

Связь TCP/IP на основе Ethernet

PMP20

Руководство по эксплуатации



PROM
POWER

Введение

Общее описание

Благодарим вас за покупку ПЛК с поддержкой Ethernet PROMPOWER. В настоящем руководстве в основном представлена информация о сетевом обмене данными по Ethernet. Перед использованием изделия внимательно прочтите данное руководство и подключайте провода только после полного понимания содержащейся в нем информации.

Пожалуйста, передайте это руководство конечному пользователю.

Информация для пользователей

Проводить электромонтажные и другие работы с изделием могут только операторы, обладающие определенными знаниями в области электротехники. В случае каких-либо вопросов обратитесь в наш технический отдел. Примеры, приведенные в руководстве и других спецификациях, предназначены только для ознакомления и в качестве справки для пользователей и не гарантируют выполнения определенных действий. Перед началом использования данного изделия вместе с другими изделиями, пожалуйста, убедитесь, что оно соответствует требованиям спецификаций и установленным принципам работы. Перед использованием изделия убедитесь, что оно соответствует требованиям и безопасно. Пожалуйста, самостоятельно настройте функции резервного копирования и защиты, чтобы избежать возможного выхода из строя оборудования или ущерба в случае отказа данного изделия.

Исключение ответственности

Содержание руководства было тщательно проверено, однако потенциальные ошибки при работе не могут быть исключены. В связи с этим мы не гарантируем полного соответствия.

Мы регулярно проверяем руководство и исправляем возникающие проблемы в последующих версиях. Мы всегда рады обратной связи.

Авторское право PROMPOWER

Не копируйте руководство или информацию из него без письменного разрешения PROMPOWER. Нарушители понесут ответственность за убытки. Пожалуйста, сохраняйте все авторские права нашей компании, включая практические модули и разработанные патенты.

Содержание

1	Обзор сетевого обмена данными по Ethernet	5
1.1	Основная концепция Ethernet	5
1.1.1	Распределение IP-адресов	5
1.1.2	Сетевой адрес ПК	5
1.1.3	PING команда	7
1.2	TCP IP-протокол	8
1.2.1	Номер порта	8
1.2.2	Протокол UDP	9
1.2.3	Протокол TCP	9
2	Параметры Ethernet	12
2.1	Параметры сети Ethernet	12
2.1.1	Параметры IP-адреса	12
2.2	Настройте параметры Ethernet в PLC Studio	13
2.3	Настройка параметров Ethernet в Config Tool	14
3	Подключение и протокол связи	15
3.1	Режим подключения	15
3.2	Протокол TCP Modbus	15
3.2.1	Обзор TCP-соединения по шине Modbus	15
3.2.2	Адрес шины Modbus	15
3.2.3	Код функции Modbus	16
3.3	Протокол свободного сокета	16
4	Инструкции связи Ethernet	18
4.1	Обзор инструкций связи Ethernet	18

4.1.1	Создание TCP-соединения/прослушивание UDP-порта [S_OPEN]....	18
4.1.2	Окончание связи [S_CLOSE]	23
4.1.3	Отправка сообщения по протоколу свободного сокета [S_SEND]	24
4.1.4	Прием сообщений в свободном формате [S_RCV]	25
4.1.5	Modbus communication [M_TCP]	27
4.1.6	Пример связи по сети Ethernet	29
4.2	Инструкция чтения-записи для компорта	36
4.2.1	Пример связи с параметрами компорта.....	36
4.3	Флаг и регистр связи Ethernet	37
4.4	Список ошибок связи по сети Ethernet	38

1 Обзор сетевого обмена данными по Ethernet

1.1 Основная концепция Ethernet

Прежде чем приступить к обмену данными по Ethernet, давайте разберемся в некоторых ключевых понятиях Ethernet, таких как распределение IP-адресов, сетевой адрес ПК и т.п.

1.1.1 Распределение IP-адресов

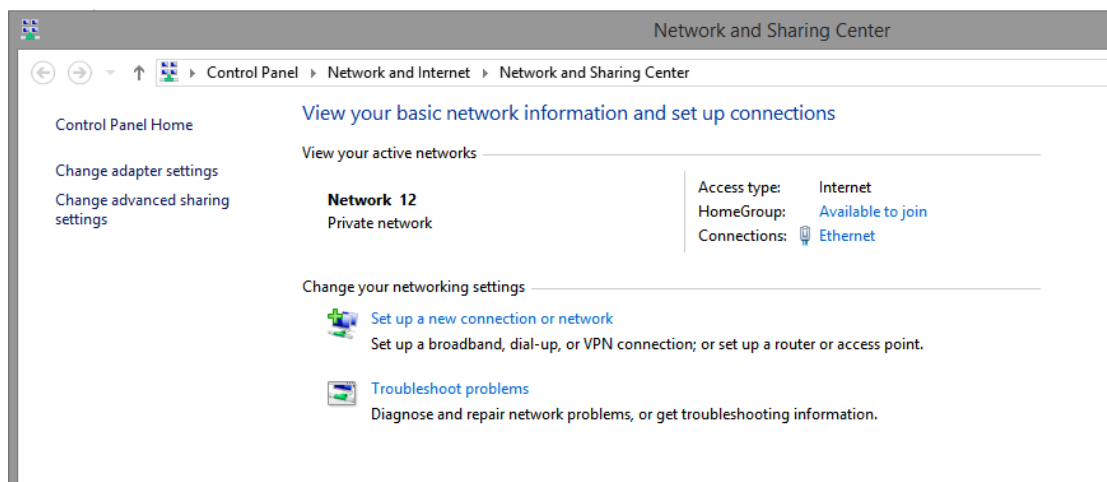
Если программируемые устройства (например, ПК) используют сетевую карту LAN для подключения к устройствам по Ethernet, ПК и настраиваемое устройство должны находиться в одной подсети. Комбинацией IP-адреса и маски подсети можно задать уникальную подсеть оборудования.

Идентификатор сети Network ID - это IP-адрес первой части, три верхние 8-битные группы (например, IP-адрес 211.154.184.16, 211.154.184 представляет собой идентификатор сети) определяют IP-сеть пользователя. Значение маски подсети обычно составляет 255.255.255.0. Однако, поскольку ваш компьютер находится в локальной сети (LAN). Если требуется установить уникальную подсеть, маска подсети (например, 255.255.254.0) может иметь другие значения. Маска подсети и IP-адрес оборудования выполняют логическую операцию AND для определения границ IP-подсети.

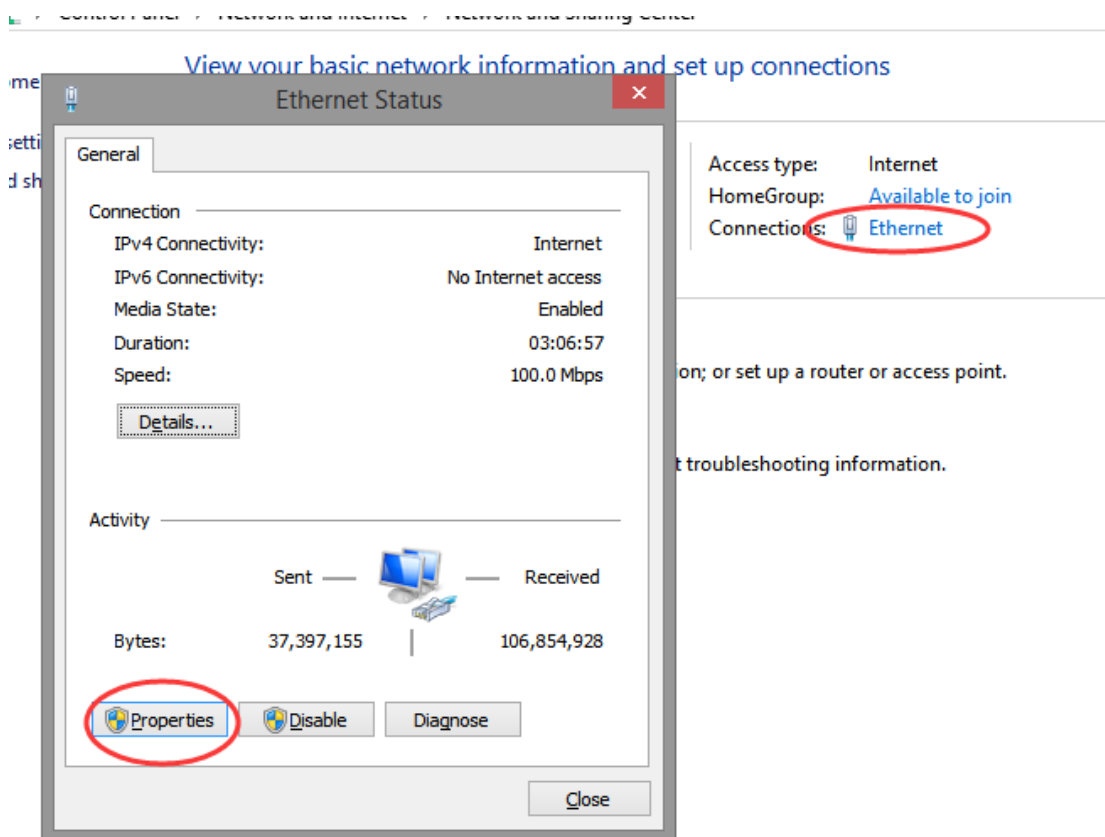
1.1.2 Сетевой адрес ПК

Проверьте IP-адрес устройства, выполнив следующие действия.

1. Откройте центр управления сетями и общим доступом:

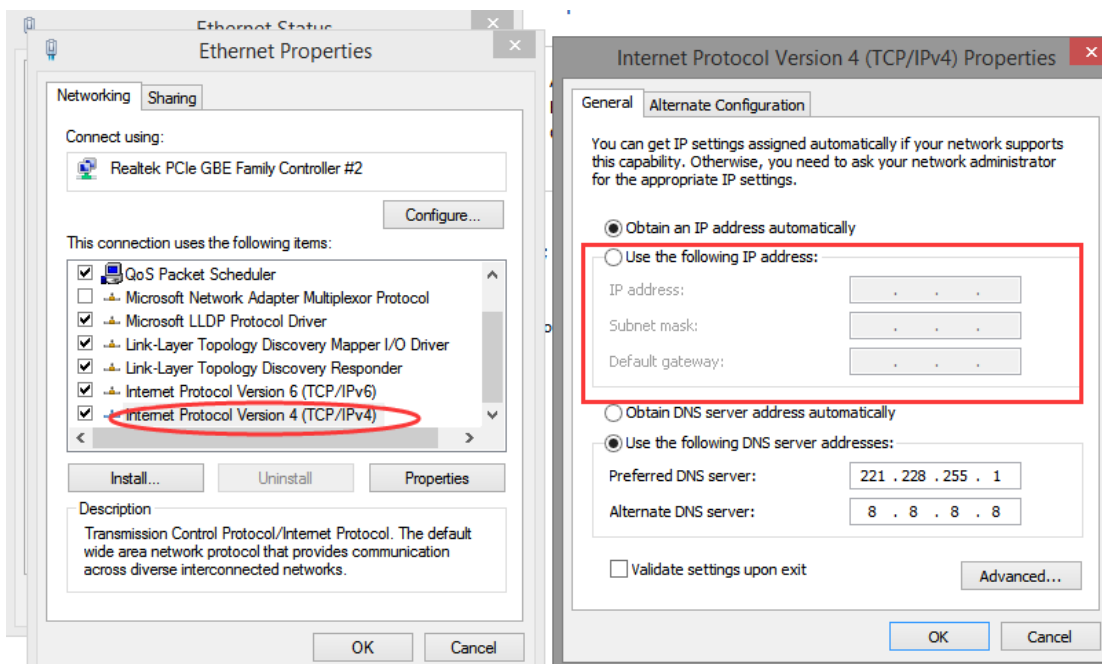


2. Нажмите на интересующее Ethernet-соединения, выберите свойства:



3. Установите IP-адрес компьютера, чтобы он находился в той же подсети.

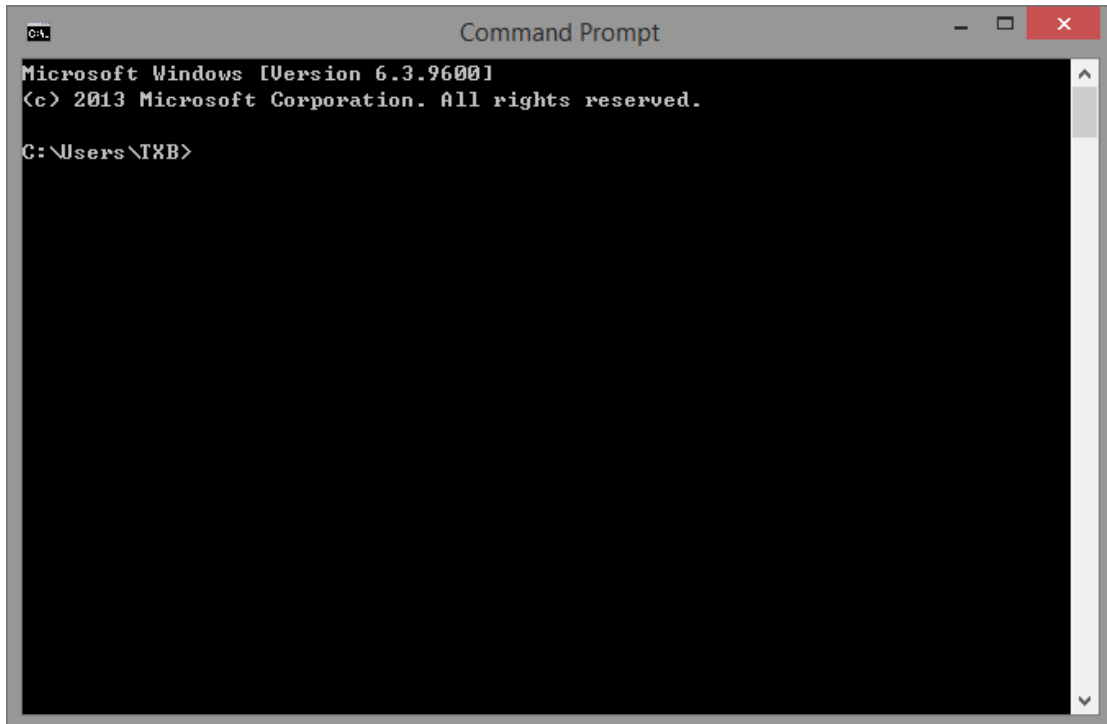
Например, IP ПЛК - 192.168.2.1 (по умолчанию IP 192.168.6.6), IP ПК - 192.168.2.200, маска подсети - 255.255.255.0. Шлюз по умолчанию можно не редактировать.



1.1.3 PING команда

С помощью команды PING можно проверить локальный протокол TCP/IP и возможность нормального соединения с другими устройствами локальной сети (LAN).

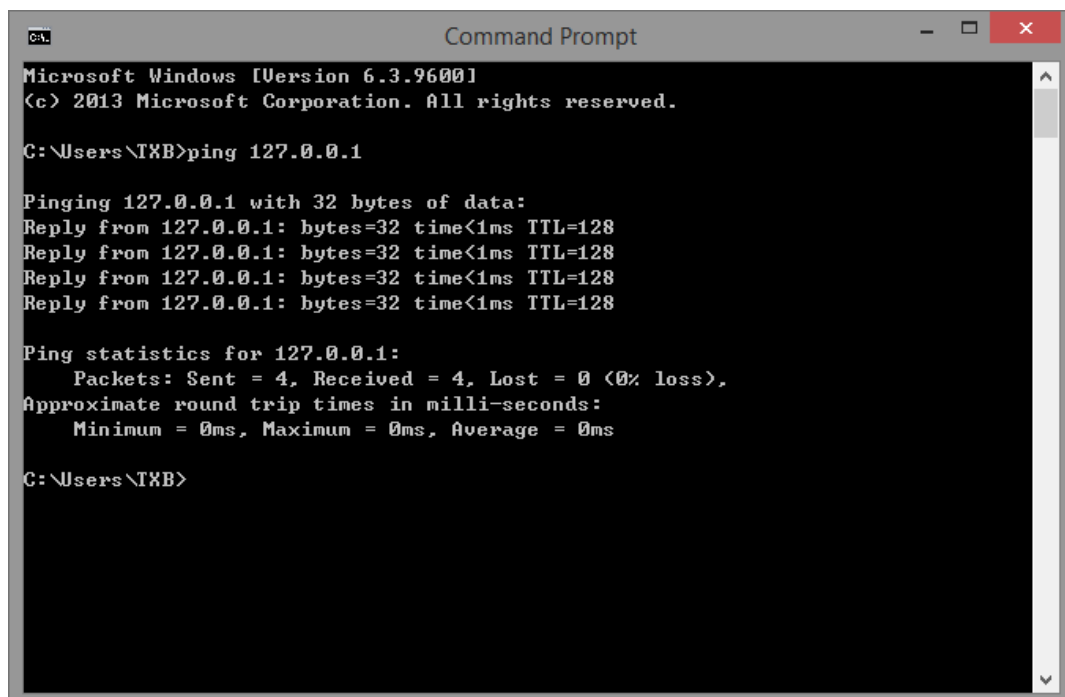
1. Откройте командную строку (Win + "R", введите "cmd")



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\TKB>
```

2. Введите "ping 127.0.0.1" для проверки локального протокола localhost TCP/IP. Часто данные приема и отправки совпадают.



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\TKB>ping 127.0.0.1

Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

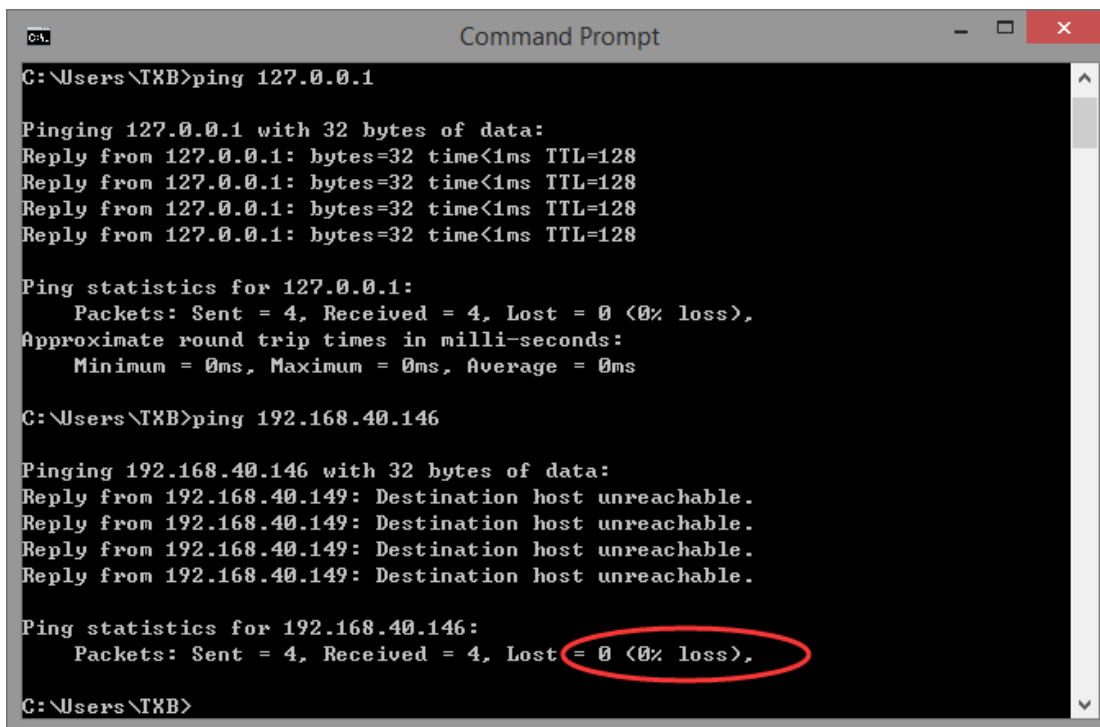
Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\TKB>
```

4. Введите команду 'ping network device ip', чтобы проверить, может ли ПК подключиться к другим устройствам в локальной сети.

(1) Введите команду "ping 192.168.40.146".

Если результат показывает "0% потерь" (0% loss), то данный ПК может подключиться к сетевому устройству с IP-адресом 192.168.40.146.



```
C:\Users\TKB>ping 127.0.0.1

Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\TKB>ping 192.168.40.146

Pinging 192.168.40.146 with 32 bytes of data:
Reply from 192.168.40.149: Destination host unreachable.
Reply from 192.168.40.149: Destination host unreachable.
Reply from 192.168.40.149: Destination host unreachable.
Reply from 192.168.40.149: Destination host unreachable.

Ping statistics for 192.168.40.146:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

```

(2) Введите команду "ping 192.168.40.127". Она показывает "100% loss", что означает невозможность подключения к устройству с IP-адресом 192.168.40.127.

Примечание: чаще всего 0% потерь означает, что связь нормальная.

1.2 TCP IP-протокол

Протокол TCP/IP является популярным протоколом связи Ethernet, который широко применяется в различных областях промышленной инженерии.

TCP/IP включает в себя протокол TCP, IP, UDP, ICMP и некоторые другие.

1.2.1 Номер порта

Для организации связи с приложениями верхнего уровня, в Ethernet связи на основе TCP или UDP должен быть использован номер порта. Диапазон портов от 0 до 65535, некоторые номера портов привязаны к уникальным функциям, например, порт 80 для обмена данными между системами типа «клиент — сеть», порт 21 для FTP-сервиса, порт 502 для связи Modbus TCP и так далее.

1.2.2 Протокол UDP

UDP — это транспортный протокол без подтверждения соединений (без механизма “handshake”), что означает, что он не тратит время на установку соединения перед отправкой данных. Вместо этого клиент может сразу отправить запрос на сервер. Этот запрос будет включать номер порта, соответствующий целевому приложению, а также данные и некоторую другую заголовочную информацию.

UDP обращается к различным функциям через различные номера портов. UDP не может обеспечить защиту при получении и отправке сообщений, но он обеспечивает проверку целостности заголовка и существенных данных (с помощью контрольных сумм для отслеживания целостности данных).

1.2.3 Протокол TCP

1. Основной принцип работы TCP

TCP - это протокол управления транспортным уровнем модели OSI, ориентированный на установку соединения. Ориентированность на установку соединения означает, что при TCP создается специальный виртуальный маршрут между TCP-клиентом и TCP-сервером. Для передачи данных по TCP необходимо установить соединение между концами хоста.

TCP обеспечивает надежную, упорядоченную и проверяющую ошибки функцию передачи сообщений для прикладной программы, запущенной на хост-машине, которая обменивается данными через Ethernet. TCP может гарантировать, что все принимаемые и отправляемые байты имеют одинаковое содержание и последовательность. Протокол TCP создает соединения между активными устройствами (т.е. устройством передачи) и пассивными устройствами (т.е. устройством приема). После установления соединения любая из сторон может начать передачу данных.

Протокол TCP представляет собой своеобразный “поток”, что означает, что у сообщения не существует флага окончания передачи, все принятые сообщения считаются частью потока данных. Например, клиентское устройство отправляет на сервер три фрагмента сообщения, каждый из которых составляет 20 байт. Сервер получит только 60-байтовый “поток” (при условии, что сервер выполняет операцию приема после получения трех фрагментов сообщения).

2. Основной принцип работы розетки

Сокет (Socket) - это основа коммуникации и базовая операционная единица в поддержке сетевого взаимодействия TCP/IP. Сокет является абстрактным представлением конечной точки в процессе сетевого взаимодействия. Он содержит пять различных видов информации для сетевого взаимодействия: протокол соединения, IP-адрес локального хоста, порт локального процесса,

IP-адрес удаленного хоста и порт удаленного процесса.

Когда прикладной уровень взаимодействует через транспортный уровень, TCP сталкивается с проблемой одновременного обслуживания нескольких процессов. Несколько TCP-соединений или несколько прикладных процессов могут проходить через один и тот же TCP-порт. Чтобы различать разные прикладные процессы и соединения, многие компьютерные операционные системы предоставляют интерфейс сокетов для взаимодействия приложений и протокола TCP/IP.

3. Установите соединение с сокетом

Для установления сокетного соединения необходима как минимум пара сокетов: один запускается на клиенте (также называемом TCP-клиентом), называемый ClientSocket, другой запускается на сервере (также называемом TCP-сервером), называемый ServerSocket.

Процесс сокетного соединения делится на три этапа: мониторинг сервером, запрос клиента, подтверждение соединения.

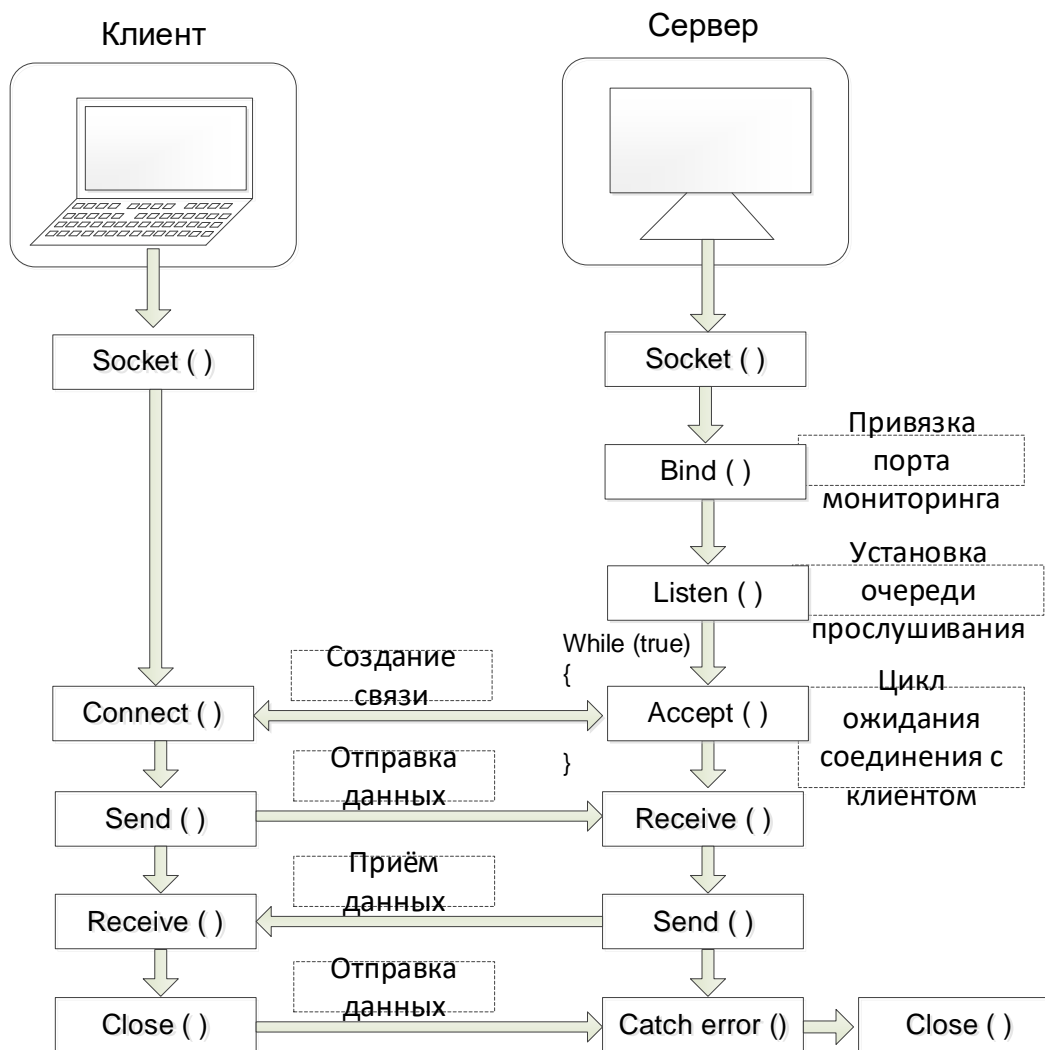
Мониторинг сервером: серверный сокет не определяет местонахождение конкретного клиентского сокета, а находится в состоянии ожидания соединения, и в режиме реального времени следит за состоянием сети, ожидая запроса на соединение от клиента.

Клиентские запросы: клиентский сокет отправляет запросы на соединение с сокетом сервера. По этой причине клиентский сокет должен сначала описать серверный сокет, указать адрес и номер порта серверного сокета, а затем направить запросы на соединение с серверным сокетом.

Подтверждение соединения: когда серверный сокет получает запрос на соединение с клиентским сокетом, он отвечает на запрос клиентского сокета, создает новый поток, отправляет описание серверного сокета клиенту, как только клиент подтверждает описание, обе стороны устанавливают соединение. Серверный сокет находится в состоянии прослушивания и продолжает принимать другие запросы на соединение с клиентским сокетом.

При создании сокетного соединения можно указать протокол транспортного уровня, сокет может поддерживать различные протоколы транспортного уровня (TCP или UDP), при использовании протокола TCP для подключения сокета, соединение является TCP-соединением.

Диаграмма взаимодействия TCP:



На приведенной выше схеме сокет сервера находится в состоянии прослушивания (мониторинга), клиент запрашивает соединение с сервером, сервер получает запрос на соединение и отправляет ответ для подтверждения информации клиенту, после того как клиент получил ответ, он отправляет подтверждающую информацию на сервер. После завершения выделения ресурсов TCP-соединение успешно устанавливается, этот процесс называется трехсторонним рукопожатием (three-way handshake).

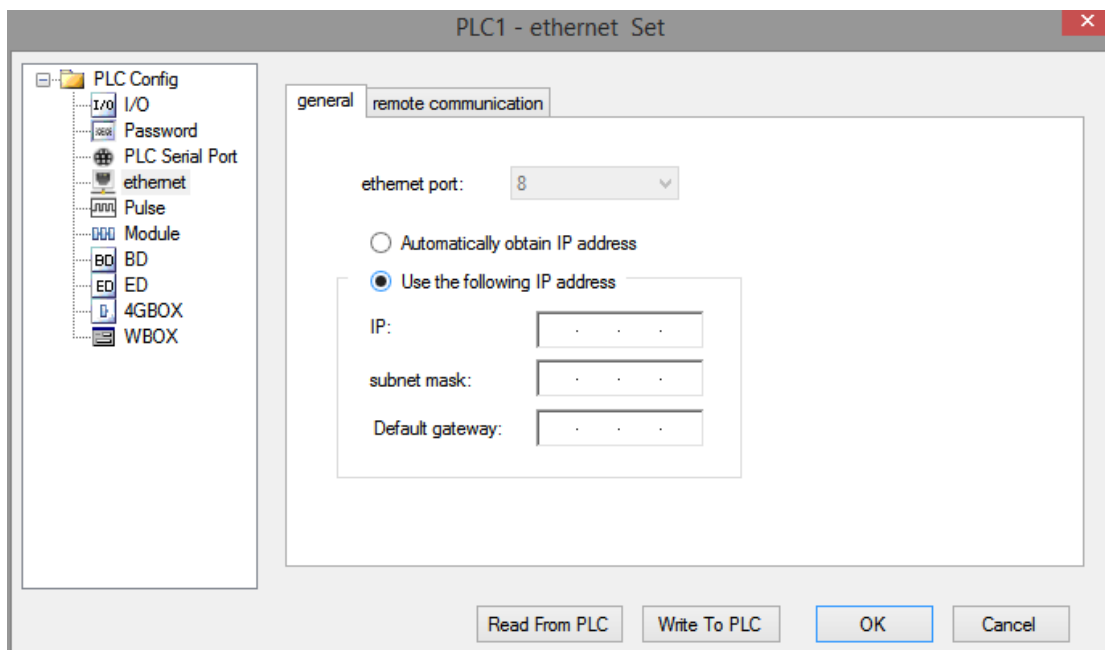
После формирования соединения клиент и сервер могут отправлять и получать данные, после завершения передачи данных клиент или сервер могут запросить закрытие соединения, после четвертого "рукопожатия" TCP-соединение закрывается, все передачи данных прерываются.

2 Параметры Ethernet

2.1 Параметры сети Ethernet

2.1.1 Параметры IP-адреса

Необходимо установить IP-адрес в сети Ethernet в качестве уникальной идентификации каждого устройства. Существует четыре параметра, следующие диаграммы представляют собой интерфейс настройки IP-адреса в программном обеспечении.



Получение IP-адреса

Поддержка автоматического получения IP-адреса, функция статической настройки, автоматическое получение начальных настроек ПЛК.

Режим автоматического получения: если в подсети есть DHCP-сервер, IP, маска подсети, шлюз по умолчанию назначаются DHCP-сервером.

Без DHCP-сервера сетевые параметры используют значения по умолчанию:

IP-адрес: 192.168.6.6

Маска подсети: 255.255.255.0

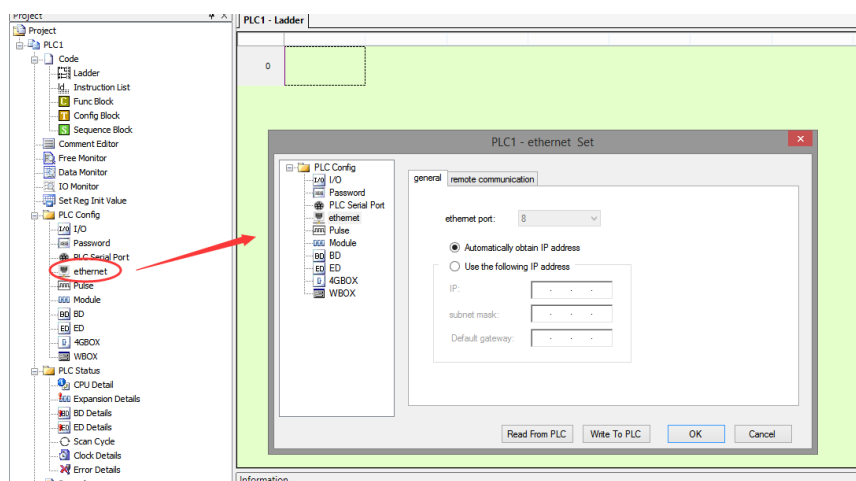
Шлюз по умолчанию: 192.168.6.1

Статический режим: пользователь назначает IP-адрес, маску подсети, информацию о шлюзе по умолчанию. Поддерживается только информация о частных IP-адресах.

Тип IP-адреса	Диапазон IP-адресов	Количество IP-устройств
Частный адрес класса А	10.0.0.0-10.255.255.255	16777216
Частный адрес класса В	172.16.0.0-172.31.255.255	1048576
Частный адрес класса С	192.168.0.0-192.168.255.255	65535

2.2 Настройте параметры Ethernet в PLC Studio

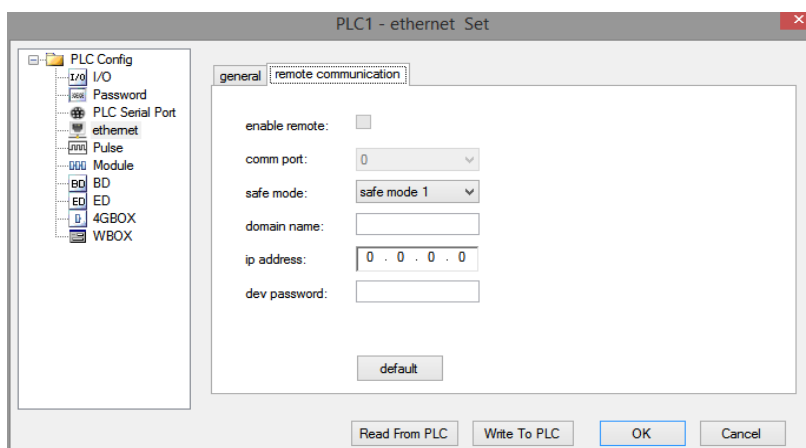
Откройте программное обеспечение для программирования ПЛК PROMPOWER PLC Studio, нажмите на Ethernet в дереве проекта, см. рисунок ниже.



Выбрав удаленную связь (remote communication) как на рисунке выше, вы сможете настроить удаленные параметры.

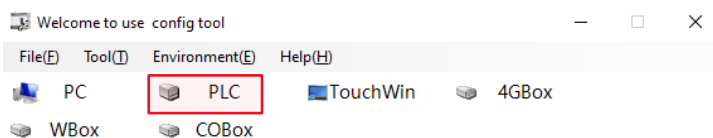
При организации локальной сети (LAN) необходимости устанавливать параметры нет.

Чтобы настройки вступили в силу, пожалуйста, после записи проекта на ПЛК (Download) перезапустите ПЛК по питанию.



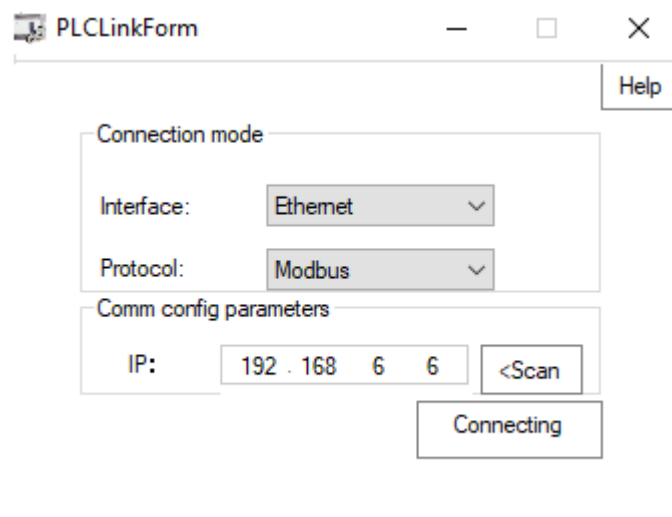
2.3 Настройка параметров Ethernet в Config Tool

Подключите ПЛК к ПК, откройте программу настройки Config Tool, нажмите на кнопку PLC.



On line

Введите IP адрес ПЛК или выберите его из списка Scan (нажмите на кнопку поиска Scan). После нажмите на кнопку Connecting.



Если сообщения об ошибке нет, это означает, что соединение установлено успешно.

3 Подключение и протокол связи

3.1 Режим подключения

Физический интерфейс модели Ethernet - RJ45. Кабель для подключения рекомендуется использовать UTP и STP. Длина одного кабеля не должна превышать 100 метров.

Тип коммутатора: рекомендуется использовать адаптивный Мбит/Гбит коммутатор.

3.2 Протокол TCP Modbus

3.2.1 Обзор TCP-соединения по шине Modbus

Modbus TCP объединил стандартный TCP/IP, физическую сеть Ethernet и Modbus в качестве метода представления данных протокола приложения данных. Коммуникационное сообщение Modbus TCP инкапсулируется в пакеты Ethernet TCP/IP, максимальная длина одного кадра протокола Modbus составляет 256 байт.

Modbus TCP/IP имеет два типа устройств: клиент и сервер.

Клиент Modbus:

Клиент (TCP Client) создаёт запрос на соединение с сервером (TCP Server).

Когда устройство Ethernet является клиентом Modbus TCP, оно устанавливает TCP-соединение с помощью инструкции S_OPEN, иницирует запрос Modbus с помощью инструкции M_TCP.

Сервер Modbus:

Сервер прослушивает порт 502, ожидая запрос на подключение клиента. После успешного установления соединения отвечает на запрос передачи данных в соответствии со спецификацией протокола Modbus TCP.

Ethernet-устройства по умолчанию запускают эту службу при включении питания.

3.2.2 Адрес шины Modbus

Для случая использования программируемого контроллера в качестве сервера Modbus, номер регистра и соответствующий ему номер адреса Modbus можно найти в руководстве по программированию ПЛК PROMPOWER "Руководство по программированию PMP20" и "Руководство по программированию PMP30".

3.2.3 Код функции Modbus

ПЛК модели Ethernet поддерживает следующие коды функций связи Modbus:

Код функции	Функция	Описания
01H	Read coil	Считывание адреса 0X, максимальное количество - 2000
02H	Read input coil	Считать 1X адрес, максимальное количество 2000
03H	Read holding register	Считывание адреса 4X, максимальное количество - 120
04H	Read input register	Считывание адреса 3X, максимальное количество - 120
05H	Write single coil	Запись одиночного адреса 0X
06H	Write single register	Запись одного адреса 4X
0FH	Write multiple coils	Запишите адрес 0X, максимальное количество - 2000
10H	Write multiple registers	Запись адреса 4X, максимальное количество - 120

3.3 Протокол свободного сокета

Протокол свободного сокета (Free format protocol) на базе Ethernet делится на две категории: TCP и UDP. Модель Ethernet с использованием TCP связи может быть использована в качестве TCP клиента (TCP client) или в качестве TCP сервера (TCP server).

1. TCP-клиент: ПЛК берёт на себя инициативу по установлению TCP-соединения с TCP-сервером, производит привязку используя идентификатор сокета (socket ID).
2. TCP-сервер: ПЛК ожидает запрос от TCP-клиента и устанавливает TCP-соединение, а также производит привязку используя идентификатор сокета (socket ID).
3. Используя UDP, прослушивает указанный локальный порт и привязывает идентификатор сокета.

На основе вышеуказанных трех пунктов, мы можем реализовать связь по протоколу свободного сокета. Обмен производится с использованием блоков данных, ограниченных кэшем ПЛК. Объём данных как при отправке, так и при приеме равняется 1000 байт.

Параметры связи в протоколе свободного сокета:

Режим буфера данных: 8 бит, 16 бит

1. 8-битная буферная связь: старший байт регистра недоступен, ПЛК использует только младший байт регистра для отправки и получения данных.
2. Обмен данными с 16-битным буфером: при приеме данных ПЛК сначала сохраняет младший байт, затем старший; при отправке данных ПЛК сначала отправляет младший байт, затем старший.
3. Если длина пакета принимаемых данных превышает установленную длину, данные будут сохранены в режиме 16-битного буфера.

4 Инструкции связи Ethernet

4.1 Обзор инструкций связи Ethernet

Инструкции связи Ethernet включают: открытие и закрытие запроса связи, отправку/получение данных. При использовании инструкции Ethernet, пожалуйста, выполните следующие действия:

1. Создайте запрос связи: подтвердите выбор протокола связи и тип связи. Настройте параметры связи, создайте TCP-соединение/UDP-порт для прослушивания и привяжите идентификатор сокета.
2. Для осуществления обмена данными: организуйте связь, добейтесь свободного обмена данными по Ethernet или Modbus TCP.
3. Закройте запрос связи: после завершения обмена сообщениями между ПЛК и целевым устройством, или если TCP-соединение с ошибками (abnormal), необходимо закрыть задачу связи.

4.1.1 Создание TCP-соединения/прослушивание UDP-порта [S_OPEN]

1. Обзор

Для инициации коммуникации используйте S_OPEN.

Создайте TCP-соединение /UDP-порт для прослушивания [S_OPEN]			
16-битная инструкция	S_OPEN	32-битная инструкция	-
Условие выполнения	Срабатывание по фронту	Подходящая модель	PMP20
Прошивка	V3.5.3 и выше	Программное обеспечение	V3.5.3 и выше

2. Операнд

Операнд	Функция	Тип
S1	Идентификатор сокета	16 бит, BIN
S2	Тип связи	16 бит, BIN
S3	Режим связи с локальным устройством	16 бит, BIN
S4	Адрес начала блока параметров	16 бит, BIN

S5	Начальное положение флага	Бит
----	---------------------------	-----

3. Подходящие регистры

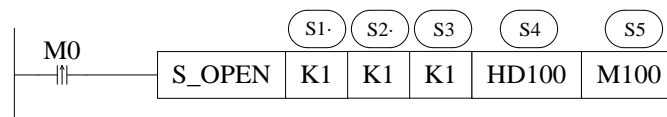
Слово	Операнд	Регистр								Постоянная	Модуль		
		D	FD	ED	TD	CD	DX	DY	DM	DS	К/Н	ID	QD
	S1	•									•		
	S2	•									•		
	S3	•									•		
S4	•												
Бит	Операнд	Регистр											
		X	Y	M	S	T	C	Dn.m					
S5			•										

Примечание:

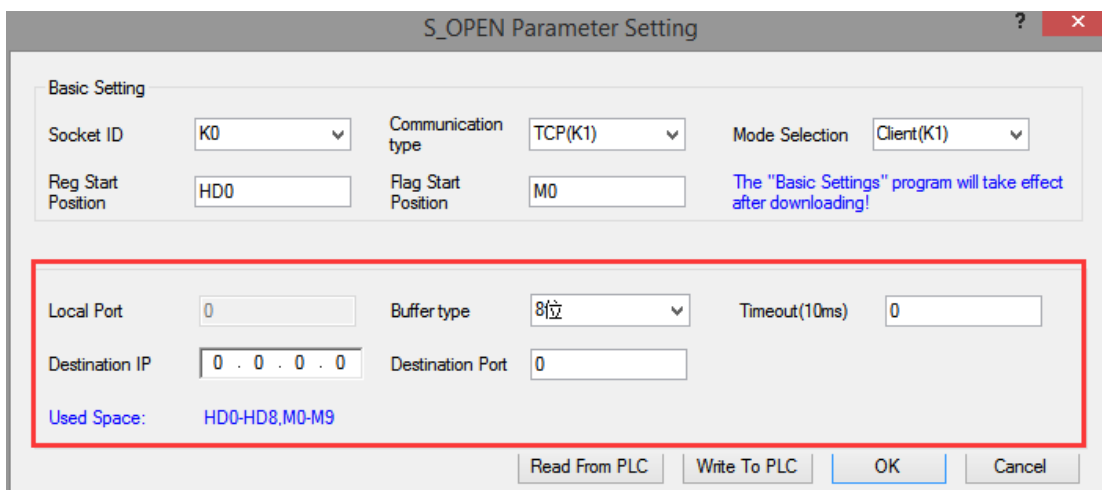
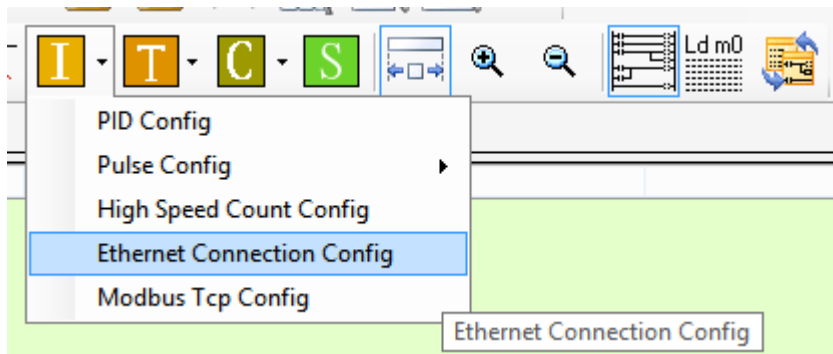
D означает D HD; TD означает TD HTD; CD означает CD HCD HSCD HSD; DM означает DM DHM;

DS означает DS DHS. M означает M HM SM; S означает S HS; T означает T HT; C означает C HC.

Функция



- Создайте коммуникационную задачу, по переднему фронту M0 инструкция создаст одно TCP-соединение или откроет UDP-порт.
- S1: идентификатор сокета, диапазон: K0~K63. Примечание: количество сокетов не может быть больше 64, количество TCP соединений не может быть больше 32, количество UDP соединений не может быть больше 32.
- S2: тип связи, диапазон: K0, K1. K0 - UDP, K1 - TCP.
- S3: режим связи. Диапазон: K0, K1. K0 - сервер, K1 - клиент.
- S4: адрес начала блока параметров, занимает 9 регистров от S4 до S4+8.
- S5: стартовая позиция флага, занимает 10 катушек от S5 до S5+9.
- Эту инструкцию можно задать через следующее окно



Примечание: параметры в красной рамке вступят в силу после повторного включения питания ПЛК.

- Флаг ошибки Ethernet SM1921 появляется, когда связь нарушена, информация об ошибке будет сохранена в SD1920 и SD1921, см. главу 4-3.

Для примера возьмем изображение выше, адрес, начинающийся с HD0, и адрес флага, начинающийся с M0, показаны ниже:

Local Port	HD0	Connection start mark	M0
Target IP Sec 1 (e.g. :192)	HD1 High Byte	Linked mark	M1
Target IP Sec 2 (e.g. :168)	HD1 Low Byte	Sending mark	M2
Target IP Sec 3 (e.g. :0)	HD2 High Byte	Passed mark	M3
Target IP Sec 4 (e.g. :1)	HD2 Low Byte	Receiving mark	M4
Destination Port	HD3	Received mark	M5
The data buffering	HD4	Closing mark	M6
Receiving Timeout	HD5	Modbus TCP communication mark	M7
The reserved	HD6	TCP exception mark	M8
Actual number of bytes received(Byte)	HD7	Error mark	M9
Error Type	HD8		

Объяснение параметров:

Коммуникационная задача, создаваемая S_OPEN, делится на три категории: TCP-клиент, TCP-сервер, UDP. Параметры, используемые в вышеперечисленных категориях, отличаются.

В таблице ниже приведена подробная информация о категориях S_OPEN.

Тип связи	Локальный порт	IP-адрес устройства	Порт устройства	Тип буфера	Тайм-аут	Полученные байты	Код ошибки
TCP-клиент	-	√	√	√	√	√	√
TCP-сервер	√	-	-	√	√	√	√
UDP	√	√	√	√	√	√	√

1. Локальный порт (Local port)

Диапазон от 1 до 60000, порт 502 и 531 - специальные порты, которые не могут быть использованы. Локальный порт может использоваться только одной коммуникационной инструкцией.

2. IP-адрес устройства (Destination IP)

IP-адрес целевого устройства, который находится в той же подсети (тот же Network ID), что и локальное устройство.

3. Порт устройства (Destination port)

Номер сетевого порта целевого устройства. Диапазон от 1 до 65535. Порт должен быть равен 502 для связи по Modbus TCP.

4. Тип буфера (Buffer type)

Когда значение равно 0, это 8-битный режим. Если значение ненулевое, это 16-битный режим. Если фактический принимаемый пакет больше установленной длины, он автоматически перейдет в 16-битный режим.

5. Тайм-аут (Timeout)

Время от запроса ПЛК на получение данных до окончания приема данных. Диапазон от 0 до 65536. Единица измерения - 10 мс. 0 означает, что тайм-аут отключен и будет продолжаться прием данных. Ненулевое значение означает, что функция тайм-аута включена. Тайм-аут приема действует для S_RCV и M_TCP.

Если тайм-аут установлен на 300 мс, то система будет выжидать начала обработки запроса 300 мс, а после успешного получения данных сразу же завершит обмен. Если в течение 300 мс данные не были получены, данная инструкция завершится и сообщит об ошибке таймаута приема.

6. TCP keep alive

(1) значение равно 0, функция TCP keep alive не включена.

(2) значение ненулевое, функция TCP keep alive включена.

При активной функции поддержания жизни TCP keep alive, ПЛК отправляет сообщение поддержания жизни объекту. Если отправитель не получит ответное сообщение, то другой хост будет подтвержден как недоступный. Время срабатывания составляет 1 ~ 5 минут, при нарушении устанавливается флаг TCP abnormal.

7. Длина приема данных

Команда S_RCV показывает фактическую длину принятых данных, единица измерения - байт.

8. Код ошибки

Сообщение об ошибке при передаче данных по Ethernet в режиме свободного сокета и Modbus TCP см. в главе 4-4.

9. Бит флага

Биты флага связи показаны ниже (в качестве примера возьмем адрес головы Mn).

Битовый адрес	Бит флага	Функция
Mn	Подключение	Создание связи, M(n) включается
M(n+1)	Подключено	Создание соединения завершено, M(n+1) включен
M(n+2)	Отправка	Идет отправка данных, M(n+2) включён
M(n+3)	Отправлено	Отправка данных завершена, M(n+3) включён
M(n+4)	Получение	Данные принимаются, M(n+4) включен.
M(n+5)	Получено	Прием данных завершен, M(n+5) включён
M(n+6)	Закрытие	Текущее соединение закрывается, M(n+6) включён

M(n+7)	Общение по протоколу Modbus TCP	При выполнении инструкции M_TCP, M(n+7) включен
M(n+8)	Нештатная ситуация с TCP	TCP-соединение нарушено, M(n+8) включен
M(n+9)	Флаг ошибки	Ошибка связи, M M(n+9) включён

4.1.2 Окончание связи [S_CLOSE]

1. Обзор

Пожалуйста, используйте инструкции завершения связи S_CLOSE вместе с инструкцией открытия S_OPEN.

Прекращение связи [S_CLOSE]			
16-бит	S_CLOSE	32-бит	-
Условие выполнения	Срабатывание по фронту	Подходящая модель	PMP20
Прошивка	V3.5.3 и выше	Программное обеспечение	V3.5.3 и выше

2. Операнд

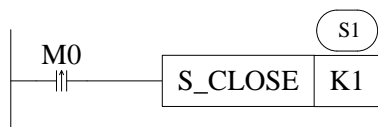
Операнд	Функция	Тип
S1	Идентификатор закрытого сокета	16 бит, BIN

3. Подходящие регистры

Слово	Операнд	Регистр								Постоянная	Модуль		
		D	FD	ED	TD	CD	DX	DY	DM	DS	K/H	ID	QD
	S1	•									•		

Примечание: D означает D HD ; TD означает TD HTD ; CD означает CD HCD HSCD HSD; DM означает DM DHM; DS означает DS DHS.

Функция



- Завершите задачу связи, когда наступит нарастающий фронт M0.
- Примечание: эта инструкция должна использоваться вместе с S_OPEN.
- S1: идентификатор сокета, который необходимо закрыть, операнд может быть регистром или константой, диапазон K0~K63.
- После выполнения S_CLOSE, инструкции M_TCP, S_SEND, S_RCV, основанные на том же идентификаторе сокета, **выполняться больше не могут**.

4.1.3 Отправка сообщения по протоколу свободного сокета [S_SEND]

1. Обзор

Инструкция отправки сообщения в свободном режиме (по протоколу свободного сокета) должна использоваться вместе с S_OPEN и S_CLOSE.

Отправка сообщения в свободном формате [S_SEND]			
16-бит	S_SEND	32-бит	-
Условие выполнения	Срабатывание по фронту	Подходящая модель	PMP20
Прошивка	V3.5.3 и выше	Программное обеспечение	V3.5.3 и выше

2. Операнд

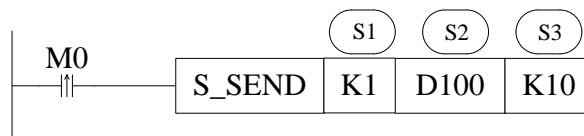
Операнд	Функция	Тип
S1	Идентификатор сокета	16 бит, BIN
S2	Адрес локального регистра передачи данных	16 бит, BIN
S3	Отправка количества данных	16 бит, BIN

3. Подходящие регистры

Слово	Операнд	Регистр								Постоянная	Модуль		
		D	FD	ED	TD	CD	DX	DY	DM	DS	К/Н	ID	QD
	S1	•									•		
	S2	•											
	S3	•									•		

Примечание: D означает D HD ; TD означает TD HTD ; CD означает CD HCD HSCD HSD; DM означает DM DHM; DS означает DS DHS.

Функция



- Команда отправки сообщения в свободном формате (по протоколу свободного сокета) отправляет данные по переднему фронту сигнала M0.
Примечание: эта инструкция должна использоваться вместе с S_OPEN и S_CLOSE.
- S1: идентификатор сокета, операнд может быть регистром или константой, диапазон K0~K63
- S2: адрес локального регистра отправки данных
- S3: количество данных. Операндом может быть регистр или константа
- Обратите внимание на тип буфера данных S_OPEN в идентификаторе сокета.

4.1.4 Прием сообщений в свободном формате [S_RCV]

1. Обзор

Инструкция получения сообщения в свободном формате (по протоколу свободного сокета) должна использоваться вместе с S_OPEN и S_CLOSE.

Прием сообщений в свободном формате [S_RCV]			
16-бит	S_RCV	32-бит	-
Условие выполнения	Нормально вкл/выкл, срабатывание по фронту	Подходящая модель	PMP20
Прошивка	V3.5.3 и выше	Программное обеспечение	V3.5.3 и выше

2. Операнд

Операнд	Функция	Тип
S1	Идентификатор сокета	16 бит, BIN
S2	Адрес локального регистра приёма данных	16 бит, BIN

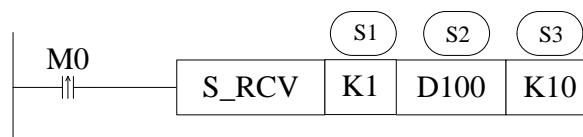
S3	Количество получаемых данных	16 бит, BIN
----	------------------------------	-------------

3. Подходящие регистры

Слово	Операнд	Регистр								Постоянная	Модуль		
		D	FD	ED	TD	CD	DX	DY	DM	DS	К/Н	ID	QD
	S1	•									•		
	S2	•											
	S3	•									•		

Примечание: D означает D HD ; TD означает TD HTD ; CD означает CD HCD HSCD HSD; DM означает DM DHM; DS означает DS DHS.

Функция



- Команда приема данных в свободном формате (по протоколу свободного сокета) будет принимать данные по переднему фронту сигнала M0.

Примечание: эта инструкция должна использоваться вместе с S_OPEN и S_CLOSE.

- S1: идентификатор сокета, операнд может быть регистром или константой, диапазон K0~K63
- S2: адрес локального регистра приёма данных
- S3: количество данных. Операндом может быть регистр или константа
- Обратите внимание на тип буфера данных S_OPEN в идентификаторе сокета.

4.1.5 Modbus communication [M_TCP]

1. Обзор

Когда ПЛК является клиентом, принимает и отправляет данные по протоколу Modbus TCP. Может использоваться вместе с S_OPEN и S_CLOSE.

Связь по протоколу Modbus TCP [M_TCP]			
16-бит	M_TCP	32-бит	-
Условие выполнения	Срабатывание по фронту	Подходящая модель	PMP20
Прошивка	V3.5.3 и выше	Программное обеспечение	V3.5.3 и выше

2. Операнд

Операнд	Функция	Модель
S1	Удаленная станция №.	16 бит, BIN
S2	Код функции связи Modbus	16 бит, BIN
S3	Целевой адрес	16 бит, BIN
S4	Количество регистров или катушек	16 бит, BIN
S5	Локальный адрес	16 бит, BIN
S6	Идентификатор сокета	16 бит, BIN

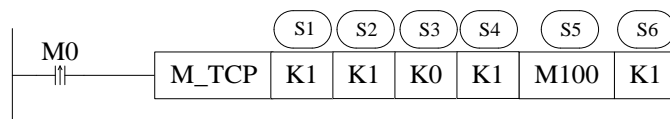
3. Подходящие регистры

	Операнд	Регистр								Постоянная	Модуль		
		D	FD	ED	TD	CD	DX	DY	DM	DS	К/Н	ID	QD
Слово	S1	•									•		
	S2	•											
	S3	•									•		
	S4	•									•		
	S5	•									•		
	S5	•									•		

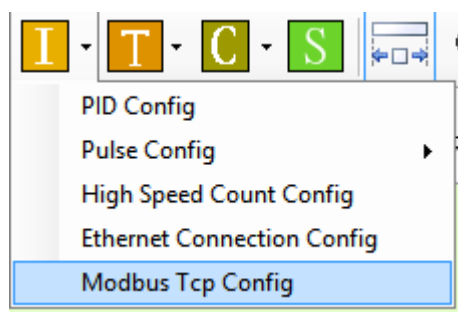
*Примечание: D означает D HD ; TD означает TD HTD ; CD означает CD HCD

HSCD HSD; DM означает DM DHM; DS означает DS DHS.

Функция



- Команда связи Modbus TCP, она будет осуществлять связь Modbus TCP один раз при наступлении нарастающего фронта импульса M0.
- S1: номер станции удаленной связи, диапазон K0~K247
- S2: Код функции связи Modbus
- S3: адрес целевой головки, это адрес связи Modbus.
- S4: количество данных связи
- S5: адрес локальной головки
- S6: идентификатор сокета, укажите TCP-соединение, целевой порт должен быть 502.
- Эта инструкция должна использоваться вместе с S_OPEN и S_CLOSE.
- M_TCP действует только тогда, когда ПЛК является клиентом и принимает и отправляет данные протокола Modbus TCP.
- Эта инструкция должна быть установлена через следующее окно



Socket ID	K1	S6	Local Strat Address	M100	S5
Station No.	K1	S1	Function Code	0x01 Read the coil	S3
Data Address	K0	S2	Count	K1	S4

Код функции:

Значение	Код функции	Значение	Код функции
K1	Прочитать катушку	K3	Считать регистр
K2	Считывание дискретной величины входного сигнала	K4	Чтение входного регистра
K5	Пишите одинарную катушку	K6	Запись одного регистра
K15	Запись нескольких катушек	K16	Запись нескольких регистров

4.1.6 Пример связи по сети Ethernet

Пример 1:

После включения питания ПЛК автоматически создает TCP-клиент, TCP-сервер и UDP-задание (инструкцию) связи, а также отправляет и получает данные на основе каждого задания (инструкции) связи. IP-адрес ПЛК - 192.168.0.60, IP-адрес целевого устройства В - 192.168.0.100.

Работа программы:

(1) после включения питания ПЛК (является TCP-клиентом), создается TCP-соединение с портом 1000 TCP-сервера устройства В и привязывает сокет с идентификатором ID1. После создания соединения он отправляет младшие 8-битные данные от D1000 до D1549 на устройство В каждую 1 секунду, получает данные и сохраняет их в младших 8-битных данных от D1600 до D1999.

Когда TCP-соединение нарушается, TCP-соединение закрывается и открывается снова.

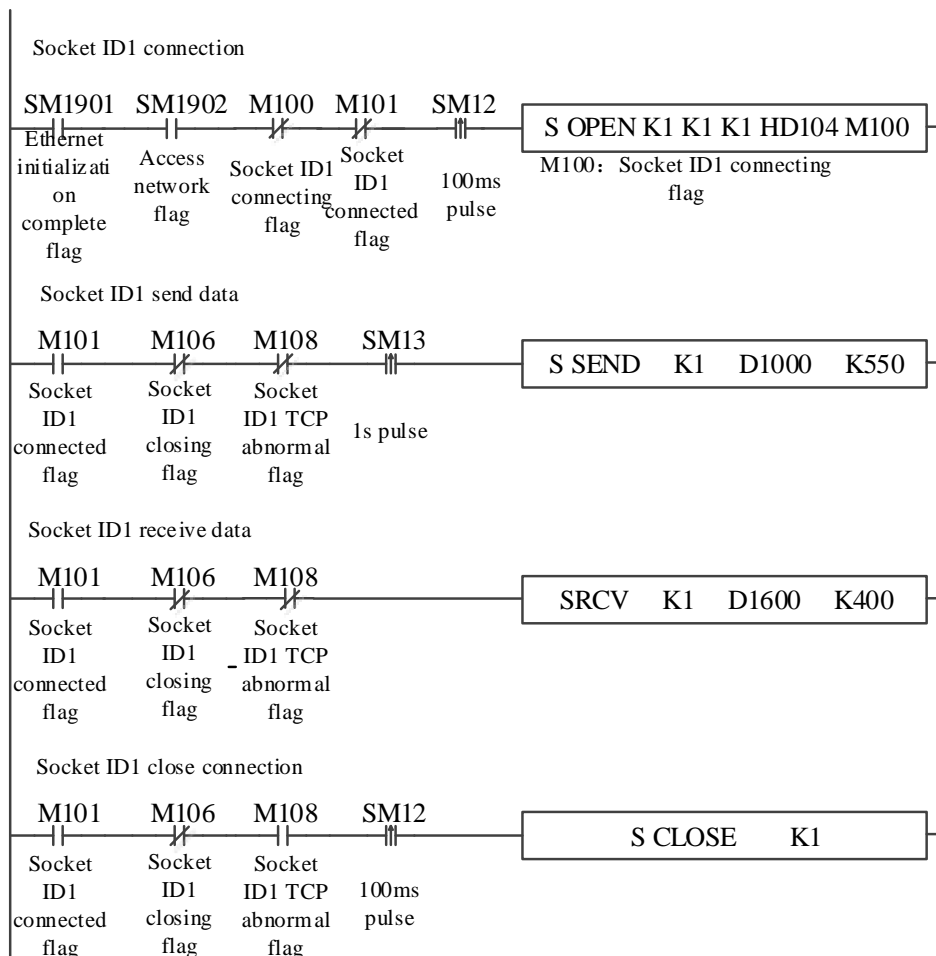
(2) после включения питания ПЛК (является TCP-сервером) прослушивает порт 1001, ожидает TCP-клиента, создающего TCP-соединение, и привязывает к обмену сокет ID2. После создания соединения ПЛК отправляет младшие 8-битные данные от D2000 до D2549 на подключенное устройство каждую 1 секунду, получает данные и сохраняет их в младших 8-битных данных от D2600 до D2999.

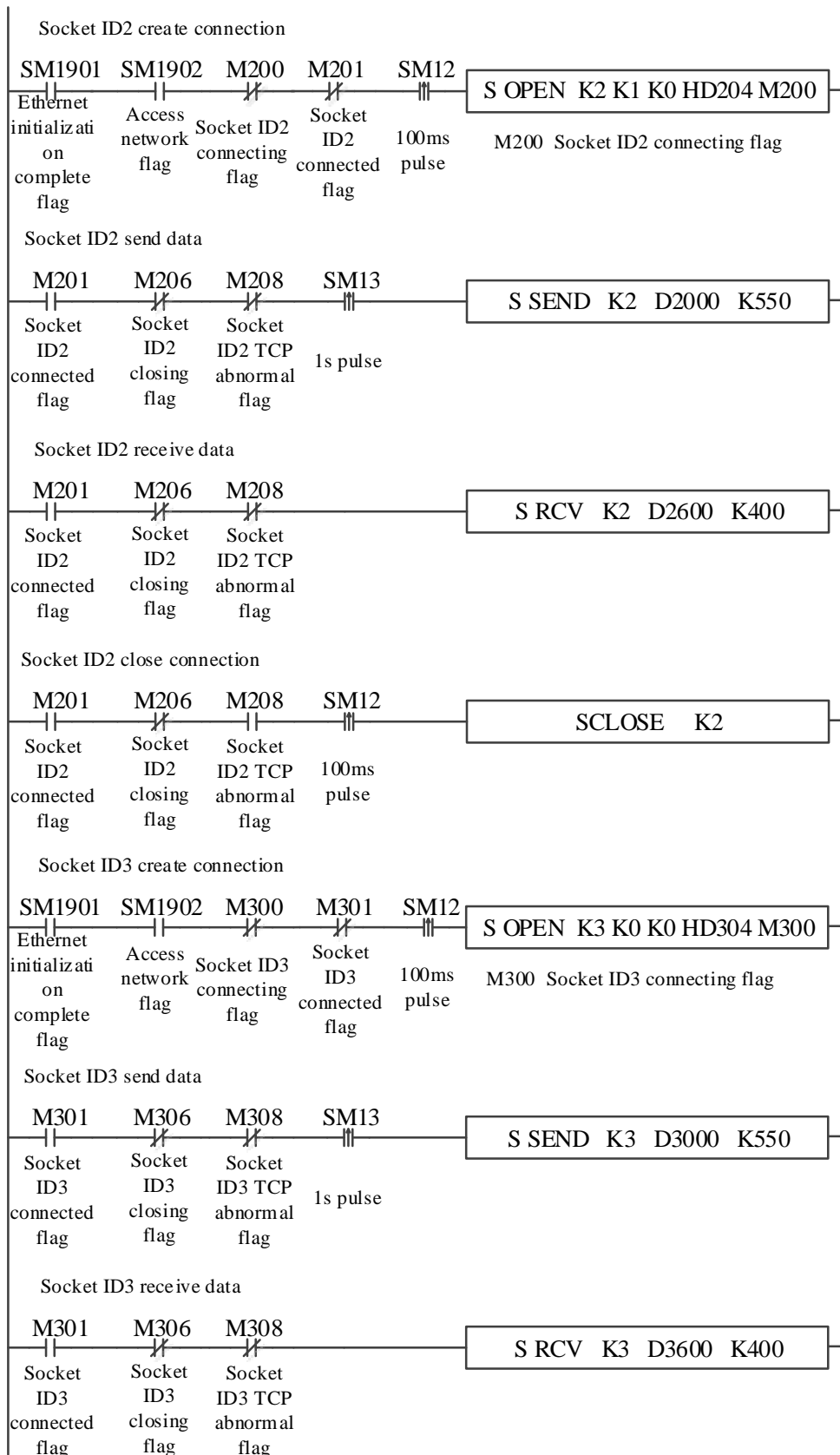
Когда TCP-соединение нарушается, TCP-соединение закрывается и открывается снова.

(3) после включения питания ПЛК (связь в режиме UDP), порт локального устройства - 1002, IP-адрес цели - 192.168.0.100, порт цели - 3000, привязывает сокет ID3. После создания соединения он отправляет младшие 8-битные данные от D3000 до D3549 на подключенное устройство каждые 1

секунду, получает данные и сохраняет их в младших 8-битных данных от D3600 до D3999.

Программа для ПЛК:





Настройка S_OPEN сокетa ID1:

S_OPEN Parameter Setting

Basic Setting

Socket ID: K1 Communication type: TCP(K1) Mode Selection: Client(K1)

Reg Start Position: HD104 Flag Start Position: M100

Local Port: 0 Buffer type: 8位 Timeout(10ms): 0

Destination IP: 192.168.0.100 Destination Port: 1000

Used Space: HD104-HD112,M100-M109

Read From PLC Write To PLC OK Cancel

The "Basic Settings" program will take effect after downloading!

Настройка S_OPEN сокетa ID2:

S_OPEN Parameter Setting

Basic Setting

Socket ID: K2 Communication type: TCP(K1) Mode Selection: Server(K0)

Reg Start Position: HD204 Flag Start Position: M200

Local Port: 1001 Buffer type: 8位 Timeout(10ms): 0

Destination IP: 192.168.0.100 Destination Port: 1000

Used Space: HD204-HD212,M200-M209

Read From PLC Write To PLC OK Cancel

The "Basic Settings" program will take effect after downloading!

Настройка S_OPEN сокетa ID3:

S_OPEN Parameter Setting

Basic Setting

Socket ID: K3 Communication type: UDP(K0) Mode Selection: Server(K0)

Reg Start Position: HD304 Flag Start Position: M300

Local Port: 1002 Buffer type: 8位 Timeout(10ms): 0

Destination IP: 192.168.0.100 Destination Port: 3000

Used Space: HD304-HD312,M300-M309

Read From PLC Write To PLC OK Cancel

The "Basic Settings" program will take effect after downloading!

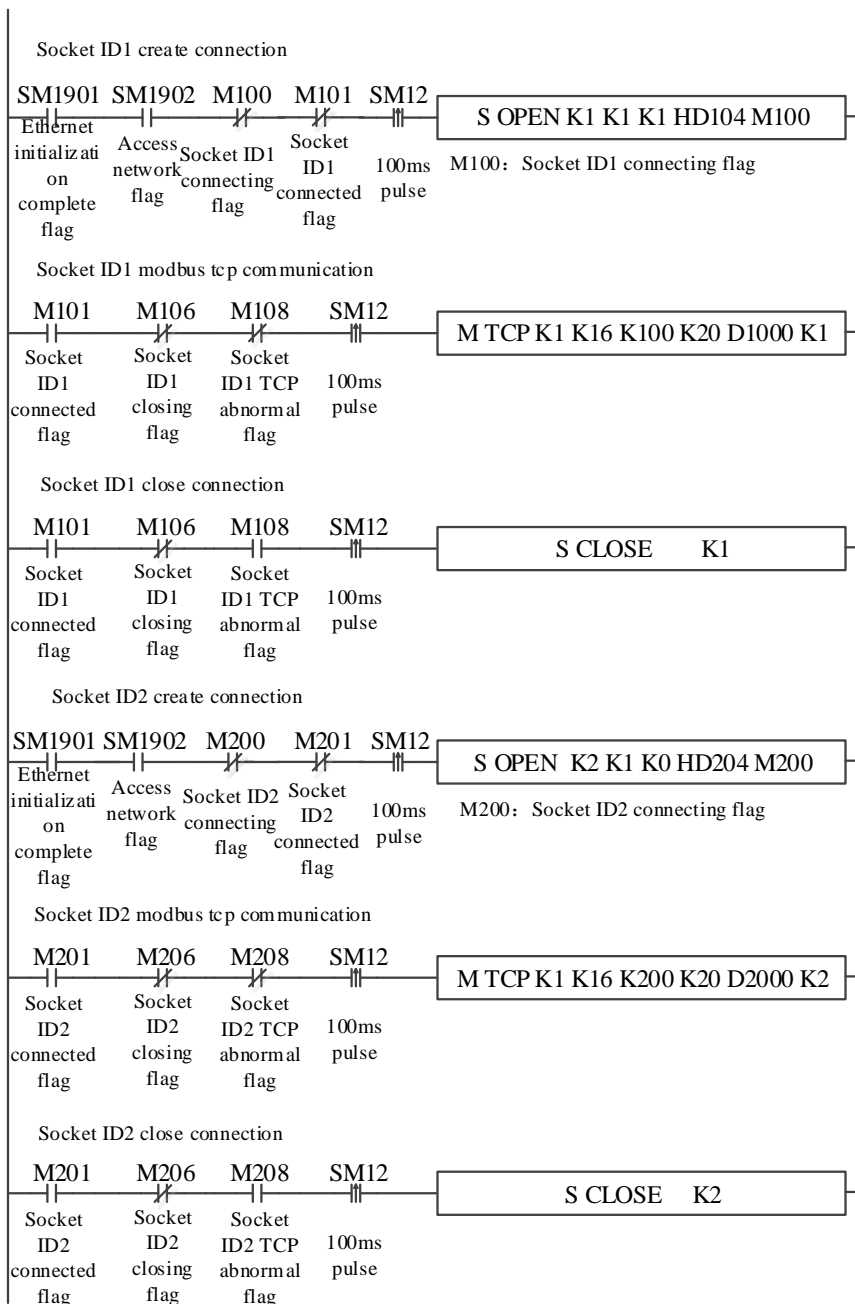
Пример 2:

После включения питания ПЛК установит связь с Modbus TCP-сервером устройством А и Modbus TCP-сервером устройством В, IP ПЛК - 192.168.0.60, IP устройства А - 192.168.0.40, номер станции Modbus - 1, IP устройства В - 192.168.0.70, номер станции Modbus - 1.

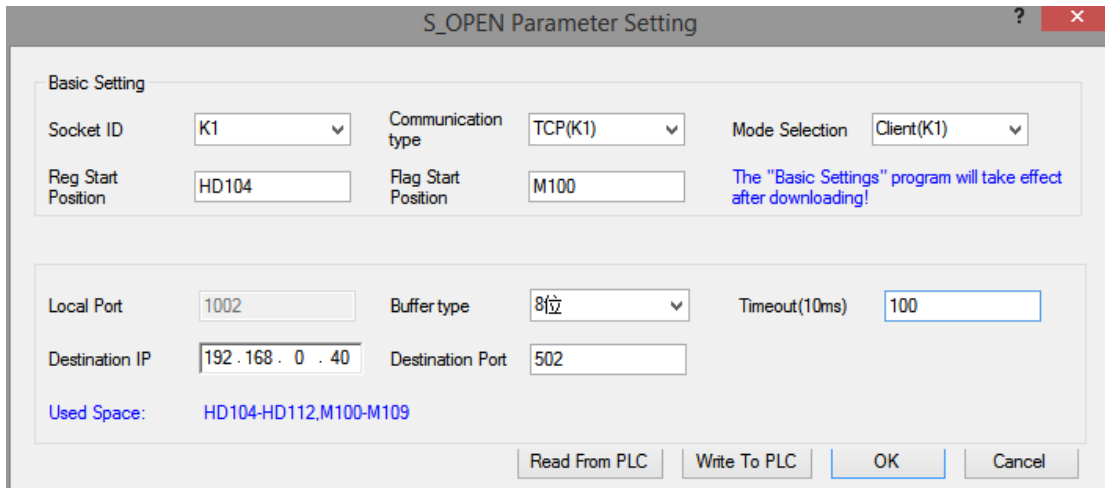
Работа программы:

- (1) После включения питания ПЛК (в качестве TCP-клиента) создаст TCP-соединение с портом 502 TCP-сервера устройства А, и привяжет сокет ID1.
После успешного создания соединения, ПЛК будет записывать значения из D1000 - D1019 в адрес устройства А 4x100 - 4x119 каждые 100 мс. Когда TCP-соединение будет нарушено, ПЛК закроет TCP-соединение и создаст его снова.
- (2) После включения питания ПЛК (в качестве TCP-клиента) создаст TCP-соединение с портом 502 TCP-сервера устройства В, и привяжет сокет ID2.
После успешного создания соединения, ПЛК будет записывать значения из D2000 - D2019 в адрес устройства В 4x200 - 4x219 каждые 100 мс. Когда TCP-соединение будет нарушено, ПЛК закроет TCP-соединение и создаст его снова.

Программа:

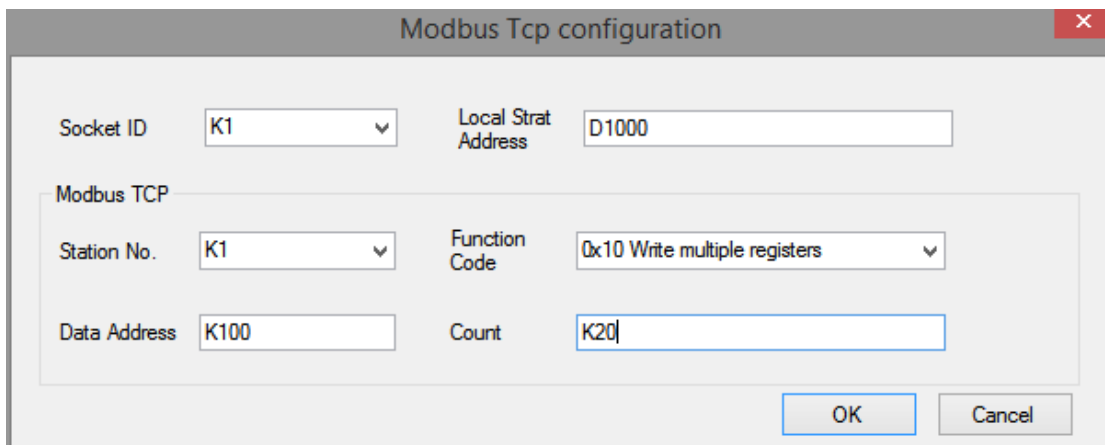


Настройка S_OPEN сокетa ID1:



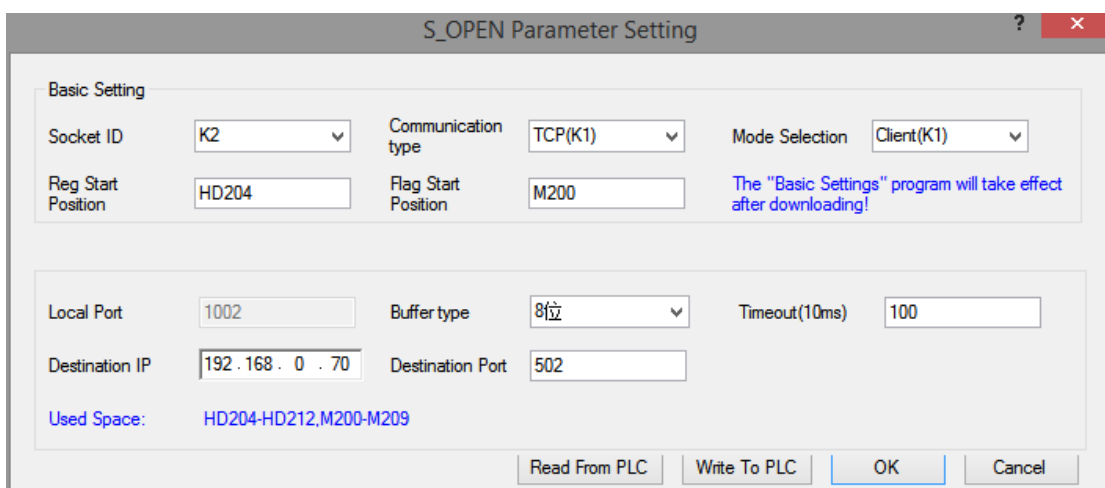
The screenshot shows the 'S_OPEN Parameter Setting' dialog box. It is divided into two main sections. The top section, 'Basic Setting', contains: 'Socket ID' set to 'K1', 'Communication type' set to 'TCP(K1)', 'Mode Selection' set to 'Client(K1)', 'Reg Start Position' set to 'HD104', and 'Flag Start Position' set to 'M100'. A blue note states: 'The "Basic Settings" program will take effect after downloading!'. The bottom section contains: 'Local Port' set to '1002', 'Buffer type' set to '8位', 'Timeout(10ms)' set to '100', 'Destination IP' set to '192.168.0.40', and 'Destination Port' set to '502'. A 'Used Space' field shows 'HD104-HD112,M100-M109'. At the bottom are buttons for 'Read From PLC', 'Write To PLC', 'OK', and 'Cancel'.

Настройка сокетa ID1 M_TCP:



The screenshot shows the 'Modbus Tcp configuration' dialog box. It contains: 'Socket ID' set to 'K1', 'Local Strat Address' set to 'D1000', 'Station No.' set to 'K1', 'Function Code' set to '0x10 Write multiple registers', 'Data Address' set to 'K100', and 'Count' set to 'K20'. At the bottom are buttons for 'OK' and 'Cancel'.

Настройка S_OPEN сокетa ID2:



The screenshot shows the 'S_OPEN Parameter Setting' dialog box for ID2. It is divided into two main sections. The top section, 'Basic Setting', contains: 'Socket ID' set to 'K2', 'Communication type' set to 'TCP(K1)', 'Mode Selection' set to 'Client(K1)', 'Reg Start Position' set to 'HD204', and 'Flag Start Position' set to 'M200'. A blue note states: 'The "Basic Settings" program will take effect after downloading!'. The bottom section contains: 'Local Port' set to '1002', 'Buffer type' set to '8位', 'Timeout(10ms)' set to '100', 'Destination IP' set to '192.168.0.70', and 'Destination Port' set to '502'. A 'Used Space' field shows 'HD204-HD212,M200-M209'. At the bottom are buttons for 'Read From PLC', 'Write To PLC', 'OK', and 'Cancel'.

Настройка M_TCP сокета 2:

Modbus Tcp configuration

Socket ID: K2 Local Strat Address: D2000

Modbus TCP

Station No.: K1 Function Code: 0x10 Write multiple registers

Data Address: K200 Count: K20

OK Cancel

4.2 Инструкция чтения-записи для компорта

Когда осуществляется связь по Ethernet, для обеспечения нормальной реализации связи рекомендуется использовать инструкцию чтения параметров порта связи [CFGCR] и инструкцию записи [CFGCW] при составлении программы связи.

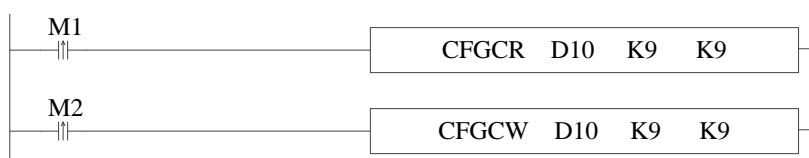
Сначала, вызвав инструкцию CFGCR, соответствующие параметры порта связи считываются в указанную группу регистров, затем пользователь изменяет соответствующие значения в группе регистров по мере необходимости, а затем записывает измененные значения группы регистров в соответствующую конфигурацию порта связи с помощью инструкции CFGCW. См. раздел 6-5 руководства по программированию PMP20.

4.2.1 Пример связи с параметрами компорта

Пример 1:

С помощью инструкции чтения параметров [CFGCR] и инструкции записи [CFGCW] параметры сети ПЛК считываются в регистр D10~D18, а параметры сети из регистра D10~D18 после модификации записываются в настройки последовательного порта ПЛК.

Программа для ПЛК:



При включении M1 происходит считывание сетевых параметров ПЛК. После изменения параметров сети включите M2, и измененные параметры сети могут

быть записаны в ПЛК. После записи, выключения и повторного включения питания ПЛК, параметры последовательного порта вступают в силу.

4.3 Флаг и регистр связи Ethernet

Регистры связи:

Адрес	Формат	Функция	Пояснение
SD1905	Hex	Номер IP-сети	Первые два байта IP-адреса
SD1906	Hex	Номер станции IP-устройства	Последние два байта IP-адреса
SD1907	Hex	Маска подсети	Первые два байта маски подсети
SD1908	Hex		Последние два байта маски подсети
SD1909	Hex	Шлюз по умолчанию	Первые два байта шлюза по умолчанию
SD1910	Hex		Последние два байта шлюза по умолчанию
SD1920	Десятичные	Некорректный идентификатор сокета	Некорректный идентификатор сокета, действует только в том случае, если соединение не было создано
SD1921	Десятичные	Код ошибки	<p>1: идентификатор сокета находится за пределами диапазона</p> <p>2: незарегистрированный идентификатор сокета отправляет запрос на связь</p> <p>3: ошибка типа связи, выход за пределы диапазона 0---TCP 1---UDP</p> <p>4: Количество TCP-соединений выходит за пределы диапазона, максимум 32</p> <p>5: Количество UDP-соединений выходит за пределы диапазона, максимум 32</p> <p>6: ошибка режима связи, вне диапазона, 0---Сервер 1---Клиент</p>

Катушки связи:

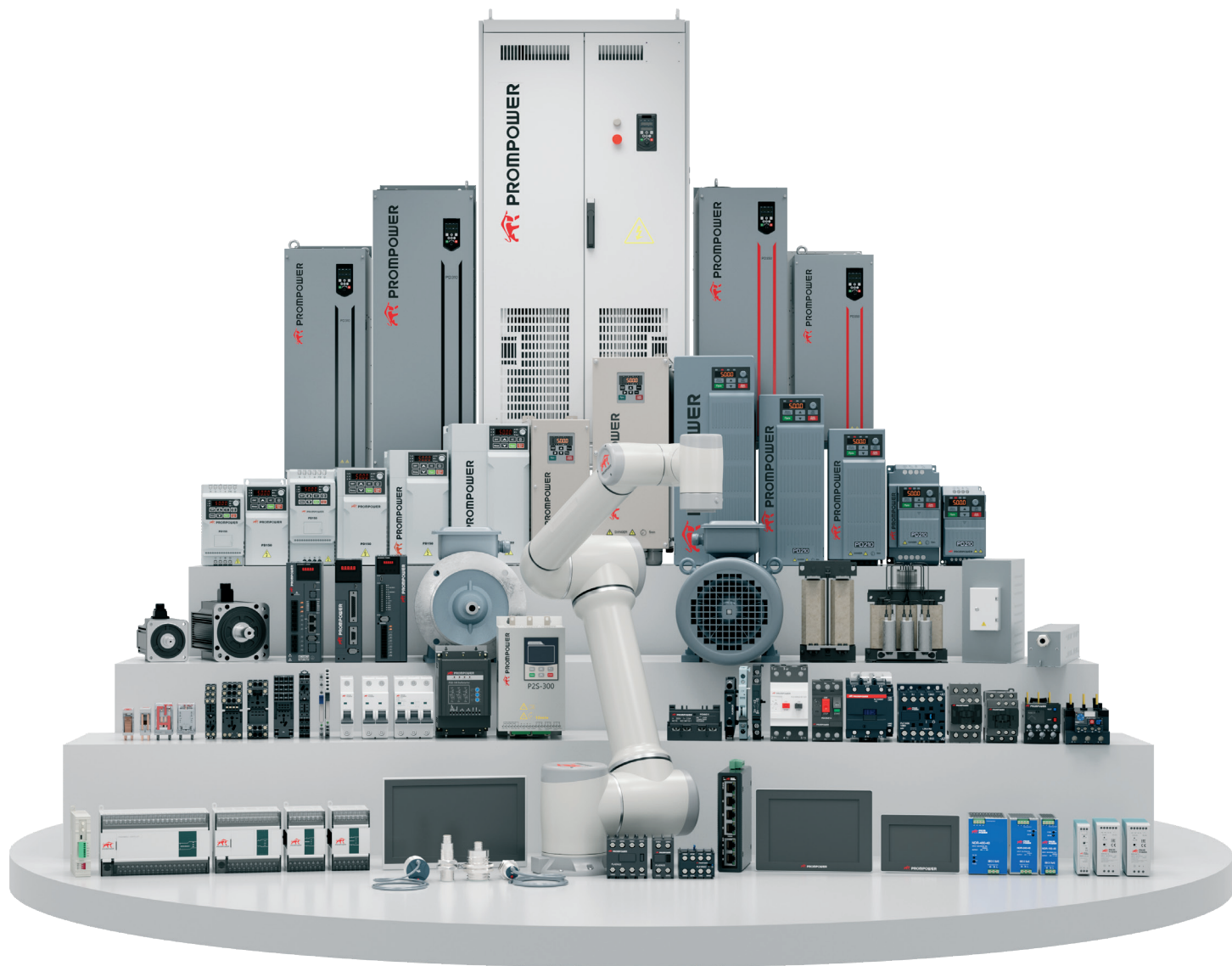
Адрес	Функция	Пояснение
SM1900	Вход на удаленный сервер успешно отмечен флагом	Включается при успешном удаленном соединении
SM1901	Флаг завершения инициализации Ethernet	Modbus TCP Server/TCP IP/ XNET
SM1902	Флаг подключения сетевого устройства	Например, коммутатор, маршрутизатор и другие сетевые устройства
SM1921	Флаг ошибки Ethernet	Устанавливается, когда ошибка в любом из сгенерированных SD1921

4.4 Список ошибок связи по сети Ethernet

Код ошибки	Объяснение
0	Коммуникация в норме
1	Сокет, который необходим для ОТКРЫТИЯ уже созданного соединения
2	Ошибка при создании сокета
3	Привязка назначенного порта не удалась
4	Ошибка TCPServerAccept
5	Ошибка TCPClientConnect
6	При вызове Send, Recv, Clos указанный сокет не создал соединение
7	Вызов инструкции Send не удался
8	Вызов инструкции Recv не удался
10	Указанная длина данных отправки выходит за пределы диапазона
11	Указанная длина принимаемых данных выходит за пределы диапазона
20	При обмене данными по протоколу UDP полученные данные не поступают с указанного IP-адреса
21	При обмене данными по протоколу UDP полученные данные не поступают с указанного порта
30	Фактическая длина полученных данных больше заданной

31	Фактическая длина полученных данных меньше заданной
40	Таймаут приема
50	Ошибка указанного целевого порта, Modbus TCP не является портом 502, Используемый порт находится вне диапазона (1~60000)
100	Ошибка приема
101	Таймаут приема
182	Ошибка № станции
183	Переполнение буфера при отправке
400	Ошибка кода функции
401	Ошибка адреса
402	Ошибка длины
403	Ошибка данных
404	Ведомая станция занята
405	Ошибка памяти (Flash ROM)

Профессиональное оборудование для промышленной автоматизации



Официальный дистрибьютор:



 **PROMPOWER**
www.prompower.ru

