

D0-DCM. Коммуникационный модуль контроллеров DL05/06.

Руководство пользователя.

D0-DCM-M-RUS

Спасибо за то, что вы купили оборудование для автоматизации фирмы Automationdirect.com™. Мы хотим, чтобы ваше новое оборудование DirectLOGIC™ работало надежно. Каждый, кто устанавливает или использует наше оборудование, должен прочитать эту инструкцию (и всю поставляемую документацию).

Чтобы минимизировать риск возможных проблем, вы должны следовать всем местным и национальным инструкциям, которые определяют правила установки и использования вашего оборудования. Эти правила различны в разных регионах и обычно меняются со временем. Вы должны сами определить, каким правилам надо следовать, проверить условия установки и функционирования оборудования на соответствие с последними редакциями этих правил.

Как минимум, вы должны следовать указаниям соответствующих разделов в национальных пожарных правилах, электрических правилах и указаниям Национальной Ассоциации Производителей Электрооборудования (National Electrical Manufacturer's Association NEMA). Возможно, существуют государственные организации, которые Вам смогут помочь определить, каким правилам и стандартам следует руководствоваться для безопасной установки и использования оборудования.

Отказ следовать национальным правилам и стандартам может привести к порче или серьезному повреждению оборудования. Мы не гарантируем, что продукция, описанная в этой публикации, подходит для вашего конкретного применения, и мы не несем ответственности за ваши проекты, установки и работы.

Наши изделия не являются отказоустойчивыми, их разработка и производство не предполагает использование или перепродажу, как управляющего оборудования работающего в опасных условиях, таких как атомные электростанции, управление полетами, жизнеобеспечивающее оборудование или системы вооружения. Отказ в таких системах может привести к гибели или травмированию персонала, к серьезным повреждениям оборудования или окружающей среды.

Пожалуйста, обратитесь к нам по телефону (495) 925-77-98, если у Вас есть вопросы по установке и применению оборудования, или если Вам необходима какая-либо дополнительная информация.

Эта публикация использует информацию, которая была доступна на момент выхода издания. В **Automationdirect.com™** мы постоянно стремимся улучшить нашу продукцию и услуги, и мы оставляем за собой право делать изменения в своей продукции или инструкциях в любое время без предупреждения и любых обязательств. Эта публикация может содержать описания возможностей, которые могут быть недоступны в определенных версиях наших изделий.

Торговые марки

Эта публикация может содержать ссылки на изделия, произведенные и (или) предлагаемые другими фирмами. Продукция и название компаний, возможно, патентованы и являются собственностью их владельцев. **Automationdirect.com™** не претендует на любые патентованные названия остальных фирм.

Право собственности Automationdirect.com™ Incorporated, 2007. Все права защищены.

Ни одна из частей этой инструкции не может быть скопирована, размножена или передана без предварительного письменного разрешения фирмы **Automationdirect.com™ Incorporated**. **Automationdirect.com™** обладает эксклюзивными правами на всю информацию, включенную в этот документ.

Перевод выполнен в ООО ПЛК Системы. Москва, 2011г.

Редакции руководства

Не забудьте указать номер редакции, если вы связываетесь с нами по поводу этого руководства.

Название: D0-DCM Коммуникационный модуль контроллеров DL05/06

Номер руководства: D0-DCM-M-RUS

Издание/Редакция	Дата	Описание изменений
Оригинал	08/05	Оригинальное издание
2-е издание	09/07	Обновлено применительно к DirectSOFT5.

Оглавление

Глава 1. Введение	1-1
Цели данного руководства.....	1-2
Дополнительные руководства	1-2
Техническая поддержка	1-2
Используемые обозначения.....	1-3
Кратко о D0- DCM	1-4
Замечания по настройке и требованиям к Фирменному ПО ПЛК.....	1-4
Особенности аппаратной части.....	1-4
Использование модуля.....	1-4
Примеры использования D0-DCM	1-5
В качестве интерфейса протокола DirectNET.....	1-5
В качестве дополнительного коммуникационного порта.....	1-6
В качестве MODBUS RTU Master/Slave.....	1-6
Глава 2. Установка, соединения и спецификация.....	2-1
Установка D0-DCM в ПЛК	2-2
Установка D0-DCM.....	2-2
Требования к Фирменному ПО ПЛК и <i>DirectSOFT</i>	2-2
Создание коммуникационного кабеля.....	2-3
Соображение 1: Физическая конфигурация соединения	2-3
Соображение 2: Интерфейс RS22C или RS422.....	2-4
Соображение 3: Схема кабеля.....	2-4
Соображение 4: Характеристики кабеля.....	2-5
Соображение 5: Рекомендации по монтажу.....	2-5
Схемы распайки кабелей	2-6
Соединение по RS-232	2-6
Соединение по RS-485	2-6
Соединение по RS-422	2-7
Спецификация модуля.....	2-8
Общие характеристики	2-8
Порт 1	2-8
Порт 2	2-9
Индикаторы состояния.....	2-9
Глава 3. Настройка D0-DCM	3-1
Информация по	3-2
настройке модуля D0-DCM.....	3-2
Информация для пользователей <i>DirectSOFT5</i>	3-2
Использование <i>DirectSOFT5</i> для настройки D0-DCM	3-3
<i>DirectSOFT5</i> : PLC>Menu>Setup.....	3-3
Выбор слота D0-DCM.....	3-3
Настройка порта 1 (ведомый).....	3-4
Настройка порта 2 (ведомый).....	3-5
Настройка порта 2 (ведущий DirectNET)	3-6
Настройка порта 2 (ведущий Modbus).....	3-7
Настройка порта 2 (Non-Sequence)	3-8
Регистры настроек D0-DCM	3-9

Регистры настроек D0-DCM	3-9
Параметры настроек по-умолчанию.....	3-9
Описание параметров настроек	3-10
A: Порт 1 - Режим передачи, протокол	3-11
B: Порт 1 – Адрес станции, скорость, четность	3-12
C: Порт 2– Задержки RTS, режим передачи, протокол, таймаут, режим RS-485.....	3-13
D: Порт 2 – Адрес станции, скорость, биты данных, стоповые биты, четность	3-15
E: Порт 2 – Задержка символов	3-16
F: Порты 1 и 2 – Код завершения настройки	3-17
G: Порт 1 и 2 – Задержка перезапуска порта.....	3-17
Использование релейной логики для настройки D0-DCM (DL05).....	3-18
Порт 1 (ведомый)	3-18
Порт 2 в режиме ведомого	3-18
Порт 2 в режиме ведущего DirectNET.....	3-19
Порт 2 в режиме ведущего Modbus	3-19
Использование релейной логики для настройки D0-DCM (DL06).....	3-20
Порт 1 (ведомый)	3-20
Порт 2 в режиме ведомого	3-20
Порт 2 в режиме ведущего DirectNET.....	3-21
Порт 2 в режиме ведущего Modbus	3-21
Глава 4. DirectNet. Использование команд RX/WX	4-1
Команды RX / WX.....	4-2
Команды чтения (RX) и записи (WX)	4-2
Создание цепи чтения (RX) или записи (WX).....	4-2
Первая команда LD	4-2
Вторая команда LD	4-3
Команда LDA.....	4-3
Команда RX	4-4
Команда WX	4-4
Адресация к различным типам памяти.....	4-5
Бит-адресуемая память	4-5
Память адресуемая по словам	4-5
Типы памяти	4-6
ЦПУ DL05.....	4-6
ЦПУ DL06.....	4-6
Специальные коммуникационные реле.....	4-7
Программа с одной командой RX	4-8
Программа для ведущего ПЛК	4-8
Программа для ведомого ПЛК.....	4-8
Программа для ведущего ПЛК	4-9
Программа для ведомого ПЛК.....	4-9
Программа с одной командой WX	4-10
Программа для ведущего ПЛК	4-10
Программа для ведомого ПЛК.....	4-10
Программа для ведущего ПЛК	4-11
Программа для ведомого ПЛК.....	4-11
Программа с несколькими командами RX и WX.....	4-12
Блокировочные реле	4-12
Первая команда RX/WX	4-13
Вторая команда RX/WX	4-14
Третья команда RX/WX.....	4-14
Возврат к первой команде RX/WX.....	4-14
Регистр сдвига	4-15



Первая команда RX/WX	4-16
Вторая команда RX/WX.....	4-16
Третья команда RX/WX.....	4-16
Глава 5. Modbus RTU. Использование команд RX/WX и MRX/MWX.....	5-1
Работа в качестве ведомого устройства сети.....	5-2
Поддерживаемые функции MODBUS	5-2
Определение адреса MODBUS.....	5-2
Если программное обеспечение требует тип данных и адрес	5-3
Если программное обеспечение требует только адрес	5-6
Работа в качестве ведущего устройства сети: команды RX/WX.....	5-9
Поддерживаемые функции MODBUS	5-9
Память ПЛК поддерживаемая в режиме ведущего	5-10
Пример 1: Вычисление адреса слова ПЛК.....	5-11
Пример 2: Вычисление адреса дискретного входа ПЛК.....	5-11
Создание программы чтения или записи	5-12
Пример программы с командой RX.....	5-14
Блокировки многократного чтения и записи.....	5-14
Работа DL06 ведущим устройством: команды MRX/MWX.....	5-15
Чтение по сети MODBUS (MRX).....	5-15
Адрес памяти ведомого устройства для команды MRX	5-16
Адреса памяти ведущего устройства для команды MRX	5-16
Число элементов в команде MRX.....	5-16
Буфер ошибок выполнения команды MRX.....	5-16
Запись по сети MODBUS (MWX).....	5-17
Адрес памяти ведомого устройства для команды MWX.....	5-18
Адреса памяти ведущего устройства для команды MWX.....	5-18
Число элементов в команде MWX.....	5-18
Буфер ошибок выполнения команды MWX	5-18
Пример использования команд MWX и MRX с блокировкой	5-19
Блокировки многократного чтения и записи	5-19
Глава 6. Использование команд IBox в коммуникационных программах	6-1
Команда конфигурации сети NETCFG.....	6-2
NETCFG	6-2
IB-700.....	6-2
Команда Чтения по сети NETRX	6-3
NETRX IB-701	6-3
Команда Записи по сети NETWX.....	6-4
NETWX IB-702	6-4
Пример использования команд NETCFG, NETRX и NETWX.....	6-5



Глава 1. Введение

В этой главе...

- Введение в руководство
- Кратко о D0-DCM
- Примеры использования D0-DCM

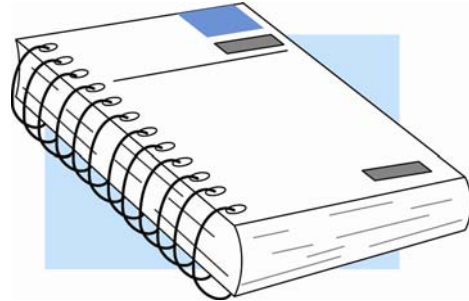
Введение

Цели данного руководства

Данное Руководство подскажет вам, как установить, запустить, запрограммировать и обслуживать коммуникационный модуль D0-DCM контроллеров серии DL05/06.

Это единственное руководство, которое Вам необходимо для использования D0-DCM в качестве дополнительного коммуникационного порта контроллеров DL05/06.

Если Вы планируете использовать D0-DCM в качестве ведущего или ведомого устройства в сети DirectNET, то в этом руководстве приведены основные шаги по настройке D0-DCM и использованию команд RX/WX в программе релейной логики.



Дополнительные руководства

В зависимости от приобретенного оборудования, в дополнение к данному Руководству, Вам понадобятся Руководства с описанием этих продуктов. Как минимум, мы рекомендуем использовать следующие руководства:

- Руководство пользователя контроллера DL05(рус.) или D0–USER–M (англ.)
- Руководство пользователя контроллера DL06(рус.) или D0–06USER–M (англ.).
- Программный пакет DirectSOFT5. Руководство пользователя (рус.).
- Руководство протокола **DirectNET** для DL05/DL06/DL05/06/DL405.

Эти и другие нужные Вам руководства доступны на сайте:
<http://www.plcsystems.ru>

Если Вы планируете использовать D0-DCM в качестве интерфейса к компьютеру или панели оператора. Вам, возможно, понадобятся руководства на эти изделия.

Техническая поддержка

Мы понимаем, что, несмотря на наши усилия, Вы не всегда сможете быстро найти ту информацию, которая Вам необходима. Первым делом воспользуйтесь следующими возможностями, которые облегчат поиск необходимой информации:

Вы также можете использовать наши оперативные возможности для получения информации по поддержке новых продуктов:

- **Интернет** — наш адрес <http://www.plcsystems.ru>;
- **Электронная почта** — info@plcsystems.ru.

Если Вы не найдете и там нужной информации, пожалуйста, позвоните нам по телефону **(495) 925-77-98** или **(495) 490-24-62**. Наша группа технической поддержки будет рада помочь Вам ответить на ваши вопросы. Мы работает с понедельника по пятницу с 9:00 до 18:00 по московскому времени.

Используемые обозначения



Когда Вы видите иконку «лампочка» в левой части страницы, то в примыкающем справа абзаце приводится для вас специальная подсказка. Слово **ПОДСКАЗКА**: при полужирном написании указывает на начало текста.



Когда Вы видите иконку «блокнот» в левой части страницы, то в примыкающем справа абзаце будет специальное примечание. Слово **ПРИМЕЧАНИЕ**: при полужирном написании указывает на начало текста.



Когда Вы видите иконку «восклицательный знак» в левой части страницы, то в примыкающем справа абзаце будет предупреждение. Данная информация поможет Вам предотвратить повреждения, потерю функциональности или даже гибель в экстремальных случаях. Любое предупреждение в этом руководстве должно быть расценено как важная информация, которая должна быть прочитана полностью. Слово **ПРЕДУПРЕЖДЕНИЕ**: при полужирном написании указывает начало текста.

Ключевые темы
в каждой главе

В начале каждой главы приводится список ключевых тем, которые можно найти в данной главе.

Глава 1. Начальные сведения

- В этой главе...
- Введение
 - Используемые обозначения
 - Компоненты системы DL205
 - Методы программирования
 - Система кодировки модулей контроллеров DL20
 - Как быстро проверить ПЛК

Кратко о D0- DCM

Замечания по настройке и требованиям к Фирменному ПО ПЛК

Параметры коммуникационных портов D0-DCM настраивают, используя диалог **DirectSOFT5** (PLC>Setup>D0-DCM), или используя команды релейной логики **DirectSOFT32** (у модуля нет DIP-переключателей).

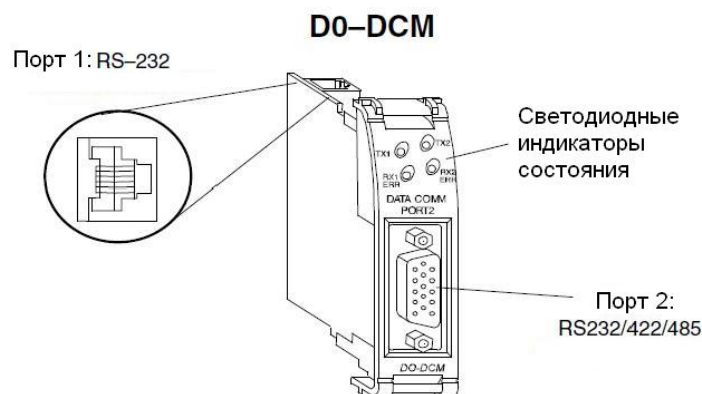
Если настройки по умолчанию 1 и 2 порта приемлемы для приложения, никакой дополнительной настройки не требуется.

Если Вы намереваетесь использовать порт 2 в качестве ведомого устройства, то необходимо настраивать порт. В Главе 3 описаны настройки по умолчанию, регистры V—памяти для хранения настроек и примеры настроек.

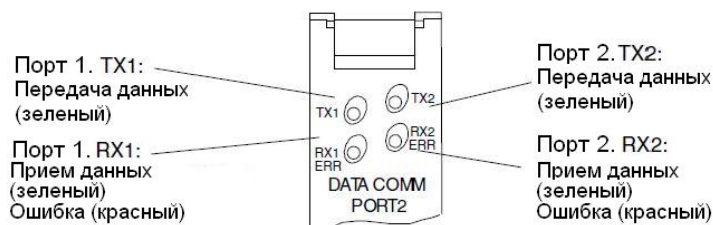
Требованию к версиям Фирменного ПО DL05/06 приведены на стр 2 – 2.

Особенности аппаратной части

На рисунке показаны основные компоненты коммуникационного модуля D0-DCM.



Индикаторы состояния



Использование модуля

Коммуникационный модуль D0-DCM – это коммуникационный интерфейс общего назначения контроллеров DirectLOGIC серии DL05/06. Модуль может быть установлен в любой слот ПЛК. Этот модуль предназначен для применения в качестве:

- Дополнительного коммуникационного порта общего назначения для связи с компьютером или с панелями оператора
- Ведущего или ведомого устройства в сети DirectNET (порт 2) и ведомого устройства в сети DirectNET (порт 1)
- Ведущего или ведомого устройства в сети MODBUS RTU (порт 2) и ведомого устройства в сети MODBUS RTU (порт 1)
- Ведомого устройства, работающего по протоколу K-sequence (оба порта)

Примеры использования D0-DCM

В качестве интерфейса протокола DirectNET

D0-DCM может быть использован в качестве сетевого интерфейса в приложениях, в которых требуется использовать общие данные для нескольких ПЛК или необходим обмен данными с интеллектуальным устройством (компьютером). D0-DCM может быть настроен, как ведущая (master) или ведомая (slave) станция сети и позволяет Вам выгружать и загружать любые данные, включая данные Таймеров/Счетчиков, Входов/Выходов и информацию из V-памяти.



Использование D0-DCM в качестве ведущего устройства сети

D0-DCM может быть использован с ЦПУ DL05 /06, чтобы служить в качестве ведущего устройства сети (Network Master).

Ведущее устройство инициирует запросы к другим станциям сети. Для этого предназначены специальные команды релейной логики: RX/WX, MRX/MWX, NETCFG/NETRX/NETWX.

Порт 2 использует эти команды и автоматически преобразует эти запросы команд в сетевые команды на чтение или запись данных к другим станциям сети. Примеры использования этих команд приведены в Главах 5 и 6.

Возможные ведомые устройства (Slave):

- DL05 / DL06 (любой порт)
- DL05 / DL06 с D0-DCM
- D2-240 / 250-1 / 260 (любой порт)
- D2-240 / 250-1 / 260 с D2-DCM
- D4-330 с DCU
- D4-340/350 (любой порт)
- D4-430/440 (нижний порт)
- D4-450 (нижний порт или RJ12)
- Любой ЦПУ DL405 /с D4-DCM

Использование D0-DCM в качестве ведомого устройства сети

D0-DCM может быть использован с ЦПУ DL05 /06, чтобы служить в качестве ведомой станции сети (Network Slave).

В этом случае D0-DCM «слушает» сеть, чтобы выбрать сообщение, которое содержит адрес D0-DCM.

D0-DCM дешифрует команды, обрабатывает запрос на чтение или запись данных и посылает подтверждение и/или информацию ведущей станции сети.

В этом случае не нужно использовать команды релейной логики. Работу в режиме ведомого устройства поддерживают порты 1 и 2.

Возможные ведущие устройства (Master):

- DL05 / DL06 (любой порт)
- DL05 / DL06 с D0-DCM
- D2- 250-1 / 260 / (нижний порт)
- D2-240 / 250-1 / 260 с D2-DCM
- D4-340/350(нижний порт)
- D4-450 (нижний порт)
- Любой ЦПУ DL405 /с D4-DCM
- Компьютер с DirectSOFT или Ker-Direct PLC

В качестве дополнительного коммуникационного порта

Порты D0-DCM подобны встроенным портам ЦПУ DL05/ DL06. Если Вы можете соединиться с устройством по портам ЦПУ DL05/ DL06, то Вы, также, соединитесь с этим устройством через D0-DCM. Этими устройствами могут быть операторские станции или компьютеры. В этом случае D0-DCM действует как ведомое устройство. Работу в режиме ведомого устройства поддерживают порты 1 и 2.



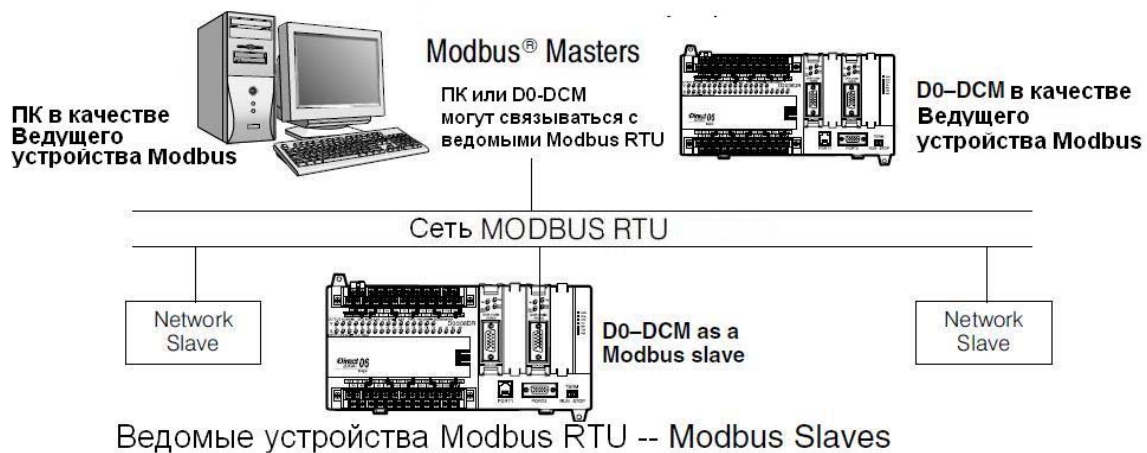
В качестве MODBUS RTU Master/Slave

D0-DCM может быть использован в качестве сетевого интерфейса для связи ПЛК DL05/06 с сетью MODBUS RTU.

Порт 1 может служить только в качестве ведомого устройства MODBUS RTU.

Порт 2 может служить в качестве ведущего и ведомого устройства MODBUS RTU.

Подробное описание протокола содержится в руководстве: Protocol Reference Guide (PI-MBUS-300), размещенном на сайте фирмы Schneider: www.schneiderautomation.com.



Глава 2. Установка, соединения и спецификация

В этой главе...

- Установка D0-DCM в ПЛК
- Изготовление коммуникационного кабеля
- Схемы распайки кабелей
- Спецификация модуля

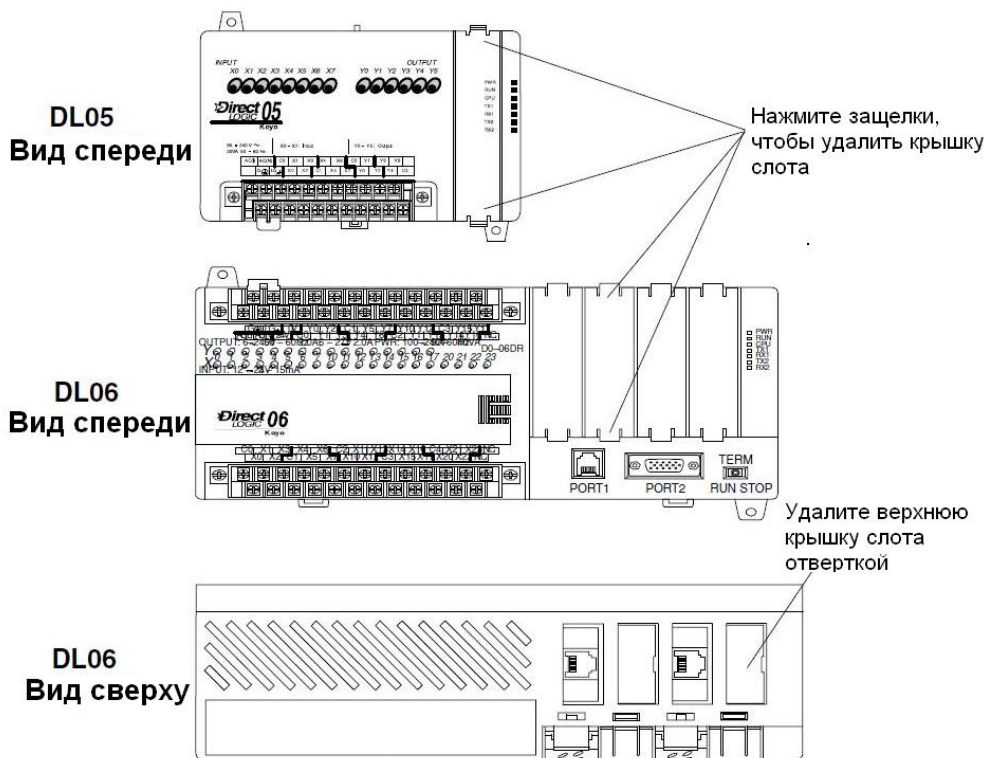
Установка D0-DCM в ПЛК

Установка D0-DCM

D0-DCM может быть установлен в любой слот каркаса DL05/06.

Для установки модуля снимите вверх защитную крышку слота, нажимая на защелки, расположенные сверху и внизу крышки. Для снятия верхней крышки используйте небольшую отвертку.

Убедитесь в том, что питание контроллера отключено, прежде чем устанавливать модуль D0-DCM.



Чтобы установить модуль в каркас направьте печатную плату модуля в прорези в верхней и нижней части каркаса. Вставьте модуль, чтобы лицевая панель модуля была заподлицо с корпусом контроллера, и нажмите фиксирующие защелки на верхней и нижней части лицевой панели модуля.

Проверьте баланс питания контроллера DL06 перед установкой модуля. *Подробнее о балансе питания Вы можете прочитать в Руководстве на контроллер DL06.*

Требования к Фирменному ПО ПЛК и DirectSOFT



ПРИМЕЧАНИЕ: Для работы контроллера DL05 с D0-DCM необходимо использовать **DirectSOFT32 Version 3.0c (или новее)** и Фирменное ПО версии 5.00. (или новее). Для работы контроллера DL06 с D0-DCM необходимо использовать **DirectSOFT32 Version 4.0, build 16 (или новее)** и Фирменное ПО версии 1.90. (или новее). Для загрузки обновлений обращайтесь на сайт: www.automationdirect.com.

Создание коммуникационного кабеля

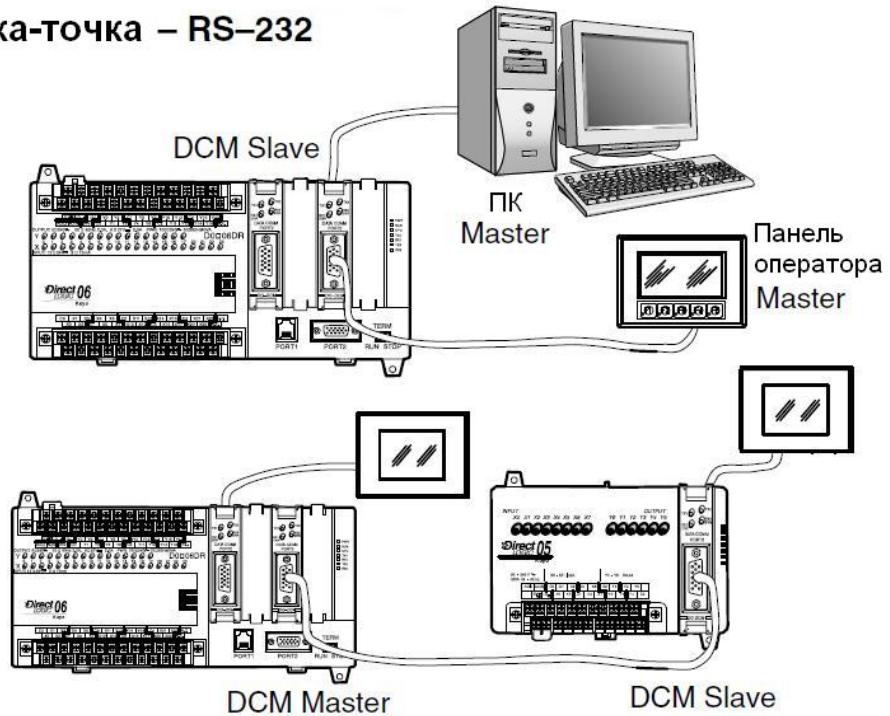
Есть несколько соображений, которые помогут Вам определить тип кабеля для Вашего варианта использования D0-DCM.

Соображение 1: Физическая конфигурация соединения

D0-DCM может быть использован в соединениях типа точка-точка или многоточка. В соединении типа точка-точка участвуют только две станции, ведущая и ведомая. Используйте это соединение для связи одного D0-DCM с компьютером, панелью оператора или интеллектуальным устройством.

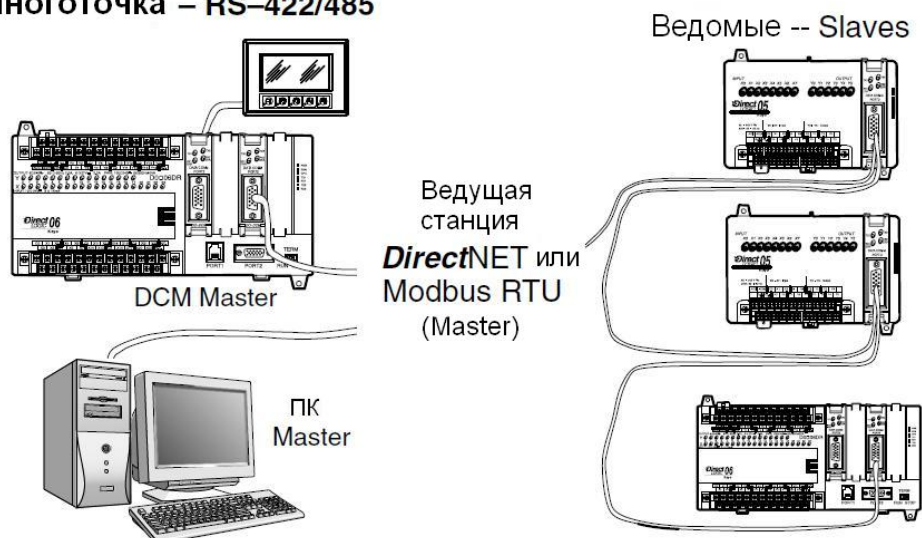
Вы должны, также, использовать эту конфигурацию, когда Вы хотите связать одну ведущую станцию DirectNET (или Modbus RTU) с одной ведомой станцией DirectNET (или Modbus RTU).

Точка-точка – RS-232



Используйте соединение многоточка для связи одного ведущего устройства с двумя или более ведомыми устройствами.

Многоточка – RS-422/485



**Соображение 2:
Интерфейс
RS22C или
RS422**

D0-DCM может поддерживать соединения по интерфейсам R-S232 (порты 1 и 2) или RS-422/485 (порт 2). Если Вы используете многоточечное соединение, Вы должны использовать RS-422 или RS-485. Если Вы используете соединение точка-точка, у Вас есть выбор между RS-232 или RS-422/485. Вы можете использовать RS-232, если длина кабеля менее 15 метров и кабель не подвержен влиянию электрических помех.

Вы должны использовать RS-422/485 для всех других приложений. Интерфейс RS422/485 позволяет использовать кабель длиной до 1000 метров и обеспечивает большую невосприимчивость к электрическим шумам.

**Соображение 3:
Схема кабеля**

Хотя конфигурация сети и электрические характеристики соединения важны, но также важно знать тип устройства, с которым будет связан D0-DCM. Корректная схема кабеля зависит от комбинации всех трех факторов.

На рисунках внизу показаны назначения контактов порта D0-DCM.



Порт 1. Описание контактов	
1	0V
2	5V
3	RXD Receive Data (RS-232)
4	TXD Transmit Data (RS-232)
5	RTS Request to Send
6	0V

Порт 2. Описание контактов	
1	5V 5 VDC
2	TXD2 Transmit Data (RS-232)
3	RXD2 Receive Data (RS-232)
4	RTS2 Ready to Send (RS-232C)
5	CTS2 Clear to Send (RS-232)
6	RXD2- Receive Data - (RS-422/485)
7	0V Logic Ground
8	0V Logic Ground
9	TXD2+ Transmit Data + (RS-422/485)
10	TXD2 - Transmit Data - (RS-422/485)
11	RTS2 + Request to Send + (RS-422/485)
12	RTS2 - Request to Send - (RS-422/485)
13	RXD2 + Receive Data + (RS-422/485)
14	CTS2 + Clear to Send + (RS422/485)
15	CTS2 - Clear to Send - (RS-422/485)

**Соображение 4:
Характеристики
кабеля**

Несмотря на то, что множество типов кабеля могут работать в Вашем приложении, мы рекомендуем использовать кабель, сконструированный с высокой степенью защиты от электрических шумов.

Следующие характеристики могут быть использованы, как рекомендация:

Структура.....	Экранированный, витая пара (RS232 использует два провода и землю).
Сечение провода.....	0.25 мм кв. и больше
Изоляция.....	Полиэтилен
Экран.....	Медная сетка или алюминиевая фольга
Полное сопротивление (импеданс).....	100 Ом @ 1МГц
Емкость.....	60 пФ / метр или меньше.

**Соображение 5:
Рекомендации
по монтажу**

В Вашей компании могут быть собственные правила прокладки кабелей и монтажа соединений. Но перед началом работ по монтажу мы рекомендуем рассмотреть несколько основных требований.

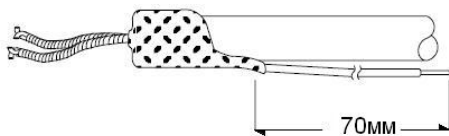
- Не прокладывать кабель вблизи больших электродвигателей, мощных переключателей или трансформаторов. Могут возникнуть проблемы с шумами.
- Прокладывайте кабель в соответствующих кабельных каналах, трубах и т.п., чтобы минимизировать риск повреждения кабеля.
- Рассмотрите возможность резервирования кабеля, если Ваше приложение требует повышенной надежности. Можно будет быстро переключить все станции на резервный кабель, в то время, когда будут восстанавливаться основной кабель.

**Заземление
экрана кабеля**

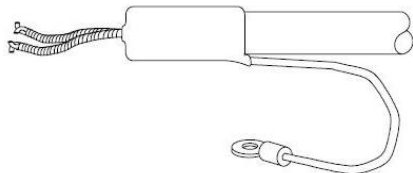
Важно заземлить экран кабеля, чтобы минимизировать влияние шумов. Предпочтительный метод – это присоединение одного конца экрана кабеля к корпусу разъема. Если проблемы с шумом сохраняются, Вы должны обеспечить хорошее заземление шкафа и подключить экран к земле шкафа.

Не заземляйте оба конца экрана, потому что это приведет к наводке шума в кабеле.

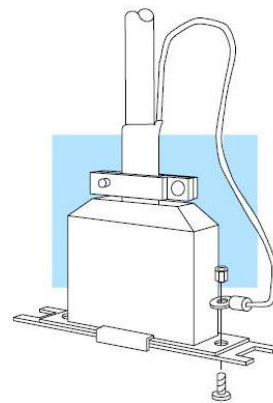
Шаг 1: Оголить около 70 мм экрана кабеля.



Шаг 2: Обжать (припаять) наконечник.



Шаг 3: Прикрепить экран к корпусу разъема.

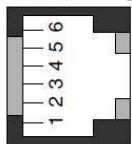


Схемы распайки кабелей

Соединение по RS-232

Соединение по RS-232

D0-DCM Порт 1

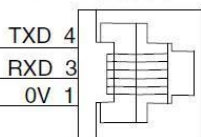


Гнездо RJ12
6P6C

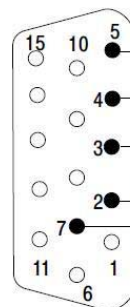


TXD
RXD
0V

D0-DCM Порт 1



D0-DCM Порт 2



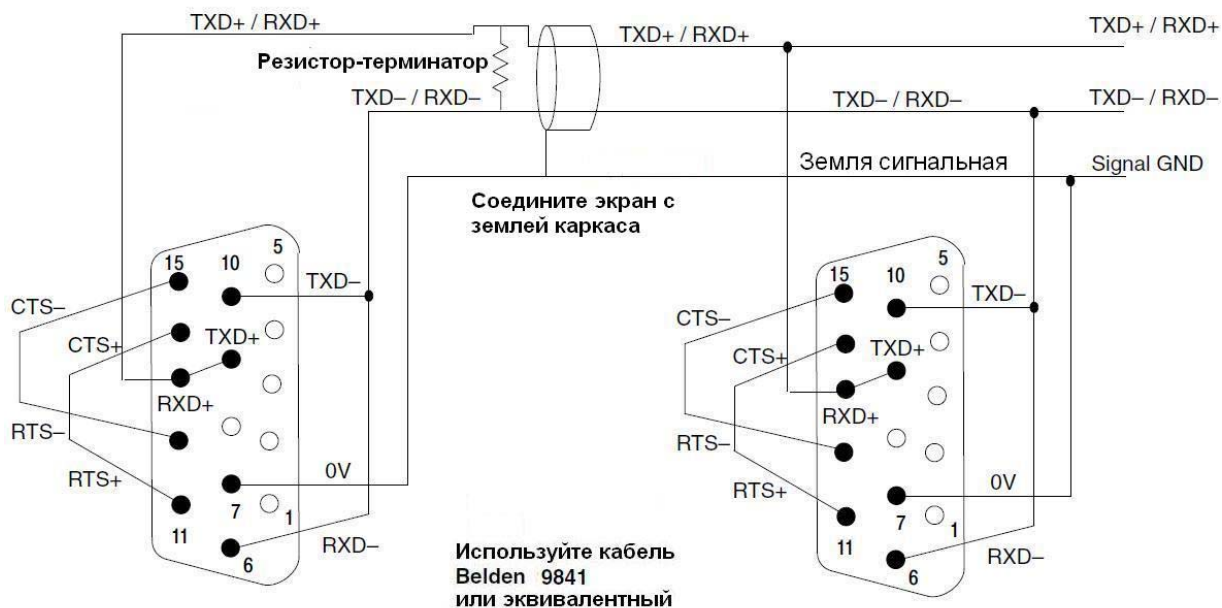
Переключатель или
использование
сигналов RTS/CTS

RTS
CTS

5 CTS → RTS
4 RTS → CTS
3 RXD → TXD
2 TXD → RXD
7 Signal GND → GND

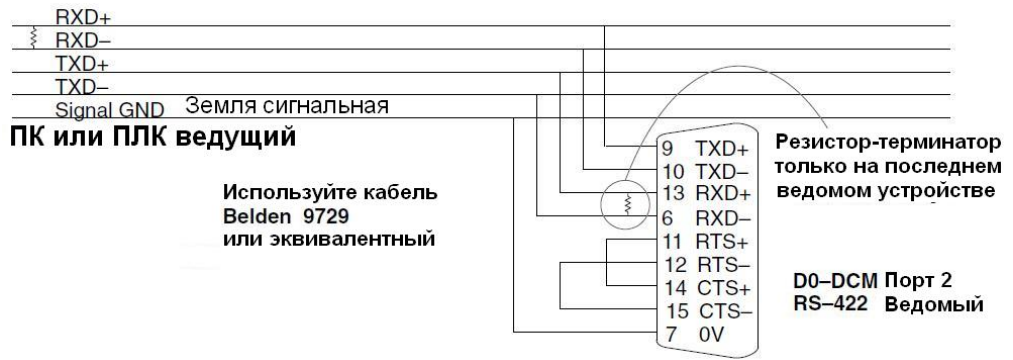
Соединение по RS-485

Сеть RS-485



Соединение по RS-422

Сеть RS-422



Резисторы-терминаторы для многоточечных соединений RS422/485.

Необходимо использовать резисторы-терминаторы на каждом конце линии связи RS422/485. Применение этих резисторов поможет уменьшить количество ошибок в передаче данных. Вы должны подобрать номинал резисторов в соответствии с импедансом используемого кабеля.

Например, для типового кабеля с монолитными проводами сечением 0.35 мм кв. и скрученными на 4.5 оборота/фут (14 оборотов/метр) полное сопротивление составляет 120 Ом.

Существует два способа подключения резисторов-терминаторов:

- Линия-Линия – этот способ позволяет сбалансировать линии получения данных (IN+ и IN-) при помощи одного резистора на каждом конце линии. Ниже приведена схема такого соединения.
- Линия-Земля - этот способ, также, позволяет сбалансировать линии получения данных, при этом уровень подавления помехи общего вида увеличивается значительно. В этом способе используют по два резистора на каждом конце линии. Так как резистора два, их сумма должна соответствовать полному сопротивлению кабеля.

Спецификация модуля

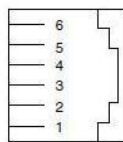
Общие характеристики

Требования по питанию	250 мА @ =5В (без учета питания внешних устройств)
Максимальное число модулей	DL05: 1; DL06: 4;
Рабочая температура	От 0° до 55° С
Температура хранения	От -20° до 80° С
Рабочая влажность	От 5 до 95% (без конденсации)
Состав воздуха	Без корродирующих газов
Вибрация	MIL STD 810C, Method 514.2
Ударопрочность	MIL STD 810C, Method 516.2
Напряжение, выдерживаемое изоляцией	~1000 V, 1 минуту
Сопrotивление изоляции	10МОм при = 500 V
Устойчивость к помехам (шумам)	NEMA ICS3-304, UL, CE, (FCC Class A) Class 1, Division 2 (C1D2)
Вес	50 грамм

Порт 1

Разъем	Телефонный, RJ12, 6P6C
Интерфейс	RS232
Протокол (автоматическое определение)	DirectNET ведомый, K-sequence ведомый, Modbus RTU ведомый.
Номер станции	0 - 247
Передаваемые данные	8 бит данных, 1 стартовый бит, 1 стоповый бит (фиксировано)
Бит контроля четности	Нет контроля (None), Нечет (Odd)
Скорость передачи	9600, 19200, 38400, 57600, 115200 бит в секунду
Режим передачи	ASCII, HEX
Максимальное расстояние	RS232: 15 метров

Порт 1



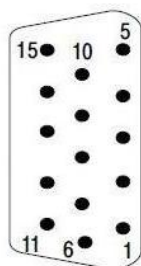
RJ12

Порт 1. Описание контактов	
1	0V
2	5V
3	RXD Receive Data (RS-232)
4	TXD Transmit Data (RS-232)
5	RTS Request to Send
6	0V

Порт 2

Разъем	15-ти контактный VGA (мама)
Интерфейс	RS232 RS422/485
Протокол (автоматическое определение)	DirectNET ведущий/ведомый, K-sequence ведомый, Modbus RTU ведущий/ведомый.
	ASCII IN/OUT (Non-sequence)
Номер станции	0 - 247
Передаваемые данные	8/7 бит данных, 1 стартовый бит, 1/2 стоповый бит
Бит контроля четности	Нет контроля (None), Нечет (Odd), Чет (Even)
Скорость передачи	300, 600, 1200, 4800, 9600, 19200, 38400, 57600, 115200 бит в секунду
Режим передачи	ASCII, HEX
Задержки передачи (таймаут)	Базовое время x (1 – 50)
Время задержки ответа	0/2/5/10/20/50/100/500 миллисекунд
Задержка символа	0 – 9999 миллисекунд
Максимальное расстояние	RS232: 15 метров RS422/485: 1000 метров

Порт 2



15-ти контактный разъем типа VGA (мама)

Порт 2. Описание контактов		
1	5V	5 VDC
2	TXD2	Transmit Data (RS-232)
3	RXD2	Receive Data (RS-232)
4	RTS2	Ready to Send (RS-232C)
5	CTS2	Clear to Send (RS-232)
6	RXD2-	Receive Data - (RS-422/485)
7	0V	Logic Ground
8	0V	Logic Ground
9	TXD2+	Transmit Data + (RS-422/485)
10	TXD2 -	Transmit Data - (RS-422/485)
11	RTS2 +	Request to Send + (RS-422/485)
12	RTS2 -	Request to Send - (RS-422/485)
13	RXD2 +	Receive Data + (RS-422/485)
14	CTS2 +	Clear to Send + (RS422/485)
15	CTS2 -	Clear to Send - (RS-422/485)

Индикаторы состояния

Индикатор	Состояние	Описание
TX1 (Зеленый)	ON (Вкл)	Порт 1 передает данные
RX1 (Зеленый)	ON (Вкл)	Порт 1 получает данные
ERR (Красный)	ON (Вкл)	Порт 1: Тайм-аут, NAK или аварийный ответ
TX2 (Зеленый)	ON (Вкл)	Порт 2 передает данные
RX2 (Зеленый)	ON (Вкл)	Порт 2 получает данные
ERR (Красный)	ON (Вкл)	Порт 2: Тайм-аут, NAK или аварийный ответ (Exception Response)

Глава 3. Настройка D0-DCM

В этой главе...

- Информация по настройке модуля
- Использование **DirectSOFT5** для настройки портов D0-DCM
- Регистры настроек портов 1 и 2 D0-DCM
- Использование RLL для настройки портов D0-DCM (DL05)
- Использование RLL для настройки портов D0-DCM (DL06)
-

Информация по настройке модуля D0-DCM

Параметры коммуникационных портов D0-DCM можно настроить при помощи диалога настроек **DirectSOFT5** (PLC>Setup>D0-DCM) или используя команды релейной логики в **DirectSOFT32**. Если настройки портов 1 и 2 по умолчанию подходят для Вашего приложения, никаких действий производить не надо. Но если Вы хотите использовать порт 2 в качестве ведущего, настройка порта необходима.



ПРИМЕЧАНИЕ: Для работы контроллера DL05 с D0-DCM необходимо использовать **DirectSOFT32 Version 3.0c** (или новее) и Фирменное ПО версии **5.00**. (или новее). Для работы контроллера DL06 с D0-DCM необходимо использовать **DirectSOFT32 Version 4.0, build 16** (или новее) и Фирменное ПО версии **1.90**. (или новее). Для загрузки обновлений обращайтесь на сайт: www.automationdirect.com.

Информация для пользователей DirectSOFT5

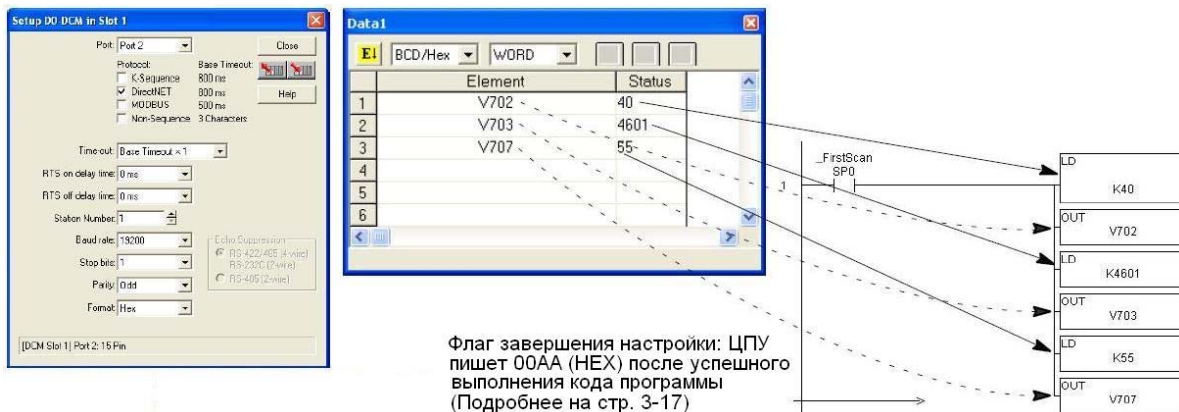


Если Вы хотите использовать команды релейной логики для настройки D0-DCM, Вы можете использовать диалог настроек (D0-DCM Setup) и окно Data View для определения BCD/HEX значений настроек. Это значительно упростит Вашу работу. Этот способ показан на рисунках внизу и описан далее в этой главе.

Шаг 1: Откройте диалог настроек **DirectSOFT5** (PLC>Setup>D0-DCM), установите нужные настройки и сохраните их в ЦПУ.

Шаг 2: В **DirectSOFT5** откройте окно Data View и наберите адреса специальных ячеек V-памяти предназначенных для хранения настроек D0-DCM (см. стр. 3-9). Значения настроек в формате BCD/HEX появятся в окне Data View.

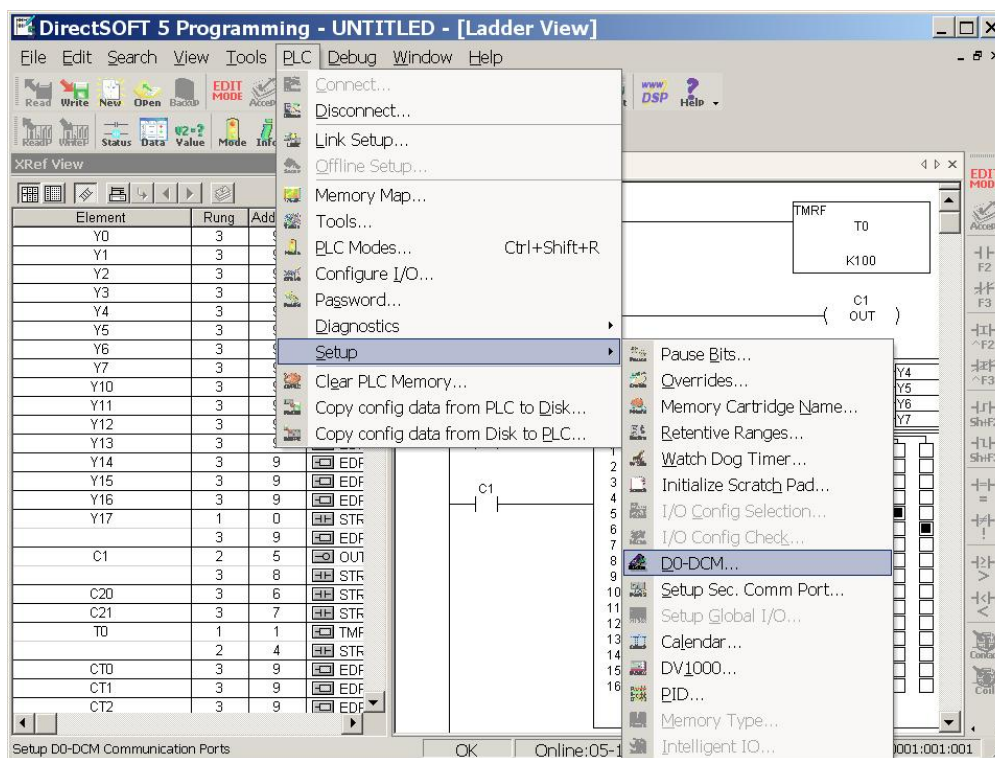
Шаг 3: Создайте цепь для записи значений настроек в специальные ячейки V-памяти предназначенные для D0-DCM. Настройки будут действовать и восстанавливаться после перехода из режима PGM в RUN, выключении/включении питания без отключения связи с портами. Пример такой программы приведен в конце главы.



Использование *DirectSOFT5* для настройки D0-DCM

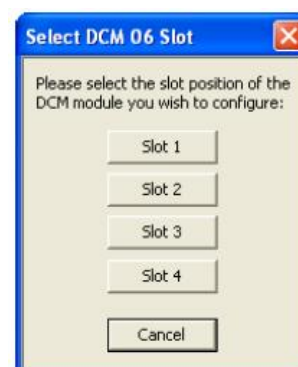
DirectSOFT5:
PLC>Menu>Setup

Использование диалога настроек D0-DCM в *DirectSOFT5* - это самый легкий способ настройки коммуникационных параметров портов D0-DCM. D0-DCM должен быть установлен в слот контроллера, контроллер должен быть запитан и соединен с *DirectSOFT5* по порту 1, 2 или по Ethernet (с помощью модуля H0-ECOM). После соединения выберите PLC>Setup>D0-DCM.



Выбор слота D0-DCM

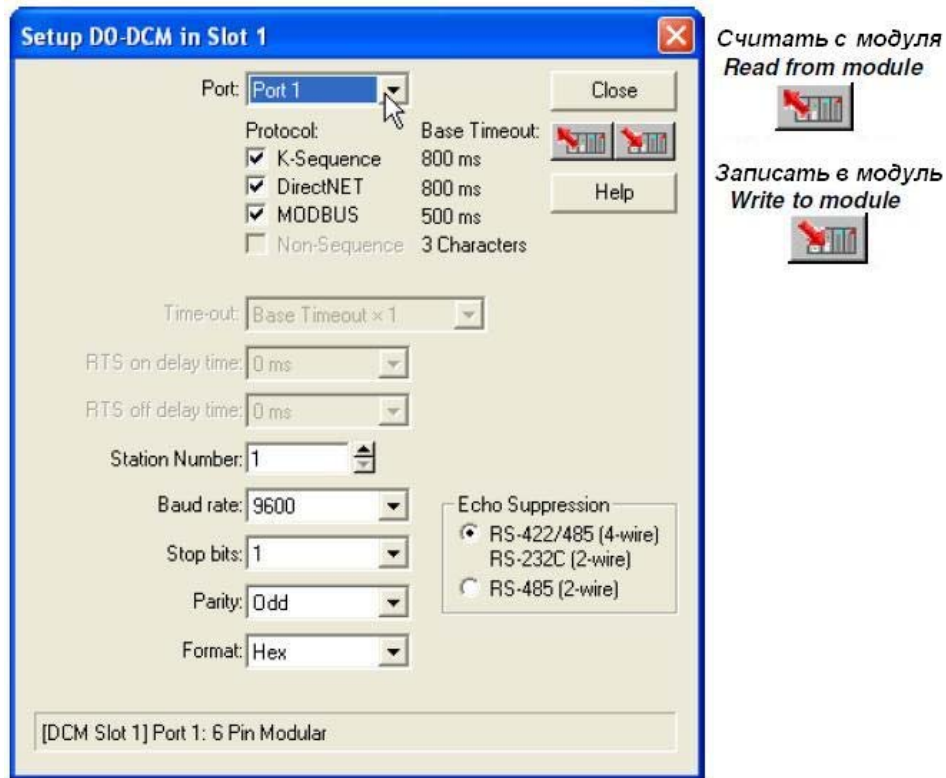
Контроллер DL06 поддерживает работу нескольких модулей D0-DCM и модуль D0-DCM можно установить в любой из четырех слотов, поэтому Вам необходимо выбрать номер слота в окне настроек, показанном справа. После этого будет открыто соответствующее окно настроек, показанное на следующей странице.



Настройка порта 1 (ведомый)

Окно настроек D0-DCM позволяет Вам просматривать и редактировать коммуникационные параметры портов D0-DCM.

- **Порт (Port):** Из списка номеров портов в верхней части окна выберите «Port 1».
- **Протокол (Protocol):** По умолчанию порт работает в режиме автоматического определения протокола обмена (auto-detect) и помечены все протоколы (за исключением Non-sequence). Порт автоматически определяет используемый протокол и действует соответствующим образом. Выбор одного из протоколов не дает никаких преимуществ



- **Номер станции (Station Number):** Допустимый диапазон номеров станций для ведомых устройств DirectNET — от 1 до 90. Допустимый диапазон номеров станций для ведомых устройств MODBUS — от 1 до 247. Каждое ведомое устройство должно иметь уникальный номер.
- **Скорость передачи (Baud Rate):** Доступны скорости передачи от 9600 до 115 200 бод. Сначала выберите наибольшую скорость передачи, и снижайте скорость, если при опытной передаче возникают ошибки в данных или помехи в линии. Вы должны установить одинаковую скорость передачи данных для всех устройств сети.
- **Стоповые биты (Stop Bits):** Для использования в протоколе выберите 1 или 2 стоповых бита.
- **Контроль четности (Parity):** Для контроля ошибок выберите «нет», «четное» или «нечетное».
- **Формат:** Выберите Hex или ASCII
- **Подавление эхо-сигнала (Echo Suppression):** Применимо только для протокола Modbus.

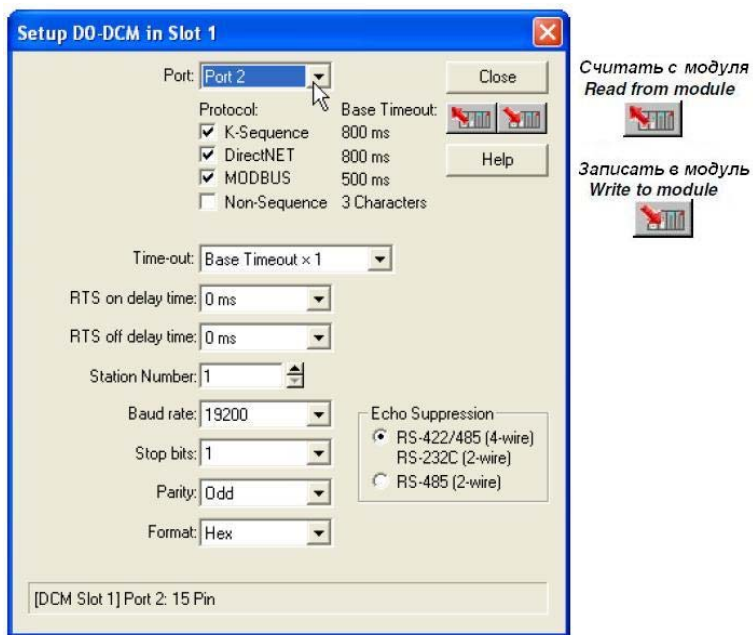


Затем нажмите клавишу, указывающую на посылку конфигурации в модуль и нажмите «Close» - «Заккрыть».

Настройка порта 2 (ведомый)

Щелкните по «Port 2» для просмотра и редактирования коммуникационные параметры порта 2 D0-DCM.

- **Порт (Port):** Из списка номеров портов выберите «Port 2».
- **Протокол (Protocol):** По умолчанию порт работает в режиме автоматического определения протокола обмена (auto-detect) и помечены все протоколы (за исключением Non-sequence). Порт автоматически определяет используемый протокол и действует соответствующим образом. Выбор одного из протоколов не дает никаких преимуществ. Выбор одного протокола необходим при использовании порта в режиме ведущего (master mode).



- **Задержка (Timeout):** Интервал времени, в течение которого, порт будет ожидать ответа после отправки сообщения без формирования ошибки.
- **Задержка (RTS On Delay Time):** Интервал времени между подачей сигнала RTS и отправкой данных.
- **Задержка (RTS Off Delay Time):** Интервал времени между сбросом сигнала RTS после отправки данных.
- **Номер станции (Station Number):** Допустимый диапазон номеров станций для ведомых устройств DirectNET — от 1 до 90. Допустимый диапазон номеров станций для ведомых устройств MODBUS — от 1 до 247. Каждое ведомое устройство должно иметь уникальный номер.
- **Скорость передачи (Baud Rate):** Доступны скорости передачи от 300 до 115 200 бод. Сначала выберите наибольшую скорость передачи, и снижайте скорость, если при передаче возникают ошибки в данных или проблемы с помехами в линии. Вы должны установить одинаковую скорость передачи данных для всех устройств сети.
- **Стоповые биты (Stop Bits):** Для использования в протоколе выберите 1 или 2 стоповых бита.
- **Контроль четности (Parity):** Для контроля ошибок выберите «нет», «четное» или «нечетное».
- **Формат:** Выберите Hex или ASCII
- **Подавление эхо-сигнала (Echo Suppression):** Выберите в соответствии с подключением. Применимо только для протокола Modbus.

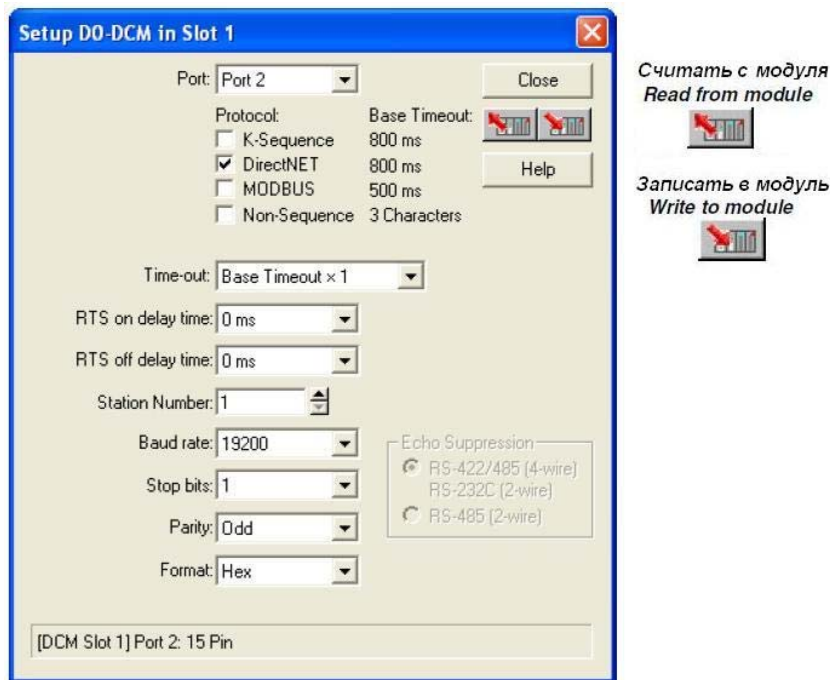


Затем нажмите клавишу, указывающую на отправку конфигурации в модуль и нажмите «Close» - «Закреть».

Настройка порта 2 (ведущий DirectNET)

Чтобы настроить порт 2 для работы в качестве ведущей станции (master mode) DirectNET, надо установить следующие настройки:

- **Порт (Port):** Из списка номеров портов выберите «Port 2».
- **Протокол (Protocol):** Сделайте пометку в поле слева от протокола DirectNET.



- **Задержка (Timeout):** Интервал времени, в течение которого, порт будет ожидать ответа после отправки сообщения без формирования ошибки.
- **Задержка (RTS On Delay Time):** Интервал времени между подачей сигнала RTS и отправкой данных.
- **Задержка (RTS Off Delay Time):** Интервал времени между окончанием отправки данных и сбросом сигнала RTS.
- **Номер станции (Station Number):** Чтобы назначить порт DCM ведущим, выберите «1». Допустимый диапазон номеров станций для ведомых устройств DirectNET — от 1 до 90. При включении питания и тогда, когда ЦПУ не выполняет команды Чтения/записи по сети (RX/WX), порт работает в режиме ведомого устройства.
- **Скорость передачи (Baud Rate):** Доступны скорости передачи от 300 до 115 200 бод. Сначала выберите наибольшую скорость передачи, и снижайте скорость, если при опытной передаче возникают ошибки в данных или помехи в линии. Вы должны установить одинаковую скорость передачи данных для всех устройств сети.
- **Стоповые биты (Stop Bits):** Для использования в протоколе выберите 1 или 2 стоповых бита.
- **Контроль четности (Parity):** Для контроля ошибок выберите «нет», «четное» или «нечетное».
- **Формат:** Выберите Hex или ASCII

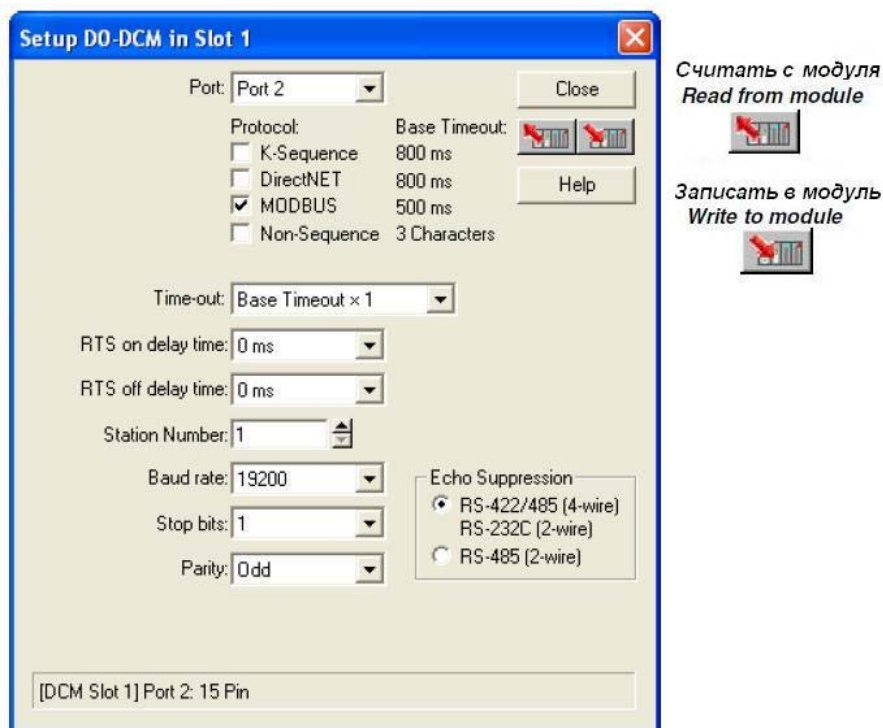


Затем нажмите клавишу, указывающую на отправку конфигурации в модуль и нажмите «Close» - «Закрыть».

Настройка порта 2 (ведущий Modbus)

Чтобы настроить Порт 2 для работы в качестве ведущей станции Modbus RTU, надо установить следующие настройки:

- **Порт (Port):** Из списка номеров портов выберите «Port 2».
- **Протокол (Protocol):** Сделайте пометку в поле слева от «Modbus».



- **Задержка (Timeout):** Интервал времени, в течение которого, порт будет ожидать ответа после отправки сообщения без формирования ошибки.
- **Задержка (RTS On Delay Time):** Интервал времени между подачей сигнала RTS и отправкой данных.
- **Задержка (RTS Off Delay Time):** Интервал времени между окончанием отправки данных и сбросом сигнала RTS.
- **Номер станции (Station Number):** Чтобы назначить порт DCM ведущим, выберите «1». Допустимый диапазон номеров станций для ведомых устройств Modbus — от 1 до 247 при использовании команд MRX/MWX и от 1 до 90 при использовании команд RX/WX. При включении питания и тогда, когда ЦПУ не выполняет команды Чтения/записи по сети (MRX/MWX, RX/WX), порт работает в режиме ведомого устройства..
- **Скорость передачи (Baud Rate):** Доступны скорости передачи от 300 до 115 200 бод. Сначала выберите наибольшую скорость передачи, и снижайте скорость, если при опытной передаче возникают ошибки в данных или помехи в линии. Вы должны установить одинаковую скорость передачи данных для всех устройств сети.
- **Стоповые биты (Stop Bits):** Для использования в протоколе выберите 1 или 2 стоповых бита.
- **Контроль четности (Parity):** Для контроля ошибок выберите «нет», «четное» или «нечетное».
- **Подавление эхо-сигнала (Echo Suppression):** Выберите в соответствии с подключением (2-х проводное или 4х проводное).

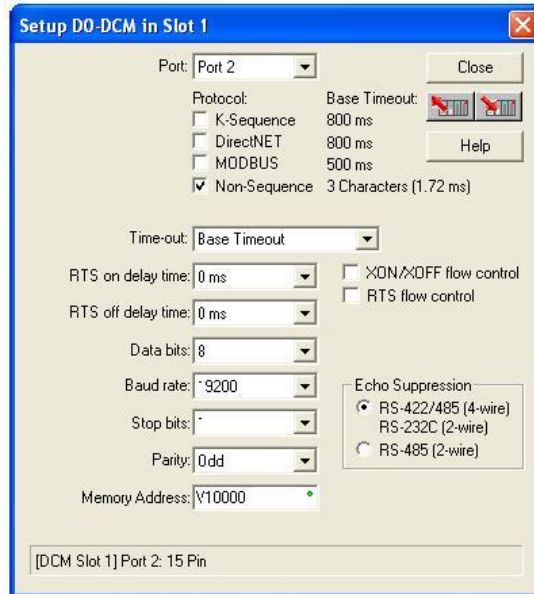


Затем нажмите клавишу, указывающую на отправку конфигурации в модуль и нажмите «Close» - «Закрыть».

Настройка порта 2 (Non-Sequence)

Настройка порта 2 для работы по протоколу «Non-Sequence» позволяет ЦПУ использовать порт D0-DCM для чтения/записи необработанных строк ASCII символов с помощью ASCII-команд. (В главе 5 Руководства пользователя DL05 или DL06 приведены описания этих команд). В **DirectSOFT5** выберите PLC>Setup>D0-DCM. Далее сделайте следующие настройки:

- **Порт (Port):** Из списка номеров портов выберите «Port 2».
- **Протокол (Protocol):** Сделайте пометку в поле слева от «Non-Sequence»



Считать с модуля
Read from module



Записать в модуль
Write to module



- **Задержка (Timeout):** Интервал времени, в течение которого, порт будет ожидать ответа после отправки сообщения без формирования ошибки.
- **Задержка (RTS On Delay Time):** Интервал времени между подачей сигнала RTS и отправкой данных.
- **Задержка (RTS Off Delay Time):** Интервал времени между сбросом сигнала RTS после отправки данных.
- **Биты данных (Data Bits):** Выберите 7 или 8 бит в соответствии с форматом принимаемых данных.
- **Скорость передачи (Baud Rate):** Доступны скорости передачи от 300 до 115 200 бод. Сначала выберите наибольшую скорость передачи, и снижайте скорость, если при опытной передаче возникают ошибки в данных или помехи в линии. Вы должны установить одинаковую скорость передачи данных для всех устройств сети.
- **Стоповые биты (Stop Bits):** Для использования в протоколе выберите 1 или 2 стоповых бита в соответствии с настройками подсоединенного устройства.
- **Контроль четности (Parity):** Для контроля ошибок выберите «нет», «четное» или «нечетное» в соответствии с настройками подсоединенного устройства..
- **Адрес памяти (Memory Address):** Укажите начальный адрес V- памяти для размещения настроек порта приведенных ниже.
- **Управление передачей (Xon/Xoff Flow Control):** Сделайте отметку в этом поле, если Вы используете аппаратное управление передачей сигналами RTS и CTS.
- **Управление передачей (RTS Flow Control):** Сделайте отметку в этом поле, если Вы используете управление передачей сигналом RTS.
- **Подавление эхо-сигнала (Echo Suppression):** Выберите в соответствии с подключением .



Затем нажмите клавишу, указывающую на отправку конфигурации в модуль и нажмите «Close» - «Заккрыть».

Регистры настроек D0-DCM

Регистры настроек D0-DCM

В таблице приведен список специальных ячеек V-памяти, используемых контроллерами DL05/DL06 для настройки D0-DCM, привязанных к слотам. Эти регистры используются вне зависимости от способа настройки модуля (Setup DS5 или программная). Далее будут описаны назначения всех регистров и приведены примеры программ релейной логики для DL05 и DL06.

Параметры настроек модуля		Слоты DL05 и DL06					
		Смещение слов	DL05	DL06 Слот 1	DL06 Слот 2	DL06 Слот 3	DL06 Слот 4
A	Порт 1 –Режим передачи (ASCII/HEX), протокол	+0000	V7700	V700	V710	V720	V730
B	Порт 1 – Адрес станции, Скорость, Четность	+0001	V7701	V701	V711	V721	V731
C	Порт 2 - Режим передачи (ASCII/HEX), протокол, Задержка (Time-out), Режим RS-485	+0002	V7702	V702	V712	V722	V732
D	Порт 2 - Адрес станции, Скорость, Биты данных, Стоповые биты, Четность	+0003	V7703	V703	V713	V723	V733
E	Порт 2 – Задержки символов	+0006	V7706	V706	V716	V726	V736
F	Порт 1/ Порт 2 – Код завершения настройки	+0007	V7707	V707	V717	V727	V737
G	Порт 1/ Порт 2 – Задержка перезапуска (Reset time-out)		V7730	V7731	V7731	V7732	V7733

Параметры настроек по умолчанию

При включении питания D0-DCM запишет необходимую информацию в регистры настроек V-памяти, соответствующую настройкам по умолчанию.

Если Вы хотите изменить эти настройки, Вы должны использовать диалог **DirectSOFT5** (PLC>Setup>D0–DCM) или, если Вы пользователь **DirectSOFT32**, то настройку необходимо производить при помощи команд релейной логики.

Настройки портов по умолчанию		
Параметр	Порт 1	Порт 2
Режим	Ведомый (Slave)	
Скорость	9600 бит/с	19200 бит/с
Четность	Нечет (Odd)	
Протокол	K-sequence/DirectNet/Modbus (Auto-detect)	
Адрес станции	1	
Биты данных	8(фиксировано)	8
Стоповые биты	1(фиксировано)	1

Описание параметров настроек

Protocol Selection: Выбор протокола. В режиме ведомого (Slave mode) порты 1 и 2 работают в режиме автоматического определения протокола обмена (помечены все протоколы, за исключением Non-sequence для порта 2). Порт автоматически определяет используемый протокол и действует соответствующим образом. Эта настройка удобна при программировании ЦПУ или при соединении с панелями оператора. Вы можете выбрать один протокол, но выбор одного из протоколов не дает никаких преимуществ, в этом случае.

В режиме ведущего (Master mode) порты 1 и 2 могут работать в качестве ведущих станций **DirectNET** или Modbus.

При использовании порта 2 в качестве ведущего, Вы должны выбрать один протокол, который будет применяться при посылке запросов ведомым станциям, и назначить адрес станции «1». При включении питания и тогда, когда ЦПУ не выполняет команды Чтения/записи по сети (MRX/MWX, RX/WX), порт работает в режиме ведомого устройства.

Communication Timeout: Задержки передачи. Эта настройка обычно используется только если Вы создаете собственную программу обмена по протоколу **DirectNET**. При отключении функции таймаута, Вы можете послать один компонент **DirectNET** без проблем с таймаутом. При возникновении ошибок передачи, в этом случае, Вы должны перезапустить передачу, посылая команду на повторение или отменить передачу командой EOT (End of Transmission). Подробности можно узнать в руководстве на протокол **DirectNET**.

Transmit Mode: Режим передачи. Выберите между режимом ASCII или шестнадцатиричным режимом (ASCII / HEX Mode): Если Вам важна скорость передачи – выбирайте формат HEX (настройка по умолчанию). Те же самые данные в формате ASCII занимают в два раза больше бит, поэтому передаются дольше. Если устройство требует использования формата ASCII, установите переключатель в положение ASCII, в других случаях используйте формат HEX.

Baud Rate: Скорость передачи. Настройка скорости передачи возможна в диапазоне от 300 до 115 200 бит в секунду. Все станции в сети должны быть настроены на работу с одной и той же скоростью, чтобы передача осуществлялась корректно. Обычно, используют самую высокую скорость возможную без возникновения ошибок. При наличии проблем с помехами, снижайте скорость передачи.

Parity: Контроль четности. Выберите один из вариантов контроля ошибок – Нет контроля/ Нет чет /Чет (None/ODD/EVEN).

RTS Delay Time: Время задержки сигнала RTS.

RTS On Delay - задержка посылки данных после установки сигнала RTS. Типичная настройка –«0» и ее необходимо изменять только при использовании D0-DCM с радио модемом. Необходимое время задержки можно найти в документации на используемый модем.

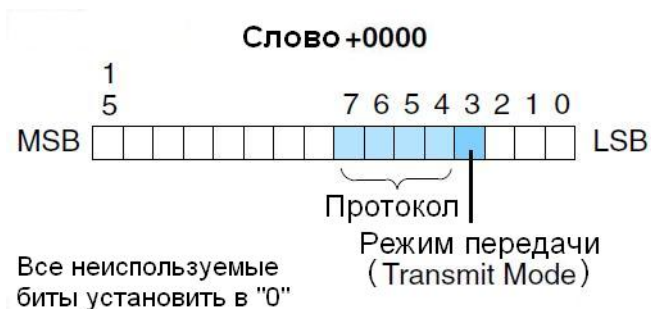
RTS Off Delay – задержка сброса сигнала RTS. Интервал времени между окончанием посылки данных и сбросом сигнала RTS.

Station Address: Адрес станции. Децимальные адреса не должны быть обязательно назначаемы в последовательном порядке, но каждая станция должна иметь уникальный номер. Смотри описание протоколов порта 2, приведенное выше.

А: Порт 1 - Режим передачи, протокол

Используйте «Слово+0» для настройки порта 1:

- K-sequence Slave, **DirectNET** Slave, MODBUS RTU Slave или все три протокола в режиме автоопределения.
- Режим передачи (transmit mode) ASCII или HEX



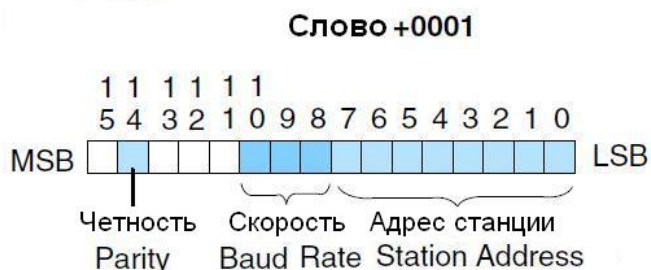
Порт 1: Режим передачи	
<i>Режим</i>	<i>Бит 3</i>
HEX	0
ASCII	1

Порт 1: Протокол					
<i>Протокол</i>	<i>Биты 7 – 4 (HEX)</i>	<i>Бит 7</i>	<i>Бит 6</i>	<i>Бит 5</i>	<i>Бит 4</i>
K-sequence	8	1	0	0	0
DirectNET	4	0	1	0	0
MODBUS RTU	2	0	0	1	0
K-sequence MODBUS RTU DirectNET	E	1	1	1	0

В: Порт 1 – Адрес станции, скорость, четность

Используйте «Слово+1» для этих настроек порта 1:

- Адрес станции в диапазоне от 0 до 247 (00 – F7 HEX).
- Скорость передачи (Baud rates) в диапазоне от 9.6 кбит/с до 115.2 кбит/с
- Контроль четности (Parity) – Нечет/Нет контроля (ODD/ No parity).



Все неиспользуемые биты установить в "0"

Порт 2: Адрес станции								
Адрес	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0
00	0	0	0	0	0	0	0	0
01	0	0	0	0	0	0	0	1
02	0	0	0	0	0	0	1	0

F6	1	1	1	1	0	1	1	0
F7	1	1	1	1	0	1	1	1

Порт 1: Скорость передачи			
Скорость	Бит 10	Бит 9	Бит 8
9.6 кбит/с	0	0	0
19.2 кбит/с	0	0	1
38.4 кбит/с	0	1	0
57.6 кбит/с	0	1	1
115.2 кбит/с	1	0	0

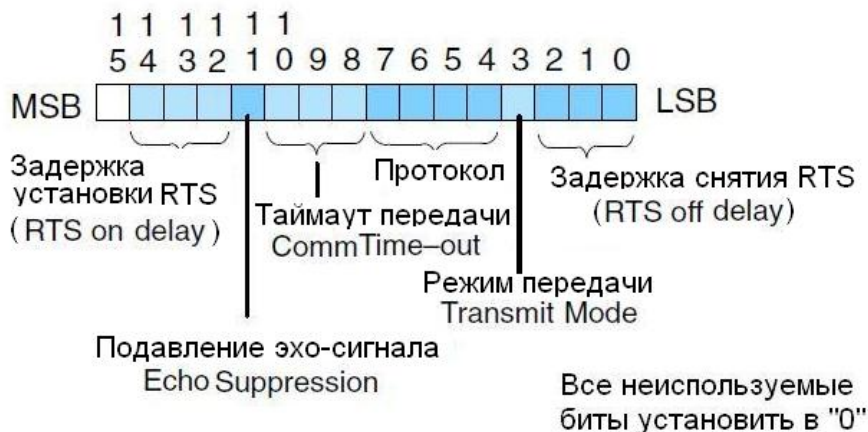
Порт 1: Четность	
Четность	Бит 14
No parity	0
ODD	1

С: Порт 2– Задержки RTS, режим передачи, протокол, таймаут, режим RS-485

Используйте «Слово+2» для этих настроек порта 2:

- K-sequence Slave, **DirectNET** Slave, MODBUS RTU Slave или все три протокола в режиме автоопределения.
- Режим передачи (transmit mode) - ASCII или HEX
- Задержки установки и снятия сигнала RTS (RTS On / Off delay)
- Скорость передачи (Baud rates) в диапазоне от 9.6 кбит/с до 115.2 кбит/с
- Подавление эхо-сигнала (Echo Suppression).

Слово +0002



Порт 2: Задержка установки RTS (RTS On delay)				Порт 2: Задержка снятия RTS (RTS Off delay)			
Время(мс)	Бит 2	Бит 1	Бит 0		Бит 14	Бит 13	Бит 12
0	0	0	0		0	0	0
2	0	0	1		0	0	1
5	0	1	0		0	1	0
10	0	1	1		0	1	1
20	1	0	0		1	0	0
50	1	0	1		1	0	1
100	1	1	0		1	1	0
500	1	1	1		1	1	1

Порт 2: Режим передачи	
Режим	Бит 3
HEX	0
ASCII	1

Порт 2: Протокол					
Протокол	Биты 7 – 4 (HEX)	Бит 7	Бит 6	Бит 5	Бит 4
K-sequence	8	1	0	0	0
DirectNET	4	0	1	0	0
MODBUS RTU	2	0	0	1	0
Non-sequence	1	0	0	0	1
K-sequence MODBUS RTU DirectNET	E	1	1	1	0

Порт 2: Таймаут передачи (мс)				
K-sequence MODBUS RTU DirectNET	Non-sequence протокол	Бит 10	Бит 9	Бит 8
Base Time x 1	0	0	0	0
Base Time x 1.2	2	0	0	1
Base Time x 1.5	5	0	1	0
Base Time x 2	10	0	1	1
Base Time x 5	20	1	0	0
Base Time x 10	50	1	0	1
Base Time x 20	100	1	1	0
Base Time x 50	500	1	1	1

Базовое значение таймаута (Base Time-out) для K-sequence / **DirectNET** = 800мс
 Базовое значение таймаута (Base Time-out) для MODBUS = 500мс

Порт 2: Подавление эхо-сигнала	
Режим	Бит 11
RS-422/485: 4-х проводный. RS-232	0
RS-485: 2-х проводный	1

D: Порт 2 – Адрес станции, скорость, биты данных, стоповые биты, четность

Используйте «Слово+3» для этих настроек порта 2

- Адрес станции в диапазоне от 0 до 247 (00 – F7 HEX).
- Скорость передачи (Baud rates) в диапазоне от 300 бит/с до 115.2 кбит/с
- Число бит данных (7 или 8)
- Число стоповых бит (1 или 2)
- Контроль четности (Parity) – Нечет/Чет/Нет контроля (ODD/EVEN/ No parity).



Порт 2: Адрес станции								
Адрес	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0
00	0	0	0	0	0	0	0	0
01	0	0	0	0	0	0	0	1
02	0	0	0	0	0	0	1	0

F6	1	1	1	1	0	1	1	0
F7	1	1	1	1	0	1	1	1

Порт 2: Скорость передачи				
Скорость	Бит 11	Бит 10	Бит 9	Бит 8
300 бит/с	0	0	0	0
600 бит/с	0	0	0	1
1200 бит/с	0	0	1	0
2400 бит/с	0	0	1	1
4800 бит/с	0	1	0	0
9.6 кбит/с	0	1	0	1
19.2 кбит/с	0	1	1	0
38.4 кбит/с	0	1	1	1
57.6 кбит/с	1	0	0	0
115.2 кбит/с	1	0	0	1

Порт 2: Биты данных	
<i>Четность</i>	<i>Бит 12</i>
8 бит	0
7 бит	1

Порт 2: Стоповые биты	
<i>Четность</i>	<i>Бит 13</i>
1 бит	0
2 бита	1

Порт 2: Четность		
<i>Четность</i>	<i>Бит 15</i>	<i>Бит 14</i>
No parity	0	0
ODD	0	1
EVEN	1	0

Е: Порт 2 – Задержка символов

Используйте «Слово+6» для настройки порта 2

- Задержка символов (Число в BCD - формате).

Этот параметр используется, когда DCM работает в качестве ведущей станции MODBUS RTU



Порт 2: Задержка символов	
<i>Биты 15 - 0</i>	<i>Время</i>
0	3.5 символа
1 – 9999 (BCD)	Тайм-аут (мс)

F: Порты 1 и 2 – Код завершения настройки

«Слово+7» используется для портов 1 и 2, чтобы:

- Сделать запрос, по которому D0-DCM распознает наличие изменений в настройках порта относительно настроек по умолчанию или от предыдущих настроек
- Подтвердить завершение настройки портов 1 и 2..



Загрузка константы K0055 (BCD) в слово +0007 пошлет запрос-команду, чтобы D0-DCM распознал новые коммуникационные параметры, записанные в регистры настроек. Если запрашиваемые изменения настроек допустимые, в слово +0007 будет записано число 00AA (HEX). Если запрашиваемые изменения настроек недопустимые или находятся вне диапазона, в слово +0007 будет записан код ошибки - число 00E*(HEX). Последняя цифра кода (*) указывает адрес, в котором произошла ошибка. Например, код ошибки 00E2 означает, что ошибка в слове+0002.

G: Порт 1 и 2 – Задержка перезапуска порта

Используйте «Слово+0024» для настройки портов 1 и 2

- Задержка перезапуска (Reset Time-out) в секундах (Число в BCD - формате).

Порт будет перезагружен через указанный интервал времени после перехода порта из состояния ЗАНЯТ (BUSY) в состояние ПАУЗА (IDLE).

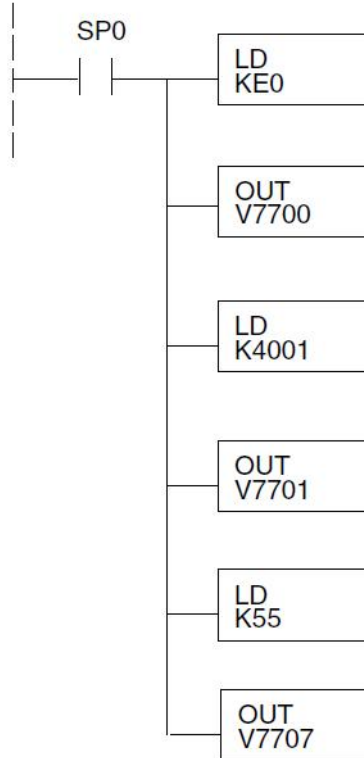


Порт 2: Задержка перезапуска		
<i>Порт</i>	<i>Время</i>	<i>Функция</i>
Порт 1: Биты 0 - 7	0	Отключена (Disable)
Порт 2: Биты 15 - 8	1 - 99	Включена (Enable)

Использование релейной логики для настройки D0-DCM (DL05)

Ниже приведены примеры программ релейной логики, созданные отделом технической поддержки Automation Direct.

Порт 1 (ведомый)



Выбор режима авто-определения протокола и режима передач HEX.

Запись настройки в V7700

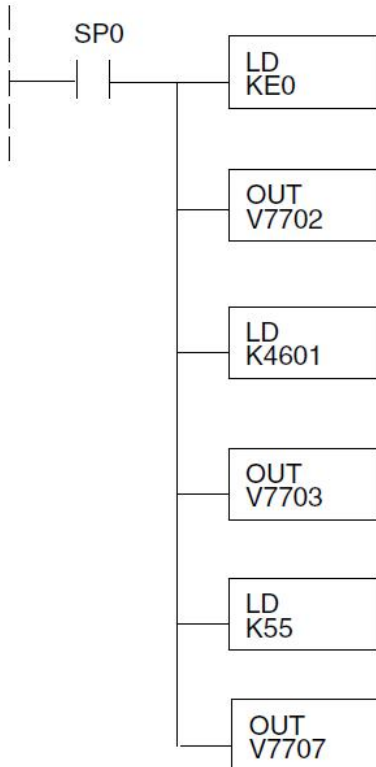
Выбор адреса станции – 1, 9600 бит/с и контроль четности – ODD.

Запись настройки в V7701

Константа K55 – это код завершения настройки, который необходим для того, чтобы ЦПУ распознал новые параметры настроек.

Запись настройки в V7707

Порт 2 в режиме ведомого



Выбор режима авто-определения протокола и режима передач HEX, задержки RTS On/Off равные 0 и 4-х проводной RS-422/485 / RS-232

Запись настройки в V7702

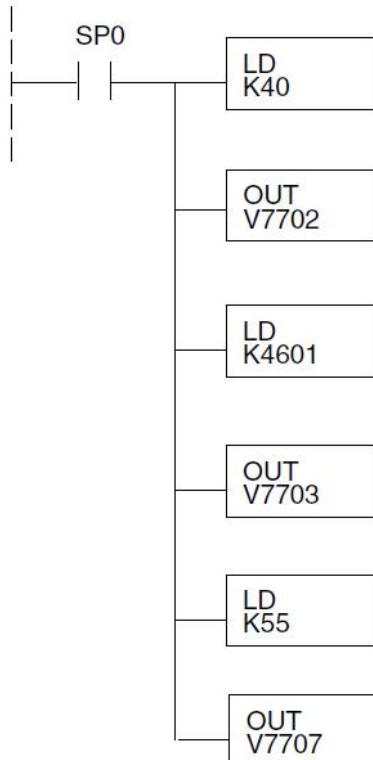
Выбор адреса станции – 1, 19200 бит/с, контроль четности – ODD, 8 бит данных и 1 стоповый бит.

Запись настройки в V7703

Константа K55 – это код завершения настройки, который необходим для того, чтобы ЦПУ распознал новые параметры настроек.

Запись настройки в V7707

Порт 2 в режиме ведущего DirectNET



Выбор протокола DirectNET и режима передач HEX. Задержки RTS On/Off равны 0, 4-х проводной RS-422/485 / RS-232.

Запись настройки в V7702

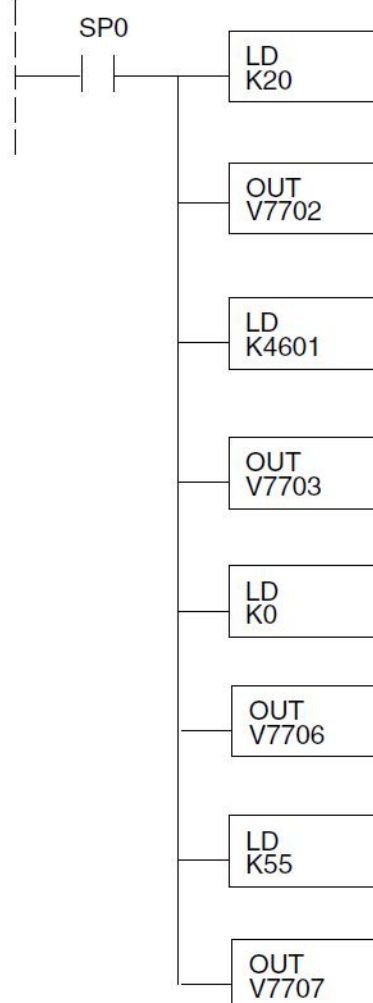
Выбор адреса станции – 1, 19200 бит/с, контроль четности – ODD, 8 бит данных и 1 стоповый бит.

Запись настройки в V7703

Константа K55 – это код завершения настройки, который необходим для того, чтобы ЦПУ распознал новые параметры настроек.

Запись настройки в V7707

Порт 2 в режиме ведущего Modbus



Выбор протокола Modbus и режима передач HEX. Задержки RTS On/Off равны 0, 4-х проводной RS-422/485 / RS-232

Запись настройки в V7702

Выбор адреса станции – 1, 19200 бит/с, контроль четности – ODD, 8 бит данных и 1 стоповый бит.

Запись настройки в V7703

Выбор задержки (тайм-аут) длиной 3.5 символа

Запись настройки в V7703

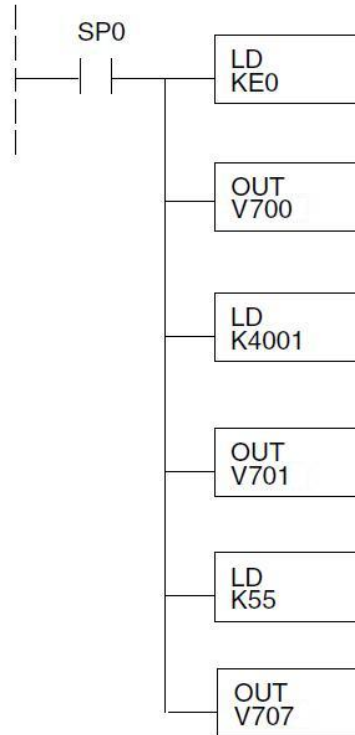
Константа K55 – это код завершения настройки, который необходим для того, чтобы ЦПУ распознал новые параметры настроек.

Запись настройки в V7707

Использование релейной логики для настройки D0-DCM (DL06)

Ниже приведены примеры программ релейной логики, созданные отделом технической поддержки Automation Direct.

Порт 1 (ведомый)



Выбор режима авто-определения протокола и режима передач HEX.

Запись настройки в V700 (Зависит от слота, где размещен D0-DCM).

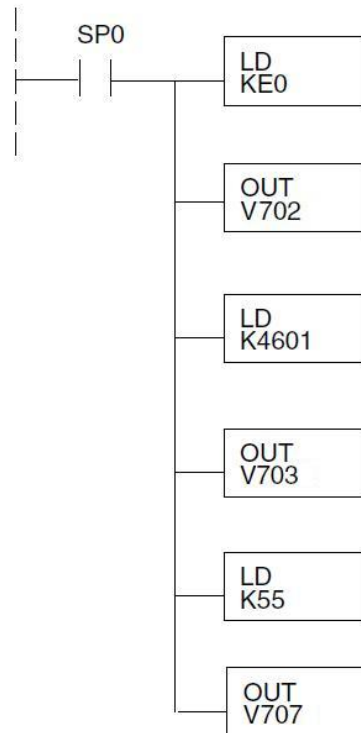
Выбор адреса станции – 1, 9600 бит/с и контроль четности – ODD.

Запись настройки в V701 (Зависит от слота, где размещен D0-DCM).

Константа K55 – это код завершения настройки, который необходим для того, чтобы ЦПУ распознал новые параметры настроек.

Запись настройки в V707 (Зависит от слота, где размещен D0-DCM).

Порт 2 в режиме ведомого



Выбор режима авто-определения протокола и режима передач HEX, задержки RTS On/Off равные 0 и 4-х проводной RS-422/485 / RS-232

Запись настройки в V702 (Зависит от слота, где размещен D0-DCM).

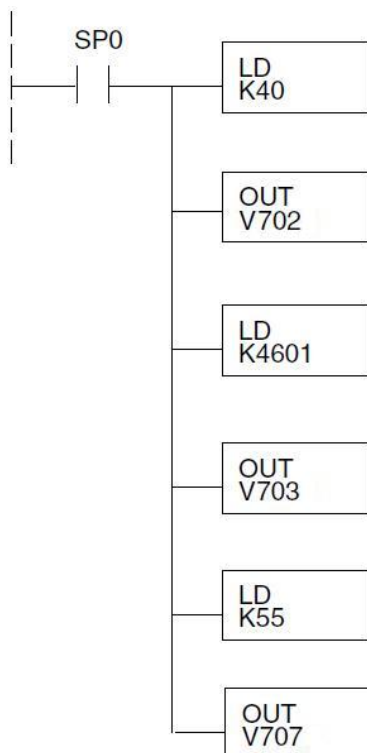
Выбор адреса станции – 1, 19200 бит/с, контроль четности – ODD, 8 бит данных и 1 стоповый бит.

Запись настройки в V703 (Зависит от слота, где размещен D0-DCM).

Константа K55 – это код завершения настройки, который необходим для того, чтобы ЦПУ распознал новые параметры настроек.

Запись настройки в V707 (Зависит от слота, где размещен D0-DCM).

Порт 2 в режиме ведущего DirectNET



Выбор протокола DirectNET и режима передач HEX. Задержки RTS On/Off равны 0, 4-х проводной RS-422/485 / RS-232.

Запись настройки в V702(Зависит от слота, где размещен D0-DCM).

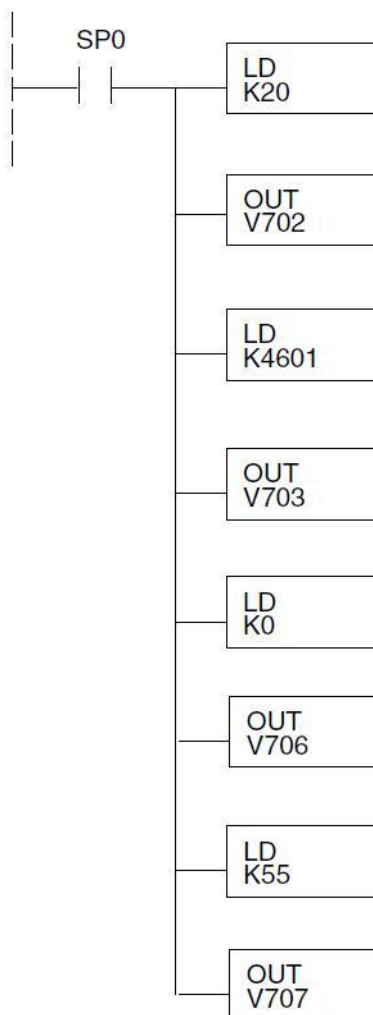
Выбор адреса станции – 1, 19200 бит/с, контроль четности – ODD, 8 бит данных и 1 стоповый бит.

Запись настройки в V703(Зависит от слота, где размещен D0-DCM).

Константа K55 – это код завершения настройки, который необходим для того, чтобы ЦПУ распознал новые параметры настроек.

Запись настройки в V707(Зависит от слота, где размещен D0-DCM).

Порт 2 в режиме ведущего Modbus



Выбор протокола Modbus и режима передач HEX. Задержки RTS On/Off равны 0, 4-х проводной RS-422/485 / RS-232

Запись настройки в V702 (Зависит от слота, где размещен D0-DCM).

Выбор адреса станции – 1, 19200 бит/с, контроль четности – ODD, 8 бит данных и 1 стоповый бит.

Запись настройки в V703 (Зависит от слота, где размещен D0-DCM).

Выбор задержки (тайм-аут) длиной 3.5 символа

Запись настройки в V7703 (Зависит от слота, где размещен D0-DCM).

Константа K55 – это код завершения настройки, который необходим для того, чтобы ЦПУ распознал новые параметры настроек.

Запись настройки в V7707 (Зависит от слота, где размещен D0-DCM).

Глава 4. DirectNet. Использование команд RX/WX

В этой главе...

- Команды RX / WX
- Адресация к различным типам памяти
- Специальные коммуникационные реле
- Пример программы с одной командой RX
- Пример программы с одной командой WX
- Использование нескольких команд RX / WX

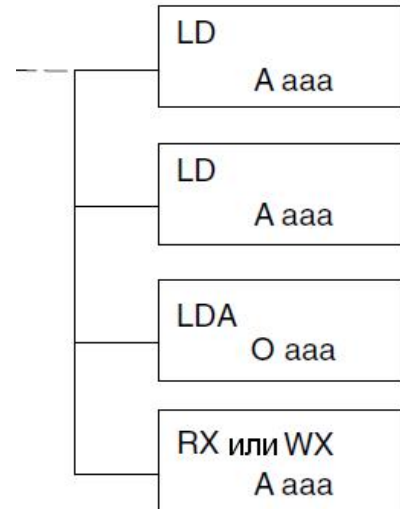
Команды RX / WX

Команды чтения (RX) и записи (WX)

Команды чтения (RX) и записи (WX) использует ведущий (master) ПЛК, чтобы читать блок данных из другого ПЛК или чтобы записать блок данных в другой ПЛК. Для выполнения своего назначения команды RX/WX должны предвоступаться двумя командами LD (Загрузить) и одной командой LDA (Загрузить адрес).
Команды LD и команда LDA загружают коммуникационные параметры в аккумулятор и в первый и второй уровень стека аккумулятора. Команда RX (или WX) использует эти параметры из стека и аккумулятора, чтобы приготовить данные, которые будут посланы по сети.

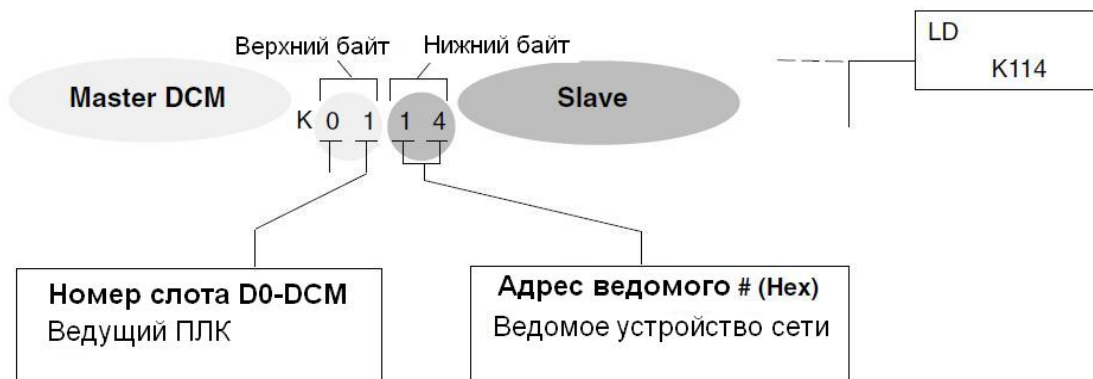
Создание цепи чтения (RX) или записи (WX)

Для организации обмена данными по сети, Вы должны встроить команду чтения (RX) или записи (WX) в программу из 4-х команд, которая показана справа.
Назначение каждой из команд будет разъяснено далее. Эти команды должны использоваться только в указанной последовательности.



Первая команда LD

В первой команде LD, в качестве операнда, может быть использована константа или переменная. Используйте префикс «K», если вводимое число является адресом регистра памяти. Используйте префикс «V», если вводимое число является регистром V-памяти. Содержимое этого регистра будет выполнять те же функции, что и константа. Например, если Вы укажете V2000, вместо K0114 (кака показано в примере внизу) и содержимым регистра V2000 будет число 114, функции команды будут те же. Использование переменных позволяет изменять параметры во время работы программы. Однако, рекомендуется, если возможно, использовать константу.



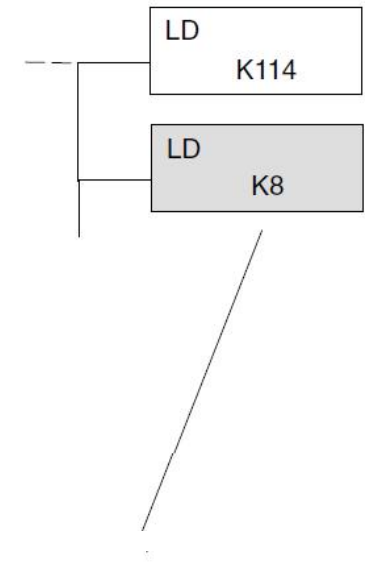
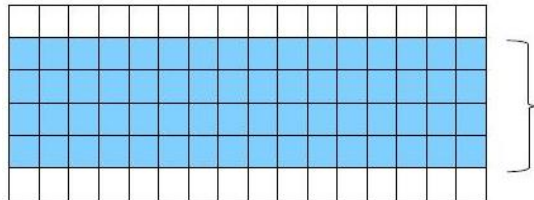
Вторая команда LD

Вторая команда LD указывает размер передаваемого командами RX или WX блока данных в байтах (128 максимально). В этой команде также можно использовать два типа данных. Используйте префикс «K», если вводимое число является адресом регистра памяти. Используйте префикс «V», если вводимое число является регистром V-памяти.

Вы должны указывать размер в целых байтах. Например, Y0–Y27 –это три байта или 24 бита в диапазоне дискретных выходов Y0–Y27 (адреса восьмеричные).

Для данных типа «Слово» можно использовать значения кратные 2 в диапазоне от 2 до 128. Для битовых данных можно использовать значения в диапазоне от 1 до 128. Подробности на странице 4-6.

Память ПЛК



4 слова = 8 байт

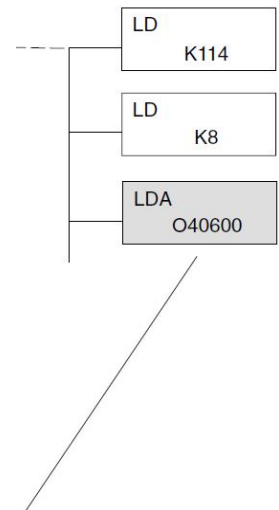
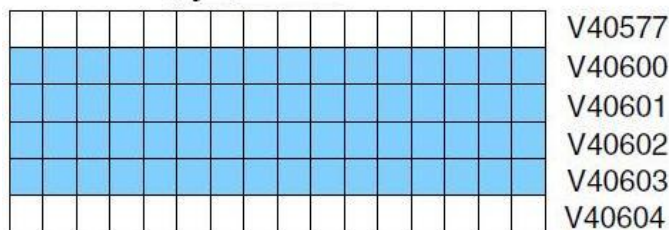
Команда LDA

Команда *Загрузить адрес* - LDA указывает начальный адрес V-памяти используемой в ведущем устройстве. Блок передаваемых данных будет начинаться с этого адреса и продолжаться на число байт, указанных в предыдущей команде. Префикс «O» показывает, что это число восьмеричное (октальное).

Команда чтения (RX) копирует блок данных из памяти ведомого ПЛК в память ведущего ПЛК.

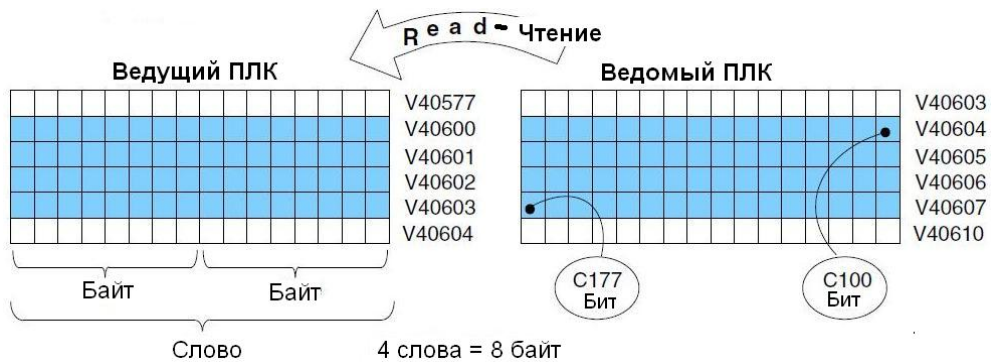
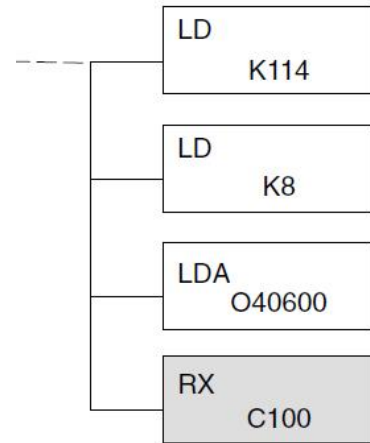
Команда записи (WX) копирует блок данных из памяти ведущего ПЛК в память ведомого ПЛК.

Ведущий ПЛК



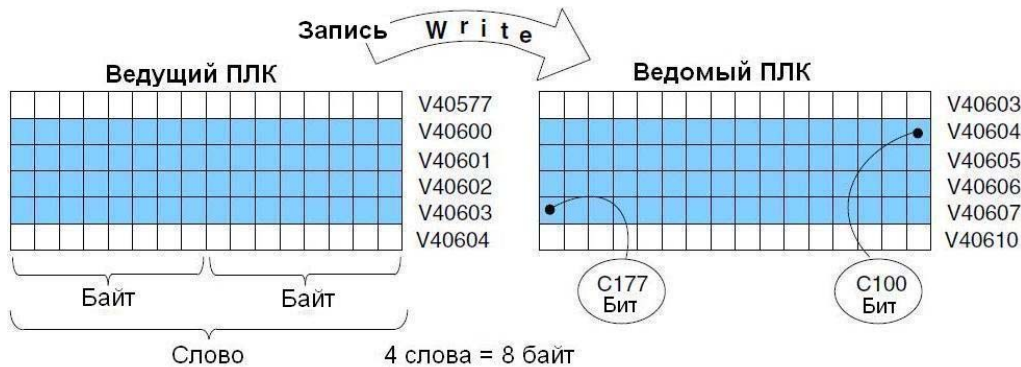
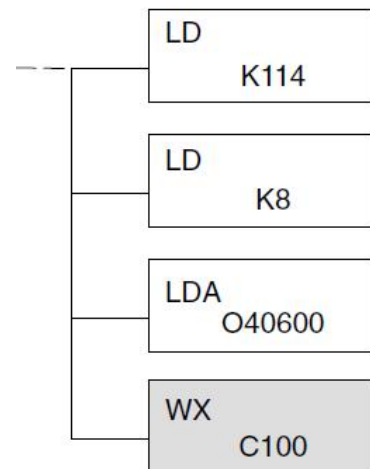
Команда RX

Команда чтения (RX) указывает место в памяти ведомого ПЛК, откуда будут считаны данные. Блок данных считывается начиная с указанного места в памяти и продолжается на число байт, указанных во второй команде LD. В этом примере блок данных длиной 8 байт (начиная с C100 и кончая C177) считывается (копируется) из памяти ведомого ПЛК в память ведущего ПЛК начинающуюся с V40600.



Команда WX

Команды записи (WX) копирует блок данных из памяти ведущего ПЛК в память ведомого. Команда записи (WX) указывает место в памяти ведомого ПЛК, откуда будут записаны данные. Блок данных записывается начиная с указанного места в памяти и продолжается на число байт, указанных во второй команде LD. В примере блок данных длиной 8 байт начинающийся с V40600 и заканчивающийся на V40603 записывается (копируется) из памяти ведущего ПЛК в память ведомого ПЛК, начиная с C100 и кончая C177.



Адресация к различным типам памяти

Некоторые данные по своему существу 16-ти битовые, например, текущие значения таймеров и счетчиков. Другие данные однобитовые, например, дискретные входы и выходы. Но те и другие данные отображаются в общей Словной памяти (Word Memory), называемой V-память (V-memory), вторичная позволяет обращаться к различным типам данным, как к 16-ти битовым словам.

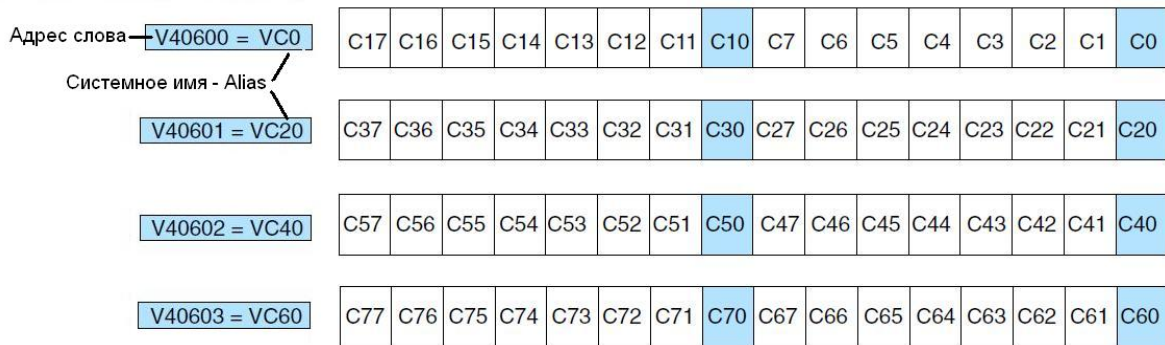
Бит-адресуемая память

К бит-адресуемой памяти (или, просто, к битовой памяти) в командах RX/WX можно обращаться по имени первого бита любого байта. Если во второй команде LD использована константа K8, то восемь байт будут переданы. Если использован адрес C0, то будут переданы биты с C0 по C77.

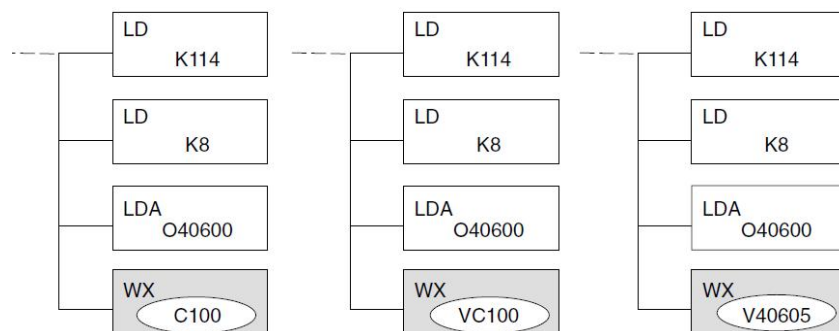
Память адресуемая по словам

В примере приведенном ниже, слово V40600 обозначает использование бит с C0 по C17. Системные имена (Aliases) могут заменять адрес в командах RX/WX. Вы можете использовать адрес V40600 или VC0.

Системные имена (Aliases) это просто имена первых битов группы из 16-ти бит. В бит-адресуемой памяти, в памяти адресуемой по словам и в системных именах (Aliases) используется восьмеричная система счисления.



Все, показанные ниже, программы эквивалентны. **DirectSOFT** позволяет использовать три различных способа обращения к регистрам памяти.



Типы памяти Вы можете адресоваться к различным типам данных, любым способом, приведенным в таблицах. Самый большой блок данных, который может быть послан одной командой чтения или записи – 128 байт. Самый маленький блок данных, который может быть послан одной командой это один байт для бит-адресуемого типа данных и 2 байта (одно слово) для памяти адресуемой по словам (Word memory). Все адреса указаны в таблицах в восьмеричной (октальной) системе счисления.3

ЦПУ DL05

ЦПУ DL05			
Типы данных	Битовая память	Слова памяти	Системные имена
Текущие значения таймеров	Нет	V0 – V177	TA0 – TA177
Текущие значения счетчиков	Нет	V1000 – V1177	CTA0 – CTA177
Данные пользователя	Нет	V1200 – V7377	Нет
Точки входов	X0 – X377	V40400 – V40417	VX0 – VX360
Точки выходов	Y0 – Y377	V40500 – V40517	VY0 – VY360
Промежуточные реле	C0 – C777	V40600 – V40617	VC0 – VC760
Специальные реле	SP0 – SP777	V41200 – V41237	VSP0 – VSP760
Биты состояния таймеров	T0 – T177	V41100 – V41107	VT0 – VT160
Биты состояния счетчиков	CT0 – CT177	V41140 – V41147	VCT0 – VCT160
Биты стадий	S0 – S377	V41000 – V41017	VS0 – VS360

ЦПУ DL06

ЦПУ DL06			
Типы данных	Битовая память	Слова памяти	Системные имена
Текущие значения таймеров	Нет	V0 – V377	TA0 – TA377
Текущие значения счетчиков	Нет	V1000 – V1177	CTA0 – CTA177
Данные пользователя	Нет	V400 – V677 V1200 – V7377 V10000 – V17777	Нет
Точки входов	X0 – X777	V40400 – V40437	VX0 – VX760
Точки выходов	Y0 – Y777	V40500 – V40537	VY0 – VY760
Промежуточные реле	C0 – C1777	V40600 – V40677	VC0 – VC1760
Специальные реле	SP0 – SP777	V41200 – V41237	VSP0 – VSP760
Биты состояния таймеров	T0 – T377	V41100 – V41107	VT0 – VT360
Биты состояния счетчиков	CT0 – CT177	V41140 – V41147	VCT0 – VCT160
Биты стадий	S0 – S1777	V41000 – V41077	VS0 – VS1760
Удаленный Ввод/вывод	GX0 – GX3777 GY0 – GY3777	V40000 – V40117 V40200 – V40317	VGX0 – VGX3760 VGY0 – VGY3760

Специальные коммуникационные реле

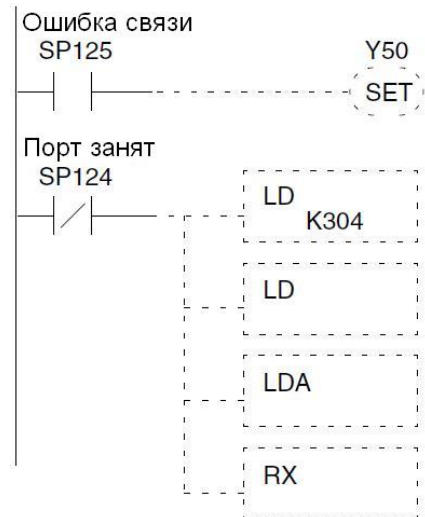
У контроллеров **DirectLOGIC** есть внутренние контакты (биты) для контроля состояния коммуникаций. Внутренние контакты называются Специальные реле (Special Relays). По два специальных реле на каждый слот ПЛК выделено для использования модулем D0-DCM. Эти два реле выполняют следующие функции:

- **Порт занят (Communication Busy)** – Этот бит в состоянии ВКЛ, когда порт модуля занят, принимая или посылая данные. Вы должны использовать этот бит (контакт реле), чтобы предотвратить наложение команд RX/WX друг на друга.
- **Ошибка связи (Communication Error)** - Этот бит в состоянии ВКЛ при возникновении ошибки при выполнении последней команды RX/WX. Эта ошибка автоматически сбрасывается (бит=0) при выполнении следующей команды RX/WX.

Например, специальные реле SP124 и SP125 соотносятся со слотом 3 DL06. В примере специальное реле SP125 использовано, чтобы активировать выход Y50, который сигнализирует появление ошибки связи. Это специальное реле должно размещаться в программе до команд RX/WX, так как оно переходит в состояние отключено, при выполнении последующей команды RX/WX.

Специальное реле SP124 отражает состояние порта D0-DCM – «Порт занят». Когда реле SP124 включено, нормально-закрытый контакт размыкается, блокируя выполнение других команд RX/WX, до завершения выполняемой операции.

Соответствующий бит **должен использоваться** в виде нормально-закрытого контакта в цепях каждой из команд RX/WX.



Специальные реле контроллеров DL05 и DL06 приведены в таблицах ниже.

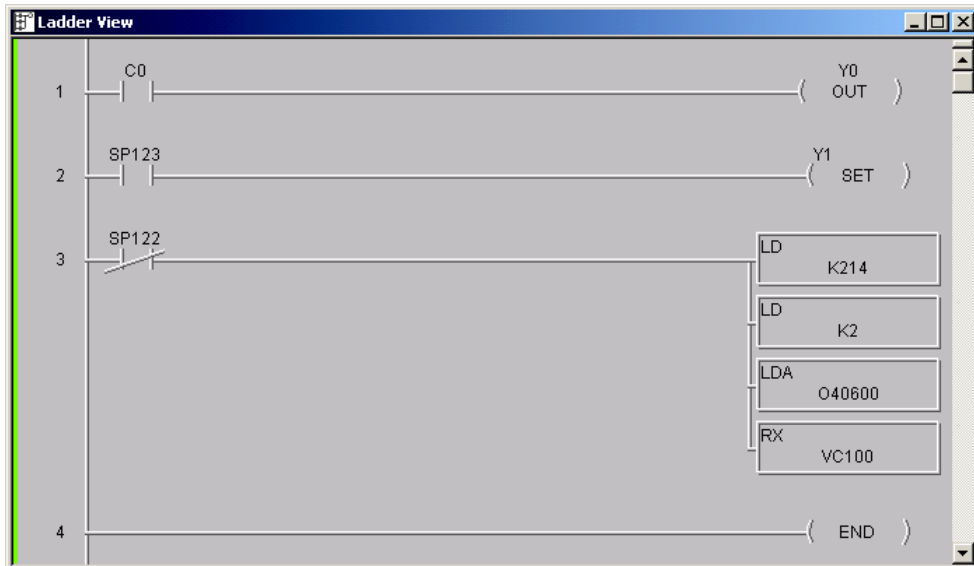
Специальные реле DL05	
Реле	Слот
Порт занят	SP120
Ошибка связи	SP121

Специальные реле DL06				
Реле	Слот 1	Слот 2	Слот 3	Слот 4
Порт занят	SP120	SP122	SP124	SP126
Ошибка связи	SP121	SP123	SP125	SP127

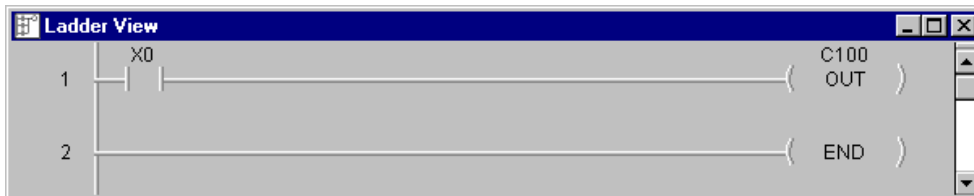
Программа с одной командой RX

Ниже приведен экран редактирования программы (Ladder View) в среде разработки **DirectSOFT**. На нем представлена программа, состоящая из 4-х цепей, подробно рассмотренная на следующей странице. Это полностью завершенная программа, хотя и с очень ограниченными функциями. На следующем рисунке программа, которая работает в ведомом ПЛК, также подробно рассмотренная на следующей странице. В этом примере D0-DCM установлен в слот 2 ПЛК DL06.

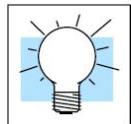
Программа для ведущего ПЛК



Программа для ведомого ПЛК



Когда вход X0 в ведомом контроллере переходит в состояние «включено» (переход из 0 в 1), бит C0 в ведущем контроллере, также, переходит из 0 в 1. Программа ведущего ПЛК включает выход Y0 при срабатывании бита C0.



Пример коммуникационных программ для контроллеров DL05/DL06 можно найти на сайте www.automationdirect.com > Technical support> Example programs> Communications> example # EP-COM-005.

Master example: Проект содержит простую программу для ведущего ПЛК. Программа считывает входы ведомого DL05/DL06 и отображает их состояние в C0 – C17, программа, также, пишет значения C20 – C37 в выходы ведомого ПЛК.

Slave example: Этот проект может быть использован в сочетании с проектом для ведущего контроллера, чтобы настроить ведомый ПЛК на выключение его выходов, если ведущий ПЛК остановит обмен информации с ведомым ПЛК

Программа для ведущего ПЛК

В нашем примере, нормально-открытый контакт C0 – это внутреннее промежуточное реле.



Цепь 1

Когда C0 находится в состоянии «включено» дискретный, выход Y0 запитан (включен).

Цепь 2

Во второй цепи использовано специальное реле, чтобы сигнализировать наличие ошибки связи (communication error). В нашем примере SP123 переходит в состоянии «Включено», если в слоте 2 присутствует ошибка связи. Используйте специальное реле соотносящееся со слотом, в котором установлен Ваш D0-DCM (см. таблицу на странице 4-7). Мы используем SP123, чтобы зажечь сигнальную лампочку присоединную к дискретному выходу Y1.

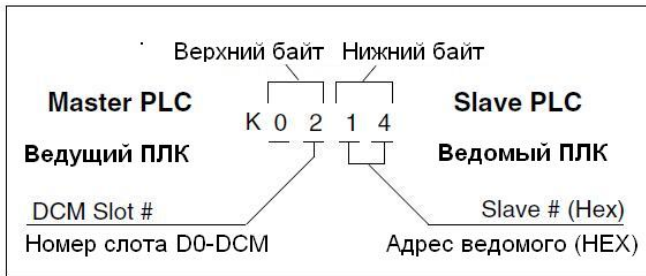


Цепь 3

Специальное реле SP122 находится в состоянии «Включено», когда D0-DCM в слоте 2 занят, передавая или принимая данные.

Выполнение команды RX может продолжаться дольше и после завершения текущего скан-цикла ПЛК.

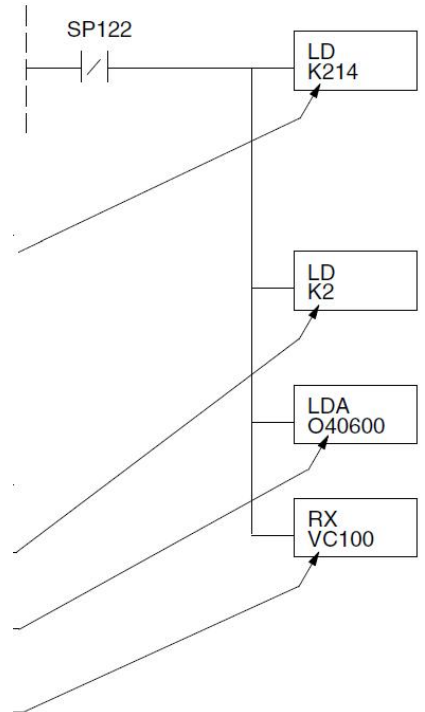
Используйте это специальное реле, чтобы предотвратить перезапись предыдущей команды в каждом скан-цикле ПЛК.



Число передаваемых байт. Максимально 128 байт.

Начальный адрес в ведущем ПЛК (восьмеричное число).

Начальный адрес в ведомом ПЛК



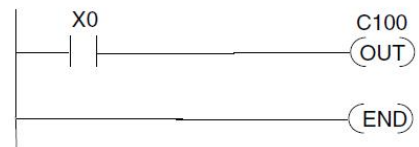
Цепь 4

В программах всех контроллеров **DirectLOGIC** обязательно использование команды END для указания последней цепи основной программы.



Программа для ведомого ПЛК

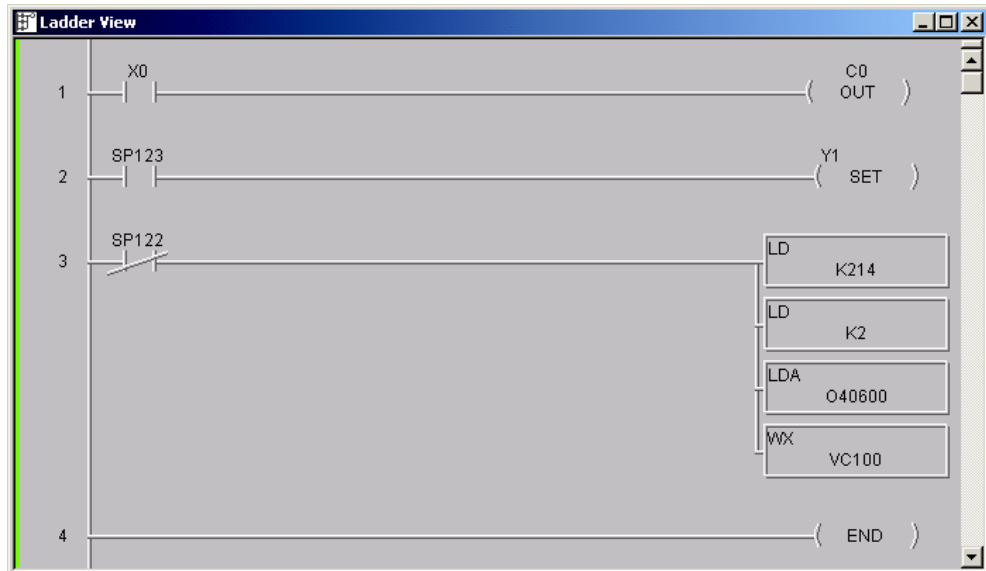
Простая программа из двух цепей в ведомом ПЛК выполняет единственную функцию – при включении контакта X0 взводит внутренне промежуточное реле C100.



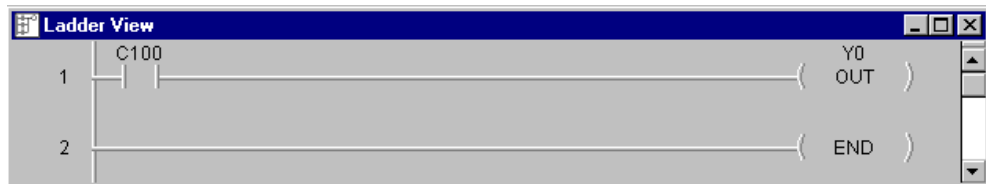
Программа с одной командой WX

Ниже приведен экран редактирования программы (Ladder View) в среде разработки **DirectSOFT**. На нем представлена программа состоящая из 4-х цепей подробно рассмотренная на следующей странице. Это полностью завершенная программа, хотя и с очень ограниченными функциями. На следующем рисунке программа, которая работает в ведомом ПЛК, также подробно рассмотренная на следующей странице. В этом примере D0-DCM установлен в слот 2 ПЛК DL06.

Программа для ведущего ПЛК



Программа для ведомого ПЛК



Когда вход X0 в ведущем контроллере переходит в состояние «включено» (переход из 0 в 1), бит C100 в ведомом контроллере, также, переходит из 0 в 1. Программа ведомого ПЛК включает выход Y0 при срабатывании бита C100.



ПРИМЕЧАНИЕ: Программа ведомого контроллера служит только примером. При обрыве связи между контроллерами, биты, записанные в ведомый контроллер, сохраняют последнее записанное состояние.



Пример коммуникационных программ для контроллеров DL05/DL06 можно найти на сайте www.automationdirect.com > Technical support> Example programs> Communications> example # EP-COM-005.

Master example: Проект содержит простую программу для ведущего ПЛК. Программа считывает входы ведомого DL05/DL06 и отображает их состояние в C0 – C17, программа, также, пишет значения C20 – C37 в выходы ведомого ПЛК.

Slave example: Этот проект может быть использован в сочетании с проектом для ведущего контроллера, чтобы настроить ведомый ПЛК на выключение его выходов, если ведущий ПЛК остановит обмен информации с ведомым ПЛК

Программа для ведущего ПЛК

В нашем примере, нормально-открытый контакт X0 привязан ко входу модуля ввода дискретных сигналов. Когда X0 находится в состоянии «включено», промежуточное реле C0 запитано (включено).



Цепь 1

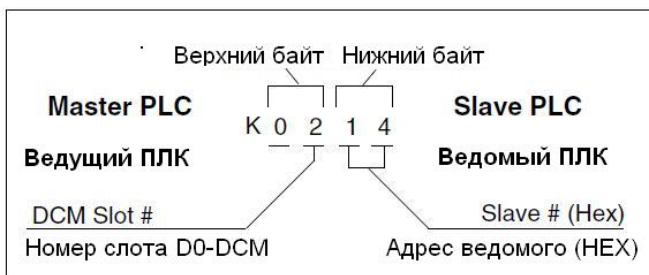
Цепь 2

Во второй цепи использовано специальное реле, чтобы сигнализировать наличие ошибки связи (communication error). В нашем примере SP123 переходит в состоянии «Включено», если в слоте 2 присутствует ошибка связи. Используйте специальное реле соотносящееся со слотом, в котором установлен Ваш D0-DCM (см. таблицу на странице 4-7). Мы используем SP123, чтобы зажечь сигнальную лампочку присоединенную к дискретному выходу Y1.



Цепь 3

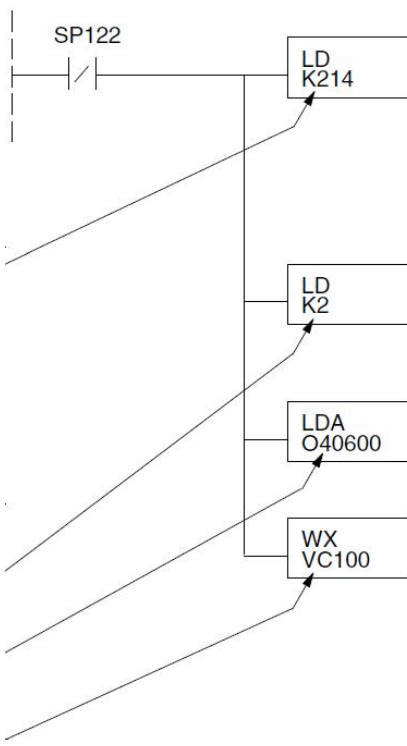
Специальное реле SP122 находится в состоянии «Включено», когда D0-DCM в слоте 2 занят, передавая или принимая данные. Выполнение команды WX может продолжаться дольше и после завершения текущего скан-цикла ПЛК. Используйте это специальное реле, чтобы предотвратить перезапись предыдущей команды в каждом скан-цикле ПЛК.



Число передаваемых байт. Максимально 128 байт.

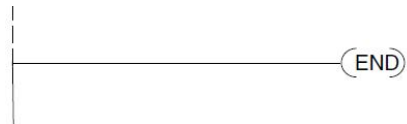
Начальный адрес в ведущем ПЛК (восьмеричное число).

Начальный адрес в ведомом ПЛК.



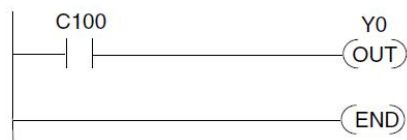
Цепь 4

В программах всех контроллеров **DirectLOGIC** обязательно использование команды END для указания последней цепи основной программы.



Программа для ведомого ПЛК

Простая программа из двух цепей в ведомом ПЛК выполняет единственную функцию – при включении контакта промежуточного реле C100 взводит физический выход Y0.



Программа с несколькими командами RX и WX

Использование нескольких команд RX/WX требует применение взаимных блокировок (interlocks), так как только одна команда RX/WX может выполняться в каждый момент времени. При использовании взаимных блокировок, только одна из команд RX/WX будет выполняться в каждом скан-цикле до тех пока все взаимных блокировок, так как только одна команды RX/WX не будут выполнены. Затем последовательность команд RX/WX начнет выполняться с первой команды RX/WX. Без взаимных блокировок, команды RX/WX могли бы выполняться в непредсказуемом порядке, и некоторые могли бы исполняться несколько раз. Контакты взаимных блокировок размыкают цепи релейной логики всех команд RX/WX так, что только одна команда RX/WX будет обработана в текущем скан-цикле ЦПУ.

Мы покажем два способа создания взаимных блокировок для управления последовательностью выполнения команд RX/WX:

- Управления последовательностью при помощи контактов реле
- Управления последовательностью при помощи регистра сдвига (Shift Register)

Далее будут приведены примеры программ использующих оба способа для выполнения одной и той же функции.



ПРИМЕЧАНИЕ: Чтобы полностью понимать материал этого раздела Вам необходимо сначала разобраться с программами примеров, приведенными на страницах 4-8 и 4-10, также как и с разделом Сетевые команды, начинающимся на стр. 4-2.

Далее приведен сегмент программы с тремя командами RX/WX (две команды WX и одна команда RX). Вы можете создать собственную программу, вставив этот сегмент с двумя блокировочными битами и расширив число блокировок в соответствии с требуемым числом команд RX/WX.

Блокировочные реле

Понять функционирование блокировочных контактов легче, если сначала создать таблицу истинности.

В столбцах таблицы мы размещаем внутренние промежуточные реле, которые мы будем использовать для организации нашей последовательности команд. Мы будем использовать в нашей таблице реле с C50 по C52. Но для этой цели можно использовать любые контакты не занятые в других частях программы.

В строках таблицы мы поместим список команд RX/WX, которые мы хотим применить в нашей программе релейной логики.

Три контакта в этой таблице истинности могут управлять восемью командами RX/WX. В нашей программе только три команды RX/WX, поэтому нам можно использовать только два контакта (см. страницу 4-13). Мы будем использовать C50 и C51. Один дополнительный контакт C53 даст 32 комбинации, так как число комбинаций увеличивается пропорционально степени двойки.

Таблица истинности	C52	C51	C50
Первая RX/WX	0	0	0
Вторая RX/WX	0	0	1
Третья RX/WX	0	1	0
Четвертая RX/WX	0	1	1
Пятая RX/WX	1	0	0
Шестая RX/WX	1	0	1
Седьмая RX/WX	1	1	0
Восьмая RX/WX	1	1	1

Наши три команды RX/WX могут быть выполняться последовательно под управлением двух контактов: C50 и C51. Два контакта обеспечивают четыре различных бинарных состояний:

- Оба отключены
- C50 – включен и C51 – отключен
- C50 – отключен и C51 – включен
- Оба включены

Нам необходимо использовать только три из четырех двоичных состояний, так как у нас надо организовать последовательность из трех команд RX/WX.

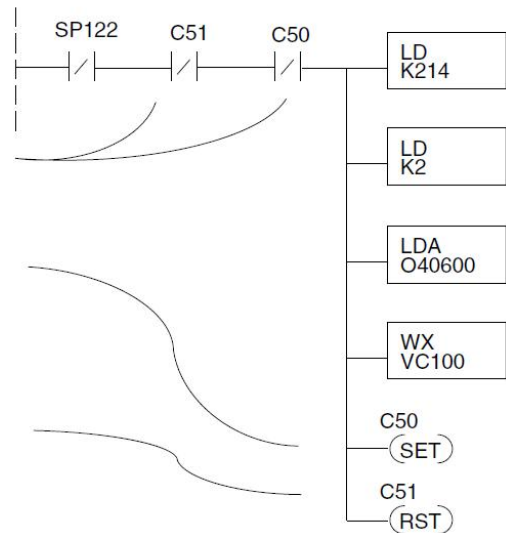
Таблица истинности	C52	C51	C50
Первая RX/WX	0	0	0
Вторая RX/WX	0	0	1
Третья RX/WX	0	1	0
Четвертая RX/WX	0	1	1
Пятая RX/WX	1	0	0
Шестая RX/WX	1	0	1
Седьмая RX/WX	1	1	0
Восьмая RX/WX	1	1	1

Первая команда RX/WX

C50 и C51 – контакты взаимной блокировки. Они нормально закрыты, чтобы пропустить ток к первой команде WX. Оба бита в состоянии «0», что соответствует первой строке таблицы истинности.

После исполнения команды WX, C50 переводится в «1» командой SET. Контакт C50 в этой цепи разрывается и закрывается контакт C50 в следующей цепи.

C51 сбрасывается командой RESET, что оставляет контакт C51 закрытым в следующей цепи.

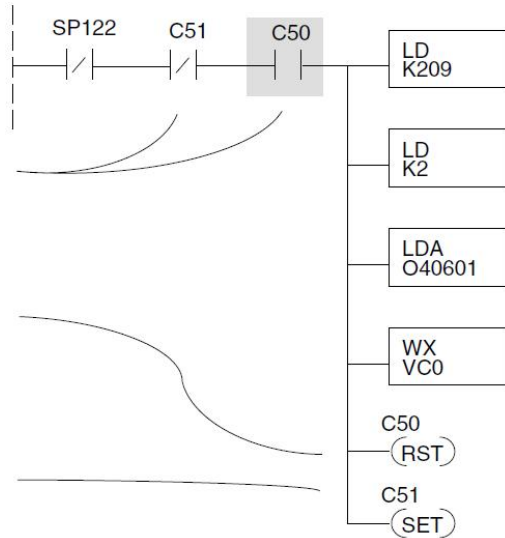


Вторая команда RX/WX

В этой цепи C50 нормально открытый, а C51 нормально закрытый. Для выполнения этой цепи C50 должен быть в «1» и C51 должен быть в «0», в соответствии со второй строкой таблицы истинности. Бит C50 был взведен в предыдущей цепи, а бит C50 был сброшен в предыдущей цепи.

После исполнения команды WX, C50 переводится в «1» командой SET. Контакт C50 в этой цепи разрывается и закрывается контакт C50 в следующей цепи.

C51 сбрасывается командой RESET, что оставляет контакт C51 закрытым в следующей цепи.

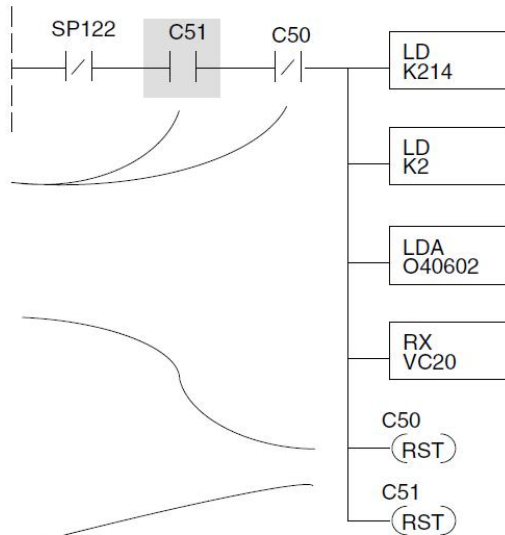


Третья команда RX/WX

В этой последней цепи C50 нормально закрытый, а C51 нормально открытый. Для выполнения этой цепи C50 должен быть в «0» и C51 должен быть в «1», в соответствии с третьей строкой таблицы истинности. Бит C51 был взведен в предыдущей цепи.

После исполнения команды RX, C50 сбрасывается в «0» командой RESET, что разрывает контакт C50 в этой цепи и позволяет закрыть контакт C50 для подготовки повторения первой цепи в следующем скан-цикле ЦПУ.

C51 сбрасывается командой RESET, что позволяет закрыть контакт C51, для подготовки повторения первой цепи управления обменом данными в следующем скан-цикле ЦПУ.



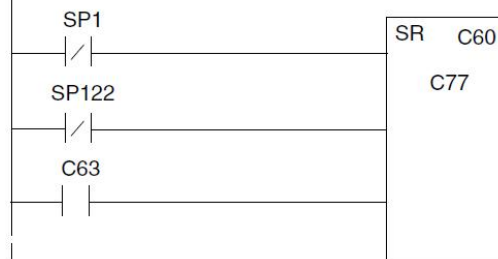
Возврат к первой команде RX/WX

В конце третьей команды RX/WX, мы делаем цикл обратно к верхней строке таблицы истинности (стр. 4-13). Оба контакта C50 и C51 в состоянии «0» и в следующем скан-цикле ЦПУ будет выполнена первая команда RX/WX.

Регистр сдвига

Регистр сдвига (*Shift Register*) может быть использован для создания взаимных блокировок, в качестве альтернативы применения блокировочных реле. Работа регистра сдвига подробно описана в руководстве пользователя Вашего ПЛК. Если у Вас много команд RX/WX, то использование блокировочных реле становится затруднительным. Регистр сдвига позволяет использовать только один контакт в каждой цепи выполнения команд RX/WX для взаимной блокировки.

Входами команды *Shift Register* (SR) являются Специальные реле. SP1 - это бит «всегда включен». В сочетании с нормально-закрытым контактом он посылает нули во вход данных команды SR. SP122 - это реле «порт занят» соединенное с тактовым входом команды SR. Каждый раз при выполнении одной из команд RX/WX регистр сдвига перемещает биты на одно место. Контакт C63 использован в этом примере для сброса всех битов Регистра сдвига в ноль.



C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	C60
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Регистр сдвига после первого скана

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	C60
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Регистр сдвига после второго скана

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	C60
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Регистр сдвига после первой команды RX/WX

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	C60
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Регистр сдвига после второй команды RX/WX

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	C60
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Регистр сдвига после третьей команды RX/WX

C77	C76	C75	C74	C73	C72	C71	C70	C67	C66	C65	C64	C63	C62	C61	C60
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

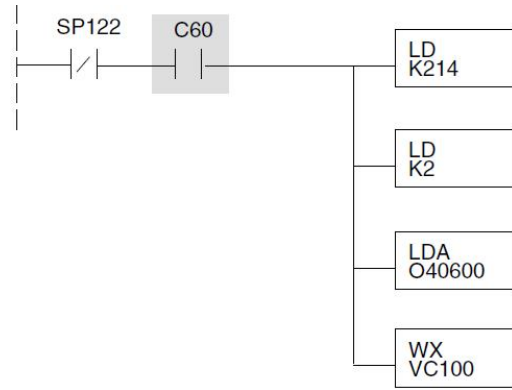
Регистр сдвига после третьей команды плюс следующий скан

Команда «Сохранить если равно» (*Store If Equal*) определяет момент, когда регистр сдвига сброшен в ноль. При выполнении этого условия бит C60 взводится командой SET. Бит C60 переходит в состояние «1» после выполнения всех команд RX/WX.



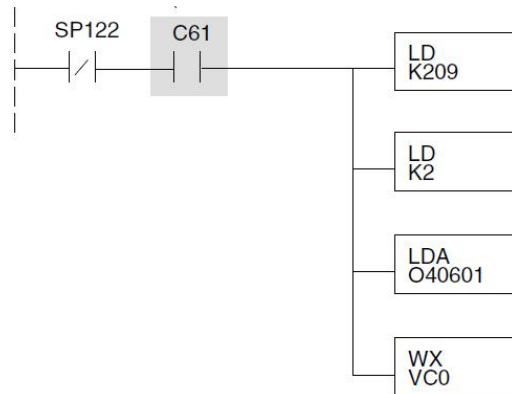
Первая команда RX/WX

C60— контакт взаимной блокировки. Бит переводится в состояние «1» командой *Store If Equal* в предыдущей цепи.



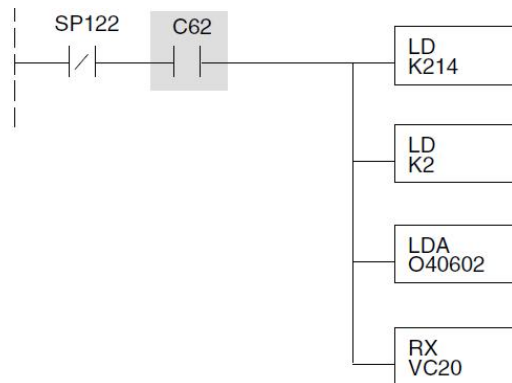
Вторая команда RX/WX

C61— контакт взаимной блокировки. Бит переводится в состояние «1» последовательными шагами команды *Shift Register* в предыдущей цепи.



Третья команда RX/WX

C62— контакт взаимной блокировки. Бит переводится в состояние «1» последовательными шагами команды *Shift Register* в предыдущей цепи.



После выполнения этой цепи, Регистр сдвига перемещает взведенный бит с C62 на C63 в следующем скан-цикле ЦПУ. C63 сбрасывает биты Регистр сдвига в нули, команда *Store If Equal* взводит бит C60 и ЦПУ начинает выполнять первую команду RX/WX.

Глава 5. Modbus RTU. Использование команд RX/WX и MRX/MWX

В этой главе...

- Работа в качестве ведомого устройства сети
- Работа в качестве ведущего устройства сети: команды RX / WX
- Работа DL06 ведущим устройством сети: команды MRX / MWX

Работа в качестве ведомого устройства сети

В этом разделе описано, как ведущая станция (master) сети MODBUS RTU может связываться с D0–DCM, используя протокол MODBUS RTU.

Ведущая станция должна посылать функциональные коды MODBUS и адреса MODBUS, чтобы специфицировать ячейку памяти ПЛК в ЦПУ DL05/DL06. Для прерывания работы ПЛК в режиме ведомого устройства (slave) не требуется создание программы релейной логики.

Поддерживаемые функции MODBUS

В приведенной ниже таблице содержатся коды функций (команды) MODBUS поддерживаемые D0–DCM при работе в режиме ведомого устройства MODBUS

Коды функций MODBUS	Функция	Доступные типы данных DL05/06
01	Чтение группы выходов (Coils)	Y, C, T, CT
02	Чтение группы входов (Inputs)	X, SP
05	Форсирование одного выхода	Y, C, T, CT
03	Чтение значений регистров общего назначения (Holding Registers)	V
04	Чтение значений регистров аналоговых входов (Input Registers)	V
06	Запись значения в один регистр	V
08	Диагностическая функция	
15	Форсирование нескольких выходов	Y, C, T, CT
16	Запись значения в группу регистров	V

Определение адреса MODBUS

Существуют два способа, с помощью которых большинство программных средств ведущих устройств могут специфицировать ячейку памяти ПЛК. Ими являются:

- Указание типа данных и адреса MODBUS,
- Указание только адреса MODBUS.

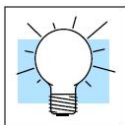
Если программное обеспечение требует тип данных и адрес

Многие программные пакеты ведущих устройств позволяют указать тип данных MODBUS и адрес MODBUS, которые соответствуют ячейке памяти ПЛК. Это наиболее простой способ, но не все пакеты имеют такую возможность.

Уравнение, применяемое для вычисления адреса, учитывает тип данных, который вы используете. В этих целях типы данных ПЛК разделяются на две категории.

- Дискретные — X, SP, Y, C, S, T, T (контакты), CT (контакты)
- Слова — V, текущие значения таймера, текущие значения счетчика

В любом случае по существу Вы преобразовываете восьмеричный адрес ПЛК в десятичный и добавляете соответствующий адрес MODBUS (если необходимо). В таблице ниже приведена точная формула, используемая для каждой группы данных.



Для автоматического преобразования адресов MODBUS/KOYO загрузите Утилиту - файл: Modbus_conversion.xls с сайта www.automationdirect.com > Technical and Application notes > PLC hardware > Communications > app note # AN-MISC-010.

Тип памяти DL05	Кол-во (Дес.)	Диапазон ПЛК (Восьмеричный)	Диапазон MODBUS	Тип данных MODBUS
<i>Для дискретного типа данных... Преобразуйте адрес ПЛК в десятичный + Начало диапазона + Тип данных</i>				
Входы (X)	256	X0 - X377	2048 - 2303	Вход (Input)
Специальные реле (SP)	512	SP0 - SP777	3072 - 3583	Вход
Выходы (Y)	256	Y0 - Y377	2048 - 2303	Реле (Coil)
Управляющие реле (C)	512	C0 - C777	3072 - 3583	Реле
Контакты таймера (T)	128	T0 - T177	6144 - 6271	Реле
Контакты счетчика (CT)	128	CT0 - CT177	6400 - 6527	Реле
Биты состояния стадий (S)	256	S0 - S377	5120 - 5375	Реле
<i>Для типа данных "слово"...Переведите адрес ПЛК в десятичный + Тип данных</i>				
Текущие значения таймеров (V)	128	VO - V177	0 - 127	Регистр входов (Input Register)
Текущие значения счетчиков (V)	128	V1000 - V1177	512 - 639	Регистр входов
V-память, данные пользователя (V)	3072	V1400 - V7377	768 - 3839	Регистр хранения (Holding Register)

Тип памяти DL06	Кол-во (Дес.)	Диапазон ПЛК (Восьмеричный)	Диапазон MODBUS	Тип данных MODBUS
Для дискретного типа данных... Преобразуйте адрес ПЛК в десятичный + Начало диапазона + Тип данных				
Входы (X)	512	X0 - X777	2048 - 2560	Вход (Input)
Специальные реле (SP)	512	SP0 - SP777	3072 - 3583	Вход
Выходы (Y)	512	Y0 - Y377	2048 - 2560	Реле (Coil)
Управляющие реле (C)	1024	C0 - C1777	3072 - 4095	Реле
Контакты таймера (T)	256	T0 - T377	6144 - 6399	Реле
Контакты счетчика (CT)	128	CT0 - CT177	6400 - 6527	Реле
Биты состояния стадий (S)	1024	S0 - S1777	5120 - 6143	Реле
Глобальные входы (GX)	2048	GX 0 - GX 3777	0 - 2047	Вход
Глобальные выходы (GY)	2048	GY 0 - GY 3777	0 - 2047	Реле
Для типа данных "слово"...Переведите адрес ПЛК в десятичный + Тип данных				
Текущие значения таймеров (V)	256	VO - V377	0 - 255	Регистр входов (Input Register)
Текущие значения счетчиков (V)	128	V1000 - V1177	512 - 639	Регистр входов
V-память, данные пользователя (V)	256	V400 - V777	256 - 511	Регистр хранения (Holding Register)
	3072	V1400 - V7377	768 - 3839	
	4096	V10000 - V17777	4096 - 8191	

В следующих примерах показано, как сформировать адрес и тип данных MODBUS для программного обеспечения ведущего устройства, в котором требуется этот формат.

**Пример 1:
V2100**

- Найти адрес MODBUS для ячейки V2100 V-памяти.
1. Найти ячейку V-памяти в таблице.
 2. Преобразовать V2100 в десятичный код (1088).
 3. Использовать тип данных MODBUS из таблицы.

Адрес ПЛК (десят.) + тип данных
2100 восьмеричное= 1088 десятичному
1088+Рег.хранен.= **Регистр хранения**
1088

Текущее значение Таймера (V)	128	V0 - V177	0 - 127	Регистр входов
Текущее значение Счетчика (V)	128	V1000 - V1177	512 - 639	Регистр входов
V память, данные пользователя (V)	1024	V2000 - V3777	1024- 2047	Регистр хранения

**Пример 2:
Y20**

- Найти адрес MODBUS для выхода Y20.
1. Найти выходы Y в таблице.
 2. Преобразовать Y20 в десятичное (16).
 3. Добавить начальный адрес диапазона (2049).
 4. Использовать тип данных MODBUS из таблицы.

Адрес ПЛК (десят.) + начальный адрес + тип данных
20 восьмеричное = 16 десятичному.

$16+2049+Реле = \text{Реле } 2065$

Выходы (Y)	320	Y0 - Y477	2049 -2367	Реле
Управляющие реле (C)	256	C0 - C377	3072 -3551	Реле

**Пример 3:
TA10**

- Найти адрес MODBUS для ячейки текущего значения таймера T10.
1. Найти текущее значение таймера в таблице.
 2. Преобразовать TA10 в десятичный код (8).
 3. Использовать тип данных MODBUS из таблицы.

Адрес ПЛК (десят.) + тип данных
10 восьмеричное= 8 десятичн.

$8 + \text{Вх. регистр} = \text{Регистр входов } 8$

Текущее значение Таймера (V)	128	V0 - V177	0 - 127	Регистр входов
Текущее значение Счетчика (V)	128	V1000 -V1177	512 - 639	Регистр входов

**Пример 4:
C54**

- Найти адрес MODBUS для управляющего реле C54.
1. Найти управляющие реле в таблице.
 2. Преобразовать C54 в десятичное (44).
 3. Добавить начальный адрес диапазона (3072).
 4. Использовать тип данных MODBUS из таблицы.

Адрес ПЛК (десят.) + начальный адрес + тип данных
C54 = 44 десятичному.

$44+3073+Реле = \text{Реле } 3117$

Выходы (Y)	320	Y0 - Y477	2049 -2367	Реле
Управляющие реле (C)	256	C0 - C377	3073 -3551	Реле

Если программное обеспечение требует только адрес

Некоторые программы ведущих устройств не позволяют задавать тип данных и адрес MODBUS. Они могут указать только адрес. В этом случае адрес определяется другим способом, но тоже достаточно просто. MODBUS разделяет типы данных по диапазонам адресов. Отсюда следует, что адреса достаточно для выбора типа данных и ячейки памяти. Это часто называют «добавление смещения».

Формула, применяемая для вычисления адреса, учитывает тип данных, который вы используете. В этих целях типы данных ПЛК разделяются на две категории.

- Дискретные — X, GX, SP, Y, C, S, T(контакты), CT (контакты)
- Слова — V, текущие значения таймера, текущая значение счетчика

В любом случае Вы, по существу, преобразуете восьмеричный адрес ПЛК в десятичный и добавляете соответствующий адрес MODBUS (если необходимо). В таблице ниже приведен точный диапазон для каждого типа данных.



Для автоматического преобразования адресов MODBUS/KOYO загрузите Утилиту - файл: Modbus_conversion.xls с сайта www.automationdirect.com > Technical and Application notes> PLC hardware> Communications> app note # AN-MISC-010.

Дискретные типы данных DL05				
Тип памяти DL05	Кол-во (Дес.)	Диапазон ПЛК (Восьмеричный)	Диапазон MODBUS	Доступ
Входы (X)	256	X0 - X377	12049 - 12304	Только чтение
Специальные реле (SP)	512	SP0 - SP777	13073 - 13584	
Резерв	-	-	13585 - 20000	
Выходы (Y)	256	Y0 - Y377	2049 - 2304	Чтение и запись
Управляющие реле (C)	512	C0 - C777	3073 - 3584	
Контакты таймера (T)	128	T0 - T177	6145 - 6272	
Контакты счетчика (CT)	128	CT0 - CT177	6401 - 6528	
Биты состояния стадий (S)	256	S0 - S377	5121 - 5376	
Резерв	-	-	6529 - 10000	

Данные типа слово DL05					
Регистры (слова) DL05	Кол-во (Дес.)	Диапазон ПЛК (Восьмеричный)	Диапазон MODBUS 40001	Диапазон MODBUS 30001	Доступ
Текущие значения таймеров (V)	128	VO - V177	40001-40128	30001-30128	Только чтение
Текущие значения счетчиков (V)	128	V1000 - V1177	40513 - 40640	30513 - 30640	Чтение и запись
V-память, данные пользователя (V)	3072	V1400 - V7377	40769 - 43840	30769 - 33840	

Дискретные типы данных DL06				
Тип памяти DL05	Кол-во (Дес.)	Диапазон ПЛК (Восьмеричный)	Диапазон MODBUS	Доступ
Входы (X)	512	X0 - X377	12049 - 12560	Только чтение
Глобальные входы (GX)	2048	GX 0 - GX 1746 GX 1747 - GX 3777	10001 - 10999 11000 - 12048	
Специальные реле (SP)	512	SP0 - SP777	13073 - 13584	
Резерв	-	-	13585 - 20000	
Глобальные выходы (GY)	2048	GY 0 - GY 3777	1 - 2048	Чтение и запись
Выходы (Y)	512	Y0 - Y777	2049 - 2560	
Управляющие реле (C)	1024	C0 - C1777	3073 - 4096	
Контакты таймера (T)	256	T0 - T377	6145 - 6400	
Контакты счетчика (CT)	128	CT0 - CT177	6401 - 6528	
Биты состояния стадий (S)	1024	S0 - S1777	5121 - 6144	
Резерв	-	-	6529 - 10000	

Данные типа слово DL06					
Регистры (слова) DL05	Кол-во (Дес.)	Диапазон ПЛК (Восьмеричный)	Диапазон MODBUS 40001	Диапазон MODBUS 30001	Доступ
Текущие значения таймеров (V)	256	VO - V377	40001-40256	30001-30256	Только чтение
Текущие значения счетчиков (V)	128	V1000 - V1177	40513 - 40640	30513 - 30640	Чтение и запись
V-память, данные пользователя (V)	256	V400 - V777	40257 - 40512	30257- 30512	
	3072	V1400 - V7377	40769 - 43840	30769 - 33840	
	4096	V10000 - V17777	44097 - 48192	34097 - 38192	

Пример 1: V2100 Найти адрес MODBUS для ячейки V2100 V-памяти.

1. Найти ячейку V-памяти в таблице.
2. Преобразовать V2100 в десятичный код (1088).
3. Добавить начальный адрес для этого режима (40001)

Адрес ПЛК (десят.) + Адрес режима
 V2100 = 1088 десятичное.
 $1088 + 40001 = 41089$

Для типа данных слова...		Адр. ПЛК (дес) +		Адрес соответствующего режима		
Текущ. знач. Таймера (V)	128	V0-V177	0 - 127	3001	30001	Регистр входов
Текущ. знач. Счетчика (V)	128	V1000-V1177	512-639	3001	30001	Регистр входов
V память данные пользователя (V)	1024	V2000-V3777	1024-2047	4001	40001	Регистр хранения

Пример 2: Y20 Найти адрес MODBUS для выхода Y20.

1. Найти выходы Y в таблице.
2. Преобразовать Y20 в десятичное (16).
3. Добавить начальный адрес диапазона (2048).
4. Добавить начальный адрес для этого режима (1).

Адрес ПЛК (десят.) + начальный адрес + режим
 Y20 = 16 десятичное.

$16 + 2048 + 1 = 2065$

Выходы (Y)	320	Y0-Y477	2048-2367	1	1	Реле
Управляющие реле (CR)	256	C0-C377	3072-3551	1	1	Реле
Контакты таймера (T)	128	T0-T177	6144- 6271	1	1	Реле

Пример 3: C54 Найти адрес MODBUS для управляющего реле C54.

1. Найти управляющие реле в таблице.
2. Преобразовать C54 в десятичное (44).
3. Добавить начальный адрес диапазона (3072).
4. Добавить адрес MODBUS для этого режима (1).

Адрес ПЛК (десят.) + начальный адрес + режим
 C54 = 44 десятичное.

$44+3072+1=3117$

Выходы (Y)	320	Y0-Y477	2048-2367	1	1	Реле
Управляющие реле (C)	256	C0-C377	3072-3551	1	1	Реле
Контакты таймера (T)	128	T0-Y177	6144 - 6271	1	1	Реле

Работа в качестве ведущего устройства сети: команды RX/WX

Введение

В данном разделе описывается, как DL05/DL06 могут работать в качестве ведущего устройства сети MODBUS RTU, используя D0-DCM. В разделе будет показано, как создать необходимую релейную логику для работы ведущего устройства сети.



При использовании D0-DCM в качестве ведущей станции применяйте сетевые команды RLL для инициализации запросов. Команда WX инициирует сетевые операции записи, а команда RX инициирует сетевые операции чтения. Перед выполнением этих команд вам необходимо загрузить в стек аккумулятора процессора данные, относящиеся к операции чтения или записи.

При выполнении команды WX или RX процессор использует информацию в этом стеке наряду с данными поля команды, чтобы полностью определить задачу, которая выдается порту.



Поддерживаемые функции MODBUS

В приведенной ниже таблице содержатся коды функций (команды) MODBUS поддерживаемые D0-DCM при работе в режиме ведущего устройства MODBUS

Коды функций MODBUS	Функция	Доступные типы данных DL05/06
01	Чтение группы выходов (Coils)	Y, C, T, CT
02	Чтение группы входов (Inputs)	X, SP
05	Форсирование одного выхода	Y, C, T, CT
03	Чтение значений регистров общего назначения (в режиме адресации 584/984 используется для чтения аналоговых выходов)	V
06	Запись значения в один регистр	V
15	Форсирование нескольких выходов	Y, C, T, CT
16	Запись значения в группу регистров	V



ПРИМЕЧАНИЕ: При работе в качестве ведущего D0-DCM не поддерживает код 4. Следовательно, адреса диапазона 30001 не могут быть считаны с ведомого устройства.

Память ПЛК под-держиваемая в режиме ведущего

Формула, применяемая для вычисления адреса, учитывает тип данных, который вы используете. В этих целях типы данных ПЛК разделяются на три категории.

- Дискретные — X, GX, SP
- Дискретные — Y, C, S, T, CT
- Слова — V, текущие значения таймера, текущая значение счетчика

В любом случае Вы берете адрес MODBUS цели вычитаете начальный адрес диапазона MODBUS, преобразуете результат в восьмеричный адрес и добавляете соответствующий начальный адрес типа данных. В таблице ниже приведены точные диапазоны для каждого типа данных.



Для автоматического преобразования адресов MODBUS/KOYO загрузите Утилиту - файл: Modbus_conversion.xls с сайта www.automationdirect.com > Technical Support > Application notes > PLC hardware > Communications > app note # AN-MISC-010.

Дискретные типы данных DL06*				
Тип памяти DL05	Кол-во (Дес.)	Диапазон ПЛК (Восьмеричный)	Диапазон MODBUS	Доступ
Входы (X)	512	X0 - X377	12049 - 12560	Только чтение
Глобальные входы (GX)	2048	GX 0 - GX 1746 GX 1747 - GX 3777	10001 - 10999 11000 - 12048	
Специальные реле (SP)	512	SP0 - SP777	13073 - 13584	
Резерв	-	-	13585 - 20000	
Глобальные выходы (GY)	2048	GY 0 - GY 3777	1 - 2048	Чтение и запись
Выходы (Y)	512	Y0 - Y777	2049 - 2560	
Управляющие реле (C)	1024	C0 - C1777	3073 - 4096	
Контакты таймера (T)	256	T0 - T377	6145 - 6400	
Контакты счетчика (CT)	128	CT0 - CT177	6401 - 6528	
Биты состояния стадий (S)	1024	S0 - S1777	5121 - 6144	
Резерв	-	-	6529 - 10000	

Данные типа слово DL06*				
Регистры (слова) DL05	Кол-во (Дес.)	Диапазон ПЛК (Восьмеричный)	Диапазон MODBUS 40001	Доступ
Текущие значения таймеров (V)	256	VO - V377	40001-40256	Только чтение
Текущие значения счетчиков (V)	128	V1000 -V1177	40513 – 40640	Чтение и запись
V-память, данные пользователя (V)	256	V400 - V777	40257 – 40512	
	3072	V1400 -V7377	40769 – 43840	
	4096	V10000 - V17777	44097 – 48192	

* - Данные для ЦПУ DL05 приведены на странице 5-6. Диапазоны памяти DL06 в некоторых регистрах шире, чем у DL05.



Вы можете использовать калькулятор Windows для преобразования чисел из десятичного в октальный. Калькулятор при этом должен быть настроен на Инженерный вид.

Пример 1: Вычисление адреса слова ПЛК

Найти адрес ПЛК соответствующий адресу MODBUS **41025** в устройстве-сервере.

- | | |
|---|---|
| 1. Вычитите начальный адрес диапазона – 40001 | 1. $41025 - 40001 = 1024$ (дец.) |
| 2. Преобразуйте десятичный результат в октальный (восьмеричный) | 2. $1024(\text{дец.}) = 2000$ (октальное) |
| 3. Добавьте начальный адрес диапазона ПЛК (Вход, Выход, Слово) | 3. $V0 + 2000 = \mathbf{V2000}$ (октальное) |

Пример 2: Вычисление адреса дискретного входа ПЛК

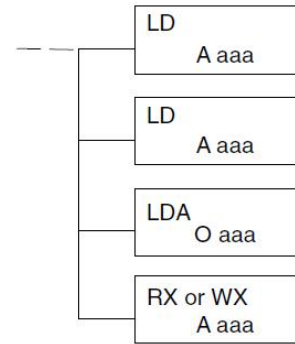
Найти адрес ПЛК соответствующий адресу MODBUS **12060** в устройстве-сервере.

- | | |
|---|---|
| 4. Вычитите начальный адрес диапазона – 12049 | 1. $12060 - 12049 = 11$ (дец.) |
| 5. Преобразуйте десятичный результат в октальный (восьмеричный) | 2. 11 (дец.) = 13 (октальное) |
| 6. Добавьте начальный адрес диапазона ПЛК (Вход, Выход, Слово) | 3. $X0 + 13 = \mathbf{X13}$ (октальное) |

Создание программы чтения или записи

Для обмена данными по сети необходимо встроить команды Чтения (RX) и Записи (WX) в программу из 4-х команд, показанную справа.

Следующая пошаговая процедура содержит информацию, необходимую для настройки программы релейной логики для получения данных из сервера сети.



Шаг1: Указать слот DCM и адрес ведомого устройства

В первой команде LD, в качестве операнда, может быть использована константа или переменная. Используйте префикс «K», если вводимое число является адресом регистра памяти. Используйте префикс «V», если вводимое число является регистром V-памяти. Содержание этого регистра будет выполнять те же функции, что и константа. Например, если Вы укажете V2000, вместо K0114 (как показано в примере внизу) и содержимым регистра V2000 будет число 114, функции команды будут те же. Использование переменных позволяет изменять параметры во время работы программы.



В зависимости от номера слота, в котором установлен D0-DCM, он может использовать контакты двух Специальных реле ассоциированных с этим слотом. Одно реле отражает состояние порта D0-DCM –«Порт занят», другое сигнализирует появление ошибки связи «Ошибка связи». Бит «Порт занят» находится в состоянии «Включен» во время обмена данными ПЛК с ведомым устройством. Когда этот бит в состоянии «Отключен», программа может инициировать следующий запрос.

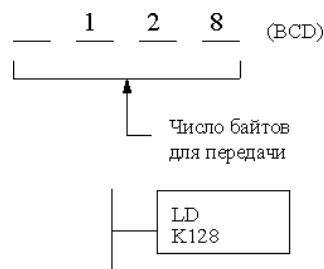
Специальные реле контроллеров DL05 и DL06 приведены в таблицах ниже.

Специальные реле DL05	
Реле	Слот
Порт занят	SP120
Ошибка связи	SP121

Специальные реле DL06				
Реле	Слот 1	Слот 2	Слот 3	Слот 4
Порт занят	SP120	SP122	SP124	SP126
Ошибка связи	SP121	SP123	SP125	SP127

Шаг 2:
Загрузить число байтов для передачи

Вторая команда загрузки (LD) определяет число байтов, которые необходимо передать между ведущим и ведомым устройствами в последующей команде WX или RX. Загружаемое значение имеет формат BCD (десятичный) от 1 до 128 байт.



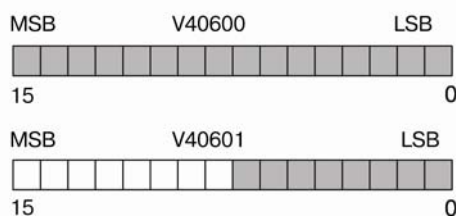
Шаг 3:
Определить область памяти ведущего устройства

Третьей командой в последовательности RX или WX является команда загрузить адрес (LDA). Ее назначение — загрузить начальный адрес области памяти, которая должна быть передана. Получая на входе восьмеричное число, команда LDA преобразует его в шестнадцатиричное число, а результат помещает в аккумулятор.



По команде WX процессор DL05/DL06 посылает предварительно определенное число байтов из области памяти, начиная с определенного инструкцией LDA адреса.

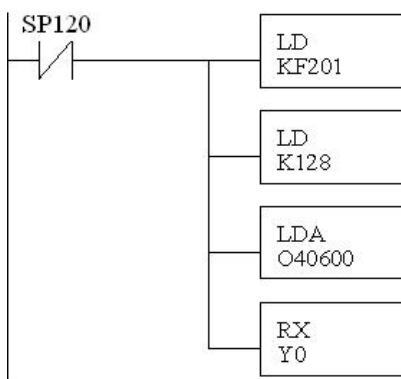
По команде RX процессор DL05/DL06 считывает предварительно определенное число байтов из ведомого устройства, помещает полученные данные в область памяти, начиная с определенного инструкцией LDA адреса.



ПРИМЕЧАНИЕ: Поскольку слова V-памяти всегда имеют 16 битов, то Вы не всегда будете использовать все слово. Например, если Вы задали только 3 байта и читаете выходы Y с ведомого устройства, то получите только 24 бита данных. В этом случае только 8 битов младших бит ячейки последнего слова будут нести информацию. Старшие 8 бит не имеют значения.

Шаг 4:
определить область памяти ведомого устройства

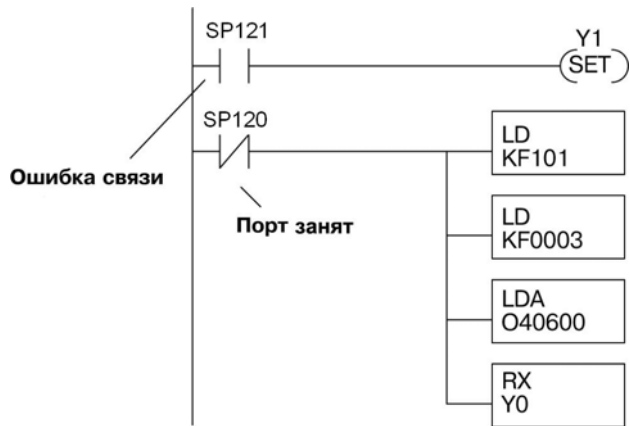
Последней командой в нашей последовательности является сама команда WX или RX. Используйте WX для записи в ведомое устройство, а RX — для чтения с ведомого устройства. Все четыре наши команды показаны справа. С помощью последней команды вы должны определить начальный адрес и действительный тип данных для ведомого устройства.



Пример программы с командой RX

Обычно передача данных по сети длится дольше, чем один цикл сканирования. Перед началом очередной операции программа должна ожидать пока закончится предыдущая передача данных.

В зависимости от номера слота, в котором он установлен, D0-DCM может использовать контакты двух специальных реле ассоциированных с этим слотом (см. страницу 5-12).



Одно из них указывает «Порт занят» (SP120), другое указывает «Ошибка связи» (SP121). Приведенный выше пример показывает использование этих контактов для D0-DCM в слоте 1. Бит «Порт занят» находится в состоянии «включен», когда ПЛК осуществляет обмен с ведомым устройством. Когда этот бит «выключен», программа может инициировать следующий сетевой запрос.

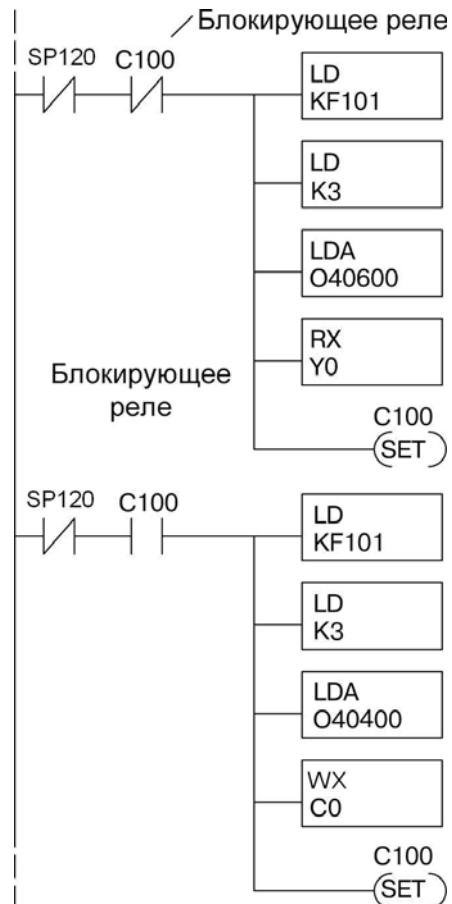
Бит «Ошибка связи» включается, когда ПЛК обнаруживает ошибку. Использование этого бита не обязательно. Используйте его впереди каждой из сетевых команд, поскольку бит ошибки сбрасывается при выполнении команд RX или WX.

Блокировки многократного чтения и записи

Если вы применяете многократные чтения и записи в программе RLL, то вы должны взаимно блокировать подпрограммы, чтобы обеспечить их корректное выполнение. Если вы не применяете взаимоблокировки, то ПЛК выполнит только первую цепь программы. Это происходит потому, что порт может одновременно управлять только одной транзакцией.

На примере справа после выполнения команды RX устанавливается реле C100. Когда порт завершает коммуникационную задачу, выполняется вторая цепь программы и C100 сбрасывается.

Если вы используете стадийное программирование RLL^{PLUS}, то вы можете включить каждую цепь программы в отдельную стадию программы для ее корректного выполнения и переходить с одной стадии на другую, давая возможность только одной из цепей программы быть активной в определенный момент времени.



Работа DL06 ведущим устройством: команды MRX/MWX



ПРИМЕЧАНИЕ: Для использования команд MRX/MWX необходимо использовать контроллер DL06 и DirectSOFT5 (v. 5.2 или новее).

Чтение по сети MODBUS (MRX)

Команда чтения из сети MODBUS (MRX) используется ведущим DL06 для чтения блока данных из памяти ведомого устройства и записи в V-память ведущего DL06. Команда позволяет пользователю указать код функции MODBUS, адрес ведомого устройства, начальные адреса ведущего и ведомого устройства, количество данных для передачи, формат данных MODBUS и адрес буфера ошибок выполнения.

- **CPU/DCM:** Выберите DCM порт
- **Slot Number:** Укажите номер слота, в котором установлен D0-DCM
- **Slave Address:** Укажите адрес ведомого устройства (0-247)
- **Function Code:** Следующие коды функций MODBUS поддерживаются командой MRX:
 - 01 – чтение группы реле (Read Coil Status)
 - 02 – чтение группы входов (Read Input Status)
 - 03 – чтения регистра хранения (Read Holding Registers)
 - 04 – чтение входных регистров (Read Input Registers)
 - 07 – чтение регистров состояния (Read Exception Status)
- **Start Slave Memory Address:** Определите начальный адрес памяти ведомого устройства, начиная с которого данные должны быть считаны.
- **Start Master Memory Address:** Определите начальный адрес памяти ведущего устройства, где будут размещены полученные данные.
- **Number of Elements:** Укажите количество реле, входов, регистров, которые необходимо прочитать.
- **Modbus Data Format:** Определите, будет ли использоваться формат MODBUS 584/984 или формат 484.
- **Exception Response Buffer (Буфер ошибок выполнения):** Укажите адрес памяти ведущего устройства, куда будет помещены ошибки выполнения (Exception Response). Необходимо 3 регистра.

Адрес памяти ведомого устройства для команды MRX

Диапазон адресов ведомого устройства MRX		
Код функции	Формат данных MODBUS	Диапазон(ы) адресов ведомого устройства
01 - Чтение состояния реле	Режим 484	1-999
01 - Чтение состояния реле	Режим 584/984	1-65535
02 - Чтение состояния входа	Режим 484	1001-1999
02 - Чтение состояния входа	Режим 584/984	10001-19999 (5 цифр) или 100001-165535 (6 цифр)
03 - Чтение регистра хранения	Режим 484	4001-4999
03 - Чтение регистра хранения	Режим 584/984	40001 -49999 (5 цифр) или 400001-465535 (6 цифр)
04 - Чтение входных регистров	Режим 484	3001-3999
04 - Чтение входных регистров	Режим 584/984	30001 -39999 (5 цифр) или 300001-365535 (6 цифр)
07 - Чтение регистров состояния (Read Exception Status)	484 и 584/984	Нет

Адреса памяти ведущего устройства для команды MRX

Диапазон адресов ведущего устройства для команды MRX		
Тип данных операндов		Диапазон DL06
Входы	X	0-1777
Выходы	Y	0-1777
Управляющие реле	C	0-3777
Биты стадий	S	0-1777
Биты таймеров	T	0-377
Биты счетчиков	CT	0-377
Специальные реле	SP	0-777
V-память	V	вся
Удаленные (Глобальные) входы	GX	0-3777
Удаленные (Глобальные) выходы	GY	0-3777

Число элементов в команде MRX

Число элементов		
Тип данных операнда		Диапазон DL06
V-память	V	Вся память
Константа	K	Биты: 1-2000 Регистры: 1-125

Буфер ошибок выполнения команды MRX

Буфер ошибок выполнения		
Тип данных операнда		Диапазон DL06
V-память		Вся память

Запись по сети MODBUS (MWX)

Команда записи по сети MODBUS (MWX) используется для записи блока данных из ведущего DL06 в память ведомого устройства. Команда позволяет пользователю указать код функции MODBUS, адрес ведомого устройства, начальные адреса ведущего и ведомого устройства, количество данных для передачи, формат данных MODBUS и Буфер ошибок выполнения.

- **CPU/DCM:** Выберите DCM порт
- **Slot Number:** Укажите номер слота, в котором установлен D0-DCM
- **Slave Address:** Укажите адрес ведомого устройства (0-247)
- **Function Code:** Следующие коды функций MODBUS поддерживаются командой MRX:
 - 01 – Чтение группы реле (Read Coil Status)
 - 02 – Чтение группы входов (Read Input Status)
 - 03 – Чтение регистра хранения (Read Holding Registers)
 - 04 – Чтение входных регистров (Read Input Registers)
 - 07 – Чтение регистров состояния (Read Exception Status)
- **Start Slave Memory Address:** Определите начальный адрес памяти ведомого устройства, начиная с которого данные должны быть считаны.
- **Start Master Memory Address:** Определите начальный адрес памяти ведущего устройства, где будут размещены полученные данные.
- **Number of Elements:** Укажите количество реле, входов, регистров, которые необходимо прочитать.
- **Modbus Data Format:** Определите, будет ли использоваться формат MODBUS 584/984 или формат 484.
- **Exception Response Buffer (Буфер ошибок выполнения):** Укажите адрес памяти ведущего устройства, куда будет помещены ошибки выполнения (Exception Response). Необходимо 3 регистра.

Адрес памяти ведомого устройства для команды MWX

Диапазон адресов ведомого устройства для команды MWX		
Код функции	Формат данных MODBUS	Диапазон(ы) адресов ведомого устройства
05 – Форсирование одного реле	Режим 484	1-999
05 – Форсирование одного реле	Режим 584/984	1-65535
06 – Установка одного регистра	Режим 484	4001-4999
06 – Установка одного регистра	Режим 584/984	40001 -49999 (5 цифр) или 400001-465535 (6 цифр)
15 – Форсирование нескольких реле	Режим 484	1-999
15 – Форсирование нескольких реле	Режим 584/984	1-65535
16 – Установка нескольких регистров	Режим 484	4001-4999
16 – Установка нескольких регистров	Режим 584/984	40001 -49999 (5 цифр) или 400001-465535 (6 цифр)

Адреса памяти ведущего устройства для команды MWX

Диапазон адресов ведущего устройства для команды MWX		
Тип данных операндов	Диапазон DL06	
Входы	X	0-1777
Выходы	Y	0-1777
Управляющие реле	C	0-3777
Биты стадий	S	0-1777
Биты таймеров	T	0-377
Биты счетчиков	CT	0-377
Специальные реле	SP	0-777
V-память	V	Вся память
Удаленные входы	GX	0-3777
Удаленные выходы	GY	0-3777

Число элементов в команде MWX

Число элементов		
Тип данных операнда	Диапазон DL06	
V-память	V	Вся память
Константа	K	Биты: 1-2000 Регистры: 1-125

Буфер ошибок выполнения команды MWX

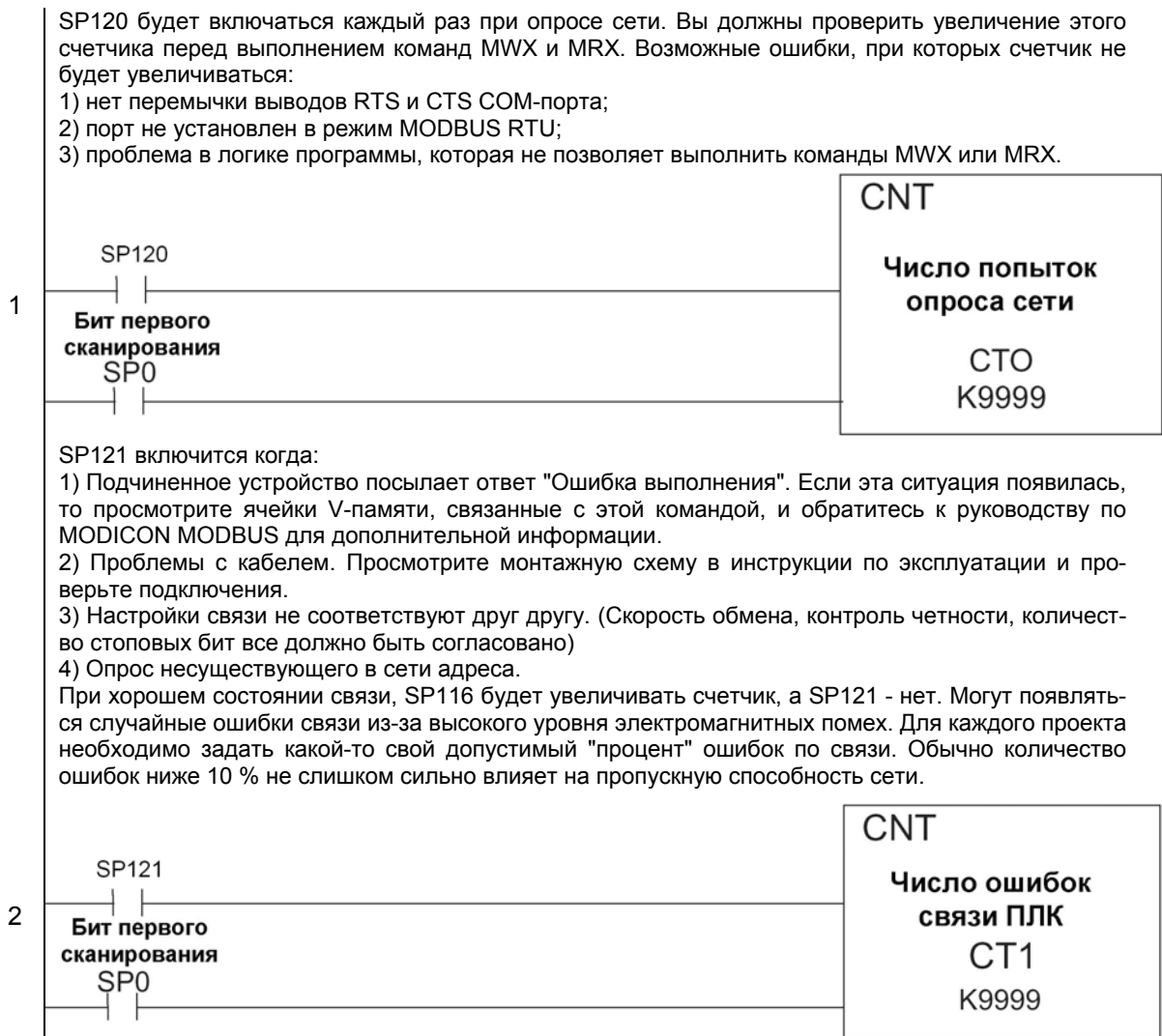
Буфер ошибок выполнения		
Тип данных операнда	Диапазон DL06	
V-память	V	Вся память

Пример использования команд MWX и MRX с блокировкой

Порт 2 D0-DCM имеет два, связанных с ним, специальных контакта реле (См. страницу 5-12). Одно реле указывает «Порт занят», другое – «Ошибка связи». Бит «Порт занят» включен, когда ПЛК связывается с ведомым устройством. Если этот бит выключен, то программа может инициировать следующий запрос к сети. Бит – «Ошибка связи» включается, когда ПЛК обнаруживает ошибку. При использовании этот бит должен проверяться перед любым обращением к сети, так как бит ошибки сбрасывается после того, как команда MRX или MWX выполнена. Обычно связь по сети продолжается дольше одного цикла сканирования процессора. Программа должна ожидать окончания сеанса связи перед стартом следующей транзакции.

Блокировки многократного чтения и записи

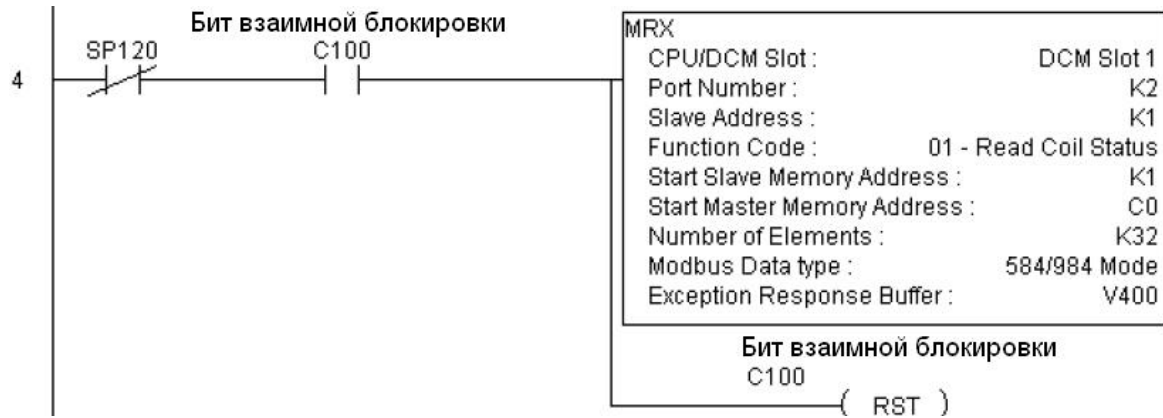
Если Вы используете несколько команд чтения и записи в программе релейной логики, то Вам необходимо взаимно блокировать команды, чтобы обеспечить их корректное выполнение. Это связано с тем, что порт одновременно может управлять только одной передачей данных. В примере, приведенном ниже, после выполнения инструкции MRX устанавливается бит управляющего реле C100. Когда порт заканчивает выполнение коммуникационной задачи, выполняется следующая команда и C100 сбрасывается. Если Вы используете стадийное программирование RLL^{PLUS}, вы можете включить каждую команду чтения и записи в отдельную стадию программы, давая возможность только одной цепи из программ быть активной в определенный момент времени.



Эта цепь делает запись по сети MODBUS в первый регистр хранения 40001 ведомого устройства с адресом 1. Записываемое значение хранится в ячейке V2000. Функциональный код (06) записывает только в один регистр. Используйте функциональный код 16, чтобы записать в группу регистров. Только одна сетевая команда (WX, RX, MWX, MRX) может быть выполнена в одном цикле сканирования. В этом причина необходимости использования битов для взаимной блокировки. При использовании многих сетевых команд на одном порту используйте команду сдвига регистра (*Shift Register*).



Эта ветвь выполняет чтение сети MODBUS из первых 32-х реле ведомого устройства, начиная с адреса 1. Данная команда последовательно разместит 32 значения реле в битовую память ведущего, начиная с C0.



Глава 6. Использование команд IPv6 в коммуникационных программах

В этой главе...

- Команды конфигурации сети NETCFG
- Команда NETRX
- Команда NETWX

Команда конфигурации сети NETCFG



NETCFG IB-700

ПРИМЕЧАНИЕ: Для использования команд IBOX необходимо использовать *DirectSOFT5 (v. 5.2 или новее)* и контроллеры *DL05 с ЦПУ v.5.10 или новее, DL06 с ЦПУ v.2.10 или новее*. Фирменное ПО ЦПУ контроллеров можно получить на сайте: www.automationdirect.com

Команда конфигурации сети (*Network Config*) определяет всю общую информацию необходимую для осуществления чтения и записи из сети с использованием команд IBox - NETRX и NETWX через D0-DCM.

Эта команда должна находиться в самом начале программы контроллера, вместе с другими конфигурационными инструкциями.

Если в вашем контроллере более одного локального последовательного порта, DCM или ECOM, то следует использовать отдельные команды конфигурирования для каждой RX/WX сети, которые используют инструкции NETRX/NETWX.

Ячейка памяти, задаваемая параметром WORKSPACE (Рабочая область) в команде NETCFG, привязывается только к этой команде, и не должна использоваться еще где либо в программе.

Параметр «CPU Port or Slot» указывает номер слота, в котором установлен D0-DCM.

Параметры команды NETCFG:

- **Network#:** уникальный номер для каждой DCM или ECOM сети.
- **CPU Port or Slot:** номер порта CPU или номер слота, в который установлен модуль.
- **Workspace:** адрес ячейки памяти, который будет использован командой.

Параметр	Диапазон DL05 / DL06
Network# K	Константа: K0-255
CPU Port or Slot K	Константа: K0-FF
Workspace V	См. карту памяти DL - Слова данных

Команда Чтения по сети NETRX

NETRX IB-701 Эта команда выполняет инструкцию RX (чтения) вместе с другими инструкциями чтения и записи Network RX (NETRX), Network WX (NETWX)) в Вашей программе и создана для упрощения организации сетевых коммуникаций.

Инструкция RX (чтения) выполняется для указанной в параметре Network# сети, которая сконфигурирована соответствующей командой Network Configuration (NETCFG) в программе ранее.

Network RX Read	
NETRX	IB-701
Network #	K0
Workspace	V400
Slave ID	K1
From Slave Element (Src)	C0
Number Of Bytes	K1
To Master Element (Dest)	TA0
Success	C0
Error	C0

Ячейка памяти, задаваемая параметром WORKSPACE, является внутренним регистром и привязывается только в этой команде, и не должна использоваться еще где-либо в программе.

При работе команды NETRX выполняется чтение элемента данных из указанного ведомого устройства-источника (Src) в заданный блок памяти (Dest), при этом могут быть выполнены и другие команды Network RX и Network WX, связанные с этой же сетью Network#.

Например, если Вы хотите считыв вы можете поместить все 5 этих инструкций в одну цепь программы с одним условием SP1 (всегда On). Тогда они все будут выполняться одна за другой автоматически.

Параметры NETRX:

- **Network#**: логический номер сети, организованной на базе одного последовательного порта CPU, модуля DCM или ECOM и ранее определенной командой NETCFG.
- **Workspace**: адрес ячейки памяти, который будет использован командой.
- **Slave ID**: номер ведомого контроллера для инструкции NETRX.
- **From Slave Element (Src)**: адрес ведомого источника данных для чтения.
- **Number of Bytes**: число читаемых байт.
- **To Master Element (Dest)**: адрес ячейки памяти ведущего приемника для сохранения читаемых данных.
- **Success**: адрес бита, который будет установлен при успешном выполнении команды.
- **Error**: адрес бита, который будет установлен при неудачном выполнении команды.

Параметр	Диапазон DL05 / DL06
Network# K	Константа: K0-255
Workspace V	См. карту памяти DL - Слова данных
Slave ID. K	K0 - 90
From Slave Element (Src) X,Y,C,S,T,CT,GX,GY,V	См. карту памяти DL
Number of Bytes K	См. карту памяти DL - Слова данных
To Master Element (Dest). V	См. карту памяти DL - Слова данных
Success X,Y,C,GX,GY,B	См. карту памяти DL
Error X,Y,C,GX,GY,B	См. карту памяти DL

Команда Записи по сети NETWX

NETWX IB-702

Эта команда выполняет инструкцию WX (записи) вместе с другими инструкциями чтения и записи Network RX (NETRX), Network WX (NETWX)) в Вашей программе и создана для упрощения организации сетевых коммуникаций.

Инструкция WX (чтения) выполняется для указанной в параметре Network# сети, которая сконфигурирована соответствующей командой *Network Configuration* (NETCFG) в программе ранее.

Ячейка памяти, задаваемая параметром WORKSPACE, является внутренним регистром и привязывается только к этой команде, и не должна использоваться еще где либо в программе.

При работе команды NETWX выполняется запись данных из буфера в памяти ведущего ПЛК в заданный блок памяти ведомого ПЛК, при этом могут быть выполнены и другие команды Network RX и Network WX, связанные с этой же сетью Network#.

Например, если Вы хотите считывать и записывать данные непрерывно из пяти различных ведомых источников, Вы можете поместить все 5 этих инструкций в одну цепь программы с одним условием SP1 (всегда On). Тогда они все будут выполняться одна за другой автоматически.

Параметры NETWX:

- **Network#**: логический номер сети, организованной на базе одного последовательного порта CPU, модуля DCM или ECOM и ранее определенной командой NETCFG.
- **Workspace**: адрес ячейки памяти, который будет использован командой.
- **Slave ID**: номер ведомого контроллера для инструкции NETRX.
- **From Slave Element (Src)**: адрес ведомого источника данных для чтения.
- **Number of Bytes**: число читаемых байт.
- **To Master Element (Dest)**: адрес ячейки памяти ведущего приемника для сохранения читаемых данных.
- **Success**: адрес бита, который будет установлен при успешном выполнении команды.
- **Error**: адрес бита, который будет установлен при неудачном выполнении команды.

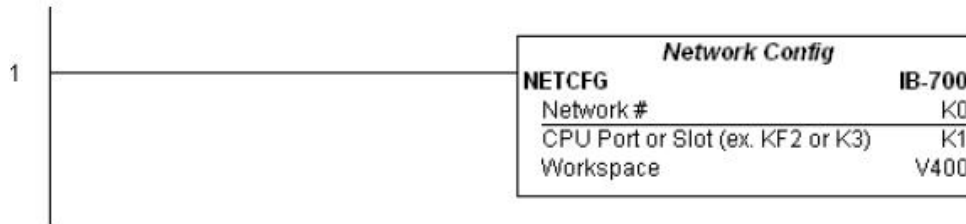
Параметр	Диапазон DL05 / DL06
Network# K	Константа: K0-255
Workspace V	См. карту памяти DL - Слова данных
Slave ID. K	K0 - 90
From Master Element (Src) X,Y,C,S,T,CT,GX,GY,V	См. карту памяти DL
Number of Bytes K	См. карту памяти DL - Слова данных
To Slave Element (Dest). V	См. карту памяти DL - Слова данных
Success X,Y,C,GX,GY,B	См. карту памяти DL
Error X,Y,C,GX,GY,B	См. карту памяти DL

Пример использования команд NETCFG, NETRX и NETWX

Цель 1: Инструкция *Network Configuration* (NETCFG) координирует все взаимодействия с другими коммуникационными инструкциями (NETRX/NETWX). Для каждой сети на основе последовательного порта, DCM или ECOM модуля должна быть отдельная инструкция *Network Configuration*. Все они должны находиться в самом начале программы и выполняться в каждом программном цикле.

Эта команда NETCFG определяет номер сети K0 для D0-DCM, установленного в слот 1.

Регистр Workspace используется для хранения информации о состоянии порта или модуля, распределяя аппаратный ресурс между командами NETRX и NETWX. Ячейка памяти, занимаемая этим регистром не должна использоваться где-либо еще в программе.



Цель 2: Используя сеть с номером K0, выполняется чтение ячеек X0-X7 из ведомого источника с номером K7, и максимально быстро записывается в ведомый приемник с номером K5. Прочитанные данные сохраняются в локальном контроллере в C200-C207, и записываются в приемнике K5 в ячейки C300-C307.

Обе команды NETRX и NETWX работают совместно с инструкцией *Network Config* чтобы упростить сетевые обращения через один ресурс. Они также предоставляют упрощенные отчеты об ошибках. Теперь нет необходимости беспокоиться о проверке состоянии занятости и статуса ошибки ("busy bits", "error bits"), или помнить о номере слота модуля, и заводить счетчики или регистры сдвига для организации взаимодействия с модулем.

В этом примере SP1 (всегда On) управляет двумя командами NETRX и NETWX в одной цепи программы. В одном скан-цикле как только заканчивает выполнение команда чтения, начинается выполнение команда записи. Как только команда записи завершится, все незаконченные операции ниже в программе будут выполнены. Если нет незаконченных операций NETRX и NETWX под этой командой, то непосредственно в следующий скан-цикл команда NETRX снова начнет свой запрос.

