

Многие инженеры автоматизации обращали внимание на «анархию протоколов», которая сложилась на рынке систем учета энергоресурсов. Количество протоколов, используемых в таких системах, равно (а скорее даже превышает) количеству производителей устройств – каждый из них использует собственный протокол обмена. При этом электросчетчики измеряют одни и те же параметры электросети, теплосчетчики – параметры теплосети, приборы используют одни и те же интерфейсы связи – но почему же протоколы у всех разные? Неужели нельзя использовать какой-либо стандартный протокол? О таком протоколе и пойдет речь в данной статье.

Главная причина того, что производители устройств систем учета разрабатывают собственные протоколы в том, что стандартные промышленные протоколы мало подходят для данной отрасли. Возьмем, к примеру, открытый и распространённый протокол Modbus. Он имеет целый ряд недостатков:

- Не стандартизированы типы передаваемых данных – по протоколу идут 2 байтовые регистры, за которыми может скрываться что угодно, и как-либо идентифицировать получаемые данные невозможно.
- Нет возможности передачи архивных данных. Да, есть возможность передачи файлов функцией 0x14, но она сложна в реализации и не очень хорошо подходит для передачи, например, часовых срезов.
- Отсутствие хоть какой-то безопасности – в Modbus отсутствует аутентификация и шифрование, что является важным для систем учета.

Несмотря на это, ряд производителей используют транспорт Modbus в качестве протокола передачи данных – например компания Взлет. Но даже в этом случае используется собственная модификация протокола – например, для передачи архива используется нестандартная функция 65, с собственной структурой кадра. Таким образом, использовать Modbus для систем учета не представляется возможным в полной мере.

Необходим новый стандарт, который был бы изначально ориентирован на данную отрасль.

Впервые данной проблемой озаботились в Европе в конце 90-х годов. Для ее решения была создана ассоциация **DLMS** (Distribution Line Message Specification), со штаб-квартирой в городе Цуг (Швейцария).



Целью организации было разработать новый протокол, способный решать все задачи по передаче данных в системах учета и стать стандартом отрасли. Протокол получил название DLMS и имеет следующие особенности:

- Возможность работы со всеми приборами учета – несмотря на то, что изначально протокол разрабатывался для учета электроэнергии, его функционал позволяет использоваться в системах учета воды, тепла и газа.
- Фиксированная структура кадра, содержащая ряд дополнительных атрибутов – номер клиента, номер сервера, идентификатор сообщения. Это позволяет исключить «перепутывание» запросов, и сделать гибкую модель авторизации.
- Различные варианты аутентификации – в зависимости от уровня подключения ограничивается доступ к тем или иным параметрам.
- Современные механизмы шифрования – если обмен идет по не защищенным линиям связи, есть возможность зашифровать данные и исключить их перехват или подмену.
- Объектно-ориентированный подход – в DLMS данные хранятся и передаются в виде объектов-классов (IC – interface class), которые имеют определенные атрибуты (шкала, единица измерения, время фиксации). Разные типы классов нацелены на хранение и передачу разных параметров устройства – абстрактных данных, текущих значений, значений с меткой времени (например, фиксация максимума мощности) или архивов.
- Передача типа данных вместе со значением – при запросе возвращается не только значение, но также тип данных, в котором он представлен. Это позволяет производителю устройства хранить данные в удобном для него формате, а клиентскому ПО корректно идентифицировать полученные данные.
- Возможность получения из устройства полного списка поддерживаемых параметров.
- Жестко фиксированные идентификаторы параметров – это, пожалуй, главное преимущество протокола. Остановится на них подробнее.

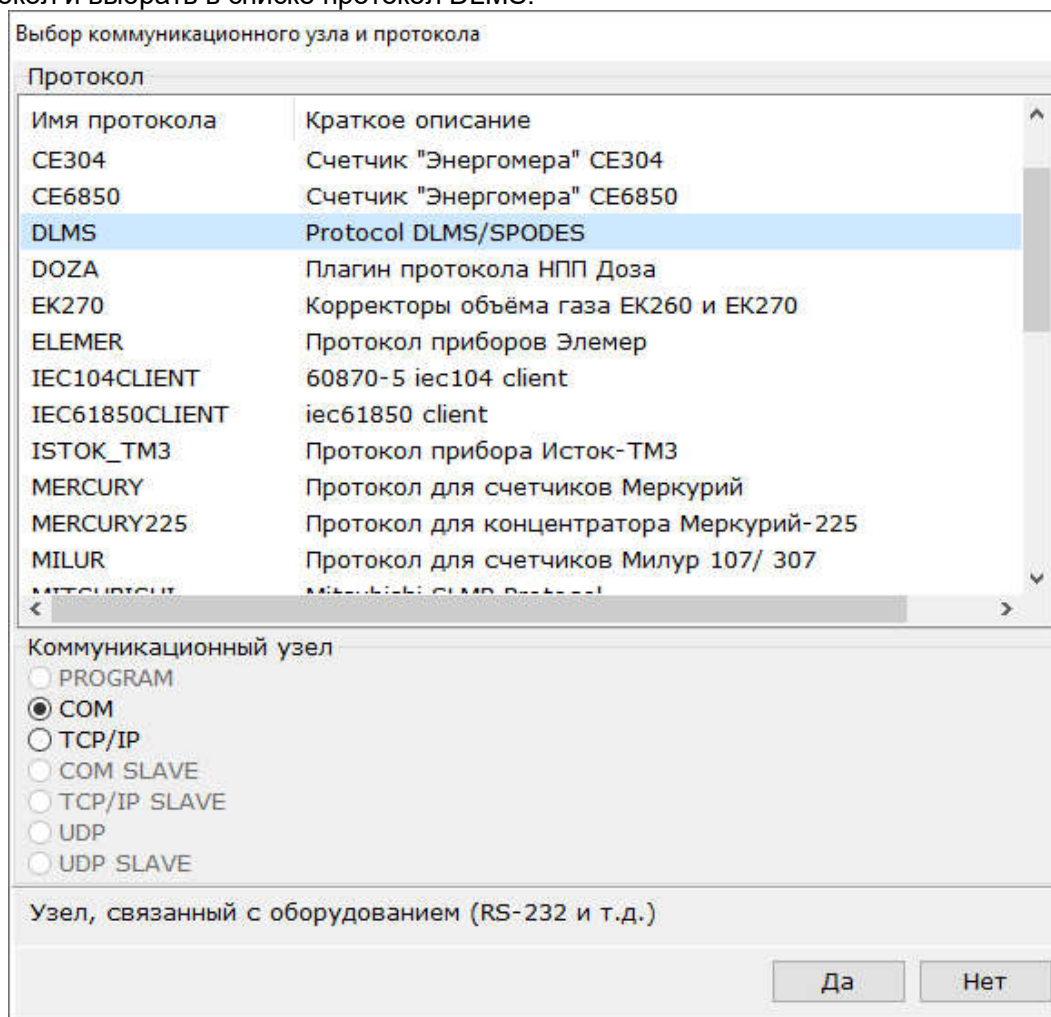
Все параметры (объекты) прибора имеют специальный идентификатор – OBIS код. OBIS код представляет собой 6-значное число, разделенное точками (A.B.C.D.E.F), например - 1.0.31.7.0.255. Каждый из разрядов числа определяет какое-либо свойство параметра устройства. Например, атрибут A определяет вид энергии: 0 – абстрактные параметры (время, производитель), 1 – электроэнергия, 4 – тепло, 7 – газ и т.д. Атрибут C – назначение параметра, например, для электросчетчиков, 1 – положительная активная мощность по всем фазам, 34 – частота по фазе 1. Все эти параметры прописаны в стандарте, и производитель устройства **обязан** использовать для параметров прибора соответствующие им идентификаторы. Что это означает на практике? К примеру, параметр электросчетчика – «ток фазы А», в стандарте за ним закреплен OBIS - 1.0.31.7.0.255, независимо от производителя устройства, параметр «ток фазы А» всегда будет иметь именно такой идентификатор. Таким образом можно включать приборы различных производителей в единую систему учета и не беспокоится об адресации параметров – они жестко унифицированы.

В настоящий момент стандарт DLMS активно применяется в системах учета в Европе, Азии, Ближнем Востоке и поддерживается большинством производителей устройств. Среди компаний поддерживающих протокол – ABB, Carlo Gavazzi, Honeywell, Socomes, Satec (всего свыше 1000 компаний).

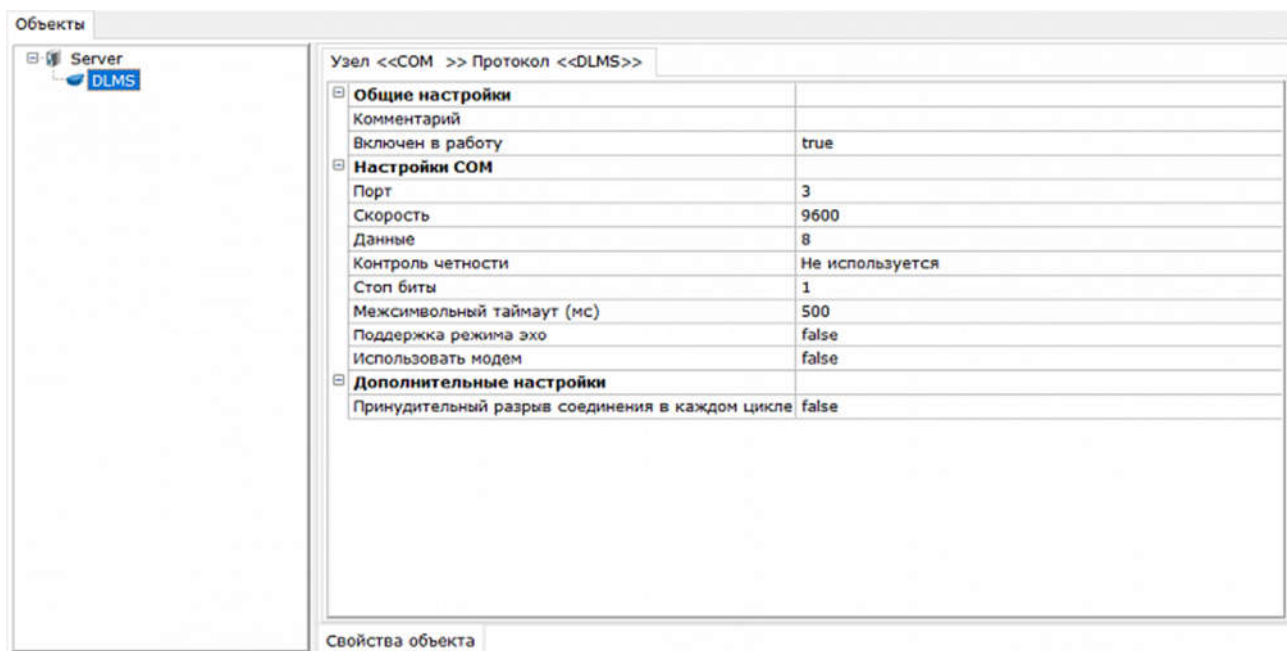
В России стандарт DLMS в настоящий момент продвигается ПАО «Россети» - главным оператором электросетей России. Ее усилиями стандарт DLMS был переведен на русский язык (текст можно посмотреть [здесь](#)) и назван СПОДЭС – Спецификация Протокола Обмена Данными Электронных Счетчиков. По сути это все тот же протокол DLMS, но имеющий ряд небольших отличий (не противоречащих основному стандарту) - комбинация номеров клиент-сервер определяют уровень доступа к параметрам счетчика (публичный клиент, считыватель показаний, конфигуратор), добавлены ряд национальных параметров, не входящих в основной стандарт – коэффициенты реактивной мощности, напряжение прямой и обратной последовательности и т.д.

В настоящий момент в России счетчики электроэнергии с протоколом DLMS поддерживают компании Инкотекс, Энергомера, ИЭК, НПО «МИР». Также ряд компаний находятся в процессе поддержки протокола. Поскольку на наших глазах происходит историческое событие, которое должно положить конец «анархии протоколов», мы также решили не оставаться в стороне и первыми в стране выпустить OPC сервер с поддержкой протокола DLMS/СПОДЭС – теперь данный протокол входит в перечень поддерживаемых Multi-Protocol MasterOPC сервера.

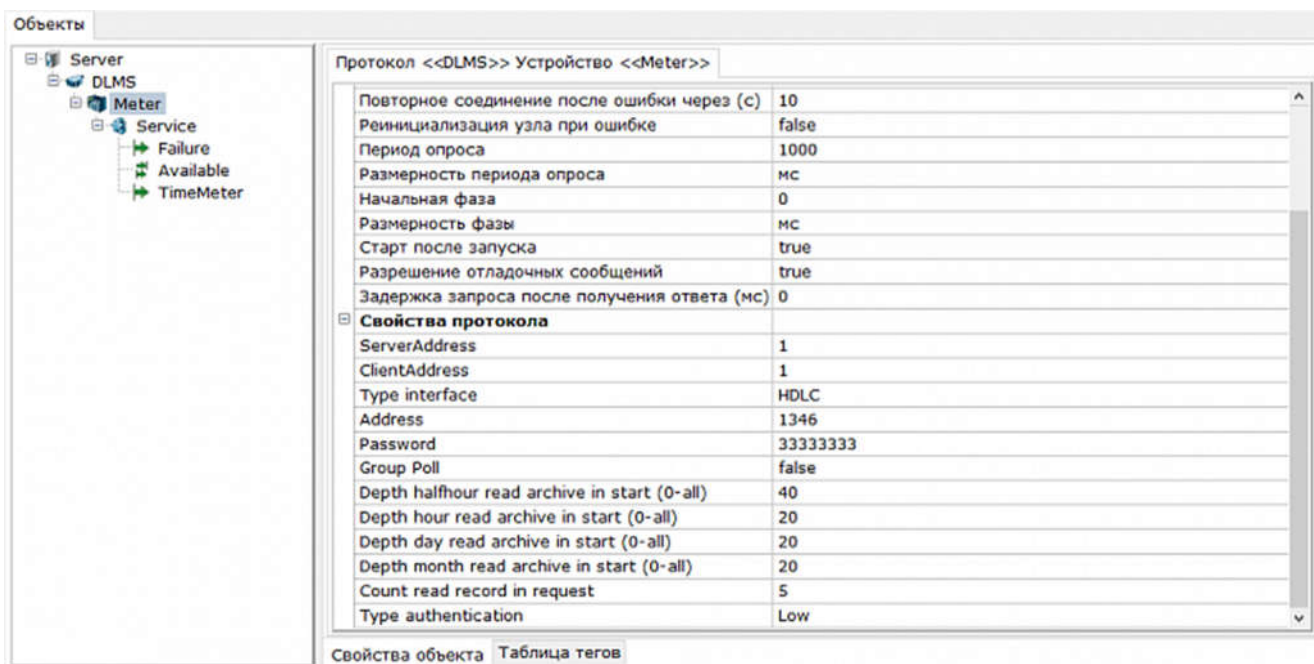
Чтобы подключить устройство с протоколом DLMS к Multi-Protocol MasterOPC нужно через контекстное меню добавить протокол и выбрать в списке протокол DLMS.



Выберите нужный тип подключения – напрямую, через RS-485 или по TCP – через конвертер Ethernet-COM. Затем задайте параметры связи – IP адрес или номер порта.



Затем через контекстное меню добавляется устройство. Оно уже содержит ряд сервисных тегов – **отказ**, **включение/отключение опроса**, а также **текущее время** счетчика. В настройках устройства нужно задать его параметры – адрес сервера и клиента, физический адрес на шине, уровень безопасности и пароль. Все эти параметры берутся из настроек устройства. Также можно включить режим шифрования, если устройство его поддерживает.



Затем мы рекомендуем сразу запустить OPC в режим исполнения и убедиться, что время счетчика считывается успешно, а значит и связь с устройством устанавливается.

Объекты

- Server
 - DLMS
 - Meter
 - Service
 - Failure
 - Available
 - TimeMeter

Device - Meter : Meter device

Теги

Идентификатор	Регистр	Адрес в регистре	Значение	Качество	Время
DLMS.Meter.Service.Failure			false	GOOD	2020-08
DLMS.Meter.Service.Available			true	GOOD	2020-08
DLMS.Meter.Service.TimeMeter			2017-05-11 11:42:05.000	GOOD	2020-08

Системные сообщения | Трассировка ввода-вывода | Сообщения протоколов и скриптов

Режим вывода: Запущен Фильтр: Meter

```

2020-08-23 9:42:23.932 DLMS.Meter:read start
2020-08-23 9:42:23.732 DLMS.Meter:read done
2020-08-23 9:42:22.998 DLMS.Meter:Current time=2017.05.11 11:42:05
2020-08-23 9:42:22.295 DLMS.Meter:Read OBIS 0.0.1.0.0.255 Index 2
2020-08-23 9:42:20.933 DLMS.Meter:read start
  
```

После этого можно приступить к добавлению тегов опроса. Можно добавить тег вручную используя контекстное меню устройства, и затем задать его параметры – **тип объекта**, **OBIS код**, **атрибут**.

Объекты

- Server
 - DLMS
 - Meter
 - Service
 - Failure
 - Available
 - TimeMeter
 - Tag1

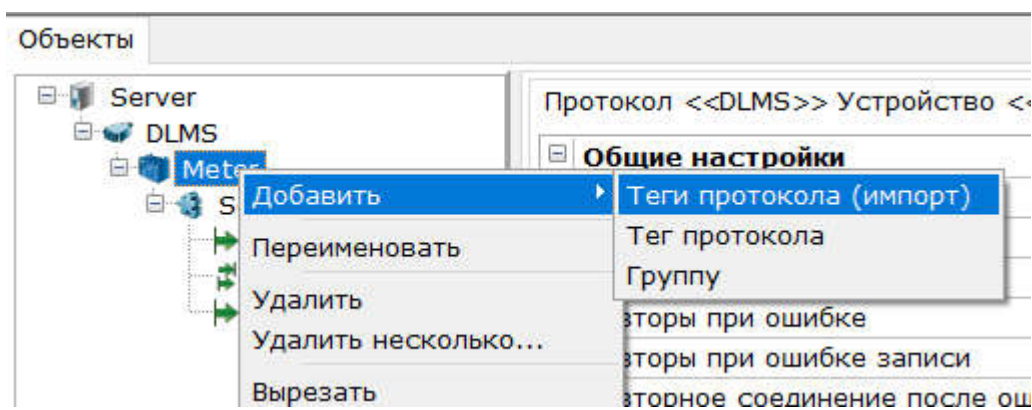
Устройство <<Meter>> Тег <<Tag1>>

Общие настройки	
Включен в работу	true
Комментарий	
Тип данных в сервере	int32
Тип доступа	ReadOnly
Сервис	
Пересчет (A*X + B)	false
Ограничение минимального значения	false
Ограничение максимального значения	false
HDA	
HDA доступ	false
Свойства протокола	
Logical Name (OBIS)	1.0.31.7.0.255
DLMS Type Object	Register (IC:3)
Scale	1.0000000000

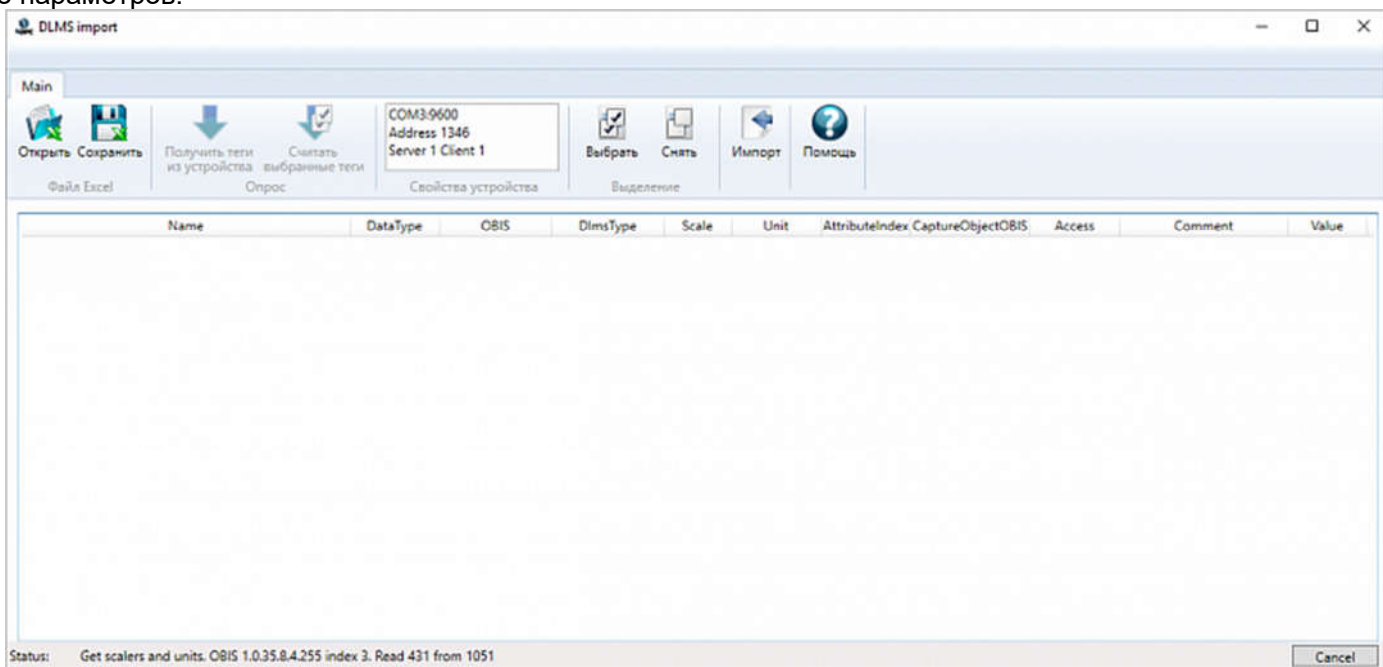
Свойства объекта | Таблица тегов

Однако создавать несколько десятков, а то и сотен параметров – трудоемкая задача, которую мы хотели максимально упростить. В случае с DLMS, изначально у нас был вариант сделать шаблонную конфигурацию, которая содержала бы основные теги (они, как упоминалось ранее, жестко фиксированы в стандарте). Однако, хоть OBIS коды и фиксированы, но типы данных и шкалу каждый производитель может использовать по своему усмотрению – один производитель передает напряжение как float число, а другой – как целое со смещением точки. Конечно, можно эти параметры прочитать при запуске сервера, но тогда увеличится время первого опроса. Кроме того, счетчики могут отличаться по составу тегов – бывают однофазные и трехфазные модели, с 4 тарифами и с 8 (и даже больше), с функциями анализатора качества сети и без них. Поэтому мы решили сделать более сложный в реализации, но более эффективный вариант – поскольку DLMS позволяет считывать состав тегов, и его атрибуты, была разработана утилита импорта, считывающая конфигурацию параметров прямо из устройства.

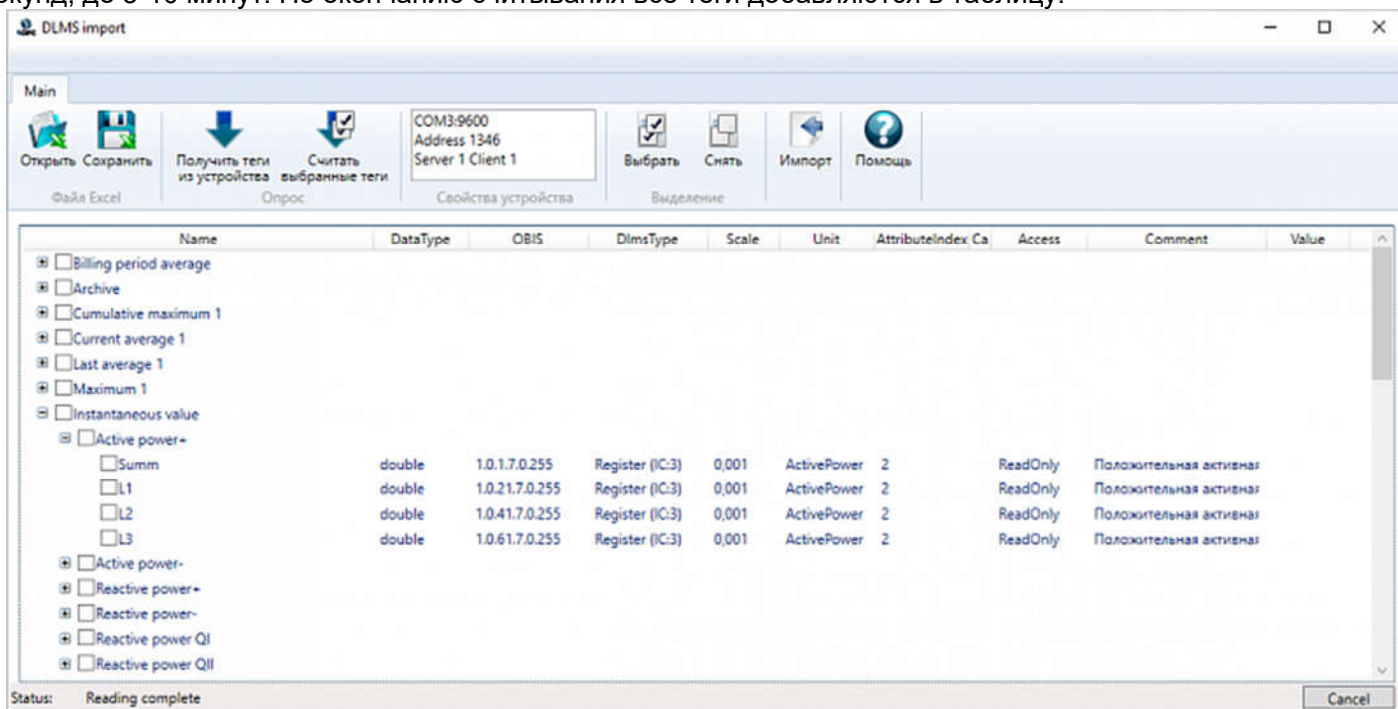
Для ее запуска необходимо вызвать контекстное меню устройства – Теги протокола (импорт).



Утилита имеет интерфейс, схожий с [представленной недавно](#) утилитой импорта тегов VASnet. Для того чтобы считать теги из устройства, нужно нажать на кнопку **Получить теги из устройства** - параметры подключения утилита автоматически берет из настроек OPC сервера. Если с устройством есть связь, начнется считывание его параметров.

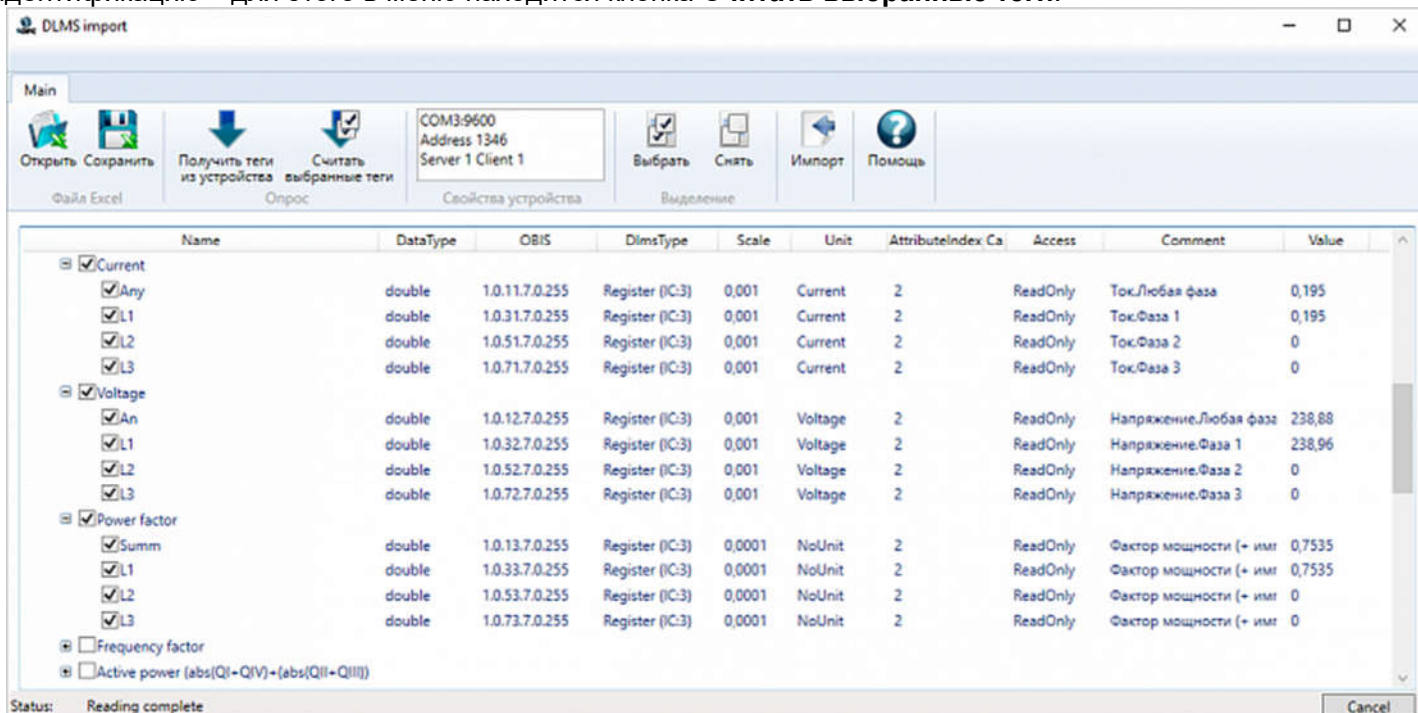


В зависимости от количества параметров в устройстве, время считывания может варьироваться от 20-30 секунд, до 5-10 минут. По окончании считывания все теги добавляются в таблицу:

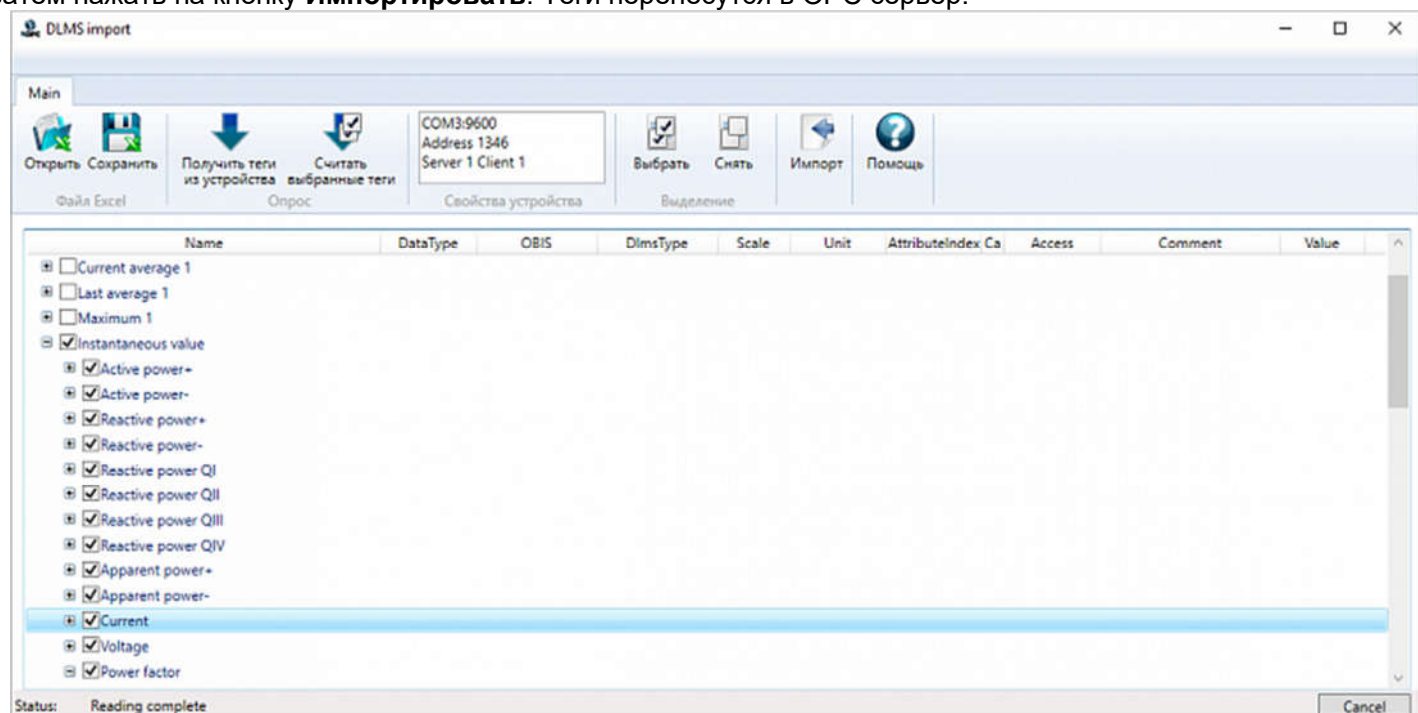


Поскольку OBIS коды фиксированы, выполняется автоматическое определение его назначения, с последующим разбиением тегов на группы – тип измерения, вид энергии, номер тарифа и т.д. Помимо

простого считывания наличия параметра можно сразу же выполнить опрос его значения, что облегчит его идентификацию – для этого в меню находится кнопка **Считать выбранные теги**.



Поскольку считывание параметров из устройства может занимать значительно время, мы реализовали возможность экспорта и импорта в Excel. Таким образом считав единожды теги из устройства, можно сохранить их в Excel файл, и впоследствии работать с ним. При этом, есть возможность с помощью Excel выполнить правку файла – исправить имена тегов, изменить их иерархию, локализовать комментарии, т.е. сделать собственный шаблон устройства, который максимально удовлетворяет вашей задаче. Теперь необходимо отметить нужные вам теги, используя флаги в дереве и/или групповое выделение, а затем нажать на кнопку **Импортировать**. Теги перенесутся в OPC сервер.



При этом у тегов, связанных с получением архивов, автоматически включается режим HDA. Тип архива (получасовой, часовой, суточный, месячный) определяется драйвером автоматически в момент старта.

Объекты

- Server
 - DLMS
 - Meter
 - Service
 - Archive
 - Monthly energy profile
 - Total integral
 - Hour energy profile
 - Total integral
 - Active power-
 - SummT0**
 - SummT1
 - SummT2
 - SummT3
 - SummT4
 - Active power+
 - Reactive power-
 - Reactive power+
 - Time integral 5
 - Daily energy profile
 - Daily energy profile
 - Instantaneous value

Устройство <<Meter>> Ter <<SummT0>>

| | |
|--|--|
| Общие настройки | |
| Включен в работу | true |
| Комментарий | Отрицательная активная мощность. Сумма фаз |
| Тип данных в сервере | uint32 |
| Тип доступа | ReadOnly |
| Сервис | |
| Пересчет (A*X + B) | false |
| Ограничение минимального значения | false |
| Ограничение максимального значения | false |
| HDA | |
| HDA доступ | true |
| Количество записей в архиве (100 - 4080) | 4080 |
| Автоматическая запись | false |
| Запись по изменению значения тега | false |
| Свойства протокола | |
| Logical Name (OBIS) | 1.0.99.1.2.255 |
| DLMS Type Object | ProfileGeneric (IC:7) |
| Capture Object OBIS | 1.0.2.8.0.255 |
| Scale | 1.0000000000 |

Свойства объекта Таблица тегов

Теперь можно запускать OPC сервер в режим исполнения.

Объекты

- Server
 - DLMS
 - Meter
 - Service
 - Archive
 - Instantaneous value

Group - Instantaneous value : Instantaneous value

Теги

| Идентификатор | Регион | Адрес в реги... | Значение | Качество | Время | Тип в сер... | Доступ | Комме... |
|----------------------------------|-----------------|-----------------|---------------|----------|--------------|--------------|----------|----------|
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.1.7.0.255 | 41.6000019759 | GOOD | 2020-08-2... | double | ReadOnly | Положа |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.21.7.0.255 | 41.6000019759 | GOOD | 2020-08-2... | double | ReadOnly | Положа |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.41.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Положа |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.61.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Положа |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.2.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Отриц |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.22.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Отриц |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.42.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Отриц |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.62.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Отриц |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.3.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Положа |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.23.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Положа |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.43.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Положа |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.63.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Положа |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.4.7.0.255 | 6.3000002992 | GOOD | 2020-08-2... | double | ReadOnly | Отриц |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.24.7.0.255 | 6.3000002992 | GOOD | 2020-08-2... | double | ReadOnly | Отриц |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.44.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Отриц |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.64.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Отриц |
| DLMS.Meter.Instantaneous valu... | Register (IC:3) | 1.0.5.7.0.255 | 0 | GOOD | 2020-08-2... | double | ReadOnly | Реакти |

Системные сообщения Трассировка ввода-вывода Сообщения протоколов и скриптов

Режим вывода: Запущен Филтр: Instantaneous value

Мы протестировали наш OPC сервер с основными устройствами, представленными на российском рынке - IEC Star128/328, Инкотекс Меркурий 234, Энергомера CE308/CE208. Со всеми устройствами OPC работал стабильно, во всех режимах работы.