



CoDeSys OPC-Server V2.0

Установка и использование

Версия документа 1.8

СОДЕРЖАНИЕ

1	КРАТКОЕ ВВЕДЕНИЕ В OPC	3
2	УСТАНОВКА OPC-СЕРВЕРА CODESYS	5
2.1	Структура необходимых для работы файлов	5
2.2	Установка и регистрация	5
3	НАСТРОЙКА OPC-СЕРВЕРА CODESYS	6
3.1	Среда программирования CoDeSys: генерация описаний переменных проекта	6
3.2	Конфигурирование OPC-сервера с помощью утилиты OPCconfig.exe	7
3.2.1	Меню 'File'	7
3.2.2	Меню 'Edit'	8
3.2.3	Настройка OPC-сервера при работе с одним ПЛК (Single PLC):	8
3.2.4	Настройка OPC-сервера для нескольких ПЛК (Multi PLC)	11
3.2.5	Записи в реестре (пример)	12
3.3	Запуск CoDeSysOPC.exe	13
4	ПРИМЕР INI-ФАЙЛА ДЛЯ КОНФИГУРАЦИИ OPC-СЕРВЕРА	14
5	ДИАГНОСТИКА И УСТРАНЕНИЕ ОШИБОК	16
5.1	Средства диагностики ошибок OPC:	16
5.2	Советы по устранению ошибок	16
5.3	Типичные ошибки	17
	ИСТОРИЯ ИЗМЕНЕНИЙ	18

1 Краткое введение в OPC

OPC¹ - это программная технология, предоставляющая стандартизированный интерфейс доступа к данным технологических процессов.

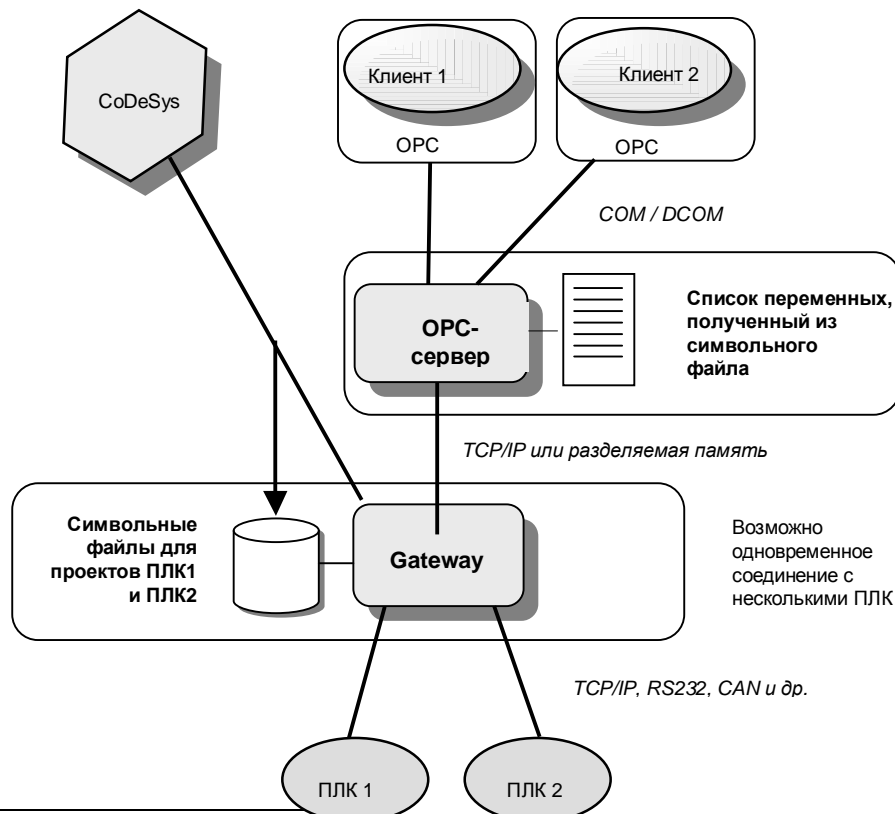
Технология OPC построена на базе Microsoft **COM/DCOM**². В основном OPC используется для чтения и записи данных ПЛК из различных программ (OPC-клиентов), таких как программы человеко-машинных интерфейсов HMI (Human-Machine Interface) и SCADA-системы (Supervisory Control And Data Acquisition). Как правило, все современные ПЛК и системы распределенного ввода/вывода снабжаются OPC-сервером.

OPC-сервер – это вспомогательная программа, которая поддерживает связь с контроллером по имеющемуся у него каналу связи и предоставляет доступ к данным контроллера клиентам через стандартный OPC интерфейс. Сервер запускается автоматически при запуске клиента.

Использовать данные одного сервера могут одновременно несколько клиентов. Благодаря COM-технологии, при разработке OPC клиента можно использовать практически любой язык программирования (C++, Visual Basic, Delphi, Java). Однако, следует учитывать, что OPC требует значительных ресурсов рабочей станции (объем памяти и производительность процессора).

Примечание: CoDeSys OPC-сервер V2.0 может соединяться с любыми контроллерами, запрограммированными в CoDeSys. Он полностью отвечает требованиям стандарта OPC V2.0.

Архитектура CoDeSys OPC-сервера V2.0:



¹ OPC = OLE for Process Control (OLE для управления производственными процессами);
OLE = Object Linking and Embedding (связывание и встраивание объектов)
Подробнее см. www.opcfoundation.org и www.opc-europe.org

² COM = Component Object Model (модель составных объектов, базируется на OLE);
DCOM = Distributed Component Object Model (распределенная модель COM)

Для связи с контроллерами OPC-сервер 3S использует шлюз **CoDeSys Gateway**, также как и сама среда программирования CoDeSys. Для работы через OPC не требуется каких-либо дополнительных соединений.

Примечание: обмен данными через CoDeSys OPC-сервер осуществляется на базе символического файла.

Символьный файл (*.sym или *.sdb) содержит описания так называемых элементов данных (items). Каждый элемент соответствует ровно одной переменной в прикладной программе. Он необходим для сопоставления символического имени переменной и ее физического расположения в памяти контроллера. Данный файл автоматически генерируется системой программирования CoDeSys и передается Gateway и контроллеру одновременно с загрузкой кода прикладного проекта.

OPC-сервер запрашивает у Gateway содержимое символических файлов. На их основе он составляет внутренний список элементов. Актуальные значения данных для элементов этого списка доступны клиентам. Всякий раз OPC-сервер считывает символический файл соответствующего прикладного проекта, который был загружен через канал Gateway последним. Получить одновременно доступ к переменным сразу нескольких проектов (приложений) можно, если символические файлы были переданы одному и тому же Gateway.

Во внутреннем списке элементов OPC-сервер кэширует данные, получаемые от контроллеров через заданные промежутки времени. В результате, клиент (клиенты) читает и записывает значения переменных с некоторой задержкой (макс. ≈ 1 мс/элемент).

На практике OPC свободно оперирует со списками до 15 000 элементов и символическими файлами до 1,5 МВ. Загруженность системы при чтении и записи переменных зависит от того какое количество элементов активно в данной конфигурации.

OPC-сервер может группировать данные. При этом группы могут быть как общими (public), так и частными (private). Общие группы создаются OPC-сервером. Частные группы создаются самим клиентом. Если соответствующая функция включена в конфигурации, то OPC-сервер выполняет группировку: например, группирует переменные проекта по ROU, в которых они используются. Одна общая группа соответствует одному ROU.

Частные группы могут создаваться клиентом из любых элементов. Изначально они не влияют на группы OPC-сервера, но при необходимости могут быть перестроены в общие группы. Активировать или деактивировать конкретную группу переменных можно одной единственной командой.

Отдельные группы опрашиваются OPC-сервером поочередно. Данные элементов внутри одной группы читаются одновременно, насколько это позволяет объем коммуникационных буферов ПЛК.

Что нового по сравнению с OPC-Server V1.0?

- Поддерживается обновление списка переменных (символов) без остановки клиента.
- Поддерживается конфигурация мульти-ПЛК, т.е. клиент может соединяться сразу с несколькими контроллерами.
- Экспорт и импорт OPC-конфигурации (текстовый файл *.ini).
- Загрузка символов в режиме оффлайн доступна с новыми версиями OPC-сервера (V2.3.8.4 и выше) и Gateway-сервера (V2.3.3.3 и выше).

При попытке соединения с сервером поиск символического файла будет осуществлен сначала в загрузочной директории проекта, а затем, в случае неудачи, в загрузочной директории Gateway. По крайней мере, клиент будет отображать символы ("Offline Load"), даже с качеством "Bad". Для этого в конфигурации OPC (см. главу 3.2) должно быть задано имя проекта!

2 Установка OPC-сервера CoDeSys

2.1 Структура необходимых для работы файлов

1. Файлы Gateway сервера:

Gateway.exe, GClient.dll, GUtil.dll, GDrvBase.dll, GDrvStd.dll и GSymbol.dll

Данные файлы должны находиться в системной директории операционной системы. Например, *windows/system32* для Windows XP.

При первом запуске Gateway путь к его рабочей директории автоматически прописывается в реестре (по умолчанию: *C:\windows\Gateway Files*). Символьные файлы, предварительно сгенерированные в CoDeSys и помещенные в директорию проекта, копируются в рабочую данную директорию Gateway при установке соединения с контроллером. В зависимости от версии Gateway символьные файлы имеют расширение *.sym или *.sdb. Файл с расширением *.sdb это бинарная версия текстового файла *.sym. Он быстрее считывается OPC-сервером.

2. Файлы OPC-сервера:

§ CoDeSysOPC.EXE (OPC сервер)
§ OPCCCommonSetup.EXE (установщик)

Эти файлы можно поместить в любую директорию. Ниже перечисленные файлы создаются установщиком:

§ OPCPROXY.DLL
§ OPCCOMN_PS.DLL
§ CALLRPROXY.DLL
§ OCSDAAuto.DLL
§ OCSSpy_PS.DLL
§ OPCenum.EXE

3. OPCCConfig.exe: инструмент конфигурирования

Используется для установки и проверки параметров соединения или настройки сервера (прописаны в реестре).

2.2 Установка и регистрация

1. Одновременная установка и регистрация

Команда "*CoDeSysOPC /Install*" (при необходимости укажите путь к EXE файлу) запускает мастер установки (OPCCCommonSetup.exe), который выполняет установку всех необходимых OPC-файлов. Затем OPC-сервер автоматически регистрируется, и пользователь получает соответствующее сообщение о выполнении регистрации.

2. Регистрация после завершения установки

Команда "*CoDeSysOPC /RegServer*" (при необходимости укажите путь к EXE файлу) выполняет только регистрацию OPC-сервера. Установка необходимых файлов должна быть предварительно выполнена. Пользователь получает соответствующее сообщение о выполнении регистрации.

Возможность отдельной регистрации может понадобиться при обновлении программы или при самостоятельном изменении компонентов установки.

(Записи реестра описаны в разделе 3.2.5)

3. Деинсталляция OPC-сервера

OPC-сервер деинсталлируется с помощью команд "*CoDeSysOPC /UnRegServer*" или "*CoDeSysOPC /DeInstall*". Данные из реестра будут удалены, но установленные файлы не стираются.

3 Настройка OPC-сервера CoDeSys

Перед началом использования установленного OPC-сервера его необходимо настроить. Для того чтобы передать переменные из проекта CoDeSys в OPC-сервер и установить соединение с ПЛК, выполните шаги 3.1 - 3.3, как описано ниже:

3.1 Среда программирования CoDeSys: генерация описаний переменных проекта

Шаг 1: Выберите 'Проект' - 'Опции' 'Символьная конфигурация'

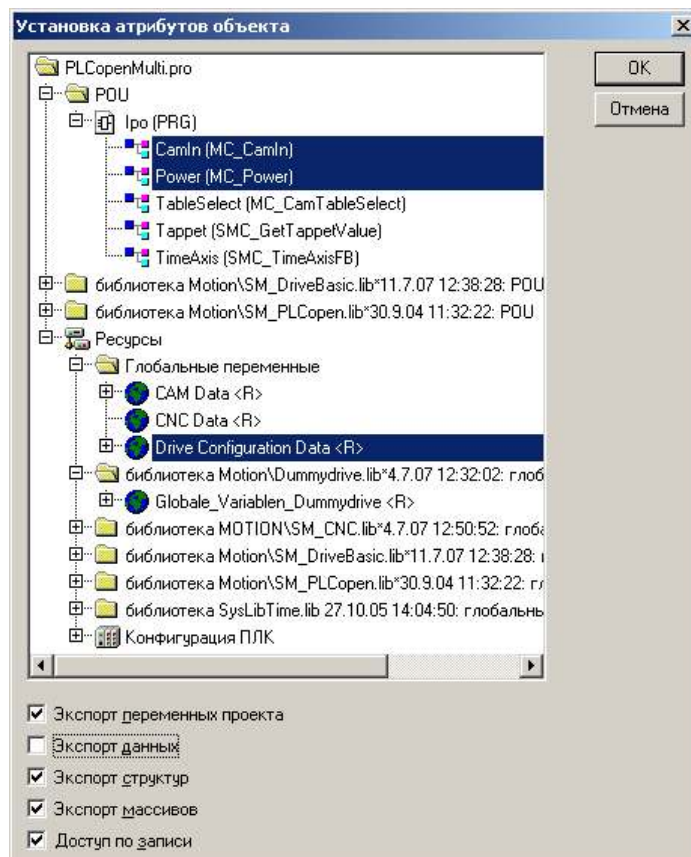
Информацию о переменных сервер получает из файла с именем проекта и расширением ".sym" (текстовый) или ".sdb" (бинарный). В нем содержатся символьные описания переменных (см. раздел 2.1). Для того чтобы символьный файл генерировался автоматически, должна быть активирована опция "Создавать описания" (Dump symbol entries).

Шаг 2: Конфигурирование символьного файла:

Атрибуты объектов проекта задаются в окне 'Установка атрибутов объекта'. Оно вызывается следующим образом: 'Опции' - 'Символьная конфигурация' - 'Настроить символьный файл...'

В этом окне доступны следующие опции:

- § 'Экспорт переменных проекта': переменные выбранного объекта передаются в символьный файл.
- § 'Экспорт данных': генерируются описания для структур и массивов.
- § 'Экспорт структур': для каждого элемента структуры генерируется свое описание.
- § 'Экспорт массивов': для каждого элемента массива генерируется свое описание.
- § 'Доступ по записи': Переменная будет доступна по записи.



Шаг 3: Установка коммуникационных параметров проекта:

Параметры используемого коммуникационного канала настраивается в окне 'Онлайн' 'Параметры связи...' (подробнее см. документ *CoDeSys_V23_RU.pdf*). Эти установки должны соответствовать установкам OPC-сервера. OPC-сервер автоматически запускается с последними используемыми параметрами (они сохраняются в реестре). Их можно проверить или изменить с помощью *OPCconfig.exe* (см. ниже).

Внимание: при конфигурации мульти-ПЛК, если пользователю необходимо получить доступ к переменным нескольких контроллеров через OPC-сервер, соответствующие проекты должны быть загружены в контроллеры через один и тот же Gateway.

Шаг 4 (необязательно): Сохранить проект Параметры соединения сохраняются вместе с проектом (после соединения параметры передаются Gateway серверу).

Шаг 5: Создание символьного файла и передача его Gateway серверу 'Онлайн' 'Подключение' - > 'Загрузка'

При компиляции проекта генерируется символьный файл и помещается в директорию проекта CoDeSys. В процессе загрузки символьный файл передается Gateway (помещается в его рабочую директорию 'Gateway Files') с текущими установками.

Внимание: Количество символов не должно превышать 15000, что соответствует файлу *.sym размером 1.5 MB.

Переменные в символьном файле описываются так же, как и в окне мониторинга.

Например:

```
PLC_PRG.A  
PLC_PRG.Structure.X[4]  
.GlobVar1
```

Доступ к переменным через прямые адреса не поддерживается.

3.2 Конфигурирование OPC-сервера с помощью утилиты OPCconfig.exe

OPC-сервер способен работать с данными нескольких проектов, но ему необходима информация о соответствии данных и проектов. Каждый раз, когда проект загружается в контроллер, специальный идентификатор проекта записывается в реестр. OPC-сервер считывает данный идентификатор и использует соответствующий символьный файл.

- ✘ Желательно загрузить проект в конкретный контроллер перед запуском OPC-сервера.
- ✘ Если состав переменных проекта изменен, то список переменных можно обновить без остановки OPC-сервера или клиентов путем перезагрузки проекта в контроллер (опция 'Создавать описания' должна быть включена). Если, например, клиент обратится к переменной, которая была удалена, то будет дано соответствующее сообщение.
- ✘ Обратите внимание, что существующий 'Загрузочный проект' уже не будет соответствовать измененному проекту.

Чтобы настроить OPC-сервер и отредактировать соответствующие записи в реестре, используйте конфигуратор *OPCConfig.exe*. Ниже описываются команды меню конфигулятора, а также диалоговые окна для настройки параметров отдельных ПЛК.

Внимание: Очень важно настроить конфигурацию канала связи, используемого OPC-сервером. Этот канал должен соответствовать каналу в проекте CoDeSys.

Обратите внимание на то, что загрузить один из вариантов подготовленных ранее настроек в конфигулятор можно с помощью командной строки (см ниже).

3.2.1 Меню 'File'

- | | |
|--------|--|
| 'Open' | Открывает конфигурацию, сохраненную с помощью 'File' 'Save'. |
| 'Save' | Сохраняет текущую рабочую конфигурацию сервера. |

'New'	Открывает новое окно конфигурации OPC-сервера, все настройки принимают значения по умолчанию.
'Export'	'Export' сохраняет текущую конфигурацию в текстовый файл. Открывается стандартное окно 'File Save as', где вы можете задать имя *.ini файла. Вы можете править данный файл в текстовом редакторе.
'Import'	

Для загрузки параметров из ini-файла в конфигуратор используйте команду 'Import'.

Пример ini-файла приведен в главе 4.

Импорт файла конфигурации через командную строку: предварительно созданный файл конфигурации ini (<Export-File.ini>), может быть импортирован в конфигуратор при его запуске:

```
OPCConfig.exe /auto <Export-File.ini>
```

Это команда автоматически открывает конфигуратор, заданный файл импортируется и сохраняется в качестве рабочей конфигурации, после чего конфигуратор закрывается.

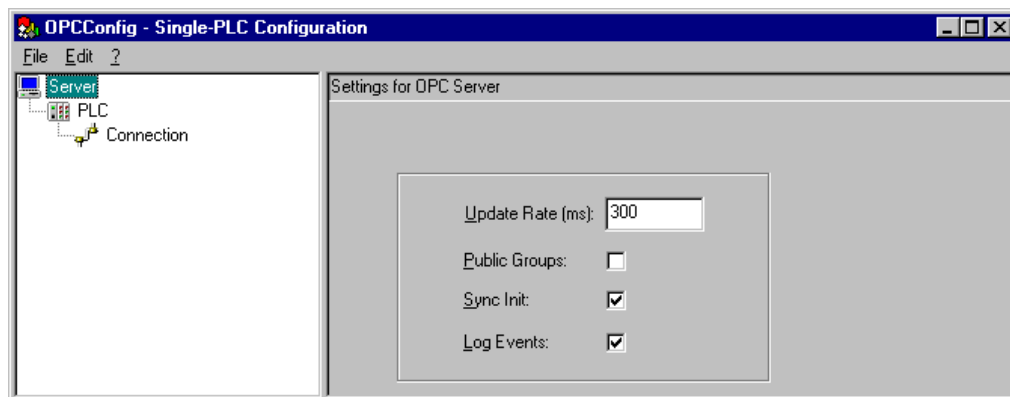
3.2.2 Меню 'Edit'

В зависимости от того, какой элемент дерева конфигурации выбран в данный момент ('Server', 'PLC' или 'Connection'), а также от того одно- или мульти- контроллерная конфигурация используется, будут доступны разные наборы команд меню.

'Append PLC'	Добавляет новый контроллер (в режиме Мульти-ПЛК).
'Rename PLC'	Переименовывает контроллер (в режиме Мульти-ПЛК).
'Delete PLC'	Удаляет контроллер (в режиме Мульти-ПЛК).
'Reset PLC'	Принимает для ПЛК параметры конфигурации, которые заданы в 'Edit' 'PLC Default Settings' (Мульти-ПЛК или одиночный ПЛК).
'PLC Default Settings'	Всегда доступна Открывает окно 'PLC Default Settings'. Здесь необходимо ввести параметры, которые в дальнейшем будут использоваться как параметры по умолчанию. Это окно аналогично окну 'Settings for PLCx' (см. 3.2.3). Для применения этих параметров к ПЛК используйте команду 'Edit' 'Reset PLC'.

3.2.3 Настройка OPC-сервера при работе с одним ПЛК (Single PLC):

После запуска конфигуратора открывается следующее окно:



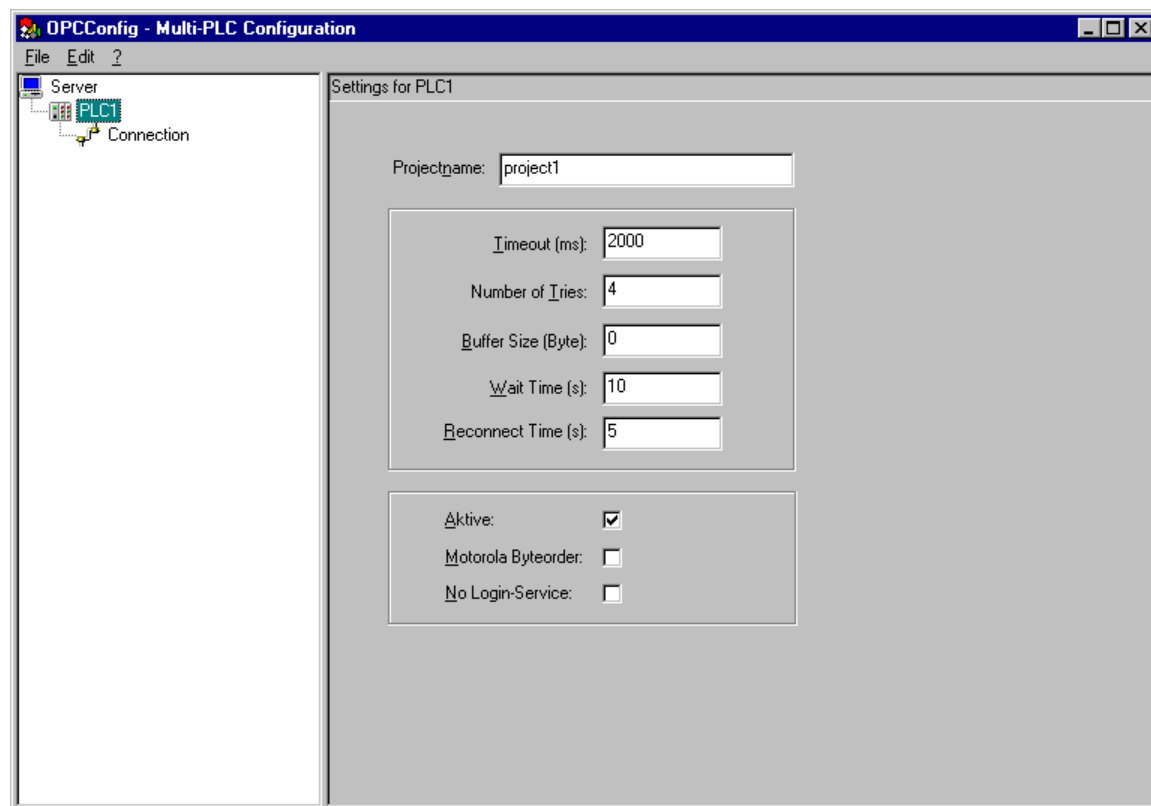
По умолчанию задана конфигурация для одного ПЛК. В ней важны следующие параметры:

В левой части окна отображается дерево конфигурации, содержащее объекты: **Server** (Сервер), **PLC** (Контроллер) и **Connection** (Соединение). Чтобы изменить параметры нужного объекта выберите его мышкой. Конфигурация выбранного объекта будет показана в правой части окна. Меню 'Edit' не используется при работе с одним ПЛК.

Объект '**Server**' имеет следующие параметры:

- **Update Rate (ms):** По умолчанию: 200
Период обновления для OPC-серверов в миллисекундах. Представляет собой период, с которым данные считываются из контроллера. Данные записываются в кэш, клиенты работают только с ним.
- **Public Groups:** По умолчанию: включено.
OPC-сервер использует общие группы для отдельных POU и для глобальных переменных.
- **Sync Init:** По умолчанию: включено.
Синхронная инициализация: OPC-сервер активируется только после полной загрузки всех символьных файлов.
- **Log Events:** По умолчанию: включено.
Ошибки и действия, выполняемые OPC-сервером, записываются в log-файл. Он помещается в директорию проекта под именем OPCServer.log. Вы можете просмотреть данный файл после закрытия OPC-сервера. Сообщения OPC-сессий добавляются в текущий log-файл, пока он не достигнет размера 1 Мб. Затем к имени этого файла добавляется текущая дата (например, *OPCServer12.01.2008.log*) и создается новый log-файл.

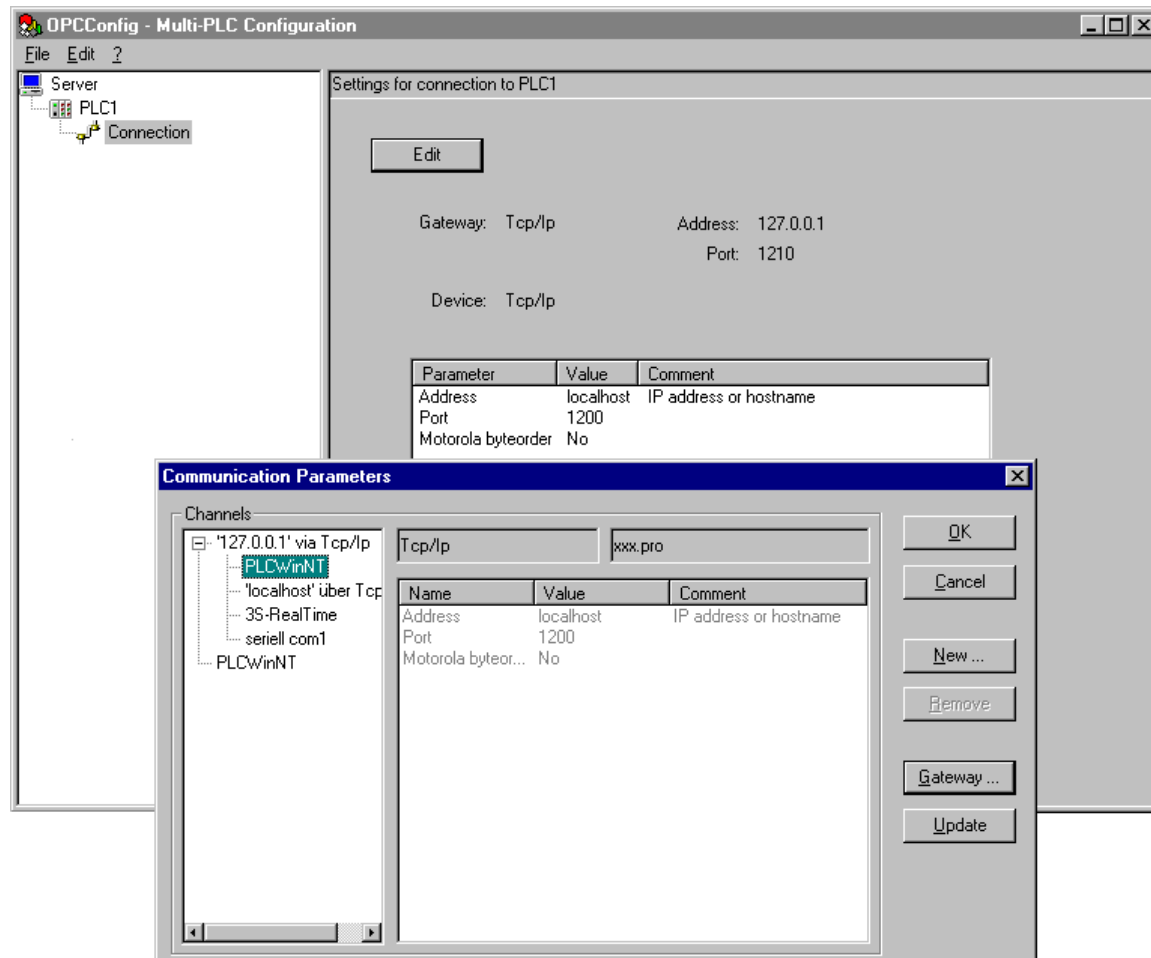
Объект '**PLC**' имеет следующие параметры:



- **Projectname:** Имя загруженного проекта. При работе с одним ПЛК это поле заполнять необязательно, за исключением случая, когда вы хотите сделать доступным работу клиентов без подключенного контроллера (Offline Load).

- **Timeout (ms):**
Если OPC-сервер не получит ответа от контроллера на свой запрос, он будет закрыт автоматически по истечении этого времени.
- **Number of Tries:**
Количество попыток драйвера Gateway передать блок данных в контроллер. Если все попытки завершены неудачно, будет выдана ошибка соединения. (Это функция доступна только для драйверов, поддерживающих передачу данных по блокам).
- **Buffer Size (Byte):**
Размер коммуникационного буфера в контроллере. Если введен "0", то сервер пытается запросить это число у коммуникационного драйвера. Если его получить не удалось, то буфер считается неограниченным.
- **Wait Time (s):**
Время в секундах, которое OPC-сервер дает контроллеру на выполнение инициализации (важно при автоматическом запуске контроллера).
- **Reconnect Time (s):**
Интервал, с которым OPC-сервер повторяет попытки восстановить связь с контроллером.
- **Active:**
Активность контроллера (доступно только в режиме Мульти-ПЛК).
- **Motorola Byteorder:**
Контроллер использует порядок следования байт Motorola (обычно с процессорами 68K).
- **No Login Service:**
Эта опция нужна для некоторых целевых систем, требующих обязательного логин-сервиса .

Объект '**Connection**' имеет следующие параметры:



Заданные параметры соединения указаны в полях 'Gateway', 'Device' и в таблице. Если OPC-сервер соединяется с удаленным Gateway, то дополнительно отображаются адрес и порт компьютера, на котором установлен шлюз. Они должны быть идентичны адресу и порту, заданным в диалоге 'Онлайн' 'Параметры связи...' CoDeSys.

Для настройки нового соединения или изменения уже созданного используйте диалог **Communication Parameters**, который открывается кнопкой **Edit**. Этот диалог аналогичен тому, что открывается в меню 'Онлайн' CoDeSys. Он подробно описан в руководстве по CoDeSys.

3.2.4 Настройка OPC-сервера для нескольких ПЛК (Multi PLC)

Для одновременной работы с данными сразу нескольких контроллеров, отключите опцию 'Single-PLC' в меню FILE.

Контроллеры добавляются в конфигурацию сервера командой 'Edit' 'Append PLC'. Настройка их параметров производится как описано выше для режима одиночного ПЛК.

Основные отличия от режима одиночного ПЛК:

- В меню **Edit** доступны дополнительные команды:

'Append PLC'	Добавляет новый объект "ПЛК". ПЛК автоматически именуются последовательно: "PLC<n>". Например, для первого ПЛК вместо n стоит 1, для второго – 2 и т.д.
'Delete PLC'	Удаляет ПЛК из конфигурации.
'Rename PLC'	Переименовывает ПЛК.

- Имя проекта, загруженного в данный контроллер, должно быть указано в поле **Projectname** в окне 'Settings for PLCx'.
- Отключив опцию **Active** в окне 'Settings for PLCx', можно исключить контроллер из данной конфигурации.

Примечание: OPC-серверы ниже версии V2.3.8.2 не могут соединяться через разные Gateway-серверы!

3.2.5 Записи в реестре (пример)

```
[HKEY_CURRENT_USER\SOFTWARE\3S-Smart Software Solutions GmbH\CoDeSys
OPC\Config]
* - установки, сделанные в OPC-конфигураторе в окне 'Settings for OPC Server'
(см. 3.2.3) *
"Check config"="yes"
"LogEvents"="yes"
"Multi"="no"
"Updaterate"="0x0000012c (300)"
"Wait for target"="0x0000000a (10)"
"Sync startup"="yes"
"Public groups"="yes"
"Reconnect Interval"="0x00000005 (5)"
"Notify on quality change"="yes"

[HKEY_LOCAL_MACHINE\SOFTWARE\3S-Smart Software Solutions GmbH\CoDeSys
OPC\PLCDefaults]
* - данные, заданные в 'Edit' 'PLC Default Settings' в OPC-конфигураторе (см. главу
3.2.2) *
"Motorola"="false"
"Timeout"="2000"
"Tries"="4"
"Waittime"="10"
"Reconnecttime"=5
"NoLogin"="true"
"Buffersize"="1"
"Activ"="true"


[HKEY_CURRENT_USER\Software\3S-Smart Software Solutions GmbH\CoDeSys
OPC\Connections]
* - установки, сделанные в окне 'Settings for PLC' в OPC-конфигураторе
(см. 3.2.3 и 3.2.4) *
"Project0"=hex (текущие параметры соединения с OPC-сервером с проектной
идентификацией)
"Motorola0"="No"
"Timeout0"="0x000007d0 (2000)"
"Tries0"="0x00000003 (3)"
"NoLogin0"="No"
```

```
"BufferSize"="0x00000000 (0)"
```

3.3 Запуск CoDeSysOPC.exe

OPC-сервер (и Gateway-сервер) автоматически запускается операционной системой, как только клиент устанавливает соединение. Его не обязательно запускать отдельно. Клиент будет автоматически соединен с работающим OPC-сервером, если тот был предварительно запущен вручную.

Как только клиенты разрывают свои соединения с сервером, он автоматически останавливается.

Когда OPC-сервер работает, на панели задач отображается иконка: 

Если вы щелкните по ней правой кнопкой мыши и выберите команду 'About', то получите информацию о версии OPC-сервера.

Запуск в тестовом режиме:

Соединение между OPC-сервером и клиентом можно протестировать, даже если нет подключенных контроллеров. Для этого OPC-сервер необходимо запустить командой:

```
CoDeSysOPC.exe /TestMode
```

После этого он будет работать в тестовом режиме. Сервер автоматически создаст набор тестовых элементов, доступных клиентам.

4 Пример ini-файла для конфигурации OPC-сервера

Ini-файл конфигурации OPC-сервера может быть экспортирован из конфигуратора. С ним можно работать как с обычным тестовым файлом. После чего его можно импортировать обратно в *OPCConfig*.

Ниже приведены примеры параметров для задания соответствующих значений в *OPCConfig*:

Параметр	Описание
[Server]	Установки OPC-сервера
updaterate=200	Update Rate
publicgroups=0	Общие группы
syncinit=1	Sync Init
PLCs=2	Соединение OPC-сервера с двумя контроллерами
PLC0=PLC_A	Имя первого контроллера: PLC_A
PLC1=PLC_B	Имя второго контроллера: PLC_B
[PLC:PLC_A]	Настройки для PLC_A
active=1	Активен в текущей конфигурации
motorola=0	Motorola Byteorder
nologin=1	Login Service
timeout=2000	Timeout
tries=3	Количество попыток
waittime=10	Время ожидания
reconnecttime=5	Время повторного соединения
bufferize=3	Размер буфера
project=opc1.pro	Имя проекта
gateway=Tcp/Ip	Параметры соединения с PLC_A (Соединение клиента с локальным Gateway по TCP/IP)
gatewayaddress=localhost	
gatewayport=1210	
device=Tcp/Ip	Параметры соединения с PLC1 (окно 'communication parameters')
parameters=3	
parameter0=Address	
value0=127.0.2.1	
parameter1=Port	
value1=1200	
parameter2=Motorola byteorder	
value2=No	

Команда	Смысл
[PLC:PLC_B]	Установки PLC_B
active=1	См. выше
motorola=0	
nologin=0	
timeout=2000	
tries=3	
waittime=10	
reconnecttime=5	
bufferize=0	
project=opc_test_s.pro	
gateway=Tcp/Ip	Соединение с gateway на локальном компьютере по TCP/IP (оно должен соответствовать тому, что используется для PLC-A!)
gatewayaddress=localhost	
gatewayport=1210	
device=Tcp/Ip	Параметры соединения с PLC2
parameters=3	
parameter0=Address	
value0=Sabine	
parameter1=Port	
value1=1200	
parameter2=Motorola byteorder	
value2=No	

5 Диагностика и устранение ошибок

5.1 Средства диагностики ошибок OPC

Для проверки правильности настройки сервера и канала связи можно использовать любой тестовый OPC-клиент. Причину возникновения проблемы позволяет выяснить анализ записей Log-файла OPC-сервера. При локализации сложных ошибок соединения в канале связи с контроллером, может оказаться полезной информация *Gateway Inspector*. Он фиксирует все переданные сервисы и ответы целевой системы.

5.2 Советы по устранению ошибок

Проверьте следующее:

1. Gateway установлен и запущен
На панели задач должна быть иконка Gateway. При наличии связи она показывается яркими цветами, всплывающая подсказка содержит надпись: *CoDeSys Gateway Server (active)*.
2. OPC-сервер установлен и зарегистрирован
Установка выполняется командами: 'CoDeSysOPC /Install' (Установка + Регистрация) или 'CoDeSysOPC /RegServer' (только регистрация)
3. CoDeSys правильно настроен
 - Для проекта генерируется символьный файл. В опциях проекта активирована опция 'Создавать описания' (Dump symbol entries).
 - Параметры соединения настроены правильно, выбран соответствующий канал Gateway.
 - Проект сохранен, скомпилирован, загружен, есть связь с контроллером.
4. OPC-сервер правильно настроен. Запустите утилиту OPCConfig.exe:
 - В настройках соединения выбран тот же Gateway, что и в проекте CoDeSys.
 - Попробуйте задать стандартные установки:
Motorola Byteorder: выключено (кроме контроллеров с процессорами 68K и PowerPC),
Public groups (общие группы): включено,
Sync Init (синхронная инициализация): включено,
Update Rate: 200 мс (время опроса),
Timeout: 2000 мс (таймаут связи),
Buffer size: 0 (если в документации на контроллер не указано иное),
Wait time: 60 с (время ожидания),
Reconnect Time: 5с (интервал восстановления связи),
Active: включено (контроллер активен в конфигурации Мульти-ПЛК),
Выполните команду 'Save' (новая конфигурация вступает в силу после того, как установлено соединение с OPC-сервером).
5. Проверьте тестовым клиентом, доступность переменных (Items):
Например, это можно сделать с помощью клиента *DiagnosticOPCClient.exe*. Этот тестовый клиент не входит в комплект установки, но его можно получить по запросу.
Последовательность команд: *Establish connection* (Connect OPC-Server) -> щелкните правой кнопкой мыши на *Private Groups* в правой части окна, закройте окно, нажав ОК, щелкните на появившейся строке правой кнопкой мыши -> *Add all items* -> должны появиться все указанные в символьном файле переменные (items).

5.3 Типичные ошибки

No connection to the PLC (Status NO_CONFIG):

Нет связи с ПЛК. Проверьте:

- ПЛК включен.
- Параметры соединения с ПЛК в OPC-конфигурации (OPCconfig.exe):
 - Выбран соответствующий протокол соединения с ПЛК.
 - ПЛК доступен через выбранный канал соединения (протестируйте с помощью „ping“, если соединение выполняется через TCP/IP).
 - Если соединение физически (присоединен ли кабель к ПЛК, к ПК).
 - Gateway запущен (если используется удаленный gateway).
- Есть связь с ПЛК в CoDeSys

No symbols available (Status NO_CONFIG):

Недоступна символьная конфигурация.

- Проверьте что в опциях проекта CoDeSys активирована опция '*Создавать описания*', нужные переменные включены в символьный файл. Просмотрите *.sym-файл текстовым редактором.
- Перезагрузите код проекта в контроллере. Доступен ли символьный файл (*.sdb) в директории Gateway (по умолчанию, директория C:\WINDOWS\Gateway Files)?
- Совпадают ли идентификаторы (ID) проекта в системе программирования и в символьном файле? Возможно, была выполнена загрузка новой версии проекта без обновления загрузочного проекта.

Items Quality „BAD“

Недостоверные значения переменных. Возможные причины:

- Ошибка соединения с ПЛК при работе OPC-сервера.
- Ошибка при загрузке или обновлении программы в ПЛК.
- Группа или элемент неактивен.

Обратите внимание:

- OPC-сервер останавливается автоматически при разрыве всех клиентских соединений с сервером.
- В менеджере задач операционной системы OPC-сервер показывается как процесс.

История изменений

Версия	Описание	Редактор	Дата
V1.8	Русская редакция, переработанная и дополненная.	IP, AG	20.01.08