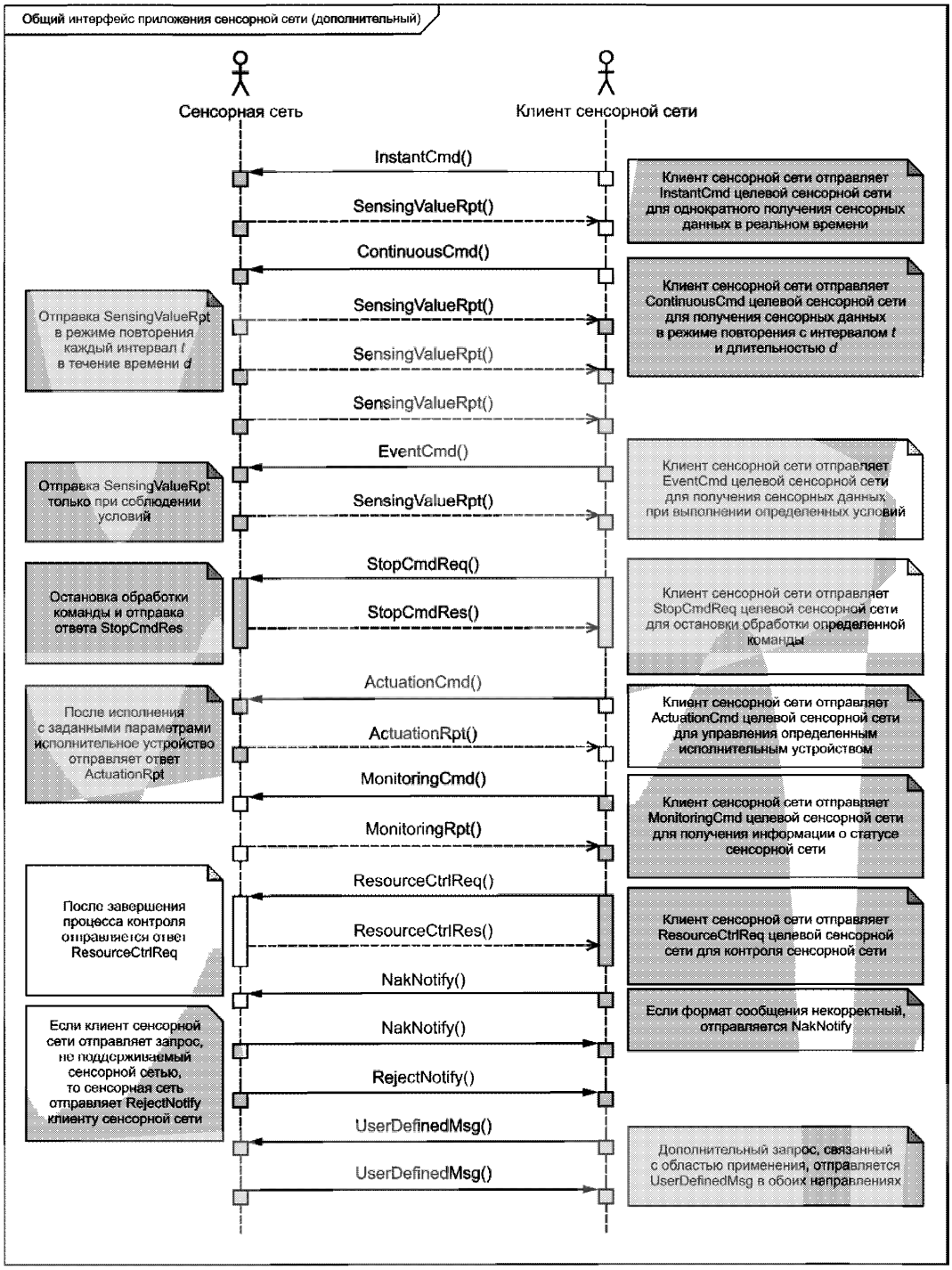


Рисунок 11 — Дополнительный интерфейс приложения сенсорной сети

9.1.2 Основная структура

Общий интерфейс приложения сенсорной сети определяется в виде набора сообщений. Все сообщения выражены с использованием нотации АСН.1. Структура сообщения приведена в таблице 1.

Примечание — Компиляторы АСН.1 создают коды уровня реализации с интерфейсами, определенными в качестве входных данных (см. [*5*]).

Таблица 1 — Основная структура сообщения

```
MessageType ::= PrintableString ("REGISTERRESOURCEREQ"| "REGISTERRESOURCERES"|"DEREGIS-
TERRESOURCEREQ"|"DEREGISTERRESOURCERES"| "INSTANTCMD"|"CONTINUOUSCMD"|"EVENTCMD"|"ACTUA-
TIONCMD"|"MONITORINGCMD"| "SENSINGVALUERPT"|"ACTUATIONRPT"|"MONITORINGRPT"|"STOPCMDREQ"|
"STOPCMDRES"|"RESOURCECTRLREQ"|"RESOURCECTRLRES"|"NAKNOTIFY"|    "REJECTNOTIFY"|"USERDE-
FINEDMESSAGE")

MessageBody ::= CHOICE {
 registerResourceReq       [0] RegisterResourceReq,
 registerResourceRes       [1] RegisterResourceRes,
 deRegisterResourceReq     [2] DeRegisterResourceReq,
 deRegisterResourceRes     [3] DeRegisterResourceRes,
 instantCmd                [4] InstantCmd,
 continuousCmd             [5] ContinuousCmd,
 eventCmd                  [6] EventCmd,
 sensingValueRpt           [7] SensingValueRpt,
 actuationCmd              [8] ActuationCmd,
 actuationRpt              [9] ActuationRpt,
 monitoringCmd             [10] MonitoringCmd,
 monitoringRpt             [11] MonitoringRpt,
 stopCmdReq                [12] StopCmdReq,
 stopCmdRes                [13] StopCmdRes,
 resourceCtrlReq           [14] ResourceCtrlReq,
 resourceCtrlRes           [15] ResourceCtrlRes,
 nakNotify                 [16] NakNotify,
 rejectNotify              [17] RejectNotify,
 userDefinedMessage        PrintableString
}

Message ::= SEQUENCE {
 sender PrintableString, -- id отправителя
 receiver [0] PrintableString, -- id получателя
 messageType MessageType,
 messageBody MessageBody
}
```

Все сообщения отформатированы в соответствии с основной структурой.

9.1.3 Типы датчиков и единицы измерения

Типы датчиков и единицы измерения не указаны, так как могут быть определены в соответствии с областью применения. В приложении В приведен пример спецификации типов датчиков и единиц измерения.

9.2 Обязательные операции

9.2.1 Регистрация

Данная операция проводит регистрацию сенсорной сети для клиента сенсорной сети. Функции регистрации включают в себя присвоение идентификаторов ресурсов сенсорной сети и объявление метаданных сенсорной сети. Идентификаторы ресурсов сенсорной сети используются клиентами сенсорной сети при обращении к конкретным ресурсам сенсорной сети.

Присвоение идентификаторов может проводиться двумя способами. В первом способе клиент сенсорной сети назначает идентификаторы ресурсам сенсорной сети на основе собственной схемы назначения идентификаторов. Во избежание дублирования сенсорные сети должны хранить таблицу сопоставления, которая отображает идентификаторы клиентов сенсорной сети на идентификаторы,

используемые в сенсорных сетях, и наоборот. Во втором способе сенсорные сети предлагают потенциальные идентификаторы, которые могут быть легко преобразованы в идентификаторы, используемые в сенсорных сетях без таблицы отображения. В этом случае клиенты сенсорной сети должны обеспечить уникальность идентификаторов.

В таблице 2 представлена структура сообщения регистрации, в таблице 3 показана информация сообщения регистрации.

Таблица 2 — Структура сообщения регистрации

```
RegisterResourceType ::= PrintableString("GATENODE"|"PAN"|"SENSORNODE"| "TRANSDUCER")

RegisterResourceReq ::= SEQUENCE {
 description GateNode,
 resourceType RegisterResourceType DEFAULT "GATENODE"
}

IDList ::= SEQUENCE SIZE(1..MAX) OF PrintableString

RetCode ::= PrintableString("SUCCESS"|"BADREQUEST"|"ERROR"|"UNDEFINED"| "STORED")

RegisterResourceRes ::= SEQUENCE {
 retCode RetCode,
 idList IDList
}
```

Таблица 3 — Информация сообщения регистрации

Поле	Описание	Примечание
RegisterResourceReq		
description	description включает в себя метаданные ресурса (шлюз, координатор PAN, сенсорный узел или преобразователь) а также поле идентификатора, чтобы предложить идентификатор для ресурса. При получении RegisterResourceReq клиент сенсорной сети должен проверить предложенный идентификатор, чтобы избежать конфликта с другими идентификаторами. description должно включать информацию о возможностях ресурса, касающуюся оперирования с сенсорными данными, оперирования исполнительными устройствами и управления сенсорной сетью. Конкретная модель description выходит за рамки положений настоящего стандарта. Для модели может быть использовано приложение А, ЭТДП или SensorML	–
resourceType	Тип ресурса	"GATENODE", "PAN", "SENSORNODE", "TRANSDUCER"
RegisterResourceRes		
retCode	Результат регистрации. Список значений может быть расширен	"SUCCESS", "BADREQUEST", "ERROR"
idList	Список зарегистрированных идентификаторов. Если description включает описания нескольких ресурсов сенсорной сети, тогда idList включает в себя несколько зарегистрированных идентификаторов	–

Сенсорная сеть включает в себя один шлюз и опционально координаторы PAN. Координатор PAN включает в себя как минимум один сенсорный узел. Сенсорный узел включает в себя более чем один преобразователь. Преобразователь представляет собой датчик или исполнительное устройства. Для предоставления служб сенсорной сети каждый элемент сенсорной сети должен быть зарегистрирован у клиентов сенсорной сети.

Регистрация может проводиться последовательно в иерархическом порядке: шлюз, далее координатор PAN, далее сенсорные узлы и в конце регистрации датчики. Все элементы сенсорной сети могут быть зарегистрированы одновременно. На рисунке 12 показаны два способа регистрации.

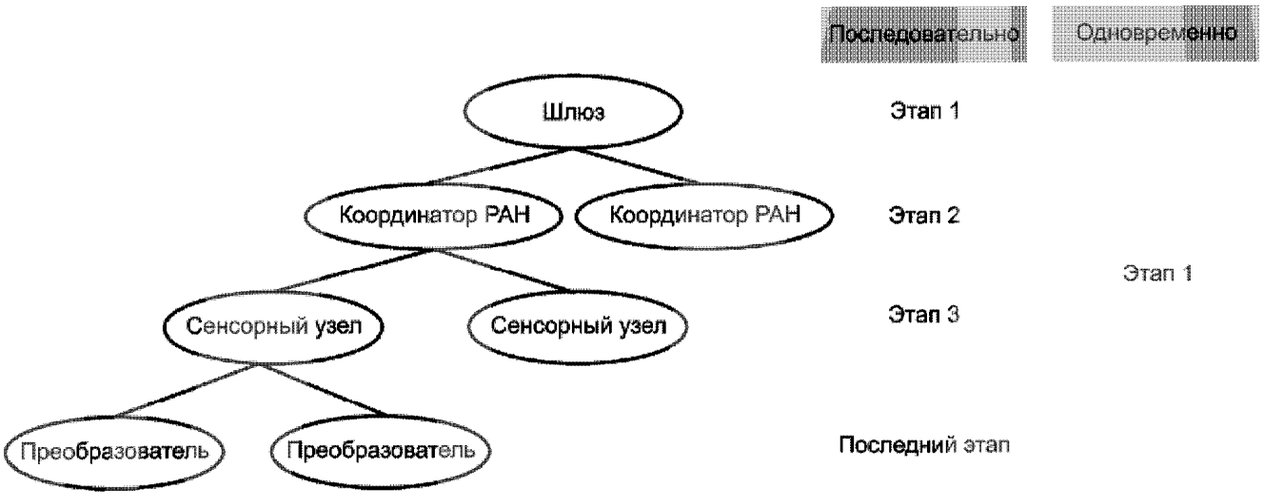


Рисунок 12 — Способы регистрации элементов сенсорной сети

При регистрации ресурса сенсорной сети у клиента сенсорной сети ресурс может предложить свой идентификатор, или клиент сенсорной сети может назначить уникальный идентификатор ресурсу. В качестве схемы идентификации для сенсорных сетей может использоваться иерархическая структура, например «gatenodeId: panId: nodeId: transducerId». Каждый идентификатор должен быть уникальным в пределах узла-родителя: transducerId является уникальным для сенсорного узла с noteId, noteId уникален для PAN с идентификатором panId, panId является уникальным для шлюза с gatenodeId. gatenodeId является уникальным идентификатором шлюза, и для этого требуется обеспечение уникальности gatenodeId в домене службы. Для обеспечения уникальности идентификатора шлюза может использоваться DNS или другие механизмы обеспечения уникальности идентификаторов шлюза в домене служб. Преимущество иерархической схемы идентификации состоит в том, что шлюзы, координаторы PAN и сенсорные узлы свободны от управления таблицей отображения идентификаторов.

Таблица 4 показывает пример схемы поля description шлюза сенсорной сети.

Таблица 4 — Пример поля description шлюза

```
SupportedTransportProtocol ::= PrintableString("XML_OVER_TCP"| "XML_OVER_UDP"|"XML_OVER_
HTTP"|"TEXT_OVER_TCP")

SupportedTransportProtocolList ::= SEQUENCE OF SupportedTransportProtocol

SupportedCommand ::= PrintableString("COMMAND_PUSH" | "COMMAND_INSTANT" | "COMMAND_EVENT"
| "COMMAND_CONTINUOUS" | "COMMAND_CONTROL" | "COMMAND_MONITORING" | "COMMAND_ACTUATION")

SupportedCommandList ::= SEQUENCE OF SupportedCommand

SupportedCommandAttribute ::= PrintableString("ATTRIBUTE_CONDITION" | "ATTRIBUTE_
FUNCTION" | "ATTRIBUTE_DURATION" | "ATTRIBUTE_SHUTDOWN" | "ATTRIBUTE_RESET" | "ATTRIBUTE_
REBOOT" | "ATTRIBUTE_START_SENSING" | "ATTRIBUTE_STOP_SENSING"|"ATTRIBUTE_PAN_ID_CHANGE"|
"ATTRIBUTE_CHANNEL_ID_CHANGE")

SupportedCommandAttributeList ::= SEQUENCE OF SupportedCommandAttribute

GateNodeSupportedOperationList ::= SEQUENCE {
  supportedCommandList SupportedCommandList,
  supportedCommandAttributeList SupportedCommandAttributeList
}
```

Окончание таблицы 4

```
Time ::= SEQUENCE {
 time INTEGER(0 .. MAX),
 unit TimeUnit DEFAULT "SEC"
}

Location ::= SEQUENCE {
 longitude REAL,
 latitude REAL,
 altitude REAL
}

MonitoringMode ::= PrintableString("MONITORING_PULL"|"MONITORING_PUSH")

GateNode ::= SEQUENCE {
 id PrintableString, -- format: host name or ip
 url [0] PrintableString OPTIONAL,
 manufacturer PrintableString OPTIONAL,
 productNo [1] PrintableString OPTIONAL,
 location Location OPTIONAL,
 dateTime [2] UTCTime OPTIONAL,
 supportedTransportProtocolList [3] SupportedTransportProtocolList OPTIONAL,
 supportedOperationList [4] GateNodeSupportedOperationList,
 panList [5] PanList OPTIONAL,
 monitoringMode [6] MonitoringMode OPTIONAL,
 monitoringPeriod INTEGER(0 .. MAX) OPTIONAL -- in seconds
}
```

В настоящем стандарте отсутствуют ограничения модели данных description. При этом description (метаданные) должно включать информацию о возможностях операций. Клиент сенсорной сети должен заранее быть информирован о предоставляемых операциях.

9.2.2 Отмена регистрации

Отмена регистрации — это операция по разрыву связи между сенсорной сетью и клиентом сенсорной сети. По одному сообщению регистрации могут быть отменены несколько элементов сенсорных сетей одновременно. Если произошла отмена регистрации родительского элемента, то в этот момент отменяется регистрация всех дочерних элементов.

В таблице 5 показана структура сообщения отмены регистрации, в таблице 6 показана информация сообщения отмены регистрации.

Таблица 5 — Структура сообщения отмены регистрации

```
DeRegisterResourceReq ::= SEQUENCE {
 idList IDList /* список идентификаторов для отмены регистрации у клиентов сенсорной
сети */
}

DeRegisterResourceRes ::= SEQUENCE {
 retCode RetCode,
 idList IDList
}
```

Таблица 6 — Информация сообщения отмены регистрации

Поле	Описание	Примечание
DeRegisterResourceReq		
idList	Список идентификаторов ресурсов сенсорной сети для отмены регистрации у клиентов сенсорной сети	—

Окончание таблицы 6

Поле	Описание	Примечание
DeRegisterResourceRes		
retCode	Результат отмены регистрации. Список значений может быть расширен	"SUCCESS", "BADREQUEST", "ERROR"
idList	Список идентификаторов ресурсов сенсорной сети, которые прошли отмену регистрации у клиентов сенсорной сети	—

9.2.3 Передача сенсорных данных

Передача сенсорных данных — это операция доставки собранных сенсорных данных клиенту сенсорной сети. Способ отправки сенсорных данных зависит от режима оперирования с сенсорными данными.

В таблице 7 показана структура сообщения передачи сенсорных данных, в таблице 8 показана информация сообщения передачи сенсорных данных.

Таблица 7 — Структура сообщения передачи сенсорных данных

```
SensingValueRpt ::= SEQUENCE {
 commandID INTEGER(0..MAX) OPTIONAL,
 sensingValueList SEQUENCE OF SensingValueList
}
```

Таблица 8 — Информация сообщения передачи сенсорных данных

Поле	Описание	Примечание
SensingValueRpt		
commandID	Идентификатор команды для сопоставления команды и соответствующей передачи	В режиме push идентификатор может отсутствовать
sensingValueList	Список собранных сенсорных исходных или обработанных данных. Датчик формирует SensingValue, сенсорный узел формирует SensingValueList, шлюз формирует SensingValueRpt. Детальная структура sensingValueList и SensingValue не ограничена. Они могут иметь структуру, представленную в [6], или структуру, приведенную в таблице 9	—

В таблице 9 приведен пример сенсорных данных, которые могут быть использованы для поля sensingValueList.

Таблица 9 — Пример структуры сенсорных данных

```
Function ::= PrintableString("MIN" | "MAX" | "AVG" | "SUM")

SensingValue ::= SEQUENCE {
 targetID PrintableString, /* идентификатор датчика */
 sensorType PrintableString, /* тип датчика */
 value PrintableString, /* измеренное значение */
 function Function OPTIONAL /*примененная функция агрегирования значения*/
}

SensingValueList ::= SEQUENCE {
 timeStamp PrintableString, /* время отправки */
 nodeID PrintableString, /* идентификатор сенсорного узла */
 values SEQUENCE SIZE(1..MAX) OF SensingValue /* список сенсорных данных, собранных
сенсорным узлом*/
}
```

9.3 Дополнительные операции

9.3.1 Команда мгновенного режима

Команда мгновенного режима — это сообщение для ресурсов сенсорной сети по выполнению однократного восприятия сенсорных данных. Команда определяет целевые идентификаторы ресурсов, типы датчиков, условия для проверки (при наличии) и период времени (если необходимо).

В таблице 10 показана структура сообщения команды мгновенного режима, в таблице 11 показана информация сообщения команды мгновенного режима.

Таблица 10 — Структура сообщения команды мгновенного режима

```
Function ::= PrintableString("MIN" | "MAX" | "AVG" | "SUM")

LogicalOp ::= PrintableString("OR" | "AND")

RelationalOp ::= PrintableString("GT" | "GE" | "LT" | "LE" | "EQ" | "NE")

SensingType ::= SEQUENCE {
 function [0] Function OPTIONAL, /* функция агрегирования для сенсорных данных */
 sensorType [1] PrintableString, /* тип сенсорных данных*/
 unit PrintableString /* единица измерения сенсорных данных
*/
}

SensingTypeList ::= SEQUENCE OF SensingType

ConditionType ::= SEQUENCE {
 type PrintableString, /* тип сенсорных данных */
 relationalOp RelationalOp,
 value PrintableString,
 logicalOp LogicalOp OPTIONAL
}

ConditionList ::= SEQUENCE OF ConditionType

InstantCmd ::= SEQUENCE {
 commandID INTEGER(0..MAX), /* идентификатор команды */
 targetIDList IDList, /* список идентификаторов ресурсов для выполнения команды */
 sensingTypeList SensingTypeList, /* список типов датчиков */
 actionTime UTCTime OPTIONAL, /* если установлено, то команда обрабатывается в указанное
время */
 conditionList ConditionList OPTIONAL /* список условий при наличии */
}
```

Таблица 11 — Информация сообщения команды мгновенного режима

Поле	Описание	Примечание
InstantCmd		
commandID	Идентификатор команды для сопоставления с командой из сообщения передачи сенсорных данных	—
targetIDList	Список идентификаторов ресурсов сенсорной сети, которые обрабатывают эту команду. Ресурсом может быть преобразователь, сенсорный узел, координатор PAN или шлюз	—
sensingTypeList	Список типов датчиков	—
actionTime	Период времени обработки команды	0 означает, что команда должна быть обработана после ее получения
conditionList	Список условий, которые должны быть проверены ресурсами перед отправкой сенсорных данных	Порядок проверки состояния — справа налево

9.3.2 Команда непрерывного режима

Команда непрерывного режима – это сообщение для обеспечения непрерывного восприятия сенсорных данных ресурсами сенсорной сети. Команда определяет идентификаторы целевого ресурса, типы датчиков, условия для проверки (при наличии), период, длительность и период времени (при необходимости).

В таблице 12 показана структура сообщения команды непрерывного режима, в таблице 13 показана информация сообщения команды непрерывного режима.

Таблица 12 — Структура сообщения команды непрерывного режима

```
ContinuousCmd ::= SEQUENCE {
 commandID INTEGER(0..MAX),
 targetIDList IDList,
 sensingTypeList SensingTypeList,
 actionTime UTCTime OPTIONAL, /* если установлено, то команда обрабатывается в указанное
время */
 period Time,
 duration [0] Time OPTIONAL,
 conditionList [1] ConditionList OPTIONAL
}
```

Таблица 13 — Информация сообщения команды непрерывного режима

Поле	Описание	Примечание
ContinuousCmd		
commandID	Идентификатор команды для сопоставления с командой из сообщения передачи сенсорных данных	–
targetIDList	Список идентификаторов ресурсов сенсорной сети, которые обрабатывают эту команду. Ресурсом может быть преобразователь, сенсорный узел, координатор PAN или шлюз	–
sensingTypeList	Список типов датчиков	–
actionTime	Период времени обработки команды	0 означает, что команда должна быть обработана после ее получения
period	Интервал времени сбора сенсорных данных	–
duration	Длительность сбора сенсорных данных. Если данное поле отсутствует, то сбор должен проводиться бесконечно долго	–
conditionList	Список условий, которые должны быть проверены ресурсами перед отправкой сенсорных данных	Порядок проверки состояния — справа налево

9.3.3 Команда режима событий

Команда режима событий — это сообщение для обеспечения восприятия сенсорных данных во время события. Команда определяет идентификаторы целевого ресурса, типы датчиков, условия для проверки и период времени (если необходимо).

В таблице 14 показана структура сообщения команды режима событий, в таблице 15 показана информация сообщения команды режима событий.

Таблица 14 — Структура сообщения команды режима событий

```
EventCmd ::= SEQUENCE {
 commandID INTEGER(0..MAX),
 targetIDList IDList,
 sensingTypeList SensingTypeList,
 actionTime UTCTime OPTIONAL, /* если установлено, то команда обрабатывается в указанное
время */
 conditionList ConditionList
}
```

Таблица 15 — Информация сообщения команды режима событий

Поле	Описание	Примечание
EventCmd		
commandID	Идентификатор команды для сопоставления с командой из сообщения передачи сенсорных данных	—
targetIDList	Список идентификаторов ресурсов сенсорной сети, которые обрабатывают эту команду. Ресурсом может быть преобразователь, сенсорный узел, координатор PAN или шлюз	—
sensingTypeList	Список типов датчиков	—
actionTime	Период времени обработки команды	0 означает, что команда должна быть обработана после ее получения
conditionList	Список условий, которые должны быть проверены ресурсами перед отправкой сенсорных данных	Порядок проверки состояния – справа налево

9.3.4 Команда остановки

Команда остановки предназначена для остановки обработки команды. Используя StopCmdReq, клиент сенсорной сети может запросить остановку обработки ContinuousCmd, EventCmd, оперирования сенсорными данными в режиме push, оперирования исполнительными устройствами или MonitoringCmd.

В таблице 16 показана структура сообщения команды остановки, в таблице 17 показана информация сообщения команды остановки.

Таблица 16 — Структура сообщения команды остановки

```
StopCmdReq ::= SEQUENCE {
 commandID INTEGER(0 .. MAX) /* отслеживается уникальность, чтобы избежать дублирования
*/
}

StopCmdRes ::= SEQUENCE {
 retCode RetCode
}
```

Таблица 17 — Информация сообщения команды остановки

Поле	Описание	Примечание
StopCmdReq		
commandID	Идентификатор команды для сопоставления с командой из сообщения передачи сенсорных данных	—
StopCmdRes		
retCode	Результат обработки StopCmdReq. Список значений может быть расширен	"SUCCESS", "BADREQUEST", "ERROR"

9.3.5 Команда исполнения

Команда исполнения — это сообщение целевому исполнительному устройству действовать в соответствии с параметрами исполнения.

В таблице 18 показана структура сообщения команды исполнения, в таблице 19 показана информация сообщения команды исполнения.

Таблица 18 — Структура сообщения команды исполнения

```
ActuationCmd ::= SEQUENCE {
 commandID INTEGER(0..MAX),
 targetIDList IDList, /* идентификатор целевого ресурса */
 actuatorType PrintableString,
 actionValue PrintableString
 actionTime UTCTime OPTIONAL, /* если установлено, то команда обрабатывается в указанное
время */
}

ActuationValue ::= SEQUENCE {
 targetID PrintableString,
 actionValue PrintableString
}

ActuationValueList ::= SEQUENCE {
 actuationValue SEQUENCE SIZE(1..MAX) OF ActuationValue
}

ActuationRpt ::= SEQUENCE {
 commandID INTEGER(0..MAX),
 actuationValueList ActuationValueList
}
```

Таблица 19 — Информация сообщения команды исполнения

Поле	Описание	Примечание
ActuationCmd		
commandID	Идентификатор команды для сопоставления с командой из сообщения передачи сенсорных данных	—
targetIDList	Список идентификаторов ресурсов сенсорной сети, которые обрабатывают эту команду. Ресурсом может быть преобразователь, сенсорный узел, координатор PAN или шлюз. Если targetIDList включает в себя идентификаторы шлюза, идентификаторы PAN или идентификаторы сенсорного узла, то это означает, что исполнительные устройства, которые являются членами указанного targetID, должны обрабатывать команду только в том случае, если сопоставлено значение actuatorType	—
actuatorType	Тип исполнительного устройства	—
actionValue	Параметры исполнения. Параметры определяются в зависимости от типа исполнительного устройства и могут быть структурированными для сложных исполнительных устройств. Детальное проектирование параметров выходит за рамки положений настоящего стандарта	—
actionTime	Период времени обработки команды	0 означает, что команда должна быть обработана после ее получения
ActuationRpt		
commandID	Идентификатор команды для сопоставления с командой из сообщения передачи сенсорных данных	—
actionValueList	Список результатов исполнения исполнительными устройствами	—

9.3.6 Команда мониторинга

Команда мониторинга — это сообщение статуса мониторинга сенсорных сетей. Реализация может происходить двумя способами. В первом способе MonitoringRpt запускается в режиме повтора без запроса от клиента сенсорной сети. В другом способе используются MonitoringCmd и MonitoringRpt. Способ является вопросом реализации и должен быть заранее объявлен клиентам сенсорной сети.

В таблице 20 показана структура сообщения команды мониторинга, в таблице 21 показана информация сообщения команды мониторинга.

Таблица 20 — Структура сообщения команды мониторинга

```
MonitoringType ::= PrintableString("BATTERY" | "LOCATION" | "ISALIVE")

MonitoringTypeList ::= SEQUENCE SIZE(1..MAX) OF MonitoringType

DataType ::= PrintableString("BOOLEAN" | "INTEGER" | "FLOAT" | "DOUBLE" | "STRING")

MonitoringCmd ::= SEQUENCE {
 commandID INTEGER(0..MAX),
 targetIDList IDList,
 monitoringTypeList MonitoringTypeList
}

MonitoringValue ::= SEQUENCE {
 monitoringType PrintableString,
 dataType DataType,
 unit PrintableString,
 value PrintableString
}

MonitoringValueList ::= SEQUENCE SIZE(1..MAX) OF MonitoringValue
 MonitoringRpt ::= SEQUENCE {
 commandID INTEGER(0 .. MAX) OPTIONAL,
 targetID PrintableString,
 monitoringValueList MonitoringValueList
}
```

Таблица 21 — Информация сообщения команды мониторинга

Поле	Описание	Примечание
MonitoringCmd		
commandID	Идентификатор команды для сопоставления с командой из сообщения передачи сенсорных данных	—
targetIDList	Список идентификаторов ресурсов сенсорной сети, которые обрабатывают эту команду. Ресурсом может быть преобразователь, сенсорный узел, координатор PAN или шлюз	—
monitoringTypeList	Список типов мониторинга	"BATTERY", "LOCATION", "ISALIVE"
MonitoringRpt		
commandID	Идентификатор команды для сопоставления с командой из сообщения передачи сенсорных данных	Идентификатор команды может быть пропущен в режиме push
targetID	Список идентификаторов ресурсов сенсорной сети, которые отправляют сообщение MonitoringRpt	—
monitoringValue-List	Список запрашиваемых значений мониторинга	—

9.3.7 Контроль ресурсов

Контроль ресурсов заключается в контроле сенсорной сети. С помощью ResourceCtrlReq клиент сенсорной сети запрашивает сброс, выключение, перезагрузку, запуск сенсорных данных в режиме push, остановку сенсорных данных в режиме push, обновление канала PAN и т. д.

В таблице 22 показана структура сообщения контроля ресурсов, в таблице 23 показана информация сообщения контроля ресурсов.

Таблица 22 — Структура сообщения контроля ресурсов

```
ResourceCtrlReq ::= SEQUENCE {
 targetID PrintableString,
 controlType SupportedCommandAttribute, /* свойство для контроля */
 controlValue PrintableString
}

ResourceCtrlRes ::= SEQUENCE {
 retCode RetCode,
 resultValue PrintableString
}
```

Таблица 23 — Информация сообщения контроля ресурсов

Поле	Описание	Примечание
ResourceCtrlReq		
targetID	Целевой идентификатор ресурса сенсорной сети датчика, который необходимо контролировать. Ресурсом может быть преобразователь, сенсорный узел, координатор PAN или шлюз	—
controlType	Тип контроля. Список значений может быть расширен	"ATTRIBUTE_SHUTDOWN" | "ATTRIBUTE_ RESET" | "ATTRIBUTE_ REBOOT" | "ATTRIBUTE_ START_SENSING" | "ATTRIBUTE_STOP_ SENSING" | "ATTRIBUTE_ PAN_ID_CHANGE" | "ATTRIBUTE_CHANNEL_ ID_CHANGE"
controlValue	Контрольное значение	В зависимости от controlType
ResourceCtrlRes		
retCode	Результат запроса. Список значений может быть расширен	"SUCCESS", "BADREQUEST", "ERROR", "UNDEFINED"
resultValue	Интерпретация результата	—

Примечание — Клиент сенсорной сети может инициировать сбор сенсорных данных в режиме push, используя ResourceCtrlReq с controlType = "ATTRIBUTE_START_SENSING". Клиент может прекратить сбор сенсорных данных в режиме push, используя ResourceCtrlReq с controlType = "ATTRIBUTE_STOP_SENSING".

9.3.8 Пользовательское сообщение

Пользовательские сообщения обеспечивают масштабируемость. Если существует определенная функция между конкретным клиентом сенсорной сети и конкретной сенсорной сетью, они могут обмениваться информацией с помощью UserDefinedMsg. Содержание сообщения предварительно определяется конкретным клиентом сенсорной сети и конкретной сенсорной сетью.

В таблице 24 показана структура пользовательского сообщения, в таблице 25 показана информация пользовательского сообщения.

Таблица 24 — Структура пользовательского сообщения

```
UserDefinedMsg ::= SEQUENCE {
 sourceID PrintableString, /* идентификатор отправителя сообщения */
 targetID PrintableString,
 message PrintableString
}
```

Таблица 25 — Информация пользовательского сообщения

Поле	Описание	Примечание
UserDefinedMsg		
sourceID	Идентификатор отправителя сообщения: идентификатор клиента сенсорной сети или идентификатор ресурса сенсорной сети	—
targetID	Идентификатор получателя сообщения: идентификатор клиента сенсорной сети или идентификатор ресурса сенсорной сети	—
message	Предопределенное пользовательское сообщение между клиентом сенсорной сети и сенсорной сетью	—

9.3.9 Уведомление об отклонении

Уведомление об отклонении означает, что сообщение, которое клиент сенсорной сети отправил в сенсорную сеть, не может быть обработано сенсорной сетью. Уведомление отправляют ресурсы сенсорной сети.

В таблице 26 показана структура сообщения уведомления об отклонении, в таблице 27 показана информация сообщения уведомления об отклонении.

Таблица 26 — Структура сообщения уведомления об отклонении

```
RejectCode ::= PrintableString("BADREQUEST" | "NOT_SUPPORTED_COMMAND" | "ERROR")
RejectNotify ::= SEQUENCE {
 commandID INTEGER(0 .. MAX),
 rejectCode RejectCode,
 rejectMessage PrintableString OPTIONAL
}
```

Таблица 27 — Информация сообщения уведомления об отклонении

Поле	Описание	Примечание
RejectNotify		
commandID	Идентификатор команды, отклоненной ресурсом сенсорной сети	—
rejectCode	Причина отклонения. Список значений может быть расширен	"BADREQUEST", "NOT_SUPPORTED_COMMAND", "ERROR"
rejectMessage	Интерпретация отклонения	—

9.3.10 Уведомление NAK

Уведомление NAK — это уведомление отправителя сообщения о наличии ошибки в сообщении. Отправителями уведомления могут быть клиенты сенсорной сети и ресурсы сенсорной сети.

В таблице 28 показана структура сообщения уведомления NAK, в таблице 29 показана информация сообщения уведомления NAK.

Таблица 28 — Структура сообщения уведомления NAK

```
NakNotify ::= SEQUENCE {
 message PrintableString,
 messageType MessageType OPTIONAL
}
```

Т а б л и ц а 29 — Информация сообщения уведомления NAK

Поле	Описание	Примечание
NakNotify		
messageType	Тип сообщения, которое вызвало уведомление	—
message	Интерпретация NAK	—

Приложение А
(справочное)

Описание сенсорной сети (пример)

А.1 Описание шлюза

```
- file: gateNodeDesc.asn1
- define interfaces between USN Middleware and G/W
- define gateway description

ISOIEC30128-GateNodeDesc DEFINITIONS IMPLICIT TAGS ::= BEGIN

EXPORTS GateNode;
IMPORTS Location, MonitoringMode, SupportedCommandList, SupportedCommandAttributeList FROM
ISOIEC30128-Types
  PanList FROM ISOIEC30128-PanDesc;

SupportedTransportProtocol ::= PrintableString("XML_OVER_TCP" | "XML_OVER_UDP" | "XML_
OVER_HTTP" | "TEXT_OVER_TCP")

SupportedTransportProtocolList ::= SEQUENCE OF SupportedTransportProtocol

SupportedTransportConnectionControlList ::= SEQUENCE OF SupportedTransportProtocol

GateNodeSupportedOperationList ::= SEQUENCE {
   supportedCommandList SupportedCommandList,
   supportedCommandAttributeList SupportedCommandAttributeList
}
GateNode ::= SEQUENCE {
  id          PrintableString, -- format: host name or ip
  url  [0] PrintableString OPTIONAL,
  manufacturer PrintableString OPTIONAL,
  productNo [1] PrintableString OPTIONAL,
  location Location OPTIONAL,
  dateTime [2] UTCTime OPTIONAL,
   supportedTransportProtocolList [3] SupportedTransportProtocolList OPTIONAL,
   supportedOperationList [4] GateNodeSupportedOperationList OPTIONAL,
  panList [5] PanList OPTIONAL,
  monitoringMode [6] MonitoringMode OPTIONAL,
  monitoringPeriod INTEGER(0 .. MAX)OPTIONAL -- in seconds
}
END
```

А.2 Описание PAN

```
- file: panDesc.asn1
- define PAN description

ISOIEC30128-PanDesc DEFINITIONS IMPLICIT TAGS ::= BEGIN

EXPORTS Pan, PanList;
IMPORTS UnsignedByte, SupportedCommandList, SupportedCommandAttributeList FROM
ISOIEC30128-Types
SensorNodeList FROM ISOIEC30128-SensorNodeDesc;

Pan ::= SEQUENCE {
  id NumericString, -- format: number
  topology [0] PanTopology OPTIONAL,
  protocolStack [1] PanProtocolStack OPTIONAL,
panChannel [2] UnsignedByte OPTIONAL,
supportedChannelList [3] PanChannelList OPTIONAL,
supportedTopologyList [4] PanTopologyList OPTIONAL,
```

```
supportedProtocolStackList [5] PanProtocolStackList OPTIONAL,
supportedOperationList [6] PanSupportedOperationList OPTIONAL,
sensorNodeList [7] SensorNodeList OPTIONAL
}
PanList ::= SEQUENCE SIZE(1..MAX) OF Pan
PanChannelList ::= SEQUENCE OF UnsignedByte
PanTopology ::= PrintableString("TREE" | "MESH" | "STAR")
PanTopologyList ::= SEQUENCE OF PanTopology
PanProtocolStack ::= PrintableString("802.15.4", "ZIGBEE" | "BLUETOOTH" | "LP_WIFI" |
"PROTOCOL_RES1" | "PROTOCOL_RES2" | "PROTOCOL_RES3" | "PROTOCOL_RES4" | "PROTOCOL_RES5")

PanProtocolStackList ::= SEQUENCE OF PanProtocolStack
PanSupportedOperationList ::= SEQUENCE {
   supportedCommandList SupportedCommandList,
   supportedCommandAttributeList SupportedCommandAttributeList
}
END
```

А.3 Описание сенсорного узла

```
- file: sensorNodeDesc.asn1
- define sensor node description
ISOIEC30128-SensorNodeDesc DEFINITIONS IMPLICIT TAGS ::= BEGIN
EXPORTS SensorNode, SensorNodeList;
IMPORTS MonitoringMode, Location, SupportedCommandList, SupportedCommandAttributeList FROM
ISOIEC30128-Types
   TransducerList FROM ISOIEC30128-TransducerDesc;
SensorNode ::= SEQUENCE {
  id NumericString, -- resource identification, format: number
  gid PrintableString, -- global identification
  monitoringMode [0] MonitoringMode OPTIONAL,
  monitoringPeriod [1] INTEGER(0 .. MAX) OPTIONAL,
  manufacturer [2] PrintableString OPTIONAL,
  productNo [3] PrintableString OPTIONAL,
  location [4] Location OPTIONAL,
  role [5] SensorNodeRole OPTIONAL,
  roleList [6] SensorNodeRoleList OPTIONAL,
  parentNodeList [7] ParentNodeList OPTIONAL,
   supportedOperationList [8] SensorNodeSupportedOperationList OPTIONAL,
  transducerList [9] TransducerList OPTIONAL
}
SensorNodeList ::= SEQUENCE SIZE(1..MAX) OF SensorNode
SensorNodeRole ::= PrintableString("COORDINATOR" | "ROUTER" | "LEAF")
SensorNodeRoleList ::= SEQUENCE OF SensorNodeRole
ParentNode ::= SEQUENCE{
  parentNode INTEGER(0..MAX)
}
ParentNodeList ::= SEQUENCE SIZE(1..MAX) OF ParentNode
SensorNodeSupportedOperationList ::= SEQUENCE {
   supportedCommandList SupportedCommandList,
   supportedCommandAttributeList SupportedCommandAttributeList
}
END
```

А.4 Описание преобразователя

```
- file: transducerDesc.asn1
- define transducer description
ISOIEC30128-TransducerDesc DEFINITIONS IMPLICIT TAGS ::= BEGIN
EXPORTS TransducerList, Transducer;
IMPORTS DataType, SupportedCommandList, SupportedCommandAttributeList FROM ISOIEC30128-Types;
Transducer::= CHOICE {
sensor [0]Sensor,
actuator [1] Actuator
```

```
}
TransducerList::= SEQUENCE SIZE(1..MAX) OF Transducer
Level ::= INTEGER(1 .. MAX)
LevelList ::= SEQUENCE OF Level
Sensor ::= SEQUENCE {
id NumericString, -- transducer identification, format: number
manufacturer [0] PrintableString OPTIONAL,
productNo PrintableString OPTIONAL,
range [1] Range OPTIONAL,
level [2] LevelList OPTIONAL,
supportedOperationList TransducerSupportedOperationList,
transducerType [3] TransducerType DEFAULT "SENSOR",
type PrintableString(FROM ("A".."Z" | "a".."z")),
unit PrintableString(FROM ("A".."Z" | "a".."z")),
dataType DataType
}Actuator ::= SEQUENCE {
id NumericString, -- transducer identification, format: number
manufacturer [0] PrintableString OPTIONAL,
productNo PrintableString OPTIONAL,
range [1] Range OPTIONAL,
level [2] LevelList OPTIONAL,
supportedOperationList TransducerSupportedOperationList,
transducerType [3] TransducerType DEFAULT "SENSOR",
type PrintableString(FROM ("A".."Z" | "a".."z" )),
unit PrintableString(FROM ("A".."Z" | "a".."z" )),
dataType DataType
}
Range ::= SEQUENCE{
min REAL(0..MAX),
max REAL(0..MAX),
offset INTEGER(1 .. MAX) OPTIONAL}
TransducerType ::= PrintableString("SENSOR" | "ACTUATOR")
TransducerSupportedOperationList::= SEQUENCE {
supportedCommandList SupportedCommandList,
supportedCommandAttributeList SupportedCommandAttributeList
}
END
```

Приложение В
(справочное)

Тип датчика и единица измерения (пример)

```
<?xml version="1.0" encoding="euc-kr"?>
<xsd:schema xmlns:xsd="
gost.e-ecolog.ru
"
xmlns:utran="http://www.isoiec30128.org/cosmos/resource/unitOfTransducerType"
targetNamespace="http://www.isoiec30128.org/cosmos/resource/unitOf TransducerType"
  elementFormDefault="qualified">

   <xsd:simpleType name="tSensorType">
     <xsd:restriction base="xsd:string">
    <!--  ============================  -->
    <!—Security -->
    <!--  ============================  -->
    <xsd:enumeration  value="EMF"/>
    <xsd:enumeration  value="VOCS"/>
    <xsd:enumeration  value="CO2"/>
    <xsd:enumeration  value="GEIGER_COUNTER"/>
    <xsd:enumeration  value="MAGNETIC"/>
    <xsd:enumeration  value="HEAT_RAY"/>
    <xsd:enumeration  value="INFRARED_RAY"/>

    <!--  ============================  -->
    <!-- Environment -->
    <!--  ============================  -->
    <xsd:enumeration  value="TEMPERATURE"/>
    <xsd:enumeration  value="ILLUMINATION"/>
    <xsd:enumeration  value="NOISE"/>
    <xsd:enumeration  value="HUMIDITY"/>
    <xsd:enumeration  value="WIND_SPEED"/>
    <xsd:enumeration  value="ULTRA_VIOLET"/>

    <!--  ============================  -->
    <!—Health -->
    <!--  ============================  -->
    <xsd:enumeration  value="DIASTOLIC_BLOOD_PRESSURE"/>
    <xsd:enumeration  value="SYSTOLIC_BLOOD_PRESSURE"/>
    <xsd:enumeration  value="BLOOD_GLUCOSE"/>
    <xsd:enumeration  value="WEIGHT"/>
    <xsd:enumeration  value="WEIGHT_CHANGE"/>
    <xsd:enumeration  value="BODY_FAT"/>
    <xsd:enumeration  value="BODY_TEMPERATURE"/>
    <xsd:enumeration  value="PASSOMETER"/>
    <xsd:enumeration  value="OXYGEN_SATURATION"/>
    <xsd:enumeration  value="PULSE"/>

    <!--  ============================  -->
    <!—Miscellaneous -->
    <!--  ============================  -->
    <xsd:enumeration  value="LONGITUDE"/>
    <xsd:enumeration  value="LATITUDE"/>
    <xsd:enumeration  value="ALTITUDE"/>
    <xsd:enumeration  value="BATTERY"/>
    <xsd:enumeration  value="WIND_DIRECTION"/>
    <xsd:enumeration  value="RAIN_FALL"/>
    <xsd:enumeration  value="PRESSURE"/>
     </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="tSensorUnit">
```

```
<xsd:restriction base="xsd:string">
 <xsd:enumeration value="SIEVERT"/>
 <xsd:enumeration value="WATT_PER_CM2"/>
 <xsd:enumeration value="MILLI_GAUSE"/>
 <xsd:enumeration value="PARTS_PER_MILLION"/>
 <xsd:enumeration value="MILLIMETERS_OF_MERCURY"/>
 <xsd:enumeration value="MILLIGRAMS_PER_DECILITER"/>
 <xsd:enumeration value="KILLOGRAMS"/>
 <xsd:enumeration value="PERCENTAGE"/>
 <xsd:enumeration value="CELSIUS"/>
 <xsd:enumeration value="LUX"/>
  <xsd:enumeration value="DECIBEL"/>
  <xsd:enumeration value="COUNT"/>
  <xsd:enumeration value="PERCENTAGE_PER_BITPERMINUTES"/>
  <xsd:enumeration value="BEAT_PER_MINUTE"/>
  <xsd:enumeration value="METER_PER_SECOND"/>
  <xsd:enumeration value="NO_UNIT"/>
  <xsd:enumeration value="MILLIMETERS_PER_HOUR"/>
  <xsd:enumeration value="HECTOPASCAL"/>
  <xsd:enumeration value="ANGLE"/>
 </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="tActuatorType">
 <xsd:restriction base="xsd:string">
  <xsd:enumeration value="FAUCET"/>
  <xsd:enumeration value="FAN"/>
  <xsd:enumeration value="MONITOR"/>
 </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="tActuatorUnit">
 <xsd:restriction base="xsd:string">
  <xsd:enumeration value="DIGITAL"/>
  <xsd:enumeration value="ANALOG"/>
  <xsd:enumeration value="STRING"/>
 </xsd:restriction>
</xsd:simpleType>
</xsd:schema>
```

Приложение ДА
(справочное)

Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте

Таблица ДА.1

Обозначение ссылочного национального стандарта	Степень соответствия	Обозначение и наименование ссылочного международного стандарта
ГОСТ Р 56947—2016/ISO/IEC/IEEE 21450:2010	IDT	ISO/IEC/IEEE 21450:2010 «Информационные технологии. Интерфейс интеллектуального преобразователя для датчиков и исполнительных устройств. Общие функции, протоколы взаимодействия и форматы электронной таблицы данных преобразователя (ЭТДП)»
ГОСТ Р 57668–2017 (ИСО 19115-1:2014)	MOD	ISO 19115-1:2014 «Географическая информация. Метаданные. Часть 1. Основные положения»
Примечание – В настоящей таблице использованы следующие условные обозначения степени соответствия стандартов: - IDT – идентичные стандарты; - MOD – модифицированные стандарты.		

Приложение ДБ
(справочное)

Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта

Таблица ДБ.1

Структура настоящего стандарта	Структура международного стандарта ISO/IEC 30128:2014
1 Область применения	1 Область применения
2 Нормативные ссылки	2 Нормативные ссылки
3 Термины и определения	3 Термины и определения
4 Сокращения	4 Сокращения
*	5 Соглашения
5 Обзор приложений сенсорных сетей (раздел 6)	6 Обзор приложений сенсорных сетей
6 Возможности сенсорной сети (раздел 7)	7 Возможности сенсорной сети
7 Защищенность (раздел 8)	8 Защищенность
8 Модель сенсорных данных и метаданных (раздел 9)	9 Модель сенсорных данных и метаданных
9 Спецификация общего интерфейса приложения универсальной сенсорной сети (раздел 10)	10 Спецификация общего интерфейса приложения универсальной сенсорной сети
Приложение А Описание сенсорной сети (пример)	Приложение А (справочное) Описание сенсорной сети (пример)
Приложение Б Тип датчика и единица измерения (пример)	Приложение В (справочное) Тип датчика и единица измерения (пример)
Приложение ДА Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте	Библиография
Приложение ДБ Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта	
Библиография	

* Данный раздел исключен в связи с нецелесообразностью приведения его в настоящем стандарте.

Примечание — После заголовков разделов настоящего стандарта приведены в скобках номера аналогичных им разделов международного стандарта.

Библиография

[1] ISO/IEC 29182-2:2013 Information technology — Sensor networks: Sensor Network Reference Architecture (SNRA) — Part 2: Vocabulary and terminology (Информационные технологии. Сенсорные сети. Эталонная архитектура для сенсорных сетей (SNRA). Часть 2. Словарь и терминология)

[2] ISO/IEC/IEEE 21451-4 Information technology — Smart Transducer Interface for Sensors and Actuators — Mixed-Mode Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats [Информационные технологии. Интеллектуальный интерфейс преобразователей для датчиков и исполнительных устройств. Часть 4. Смешанные протоколы связи и форматы хранения данных преобразователя (TEDS)]

[3] ISO/IEC/IEEE 21451-7 Information technology — Standard for a Smart Transducer Interface for Sensors and Actuators — Transducers to Radio Frequency Identification (RFID) Systems Communication Protocols and Transducer Electronic Data Sheet Formats [Информационные технологии. Интеллектуальный интерфейс преобразователей для датчиков и исполнительных устройств. Часть 7. Протоколы связи преобразователя с системами радиочастотной идентификации и форматы хранения данных преобразователя (TEDS)]

[4] ISO 19123:2005 Geographic information — Schema for coverage geometry and functions (Географическая информация. Схема геометрии и функций покрытия)

[5] ISO/IEC 30128:2014 Information technology — Sensor networks — Generic Sensor Network Application Interface (Информационная технология. Сенсорные сети. Основной интерфейс приложения сенсорной сети)

[6] ISO 19156:2011 Geographic information — Observations and measurements (Географическая информация. Наблюдения и измерения)

УДК 004.738:006.354 ОКС 35.110

Ключевые слова: информационные технологии, сенсорные сети, интерфейс приложения, сенсорные данные

БЗ 9—2020

Редактор *Н.В. Таланова*
Технический редактор *В.Н. Прусакова*
Корректор *О.В. Лазарева*
Компьютерная верстка *Е.О. Асташина*

Сдано в набор 20.07.2020. Подписано в печать 11.08.2020. Формат 60×84 1/8. Гарнитура Ариал.
Усл. печ. л. 4,18. Уч.-изд. л. 3,78.
Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

ИД «Юриспруденция», 115419, Москва, ул. Орджоникидзе, 11.
gost.e-ecolog.ru gost.e-ecolog.ru
Создано в единичном исполнении во ФГУП «СТАНДАРТИНФОРМ» для комплектования Федерального информационного фонда стандартов, 117418 Москва, Нахимовский пр-т, д. 31, к. 2.
gost.e-ecolog.ru gost.e-ecolog.ru

gost.e-ecolog.ru