



DSData сервер

Руководство пользователя

DS-DSDATA-M



Содержание

ГЛАВА 1: ВСТУПЛЕНИЕ

Об этом руководстве	1-2
Цель этого руководства	1-2
Дополнительные руководства	1-2
Техническая поддержка	1-2
Главы	1-3
Приложения	1-3
Введение в DDE	1-4
Короткий исторический урок	1-4
Что такое DDE?	1-4
Что такое DSData Сервер	1-4
Почему заменено имя (с DDE сервера на DSData сервер)?	1-4
Клиент / Сервер	1-5
Как это работает?	1-5
Как получить доступ к DSData Серверу из моего программного обеспечения?	1-5
Как я могу узнать, работает ли мое программное обеспечение с DDE?	1-5
Структура адресации из трех частей для DDE	1-6
Введение в OPC	1-7
Что такое OPC?	1-7
Различия между DDE и OPC	1-7
Как OPC работает в DSData Сервере?	1-7
OPC клиент / сервер	1-7
Данные OPC	1-8
OPC клиент	1-8
DSData Сервер как OPC сервер	1-8
Где я могу получить OPC клиент	1-8
Установка программного обеспечения	1-9
Лицензирование	1-9
Требование к аппаратуре	1-9
Запуск Setup.exe	1-9
Выбор каталога	1-9
Кнопки панели инструментов	1-10
Возможности меню	1-11

ГЛАВА 2: КОНФИГУРАЦИЯ СВЯЗЕЙ И ТЕМ

Установка связи с ПЛК	2-2
Создание стандартной последовательной связи	2-3
Создание модемной связи	2-6
Создание связи через Ethernet	2-11
Список модулей (Module List)	2-12
Режим адресации	2-12
Создание Темы	2-15
Договоренности по имени темы	2-16
Выбор коммуникационной связи	2-16
Присоединение документации	2-17
Интервал опроса	2-17
Интервал опроса как инструмент	2-18
Связь для каждой темы	2-18

Экран состояния DDE	2-19
Создание элементов в теме	2-23
Создание подтем	2-21
Создание элементов в подтеме	2-22
Разрешение вопросов	2-23
Установка сетевой карты	2-24
Сетевой протокол и присоединение	2-24
Сетевая конфигурация Windows 95	2-24
Установка сетевого протокола в Windows 95	2-24
Конфигурация сети в Windows NT	2-25
Установка сетевого протокола в Windows NT	2-25

ГЛАВА 3: СОЗДАНИЕ ТЕГОВ

Система счисления ПК	3-2
Ресурсы ПЛК	3-2
V-память	3-3
Двоичный код десятичного числа	3-3
Шестнадцатеричные числа	3-3
Теги регистров	3-4
Создание тегов регистров «только для чтения»	3-4
Создание тегов регистров для чтения и записи	3-4
Модификаторы данных	3-5
Без модификатора	3-5
:D Двойное слово	3-5
:V Преобразовать в BCD	3-6
:DB Двойное слово с преобразованием в BCD	3-6
:nn Бит в слове V-памяти (только для чтения)	3-6
:R Преобразует в реальное число	3-6
:W Чтение 16 бит (DL330, DL330P, DL340 и DL305 совместимые)	3-7
:WB Чтение 16 бит (DL330, DL330P, DL340 и DL305 совместимые)	3-7
Теги таймеров и счетчиков	3-8
DL105, DL205, DL350, DL405 и DL405 совместимые процессоры	3-8
DL330, DL330P, DL340 и DL305 совместимые процессоры	3-8
Теги дискретных входов/выходов	3-9
Нумерация дискретных входов/выходов	3-9
Создание тегов дискретных входов/выходов только для чтения	3-9
Создание тегов входов/выходов для чтения и записи	3-10
Использование псевдонимов и обозначений	3-11
Использование псевдонимов	3-11
Использование обозначений	3-11
Список возможных обозначений	3-11
Модификаторы операции записи	3-12
Объединение операций записи	3-12
Очередь операций записи	3-13
Установки по умолчанию	3-13
Использование модификатора для изменения значения по умолчанию	3-13
Использование модификаторов данных с модификаторами операций записи	3-13
Наблюдение за коммуникационной связью	3-14
Использование бита состояния для наблюдения за связью	3-14
Встроенные системные элементы	3-16

Управление коммуникационной связью	3-16
Использование состояния доступности для управления связью	3-16

ГЛАВА 4: ИСПОЛЬЗОВАНИЕ DSDATA СЕРВЕРА С ЭЛЕКТРОННЫМИ ТАБЛИЦАМИ

Вступление	4-2
Два способа доступа к данным ПЛК	4-2
Формулы - это просто	4-2
Макросы более гибки	4-2
Мы включили демонстрационную программу	4-2
Использование формул для чтения данных	4-3
Ввод формулы	4-3
Примеры формул	4-3
Создание макросов	4-4
Базовая структура макроса чтения	4-4
Базовая структура макроса чтения	4-4
Быстрый пример	4-5
Пример Excel №1 – Формулы и простые макросы	4-7
Демонстрация №1 EXCEL5.XLS	4-7
Что делает пример	4-8
Использование примера	4-8
Использование функции Excel VLOOKUP	4-9
Как работает VLOOKUP	4-9
Наблюдение за изменением состояния в реальном времени	4-9
Графическое представление данных	4-10
Использование мастера графиков	4-10
Пример Excel №2 – макрос сбора данных	4-11
Демонстрация №2 Collect.XLS	4-11
Использование Lotus 1-2-3	4-12
Формула	4-12
Макрос	4-12
DDE команды Lotus 1-2-3	4-12
Использование Quattro Pro	4-13
Формула	4-13
Макрос	4-13
DDE команды Quattro Pro	4-13
Разрешение вопросов	4-14

ГЛАВА 5: ИСПОЛЬЗОВАНИЕ DSDATA СЕРВЕРА С VISUAL BASIC

Вступление	5-2
Два способа доступа к данным	5-2
Окно свойств использовать просто	5-2
Подпрограммы дают гибкость	5-3
Мы включили демонстрационную программу	5-3
Использование свойств объекта для чтения данных	5-4
Простой пример чтения V-памяти ПЛК	5-4
Чтение значений дискретных входов/выходов	5-7

Использование подпрограмм для чтения и записи данных	5-8
Простой пример записи значения	5-8
Запись значения дискретных входов/выходов	5-12
Демонстрационная программа	5-14
Разрешение вопросов	5-14

ГЛАВА 6: ИСПОЛЬЗОВАНИЕ DSDATA СЕРВЕРА С INTOUCH ФИРМЫ WONDERWARE

Вступление	6-2
Что такое InTouch?	6-2
Как он общается с ПЛК	6-2
Начинаем работать с InTouch	6-3
Вам нужна Связь и Тема	6-3
Имя доступа к DDE	6-3
Чтение значений регистров	6-4
Выбор мастера	6-6
Чтение значений дискретных входов/выходов	6-10
Запись значений регистров	6-11
Запись значений дискретных входов/выходов	6-13
Заключение	6-15
Разрешение вопросов	6-16

ГЛАВА 7: OPC И DSDATA СЕРВЕР

OPC (OLE для управления процессом)	7-2
Использование OPC клиента для тестирования	7-2
Шаг 1: Соединение с DSData сервером	7-2
Шаг 2: Создание группы	7-3
Шаг 3: Создание элемента	7-3
Шаг 4: Проверка тега	7-4
Шаг 5: Просмотр и запись новых значений элемента	7-5
Шаг 6: Создание нового элемента, и т.д.	7-5
Использование программы HMI как OPC клиент	7-6

ПРИЛОЖЕНИЕ А: РЕГИСТРАЦИЯ ДАННЫХ, ПРОТОКОЛЫ И КАБЕЛИ

Регистрация данных	A-2
Регистрация отдельных элементов	A-2
Наблюдение за созданием новых OPC (или DDE) элементов	A-3
Протоколы и кабели	A-4
Контакты кабелей	A-5

Изменения в руководстве

Если вы обратитесь к нам, ссылаясь на это руководство, не забудьте добавить номер редакции.

Заголовок: DSData Server User Manual (DSData руководство пользователя)

Номер руководства: DS-DSDATA –M (ранее DA-DDE-M)

Публикация	Дата	Эффективные страницы	Описание изменений
Оригинал	9/96	Обложка/Права собственности Содержание Изменения в руководстве 1 — 57	Оригинальная публикация
2-е издание	2/98	Обложка/Права собственности Содержание Изменения в руководстве 1-1 — 1-9 2-1 — 2-22 3-1 — 3-15 4-1 — 4-14 5-1 — 5-15 6-1 — 6-16 A-1 — A-3 B-1 — B-3 C-1 — C-3	Оригинальная публикация для версии 1.6 Уменьшен формат
2-е издание, редакция А	3/00	Различные страницы	Добавлена глава 2 об OPC Изменено имя на DSData Server Убраны приложения B и C

Вступление

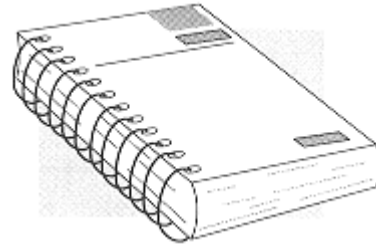
В этой главе...

- Об этом руководстве
 - Введение в DDE
 - Введение в OPC
 - Установка программного обеспечения
 - Кнопки панели инструментов
 - Возможности меню
 - Структура адресации из трех частей
-

Об этом руководстве

Цель этого руководства

Спасибо за покупку **DirectSoft32** Data Server (для краткости DSDData Сервер). DSDData сервер удивительный программный инструмент, который позволяет Вам использовать различные Windows приложения для быстрого и простого обмена данными с Вашим PLC.



В этом руководстве описано, как установить DSDData Сервер, как сконфигурировать необходимые связи с PLC, и дается базовая концепция, которая даст Вам возможность использовать DSDData сервер со всеми Windows приложениями, поддерживающими DDE или OPC.

Дополнительные руководства

Вы можете дополнительно прочитать инструкции, написанные для другого оборудования, которое вы собираетесь использовать. Мы рекомендуем:

- руководство пользователя для той модели PLC, которую вы используете (D4-USER-M, D3-350-M, D3-USER-M, D2-USER-M, D1-USER-M, D0-USER-M);
- DA-DSoft-M (руководство пользователя **DirectSoft32** не продается отдельно, но включается в программное обеспечение DirectSoft32).
- Руководство пользователя для программного обеспечения (Excel, InTouch и др);

Техническая поддержка

Хотя мы стремимся делать все хорошо, однако, информация может быть расположена таким образом, что вы не сможете найти сразу то, что ищете. Если так произошло, то используйте следующие источники для нахождения необходимой информации:

оглавление – список глав и разделов, в начале этого руководства;

краткое содержание – краткое описание глав, на следующей странице;

приложение – перекрестные ссылки на ключевые термины, в конце руководства, перед алфавитным указателем;

Также вы можете использовать оперативные источники получения свежей информации:

Интернет – адрес нашего WEB-сайта: www.plcsystems.ru

Электронная почта - support@plcsystems.ru

Главы



Основное содержание этого руководства содержится в следующих 6 главах

Введение Содержит краткое описание некоторых концепций DDE и OPC и процедуру установки DSData Сервера.



Конфигурирование связей и тем Содержит шаги для установки необходимых связей (Links) с ПЛК и создания Тем (Topics) для соединения DSData сервера с другими приложениями Windows.



Создание тегов Описывает основы DDE обмена, который используется программным обеспечением. Объясняется создание тегов (Tags) для доступа к данным в ПЛК, использование модификаторов данных, и использование псевдонимов и обозначений. Также содержится информация, как следить и управлять связью с ПЛК



Использование DSData Сервера с электронными таблицами описывает основные правила использования электронных таблиц (относящиеся к DDE) для распространенных программных продуктов, таких как Excel и Lotus® 1-2-3®. Мы включили два примера программы Microsoft Excel.



Использование DSData Сервера с Visual Basic показывает Вам, как использовать DSData Сервер, чтобы связать данные в ПЛК с анимационными объектами, созданными с помощью Visual Basic. Мы также покажем Вам несколько полезных демонстрационных программ, находящихся на вашем демонстрационном диске.



Использование DSData Сервера с Wonderware InTouch показывает Вам, как использовать 'мастеров - wizard' InTouch для создания тегов, используемых для оживления объектов из библиотеки объектов InTouch.



OPC и DSData сервер показывает использование приложения OPC клиента совместно с DSData сервером. Также описывает регистрацию данных.

Приложения



Регистрация данных, протоколы и кабели содержит расположение портов и протоколов, поддерживаемых этими портами для всех ПЛК и коммуникационных модулей. Также содержит схемы подключения кабелей для программирования.



Иконка блокнота слева на полях показывает, что параграф, находящийся справа от нее, содержит специальное примечание



Иконка восклицательного знака слева на полях показывает, что параграф, находящийся справа от нее, содержит предупреждение или предостережение. Этот значок очень важен, потому что эта информация может помочь Вам избежать порчи или повреждения оборудования.

Введение в DDE

Краткая историческая справка

Многие из Вас видели большое количество программных пакетов для программирования ПЛК, предлагаемых различными поставщиками ПЛК. Эти пакеты позволяют Вам наблюдать за данными в ПЛК в режиме отладки, но они не предназначены для запросов данных или управления на базе компьютера. Они не работают, когда вы пытаетесь отобразить данные в графическом формате, особенно в сравнении с такими приложениями, как Excel, InTouch и др.

Раньше, если Вы нуждались в лучшем отображении данных, или, если вы хотели использовать свой компьютер для управления, вам или надо было знать специфические детали коммуникационного протокола, для написания собственных драйверов, или использовать программное обеспечение сторонних производителей, содержащее драйверы Вашего ПЛК. Если Вы решили изучить коммуникационный протокол, то Вам потребуется несколько часов (или даже дней) для изучения того, как отформатировать различные запросы на чтение и запись данных. Вам придется сделать это, так как ПЛК разговаривает на специфическом языке. Если вы выбрали путь сторонних разработчиков, поставщики программного обеспечения уже имеют написанные драйвера, которые позволяют программному обеспечению понять язык общения Вашего ПЛК. Проблема здесь заключается в том, что поставщики не успевают обновлять версии своих драйверов. Идеальным решением был бы такой тип пакета, который бы отслеживал все изменения для вас. Итак, лучшее решение вы найдете во вступлении к нашему DSData Серверу.

Что такое DDE?

DDE или динамический обмен данными (Dynamic Data Exchange) – стандартный протокол обмена сообщениями, который позволяет двум или более независимым приложениям разделять и обновлять информацию в Windows окружении. DDE действует только как переводчик, что позволяет различным приложениям разговаривать на общем языке и понимать друг друга. При помощи DDE мы можем управлять передачей данных между несколькими приложениями Windows – например InTouch может разговаривать с Visual Basic, или Word может общаться с Excel, или каждая из этих программ может общаться с любой другой из них, или любая из этих программ может общаться с ПЛК. Фактически, любая программа, использующая DDE, может общаться с другой программой, которая также использует DDE.

Что такое DSData Сервер

DSData Сервер – это специальное программное обеспечение, разработанное с использованием Wonderware Server Development Tool Kit в комбинации с коммуникационным сервером, который находится в программном пакете **DirectSoft32**. Вы можете представлять себе DDE как трубу, по которой идут данные между вашим приложением и ПЛК, где переводятся запросы, идущие от приложения, в запросы, которые будут понятны ПЛК.

Почему заменено имя (с DDE сервера на DSData сервер)?

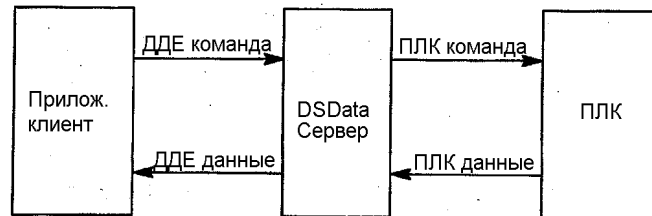
Новый DSData сервер позволяет больше, чем только DDE. Он также позволяет вам использовать всю мощь OPC (Object linking and embedding for Process Control – встраивание и внедрение объектов для управления процессами). Для более подробной информации смотрите вступление в OPC на странице 1-7 и главу 7.

Клиент / Сервер

Существует два ключевых понятия, которые вы будете часто встречать в любом тексте при обсуждении DDE:

Сервер – программный пакет, который управляет обменом и переводом данных, который в нашем случае – DSDData Сервер.

Клиент – программный пакет, который запрашивает или посылает данные. Клиентом может быть любое Windows приложение, которое поддерживает DDE.

**Как это работает?**

Вы используете инструменты DSDData Сервера для создания коммуникационной связи с ПЛК. Мастер коммуникации поможет Вам определить некоторые базовые параметры, такие как тип ПЛК, скорость передачи и др. После этого Вы создаете «Тему(Topic)», которая ссылается на связь и делает связь доступной для приложений, которые вы собираетесь использовать. Однажды сконфигурированный, DSDData Сервер выполняется как любое другое приложение Windows и управляет запросами на обмен данными с ПЛК.

Как получить доступ к DSDData Серверу из моего программного обеспечения?

Сначала Вы должны вставить простые DDE команды в ваше программное обеспечение, которые укажут DSDData Серверу, где взять данные для запроса на чтение, или куда записать данные в ПЛК при запросе на запись. Базовая структура команд очень проста. Вот пример, который читает данные из V-памяти в электронную таблицу Excel.

=Dsddata|DemoTopic!'V2000:B'

Сервис (всегда Dsddata) Тема (определяет связь с ПЛК) Элемент данных Модификатор (преобразование данных)

Команды всегда содержат подобную базовую структуру, но синтаксис отличается в различных программных продуктах. Синтаксис может использовать скобки, кавычки и пр., чтобы точно определить запрос. Например, подобная команда будет работать, если ее поместить на лист Excel, но вам придется незначительно изменить команду (кавычки, например), для использования команды в какой-либо другой электронной таблице.

Как я могу узнать, работает ли мое программное обеспечение с DDE?

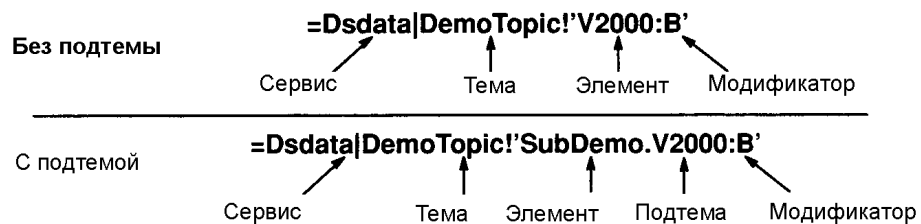
Не все пакеты, поддерживающие DDE, будут работать с DSDData Сервером. Вы всегда можете попробовать их, но самый простой способ посмотреть в документации, может ли программное обеспечение *принимать числовые данные в реальном времени через DDE* механизм. Вы можете позвонить в службу поддержки производителя вашего программного обеспечения, если вы не смогли найти эту информацию в документации.

Структура адресации из трех частей для DDE

Хотя каждое программное обеспечение поддерживает свой собственный синтаксис для ввода DDE команд, они все используют структуру адресации из трех частей. Она состоит из Сервиса, Темы и Элемента (Service, Topic, Item), к которым вы хотите получить доступ.

Перед тем, как вы будете использовать DSData Сервер, вам необходимо создать все необходимые DDE Темы, обычно по одной теме для каждого ПЛК. Две другие части структуры команды (Тема и Элемент) определяются тогда, когда вы вводите команду в прикладном программном обеспечении.

Прежде чем вы начнете использовать DSData Сервер, будет полезно посмотреть, как отдельная ячейка данных в ПЛК определяется в приложениях. Рисунок внизу показывает простую команду, которая может быть использована в листах Excel для чтения данных из V-памяти ПЛК



- **Сервис** – в некоторых книгах «Сервис» объясняется, как «название приложения». Это просто название DSData Сервера. Название нашего DSData Сервера – Dsddata. Оно не будет никогда изменено. Текст не чувствителен к регистру. Символ после "Dsddata" – символ трубы (вертикальная черта).
- **Тема/Подтема** – определяет логическое соединение между Сервисом и приложением, которое предоставляет доступ к данным, описанным элементом. Это название должно иметь смысл, например, Tank2. Подтема подобна теме внутри темы.
- **Элемент** – это может быть любой тип данных и адрес, распознаваемый ПЛК. Это может быть V-память, входные и выходные точки, таймеры, счетчики, и т.д. Если Вы используете **DirectSoft32** для присвоения имени ячейки памяти или элементу программы, вы можете написать имя ячейки вместо адреса .
- **Модификатор** – определяет, какой тип преобразования данных должен использовать DSData Сервер перед тем, как переслать их пользователю. Существует несколько разных модификаторов, которые описаны в главе 3. Модификатор «:B», используемый в этом примере, переводит значение в формат BCD, прежде чем переслать его в запрашивающее приложение.

Надо сделать всего два шага, что бы сконфигурировать DSData Сервер. Мы опишем их более подробно в следующей главе.

- **Установить связь с ПЛК** – вы должны иметь физическое соединение между компьютером, на котором запущен DSData Сервер и ПЛК. Связь описывает физические характеристики соединения между PC и ПЛК. Например, она содержит информацию, какой порт вы собираетесь использовать, какой процессор в ПЛК, какую скорость передачи он поддерживает, адрес узла и т.д.
- **Создать Тему** – создание темы подразумевает присвоение имени выбранной Связи с ПЛК для дальнейшего использования, присоединения документации из проекта **DirectSoft32** и, при необходимости, определения интервала обновления.

Введение в OPC

Что такое OPC

OPC (OLE for Process Control – OLE для управления процессом) основан на Microsoft OLE (сейчас Active X), COM (component object model – модель компонентных объектов) и DCOM (distributed component object model – распределенная модель компонентных объектов) технологиях. OPC состоит из стандартного набора интерфейсов, свойств и методов для использования в управлении процессом и приложениях автоматизации. Технологии Active X/COM определяют, как отдельный программный компонент сможет взаимодействовать и совместно использовать данные. OPC предоставляет интерфейс для общения с различными устройствами управления процессом независимо от управляющей программы или устройства в процессе.

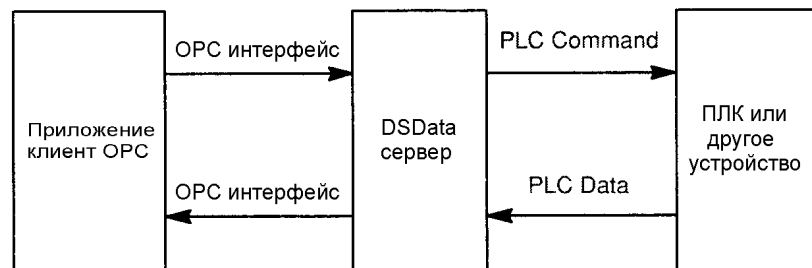
Более подробная информация на сайте организации OPC:

www.opcfoundation.org

Различия между DDE и OPC

Вы можете поинтересоваться, в чем же отличие между DDE и OPC? OPC предоставляет более интуитивный интерфейс Клиент/Сервер: не надо печатать команды или макросы.

С помощью DDE вы можете использовать данные непосредственно в электронных таблицах.



Как OPC работает в DSData Сервере?

DSData Сервер служит как связь или мост между вашим OPC клиентом и вашим ПЛК. Вы используете инструменты DSData Сервера для построения коммуникационной связи с ПЛК. Однажды сконфигурированный, DSData сервер работает как любое другое приложение Windows, сервер управляет запросами на обмен данными с ПЛК.

OPC клиент / сервер

Подобно технологии DDE, OPC использует модель клиент/сервер.

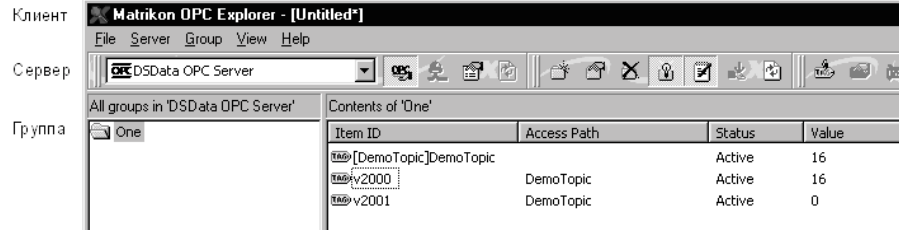
- **OPC сервер** – программный пакет, который управляет обменом данных и переводом, которым в нашем случае является DSData сервер.
- **OPC клиент** – программа, которая запрашивает чтение и запись данных. Это может быть любой из нескольких доступных OPC клиентов.

Данные OPC

OPC организует и обменивается данными в группах и элементах.

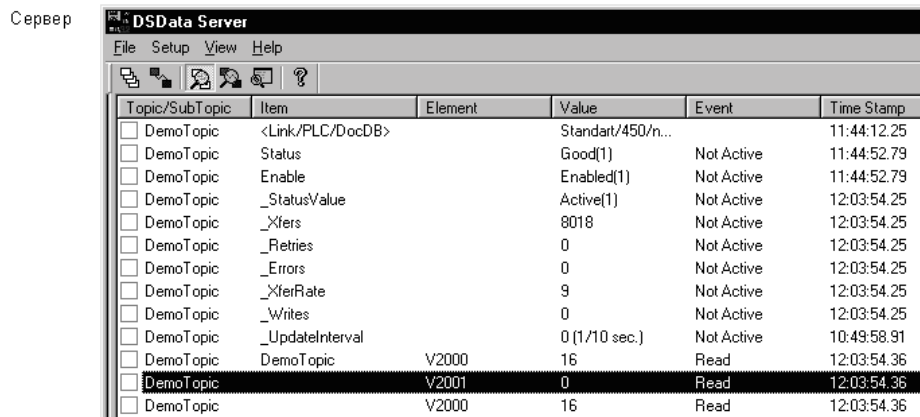
- **Группа (Group)** – объект данных, который содержит один или несколько элементов данных
- **Элемент (Item или Item ID)** – объект данных, связанный с отдельной точкой данных (например, регистр ПЛК). Элемент должен быть определен внутри группы. Примечание: элемент (item) и идентификатор элемента (Item ID) являются синонимами.
- **Путь доступа (Access Path)** – связь с данными. В DSData Сервере это Тема (Topic).

OPC клиент



Тема (путь доступа) Идентификатор пункта данных

DSData Сервер как OPC сервер



Тема (Путь доступа) Идентификатор пункта данных

Где я могу получить OPC клиент

Несколько OPC клиентов свободно распространяются в Интернете. Один из них мы будем использовать в будущем в этом руководстве – Matrikon OPC Explorer. Его можно получить на www.matrikon.com.

Установка программного обеспечения

Лицензирование

Наша программа, в отличие от многих других программных продуктов, не требует аппаратного ключа. Пожалуйста, не забывайте, что недопустимо делать копии программного обеспечения для нескольких инсталляций. Также недопустимо делать копии программного обеспечения и встраивать его в выполняемые приложения, которые вы поставляете заказчику. Вам необходимо заказывать копии DSData Сервера для каждой рабочей станции, на которую вы устанавливаете программное обеспечение. Для производителей компьютеров и системных интеграторов мы предлагаем специальный пакет лицензий, который дает экономный путь для поставки программы в каждую систему или продукт, которым вы торгуете. Этот пакет (код заказа PC-DDEOEM) содержит определенное количество копий программного обеспечения с руководствами по уменьшенной цене.

Требование к аппаратуре

Прежде чем устанавливать программное обеспечение, предварительно убедитесь, что ваша система подходит для использования DSData Сервера. Ваш персональный компьютер должен иметь следующие характеристики:

Требования ->	Минимальные	Рекомендованные
Операционная система	Windows 95/98/2000 или Windows NT	Windows NT
Оперативная память	32 Мб	128 Мб
Процессор	133 MHz Pentium	350 MHz Pentium II
Жесткий диск	5 Мб свободно	10 Мб Свободно
Монитор	Цветной VGA	Цветной SVGA

Запуск Setup.exe

Установка программного обеспечения - очень простая операция. Она состоит из установки инсталляционного компакт-диска и запуска Setup.exe. **Выберите Run из меню Пуск и в открывшемся диалоговом окне введите D:\Setup.exe. Вашему устройству для чтения компакт-дисков в системе может быть назначена другая буква. Убедитесь, что вы используете правильную букву дисковода компакт-дисков.**

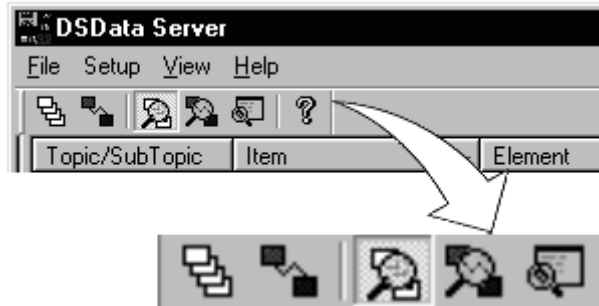
Выбор каталога

Единственная опция, которую можно изменить в процессе установки – это каталог установки. Каталог по умолчанию – 'c:\DirctSoft32'. Если вы используете программу **DirectSoft32**, вы должны установить DSData Сервер в тот же каталог. Это дает возможность впоследствии легко модернизировать оба пакета одновременно. Если предлагаемое расположение инсталляции вас устраивает, выберите ОК, и программное обеспечение будет распаковано на ваш жесткий диск. Если вы хотите указать другое расположение, просто наберите нужный путь и нажмите ОК. Если каталог, который вы выберете, не будет существовать, то он создастся автоматически.

Кнопки панели инструментов

Панель инструментов

Вверху окна программы вы можете увидеть четыре кнопки. Будем называть эти кнопки панелью инструментов. Этот раздел содержит краткое описание этих кнопок. Вы можете встретить некоторые термины, которые пока не в состоянии понять. Не огорчайтесь, мы опишем их более подробно позже, когда будем объяснять, как установить DDE связь.



Setup|Topics – Эта кнопка позволяет вам создать Тему. Тема – это название определенного источника данных. Например, вы сможете захотеть получать данные с “Насосная станция 2” или “Производственный конвейер 3”, и т. д. В демонстрационных программах, включенных в инсталляционный диск, мы назвали все темы «DemoTopic».



Setup|Links – Эта кнопка открывает настроечное окно, которое позволяет создавать или добавлять коммуникационные связи. Если вы использовали **DirectSoft32** ранее, вы можете вызвать установки для общения с различными ПЛК.



View|Topic Status – Эта кнопка включает и выключает индикацию состояния темы каждый раз, когда вы щелкаете по иконке. Панель состояния показывает, какая тема используется DSData Сервером в данный момент времени. Существует индикатор, который для активной связи показывает режим и состояние чтение/записи.



View|Link Info – эта кнопка открывает окно, которое позволяет вам наблюдать за состоянием коммуникационной связи в *реальном времени*. Выделите связь, которую вы хотите проверить, и нажмите на кнопку Select. Вы сможете увидеть описание всех ошибок, и время их появления.



View|Logger – эта кнопка запускает программу отладки DBWin32, которая работает как регистратор данных.



Help|About – эта кнопка отображает на дисплее диалог DSData Сервера “About” (“О программе”)

Возможности меню

Панель управления – это самый легкий путь выполнить любое действие, которое DSDData Сервер способен выполнять. Однако, также имеется панель меню с выпадающими пунктами, которые предоставляют вам доступ к тем же самым возможностям. Она может быть очень полезными, если у вас нет доступа к мышке.



File Setup View Help

Вы можете использовать или мышь, или клавиатуру для доступа к командам меню. Можно щелкнуть на пункте меню мышкой, или нажать и удерживать клавишу alt и подчеркнутую букву в пункте меню для его раскрытия. Подобным образом вы можете выбрать отдельный пункт меню. Ниже приведено краткое описание меню.

File – единственная возможность – выход (Exit) из DSDData Сервера

Setup – вы можете сконфигурировать Тему (Topics) или Связь (Links) из этого меню

View – используйте этот пункт меню для изменения внешнего вида DSDData Сервера. Метка слева от пункта меню означает, что пункт меню выбран. Например, вы можете скрыть панель управления и панель информации, которая появляется внизу окна DSDData Сервера.

Help- Если вы хотите вспомнить что-то, что вы забыли из этого руководства, используйте встроенную помощь – это кратчайший путь для получения основной информации, которая вам может потребоваться. Просто нажмите на пункт Index для просмотра информации.

Конфигурация Связей и Тем

В этой главе...

- Установка связи с ПЛК
 - Создание Темы
 - Создание элементов в теме
 - Создание подтемы
 - Создание элементов в подтеме
 - Ответы на вопросы
-

Установка связи с ПЛК

В этом разделе мы объясним, как с помощью Мастера Коммуникации установить связь между компьютером, на котором работает DSData Сервер и ПЛК. Если у вас установлен **DirectSoft32**, то вы можете использовать его для установления связи с ПЛК, эта связь будет доступна и для DSData Сервера. Для продолжения разговора, мы предположим, что пока связи не определены. Существует три различных типа связей, которые вы можете создать:

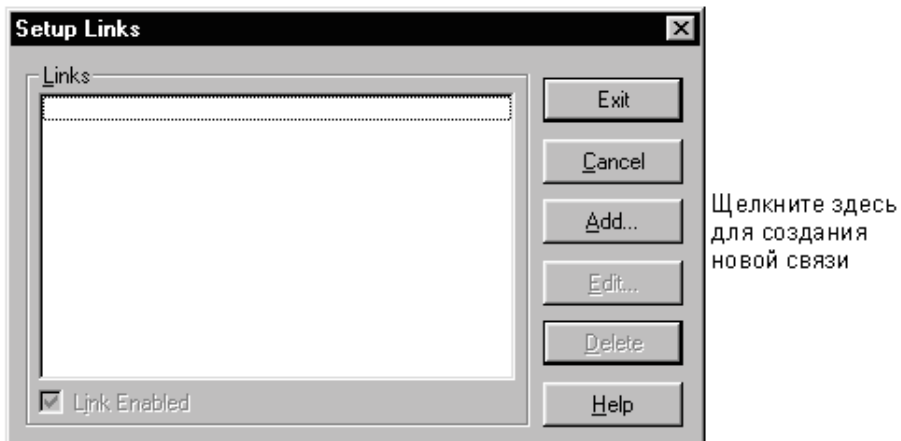
- последовательное соединение, использующее порты с COM1 по COM8 для прямой связи с ПЛК
- последовательное соединение, которое соединяется через пару модемов с портом ПЛК
- соединение через сетевую карту Ethernet с модулем H2-ECOM или H4-ECOM ПЛК

Независимо от типа соединения с ПЛК, прежде чем пытаться установить связь, необходимо подсоединить соответствующий кабель между ПЛК и компьютером, и ПЛК должен быть включен.

Чтобы начать, выберите **Setup | Links** из панели меню (или щелкните по кнопке конфигурации связи на панели инструментов).



Когда вы выберете один из элементов, показанных выше, появится окно конфигурации связи. Щелкните по кнопке ADD для создания новой коммуникационной связи.

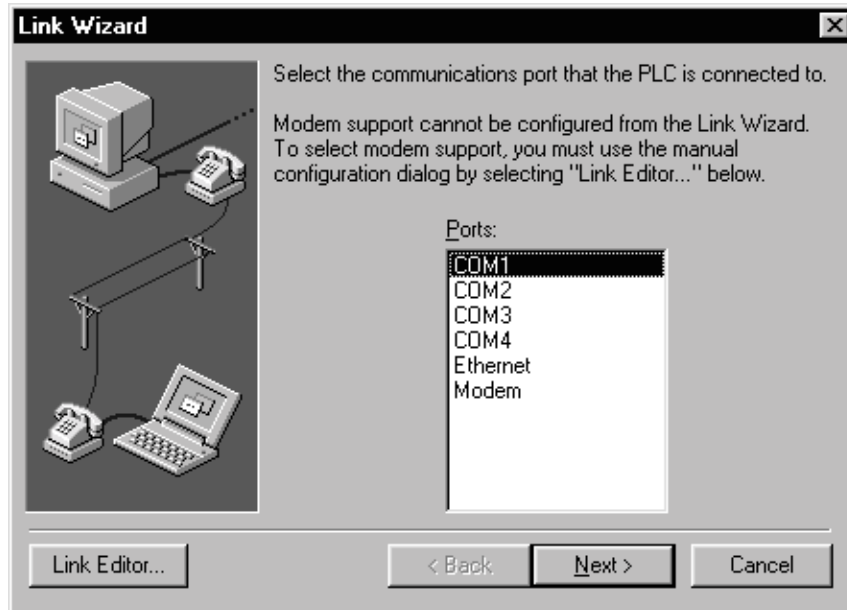


Примечание: Если вы имеете сеть ПЛК, то вы должны создать коммуникационную связь для каждого ПЛК. Например, возможно, что все ПЛК в сети имеют одинаковые коммуникационные установки (протокол, скорость и т. д.). Однако каждый из ПЛК должен иметь уникальный адрес станции, и вы должны создать отдельную связь для каждого из ПЛК.

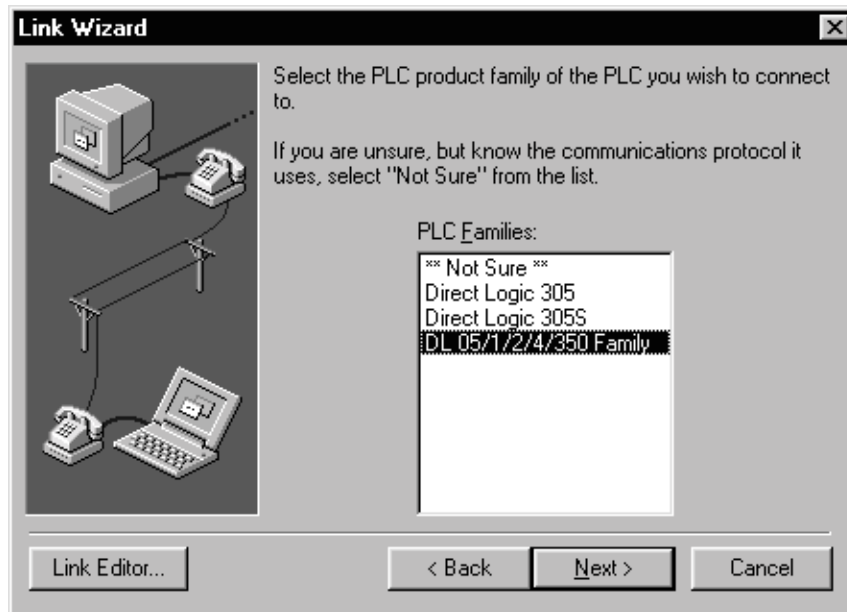
Создание стандартной последовательной связи

В первой части будет обсуждена конфигурация связи, которая использует стандартный последовательный порт. Если вы хотите создать последовательную связь через модем, перейдите к странице 2-6, где мы описываем связь через модемы. Если вы хотите создать связь по Ethernet, перейдите вперед к странице 2-11, где приведена информация об Ethernet соединении.

1. В основном, все коммуникационные настройки будут определяться для вас автоматически **Мастером Связи (Link Wizard)**. Сначала выберите порт, к которому подключен ПЛК. После того, как сделаете выбор, щелкните на клавише **Next>**(Следующий).

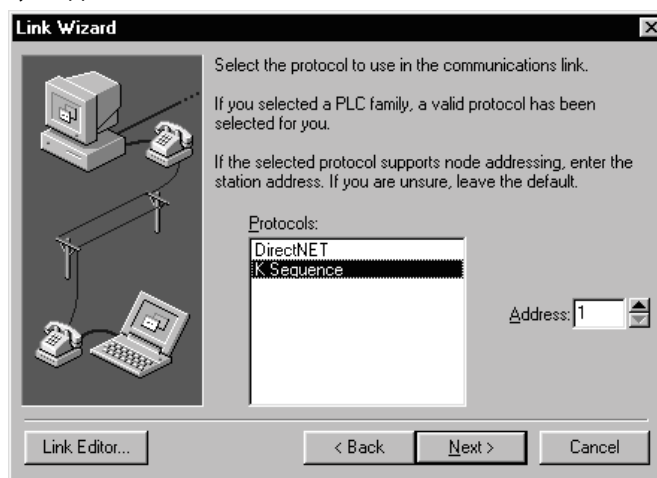


2. Выберите семейство ПЛК и щелкните на нем мышью. Если вы не уверены, какое семейство ПЛК вы используете, но знаете, какой коммуникационный протокол используется, вы можете выбрать 'Not Shure' (не уверен). Если вы используете ПЛК, совместимые с **AutomationDirect.com**, Мастер Соединения попытается определить модель автоматически. Щелкните на **Next>**, когда сделаете выбор.

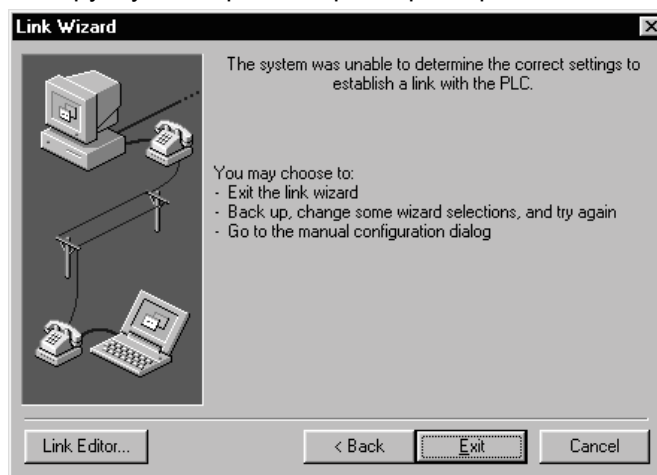


Конфигурация связей и Тем

3. Выберите или **DirectNet** или **K-sequence** протокол. Если на прошлом шаге вы выбрали семейство ПЛК из списка, подсвеченный прямоугольник будет появляться на возможных протоколах для этого семейства. Выберите протокол, основываясь на двух фактах:
 - Порт ПЛК, с которым вы соединяетесь, должен поддерживать протокол, который вы выбираете. Смотрите список доступных протоколов для каждого порта для всех **AutomationDirect.com** совместимых процессоров в приложении А.
 - Если вам надо выполнять операции записи отдельных дискретных входов/выходов или управляющих реле, тогда вы должны выбрать K-sequence протокол. DirectNet протокол не позволяет записывать отдельные биты.
4. Если адрес ПЛК отличен от 1, введите этот адрес сейчас. Щелкните на **Next>**, когда вы закончите.



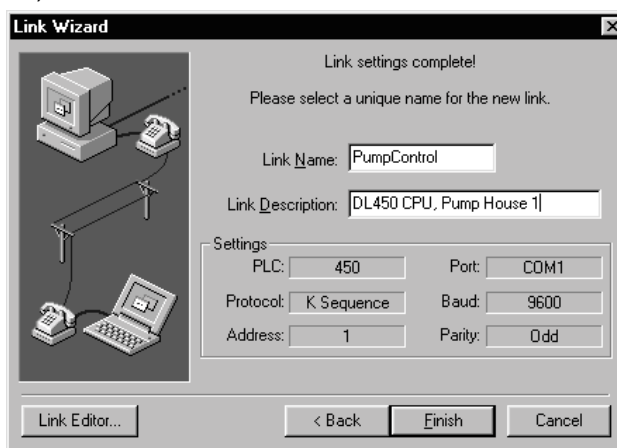
5. Теперь DSDData Сервер попытается установить связь с ПЛК, используя адрес узла и протокол, который вы выбрали. Он попытается использовать комбинацию скорости 9600 и проверки на нечетность данных. Если эта комбинация окажется неудачной, будет запущен механизм автоматического определения параметров скорости и проверки на четность. Если определение параметров окажется неудачным, появится диалог, приведенный ниже. Вы можете щелкнуть на кнопке **Link Editor** (редактор связи), чтобы попытаться вручную настроить параметры порта.



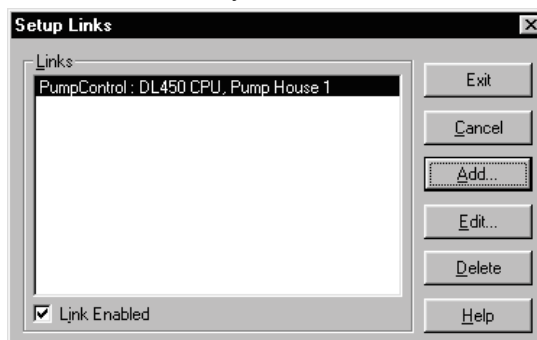
- Если DSDData Серверу не удалось установить связь с ПЛК, вы сможете увидеть следующее окно, поясняющее причину проблемы.



- Если DSDData Сервер успешно связался с ПЛК, введите уникальное имя Связи и описание (Link Description) к ней. Каждая связь должна иметь свое собственное, уникальное имя. Имя может иметь до 16 букв и содержать символы пробелов. Поле описания может содержать до 32 букв. Введите имя связи и описание, а затем щелкните на кнопке **Finish** для возврата к окну Установка Связей (Setup Links).



- После успешного создания связи диалог Установка Связей будет содержать Связь, которую вы только что создали. Кнопка Link Enabled (связь доступна) показывает, что параметры конфигурации заданы правильно, и ПЛК доступен для использования.

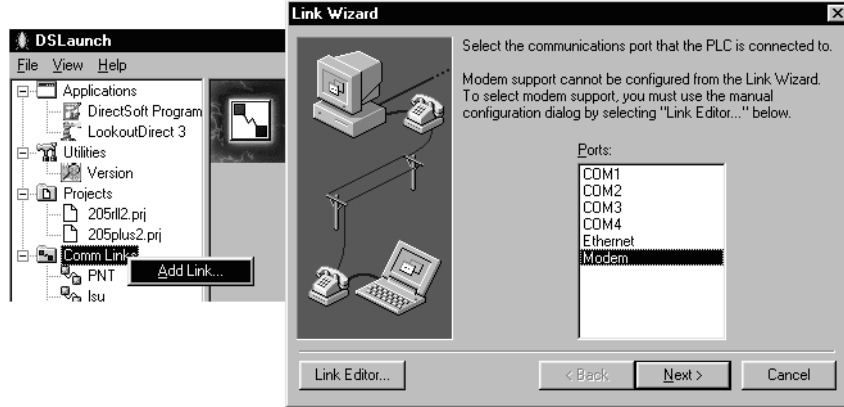


- Щелкните на **Exit** для возврата к главному окну DSDData Сервера.

Создание модемной связи

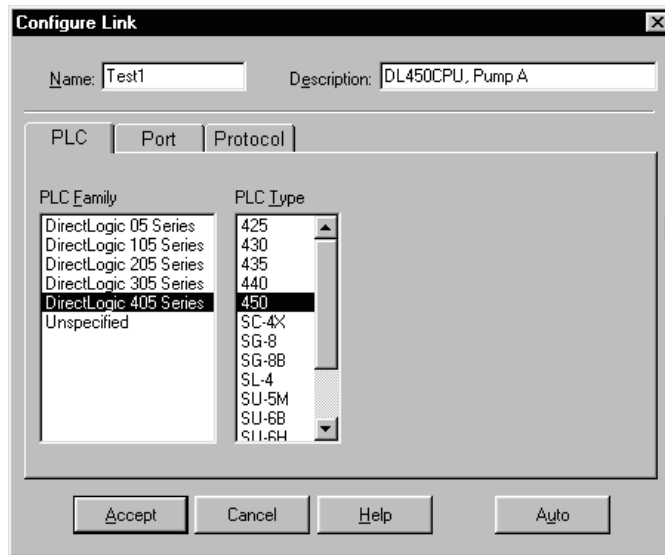
Этот раздел объясняет, как сконфигурировать Связь, которая использует стандартный последовательный порт для соединения через модем.

1. Из программы DSLaunch, щелкните правой кнопкой мышки по CommLinks. Выберите Add Link (добавить связь).

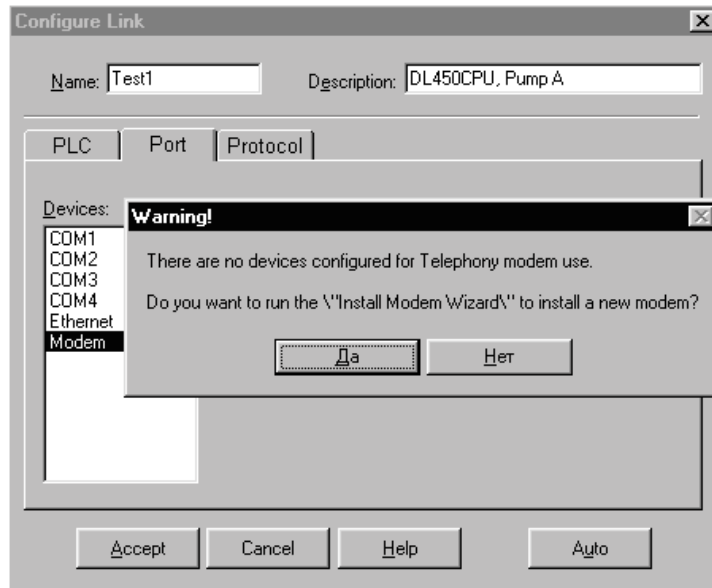


Конфигурация связей и Тем

2. Вам необходимо выбрать семейство ПЛК и тип процессора ПЛК. Щелкните в списке **PLCFamily** по одному из семейств, и список всех возможных процессоров для этого семейства появится справа. Выберите щелчком соответствующий тип процессора.

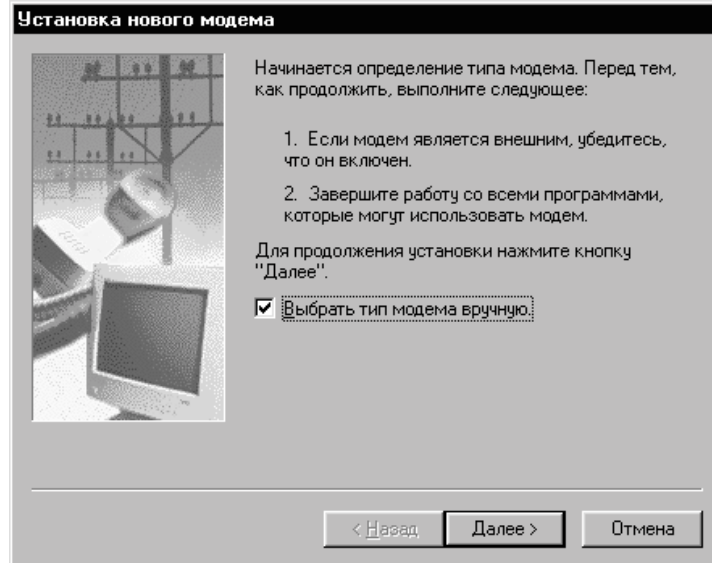


3. Выберите вкладку Port (порт) и выберите модем. Появится предупреждение, что будет запущен *мастер установки модема*. Выберите “Да” для продолжения.

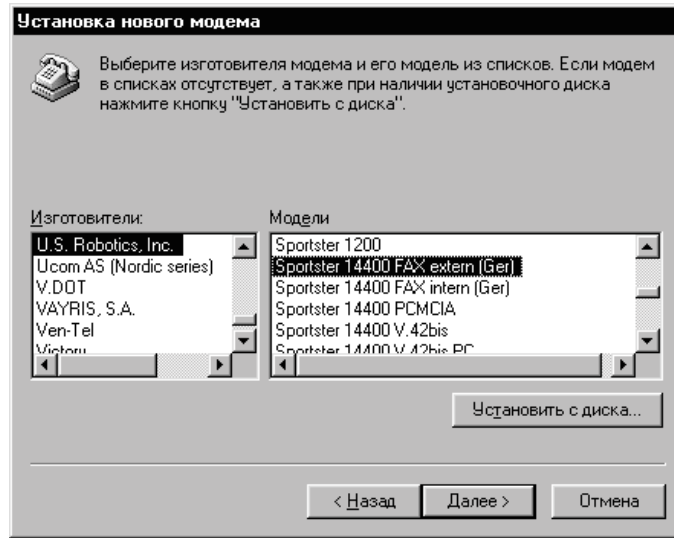


Конфигурация
связей и Тем

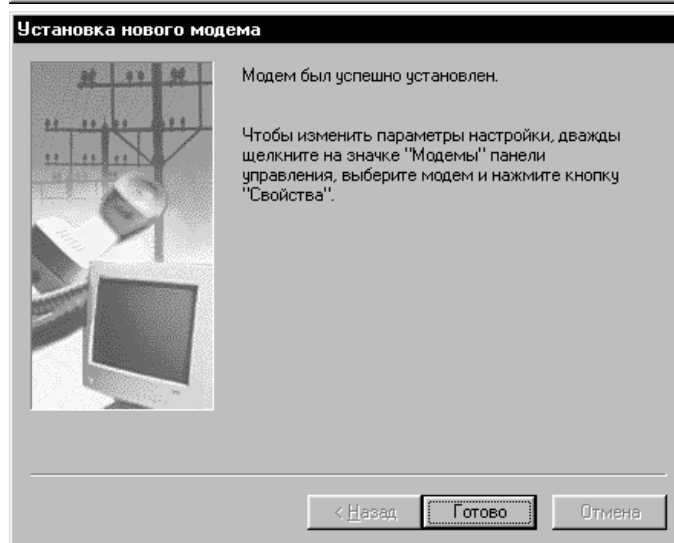
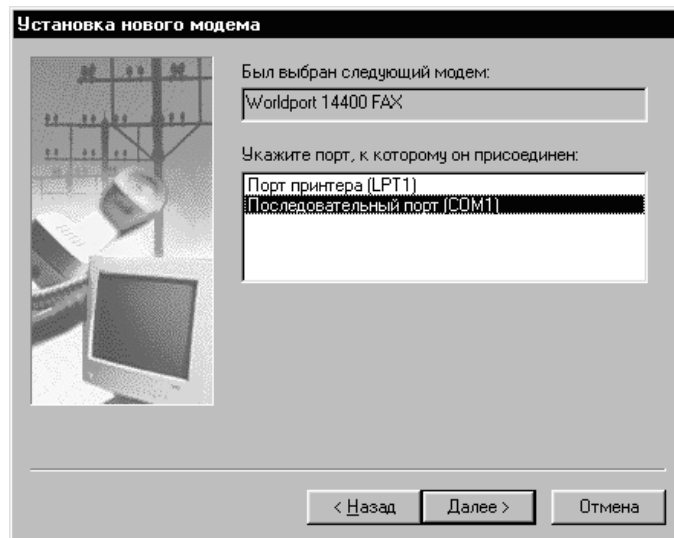
4. Выберите “Выбрать тип модема вручную” и щелкните **Далее>**.



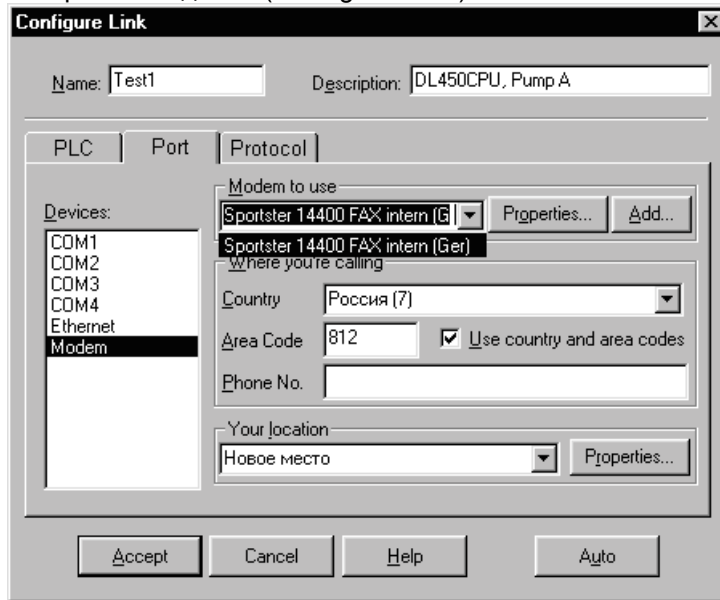
5. Выберите модем, который совпадает с вашим модемом. Если вы не обнаружили ваш модем в списке, хорошим решением будет выбрать Hayes совместимый модем. Щелкните **Далее>** для продолжения.



6. Выберите порт и щелкните **Далее>**. Затем щелкните **Готово**.

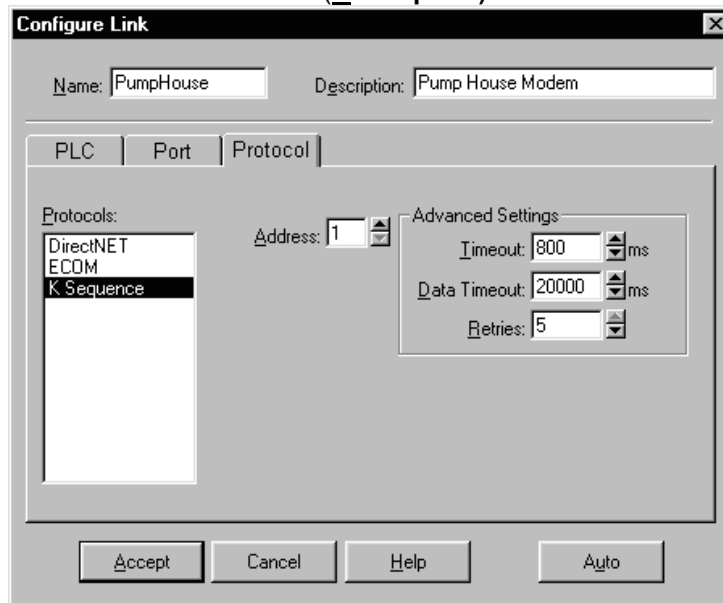


7. Сейчас вы можете увидеть имя конфигурации вашего модема в списке Modem to use. Выберите только что разработанную конфигурацию, чтобы сделать данную конфигурацию текущей. Теперь вы можете ввести номер телефона (Phone Number), режим звонка и время ожидания (Dialing Timeout).



Конфигурация
связей и Тем

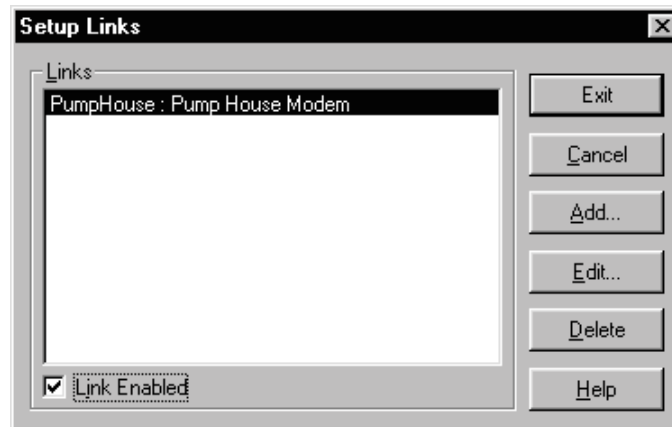
8. Щелкните по вкладке Protocol (протокол), чтобы вызвать диалог коммуникационного протокола. Это диалог позволяет вам дополнительно устанавливать коммуникационные параметры.
 - Выберете используемый коммуникационный протокол. Обратитесь к схеме в конце этой главы, чтобы определить какие протоколы поддерживаются процессорами ПЛК.
 - Введите адрес (Address) узла, если он отличен от 1.
 - Мы рекомендуем, чтобы вы изменили время ожидания (Timeout) на 2000 и количество попыток (Retries) на 5. Установленное по умолчанию время ожидания данных (Data Timeout) достаточно.
 - Введите уникальное имя (**Name**) связи и можете ввести необязательное описание (**Description**).



9. Щелкните по кнопке **Accept**, чтобы сохранить конфигурацию Связи. Перед вами появится следующий диалог. Если вы выберете **Yes** (да), DSDData Сервер свяжется с модемом и попытается проверить, действительно ли модем, подключенный к компьютеру, может связаться с модемом у ПЛК, используя параметры, определенные для данной Связи. Если вы выберете **No** (нет), конфигурация связи просто запишется на диск.



10. После успешного создания связи диалог конфигурации связей будет содержать связь, которую вы только что создали. Поле **Link Enable** (связь доступна) показывает, что коммуникационные параметры настроены правильно, и ПЛК доступен для использования.

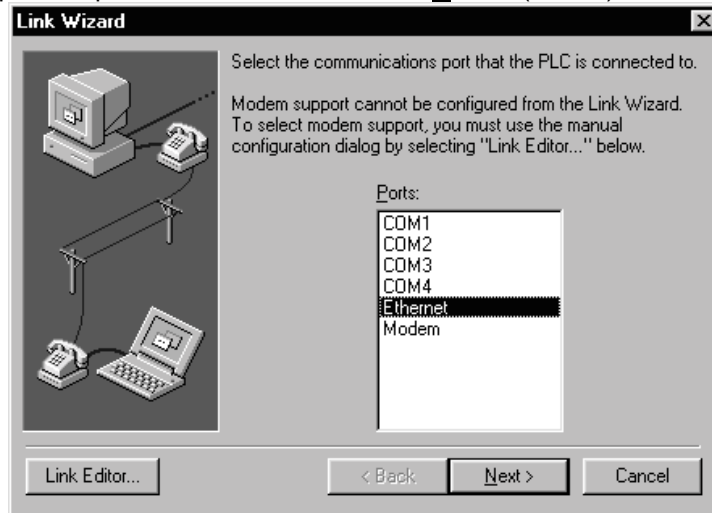


11. Выберете **Exit** для возвращения в главное окно DSDData Сервера.

Создание связи по сети Ethernet

Этот раздел описывает настройку связи, которая использует стандартную сетевую плату для подключения к модулю Ethernet в ПЛК.

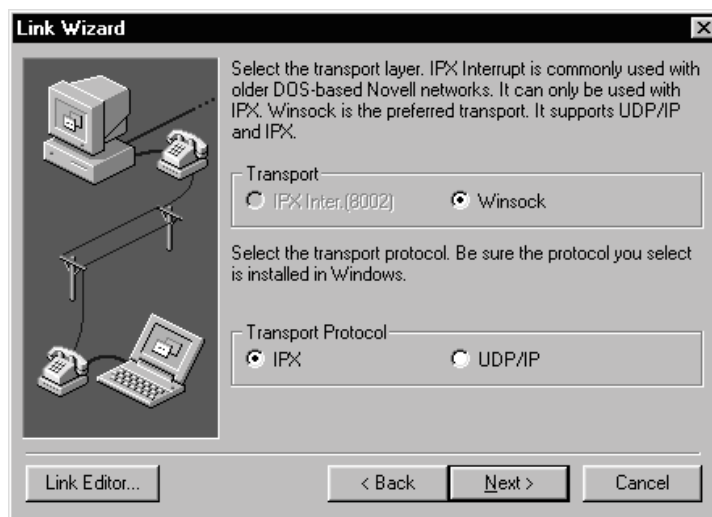
1. Мастер соединения может автоматически определить основные коммуникационные настройки сети. Выберете Ethernet, так же как вы выбирали порт и щелкните по клавише **Next->** (далее).



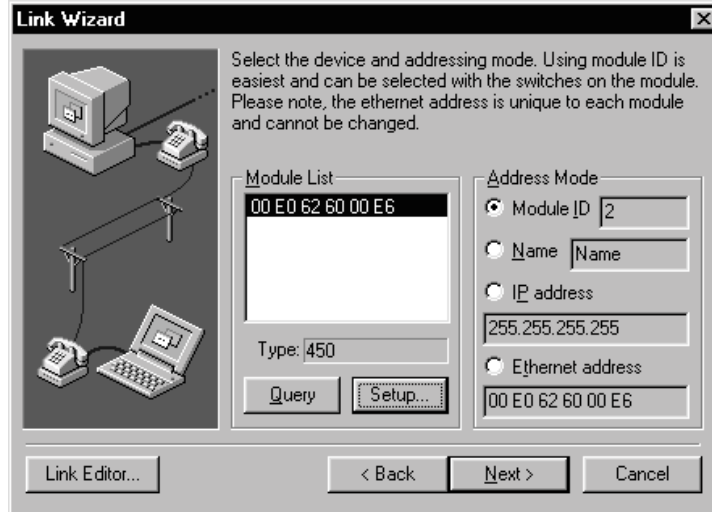
2. Этот диалог позволяет вам выбрать транспорт и транспортный протокол, которые вы будете использовать. Если операционная систем – Windows NT, доступен единственный транспорт (Transport) – Winsock. В операционных системах Window 3.1x и Windows 95 доступны две возможности: IPX Interrupt и Winsock.

Опция Transport Protocol (транспортный протокол) позволяет выбрать для использования протокол сети. Выбранный протокол должен быть загружен, и использоваться сетевой картой вашего компьютера.

Если вы выбрали сетевой протокол UDP/IP, то вы должны ввести правильный IP-адрес для каждого модуля Ethernet, только после этого вы сможете закончить процесс описания связи. Сетевая карта на вашем компьютере также должна иметь правильный IP-адрес.



- Теперь DSData Сервер сможет использовать транспорт и транспортный протокол, который вы определили, чтобы просканировать сеть для нахождения всех модулей Ethernet. После того, как поиск будет завершен, появится следующий диалог. Давайте рассмотрим две части этого диалога более подробно.



**Список модулей
(Module List)**

Каждый модуль Ethernet имеет уникальный 12-цифровой адрес, заданный производителем, который называется Ethernet-адрес. IP-адрес напечатан на ярлыке, который находится или сзади PC-карты модуля, или на панели самого модуля. В списке модулей (**Module List**) будут показаны все найденные модули Ethernet и отсортированы по Ethernet-адресу. Если вы выберете устройство в списке модулей, текущая конфигурация этого устройства будет показана в секции **Address Mode** (режим адресации).

Кнопка **Query** (запрос) запускает сканирование сети, используя Транспорт и Транспортный Протокол, определенные в предыдущем диалоге.

Кнопка **Setup...** (настройка) высвечивает диалоговое окно, которое позволяет определять Имя, описание и IP-адрес для выбранного модуля.

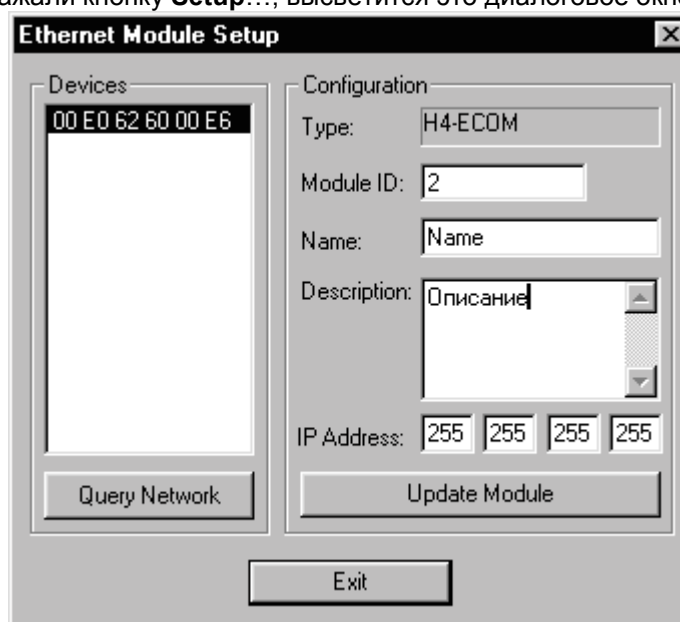
Кнопка **Link Editor...** вызывает диалоговое окно, в котором вы можете вручную сконфигурировать параметры Связи для определенного модуля Ethernet.

**Режим адресации
(Address Mode)**

Каждый модуль Ethernet должен иметь несколько способов для уникальной идентификации его DSData Сервером. Ethernet-адрес всегда уникален, но очень неудобен для запоминания человеком. Существуют три удобных идентификатора, определяемых пользователем: Идентификатор Модуля (**Module ID**), Имя (**Name**) и IP-адрес (**IP-address**) (Ethernet-адрес пользователь изменить не может). Выбор Режима Адресации (**Address Mode**) определяет способ, который DSData Сервер будет использовать для поиска Ethernet-модуля в вашей сети.

Важно помнить, какой бы идентификатор вы не выбрали, этот идентификатор должен быть уникальным в вашей сети. Если вы хотите изменить настройки модуля, щелкните по кнопке **Setup** и введите настройки, необходимые для вашей сети (этот диалог детально описан на следующих страницах). Нажмите **Next->** (далее), когда закончите.

Если вы нажали кнопку **Setup...**, высветится это диалоговое окно.



Выберите модуль в секции устройств (Devices). Текущие настройки этого модуля появятся справа, в секции конфигураций (Configuration). Прежде чем вы измените какое-либо значение, давайте рассмотрим каждый пункт более подробно.

- Идентификатор модуля (Module ID) – уникальный цифровой идентификатор, данный каждому модулю в сети. Этот номер может выставляться переключателями или устанавливаться пользователем (если номер на переключателях – 0). В инструкции по модулям Ethernet (H24-ECOM-M), приведена более подробная информация об установке идентификатора модуля.
- Поле Имя (Name) может состоять из 32 –алфавитно-цифровых символов. Значение, определенное производителем – ‘Name’. Если вы хотите использовать этот метод идентификации, вы должны изменить ‘Name’ так, чтобы идентификатор был уникальным в сети.
- Поле Описание (Description) может состоять из 32 – символьной алфавитно-цифровой строки.
- IP-адрес состоит из 4 частей, каждая из которых содержит 3 числа от 0 до 254 (255- зарезервировано для радиовещания). Адрес целиком должен быть уникальным в вашей сети. Если вы собираетесь использовать IP-адрес, вы должны заменить значение 255 в каждой из 4 частей на правильное. Спросите у своего сетевого администратора о доступных адресах, чтобы быть уверенным, что ваш IP-адрес не будет дублироваться в сети.



Примечание: Вы должны быть осторожны, чтобы не продублировать эти идентификаторы в сети. Вы не сможете завершить процесс создания Связи, если вы выберете идентификатор, который будет продублирован в сети.

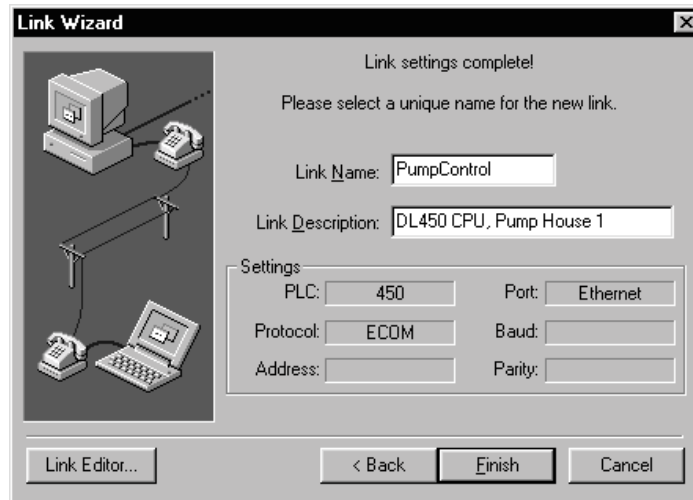
Если вы ввели новое значение, нажмите на кнопку **Update Module** (обновить модуль), чтобы записать это значение в EEPROM модуля Ethernet.

Кнопка **Query Network** (опросить сеть) активизирует опрос сети заново, используя предыдущие установки.

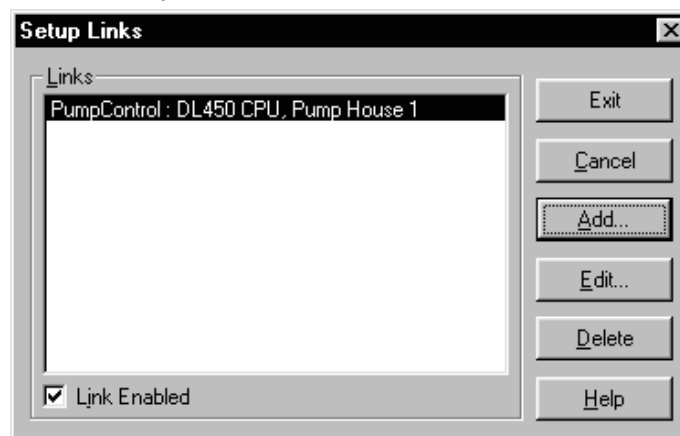
Нажмите кнопку Exit (Выход), когда выполните все изменения.

4. Теперь DSData Сервер сделает попытку соединиться с модулем, используя только что заданные параметры.

Если соединение произойдет успешно, появится запрос на ввод Имени Связи (Link Name)(до 16 символов) и описания связи (Link Description) (до 32 символов). Каждое Имя Связи должно быть уникальным. Нажмите **Finish** (финиш) для записи параметров связи на диск.



5. После успешного создания связи, Диалог настройки связей будет содержать Связь, которую вы только что создали. Индикатор Link Enable (связь доступна) показывает, что коммуникационные настройки связи верны и ПЛК готова к использованию.



6. Нажмите **Exit**, чтобы вернуться в главное окно DSData Сервера.

Создание Темы



Вы, наверное, помните из вступления, что надо выполнить две задачи, чтобы создать путь от вашего приложения к ПЛК через DSDData Сервер. Мы выполнили первую задачу – создали Связь с ПЛК. Теперь нам надо создать Тему (Topic), которая будет ссылаться на Связь.

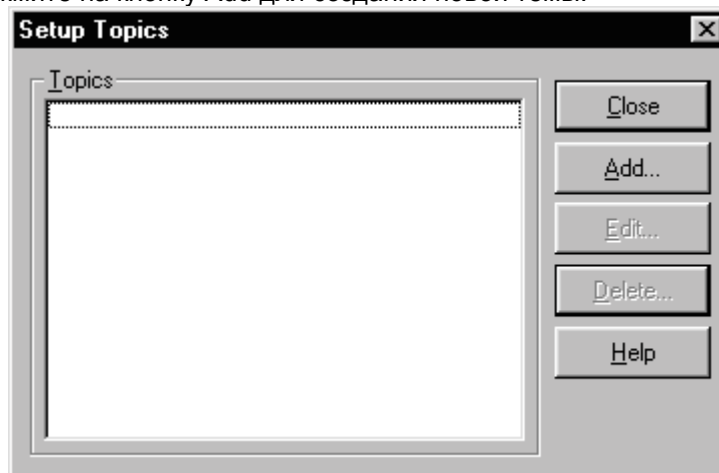
Примечание: Если вы уже имеете темы, и у вас имеются приложения, которые запущены и используют тему, тогда вы *не сможете* создать другую тему, пока не закроете приложения, открытые ранее. В этом случае, пункт меню Topics... будет отображен светло-серым цветом (не доступен).

1. Выберите **Setup|Topics** (Настройки|Тема), или щелкните по кнопке Create Topic (Создать Тему).

Щелкните здесь или
Щелкните здесь

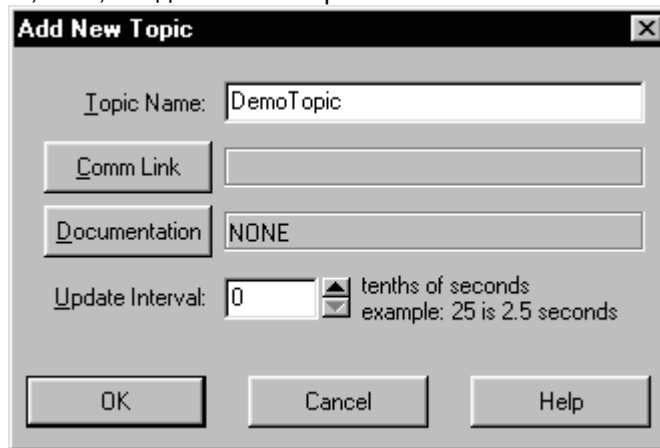


2. Нажмите на кнопку Add для создания новой темы.

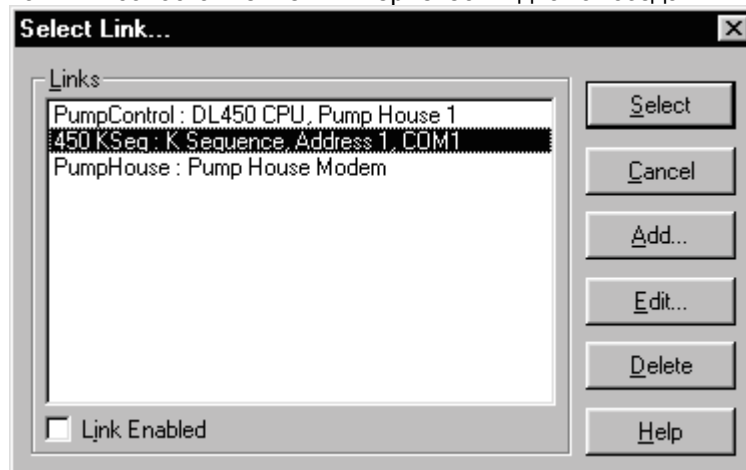


3. Введите имя темы. Имя Темы (Topic Name) может содержать до 40 алфавитно-цифровых символов, но некоторые приложения имеют ограничения на название темы. Вот несколько основных указаний для определения имени:
 - Оно не должно содержать скобок, двух вертикальных линий, разделителей и т.д.
 - Оно не должно начинаться с числа
 - Оно не должно содержать пробелов
 - Оно не различает регистра

Большинство примеров в этом руководстве будут использовать Тему – 'DemoTopic'; итак, введите DemoTopic в поле имени.



4. Щелкните по кнопке Comm Link (Коммуникационная Связь), чтобы вызвать диалог выбора связи. Этот диалог предоставляет инструмент для управления коммуникационной связью с ПЛК. С помощью него вы сможете создать Связь, удалить Связь, отредактировать параметры существующей Связи, или, как в нашем случае, выбрать существующую Связь для Темы, которую вы собираетесь использовать для общения с ПЛК. Щелкните по выбранной Связи, потом щелкните по кнопке **Select** (Выбор), чтобы выполнить сопоставление. Вы вернетесь в диалог создания Темы.



- Если вы используете программный пакет **DirectSoft32** для программирования контроллеров, вам хорошо знаком процесс создания псевдонима для ссылки на элемент данных в ПЛК. Псевдоним позволяет ссылаться на элемент данных в ПЛК через логическое имя вместо 'машинного имени' (например, «Кнопка Пуск» вместо X0). Кнопка **Documentation** (документация) соединяет псевдонимы, созданные в **DirectSoft32**, с используемым программным обеспечением, связывающимся с DSData Сервером. Например, лист Excel теперь может обращаться к «Кнопка Пуск» так же, как и **DirectSoft32**.

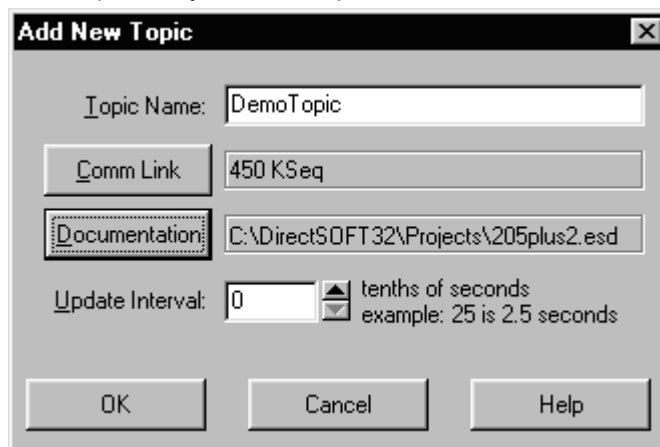
Кнопка Documentation вызовет стандартный диалог открытия файла, который позволяет вам найти каталог, в котором хранятся проекты **DirectSoft32** и выбрать соответствующий '.ESD' файл, содержащий псевдонимы, которые станут доступны.

Если у вас имеется '.ESD' файл, который вы хотите использовать, выделите его и затем нажмите **OK**. Для выхода из этого диалога без выбора файла нажмите **Cancel** (Отменить). Оба выбора вернут вас к диалогу открытия Темы.



- Время обновления (**Udate Interval**) позволяет вам задать DSData Серверу, как часто обновлять переменные, ссылающиеся на эту Тему. Определите интервал времени, введя значение между 0 и 999.9 (секунд).

Значение по умолчанию 0 означает, что DSData Сервер будет обновлять переменную так быстро, как это позволяет система.



Время обновления (Update Interval) – очень полезный инструмент для улучшения характеристик DDE обмена. Большинство DDE приложений могут создавать переменные, которые будут непрерывно запрашивать данные. Такие запросы часто называются «горячая линия». Примером такой связи является формула, введенная в Excel. Некоторые пакеты, к тому же, позволяют создать переменные, которые обновляются в ответ на запрос пользователя. Примером этого является макрос, подключенный к кнопке на листе Excel, т.к. данные обновляются только тогда, когда вы нажимаете на кнопку.

Большинство приложений поддерживают оба типа связей. Таким образом, вы можете истощать коммуникационные возможности на параметры, которые могут не меняться и потерять некоторые данные, которые меняются слишком часто.

Также вы можете попасть в ситуацию, когда необходимо использовать сеть ПЛК. Если вы зададите время обновления равным нулю, и каждая Тема использует отдельный ПЛК, тогда DSData Сервер будет пытаться обновлять данные от каждого ПЛК в сети так быстро, как это возможно. Это может привести к нехватке системных возможностей, если в сети имеются ПЛК, которые требуется обновлять чаще, чем другие.

В этой ситуации воспользуйтесь следующими советами.

- Горячая линия использует время обновления так, чтобы доставлять вам данные так часто, как вам это действительно необходимо, или когда вы ожидаете, что они изменятся. Если данные меняются медленно, не расходуйте ресурсы, используя короткий интервал обновления.
- Ручная связь – время обновления ни на что не влияет, потому что DSData Сервер активизирует этот тип запроса тогда, когда пользователь запрашивает данные вручную. В этом случае DSData Сервер не ждет истечения временного интервала для запроса данных.
- Сеть ПЛК – используйте минимальное время опроса для тех ПЛК, которые имеют критические требования, и большее для тех, которые не имеют.

Если существуют несколько входов/выходов в одном ПЛК, которые требуется часто опрашивать, а некоторые не требуется, вы можете создать подтему, которая будет иметь отличный интервал обновления, тогда как раньше нам приходилось создавать несколько тем с разными интервалами обновления. Смотрите создание подтем далее в этой главе.

- Щелкните **OK** для возврата в главное окно DSData Сервера, где в списке отобразится только что созданная тема. Заметьте, что в колонке Value справа от состояния Status ничего не отображается.

Topic/SubTopic	Item	Element	Value	Event
<input type="checkbox"/> DemoTopic	<Link/PLC/DocDB>		450 KSeq/450/...	
<input type="checkbox"/> DemoTopic	Status			???
<input type="checkbox"/> DemoTopic	Enable			???
<input type="checkbox"/> DemoTopic	_StatusValue			???
<input type="checkbox"/> DemoTopic	_Xfers			???
<input type="checkbox"/> DemoTopic	_Retries			???
<input type="checkbox"/> DemoTopic	_Errors			???
<input type="checkbox"/> DemoTopic	_XferRate			???
<input type="checkbox"/> DemoTopic	_Writes			???
<input type="checkbox"/> DemoTopic	_UpdateInterval		0 (1/10 sec.)	Not Active

- Как только другая программа, использующая DDE или OPC для чтения и записи тегов этой темы, подключится к серверу, в поле значения Value статуса отобразится Good (если со связью все хорошо), и OPC (или DDE) элементы (Item) появятся в колонке элементов. Столбец Value (значение) показывает текущие состояния элементов и столбец Event (событие) отображает действия, выполняемые через OPC или DDE, такие как Read (чтение) и Write (запись).

Topic/SubTopic	Item	Element	Value	Event
<input type="checkbox"/> DemoTopic	<Link/PLC/DocDB>		450 KSeq/450/...	
<input type="checkbox"/> DemoTopic	Status		Good[1]	Not Active
<input type="checkbox"/> DemoTopic	Enable		Enabled[1]	Not Active
<input type="checkbox"/> DemoTopic	_StatusValue		Active[1]	Not Active
<input type="checkbox"/> DemoTopic	_Xfers		382	Not Active
<input type="checkbox"/> DemoTopic	_Retries		0	Not Active
<input type="checkbox"/> DemoTopic	_Errors		0	Not Active
<input type="checkbox"/> DemoTopic	_XferRate		10	Not Active
<input type="checkbox"/> DemoTopic	_Writes		0	Not Active
<input type="checkbox"/> DemoTopic	_UpdateInterval		0 (1/10 sec.)	Not Active
<input type="checkbox"/> DemoTopic		V2000	16	Read
<input type="checkbox"/> DemoTopic		V2001	0	Read

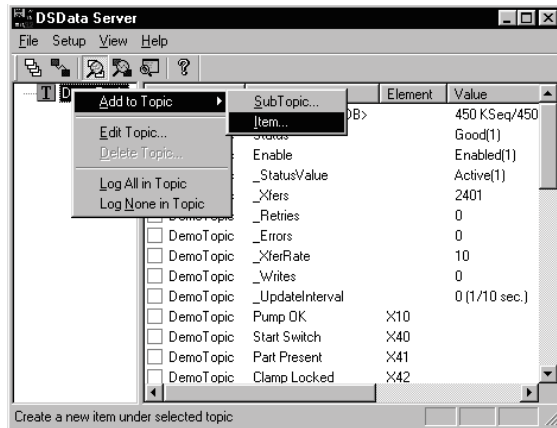
Создание элементов в теме

Существуют две возможности, с помощью которых элемент может быть создан:

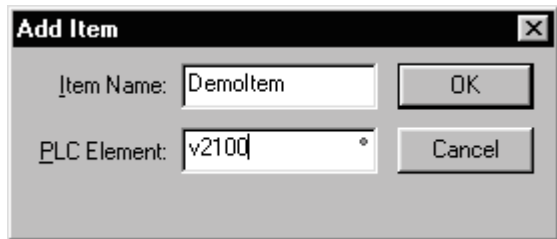
- 1) в DSData Сервере, как показано ниже и
- 2) клиентом OPC или DDE.

В этом разделе описан первый способ.

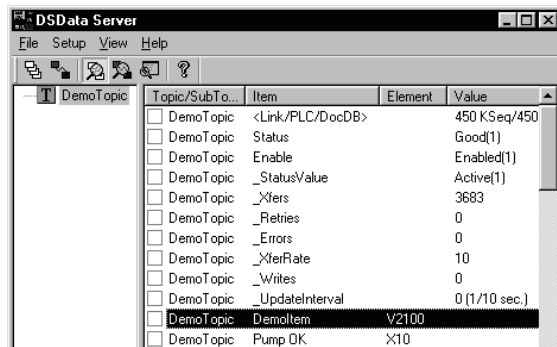
Для создания нового элемента щелкните правой кнопкой мыши на теме и выберите Add to Topic.../Item.



Далее задайте имя элемента и введите элемент ПЛК, на который будет ссылаться элемент.



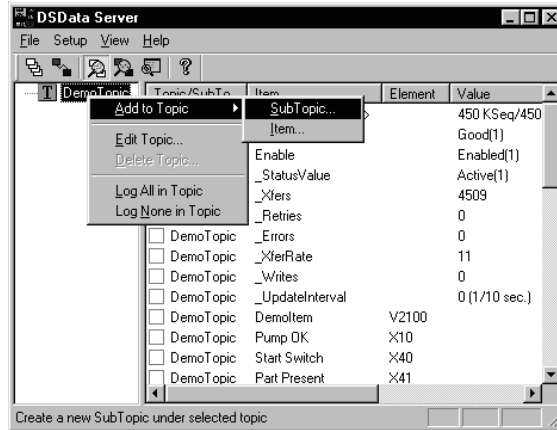
Теперь новый элемент будет в списке темы.



Создание подтем

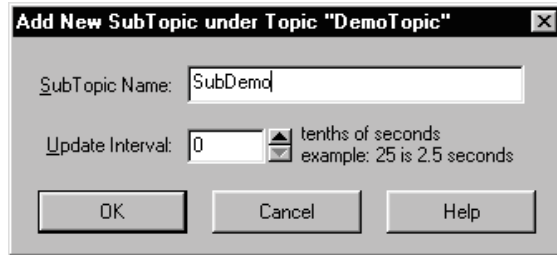
Теперь можно создать несколько подтем в каждой теме, которые позволяют задать различные времена опроса для каждой подтемы. Теперь Вы можете создать подтемы с большим или меньшим приоритетом, которые используют одну и ту же коммуникационную связь, и базу данных псевдонимов **DirectSoft32** назначенную для родительской темы.

Для создания подтемы щелкните правой кнопкой мыши на теме и выберите Add to Topic.../SubTopic.



Конфигурация
связей и Тем

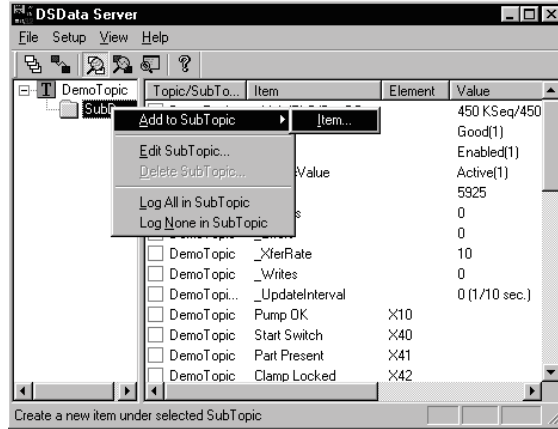
Далее задайте имя подтемы и определите время опроса.



Создание элементов в подтеме

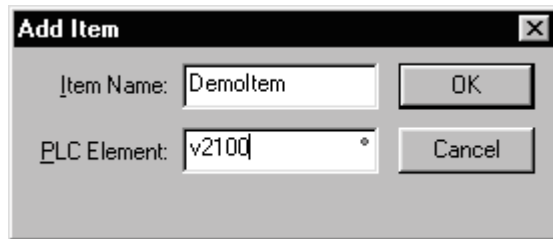
Процесс создания элементов внутри подтемы, фактически не отличающийся от создания элемента внутри темы, приведен ниже для полноты.

Для создания нового элемента в подтеме щелкните правой кнопкой мыши на подтеме и выберите Add to SubTopic.../Item.

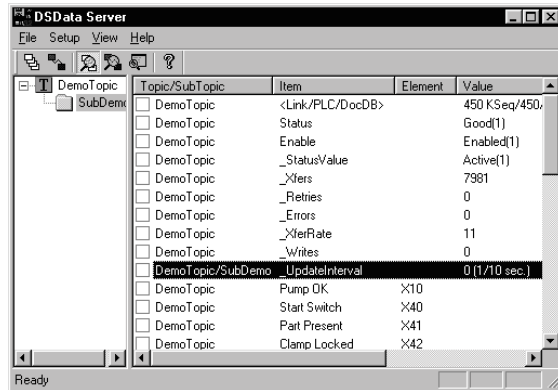


Конфигурация
связей и Тем

Далее задайте имя элемента и введите ячейку ПЛК, на который будет ссылаться этот элемент.



Теперь новый элемент будет в списке темы.



Ответы на вопросы

Могут ли несколько Тем использовать одну Связь?

Да. Имеет смысл использовать несколько тем для одной связи, если установить для Тем разное время обновления. Это позволит вам быстрее обновлять переменные, которые быстро меняются, за счет медленно меняющихся переменных, которые будут редко обновляться. Однако, лучший способ для создания нескольких тем является использование подтем внутри темы для управления определенными элементами.

Могу я получить несколько данных через одну DDE переменную?

Нет, каждый DDE запрос, будь то запрос на чтение или на запись, может передать только единицу данных. Вы должны создать отдельную переменную для отдельных данных, к которым Вы хотите получить доступ.

В списке модулей (Module List) мастера связей нет модулей Ethernet. Что неверно?

1. Проверьте индикатор связи (LINKGD) на самих модулях Ethernet и на сетевой карте вашего ПК. Это самый простой способ проверить кабель. Обратитесь к руководству пользователя по модулю Ethernet, если индикатор LINKGD не горит.
2. Протокол, который был выбран как транспортный протокол, должен быть загружен на ваш компьютер и подключен к сетевой карте. В Windows 95 и в Windows NT используйте Сетевое Окружение для проверки того, что протокол правильно загружен и подключен к вашей сетевой карте.
3. Используйте программу NetEdit для нахождения модуля.

У меня нет UDP/IP драйвера, что я могу использовать?

Используйте драйвер TCP/IP протокола, который поставляется вместе с операционной системой. Когда Вы устанавливаете TCP/IP протокол, то получаете два компонента протокола TCP и UDP. TCP – это часть, ориентированная на потоки, UDP – часть протокола, называемая «независимые пакеты». Для ПЛК удобнее пакетное общение, поэтому UDP является логичным выбором.

Сервер DHCP присваивает IP адрес, когда я регистрируюсь. Всегда ли это будет работать?

Сервер Динамической Конфигурации Протоколов (Dynamic Host Configuration Protocol) присваивает IP адрес клиентам, когда те регистрируются в сети. Это будет работать до тех пор, пока Вы будете иметь доступ одновременно к DHCP серверу и к модулю Ethernet, т.е. они должны находиться в одной сети и быть доступны. Если Вы не смогли получить доступа к DHCP серверу во время загрузки компьютера, IP адрес не будет присвоен вашей сетевой карте. В этом случае Вы не сможете использовать UDP/IP протокол для доступа к модулю Ethernet.

Если Вы установили протокол IPX дополнительно к TCP/IP, то сможете использовать IPX связь с модулем Ethernet, даже когда не сможете получить доступа к DHCP серверу.

Установка сетевой карты

Сетевой протокол и привязка

Когда говорят, что протокол является «оболочкой» сетевой карты, имеют в виду, что карта может использовать протокол для сетевого общения, если он пересылает запросы от операционной системы, также дающей ему указания.

Windows 95 и Windows NT могут поддерживать несколько протокольных оболочек для одной сетевой карты. Если вы хотите использовать UDP/IP, мы рекомендуем также установить IPX протокол. В этом случае, при неправильном присвоении IP адреса, вы все равно сможете использовать сеть.

Сетевая конфигурация Windows 95

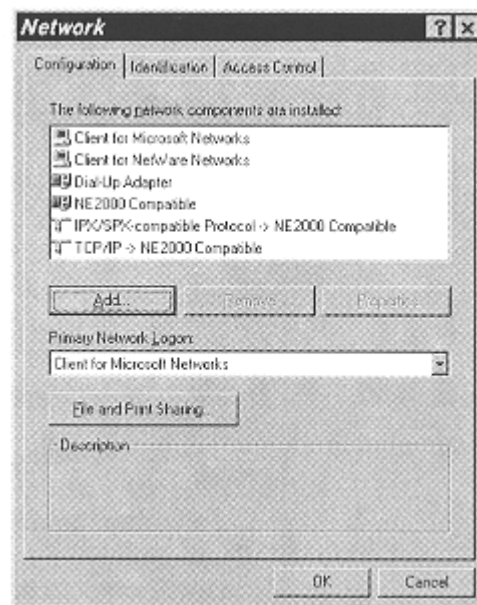
Для проверки того, что сетевой протокол установлен правильно, и что он присоединен к сетевой карте вашего компьютера, щелкните правой клавишей мыши по иконке Сетевое Окружение на рабочем столе. В выпадающем меню выберите Свойства. Появится следующий диалог:

На этом компьютере установлено два адаптера (соединительных устройств) – адаптер Dial-Up (с доступом по коммутированной линии) и совместимая с NE2000 сетевая карта.

IPX/SPX совместимый протокол загружен, знак «->» означает, что протокол присоединен к соответствующей сетевой карте.

TCP/IP протокол также загружен и привязан к NE2000 совместимой сетевой карте.

Если в компьютере установлен только один адаптер, символ “->” не появится.



Для добавления сетевого протокола в конфигурацию сети, выполните следующие действия в диалоге Сеть (Network):

1. Щелкните по кнопке **Add...** (добавить)
2. Выберите протокол и снова щелкните по кнопке **Add...**
3. Под “Manufactures” (производитель) выберите Microsoft
4. Под “Network Protocols” выберите протокол, который Вы хотите добавить
5. Щелкните **OK**
6. Щелкните **OK**

Для завершения установки протокола понадобится диск Windows 95.

Установка сетевого протокола в Windows 95

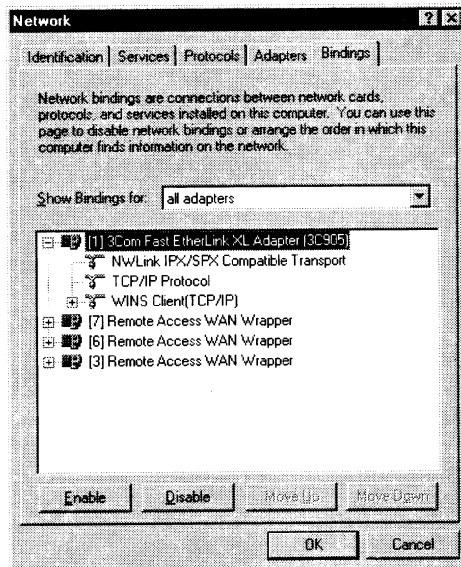
Конфигурация сети в Windows NT 4.0

Для проверки того, что сетевой протокол установлен правильно, и что он присоединен к сетевой карте вашего компьютера, щелкните правой клавишей мыши по иконке Сетевое Окружение на рабочем столе. В выпадающем меню выберите Свойства. Щелкните по вкладке Bindings (привязка), в поле "Show Bindings for:" (показать привязки для) выберите "all adapters" (все адаптеры). Щелкните по символу "+" рядом с названием адаптера для отображения списка протоколов, которые привязаны к соответствующей сетевой карте. Появится следующий диалог:

На этом компьютере установлен один сетевой адаптер – 3Com Fast EtherLink XL Adapter (C905).

Загружен NWLink IPX/SPX совместимый транспорт.

Загружен также протокол TCP/IP



Установка сетевого протокола в Windows NT

Для добавления сетевого протокола в конфигурацию сети, выполните следующие действия в диалоге Сеть (Network):

1. Щелкните по вкладке Protocols (протоколы)
2. Щелкните по кнопке **Add...** (добавить)
3. Под "Network Protocols" выберите протокол, который Вы хотите добавить
4. Щелкните **OK**

Для завершения установки протокола понадобится диск Windows NT.

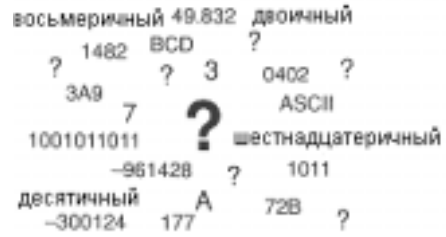
Создание Тегов

В этой главе...

- Система счисления ПЛК
 - Теги регистров
 - Теги таймеров и счетчиков
 - Теги дискретных входов/выходов
 - Использование псевдонимов и обозначений
 - Модификаторы операции записи
 - Наблюдение за коммуникационной связью
 - Встроенные системные элементы
 - Управление коммуникационной связью
-

Система счисления ПК

Если вы новичок в использовании ПЛК контроллеров **DirectLOGIC**, прежде всего, необходимо изучить, как наши ПЛК работают с числами. Вы можете заметить, что производители ПЛК использует разные методы представления чисел в ПЛК. Вам надо затратить некоторое время, чтобы понять, как числа используются в ПЛК **DirectLOGIC**.



Так же как и все компьютеры, ПЛК хранят и манипулируют числами в двоичной форме: набор ноликов и единиц. Так почему же мы тогда имеем дело с числами различных форматов? Потому что для некоторых применений специальное представление чисел более удобно. Иногда мы используем числа для представления размера или количества чего-то. Другие числа указывают на месторасположение или являются адресом, или временем. В технике, мы добавляем наименования единиц, чтобы придать числам специальный смысл.

Ресурсы ПЛК

ПЛК имеет ограниченное количество ресурсов, зависимое от модели. В понятие «ресурс» мы включаем память переменных (V-память), память входов/выходов, таймеры, счетчики и т.д. Большинство модулей ПЛК объединяют входы/выходы в группы по восемь. Вообще, все ресурсы ПЛК пронумерованы в восьмеричной системе. Компьютеру проще обчислять группы из восьми, чем из десяти, так как восемь – это степень двойки.

Восьмеричное счисление позволяет просто считать предметы в группах по восемь. На рисунке изображено восемь кружков. Количество кружков в десятичной системе счисления – «8», а в восьмеричной – «10» (цифры 8 и 9 не применяются в восьмеричной системе). В восьмеричной системе 10 означает – одна группа по 8 плюс 0 (нет отдельных кружков).

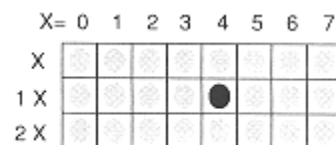


На следующем рисунке изображено две группы из восьми кружков. В восьмеричной системе мы имеем «20» предметов, т.е. две группы по 8 плюс 0 отдельных. Не говорите «двадцать», говорите «два ноль восьмеричные». Это позволит ощутить разницу между числами в разных системах счисления.



После подсчета ресурсов ПЛК наступило время перейти к получению доступа к этим ресурсам (что имеет отличия). Процессор получает доступ к ресурсам через их восьмеричный адрес. Восьмеричный адрес тот же, что и восьмеричное число, кроме того, что отсчет начинается с нуля. Цифра ноль является значимой для компьютера, и мы не можем пропустить ее.

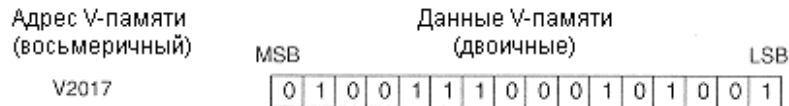
Наши кружочки расставлены по ячейкам в прямоугольнике. Для доступа к ячейке, команда ПЛК должна адресоваться к ее расположению так, как показано на рисунке справа. Если это счетчик, то «СТ14» будет указывать на место, помеченное черным кружком.



V-память

Память переменных (называемая V-память) хранит данные программы и параметры. Место в V-памяти или адрес V-памяти – исчисляются в восьмеричной системе. Например, V2073 правильное значение, а V1983 – неправильное («9» и «8» неверные восьмеричные цифры).

Каждая ячейка памяти – это одно слово, т.е. 16 бит. Сейчас мы рассмотрим каждый бит в слове V-памяти. Справа находится младший значимый бит (LSB), а слева старший значимый бит (MSB). Мы используем эти понятия, имея в виду, относительный двоичный вес бита (в какую степень бит возводит двойку).

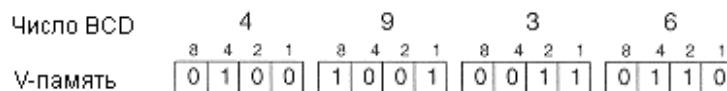


Хотя V-память и состоит из шестнадцати бит, но очень редкой программе необходимо использовать только один бит. Мы используем инструкции или визуальные инструменты, которые позволяют нам использовать двоичный, десятичный, восьмеричный и двоично-десятичный форматы. Все эти форматы преобразуются и хранятся как двоичные.

Часто спрашивают, «Как мне показать, какие у меня числа: двоичные, восьмеричные, двоично-десятичные (BCD) или шестнадцатеричные?». Обычно на этот вопрос невозможно ответить, только посмотрев на данные. Но есть ключевые моменты: источник или механизм, которые пишут в V-память и потребитель, который читает эти данные, должны использовать одинаковые типы данных (т.е. восьмеричный, шестнадцатеричный, двоичный или какой либо еще). V-память только место хранения. Она не преобразует и не передает данные.

Двоичный код десятичного числа

Так как люди привыкли считать в десятичной системе, мы предпочитаем вводить и видеть данные в ПЛК в десятичной системе счисления (через операторский интерфейс). К сожалению, компьютер более эффективно работает с двоичными цифрами. Компромиссным решением является двоично-десятичный (BCD) формат данных. В BCD формате диапазон цифр от 0 до 9, и они хранятся как четыре двоичных бита. Это позволяет каждой ячейке V-памяти содержать четыре цифры BCD. С диапазоном числа от 0 до 9999.



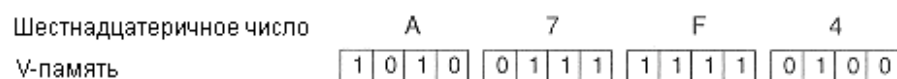
В простом двоичном формате 16-битовое слово может содержать числа от 0 до 65535. Если мы храним данные в BCD формате, диапазон данных уменьшается – от 0 до 9999. Многие математические инструкции выполняются над BCD данными, **DirectSoft32** и ручной программатор позволяют нам вводить и видеть BCD данные. Специальные RLL инструкции преобразует данные BCD в двоичные и обратно.

Шестнадцатеричные числа

Шестнадцатеричные числа – это простые BCD числа, но в них используются полностью все 4 бита цифры. Используются буквы от A до F, чтобы расширить цифры десятичной системы счисления (от 0 до 9).

Десятичный	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Шестнадцатеричный	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Четыре цифры шестнадцатеричного числа могут содержать все 65536 значений в одном слове памяти. Диапазон числа от 0000 до FFFF (шестнадцатеричной системы). Шестнадцатеричная система очень удобна для просмотра двоичных данных.



Теги регистров

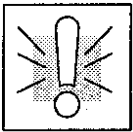
Создание тегов регистров «только для чтения»

Ваше приложение должно знать, какого типа данные вы хотите получить. Вы можете это сделать путем создания описания данных, которое часто называется «Тег». Для создания тега вам необходимо знать не только расположение данных в ПЛК, но также и формат данных.

Давайте посмотрим на тег, который вы можете создать, используя Microsoft Excel.

```
=Dpdata|DemoTopic!V1000:B'
```

Этот тег предоставляет доступ к текущему значению счетчика СТО. Из инструкции пользователя вы знаете, что счетчик хранит текущее значение в BCD формате. При определении тега вы должны добавить модификатор «:B», для доступа к данным в формате BCD.



ПРЕДУПРЕЖДЕНИЕ: Следующий пример использует ячейку V-памяти V2000. Если этот пример запустить, он запишет значение в данную ячейку. Прежде чем запустить пример, пожалуйста, убедитесь, что вы не используете эту ячейку для чего-либо важного.

Создание тегов регистров для чтения и записи

Создание тегов, которые могут записывать значение в память ПЛК, абсолютно не похоже на создание тегов только для чтения. Для записи данных в ПЛК большинство приложений использует макросы, скрипты или специально разработанные для этой цели графические объекты. Мы не будем вдаваться в детали о создании тегов чтения / записи в этой главе, так как каждое приложение использует собственные методы для выполнения DDE запроса на чтение / запись. Теги чтения / записи будут описаны детально в следующих главах, для каждого приложения отдельно. Сейчас посмотрим на макрос Microsoft Excel.

Этот пример записывает значение, хранящееся в ячейке (ряд 1, колонка 1) в память ПЛК по адресу V2000.

```
Sub WriteDataBtn_Click()
    Chanel = DDEInitiate ("Dpdata", "DemoTopic")
    DdePoke Chanel, "V2000", Cells(1, 1)
    DdeTerminate Chanel
End Sub
```

Этот пример читает данные из памяти ПЛК по адресу V2001 и сохраняет их в ячейке Excel (ряд 1, колонка 2).

```
Sub WriteDataBtn_Click()
    Chanel = DDEInitiate ("Dpdata", "DemoTopic")
    Cells(1, 2) = DdeRequest ("V2001", Chanel)
    DDETerminate Chanel
End Sub
```

Модификаторы данных

DSDData Сервер использует модификаторы данных для того, чтобы сделать процесс преобразования данных настолько простым, насколько это возможно. Просто добавьте правильный модификатор к определению данных в теге, который вы создали, и DSDData Сервер будет преобразовывать данные каждый раз, когда будет необходимо выполнить операцию чтения / записи.

Однако, вы должны точно знать формат данных в ПЛК, прежде чем вы сможете правильно создать тег.

Вот список возможных модификаторов для всех PLCDirect и PLCDirect совместимых процессоров

- Без модификатора Десятичное значение (16 бит)
Десятичное значение (32 бита)
- :D Преобразовать в BCD (16 бит)
- :B Преобразовать в BCD (32бита)
- :DB Чтение отдельного бита в слове (недоступно для записи бита)
- :np

Этот модификатор возможно использовать с процессорами DL250, DL350, DL450.

- :R Преобразовать в реальное число (32 бита)

Эти два модификатора возможно использовать с процессорами совместимыми с DL330, DL330P, DL340 и PLCDirect DL305.

- :W Слово (16 бит)
- :WB Слово (32 бита)

Примеры

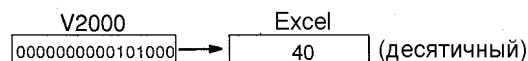
Посмотрите, как использовать модификаторы данных при запросе данных от ПЛК. Модификаторы применяются таким же образом при записи данных в ПЛК. Все примеры могут работать в листе Excel, если использовать формулу ячейки “=Dsddata|DemoTopic!DataItem:modifier”.

Без модификатора

Без модификатора получаем 16 битовое значение в десятичном формате.

Пример без модификатора

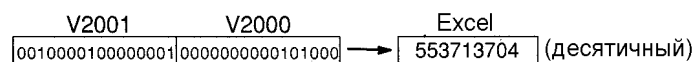
=Dsddata|DemoTopic!"V2000"

**:D
Двойное слово**

:D указывает на два последовательных 16-битовых регистра, как на одно 32-битовое. Например, если вы используете адрес V2000, будет использоваться содержимое ячеек V2000 и V2001.

Пример с модификатором :D

=Dsddata|DemoTopic!"V2000:D"

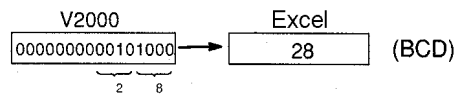


:B
Преобразовать в
BCD

:B преобразует двоичные числа в BCD.

Пример с модификатором :B

=Dpdata|DemoTopic!"V2000:B"

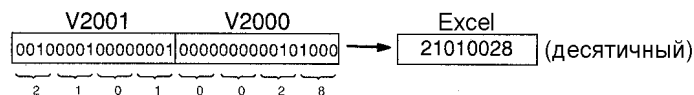


:DB
Двойное слово с
преобразование
м в BCD

:DB указывает на два последовательных 16-битовых регистра, как на одно 32-битовое. Например, если вы используете адрес V2000, будет использоваться содержимое ячеек V2000 и V2001.

Пример с модификатором :DB

=Dpdata|DemoTopic!"V2000:DB"



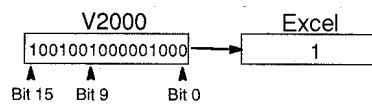
:nn
Бит в слове
V-памяти (только
для чтения)

:nn – это число от 0 до 15, которое определяет один бит из 16 в V-памяти. Если бит включен, возвращается 1, иначе 0. Проверяется только бит, определенный модификатором. Остальные биты игнорируются.

Этот модификатор может быть использован только для чтения данных, он не будет работать как модификатор для записи.

Пример с модификатором nn, равным 9

=Dpdata|DemoTopic!"V2000:9"

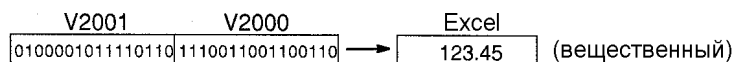


:R
Преобразует в
реальное число
(только для
DL250,350,450)

:R преобразует число с плавающей точкой стандарта IEEE 32-бит в реальное число.

Пример с модификатором :R

=Dpdata|DemoTopic!"V2000:R"



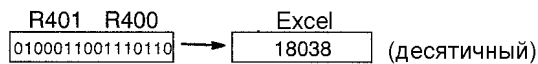
Следующие два модификатора применимы только с процессорами DL330, DL330P, DL340 и с DL305 совместимыми процессорами. Они применяются для доступа к регистрам с различными размерами. Перечисленные процессоры имеют и 8-битовые регистры и 16-битовые. Этот модификатор позволяет получить доступ к двум последовательным 8-битовым регистрам как к одному 16-битовому.

:W Чтение 16 бит (DL330, DL330P, DL340 и DL305 совместимые)

:W ссылается на два последовательных 8-битовых регистра как на один 16-битовый. Например, если вы определяете адрес R400, будет использоваться содержимое ячеек R400 и R401.

Пример с модификатором :W

=Dsdata|DemoTopic!"V2000:W"



:WB Чтение 16 бит (DL330, DL330P, DL340 и DL305 совместимые)

:WB ссылается на два последовательных 8-битовых регистра как на один 16-битовый и переводит значение в BCD. Например, если вы определяете адрес R400, будет использоваться содержимое ячеек R400 и R401.

Пример с модификатором :WB

=Dsdata|DemoTopic!"V2000:WB"



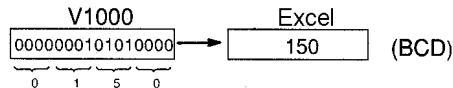
Теги таймеров и счетчиков

DL105, DL205, DL350, DL405 и DL405 совместимые процессоры

В DL105, DL205, DL350, DL405 и DL405 совместимых процессорах обыкновенные таймеры (TMR, TMRF) и счетчики (CNT) хранят свои текущие значения в V-памяти. Текущие значения таймеров хранятся в памяти, начиная с адреса V0000, а текущие значения счетчиков - начиная с адреса V1000. Эти значения хранятся в BCD формате и имеют диапазон от 0 до 9999, и все теги, ссылающиеся на эти значения, должны использовать модификатор “:B”.

Вот пример тега, отображающего текущее значение счетчика CT0.

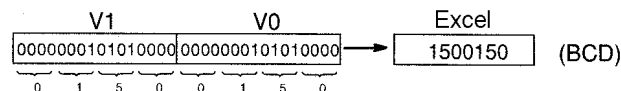
=Dsdata|DemoTopic!'V1000:b'



В DL105, DL205, DL350, DL405 и DL405 совместимых процессорах накапливающий таймер (TMRA, TMRAF) и таймеры больше / меньше (UDC) занимают две последовательные ячейки памяти для хранения текущих значений. Эти значения также хранятся в формате BCD, и диапазон значений 00000000-99999999. Все теги, ссылающиеся на эти значения, должны использовать модификатор “:DB”.

Вот пример тега, отображающего текущее значение накапливающего таймера T0.

=Dsdata|DemoTopic!'V1000:db'

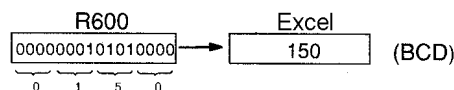


DL330, DL330P, DL340 и DL305 совместимые процессоры

В DL330, DL330P, DL340 и DL305 совместимых процессорах значение таймер/счетчиков хранятся в регистрах таймеров и счетчиков. Например, текущее значение таймер/счетчика CT600 будет регистр R600. К сожалению, существует разница в хранении значений таймеров и счетчиков. **Текущее значение счетчиков хранится в BCD формате**, диапазон значений от 0000 до 9999, и теги должны иметь модификатор “:B”. **Текущее значение таймеров хранится в десятичном формате**, диапазон значений от 0000 до 9999, и теги не должны иметь модификаторов.

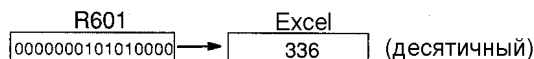
Вот пример тега, отображающего текущее значение счетчика, хранящееся в CT600.

=Dsdata|DemoTopic!'R600:b'



Вот пример тега, отображающего текущее значение таймера, хранящееся в T601.

=Dsdata|DemoTopic!'R601'



Теги дискретных входов/выходов

Нумерация дискретных входов/выходов



В PLCDirect совместимых процессорах дискретные входы/выходы адресуются с помощью указания типа памяти и номера, т.е. X0, Y7, C26, IO000, T0, SP540 и т.д. Возможные типы памяти и максимальное число точек для каждого типа процессора различно. Обратитесь к руководству пользователя, которое вы получили вместе с процессором, для определения типа и количества точек.

Примечание: Если вашему приложению необходимо записывать входы/выходы в контроллер, убедитесь, что порт ПЛК, который вы собираетесь использовать, поддерживает протокол K-Sequence. Обратитесь к приложению А, что бы определить, поддерживает ли ваш ПЛК этот протокол. Также убедитесь, что Связь, которую вы создали, использует K-Sequence протокол (или ECOM протокол, если вы используете модули D2-ECOM или D4-ECOM). DirectNet протокол не поддерживает возможности записи точек дискретного входа/выхода.

Создание тегов дискретных входов/выходов только для чтения

Создание тегов дискретных входов/выходов только для чтения существенно не отличается от создания тегов регистров, за исключением того, что они *не требуют модификаторов данных*.

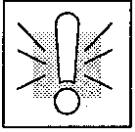
Все теги дискретных входов/выходов возвращают 1, если точка (вход/выход) включена, и 0 если выключена.

Посмотрите на следующие примеры тегов дискретных входов/выходов, которые получают данные из дискретных точек DL105, DI205, DI350, DL405 и DL405 совместимых процессоров:

- =Dpdata|demotopic!'X0' Вход 0
- =Dpdata|demotopic!'Y7' Выход 7
- =Dpdata|demotopic!'C47' Управляющее реле 47
- =Dpdata|demotopic!'SP540' Специальное реле 540
- =Dpdata|demotopic!'T0' Таймер 0

Посмотрите на следующие примеры тегов дискретных входов/выходов, которые получают данные из дискретных точек DL330, DL330P, DL340 и DL305 совместимых процессоров:

- =Dpdata|demotopic!'C160' Управляющее реле C160
- =Dpdata|demotopic!'IO000' Точка входа/выхода 0
- =Dpdata|demotopic!'T600' Таймер/счетчик T600



ПРЕДУПРЕЖДЕНИЕ: Следующий пример использует управляющее реле C26. Если этот пример запустить на выполнение, значение по этому адресу может быть изменено. Пожалуйста, прежде чем запускать пример убедитесь, что это реле не используется для чего-либо важного.

Создание тегов входов/выходов для чтения и записи

Для активизации дискретного входа/выхода вы должны записать в него значение 1. Для деактивизации – записать 0. Следующий пример Microsoft Excel демонстрирует, как записать значение, находящееся в ячейке в строке 1, столбце 1 в точку дискретного входа/выхода C26.

```
Sub WriteDataBtn_Click()  
    Chanel = DDEInitiate ("Dsdata", "DemoTopic")  
    DdePoke Chanel, "C26", Cells(1, 1)  
    DdeTerminate Chanel  
End Sub
```



ПРИМЕЧАНИЕ: ПЛК DirectLogic не поддерживают форсирование входов (нельзя задать значение реальным входам).

Если дискретная точка, в которую вы хотите записать данные, реально используется **входным модулем**, информация, которая запишется через DSData Сервер, будет действительна только в течение одного цикла. Так произойдет потому, что ПЛК обновляет состояние всех входных модулей в начале каждого цикла. Это обновление перезапишет изменения, сделанные DSData Сервером.

Это также верно для выходных модулей и управляющих реле, за исключением тех случаев, когда программа ПЛК разрешает выполнить обновление. Если по программе точка должна быть деактивизирована, она будет деактивизирована, и перезапишет любую информацию, переданную через DSData Сервер.

Использование псевдонимов и обозначений



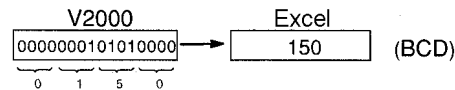
Использование псевдонимов

ПРИМЕЧАНИЕ: Для использования псевдонимов, при создании Темы необходимо подсоединить к ней файл описания элементов **DirectSoft32**. Если вам нужна дополнительная информация, обратитесь к главе 2.

Мы уже упоминали, что вы можете использовать в DSData Сервере псевдонимы, созданные в **DirectSoft32**. Это обычно проще, чем точно запоминать адреса, где хранятся данные. В следующем примере, адресу памяти V2000, с помощью **DirectSoft32**, присвоен псевдоним "DelayTime". В момент создания тега вы просто заменяете адрес псевдонимом. Не забудьте, что *вы все равно должны использовать соответствующий модификатор*.

Пример использования псевдонима

=Dsdata|DemoTopic!"DelayTime:B"



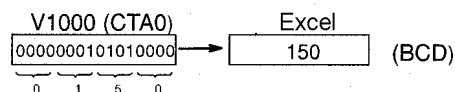
Использование обозначений

Примечание: Некоторое программное обеспечение не поддерживает пробелы и разделители в псевдонимах. Например, если вы создали в **DirectSoft32** псевдоним "Delay Time", тогда вы не сможете использовать его в формуле Excel. Для того, чтобы использовать псевдоним в формулах, вы должны войти в **DirectSoft32** и изменить его на "DealyTime".

Если вы использовали программное обеспечение **DirectSoft32**, вы уже должны были использовать и «обозначения». Это предопределенное имя для некоторых адресов и определенных элементов процессора. Не все адреса и элементы данных имеют обозначение. Например, вы можете создать тег текущего значения счетчика CT0, используя обозначение СТА0. В примере мы заменили адрес обозначением СТА0. Напомним снова, что *вы по-прежнему должны использовать соответствующие модификаторы данных*.

Пример использования обозначения

=Dsdata|DemoTopic!"СТА0:B"



Список возможных обозначений

Список возможных обозначений для определенного типа процессора содержится в файле \Dirctsoft\Help\Esmain.hlp. Найдите файл, используя менеджер файлов (проводник), и дважды щелкните по нему. Вы увидите пункты: **105/205/350/405 Aliases** (обозначения для 105/205/350/405 процессоров) и **305 Aliases** (обозначения для 305 процессоров). Щелкните по соответствующему пункту для просмотра списка обозначений.

Модификаторы операции записи

В большинстве случаев, установки DSData Сервера по умолчанию сделаны для достижения наибольшей скорости работы. Однако в некоторых случаях будет полезно понять, как DSData Сервер фактически выполняет запросы на запись, и какие параметры этой операции вы можете поменять. Этот раздел описывает ситуации, когда используются дискретные (X, Y, C и т.д.) и аналоговые (V-память, данные регистров) операции записи. Когда DSData Сервер получает запрос на запись данных от вашего приложения, он может использовать один из следующих двух методов:

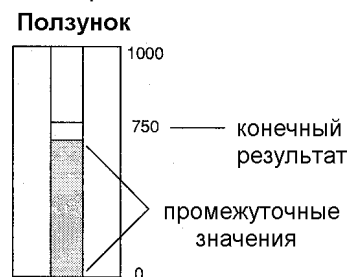
- **Объединение (Combine)** – объединяет несколько запросов на запись по одному адресу в один запрос.
- **Очередь (Queue)** – сохраняет отдельно любые запросы на запись по одному адресу, сохраняя их в полученном порядке.

Объединение операций записи (Combine Writes)

Что подразумевается под «Объединением операций записи»? Предположим, вы используете ползунок для изменения значения V-памяти от 0 до 1000. Это очень плавное управление аналоговым сигналом, т.к. ползунок может принимать любое значение от 0 до 1000. Предположим, вы перетаскиваете ползунок к значению 750. Что в этом случае будет делать DSData Сервер? Запишет ли он только последнее значение ползунка (750)? Или будет пытаться записать каждое значение, на всем протяжении пути ползунка, от 0 до 750?

В этом случае запишется конечный результат. Итак, по умолчанию, если возможно, DSData Сервер пытается комбинировать промежуточные значения. Некоторые промежуточные значения могут быть записаны в контроллеры, в зависимости от того, как быстро вы перемещаете ползунок, и как написана программа ползунка. Например, если вы медленно передвигаете ползунок, то больше значений запишется в ПЛК и меньше запросов будет объединено. Однако даже если вы перетаскиваете ползунок быстро, а DSData Сервер готов выполнить следующий запрос на запись, как только придет следующий запрос, он будет немедленно выполнен. Таким образом, DSData Сервер комбинирует запросы на запись только тогда, когда находится в режиме ожидания выполнения предыдущего запроса.

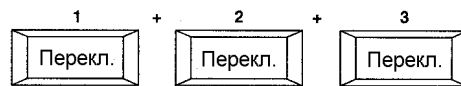
По умолчанию, DSData Сервер пытается объединить запросы на запись в одно слово памяти.



Очередь операций записи (Queue Writes)

Что означает «Очередь операций записи»? Предположим, что у вас имеется кнопка, которая активизирует реле, когда вы нажимаете и удерживаете кнопку, и деактивизирует, когда вы ее отпускаете. Что будет делать DSData Сервер, если вы нажмете и отпустите кнопку несколько раз подряд? В этом случае вам бы хотелось, чтобы все запросы были выполнены. Вам бы хотелось, чтобы точки ПЛК столько раз активизировались и деактивизировались, сколько раз вы нажимаете на кнопку. По умолчанию, DSData Сервер использует очередь запросов и обеспечивает заданную последовательность состояний точки.

По умолчанию, DSData Сервер пытается поставить в очередь несколько запросов на запись в одну и ту же дискретную точку.



Установки по умолчанию

Предыдущие примеры показывают, как важны концепции «объединение запросов» и «очередь операций записи». Действия при записи дискретных и аналоговых параметров различны. Это происходит потому, что установки по умолчанию зависят от типа используемых данных.

- **Дискретные точки** – запросы на запись дискретной точки устанавливаются в очередь. Это дает уверенность, что запросы активизации/деактивизации параметра будут произведены в заданной последовательности.
- **Слово памяти** – по умолчанию, запросы на запись в V-память или в регистры данных объединяются. Это помогает улучшить характеристики и достичь конечного значения параметра как можно быстрее.

Использование модификатора для изменения значения по умолчанию

Может так случиться, что вы не захотите использовать установленные по умолчанию управляющие запросами характеристики. Например, предположим, вы используете ползунок для изменения ячейки V-памяти, и вы хотите, чтобы он записывал все возможные значения по ходу движения. Вы, конечно, можете двигать ползунок медленнее, но это у вас отнимет много времени. Существует два модификатора, которыми вы можете указать DSData Серверу, хотите вы использовать очередь операций записи или хотите комбинировать подобные запросы.

- **:Q** – использовать очередь и индивидуально выполнять каждый запрос.
- **:C** – объединять запросы для одного адреса памяти.

Использование модификаторов данных с модификаторами операций записи

Модификаторы данных могут использоваться вместе с модификаторами, описанными ранее в этой главе. Вы не можете использовать вместе модификаторы “:Q” и “:C”. Если вы попытаетесь использовать оба модификатора, то будет использован последний модификатор в строке. Например, команда “:BQ” вполне допустима. Однако если вы введете “:BQC”, операции записи будут объединены, т.к. модификатор “C” стоит последним из двух.

Следующий пример показывает, как использовать модификатор “:Q”, для замены объединения данных, установленных по умолчанию совместно с модификатором “:B”, для перевода данных в формат BCD.

```
=Dsddata|Demotopic!V2000:BQ'
```


Наблюдение за коммуникационной связью

Когда вы создаете Тему, вы также должны создать коммуникационную связь с ПЛК. Если физическая линия не доступна, вы не сможете обмениваться данными с ПЛК. Потому необходимо, что бы вы имели возможность наблюдать и управлять физической связью из вашего программного обеспечения. Это очень полезная возможность, если вы используете DSData Сервер, как часть вашего пакета операторского интерфейса для управления оборудованием.

Например, предположим, вы имеете сеть из шести контроллеров. Каждый контроллер имеет свою коммуникационную связь и ассоциированную с ней Тему. Теперь, предположим, что пользователь забыл подать питание к двум контроллерам. Когда DSData Сервер запустится и не сможет установить общение с двумя ПЛК, их коммуникационная связь станет недоступна, а бит состояния деактивируется. Пользователь, возможно, не захочет сам искать неисправность, и было бы полезно встроить что-то в ваше программное обеспечение, что бы могло показывать и управлять состоянием коммуникационной связи. К счастью, такая возможность существует.

Использование бита состояния для наблюдения за связью

Каждая Тема, которую вы создаете, имеет специальный бит, называемый "Status" (Состояние). Вы можете использовать название бита как элемент при создании тэга. В этом случае данные не будут запрашиваться в ПЛК. Вместо этого DSData Сервер сообщит вам состояние связи, которая соединяет с ПЛК. Состояние определяется по нескольким параметрам. Вот они:

- Коммуникационная связь доступна?
- Есть ошибки?
- Связь приостановлена? (кто-то редактирует параметры связи?)

Состояние очень полезно как простой инструмент нахождения неисправностей, так как состояние дает ответ на все три вопроса. Следующий пример показывает, как использовать состояние в формуле.

```
=Dsdata|Demotopic!'Status'
```

Следующие значения возможны:

- 1 – все нормально
- 0 – есть проблемы

Так как состояние проверяет значения нескольких параметров, то возможен случай, когда связь доступна, а состояние возвращает значение 0, которое указывает на проблему. Итак, если у вас есть проблема, сначала убедитесь, доступна ли связь. Если связь доступна, убедитесь, что кабель подключен, и питание на контроллер подано. Это две основные причины огромного количества возможных проблем.

В демонстрационной программе для Excel мы покажем, как использовать специальную возможность VLOOKUP для вывода сообщения, вместо 0 или 1.

Встроенные системные элементы

Подобно элементу состояния (Status) Dsdata сервер также имеет следующие системные элементы, используемые для наблюдения за состоянием системы:

- **_StatusValue** Подробное описание состояния
 1 – Активен и без ошибок
 2 – Приостановлен
 3 – Ошибка
 4 – Заблокирован
- **_Xfers** Количество выполненных передач
- **_Retries** Количество попыток
- **_Error** Индикатор ошибки
- **_XferRate** Количество передач в секунду
- **Writes** Количество ожидающих операций записи
- **_UpdateInterval** Время между обновлениями с точностью 1/10 секунды

Topic/SubTo...	Item	Element	Value
<input type="checkbox"/> DemoTopic	<Link/PLC/DocDB>		450 KSeq/450
<input type="checkbox"/> DemoTopic	Status		Good(1)
<input type="checkbox"/> DemoTopic	Enable		Enabled(1)
<input type="checkbox"/> DemoTopic	_StatusValue		Active(1)
<input type="checkbox"/> DemoTopic	_Xfers		3683
<input type="checkbox"/> DemoTopic	_Retries		0
<input type="checkbox"/> DemoTopic	_Errors		0
<input type="checkbox"/> DemoTopic	_XferRate		10
<input type="checkbox"/> DemoTopic	_Writes		0
<input type="checkbox"/> DemoTopic	_UpdateInterval		0 (1/10 sec.)

Управление коммуникационной связью

Использование состояния доступности для управления связью

Когда запускается DSData Сервер, он опрашивает все коммуникационные связи, которые вы создали и проверяет, что параметры связи действительны. Он проверяет правильность конфигурации коммуникационных параметров, тип ПЛК и т.д. Если DSData Сервер сможет установить связь с описанным контроллером, то связь будет доступна, а если не сможет – недоступна. Вы не сможете использовать недоступную связь для общения с ПЛК.

Существует механизм, который позволяет перевести связь в доступное состояние. Вы можете для каждой темы, которую вы создали, создать теги для бита "Enable" (доступна). Посмотрите на пример тега для Excel:

```
=Dsddata|DemoTopic!'Enable'
```

Возвращаемое значение 1 говорит о том, что связь доступна и готова к использованию, 0 показывает, что связь недоступна. Существует много возможных применений, когда очень полезно попытаться вручную сделать недоступную связь доступной. Предположим, Вы имеете сеть из шести ПЛК, и заказчик забыл подать питание на два из них. Когда DSData Сервер запустится, он не сможет установить связь с этими ПЛК и сделает связь недоступной. Когда заказчик устранил проблему, он может попробовать вручную восстановить связь, записав единицу в бит "Enable".

Метод, который мы предлагаем использовать, используется в функции Auto редактора связей, когда вы только что начали создавать связь. Это функция проверяет, что связь с ПЛК установлена и все коммуникационные параметры верны.



Примечание: Если вы вручную сделаете связь доступной, это не означает, что коммуникационная связь станет работать. Этот процесс не проверяет коммуникационные параметры связи. Он просто устанавливает флажок, указывающий, что связь доступна. Поэтому мы рекомендуем дополнительно использовать бит "Status", для определения готовности связи к работе.

Следующий пример показывает макрос, привязанный к кнопке Excel. Когда вы нажимаете на кнопку, значение, находящееся в ячейке в строке 1, столбце 1 (единица или ноль) будет записано в бит "Enable" Темы DemoTopic.

```
Sub WriteDataBtn_Click()
    Chanel = DDEInitiate ("Dsddata", "DemoTopic")
    DdePoke Chanel, "Enable", Cells(1, 1)
    DdeTerminate Chanel
End Sub
```

Использование DSData Сервера с электронными таблицами

В этой главе...

- Вступление
 - Использование формул для чтения данных
 - Создание макросов
 - Пример Excel №1 – Формулы и простые макросы
 - Пример Excel №2 – макрос сбора данных
 - Использование Lotus 1-2-3
 - Использование Quattro Pro
 - Ответы на вопросы
-

Вступление



Примечание: Для того, чтобы правильно использовать информацию в этой главы, вы должны знать основы электронных таблиц. Если при использовании электронных таблиц вы себя чувствуете некомфортно, сначала ознакомьтесь с документацией к ним для более полного знакомства с использованием формул и макросов.

Два способа доступа к данным ПЛК

Большинство популярных электронных таблиц поддерживает DDE. Вы можете читать данные из ПЛК, или записывать их, используя DDE команды, встроенные в программное обеспечение. Существует два способа использования этих команд в электронных таблицах.

- Вставлять DDE команды в формулы ячеек (только для чтения)
- Вставлять DDE команды в макросы (для чтения и записи)

Формулы - это просто

При использовании формул для запроса данных вы просто должны ввести DDE команду в ячейку, также как и любую другую формулу электронной таблицы. Данные будут автоматически получены из ПЛК. Это самый легкий способ получения данных из ПЛК, но ваши возможности ограничены только чтением. Следующий пример использует простую формулу Excel для получения данных из ячейки V-памяти V2000.

```
=DSDATA|DemoTopic!V2000'
```

Макросы более гибки

Использование макросов для чтения и записи требует несколько больше усилий, но вы имеете большую гибкость. Используя макросы, вы можете как читать, так и писать данные в ПЛК. Вот пример макроса Microsoft Excel для записи значения ячейки в строке 1, столбце 1 в V-память ПЛК V2000.

```
Sub WriteDataBtn_Click()
    Chanel = DDEInitiate ("DSDATA", "DemoTopic")
    DdePoke Chanel, "V2000", Cells(1, 1)
    DdeTerminate Chanel
End Sub
```

Синтаксис доступных команд зависит от программного обеспечения, а здесь мы дадим краткое описание основных типов команд. Для решения типовых задач все DDE приложения содержат следующие команды.

- **Команда открытия канала** – прежде чем вы получите доступ к Теме, необходимо *открыть* или *инициализировать* коммуникационный канал к ней.
- **Команда чтения** – эта команда сообщает DSDATA Серверу, какой элемент надо запросить и где данные сохранить.
- **Команда записи** – эта команда сообщает DSDATA Серверу данные для записи и адрес назначения.
- **Команда закрытия канала** – когда заканчиваете DDE Сессию, вы должны уничтожить канал.

Мы включили демонстрационную программу

На следующих страницах мы даем краткое описание этих концепций и необходимый синтаксис для написания команд в некоторых популярных электронных таблицах. Если вы хотите получить более подробную информацию, обратитесь к демонстрационной программе Excel, которую мы записали на диск DSDATA Сервера. Демонстрационный файл может помочь вам понять, как базовые понятия, описанные в этой главе, используются в электронных таблицах.

Использование формул для чтения данных

Как мы говорили ранее, самый простой способ получения данных из ПЛК, это ввод формул в ячейку электронной таблицы. Формула ячейки является «горячей линией», т.е. начинает запрашивать данные, как только будет введена. Интервал обновления данных будет равен интервалу обновления, заданному для Темы.

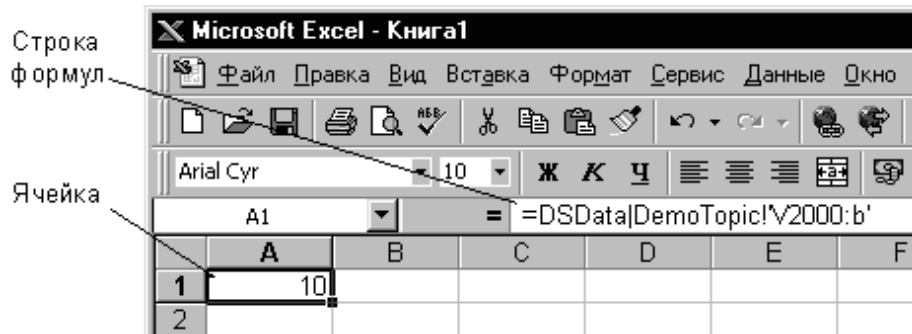
Прежде чем вводить формулу, вы должны создать Тему, которая будет указывать на связь с ПЛК. Если вам для выполнения этой задачи необходима дополнительная информация, обратитесь к главе 2.



Примечание: Несколько следующих примеров формул будут работать только с DLI05, DL205, DL350, DL405 и DL405 совместимыми процессорами. DL330, DL330P, DL340 и DL305 совместимые процессоры имеют другие типы данных и размеры регистров. В любом случае, вы должны свериться с руководством пользователя вашей модели ПЛК для определения типов памяти и соответствующих диапазонов адресов.

Ввод формулы

Для ввода формулы просто щелкните по ячейке, в которой вы хотите получить данные. Вы можете набирать формулу прямо в ячейке или использовать для ввода формулы командную строку. Если DDE канал (Тема и Связь) функционирует, ячейка начнет обновлять свое содержимое значениями из ПЛК. Это все, что вам надо было сделать.



Примеры формул

Далее мы покажем несколько простых примеров использования формул Excel для чтения различных типов данных. Обратите внимание, что базовая структура одинаковая. Единственная часть, которая меняется, это элемент (и модификатор) данных, которые вы хотите прочитать

=DsdData DemoTopic!'X0'	Ввод X0
=DsdData DemoTopic!'V1007:B'	Текущее значение счетчика CT7 (BCD)
=DsdData DemoTopic!'V2000:R'	Регистр V2000 (реальное)
=DsdData DemoTopic!'V40600'	Состояние управляющего реле C0-C17
=DsdData DemoTopic!'IO016'	Вход / выход 16 (DL305)
=DsdData DemoTopic!'R400:W'	Регистры R400 и R401
=DsdData DemoTopic!'TCA600'	Текущее значение таймера / счетчика 600 (DL305)
=DsdData DemoTopic!'T610'	Состояние таймера / счетчика 600 (DL305)

Создание макросов

Макросы дают больше возможностей и большую гибкость, по сравнению с формулами. Вы можете использовать макрос как для чтения, так и для записи данных. Некоторые пакеты позволяют присваивать макрос комбинации клавиш, другие создавать элементы управления, например кнопки, и привязывать к ним выполнение макроса. Использование макросов требует более глубокого понимания электронных таблиц. Если вы недостаточно хорошо знакомы с процедурой написания макросов, вам следует потратить некоторое время на изучение документации по электронным таблицам, прежде чем пытаться использовать макросы для DDE обмена.

Для иллюстрации примеров мы выбрали Microsoft Excel. Процедура создания макросов различна, в зависимости от пакета. Если вы имеете старую версию Excel, эта процедура, возможно, будет отличаться.

Следующий пример показывает, как использовать макрос для чтения данных из памяти ПЛК и отображение их в ячейке электронной таблицы.

Базовая
структура
макроса чтения

```
Sub Button1_Click()
    Chanel = DDEInitiate ("Dpdata", "DemoTopic")
    DDERequest (Chanel, "V2000:B")
    DDETerminate Chanel
End Sub
```

Sub Button1_Click() – начало кода макроса.

Chanel = DDEInitiate ("Dpdata", "DemoTopic") – выполняет открытие коммуникационного канала. Параметры, передаваемые процедурой: Сервис ("Dpdata") и Тема ("DemoTopic").

DDERequest (Chanel, "V2000:B") – передает системе запрос на чтение данных, используя ранее открытый канал (Chanel), из V2000, перевода значения в BCD и сохранение его в ячейке (строка 1, столбец 1).

DDETerminate Chanel – сообщает системе, что мы закончили текущий запрос.

Следующий пример показывает, как использовать макрос для записи данных из ячейки электронной таблицы в память ПЛК.

Базовая
структура
макроса чтения

```
Sub Button1_Click()
    Chanel = DDEInitiate ("Dpdata", "DemoTopic")
    DDEPoke Chanel, "V2000:B", Cells(1, 1)
    DDETerminate Chanel
End Sub
```

Sub Button1_Click() – начало кода макроса.

Chanel = DDEInitiate ("Dpdata", "DemoTopic") – выполняет открытие коммуникационного канала. Параметры, передаваемые процедурой: Сервис ("Dpdata") и Тема ("DemoTopic").

DDEPoke Chanel, "V2000:B", Cells(1, 1) – передает системе запрос на запись данных, используя ранее открытый канал (Chanel), из ячейки (строка 1, столбец 1), перевода значения в BCD и сохранение его в V2000.

DDETerminate Chanel – сообщает системе, что мы закончили текущий запрос.



ПРИМЕЧАНИЕ: ПЛК DirectLogic не поддерживают форсирование входов.

Если дискретная точка, в которую вы хотите записать данные, реально используется **входным модулем**, информация, которая запишется через DSData Сервер, будет действительна только в течение одного цикла. Так произойдет потому, что ПЛК обновляет состояние всех входных модулей в начале каждого цикла. Это обновление перезапишет изменения, сделанные DSData Сервером.

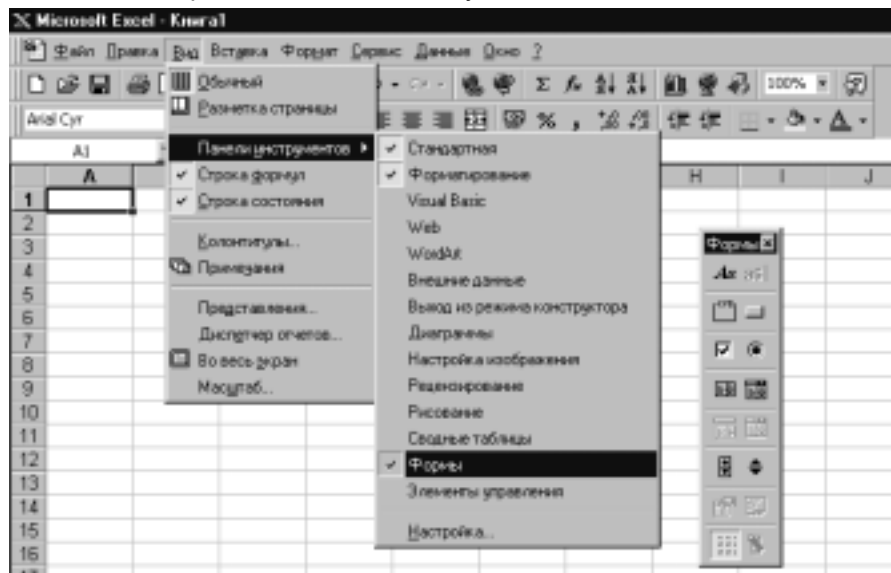
Это также верно для выходных модулей и управляющих реле, за исключением тех случаев, когда программа ПЛК разрешает выполнить обновление. Если по программе точка должна быть деактивизирована, она будет деактивизирована, и перезапишет любую информацию, переданную через DSData Сервер.

Быстрый пример

Понимание синтаксиса DDE команд поможет вам понять основные шаги по созданию макроса. Следующий пример дает краткое описание того, как создать кнопку с присоединенным к ней макросом. Когда пример будет готов, вы сможете получить данные простым нажатием кнопки.

Этот пример требует создать Тему "DemoTopic", присоединенную к Связи с DL105, DL205, DL350, DL405 или DL405 совместимому процессору.

1. Начните с открытие Excel и создания новой электронной таблицы. Если вы не видите в окне Excel панели инструментов **Forms** (Формы), используйте меню **View|ToolBars** (Вид|Панели инструментов) для вызова окна выбора панелей инструментов. Щелкните по пункту Forms (Формы). Когда вы это сделаете, появится панель инструмента, показанную на рисунке справа. Это вам позволит рисовать кнопки, ползунки и т.д.



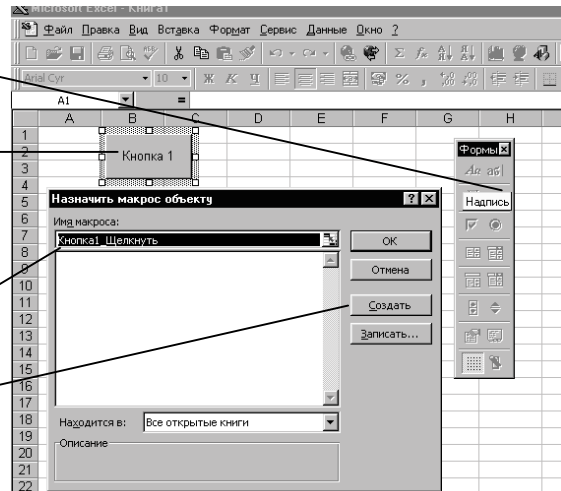
- Теперь нарисуем кнопку на листе Excel. Щелкните по инструменту 'CommandButton' (кнопка управления), и передвиньте курсор на поле, где вы хотите нарисовать кнопку. Чтобы нарисовать кнопку, нажмите левую клавишу мыши и, удерживая ее, переместите курсор по диагонали. Когда вы закончите определять размер кнопки, отпустите левую клавишу и появится запрос на присвоение кнопке макроса.

- Щелкните по кнопке управления.

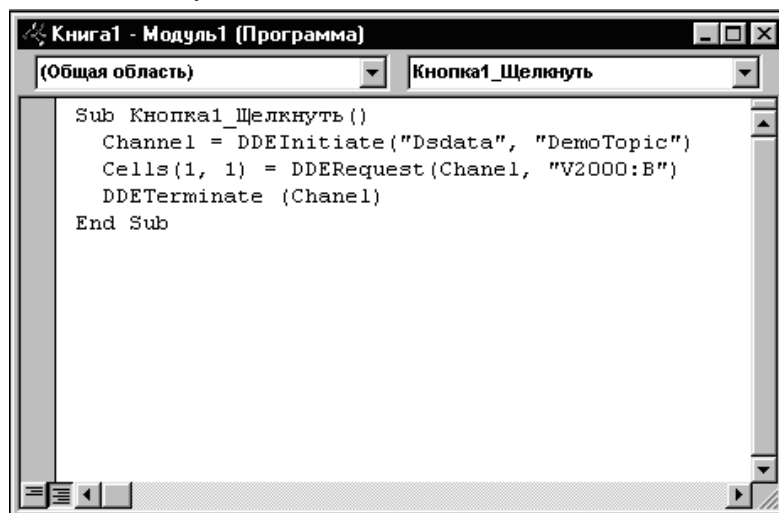
- Установите курсор на место, где вы хотите поместить кнопку.

- Диалог назначения макроса отобразит название макроса.

- Щелкните по кнопке New для создания нового макроса.

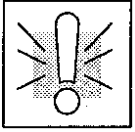


- Щелкните по кнопке 'New' (новый) на диалоге для создания нового макроса. Вызовется редактор Visual Basic, который позволяет вам вводить код программы для общения с DSDData Сервером. Наберите код программы, изображенной в примере. Когда вы закончите вводить команды, выберите **File|Save** (Файл|Сохранить) и задайте название своей работы. Щелкните по электронной таблице, чтобы увидеть снова кнопку.



- Щелкните по кнопке на листе Excel, чтобы запустить созданный макрос. Пример запрашивает данные из V2000, переводит из в BCD и помещает в ячейку (строка 1, столбец 1).

Пример Excel №1 – Формулы и простые макросы



ПРЕДУПРЕЖДЕНИЕ: Следующий пример использует ячейку V-памяти V2000. Если этот пример запустить, он запишет значение в данной ячейке. Прежде чем запустить пример, пожалуйста, убедитесь, что вы не используете эту ячейку для чего-либо важного.

**Демонстрация
№1
EXCEL5.XLS**

На дискете с DSDData Сервером находится простой пример использования Microsoft Excel. Этот пример использует основные концепции, описанные в ранее этой главе.

Чтобы запустить демонстрацию, запустите DSDData Сервер, и выполните следующие действия:

- Создайте Связь с ПЛК. Этот пример для DI105, DL205, DL350, DL405 и DL405 совместимых процессоров.
- Создайте Тему "DemoTopic", которая использует созданную Связь. Не подключайте документацию. Установите интервал обновления (Update Interval) в 0.

Для открытия демонстрационной таблицы запустите Excel и выберите меню File|Open (Файл|Открыть). Выберите директорий, в котором хранится демонстрационная программа. Если при инсталляции программы вы использовали директорий по умолчанию, то найдите C:\Dirctsf\DdeDemo. Выберите файл Excel5.xls и щелкните по ОК. Когда таблица откроется, Excel автоматически определит, что файл содержит ячейки, содержащие DDE теги. Выберите Yes, когда появится запрос, хотите ли вы установить связь с DSDData Сервером. Таблица продолжит загружаться и на экране появится следующее окно:

A	B	C	D	E	F
DDE topic status:					
1	DDE topic status:	1			
2	DDE topic status using lookup:	Active			
3	V2000 as word decimal:	0			
4	V2000 as word BCD	0			
5	V2000 as dword decimal:	0			
6	V2000 as dword BCD	0			
7	Bit 3 of V2000:	Off			
8	Value to write to V2000:	0	Write it!	<input checked="" type="checkbox"/> BCD	
9	Value to write to V2001:	0	Write it!	<input checked="" type="checkbox"/> BCD	

1 = V2000
2 = V2000 BCD

Что делает пример

Пример использует две ячейки памяти V2000 и V2001 для получения данных в несколько ячеек Excel. Этот пример использует несколько модификаторов данных, чтобы показать, как они действуют на значения.

Пример использует две клавиши для записи значений в те же ячейки памяти, и вы можете увидеть, как значения изменяются. Вы можете использовать флажки BCD, находящиеся справа от кнопок. Они позволяют посылать значения в ПЛК или в BCD формате, или в десятичном.

Использование примера

Ячейки в столбце В используют формулы для чтения данных из ПЛК. Они показывают данные в разных форматах, описанных ранее в этой главе. Для запуска процесса обновления выполните следующие действия:

1. Введите значение в ячейки В8 или В9, или в обе.
2. Пометьте флажок рядом с кнопкой **'Write it!'** (записать), если вы хотите записывать значение в формате BCD.
3. Щелкните по одной или по обоим кнопкам **'Write it!'**.

Когда вы щелкните по кнопке, данные, находящиеся в ячейках В8 или В9, будут записаны в ПЛК. Ячейки выше значений читают ту же информацию и показывают записанные данные. График отображает ту же информацию в графической форме.

Ячейка B1 содержит формулу, которая читает бит состояния (status) темы. Возвращаемое значение будет 1, если все нормально, или 0, если есть коммуникационные проблемы.

B1		=	=DSDData DemoTopic!status
	A		B
1	DDE topic status:		1
2	DDE topic status using lookup:		Active

Использование функции Excel VLOOKUP

Этот пример использует функцию VLOOKUP (в локализованной русской версии Excel эта функция называется ВПР) для связи числового значения в ячейке и текстовой строки в соседней ячейке. Мы используем эту функцию в нашей демонстрационной программе для того, чтобы отобразить состояние DDE связи в текстовом виде в ячейке B2. Давайте посмотрим, как это работает, и как мы это сделали. Щелкните на вкладке листа "String Table" (строковая таблица), чтобы увидеть созданную таблицу. Вы увидите следующее:

C3		=
	A	B
1	0 Off	Error!
2	1 On	Active

Как работает VLOOKUP

Запомните, ПЛК передает 1 или 0 в ячейку B1. Теперь посмотрите в окне Formula/Command, что мы написали в ячейке B2.

B2		=	=ВПР(DSDData DemoTopic!status;StringTable!A1:C2;3)
	A	B	C
1	DDE topic status:	1	
2	DDE topic status using lookup:	Active	

В этом примере строковая таблица находится в диапазоне A1:C2. Последний параметр функции строковой таблицы определяет, какой столбец использовать. Таблица берет значение состояния и использует его для выбора строки. Если значение бита состояния 1, функция возвращает слово 'Active' (активен), если значение 0, то функция возвращает 'Error' (ошибка).

Если вы измените последний параметр функции с 3 на 2, состояние DDE канала будет определяться словами 'On' и 'Off' (выключено или включено).

Наблюдение за изменением состояния в реальном времени

Чтобы увидеть изменения значения состояния, включите и выключите питание ПЛК. Вы увидите соответствующие изменения в электронной таблице.

Графическое представление данных

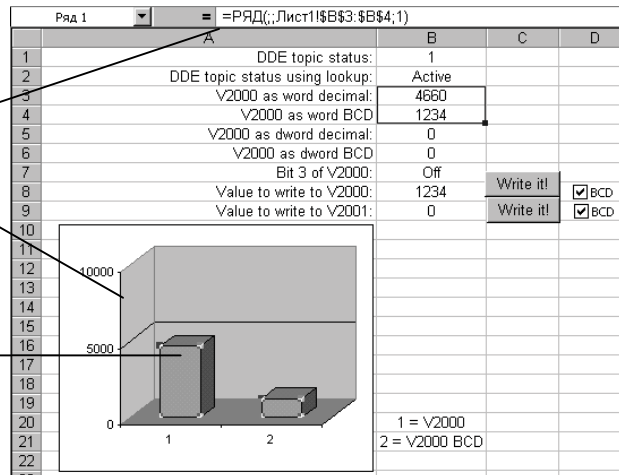
Если вы ввели новое значение и записали его в ПЛК, вы, возможно, заметили, что гистограмма изменяется каждый раз, когда вы записываете новые данные. Гистограммы очень полезны для разных применений. Они могут быть частью электронной таблицы реального времени. Давайте посмотрим на нашем примере, как такой график связан с данными в ПЛК.

Если вы дважды щелкните по графику, а потом щелкните по любому столбику, вы сможете увидеть формулу, которая берет данные ПЛК из ячейки В3 и В4 и связывает их с вертикальным размером столбика. Эта формула не содержит DDE команд. Это только команды Excel Basic, которые используются для построения графиков.

Эта формула определяет изменение графика данными из ячеек В3 и В4.

Для выбора графика дважды щелкните по нему в любом месте.

Далее щелкните по столбцу для отображения формулы, задающей высоту столбца.



Использование мастера графиков (Chart Wizard)

Если вы изучаете Excel одновременно с изучением DSDData Сервера, вы не знаете, как вводятся формулы и как использовать их для построения графиков. Excel содержит инструмент, называемый мастером диаграмм, который делает процесс построения графиков очень простым. Вы можете построить графики за шесть простых шагов. Справочная система Excel содержит подробное описание процесса построения графиков. Мы советуем всем начинающим пользователям воспользоваться справкой, выбрав в меню пункт Help|Serch, и найти описание по теме **Chart Wizard Button** (кнопка мастера диаграмм).

Пример Excel №2 – макрос сбора данных

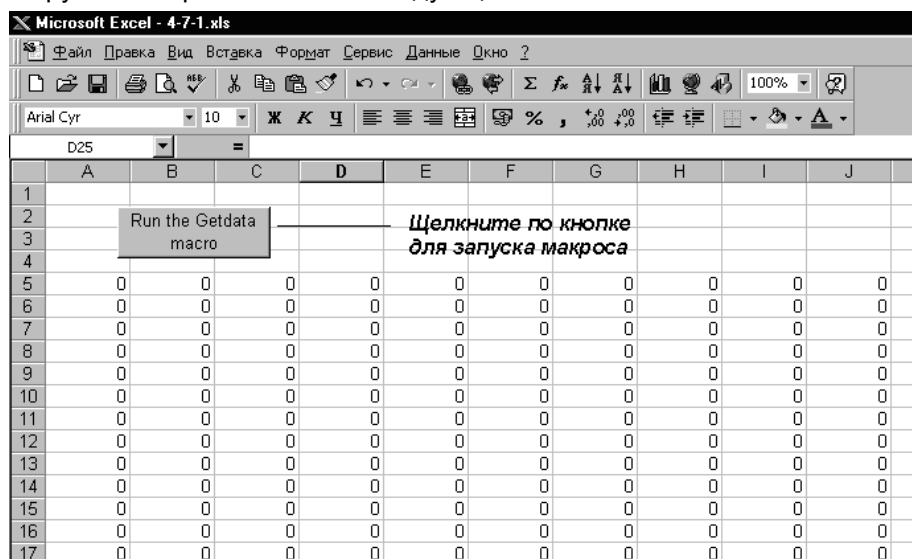
Демонстрация №2 Collect.XLS

Мы включили простой пример Microsoft Excel, демонстрирующий макрос, который архивирует данные из ПЛК, когда вы щелкните по кнопке. Демонстрация содержит идеи, описанные ранее в этой главе.

Чтобы запустить демонстрацию, запустите DSDData Сервер и выполните следующие действия:

- Создайте Связь с ПЛК. Этот пример для DI105, DL205, DL350, DL405 и DL405 совместимых процессоров.
- Создайте Тему "DemoTopic", которая использует созданную Связь. Не подключайте документацию. Установите интервал обновления (Update Interval) в 0.

Для открытия демонстрационной таблицы запустите Excel и выберите меню **File|Open** (Файл|Открыть). Выберите директорию, в котором хранится демонстрационная программа. Если при инсталляции программы вы использовали директорию по умолчанию, то найдите C:\Dirctstf\DdeDemo. Выберите файл Collect.xls и щелкните по **OK**. После загрузки на экране появится следующее окно:



Этот макрос не начнет выполняться сам по себе, этот процесс должен инициировать пользователь. Если вы щелкните по кнопке 'Run the GetData Macro', начнется выполнение кода с циклом For/Next, в котором опрашиваются данные.

Макрос будет опрашивать регистры от V2000 до V3777, и сохранять их в секции электронной таблицы размером и 16 строк на 16 столбцов. Данные размещаются в 5 строке сверху, чтобы оставить место для заголовков. Просто отредактируйте макрос для изменения любых настроечных параметров.

Использование Lotus 1-2-3

- Введение** Мы очень рекомендуем использовать Excel для связи с нашим DSData Сервером. Мы находим, что это наиболее гибкий и профессиональный инструмент для работы с ним. Однако мы знаем, что некоторые покупатели DSData Сервера захотят использовать его с Lotus 1-2-3. К сожалению, невозможно включить примеры использования DSData Сервера для всех электронных таблиц. Несмотря на это, мы все-таки включили простую демонстрационную программу **LOTUS5.WK4**.
- Подобные понятия** Так же, как Excel, Lotus 1-2-3 позволяет вам создавать связь между ПЛК и электронной таблицей двумя способами:
- Написать формулу в окно Formula/Command
 - Написать код макроса
- Формула** Для связывания данных в ПЛК с ячейкой используйте функцию @DDELINK. Вот пример формулы чтения ячейки V2000:
@DDELINK("DSDATA", "DemoTopic", "V2000")
Поместите эту формулу в окно Formula/Command, так же, как делали в Excel.
- Макрос** Lotus 1-2-3 позволяет использовать макрос. Мы используем кнопку, к которой привязан макрос, точно так же, как в нашем примере Excel. Руководство по Lotus 1-2-3 описывает, как создать элемент, называемый **SmartIcons**, находящийся в панели инструментов **SmartIcon palette** сверху электронной таблицы. Ниже показан код из нашего демонстрационного примера, который выводит на экран одну из двух фраз, в зависимости от того, какие установки электронной таблицы вы выберете:
- ```
POKEIT {DDE-OPEN "Dsdata"; "DemoTopic"}{ERROR}
 {DDE-POKE B1; "V2001:B"}
 {LET RESULT; Ding Ding Ding! You win!!!}
 {DDE-CLOSE}
ERROR {LET RESULT; Survey sez...buzzzzz! Sorry, you lose...}
 {DDE-CLOSE}
 {QUIT}
RESULT Ding ding ding ding! You win!!
```

### DDE команды Lotus 1-2-3

| Команда        | Пример синтаксиса                          | Применение                                                             |
|----------------|--------------------------------------------|------------------------------------------------------------------------|
| {DDE-OPEN}     | Channel = {DDE-OPEN "Dsdata"; "DemoTopic"} | Инициализирует связь, определяя Сервис и Тему                          |
| {DDE-POKE}     | {DDE-POKE B1; "V2001:B"}                   | Записывает данных по определенному адресу в ПЛК                        |
| {DDE-REQUEST}  | Data = {DDE-REQUEST B1; "V2000:B"}         | Читает данные из ПЛК                                                   |
| {DDE-CLOSE}    | {DDE-CLOSE}                                | Закрывает коммуникационный канал                                       |
| {LET variable} | {LET RESULT; You are winner!}              | Определяет переменную и показывает текст, который будет использоваться |
| {QUIT}         | {QUIT}                                     | Заканчивает текущий макрос и возвращает управление клавиатуре          |

## Использование Quattro Pro

- Введение** Мы настоятельно рекомендуем использовать Excel для связи с нашим DSData Сервером. Мы находим, что это наиболее гибкий и профессиональный инструмент для работы с ним. Однако мы знаем, что некоторые покупатели DSData Сервера захотят использовать его с Quattro Pro. К сожалению, невозможно включить примеры использования DSData Сервера для всех электронных таблиц. Несмотря на это, мы все-таки включили простую демонстрационную программу **QUATTRO5.WB1**.
- Подобные понятия** Так же, как Excel, Quattro Pro позволяет вам создавать связь между ПЛК и электронной таблицей двумя способами:
- Написать формулу в окно Formula/Command
  - Написать код макроса
- Формула** Для связывания данных в ПЛК с ячейкой используя формулу, используйте функцию @DDELINK. Например, вот пример формулы чтения ячейки V2000:
- ```
@DDELINK([DSDATA|DemoTopic], "V2000:B")
```
- Поместите эту формулу в окно Formula/Command, так же, как делали в Excel.
- Макрос** Так же, как и Excel, Quattro Pro позволяет использовать макрос. Посмотрите, как мы используем макросы в Excel для решения различных задач. Вы также заметите, что мы используем кнопку, к которой привязан макрос, точно так же, как в нашем примере Excel. Руководство по Quattro Pro описывает, как использовать инструмент **SpeedButton** для создания такой кнопки. Ниже приведен пример кода макроса:
- ```
POKEIT {INITIATE "Dpdata", "DemoTopic", D1}
 {POKE D1; "V2001:B", B1}
 {TERMINATE D1}
```
- Посмотрите руководство** **В** Посмотрите дополнительные макрокоманды в руководстве по Quattro Pro. Мы включили таблицу, где описываются все команды, которые мы используем. Quattro Pro не обновляет данные. Если вы хотите сделать что-нибудь с данными в реальном времени, используйте Excel. Он очень хорошо выполняет задачи с данными в реальном времени и имеет более гибкий макроязык.

### DDE команды Quattro Pro

| Команда     | Пример синтаксиса                       | Применение                                      |
|-------------|-----------------------------------------|-------------------------------------------------|
| {INITIATE}  | {INITIATE "Dpdata", "DemoTopic", D1}    | Инициализирует связь, определяя Сервис и Тему   |
| {POKE}      | {POKE D1; "V2001:B", B1}                | Записывает данные по определенному адресу в ПЛК |
| {REQUEST}   | Data = {REQUEST Channel, "V2000:B", B1} | Читает данных из ПЛК                            |
| {TERMINATE} | {TERMINATE D1}                          | Закрывает коммуникационный канал                |



## Ответы на вопросы

### В ячейке, после ввода формулы, высветилось #Name?

Допущена синтаксическая ошибка в определении тега в секции элемент, т.е., например, вы написали G2000 вместо V2000.

Вы ссылаетесь на тип данных, не поддерживаемый характеристиками ПЛК, т.е. пытаетесь прочитать R400 из DL205 ПЛК.

### В ячейке, после ввода формулы, высветилось #VALUE!

Если для выполнения преобразования над данными вы используете модификатор данных, вы должны и элемент, и модификатор поместить вместе в одинарные кавычки, т.е. 'V2000:b'.

### В ячейке, после ввода формулы, высветилось #N/A

Синтаксис формулы верен, но Excel не получил данные от DSDData Сервера. Если вы имеете много тегов на одной электронной таблице, может потребоваться несколько минут для получения всех данных. Excel помещает #N/A во все ячейки, ожидающие обновления.

### В ячейке, после ввода формулы, высветилось #REF!

В настоящий момент DSDData Сервер не запущен. Вам необходимо запускать DSDData Сервер перед открытием электронной таблицы. Выбор Yes на запрос не решает проблему. Вам необходимо закрыть Excel, запустить DSDData Сервер и перезапустить Excel.

### Появилось сообщение: "Remote data not accessible. Start application 'DSDATA.EXE?'" (Удаленные данные не доступны. Запустить приложение DSDATA?)

В настоящий момент DSDData Сервер не запущен. Вам необходимо запускать DSDData Сервер перед открытием электронной таблицы. Выбор Yes на запрос не решает проблему. Вам необходимо закрыть Excel, запустить DSDData Сервер и перезапустить Excel.

Допущена ошибка при определении имени Сервиса. Имя Сервиса не чувствительно к регистру, однако, должно быть грамматически верным. Имя Сервиса для нашего DSDData Сервера всегда "Dsdata".

Допущена ошибка при определении имени Темы. Имя Темы не чувствительно к регистру, однако, должно быть точно таким же, как появляется в экране конфигурации Сервиса. Одна из наиболее распространенных ошибок – использование пробелов в имени Темы. Пробелы, в принципе, допустимы, но вы должны имя Темы поместить в одинарные кавычки, т.е. 'Demo Topic'.

Выполняемый файл DSDATA.EXE не найден в каталогах, определенных переменной окружения PATH. Допишите каталог BIN DSDData Сервера в определение переменной PATH в файле autoexec.bat, например, PATH=C:\Dirctsf\bin;C:\;C:\Dos;C:\Windows.

В этой главе...

- Вступление
  - Использование свойств объекта для чтения данных
  - Использование подпрограмм для чтения и записи данных
  - Демонстрационная программа
  - Ответы на вопросы
-

## Вступление

### Два способа доступа к данным



### Окно свойств элемента использовать просто!

Если вы прочитали главу по использованию Microsoft Excel, вы узнали два способа доступа к данным ПЛК через DSData Сервер. Существует формулы для непрерывного чтения данных и макросы, позволяющие читать и записывать в ответ на действия пользователя. Visual Basic предоставляет вам доступ к данным двумя очень похожими способами: свойства объекта и процедура.

**Примечание:** Примеры и демонстрационные программы, используемые в этой главе, разработаны с использованием **Visual Basic 6.0 Professional Edition**. Если вы не используете эту версию Visual Basic, не огорчайтесь, вы можете использовать эту главу как руководство. Используемые методы и, в частности, синтаксис может отличаться, но, в общем, процесс должен быть очень похож.

Вводя DDE команды в окно свойств объектов Visual Basic, вы используете простой способ непрерывного обновления данных ПЛК. Эти объекты, такие как элемент редактирования, кнопка и т.д. позволяют вам читать данные из приложений, поддерживающих DDE. Подобно формулам в Excel, этот метод не позволяет вам записывать данные через DSData Сервер, а дает возможность только читать их.

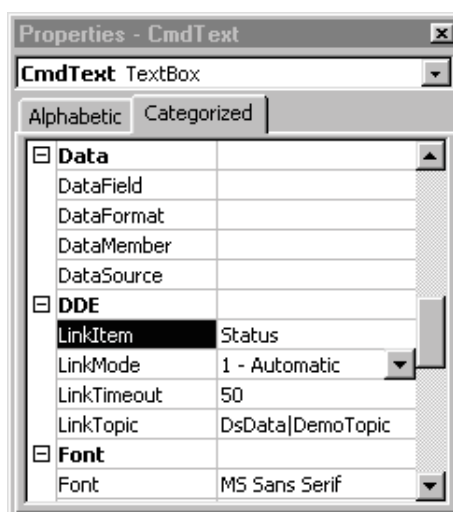
Следующий рисунок показывает часть окна свойств для метки. Как показано ниже, вы должны определить три пункта, когда настраиваете любые компоненты Visual Basic для использования DSData Сервера.

**LinkItem** (элемент связи) – это может быть любой доступный элемент ПЛК вместе с любыми необходимыми модификаторами.

**LinkMode** (режим связи) – Вы можете выбрать четыре опции:

- 0 – **None**, используется для прекращения связи с ПЛК
- 1 – **Automatic**, используется, если вы хотите использовать непрерывное обновление данных
- 3 – **Manual**, используется в подпрограммах
- 4 – **Notify**, работает так же, как Automatic.

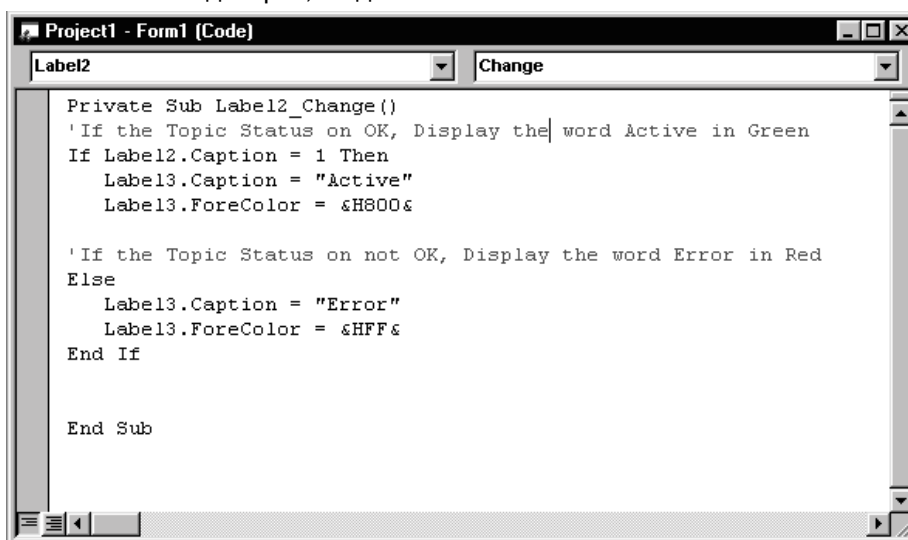
**LinkTopic** (Тема связи) – этот пункт состоит из двух частей: название Сервиса (в нашем случае это всегда Dsdata) и название Темы, к которой вы хотите получить доступ. Обе части разделяются символом трубы (две вертикальные черточки друг над другом).



### Подпрограммы дают гибкость

Вторым методом использования DDE команд является подпрограмма. Она включает в себя код Visual Basic, который дает вам мощь и гибкость, но требует некоторого опыта программирования. Эти подпрограммы могут быть запущены в ответ на действия пользователя или по событию, произошедшему при выполнении приложения Visual Basic.

Вы можете вызвать редактор кода процедур двумя способами: дважды щелкните на объекте Visual Basic, таком как элемент редактирования, кнопка и т.д., или выберите объект единственным щелчком, затем выберите пункт меню View | Code. Следующее окно показывает код, написанный для изменения свойств метки под именем Label2. Эта процедура, обычно называемая процедурой изменения данных, вызывается каждый раз, когда значение метки изменяется.



```
Project1 - Form1 (Code)
Label2 Change
Private Sub Label2_Change()
'If the Topic Status on OK, Display the word Active in Green
If Label2.Caption = 1 Then
 Label3.Caption = "Active"
 Label3.ForeColor = &H800&

'If the Topic Status on not OK, Display the word Error in Red
Else
 Label3.Caption = "Error"
 Label3.ForeColor = &HFF&
End If

End Sub
```

### Мы включили демонстрационную программу

На следующих страницах мы дадим вам описание концепций и необходимый синтаксис. Если вы хотите узнать больше, мы рекомендуем поработать с примерами, использующими **Visual Basic 6.0 Professional** и находящимися на дискетах с DSDData Сервером. Если вы умеете работать с Visual Basic, демонстрационный файл поможет вам понять, как основные идеи, описанные в этой главе, реализуются внутри программы.

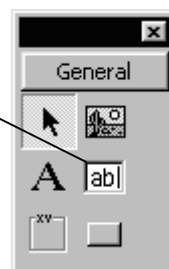
## Использование свойств объекта для чтения данных

### Простой пример чтения V-памяти ПЛК

В Visual Basic чтение данных из ПЛК относительно простая задача. Следующий пример поведет Вас шаг за шагом к созданию текста, который будет показывать текущее значение V-памяти ПЛК V2000. Текст будет автоматически обновляться каждый раз, когда данные в ПЛК будут меняться. Чтобы сделать этот пример рабочим, надо иметь работающую связь с ПЛК, т.е. правильно подключенный кабель и поданное питание в ПЛК.

1. Запустите DSData Сервер и создайте Связь с ПЛК, которую Вы собираетесь использовать. Если вы хотите вспомнить, как это делается, обратитесь снова к главе 2. Затем создайте тему, назовите ее DdeTopic и установите интервал обновления 0.
2. Откройте Visual Basic и создайте новый проект. Если панель инструментов не видна, вызовите ее через меню **View|Toolbox**.
3. Сначала надо создать объект, который поддерживает встроенные DDE функции. Visual Basic имеет два элемента, из которых вы можете выбрать: текстовое поле и метка. Добавим на форму, например, текстовое поле. Щелкните по иконке текстового поля на панели инструментов, чтобы выбрать тип объекта, который Вы хотите использовать.

Щелкните по иконке текстового поля на панели инструментов, чтобы выбрать тип объекта, который хотите использовать.

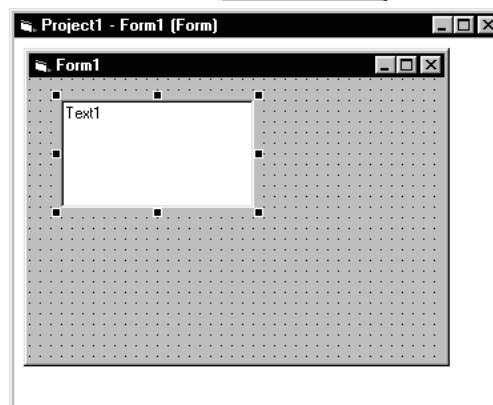


Установите курсор на место, где бы Вы хотели расположить верхний левый угол текстового поля.

Нажмите и удерживайте правую клавишу мыши, пока ведете мышь вправо и вниз. Размер текстового поля будет увеличиваться.

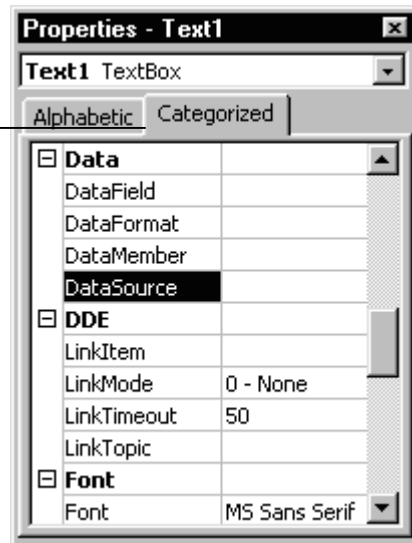
Отпустите клавишу мыши, когда текстовое поле достигнет нужного размера.

Когда вы отпустите клавишу мыши, вы увидите форму, которая будет выглядеть как эта.



4. Теперь вам надо заполнить соответствующие поля данных для того, чтобы связать текстовое поле и V-память ПЛК V2000.

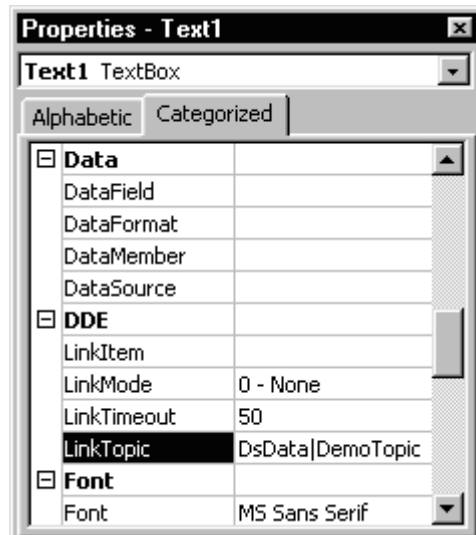
Щелкните по вкладке Categorized (категории) для группировки полей данных.



5. Введите значения в поля LinkTopic, LinkItem и, наконец, LinkMode. Значения должны быть введены именно в таком порядке. Смысл в том, что Visual Basic начнет попытки обновлять объект сразу же, как вы измените значение LinkMode с **0-None** на **1-Automatic**. Если вы не сделаете правильную установку всех этих полей, вы будете постоянно получать сообщение об ошибке. Вы можете запросто исправить это, сменив LinkMode обратно. Давайте остановимся на каждом из этих полей более подробно.

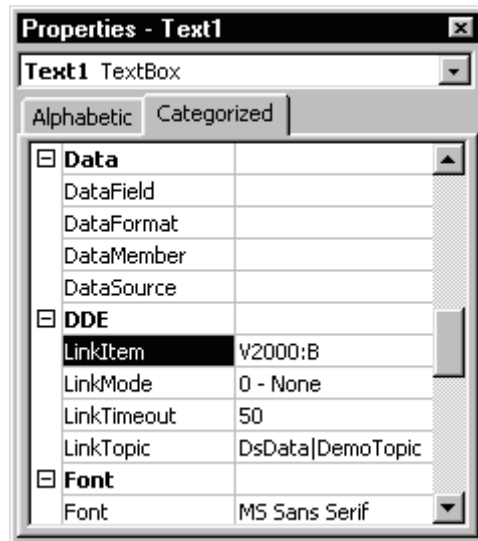
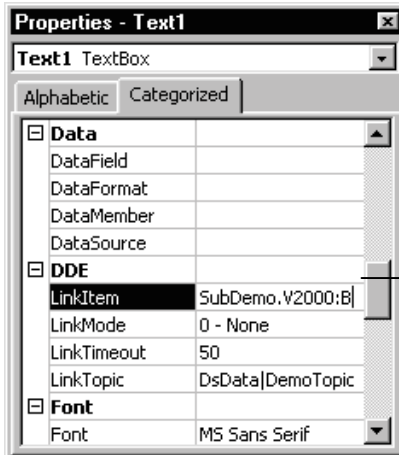
Поле **LinkTopic** содержит и Сервис, и Тему, разделенные символом трубы. Сервис, в нашем случае, всегда Dsdata. Тема та же, что вы создали на шаге 1.

Для нашего примера введите Dsdata|Demotopic.



Поле **LinkItem** содержит адрес ячейки памяти ПЛК со всеми необходимыми модификаторами данных. (Смотрите в главе 3 описание возможных модификаторов данных).

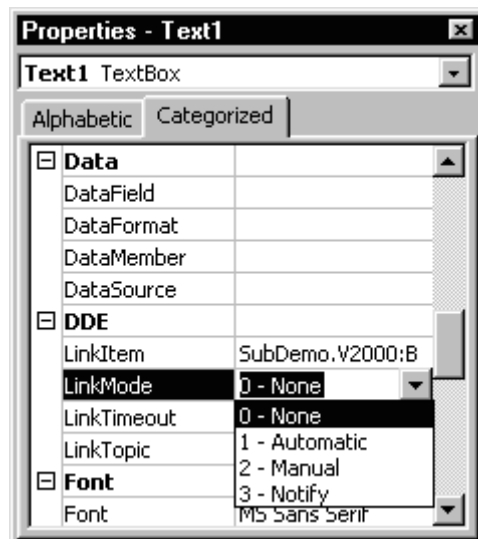
Для этого примера введите V2000:B – чтение из ячейки памяти V2000 и перевод в BCD.



Если вы используете подтемы, следуйте следующему синтаксису.

Поле **LinkMode** может содержать одно из четырех следующих значений, которые будут определять, как управляется обновление данных.

- 0 – **None**, используется для прекращения связи с ПЛК
- 1 – **Automatic**, используется, если вы хотите непрерывное обновление данных
- 3 – **Manual**, используется в подпрограммах
- 4 – **Notify**, работает так же, как Automatic.



Для изменения значения щелкните на поле ввода данных, и справа на поле появится стрелка вниз. Щелкните по стрелке, и появится список всех возможных опций.

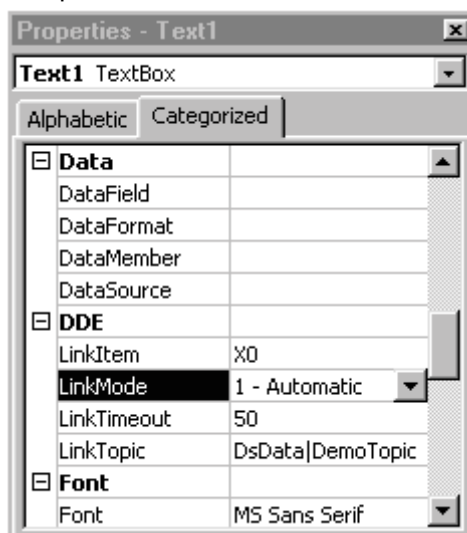
Для нашего примера, щелкните по **1- Automatic**. Как только вы это сделаете, начнется обновление данных текстового поля. Оно будет содержать данные, запрошенные из V-памяти ПЛК.

**Чтение значений дискретных входов/выходов**

Предыдущий пример читает V-память ПЛК и сохраняет его в текстовом поле. Процесс чтения дискретных входов/выходов, по существу, точно такой же. LinkItem содержит желаемую точку данных, например, X0, Y1, C16, GX100 и т.д.

Значение, получаемое из ПЛК, определяющее состояние дискретного входа/выхода, может быть "0", если точка деактивизирована - OFF или "1", если активизирована - ON. Посмотрите, как в следующем примере мы запрашиваем состояние точки дискретного входа X0.

Поле **LinkItem** содержит дискретную точку вход/выход.

**Это именно то, что просто**

Это все, что использует окно свойств для создания связи между текстовым полем или меткой и V-памятью ПЛК, или дискретной точкой. Этот метод дает простой путь к получению данных из ПЛК и отображению их на экране компьютера. Однако этот метод ограничен только чтением данных. Если Вы хотите записывать значения обратно в ПЛК или, если хотите ощутить реальную мощь Visual Basic и DDE, необходимо использовать подпрограммы. В следующем разделе описан шаг за шагом процесс создания подпрограмм, которые будут записывать значения в память ПЛК V2000.



## Использование подпрограмм для чтения и записи данных

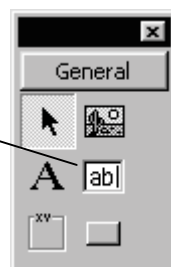
Процесс записи значений данных в ПЛК через DSData Сервер может быть выполнен подпрограммой. Подпрограмма - это программа Visual Basic, которая получает управление в ответ на определенное событие, которое происходит в момент выполнения вашего приложения. Это событие может быть действием пользователя - таким, как щелчок мыши или ввод с клавиатуры. Эти события могут также генерироваться самим Visual Basic приложением, например, системным таймером, закончившим отсчет времени.

### Простой пример записи значения

Следующий пример добавляет кнопку к созданной ранее форме. Каждый раз, когда вы нажимаете эту кнопку, значение в памяти ПЛК V2000 будет увеличиваться. Это происходит путем чтения значения V2000 из ПЛК, добавления 1 и записи значения обратно в ПЛК. Новое значение автоматически появится в текстовом поле, созданном нами ранее. Давайте начнем делать!

1. Сначала нам необходимо создать второе текстовое поле. Нам необходим второй "тег", который мы сможем использовать для временного доступа к V-памяти, которую мы хотим изменить.

Щелкните по иконке текстового поля для выбора типа создаваемых объектов.

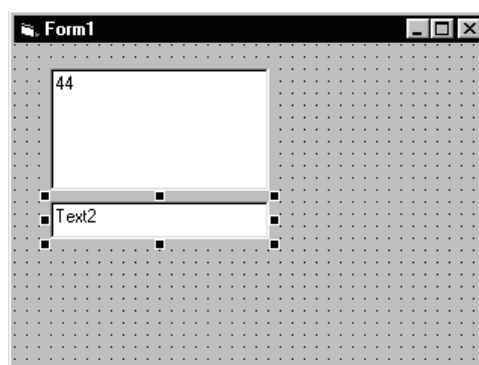


Установите курсор на место, где бы вы хотели расположить верхний левый угол текстового поля.

Нажмите и удерживайте правую клавишу мыши, пока ведете мышью вправо и вниз. Размер текстового поля будет увеличиваться.

Отпустите клавишу мыши, когда текстовое поле достигнет нужного размера.

Когда вы отпустите клавишу мыши, вы увидите форму, которая будет выглядеть как эта.



Щелкните по вкладке Categorized (категории) для группировки полей данных.

**Чтобы сделать это текстовое поле невидимым**, когда приложение запущено, измените свойство Visible (видимо) на False (ложь).

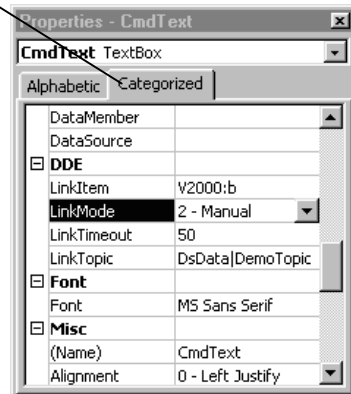
**Введите следующие значения** для DDE полей:

LinkTopic Dsdata|DemoTopic

LinkItem V2000:b

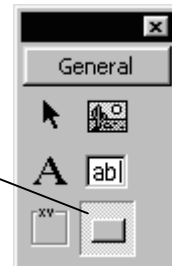
LinkMode 2-Manual

**Измените имя текстового поля на CmdText**, щелкнув по полю Name (имя), удалив его содержимое (Text2) и напечатав новое имя.



- Теперь Вам надо создать кнопку, которая будет выполнять подпрограмму, когда Вы щелкаете мышкой по кнопке.

Щелкните по кнопке для выбора типа создаваемых объектов.



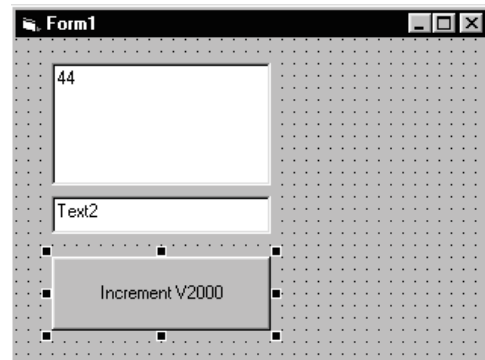
Установите курсор на место, где бы Вы хотели расположить верхний левый угол текстового поля.

Нажмите и удерживайте правую клавишу мыши, пока ведете мышь вправо и вниз. Размер текстового поля будет увеличиваться.

Отпустите клавишу мыши, когда текстовое поле достигнет нужного размера.

Для изменения текста, отображенного в текстовом поле, введите в поле Caption (шапка) "Increment V2000"

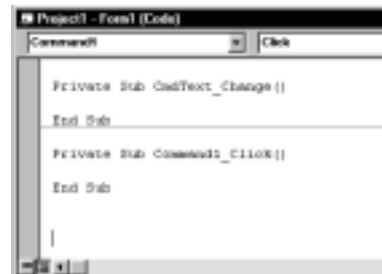
Когда вы отпустите клавишу мыши, вы увидите форму, которая будет выглядеть как эта.



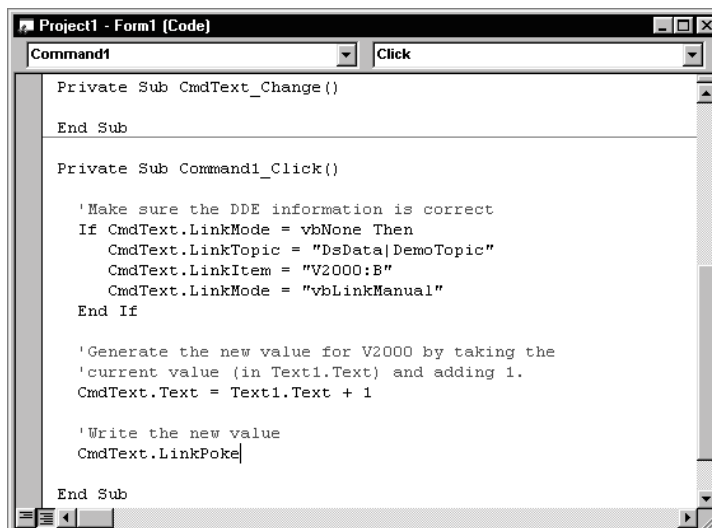
3. Теперь надо создать подпрограмму, которая будет выполняться, когда Вы щелкните по созданной кнопке. Когда Вы создаете кнопку, Visual Basic создает несколько пустых подпрограмм, по одной на каждый тип событий, на которые кнопка может реагировать. В нашем примере, мы должны ввести код в подпрограмму 'Command1\_Click()'.

Щелкните по кнопке правой кнопкой мыши и выберите View Code в выпадающем списке.

Появится подпрограмма, которую создал Visual Basic, когда Вы сделали кнопку.



4. Появившееся окно дает вам возможность вводить код подпрограммы Visual Basic, которую вы хотите выполнять. Введите код, который вы видите здесь. Когда вы закончите вводить код, для закрытия диалога и запоминания кода щелкните по 'X' в правом верхнем углу окна.



5. Вот и вся работа, которую вам необходимо выполнить для этого примера. Сейчас вам надо запустить приложение, чтобы увидеть увеличение ячейки памяти V2000. Щелкните мышкой по кнопке Start на панели управления.

Щелкните по кнопке Start (Пуск) для запуска приложения.

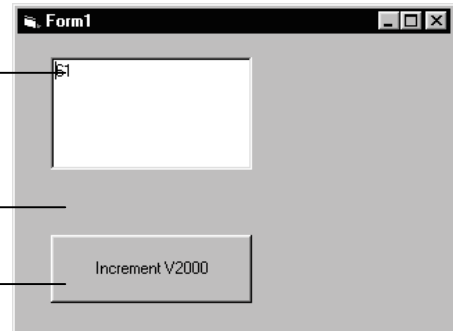


6. Каждый раз, когда вы щелкните по кнопке, ячейка памяти ПЛК V2000 увеличится на 1. Когда вы захотите остановить выполнение приложения, щелкните 'X' в правом верхнем углу, чтобы закрыть эту форму и вернуться в окно разработки.

При каждом нажатии кнопки значение обновляется здесь.

Заметьте, что текстовое поле с именем CmdText при выполнении приложения невидимо.

Щелкните для увеличения V2000.



**Запись значения дискретных входов/выходов**

Предыдущий пример записывал значение в ячейку V-памяти ПЛК. Процесс записи значений дискретных входов/выходов по существу такой же.

Если Вы используете кнопку, такую же, как в прошлом примере, вот шаги, которые надо сделать для записи дискретного значения:

1. Установите в поле LinkItem название точки дискретного входа/выхода, которую вы хотите записать, например X0, Y1, C16, GX100 и т.д.
2. Установите свойство Text для текстового поля в 0 для выключения или 1 для включения дискретной точки.
3. Выполните команду LinkPoke для текстового поля для записи значения в ПЛК.



---

**ПРИМЕЧАНИЕ:** ПЛК **DirectLOGIC** не поддерживают форсирование входов.

---

Если дискретная точка, в которую Вы хотите записать данные, реально используется **входным модулем**, информация, которая запишется через DSData Сервер, будет действительна только в течение одного цикла. Так произойдет потому, что ПЛК обновляет состояние всех входных модулей в начале каждого цикла. Это обновление перезапишет изменения, сделанные DSData Сервером.

Это также верно для выходных модулей и управляющих реле, за исключением тех случаев, когда программа ПЛК разрешает выполнить обновление. Если по программе точка должна быть деактивизирована, она будет деактивизирована, и перезапишет любую информацию, переданную через DSData Сервер.

**Это действительно просто!**

Этот пример демонстрирует базовую процедуру для создания тегов, которые получают доступ к данным ПЛК. Не имеет значения, какой тип данных вы запрашиваете - V-память или дискретные точки, процесс будет таким же, как мы только что описали.

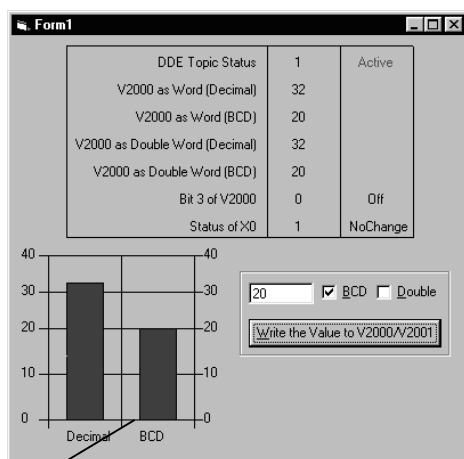
Мы приоткрыли завесу над реальной мощью Visual Basic. Насколько привлекательна та простота, с которой профессионально смотрящееся приложение может быть создано. Мы включили маленькую демонстрационную программу на ваш инсталляционный диск. Она показывает несколько дополнительных способов, которые могут быть полезны вашему приложению.

## Демонстрационная программа

Демонстрационная программа создана с помощью Visual Basic 6.0 Professional. Это приложение использует элемент управления MSChart, поэтому если вы не имеете этой версии Visual Basic, вы, возможно, не сможете открыть и редактировать это приложение.

Мы включили скомпилированное приложение VB6Demo.Exe. Не включена требуемая для этого приложения библиотека MSVbVm60.dll. Эти два файла являются частью Visual Basic и не могут передаваться.

Мы также включили сам проект, который включает файлы VB6Demo.Vbp, VB6Demo.frm и VB6Demo.frx. Это позволит вам редактировать приложение и делать в нем все изменения, которые вы пожелаете.



Фраза 'No change' (без изменений) означает принцип DDE, когда изменения параметров только обновляют окно приложения. Это поле может принимать значения 0 или 1, когда состояние входа/выхода ПЛК изменяется.

График будет обновляться каждый раз, когда изменяется значение в ячейке V2000, а это происходит, когда вы записываете **новое** значение в ПЛК.

## Ответы на вопросы

### Я получил сообщение “No foreign application responded to a DDE initiate” (Нет внешнего приложения, отвечающего на инициализацию DDE)

DSDData Сервер в данный момент не запущен. Если приложение использует **автоматическую** установку связи (**Automatic links**), Вы должны запустить DSDData Сервер перед запуском Visual Basic. Если Вы используете только **ручную** установку связи (**Manual links**), то надо запустить DSDData Сервер перед запуском приложения.

Существует синтаксическая ошибка в имени сервиса. Название сервиса не зависит от регистра, но должно быть написано без ошибок. Имя сервиса для нашего DSDData Сервера всегда “Dsdata”.

Существует синтаксическая ошибка в имени темы. Название темы не зависит от регистра, но должно быть написано в точности так же, как оно появляется в конфигурационном экране DSDData Сервера. Одной из самых распространенных ошибок является использование пробелов в имени темы. Пробелы использовать допускается, но в этом случае надо помещать имя темы в одинарные кавычки, например, ‘Demo Topic’.

### Я получил сообщение “Foreign application won’t perform DDE method or operation” (Внешнее приложение не поддерживает DDE метод или операцию)

Существует синтаксическая ошибка в части элемента тега, например, введено G2000 вместо V2000.

В элементе есть ссылка на тип данных, который является неверным для указанного ПЛК, например, пытаетесь прочитать R400, тип данных регистров DL305, а связь сконфигурирована для DL205 ПЛК.

В поле элемента не указано значение.

### Я получил сообщение “Timeout while waiting for DDE response” (Истекло время ожидания DDE ответа)

Существует проблема в кабельном соединении с ПЛК. Проверьте все соединения между вашим компьютером и ПЛК. Если установлен концентратор между ПК и ПЛК, проверьте индикатор ‘Link good’ (есть связь) для каждой точки подключения.

### Мое приложение плохо работает, когда существует несколько автоматически устанавливаемых связей на форме.

Мы обнаружили проблему Visual Basic версии 4 и 5 с приложениями, которые имеют более 32 автоматически устанавливаемых связей на одной форме. Если ваше приложение требует более 32 автоматически устанавливаемых связей, создайте несколько форм и поместите на них не более 32 автоматически устанавливаемых связей.

### VB6 обваливается с 33 автоматическими связями.

Эта проблема исправлена в пакете обновления 2 (Service Pack 2), который доступен на сайте Microsoft.

## Использование DSData Сервера с InTouch фирмы WonderWare

---

В этой главе...

- Вступление
  - Начинаем работать с InTouch
  - Чтение значений регистров
  - Чтение значений дискретных входов/выходов
  - Запись значений регистров
  - Ответы на вопросы
-



## Вступление

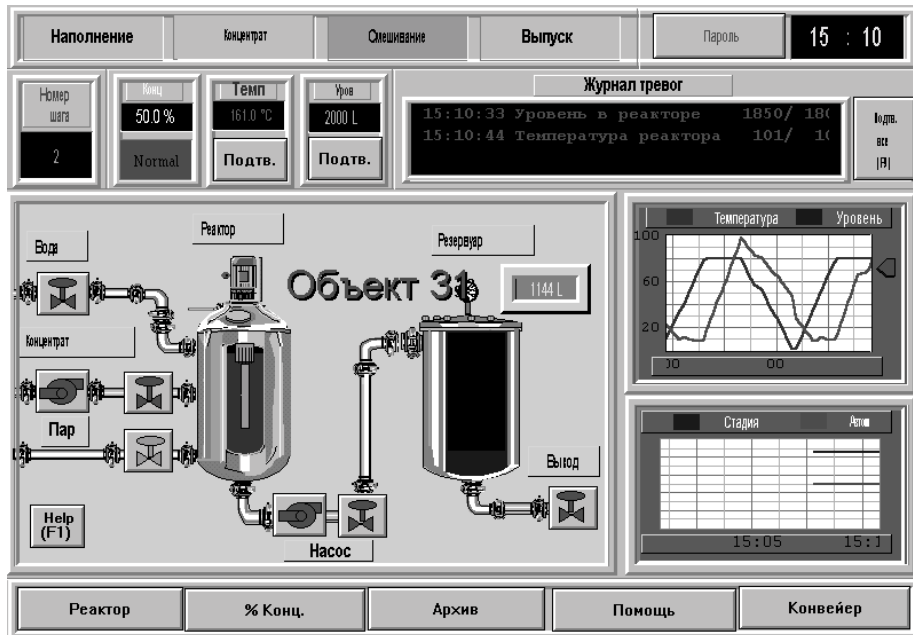
### Что такое InTouch?

Программное обеспечение InTouch фирмы WonderWare - это Windows-приложение, разработанное для использования компьютера при решении задач сбора информации и управления механизмами или процессами, т.е. операторского интерфейса (HMI). В отличие от других пакетов, обсуждаемых в этом руководстве, InTouch разработан специально как пакет для промышленной автоматизации. Как результат, элементы, необходимые для приложений управления ПЛК, еще более просты в конфигурации и использовании. InTouch поставляется с большой библиотекой графических элементов для создания визуального представления процесса на экране вашего компьютера. Конфигурирование объектов для общения с ПЛК состоит из выбора одного из графических объектов, размещения его в окне управления процессом и заполнения соответствующих полей, которые описывают необходимые точки в ПЛК.

### Как он общается с ПЛК

Так же, как мы это делали с другими программным продуктам, описанными в этом руководстве, DSDData Сервер предоставляет необходимый инструмент для конфигурации необходимой связи и темы и коммуникационного драйвера для всех ПЛК **DirectLOGIC** и совместимых с ними. InTouch предоставляет необходимый инструмент для создания имени доступа к DDE и 'Тегов', которые отображают точки данных ПЛК.

Экран внизу является примером некоторой комплексной схемы управления, которую вы можете создать с помощью InTouch. В этом руководстве мы не будем пытаться создавать что-нибудь столь же замысловатое. Мы покажем вам основные шаги, необходимые для доступа к точкам данных ПЛК с помощью InTouch.



## Начинаем работать с InTouch

### Вам нужна Связь и Тема

Перед тем, как вы начнете использовать InTouch, вам необходимо сделать следующее:

- Подать питание в ПЛК.
- Соединить нужным кабелем ПЛК и компьютер.
- Создать коммуникационную Связь с ПЛК.
- Создать Тему с именем DemoTopic, которая будет ссылаться на эту Связь, оставить время обновления 0 и не присоединять документации.

Пожалуйста, обратитесь к главе 2, если Вам нужна дополнительная информация для выполнения этих задач.



---

**Примечание:** Пример и демонстрационная программа, используемая в этой главе, разработана с использованием **Wonderware InTouch Version 7.0** (русская версия). Если Вы не используете эту версию InTouch, не огорчайтесь, все равно можно использовать эту главу как инструкцию. Методы и синтаксис могут отличаться, но процесс будет очень похожим.

---

### Имя доступа к DDE (DDE Access Name)

Реализация DDE требует, чтобы Вы определили **Сервис** и **Тему** для каждого тега, который вы создаете в своем приложении. InTouch пытается уменьшить повторение комбинаций, используя имя доступа к DDE. Смысл использования имени доступа к DDE в комбинировании двух частей Сервис и Тема в одну. Для создания тега теперь необходимо просто задать имя доступа к ПЛК и элемент со всеми необходимыми модификаторами.

Имя доступа к DDE почти единственное, что надо определить, но имеет смысл использовать что-то очень похожее на имя Темы. Имя доступа к DDE даже может быть точно таким же, как имя темы.



---

**Примечание:** Обратитесь к руководству по InTouch для получения всех ограничений на имя доступа к DDE и Темы.

---

Пример ограничений:

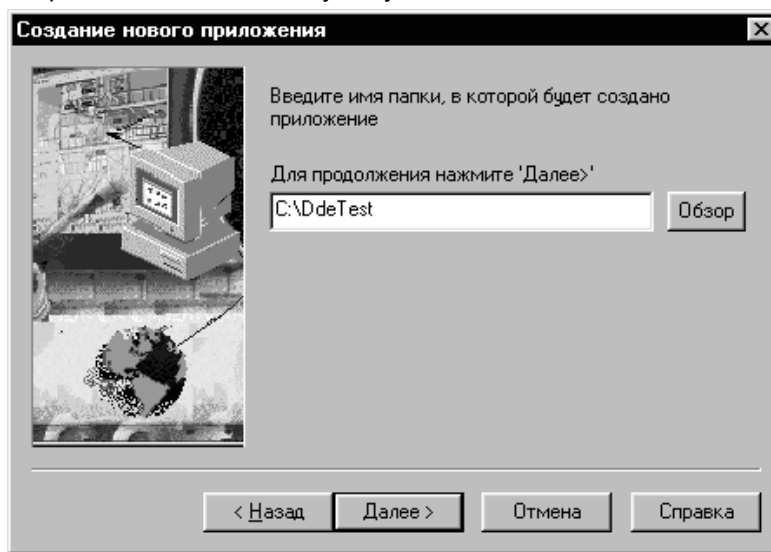
**нельзя использовать первым символом число, использовать двоеточие, апостроф или символ пробела в имени доступа к DDE и имени Темы.**

---

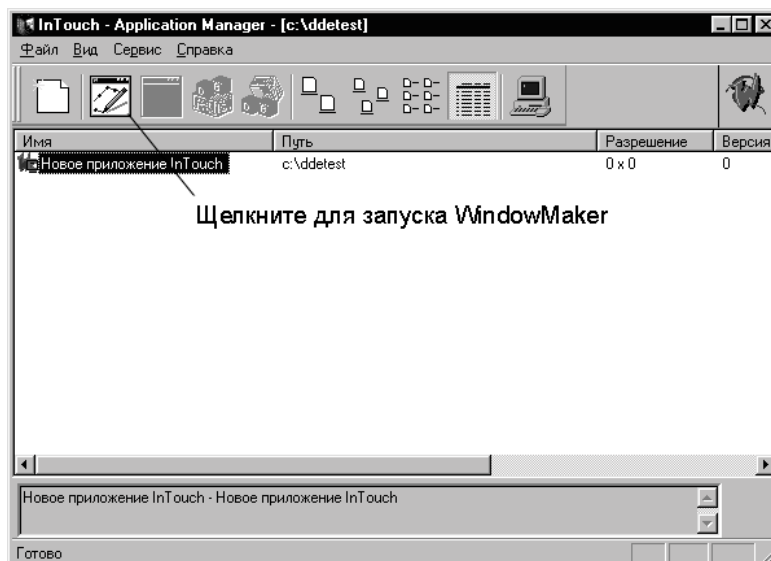
## Чтение значений регистров

Пример показывает шаги, необходимые для чтения значения регистров ПЛК, затем использование этих значений для анимации объектов. Это все кажется сложнее, чем есть на самом деле. Мы прочитаем второе значение из часов, встроенных в ПЛК, и отобразим его в виде стрелки уже имеющегося в InTouch измерительного прибора. Этот прибор будет работать только с ПЛК, которые имеют встроенные часы/календарь. Посмотрите в руководстве пользователя, пришедшим с ПЛК, содержит ли ПЛК встроенные часы.

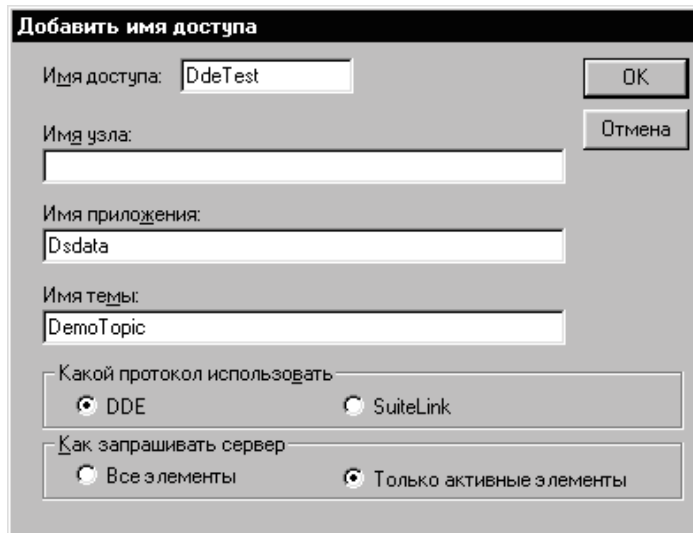
1. Нам надо создать новый каталог для нашего примера. Щелкните по кнопке Создать. Введите название каталога, в котором Вы хотите сохранить пример. Когда закончите, щелкните по кнопке **Далее**, затем по **Готово** в появившемся диалоге для возвращения к начальному окну.



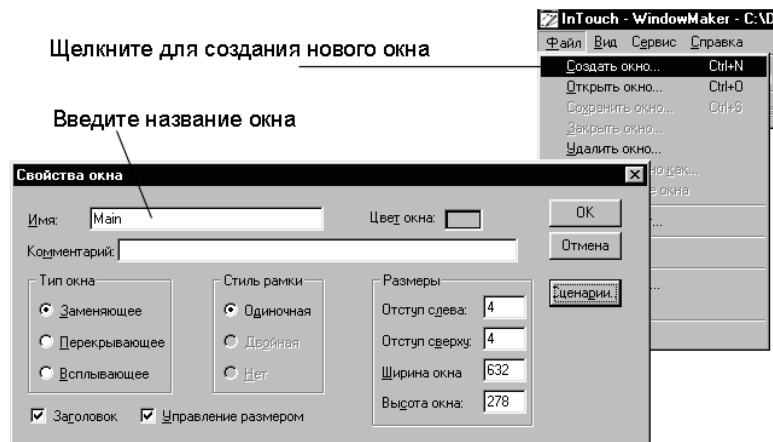
2. Одна часть InTouch – приложение, называемое WindowMaker. Она используется для создания окон и рисования на них управляющих элементов. Она также необходима для создания новых тегов и управления базой данных тегов. Запустите WindowMaker, для этого выберите нужный каталог и щелкните по кнопке WindowMaker в начальном окне.



- Первое, что надо создать в этом примере, это имя доступа к DDE. В панели меню выберите **Сервис | Имена доступа**. Затем щелкните кнопку **Добавить** для вызова диалога добавления имени доступа к DDE. Имя DDE приложения всегда Dsdata. Имя DDE Темы – это имя темы, которую вы хотите использовать. В нашем примере мы используем только что созданную тему, названную DemoTopic. Когда закончите, щелкните по кнопке **ОК**, затем по **Закр~~ыть~~** в появившемся диалоге для возвращения к начальному окну.



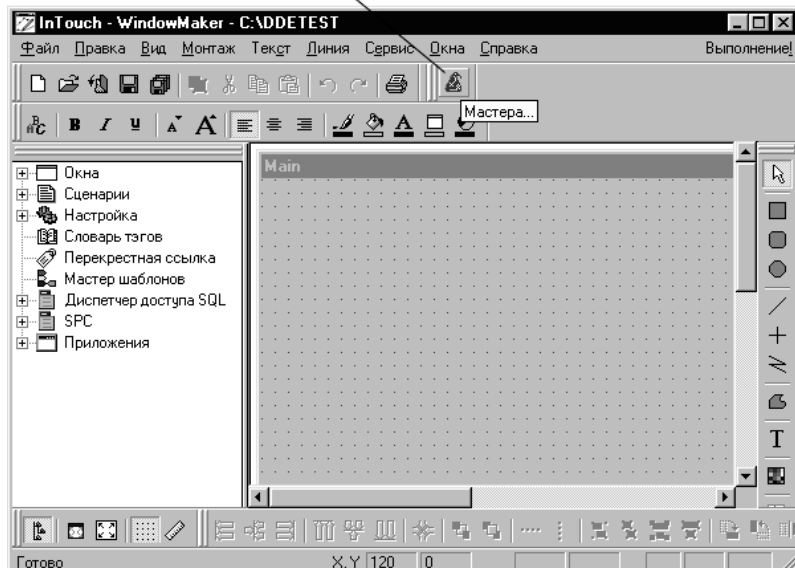
- Следующая необходимая вещь - это окно для отображения наших управляющих элементов. В панели меню выберите **Файл | Создать окно**. Появившийся диалог позволит ввести имя окна и определить его размер и цвет.



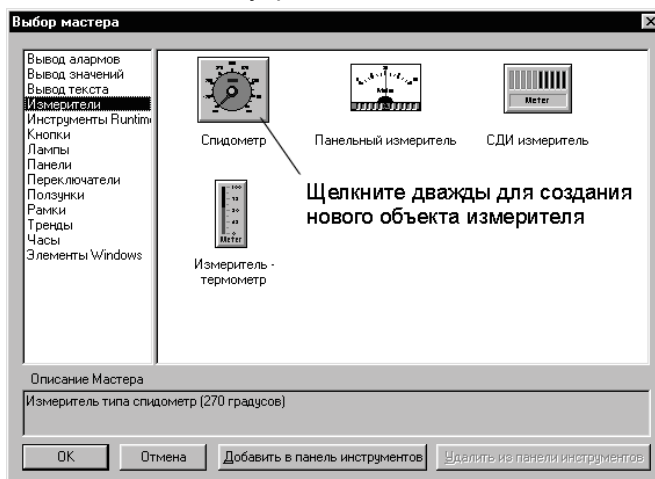
## Выбор мастера

Теперь Вы увидите экран, похожий на приведенный. Он содержит пустое управляющее окно и палитру инструментов. Палитра инструментов содержит инструменты для рисования, предназначенные для манипулирования графическими объектами в окне управления. Один из широко используемых инструментов - это мастер. Мастер содержит несколько predefined графических объектов, которые представляют собой различные управляющие элементы, такие как кнопки, измерительные приборы, лампочки и т.д.

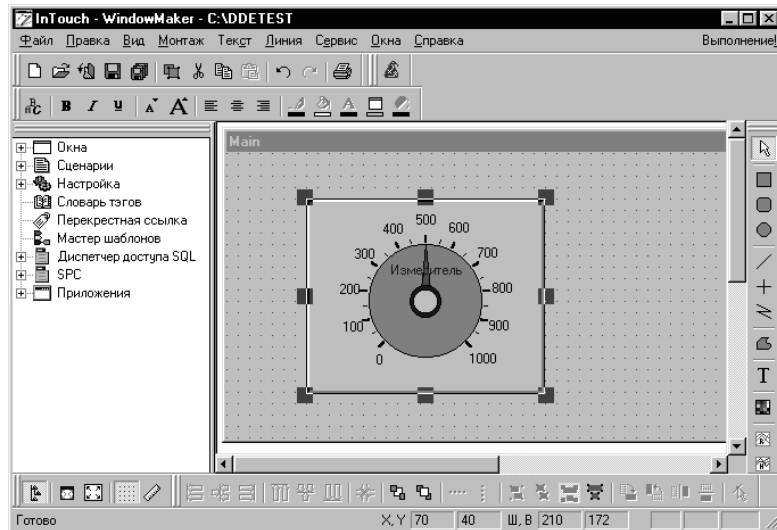
Щелкните здесь для выбора мастера



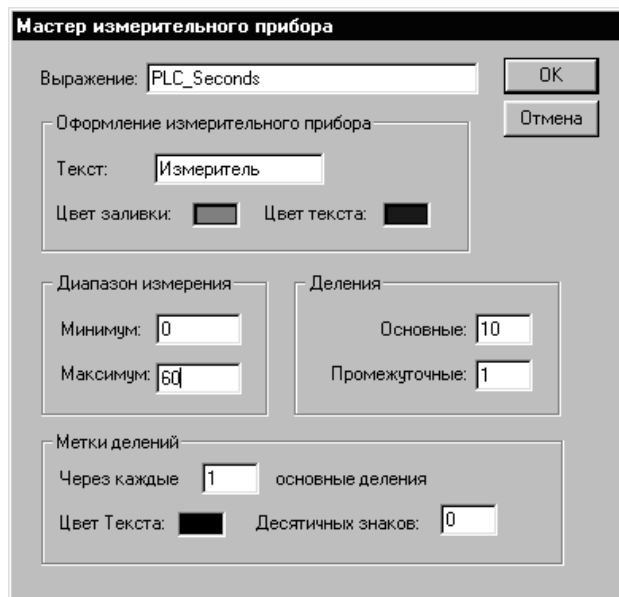
- После щелчка по кнопке мастера, появится следующий диалог. Список с левой стороны показывает группы управляющих элементов, которые можно выбрать. Справа показаны управляющие объекты, содержащиеся в каждой группе. В нашем случае щелкните по группе **Измерители** в списке слева, затем дважды щелкните по первому измерителю в списке справа. Это выберет измеритель для размещения его в окне управления.



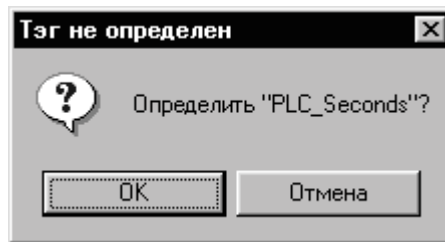
6. Курсор поменяет тип для напоминания о том, что Вы собираетесь поместить объект на окно. Курсор определяет место левого верхнего угла объекта. Поместите курсор туда, где Вы хотите расположить измеритель и щелкните по левой клавише мыши. Будьте *осторожны*, каждый щелчок мышки создаст *новый измеритель*. Экран будет выглядеть как этот.



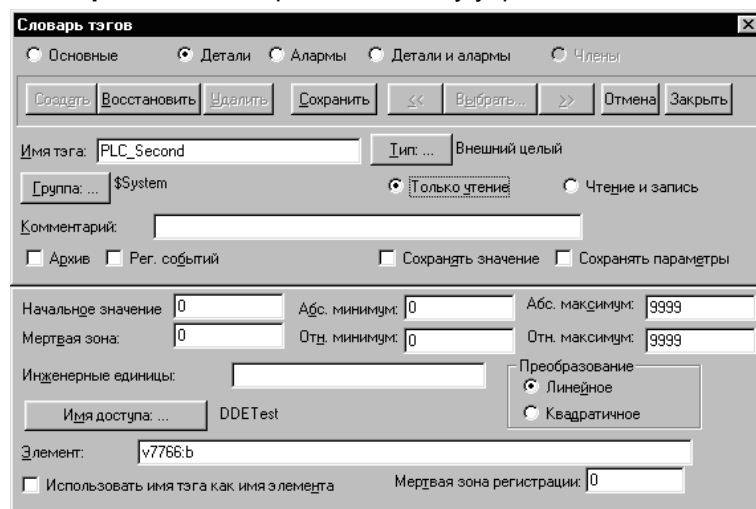
7. На следующем шаге надо сконфигурировать измеритель. Начните с двойного щелчка по измерителю для вызова мастера измерителя. Этот диалог позволит определить визуальные характеристики измерителя. Сделайте следующие изменения:
- Замените **Выражение** на PLC\_Second.
  - Замените **Текст** на Second.
  - Установите **Диапазон измерения Минимум** в 0 и **Диапазон измерения Максимум** (максимум диапазона измерителя) в 60.
  - Установите **Деления основные** в 10 и **Деления промежуточные**.



8. Когда Вы щелкните по кнопке **ОК**, Вас попросят определить тег "PLC\_Seconds". Щелкните **ОК** для перехода в справочнике имен тегов.

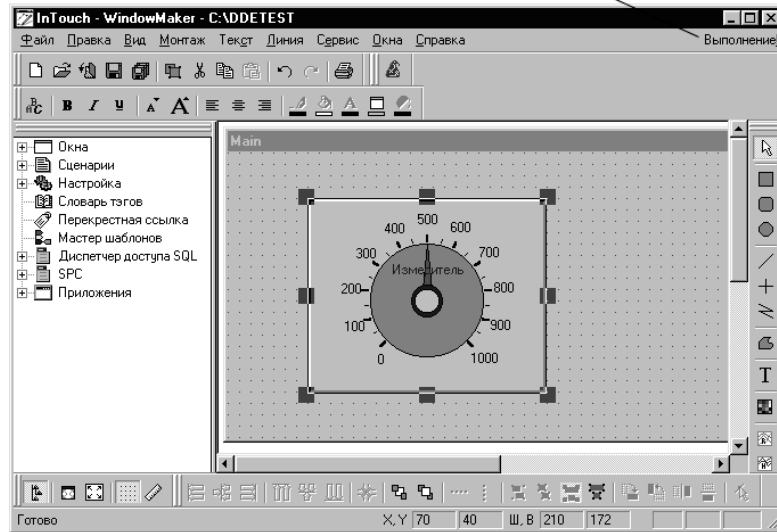


9. Следующим шагом будет заполнение значений, необходимых для определения PLC\_Seconds в базе данных тегов.
- Щелкните по кнопке **Тип** и смените тип на **Внешний целый**, т.к. значение хранится в регистре ПЛК.
  - Щелкните по кнопке **Только чтение**, чтобы сделать тег только для чтения.
  - Щелкните по кнопке **Имя доступа** и выберите имя доступа к DDE DDETest, затем щелкните по **Заккрыть**. Вы настроили InTouch ссылаться для обновления данных на имя доступа к DDE.
  - Введите значение "V7766:B" в поле **Элемент**. Это адрес регистра ПЛК, который содержит число секунд встроенных часов. Это значение хранится в формате VCD, поэтому мы должны добавить модификатор ":B".
  - Остальные значения по умолчанию вполне подойдут. Когда закончите вводить эти значения, щелкните **Заккрыть** для возвращения к окну управления.



10. Все, что осталось - это перейти в режим Выполнение! (выполнения) и наблюдать за изменением измерителя в соответствии с изменением данных в ПЛК. Щелкните **Выполнение!** в правом верхнем углу панели меню для запуска WindowViewer, который будет выполнять наше приложение. Стрелка на измерителе теперь будет увеличивать значение каждую секунду, так же, как секундная стрелка на ваших часах. Для возвращения в WindowMaker щелкните по **Разработка!** в правом верхнем углу WindowViewer.

Щелкните здесь для перехода в режим выполнения и активизации приложения



Это все, что надо сделать для чтения значения из ПЛК и отображения его в графическом объекте. Процесс доступа к любым другим регистрам ПЛК точно такой же, как мы здесь описали.



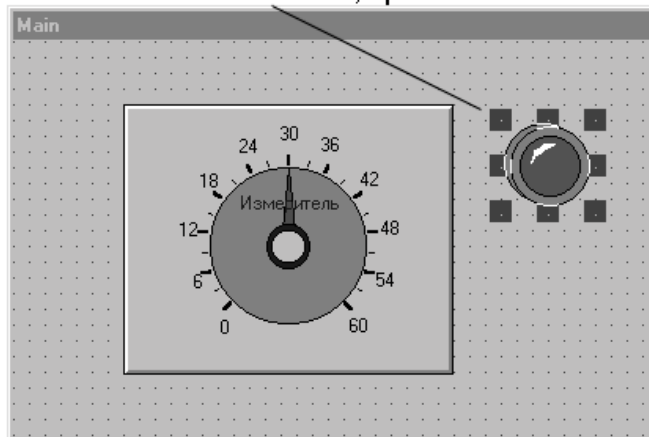
## Чтение значений дискретных входов/выходов

Этот процесс очень похож на процесс добавления объектов для чтения дискретных входов/выходов. Для примера, давайте добавим лампочку к существующему приложению, которая будет отображать состояние дискретной точки входа X0. Лампочка будет одного цвета, если вход выключен и другого цвета, если вход включен.

Прежде чем вы сможете добавить к приложению следующий элемент управления, запустите WindowMaker.

1. В окне выбора мастеров выберите один из объектов из группы **Лампы (Lights)** и поместите его на окно управления.
2. Дважды щелкните по лампочке, установите **Выражение (E[pression])** в **X0\_Status**. Щелкните **ОК (Done)**.
3. Щелкните **ОК** на запрос “Определить X0\_Status?”.
4. Щелкните по кнопке **Тип** и выберите **Внешний дискретный (DDE Discrete)**.
5. Если имя доступа не установлено в DdeTest, щелкните по кнопке и выберите DdeTest, как имя доступа к DDE.
6. В поле **Элемент(Item)** введите X0.
7. Щелкните **Заккрыть (Done)** для возвращения в окно управления. Щелкните **Выполнение! (Runtime!)** для запуска WindowViewer.

**Эта лампочка отображает состояние X0, зеленая если X0 выключен, красный если включен**



Лампочка на экране будет менять цвет каждый раз, когда Вы меняете значение входа X0.

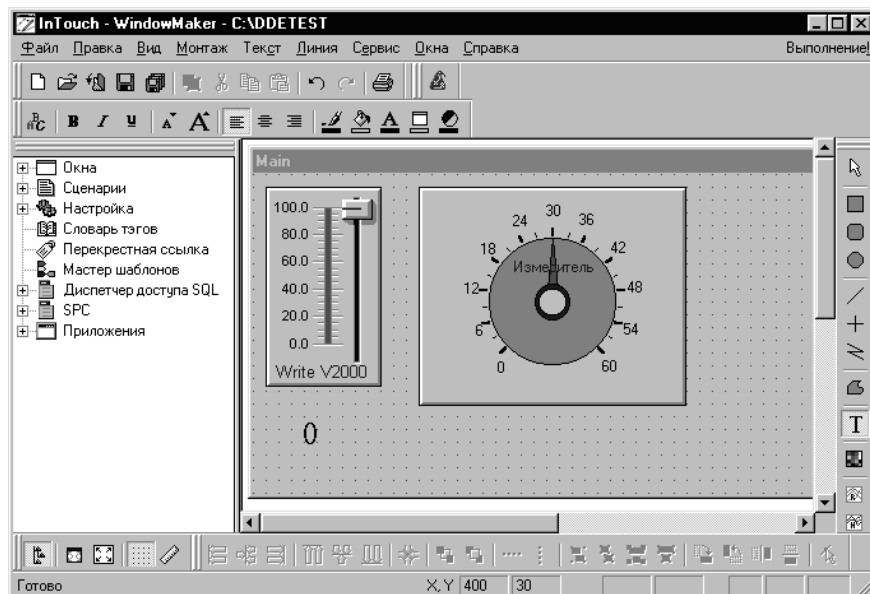
## Запись значений регистров

Создание тегов, записывающих значение в ПЛК, очень похоже на создание тегов, читающих значение регистров ПЛК. Используя окно выбора мастера, можно выбрать объект, который может записывать значения, поместить объект на управляющее окно, изменить его внешний вид и создать тег, который будет отображать объект на регистр ПЛК.

Посмотрите на примере, как использовать ползунок для записи значений в регистр V2000 ПЛК. Мы дополнительно поставим объект для отображения значений, который будет читать только что записанное значение, чтобы точно убедиться в том, что запись произошла.

Прежде чем вы сможете добавить к приложению следующий элемент управления, запустите WindowMaker.

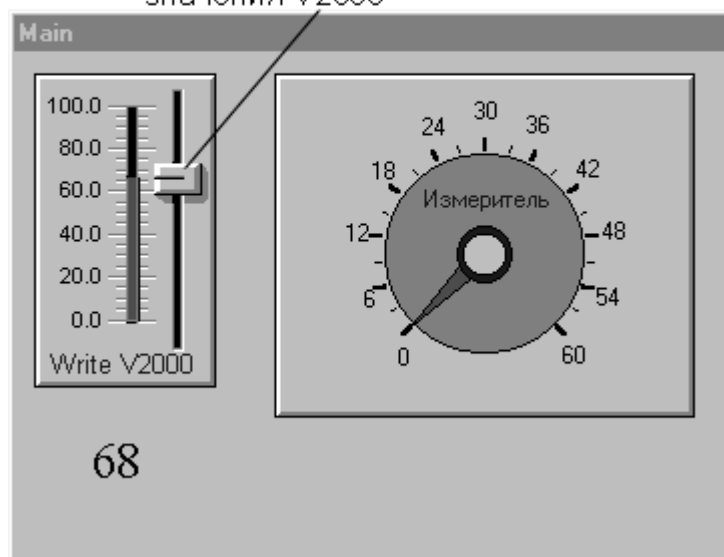
1. В окне выбора мастеров (Wizard Selection), выберите группу Ползунки (Slider), выберите единственно доступный ползунок и поместите его на окно управления.
2. Дважды щелкните по ползунку, установите имя тега в **V2000\_Slider** и метку (Label) в "Write V2000". Щелкните **Ок**.
3. Щелкните **ОК** на запрос "Определить V2000\_Slider?".
4. Щелкните по кнопке **Тип** и выберите **Внешний целый**.
5. Если имя доступа к DDE не установлено в DdeTest, щелкните по кнопке и выберите DdeTest, как имя доступа к DDE.
6. В поле **Элемент** введите V2000:B, затем щелкните **Заккрыть**.
7. В окне выбора мастеров выберите группу Вывод значений, выберите элемент Вывод аналогового тега и разместите их в окне управления.
8. Дважды щелкните по объекту для отображения значений. Установите имя тега в **V2000\_Slider** (мы определили то же имя тега для ползунка). Установите формат отображения значения так, чтобы отображалась одна цифра (щелкните кнопку вниз в поле формата числа и выберите единственный 0), затем щелкните **Ок**. Ваш экран будет похож на следующий:



9. Щелкните **Выполнение! (Runtime!)** для запуска WindowViewer и оживления приложения.

10. Нажмите левую клавишу мыши на ползунке и удерживайте ее. Теперь потяните ползунок вверх и вниз, и ползунок будет записывать текущие значения своей позиции в регистр V2000. Объект для отображения значений также будет показывать текущее значение регистра.

Передвигайте ползунок мышкой  
вверх или вниз для изменения  
значения V2000



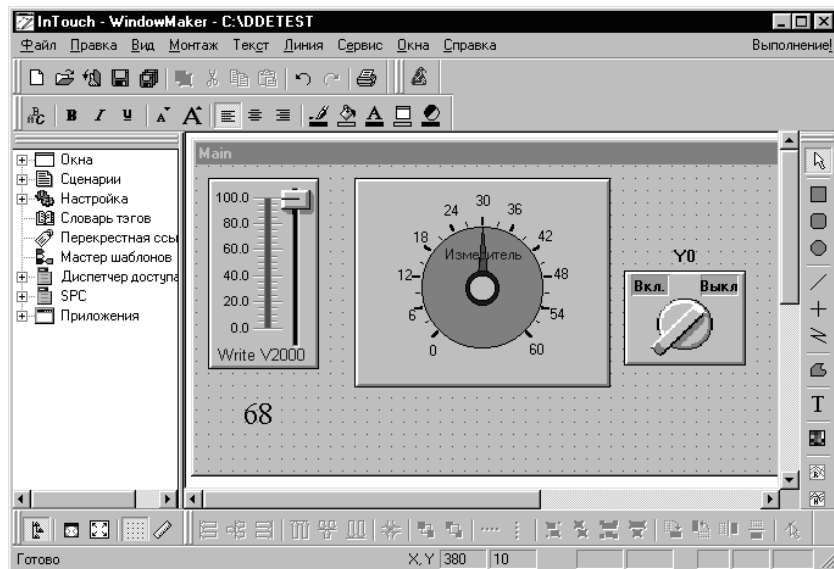
## Запись значений дискретных входов/выходов

Создание тегов, записывающих значение дискретных входов/выходов в ПЛК, очень похоже на создание тегов, читающих значение дискретных входов/выходов. Используя окно выбора мастера, можно выбрать объект, который может записывать значения, поместить объект на управляющее окно, изменить его внешний вид и создать тег, который будет отображать объект на дискретный вход/выход ПЛК.

Посмотрите на следующем примере, как использовать переключатель для переключения выхода Y0 в ответ на щелчок мыши.

Прежде чем добавить к приложению следующий элемент управления, запустите WindowMaker.

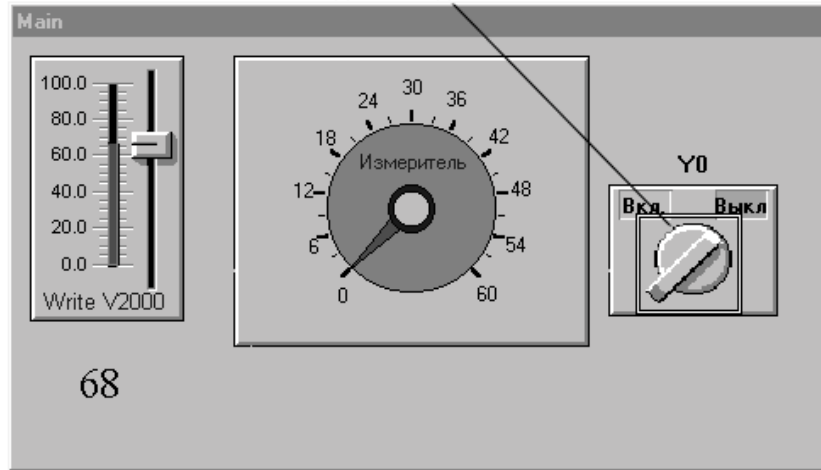
1. В окне выбора мастеров выберите группу Переключатели, выберите любой доступный тип переключателя и поместите его на окне управления.
2. Дважды щелкните по переключателю, установите имя тега в **Y0\_Status** и поле надпись в "Y0". Щелкните **Ок**.
3. Щелкните **ОК** на запрос "Определить Y0\_Status?".
4. Щелкните по кнопке **Тип** и выберите **Внешний дискретный**.
5. Если имя доступа к DDE не установлено в DdeTest, щелкните по кнопке и выберите DdeTest, как имя доступа к DDE.
6. В поле **Элемент** введите Y0, затем щелкните **Заккрыть**. Ваш экран будет похож на следующий:



7. Щелкните **Выполнение! (Runtime!)** Для запуска WindowViewer и оживления приложения.

8. Каждый раз, когда Вы щелкаете по переключателю, выход Y0 должен изменить состояние. Если он включен, он должен стать выключен и наоборот.

Каждый щелчок мыши будет переключать состояние точки ввода/вывода Y0



**ПРИМЕЧАНИЕ:** ПЛК *DirectLOGIC* не поддерживают форсирование входов (нельзя задать значения реальным входам).

Если дискретная точка, в которую Вы хотите записать данные, реально используется **входным модулем**, информация, которая запишется через DSData Сервер, будет действительна только в течение одного цикла. Так произойдет потому, что ПЛК обновляет состояние всех входных модулей в начале каждого цикла. Это обновление перезапишет изменения, сделанные DSData Сервером. Это также верно для выходных модулей и управляющих реле, за исключением тех случаев, когда программа ПЛК разрешает выполнить обновление. Если по программе точка должна быть деактивизирована, она будет деактивизирована, и перезапишет любую информацию, переданную через DSData Сервер.

## Заключение

Мы показали самый простой пример в этом руководстве и не использовали всех способов, которые InTouch может использовать для обмена данными с ПЛК через DSDData Сервер.

Существует несколько способов, с помощью которых InTouch выполняет DDE-передачу данных. Наш DSDData Сервер обрабатывает все запросы на передачу данных одинаково. Теперь вы знаете, как создать каждый из четырех типов тегов (чтение и запись дискретных входов/выходов, чтение и запись регистров), это все аспекты использования DSDData Сервера для разработки вашего приложения.

Осталось узнать, как эффективно использовать InTouch сам по себе. Пожалуйста, обратитесь к документации по InTouch.

## Ответы на вопросы

### Я получил сообщение “Could not initiate DDE Conversation... Dsdata|Topicname”

DSDData Сервер не запущен. Вам необходимо запустить DSDData Сервер перед запуском WindowViewer.

Синтаксическая ошибка в имени Сервиса. Имя Сервиса не чувствительно к регистру, однако, должно быть без ошибок. Сервис для нашего DSDData Сервера всегда “Dsdata”

Синтаксическая ошибка в имени Темы. Имя Темы не чувствительно к регистру, однако, должно быть таким же, какое появляется в окне конфигурации DSDData Сервера.

Выполняемый файл DSDData Сервера DSDATA.EXE не доступен через переменную окружения PATH (путь). Добавьте папку, где находится выполняемый файл DSDData Сервера в определение PATH, находящееся в C:\Autoexec.Bat, например, PATH=C:\Dirctsf32\bin;C:\;C:\Windows.

В отличие от Visual Basic, InTouch не разрешает использовать имя темы, которое содержит пустые символы (табуляция и пробел). Проконсультируйтесь в документации по InTouch обо всех других ограничениях, важных для определения имени темы.

### Объект не оживает

Синтаксическая ошибка в имени элемента, например, введено G2000 вместо V2000.

Вы ссылаетесь в поле элемента на тип данных, который не верен для ПЛК, определенного в конфигурации связи, например, пытаетесь читать R400 – тип регистров только в DL305, когда связь сконфигурирована на использование DL205 ПЛК.

### Я получил сообщение “Timeout while waiting DDE response” (Время ожидания ответа DDE вышло).

Проблема в соединительном кабеле с ПЛК. Проверьте все соединения между компьютером и ПЛК. Для соединения посредством Ethernet, если между ПК и ПЛК установлен концентратор, проверьте индикатор “Link Good” в каждой точке соединения.

В этой главе...

- OPC (OLE для управления процессом)
  - Использование OPC клиента для тестирования
  - Использование программы операторского интерфейса (HMI) как OPC клиента
-



## OPC (OLE для управления процессом)

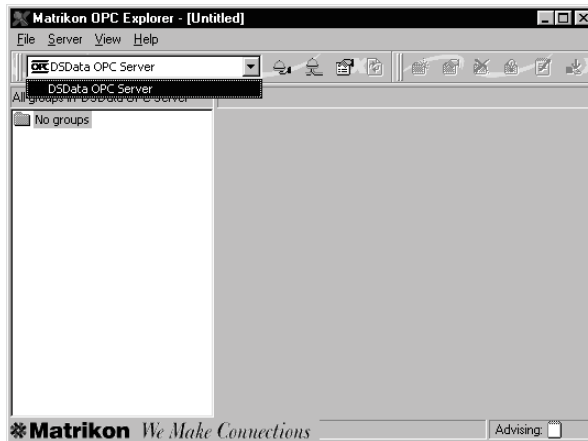
DSData сервер работает в качестве OPC сервер также хорошо, как и в качестве DDE сервера. Он предоставляет связь между клиентским приложением OPC и ПЛК (и вводами/выводами). Обратитесь к вступлению к OPC на странице 1-7 для получения более подробной информации, относящейся к OPC.

### Использование OPC клиента для тестирования

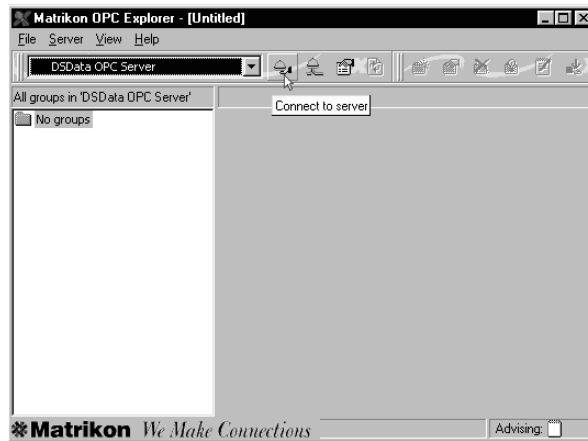
Если вы хотите попробовать использовать DSData сервер без использования HMI (программы человеко-машинного интерфейса) или пакета управления, обратитесь к примеру ниже. В дальнейшем показывается использование приложения OPC клиента – Matrikon Explorer совместно с DSData сервером. Во время написания этого руководства Matrikon Explorer мог быть бесплатно получен с [www.matrikon.com](http://www.matrikon.com).

**Шаг 1:  
Соединение с  
DSData сервером**

Сначала выберите DSData сервер из ниспадающего списка.



Затем щелкните по кнопке *Connect to server* (соединения с сервером).



**Примечание: DSData сервер должен быть запущен и подсоединен к ПЛК. (Смотрите главу 2 для создания связи с ПЛК).**

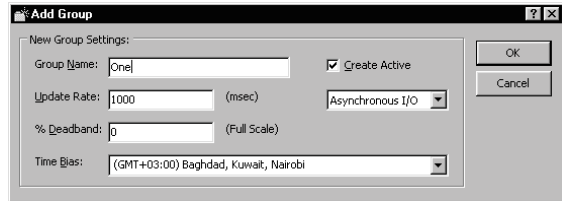
OPC и DSData сервер

**Шаг 2:  
Создание  
группы**

Щелкните по кнопке *Create New Group (Создать новую группу)* для добавления новой группы.

Группа OPC подобна папке для размещения в ней элементов.

Дайте группе название и щелкните ОК.



**Шаг 3:  
Создание  
элемента**

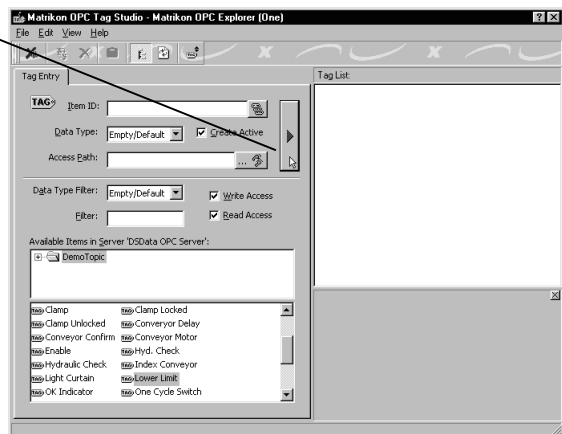
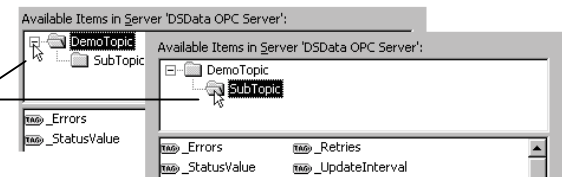
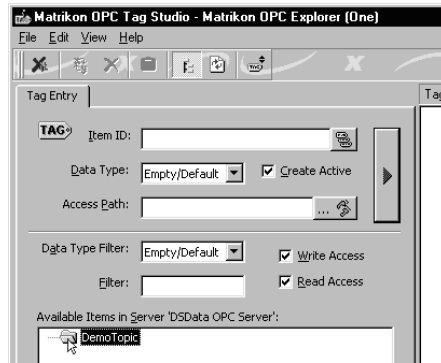
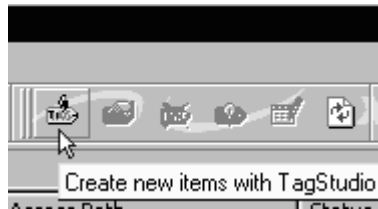
Щелкните по кнопке *Create New Item (Создать новый элемент)* для создания нового элемента (или Тега).

Также как на рисунке справа отображена DemoTopic, появятся все текущие темы DSData сервера. Дважды щелкните по соответствующей теме для отображения обозначений элементов.

Дважды щелкните по обозначению для использования его как элемент. Если нет ни одного обозначения, можно просто набрать регистр ПЛК как идентификатор элемент (Item ID) (например, V2000).

Если Вы хотите создать элемент внутри подтемы, щелкните по символу плюс и выберите подтему.

Затем щелкните по кнопке *Add tag to list (Добавить тег в список)*.

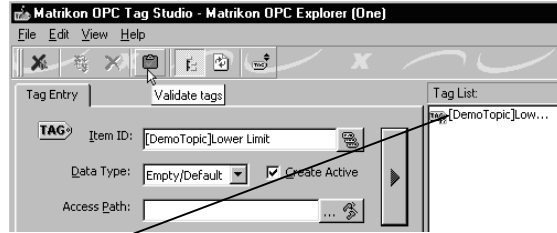


**Шаг 4:**  
Проверка тега

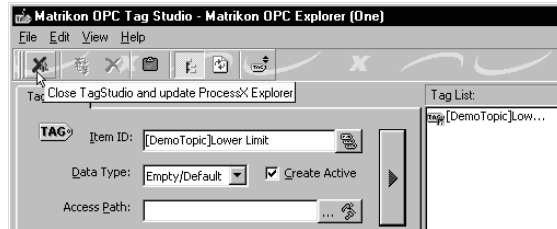
Щелкните по кнопке *Validate tags* (Проверка тега).

Здесь Вы увидите маленькую зеленую галочку, если правильность тега или элемента подтверждается DSDdata сервером.

Если правильность не подтверждается, смотрите примечание ниже.

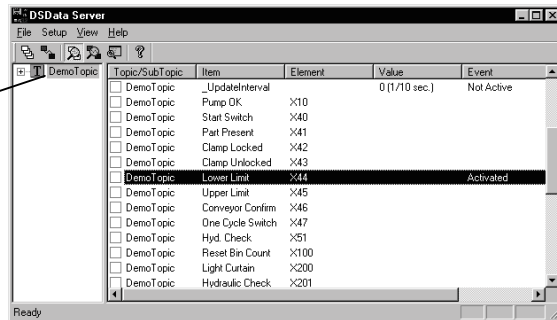


Щелкните по кнопке *Close Tag Studio* (Закреть мастерскую тегов).



**ПРИМЕЧАНИЕ:** Если при проверке тега возникла ошибка (красный X вместо зеленой галочки), убедитесь, что имя темы и имя элемента верны и связь между ПЛК и DSDdata сервером корректна.

Заметьте, что иконка слева ДемоТopic зеленого цвета означает, что связь с ПЛК корректна.



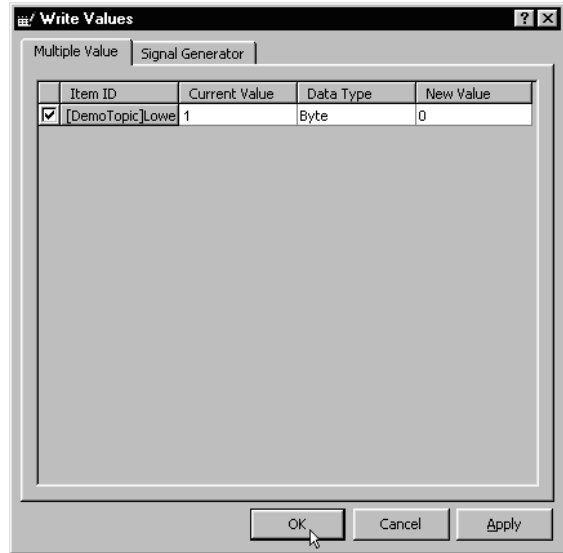
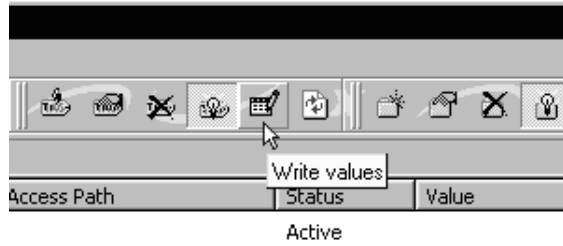
**ПРИМЕЧАНИЕ:** В Matrikon Explorer тег и элемент означают одно и тоже.

**Шаг 5:  
Просмотр и  
запись новых  
значений  
элемента**

Заметьте, что новый элемент Lower Limit отображен со своим текущим значением (1), хранящимся в ПЛК.

Щелкните по кнопке *Write Value (Запись значения)* для записи нового значения в Lower Limit.

Введите новое значение, которое будет храниться в Lower Limit и щелкните ОК.

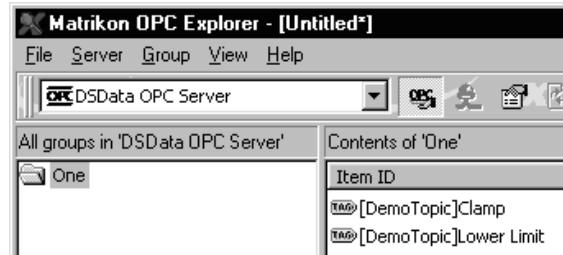


Теперь можно увидеть новое значение (0), хранящееся в Lower Limit, что соответствует ячейке X44 в ПЛК.



**Шаг 6:  
Создание нового  
элемента и т.д.**

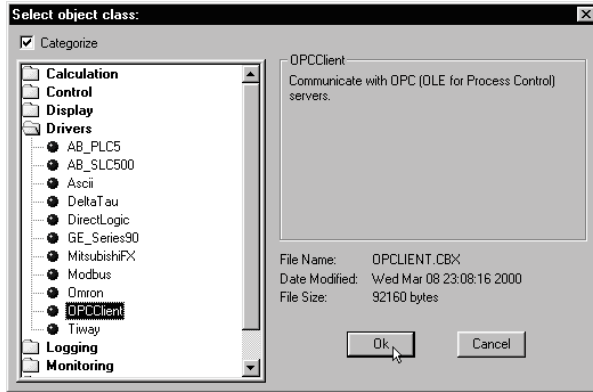
Возвратитесь к шагу 3 и создайте новый элемент.



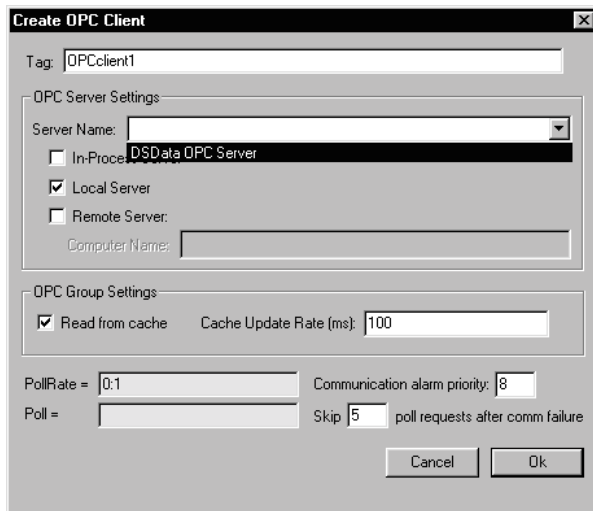
## Использование программы операторского интерфейса (HMI) как OPC клиента

Пример ниже показывает использование связи OPC между DSDdata сервером и программным пакетом HMI (человеко-машинного интерфейса). Поскольку Ваше программное обеспечение HMI может быть другим, процесс соединения и связывания будет достаточно близок, чтобы подсказать Вам, как это происходит.

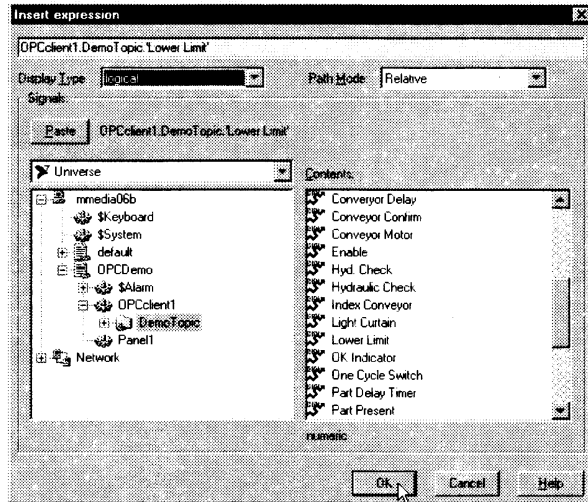
Создайте клиента OPC.



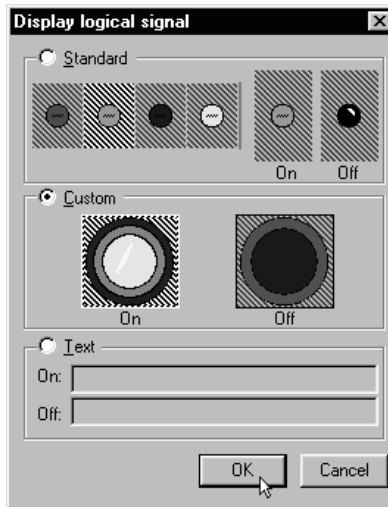
Соедините его с DSDdata сервером.



Создайте индикаторную лампочку (в нашем пакете HMI это называется Expression (выражение)).

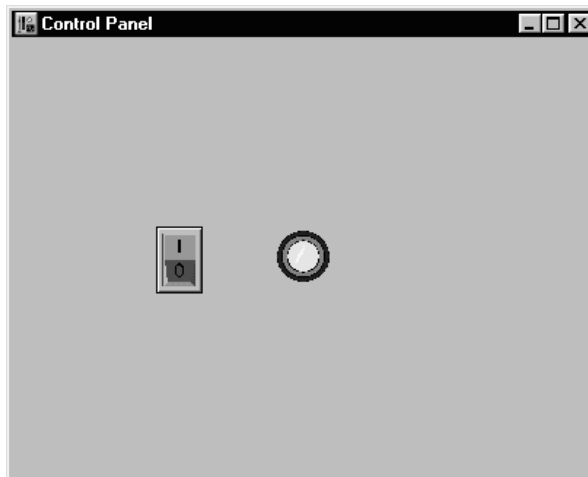


Если у Вас есть желание, измените отображение лампочки.



Подобно создайте переключатель и назначьте ему запись элемента в DSData сервере.

Обычно программные продукты HMI уже имеют кнопки и переключатели (которые записывают значения) и лампочки и индикаторы (которые читают и отображают значения).



## Регистрация данных, протоколы и кабели

---

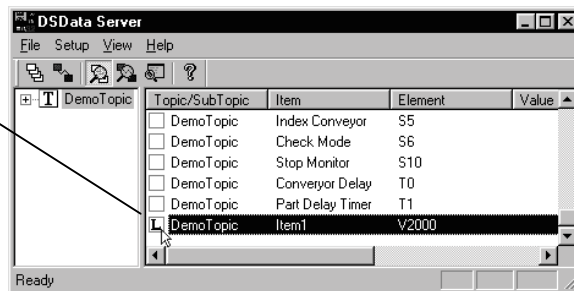
В этой главе...

- Регистрация данных
    - Регистрация отдельных элементов
    - Наблюдение за созданием новых OPC (или DDE) элементов
  - Протоколы
  - Кабели
-

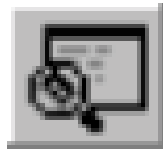
### Регистрация отдельных элементов

DSData сервер использует DBWin32 как инструмент для отладки и для регистрации (архивирования) данных. Этот раздел описывает базовые принципы отображения изменений и создания OPC (и DDE) элементов.

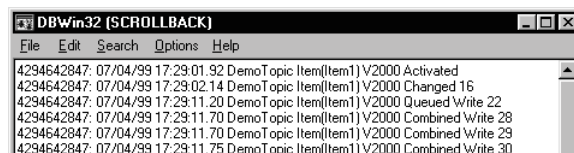
Установите флажок, как показано для регистрации отдельного элемента.



Щелкните по кнопке Launch Logger(Запуск регистратора) для отображения окна DBWin32.



Теперь, когда меняется значение V2000, это отображается в журнале регистрации.

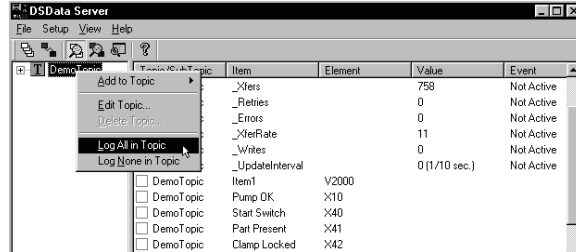




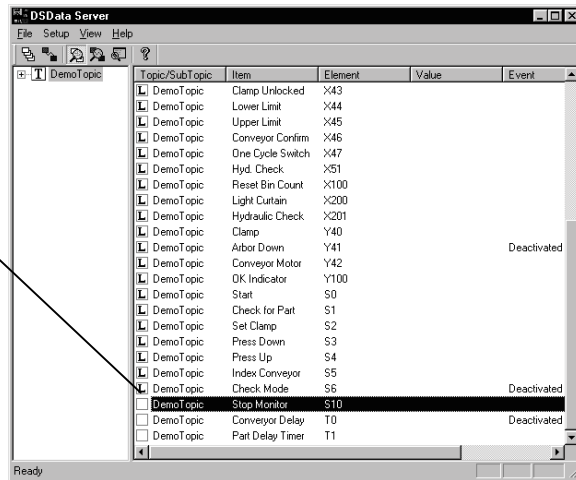
**Наблюдение за созданием новых OPC (или DDE) элементов**

Когда OPC клиент соединяется с DSDData сервером, он может создать новые OPC элементы, например V2400. Если у вас уже есть достаточно большое количество OPC элементов (или DDE), изображенных в DSDData сервере, это способ «захватить» новый элемент, когда он создается клиентом. Таким образом можно «захватить» и новые DDE элементы, на которые ссылаются DDE запросы на чтение или на запись.

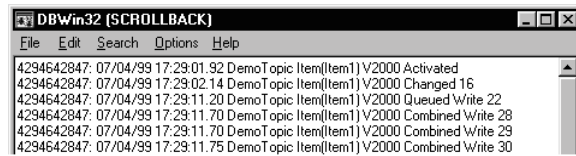
Сначала щелкните правой кнопкой мыши по значку DemoTopic и выберите пункт *Add All in Topic* (Выбрать все в теме). Это приведет к тому, что все элементы в данной теме будут регистрироваться.



Затем установите флажок на всех изображенных элементах.



Теперь, когда будет создаваться новый элемент, это событие будет отображаться в DBWin.32



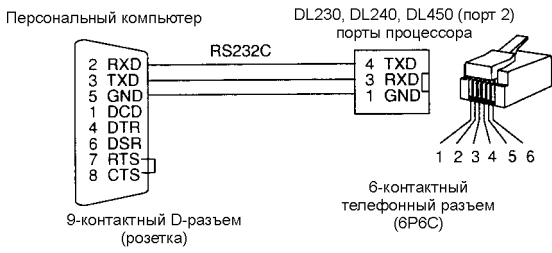
Протоколы и кабели

| Семейство                | Процессор или другое устройство       | Порт              | Кабель               | Поддерживаемые протоколы |            |       |        |
|--------------------------|---------------------------------------|-------------------|----------------------|--------------------------|------------|-------|--------|
|                          |                                       |                   |                      | DirectNET                | K-Sequence | E-COM | MODBUS |
| DirectLogic DL05         | D0-05 (все версии)                    | Только один       | Используйте D2-DSCBL |                          |            |       |        |
| DirectLogic DL105        | DL130 (все версии)                    | Только один       | Используйте D2-DSCBL |                          |            |       |        |
| DirectLogic DL205        | DL230                                 | Только один       | D2-DSCBL             |                          | •          |       |        |
|                          | DL240                                 | Верхний           | D2-DSCBL             |                          |            |       |        |
|                          |                                       | Нижний            | D2-DSCBL             |                          |            |       |        |
|                          | DL250                                 | Верхний           | D2-DSCBL             |                          |            |       |        |
|                          |                                       | Нижний            | D2-DSCBL-1           |                          |            |       |        |
|                          | D2-DCM (модуль)                       | Только один       | D3-DSCBL-2           |                          |            |       |        |
| H2-ECOM (модуль)         | Только один                           |                   |                      |                          |            |       |        |
| DirectLogic DL305        | DL330/DL330P                          | треб. DCU*        | D3-DSCBL-2           |                          |            |       |        |
|                          | DL340                                 | Верхний           | D3-DSCBL-1           |                          |            |       |        |
|                          |                                       | Нижний            | D3-DSCBL-1           |                          |            |       |        |
| D3-DCM (модуль)          | Только один                           | D3-DSCBL-2        |                      |                          |            |       |        |
| DirectLogic DL405        | DL430,DL440                           | Верхний (15-pin)  | D4-DSCBL             |                          |            |       |        |
|                          |                                       | Нижний (25-pin)   | D3-DSCBL-2           |                          |            |       |        |
|                          | DL450                                 | Телефонный разъем | D2-DSCBL             |                          |            |       |        |
|                          |                                       | Верхний (15-pin)  | D4-DSCBL             |                          |            |       |        |
|                          |                                       | Нижний RS232      | D3-DSCBL-2           |                          |            |       |        |
|                          |                                       | Нижний RS422      |                      |                          |            |       |        |
|                          | D4-DCM (модуль)                       | Только один       | D3-DSCBL-2           |                          |            |       |        |
| H4-ECOM                  | Только один                           |                   |                      |                          |            |       |        |
| Модули Slice I/O         | Только один                           | D4-DSCBL          |                      |                          |            |       |        |
| GE• Series 1             | IC610CPU105/106                       | треб. DCU*        | D3-DSCBL-2           |                          |            |       |        |
| TI305™ / SIMATIC® TI305  | 325-07, PPX:325-07                    | треб. DCU*        | D3-DSCBL-2           |                          |            |       |        |
|                          | 330-37, PPX:330-37                    | треб. DCU*        | D3-DSCBL-2           |                          |            |       |        |
|                          | 325S-07 (или 325 вместе со Stage Kit) | треб. DCU*        | D3-DSCBL-2           |                          |            |       |        |
|                          | 330S-37, PPX:330S-37                  | треб. DCU*        | D3-DSCBL-2           |                          |            |       |        |
|                          | 335-37, PPX:335-37                    | Телефонный разъем | D3-DSCBL-1           |                          |            |       |        |
| Если используется DCU*   |                                       | D3-DSCBL-2        |                      |                          |            |       |        |
| TI405™ / SIMATIC® TI405™ | 425-CPU, PPX:425-CPU                  | Только один       | D4-DSCBL             |                          |            |       |        |
|                          | N/A, PPX:430-CPU                      | Верхний (15-к.)   | D4-DSCBL             |                          |            |       |        |
|                          |                                       | Нижний (25-к.)    | D3-DSCBL-2           |                          |            |       |        |
|                          | 435-CPU, PPX:435-CPU                  | Верхний (15-к.)   | D4-DSCBL             |                          |            |       |        |
|                          |                                       | Нижний (25-к.)    | D3-DSCBL-2           |                          |            |       |        |
| Smart Slice™ I/O         | Только один                           | D4-DSCBL          |                      |                          |            |       |        |

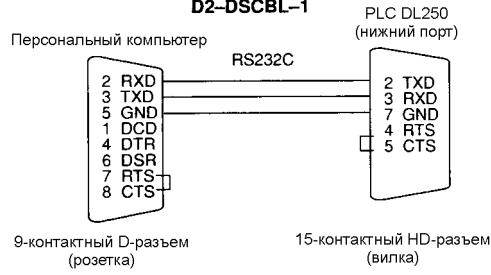
\*-требуется RS232 коммуникационный модуль (D3-232-DCU)

Маркировка контактов соответствует стандартам IBM DTE и DCE

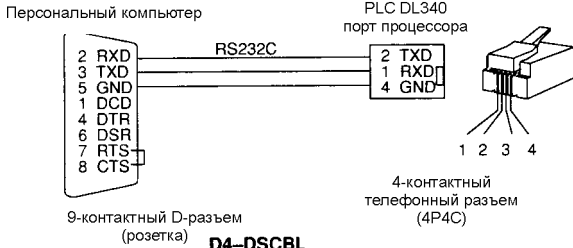
**D2-DSCBL**



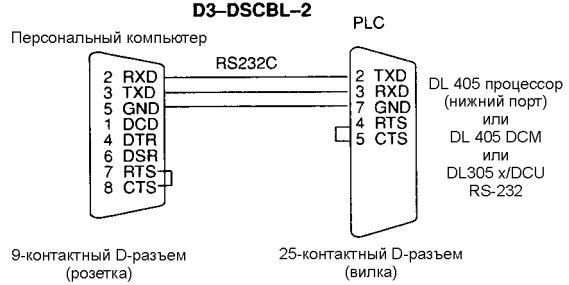
**D2-DSCBL-1**



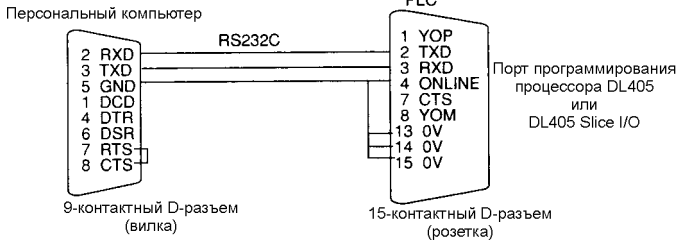
**D3-DSCBL-1**



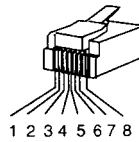
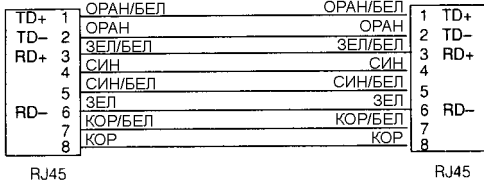
**D3-DSCBL-2**



**D4-DSCBL**

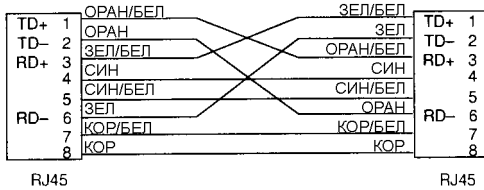


Прямой кабель



8-контактный разъем RJ45 (8P8C)

Перекрещенный кабель



Все Ethernet и Ecom соединительные кабели должны быть 5 категории, UTP. На схеме показано расположение проводов в разьеме RJ45