

Обмен данными с устройствами по EtherNet/IP в ПЛК Do-more

В версиях программного обеспечения контроллеров Do-more, начиная с 1.4.1 реализован протокол EtherNet/IP (ODVA) с поддержкой функций Client (Master) и Server (Slave), использующих Explicit messages (Явные, подробные сообщения), на встроенном порту Ethernet.

- Для настройки и оптимизации функции EtherNet/IP Server необходимо использовать только окно настроек Сервера: **PLC>> System Configuration>> CPU Configuration >> EtherNet/IP Settings>> Setup EtherNet/IP Explicit message Server.**
- Для реализации функции EtherNet/IP Client предназначена команда **Send EtherNet/IP Message (EIPMSG).**

EtherNet/IP Explicit Message Server

EtherNet/IP Server может поддерживать до 16 одновременных сессий с устройствами *Explicit Unconnected EtherNet/IP Client*. В конфигурации ЦПУ по умолчанию драйвер устройства *EtherNet/IP Explicit Message Server* отключен. Эту функцию можно активировать в окне «CPU Configuration»

Setup EtherNet/IP Explicit Message Server

Server Settings

Max Concurrent Sessions: 1 1 - 16 Device Name: Do-more EtherNet/IP UCMM Servi (max 32)

Client Inactivity Timeout: 60 seconds NOTE: Server status is reported through the structure named \$IntEIPServer.

TCP Port 44818 (default is 44818)

☒ Block 1 ☒ Block 2 ☐ Block 3 ☐ Block 4 ☐ Block 5 ☐ Block 6 ☐ Block 7 ⇌

Access Control

☒ Enable Reads (Get Attribute Single - Service Code: 0x0E)
☐ Enable Writes (Set Attribute Single - Service Code: 0x10)

Do-more Mapped Memory

Element Type: N
Starting Element: 1
Element Count: 1
Mapped Range: N1-N1 Size: 2 bytes

Explicit Message Assembly Instance

Class: 4 (0x4)
Instance: 101 (0x65)
Attribute: 3 (0x3)

CONFIGURATION NOTES:

* For bit data types, the start and length must be multiples of 8.
* The total message size cannot exceed 500 bytes.

OK Cancel

Server Settings - Настройки сервера

- **Maximum Concurrent Sessions (1 - 16)** - Значение этой константы определяет, сколько одновременных подключений может обрабатывать драйвер устройства «Ethernet / IP Server». Создание сессии EtherNet / IP требует время для обработки, что неизбежно повлияет на время цикла сканирования ПЛК.

Если появятся EtherNet / IP Clients, запрашивающие число соединений, больше чем допустимое число одновременных сессий, то часть этой обработки включает в себя закрытие старейшей сессии, так чтобы новая сессия могла быть открыта. Необходимость принудительно закрывать сессии, можно свести к минимуму путем установки этого значения до максимального числа клиентов EtherNet / IP, которые будут в соединения с этим PLC в любой момент времени.

- **Client Inactivity Timeout (seconds)** – Указывает время ожидания (в секундах) до закрытия соединения с EtherNet/IP Client, который прекратил обмен данными (диапазон 0 – 65535).

- **TCP Port Number (44818 is default)** – Номер порта TCP используемый EtherNet/IP Explicit Message Server. Значение по умолчанию – 44818 (диапазон 0 – 65535).

- **Device Name** – Имя устройства длиной до 32-х символов, которое возвращается в ответ на запрос об «Identity Class».

Состояние Сервера во время работы можно контролировать при помощи встроенной структуры **\$IntEIPServer**, которая содержит следующие члены (*members*):

- .ActiveSessions** - (read-only) – число одновременно открытых соединений с Клиентами EtherNet/IP.
- .LastError** – Последняя ошибка посланная Клиенту EtherNet/IP.
- .Errors** – Общее число ошибок, посланное всем Клиентам EtherNet/IP.
- .Transactions** - Общее число завершенных запросов Клиентов EtherNet/IP.

Data Blocks – Блоки данных

До 8-ми Блоков данных могут быть сделаны доступными для Клиентов EtherNet/IP Clients. Здесь надо указать число Блоков данных и сконфигурировать каждый блок, заполнив поля в секциях: *Access Control, Do-more Mapped Memory, Explicit Messaging Assembly Instance*

- Access Control – Управление доступом

Enable Reads (Get Single Attribute - Service Code: 0x0E) – позволяет Клиентам EtherNet/IP читать данные этого блока, используя один Атрибут - *Get Single Attribute*.

Enable Writes (Set Single Attribute - Service Code: 0x10) - позволяет Клиентам EtherNet/IP писать данные в этот блок, используя один Атрибут - *Set Single Attribute*.

- Do-more Mapped Memory – Адреса памяти ПЛК Do-more

Эти адреса будут доступны при получении запроса (Class/Instance/Attribute).

Element Type – Тип Элемента цифровой (*numeric*) памяти используемый в Блоке.

Starting Element – Первый Элемент памяти, используемый в Блоке

Element Count – Число используемых в Блоке последовательных Элементов памяти Do-more. Максимальный размер в запросе EtherNet/IP – 500 байт, поэтому максимальное число Элементов в блоке зависит от Типа Элемента.

Element Type	Макс. число в Data Block
Bit	4000
Byte	500
Word	250
DWord	125
Real	125

Mapped Range – Отображает выбранный диапазон Элементов.

Size - Отображает размер выбранного диапазона Элементов в байтах.

- Explicit Messaging Assembly Instance – Отображает Путь - Path (Class / Instance / Attribute) сконфигурированного Блока данных.

Class – Класс всех этих блоков фиксирован - 0x04 (assembly class).

Instance – Каждому из 8 блоков присвоена уникальный *Instance* (номер экземпляра), как показано в таблице:

Attribute - Атрибут всех блоков фиксирован - 0x03.

Block Number	Instance ID
0	101 (0x65)
1	102 (0x66)
2	103 (0x67)
3	104 (0x68)
4	105 (0x69)
5	106 (0x6A)
6	107 (0x6B)
7	108 (0x6C)

Configuration Notes – Примечания относящиеся к конфигурированию Блоков данных.

EtherNet/IP Explicit Message Server поддерживает следующие Пути - **Paths:**
(**Class / Instance / Attribute**)

Class	Instance	Attribute	Описание
0x01 - Identity	1	1	Vendor ID
		2	Device Type
		3	Product Code
		4	Revision
		6	Serial Number
		7	Product Name
0x04 - Assembly	101 (0x65)	3	Data Block 1
	102 (0x66)	3	Data Block 2
	103 (0x67)	3	Data Block 3
	104 (0x68)	3	Data Block 4
	105 (0x69)	3	Data Block 5
	106 (0x6A)	3	Data Block 6
	107 (0x6B)	3	Data Block 7
	108 (0x6C)	3	Data Block 8
0xF5 - TCP/IP	1	1	Status
		3	Configuration Control
		3	Product Code
		4	Physical Link
		5	Interface Configuration
0xF6 - Ethernet Link	1	2	Interface Flags
		3	Physical Address

EtherNet/IP Explicit Message Client

Соединение по EtherNet /IP использует объектно-ориентированный протокол *Common Industrial Protocol* (CIP), на верхнем уровне. Каждый объект CIP имеет Атрибуты (данные), Сервисы (команды) и Поведения (реакции на события). На прикладном уровне протокола CIP, каждое сетевое устройство выступает в качестве набора объектов. Объект является группой смежных значений данных в устройстве. Объекты позволяют пользователю организовать данные, которые специфичны для конкретного вида устройства.

Объекты прикладного уровня предопределены для большого числа типов устройств. Один и тот же тип устройств CIP должен содержать ту же самую последовательность прикладных объектов. Последовательность Прикладных объектов конкретного типа устройства известна, как профиль устройства. Объекты не находящиеся в профиле класса устройств называются специфическими - *Vendor Specific*. Эти объекты поставщик (*Vendor*) включает в качестве дополнительных свойств устройства. Протокол CIP предоставляет доступ к этим объектам поставщика точно так же, как к другим объектам. Команда **Send EtherNet/IP Message** получает доступ к данным по сети к устройству через **Object** (или **Class**) **Number**, **Instance Number** (номер экземпляра объекта) и, необязательно, **Attribute Number**.

«**Instance**» является способом работы с данными одного рода, например, для совместного использования одного Атрибута.

Send EtherNet/IP Message (EIPMSG)

Команда посылает сообщение серверу -> *EtherNet/IP Explicit Message Server*. Применение этой команды возможно только при использовании Do-more Technology версии 1.4 и последующих.

Команда **Send EtherNet/IP Message (EIPMSG)** реализует функции Клиента - **Explicit Unconnected EtherNet/IP Client** на встроенном порту Ethernet процессора Do-more. Этот Клиент инициирует обмен данными «запрос / ответ» с Серверами EtherNet/IP.

The screenshot shows the 'Send EtherNet/IP Message' configuration window. Key settings include:

- Device:** @IntEIPClient
- IP Address:** Fixed, 10.0.0.189
- TCP Port:** 44818
- Path:** Class 0xA0, Instance 0x1, Use Attribute 0x1
- Service:** Specific Service, Get Attribute (14, 0x0E)
- Enable:** Once on Leading Edge
- On Success:** JMP to Stage MyProgram.S0
- On Error:** JMP to Stage MyProgram.S10
- Right-hand section:** Includes checkboxes for 'Use Request Service Data Buffer' and 'Use Response Service Data Buffer', with fields for 'Req is String Structure', 'Req is Numeric Data Block', 'Res is String Structure', and 'Res is Numeric Data Block'. It also includes 'General Status Code' and 'Extended Status' sections.

Примечание по заполнению: Используйте клавишу **F9 key**, чтобы открыть Инструмент выбора элементов (Element Picker или Element Browser) или используйте клавишу **Down-Arrow** на поле параметра.

Device – выбор одного из устройств (device) EtherNet/IP Explicit Message Client заранее сконфигурированное на встроенном порту Ethernet ПЛК. По умолчанию, EtherNet/IP Explicit Message Client именован - **@IntEIPClient**. Создание нового клиента производится в Device Configuration.

Примечание: Этот выбор требует установки TCP соединение с EtherNet/IP Explicit Message Server до начала коммуникации. Если необходимо соединятся с несколькими EtherNet/IP Server (Серверы с различными IP-адресами), то мы рекомендуем создать отдельное устройство EtherNet/IP Explicit Message Client для каждого EtherNet/IP Servers. Это уменьшит дополнительные издержки TCP для управления соединением с несколькими Серверами. Каждый раз, при использовании одного устройства EtherNet/IP Client Device для нескольких IP-адресов, будет появляться сообщение M240 в окне «Output Window»

IP Address - IP Адрес EtherNet/IP Server (Slave) к которому будут направлять сообщения. Это может быть **Fixed** (статический) IP Адрес или Адрес DWord переменной Variable, которая содержит IP Адрес в виде шестнадцатеричного эквивалента каждого числа адреса (Например, IP Address 192.168.100.006 в Hex будет C0.A8.64.06, а в Variable Address будет храниться, как 0xC0A86406). Чтобы увидеть IP Адрес, хранящийся в DWord, в окне Data View выберите формат 'IP Address'. Тогда Вы увидите адрес в традиционной десятичной форме (000.000.000.000).

TCP Port Number – Номер порта для приема сообщений EtherNet/IP Server (Slave). По умолчанию этот номер для протокола EtherNet/IP: 44818

Path – Путь указывает параметры для запросов. Конкретные значения полей предоставляются производителем EtherNet/IP. Это могут быть любые положительные значения или числовые ячейки памяти.

Class - Значение идентификатора - Class ID.

Instance - Значение идентификатора - Instance ID.

Use Attribute – Активация этого выбора позволит указывать Значение Атрибута.

Service – Службы указывают действие производимое с набором объектов в запросе. Можно выбрать из представленного ниже списка Служб, или выбрать Generic и ввести номер Службы. Список предопределенных Служб (**Specific Service**):

Get Single Attribute (14, 0x0E) – Запрос одного Атрибута

Set Single Attribute (16, 0x10) - Запись одного Атрибута

Get All Attributes (1, 0x01) - Запрос всех Атрибутов

Set All Attributes (2, 0x02) - Запись всех Атрибутов

Generic – Укажите номер Службы, которой HET в списке предопределенных служб.

Enable – Выберите вариант выполнения команды:

- **Once on Leading Edge** – Один раз по переднему фронту.
- **Continuous on Power Flow at Interval** – Непрерывно (когда активирована) с заданным постоянным или переменным интервалом (**Constant/ Variable**)

Действия при успешном - **On Success** или неуспешном - **On Error** завершении команды

Можно выбрать: **Set Bit** или **JMP to Stage** (В том же самом Program code-block).

Можно установить счетчики завершения команды: **On Success Counter** и **On Error Counter** – и указать адрес DWord для хранения общего числа событий.

Кнопка **Create Data Block** - Если нужного блока нет или необходим дополнительный Блок Данных, нажмите эту кнопку.

The screenshot shows the 'Create Data Block' dialog box. The 'Data Block Name' is 'PowerV'. The 'Number of Elements' is '256'. The 'Unsigned Word Data Block Range' is 'PowerV0 - PowerV255'. The 'Make data block retentive' checkbox is checked. The 'Data Type' list shows 'Unsigned Word' selected. The 'Maximum String Length' is '32'. The 'OK', 'Cancel', and 'Help' buttons are at the bottom.

Data Block Name (1 to 16 letters) – Уникальное имя Блока Данных длиной от 1 до 16 символов (A-Z, a-z; Без чисел, Без пробелов).

Number of Elements – Число Элементов в Блоке Данных. Блоки Данных должны создаваться внутри границ DWord (4-байта). Максимальное число Байт, в одном пакете - 1024.

Unsigned Word Block Range – Отображение Типа данных и первого / последнего Элементов созданного Блока Данных.

Data Type – Тип данных в Блоке (на рисунке - Unsigned Word).

Make Data Block Retentive (retain values after power loss) – Блок данных помеченный, как «*retentive*» будет сохранять свое состояние при выключении и включение питания или при переходе режима работы *Program* ->*Run*. Без этой пометки Блок будет очищаться при включении питания и переходе *Program* ->*Run*.

Use Request Service Data Buffer – Этот выбор устанавливается автоматически при выборе «*Set Attribute service*» и сбрасывается при выборе «*Get Attribute service*». Для *Generic Service* выбор может быть изменен в любое время.

- **Req is String Structure** – Этот вариант используется, когда *Set Attribute service* или *Generic service* содержатся в String, введите *String*, которые будет использован. Это может быть определенный системой *Short Strings*, *Long Strings*, или определенный пользователем *Strings*. Максимальная длина *String* - 500 байт.
- **Req is Numeric Data Block** - Этот вариант используется, когда *Set Attribute service* или *Generic service* содержатся в числовом блоке памяти. Максимальный размер блока данных, который может быть послан в одной «Службе» требует 500 байт (250 Words, 125 DWords, 125 Reals).

Req Start – Первый элемент блока памяти, которая служит данными для *Set Attribute* или *Generic service*.

Req Number of Bytes – Число последовательных BYTE для *Set Attribute* или *Generic service* (Words = 2 Bytes, DWord = 4 Bytes, Real = 4 Bytes).

Use Response Service Data Buffer - Этот выбор устанавливается автоматически при выборе «*Get Attribute service*» и сбрасывается при выборе «*Set Attribute service*». Для *Generic Service* выбор может быть изменен в любое время.

- **Res is String Structure** – Этот вариант используется, чтобы сохранить данные из *Get Attribute* или *Generic service* в переменных типа *String*, введите *String*, которые будет использован. Это может быть определенный системой *Short Strings*, *Long Strings*, или определенный пользователем *Strings*. Максимальная длина *String*, которая может быть получена - 500 байт.
- **Res is Numeric Data Block** - Этот вариант используется, чтобы сохранить данные из *Get service* или *Generic service* в числовом блоке памяти. Максимальный размер блока данных, который может быть прочитан в одной «Службе» требует 500 байт.

Res Start - Первый элемент блока памяти, который будет хранить данные возвращенные *Get Attribute* или *Generic service*. Это может быть любая числовая ячейка, в которую можно писать.

Res Length in BYTES – Адрес памяти для хранения реального числа байт данных, которые были возвращены *Get Attribute* или *Generic service*.

Res Max Length in BYTES – Максимальное число байт возвращаемых данных для хранения их в блоке данных. Это может быть константа от 1 до 500 или ячейка числовой памяти.

General Status Code – Позволяет хранить значение состояния возвращаемое из Сервера EtherNet/IP в ответ на запрос - Service Request. Это значение может показывать код успешного завершения или ошибки, интерпретация кода находится в документации на EtherNet/IP Server.

Extended Status - Позволяет хранить значение любого расширенного состояния возвращаемое из Сервера EtherNet/IP.

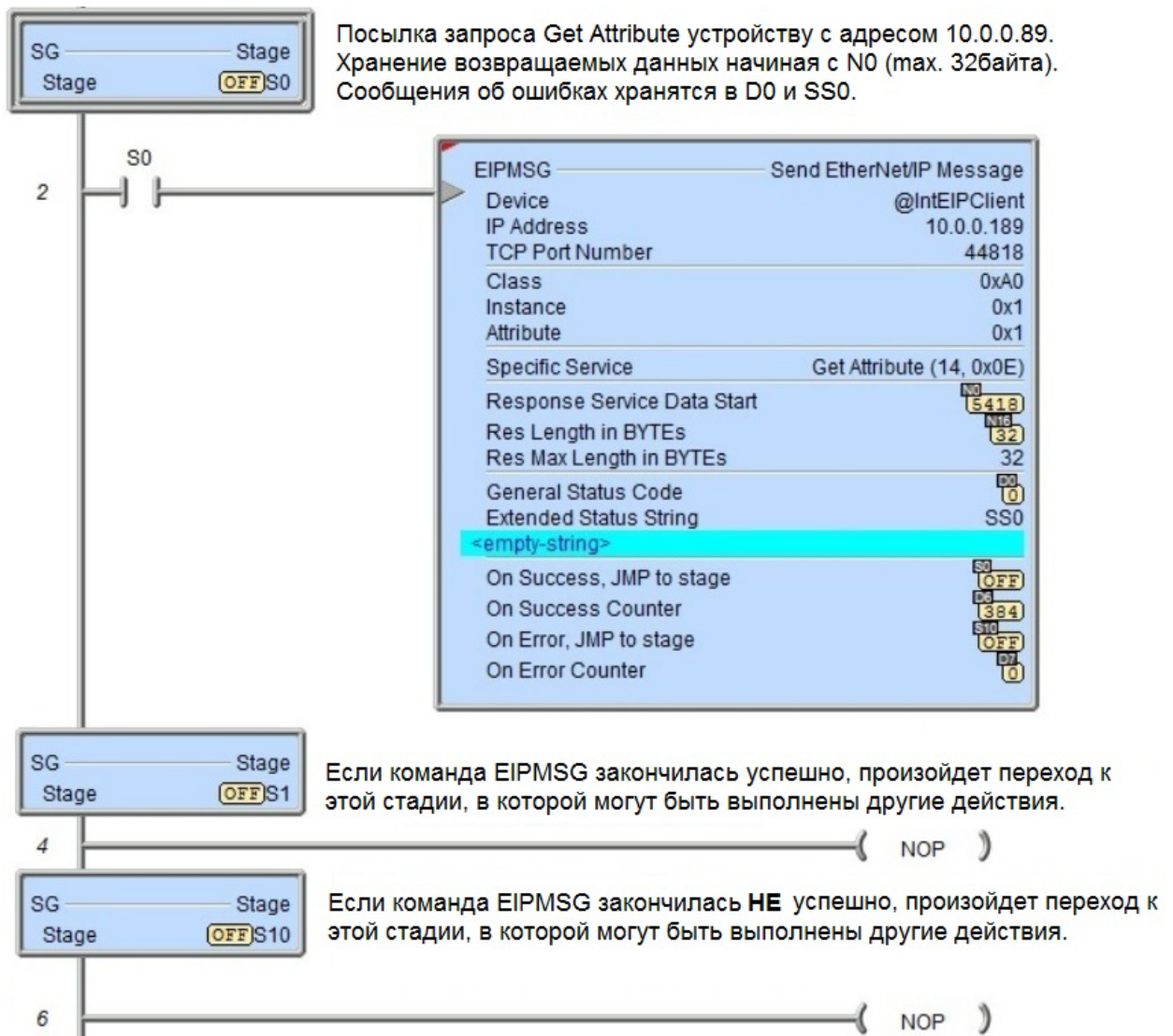
- **Ext is String Structure** – Хранение Extended состояния в *Strings*.
- **Ext is Numeric Data Block** - Хранение Extended состояния в Status в числовых блоках данных.

Ext Start – Первый элемент числового блока данных для хранения значений состояния.

Ext Length in BYTES – Адрес памяти для хранения актуального числа байтов Extended Status.

Ext Max Length in BYTES – Максимальное число BYTE значений Extended Status в блоке данных. (от 1 до 500, числовая ячейка памяти).

Пример программы с использованием команды EIPMSG с применение Стадий.



Красный треугольник в верхнем левом углу команды показывает, что эта команда *Fully Asynchronous* – Полностью асинхронная.