Ethernet

6/04

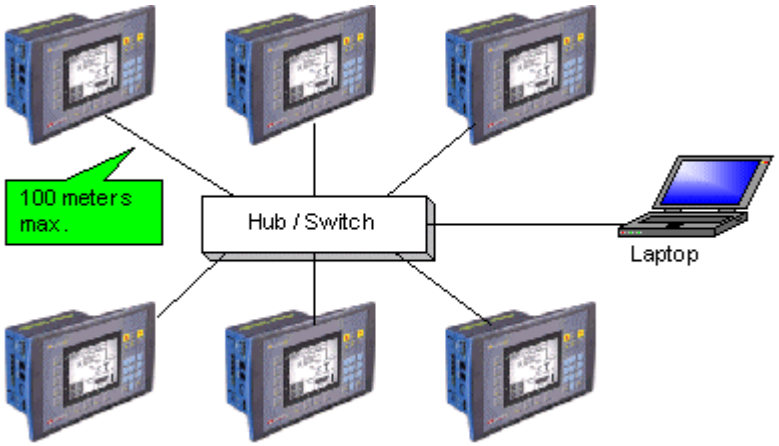
Table Of Contents

Ethernet

About Ethernet

General information regarding the parameters required to implement Ethernet is given below. A glossary of Ethernet terms is included at the bottom of this topic. To learn how to specifically define parameters within the VisiLogic Ethernet FBs, refer to Using Ethernet.

Unitronics' Ethernet uses *star topology.*



About Networks

Generally, controllers are part of a closed, internal control network. A closed network may be referred to as a LAN (Local Area Network) or an **Intra**net. When Intranets are connected via gateway devices, they form a WAN (Wide Area Network). The **Inter**net, which is made up of connected Intranets or LANs, is a form of WAN. Internet communications are via the TCP/IP protocol.

Large manufacturing companies, for example, may be made up of a number of factories, each of which contains its own LAN, closed control network. Within the company, all of these LANs may be connected by gateway devices, forming a proprietary WAN--a company Internet, which in turn may be connected to the Internet--the World Wide Web.

Within closed controller networks, Ethernet is becoming a common protocol. Ethernet communications are also via the TCP/IP protocol.

What is an IP address?

In order to enable a controller to communicate over Ethernet, you must assign it an IP address.

An IP address is a unique number which identifies a computer or controller on a TCP/IP network. These networks use the IP address to route messages to their destination. An IP address is a 32-bit numeric address which is divided into four numbers (octets). Each octet is separated by a period formatted as follows: 1.160.10.240. The decimal value in each octet can range from zero to 255, or 00000000 - 11111111 in binary notation.

Note ♦ The values '0' and '255' are restricted and should not normally be used.

Internally, within an Intranet, you can assign IP addresses at random as long as each one is unique within the Intranet. The common IP may be: 192.168.192.xx, where the last octet is the identifies the device on the network.

Note ♦ In the majority of cases, controllers are part of a closed control network (Intranet). The controllers' IP addresses are unique **only** within the Intranet, and cannot be accessed via the

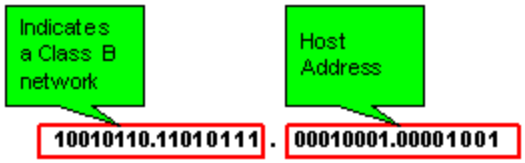
Internet--**unless an valid Internet IP address is purchased from a ISP and assigned to the controller**.

IP Addresses and Networks

In binary form, the IP address 68.212.226.204 is 10101000.11010100.11100010.11001100.

The 4 octets of the address are used to create classes of IP addresses. Networks are divided into 5 classes, according to size, as explained below. The octets are split into two sections: Net and Host. The Net section is represented by the first octet. It is used to identify the network that a device belongs to. The Host (sometimes called Node) section identifies the actual device on the network. The Host section is always contains by the final octets; how many octets is determined by the network class. There are five IP classes plus certain special addresses.

Although decimals are generally used to represent IP addresses, it is the binary value which determines which class of network the IP address belongs to. All nodes on a given network share the same network prefix but must have a unique host number.



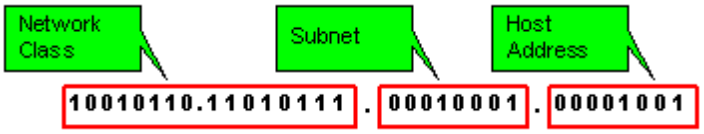
Default Network	The IP address of 0.0.0.0 is used for the default network.
Class A Network	This class is for very large networks. Binary address start with '0', meaning that the decimal value can be anywhere from 1 to 126. The first octet bits identify the network as Class A ; Octets 2, 3, and 4 (the next 24 bits) indicate the host within the network. An example of a Class A IP address is 102.168.212.226, where "102" identifies the network and "168.212.226" identifies the host on that network. **Note** ♦ The IP address 127.0.0.1 is used as the loopback address. This means that it is used by the host computer to send a message back to itself. It is commonly used for troubleshooting and network testing.
Class B Network	This class is used for medium-sized networks. The first two octets identify the network as Class B; Octets 3 and 4 (the remaining 16 bits) indicate the host within the network. Binary addresses start with '10', meaning that the decimal value can be anywhere from 128 to 191. An example of a Class B IP address is 168.212.226.204 where "168.212" identifies the network and "226.204" identifies the host on that network.
Class C Network	This class is used for small to medium-sized networks. This is the most common type of network. The first three octets identify the network as Class C; Octet 4 (8 bits) indicate the host within the network. Binary addresses start with '110', meaning that the decimal number can be anywhere from 192 to 223. An example of a Class C IP address is 200.168.212.226 where "200.168.212" identifies the network and "226" identifies the host on that network.
Class D Network	This class is used for multicasting, where a node sends a packet addressed to a special group address. Binary addresses start with '1110', therefore the decimal number can be anywhere from 224 to 239.
Class E Network	This class is used for experimental purposes only. Binary addresses start with '1111', therefore the decimal number can be anywhere from 240 to 255. Class E networks are used for experimentation. They have never been documented or utilized in a standard way.
Broadcast	Messages that are intended for all computers on a network are sent as broadcasts. These messages always use the IP address 255.255.255.255.

Subnet

A subnet is a part of a network.

All of the devices within a subnet share a common address component. On TCP/IP networks, subnets are defined as all devices **whose IP addresses have the same prefix**. Devices within a particular subnet might, for example, have IP addresses that start with 100.100.100.

Subnetting enables the network administrator to further divide the host part of the address into two or more subnets. In this case, a part of the host address is reserved to identify the particular subnet.



Subnet Mask

One of the crucial tasks for any router is knowing when a packet of information stays on its local network. For this, it uses a 'subnet mask'.

A network mask indicates which portion of the address identifies the network and which portion of the address identifies the node. Class A, B, and C networks have default masks, also known as natural masks, as shown below.

Class A: 255.0.0.0 - binary - 11111111.00000000.00000000.00000000

Class B: 255.255.0.0 - binary - 11111111.11111111.00000000.00000000

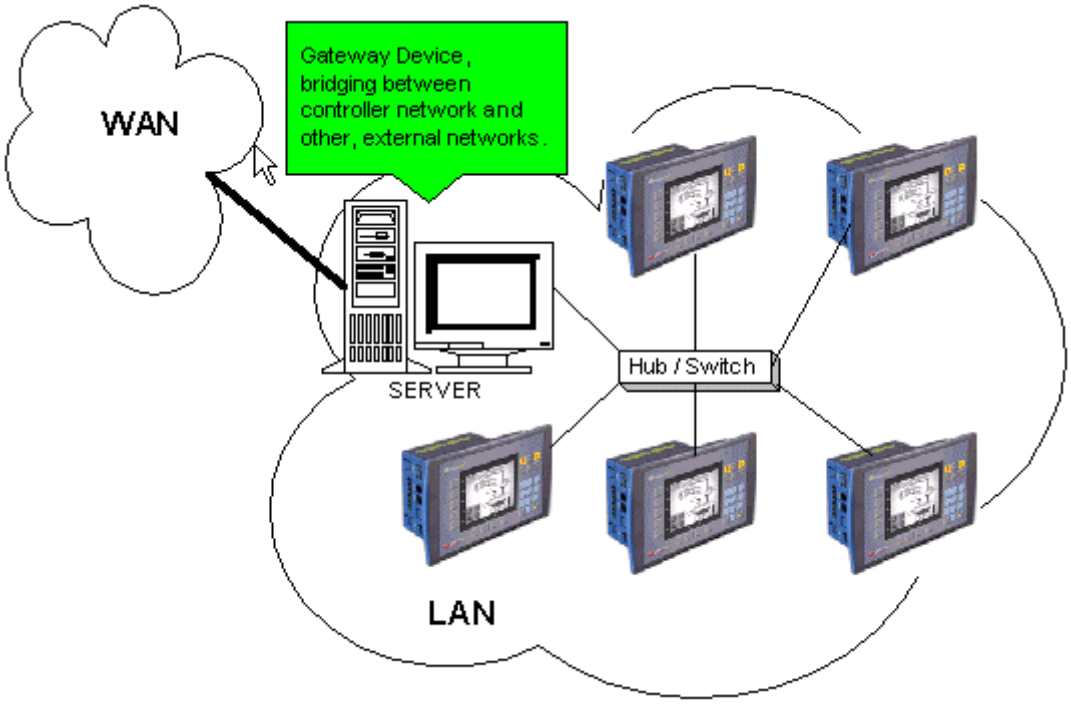
Class C: 255.255.255.0 - binary - 11111111.11111111.11111111.00000000

Since class C networks are the most common type, the most commonly used subnet mask usually reads "255.255.255.xx.". This tells the router that all messages with the sender and receiver having an address sharing the first three groups of numbers are on the same network, and shouldn't be sent out to another network. For example: The computer at address 192.168.192.254 sends a request to the computer at 192.168.192.252. The router, which sees all the packets, matches the first three groups in the address of both sender and receiver (192.168.192.), and keeps the packet on the local network.

Gateway

A gateway is special software, or a device running special software, that routes data between different networks.

In the case of control networks, the gateway is generally a PC. The gateway PC has its own IP address.



For example, a proxy server provides a gateway between a private network to the Internet. The proxy server is configured to enable a workstation to communicate with remote services on the Internet. In this case, the gateway acts as a barrier that allows a device to request information from the Internet and to receive information, but does not allow access to the host network by unauthorized users.

Note ♦ The IP address assigned to the gateway device is generally the last available address.

Socket

A software mechanism that connects an application to a network protocol. A program can, for example, send and receive TCP/IP messages by opening a socket and reading and writing data to and from the socket. Note that a socket is a software object, not a physical component.

Note that when TCP is used, the formal 'handshake' required by the protocol means that during each session occurring via a defined socket, other communications cannot flow through any of the other sockets until the current session has been terminated.

Such is not the case with UDP. Since there is no formal handshake, communications can continue to flow through a socket even when there are multiple requests.

Protocols: UDP and TCP

UDP stands for User Datagram Protocol. It is a connectionless protocol that, like TCP, runs on top of IP networks. Unlike TCP/IP, UDP/IP provides very few error recovery services, offering instead a direct way to send and receive datagrams over an IP network. It's used primarily for broadcasting messages over a network.

Note ♦ In Unitronics' implementation of Ethernet, UDP is a secure protocol. Here, UDP runs under MODBUS as well as under Unitronics' proprietary protocols; these additional layers provide the level of data security required by control applications.

TCP stands for Transmission Control Protocol. TCP is one of the main protocols in TCP/IP networks. Whereas the IP protocol deals only with packets, TCP enables two hosts to establish a connection and exchange streams of data. TCP guarantees delivery of data and also guarantees that packets will be delivered in the same order in which they were sent.

UDP takes messages from application process, attaches source and destination port number fields for the multiplexing/demultiplexing service, adds two other fields of minor importance, and passes the resulting "segment" to the network layer. The network layer encapsulates the segment into an IP datagram and then makes a best-effort attempt to deliver the segment to the receiving host. If the segment arrives at the receiving host, UDP uses the port numbers and the IP source and destination addresses to deliver the data in the segment to the correct application process. Note that with UDP there is no handshaking between sending and receiving transport-layer entities before sending a segment. For this reason, UDP is said to be connectionless.

TCP uses a three-way handshake before it starts to transfer data. UDP just blasts away without any formal preliminaries. Thus UDP does not introduce any delay to establish a connection. This is probably the principle reason why DNS runs over UDP rather than TCP -- DNS would be much slower if it ran over TCP. HTTP uses TCP rather than UDP, since reliability is critical for Web pages with text. But the TCP connection establishment delay in HTTP is an important contributor to the "world wide wait".

TCP maintains connection state in the end systems. This connection state includes receive and send buffers, congestion control parameters, and sequence and acknowledgment number parameters. UDP, on the other hand, does not maintain connection state and does not track any of these parameters. For this reason, a server devoted to a particular application can typically support many more active clients when the application runs over UDP rather than TCP.

The TCP segment has 20 bytes of header overhead in every segment, whereas UDP only has 8 bytes of overhead.

TCP has a congestion control mechanism that throttles the sender when one or more links between sender and receiver becomes excessively congested. This throttling can have a severe impact on real-time applications, which can tolerate some packet loss but require a minimum send rate. On the other hand, the speed at which UDP sends data is only constrained by the rate at which the application generates data, the capabilities of the source (CPU, clock rate, etc.) and the access bandwidth to the Internet. We should keep in mind, however, that the receiving host does not necessarily receive all the data - when the network is congested, a significant fraction of the UDP-transmitted data could be lost due to router buffer overflow. Thus, the receive rate is limited by network congestion even if the sending rate is not constrained.

Local Port

In TCP/IP and UDP networks, a port is an endpoint to a logical connection and the way a client program specifies a specific server program on a computer in a network.

The port numbers are divided into three ranges: the Well Known Ports, the Registered Ports, and the Dynamic and/or Private Ports.

- The Well Known Ports, sometimes called the contact port, are those from 0 through 1023. The Well Known Ports numbers are assigned by the IANA and on most systems can only be used by system (or root) processes or by programs executed by privileged users.
  Note ♦ Port 502 is reserved for SCADA.

- The Registered Ports are those from 1024 through 4915. The Registered Ports are listed by the IANA and on most systems can be used by ordinary user processes or programs executed by ordinary users.
- The Dynamic and/or Private Ports are those from 49152 through 65535

To the extent possible, these same port assignments are used with the UDP [RFC768].

Port Number Description
1 TCP Port Service Multiplexer (TCPMUX)
5 Remote Job Entry (RJE)
7 ECHO
18 Message Send Protocol (MSP)
20 FTP -- Data
21 FTP -- Control
22 SSH Remote Login Protocol
23 Telnet
25 Simple Mail Transfer Protocol (SMTP)
29 MSG ICP
37 Time
42 Host Name Server (Nameserv)
43 WhoIs
49 Login Host Protocol (Login)
53 Domain Name System (DNS)
69 Trivial File Transfer Protocol (TFTP)
70 Gopher Services
79 Finger
80 HTTP
103 X.400 Standard
108 SNA Gateway Access Server
109 POP2
110 POP3
115 Simple File Transfer Protocol (SFTP)
118 SQL Services
119 Newsgroup (NNTP)
137 NetBIOS Name Service
139 NetBIOS Datagram Service
143 Interim Mail Access Protocol (IMAP)
150 NetBIOS Session Service
156 SQL Server
161 SNMP
179 Border Gateway Protocol (BGP)
190 Gateway Access Control Protocol (GACP)
194 Internet Relay Chat (IRC)
197 Directory Location Service (DLS)
389 Lightweight Directory Access Protocol (LDAP)
396 Novell Netware over IP
443 HTTPS
444 Simple Network Paging Protocol (SNPP)
445 Microsoft-DS
458 Apple QuickTime
502 MODBUS
546 DHCP Client
547 DHCP Server
563 SNEWS
569 MSN
1080 Socks

Glossary

ARP

Address Resolution Protocol associates an IP address to a hardware address by requesting the sending machine for additional information called a MAC address. This only applies to Ethernet based networks.

Client

The client is generally an application that runs on a personal computer or workstation and relies on a server to perform some operations. For example, an e-mail client is an application that enables you to send and receive e-mail.

Client/server architecture

In this type of network architecture, each computer or process on the network is either a client or a server. Servers are powerful computers or processes dedicated to managing disk drives (file servers), printers (print servers), or network traffic (network servers). Clients are PCs or workstations on which users run applications. Clients rely on servers for resources, such as files, devices, and even processing power.

Another type of network architecture is known as a peer-to-peer architecture because each node has equivalent responsibilities. Both client/server and peer-to-peer architectures are widely used, and each has unique advantages and disadvantages.

DHCP

Dynamic Host Configuration Protocol is a protocol for organizing and simplifying the administration of IP addresses for local machines. In many cases (such as with WinRoute) A DHCP server is built into the gateway for further simplification.

DNS

Domain Name System is a naming scheme for IP addressing. For example www.kerio.com is a domain name and has an associated IP address. A DNS server matches domain names to an IP address. We use the domain name system because it is easier to remember a domain name than a string of numbers.

Firewall

A filtering module located on a gateway machine that examines all incoming and outgoing traffic to determine if it may be routed to its destination. WinRoute Lite is a simple Firewall based on Network Address Translation.

Gateway

The point of entrance from one network to another. A gateway is responsible for the proper distribution of data coming in and going out of a local area network. WinRoute must be installed on the gateway machine, also referred to as the host computer or network router.

ICMP

Internet Control Message Protocol uses datagrams to report errors in transmission between the host and gateway.

IP address

An IP address is the unique 32-bit number, which identifies a computer in a network. In order to communicate across wide area networks, each computer must have a unique IP address. Local area networks cannot directly communicate across wide area networks because they are defined by a private class of IP's.

Local Area Network

A Local Area Network (LAN) is a group of interconnected computers with the ability to share resources without having to access a wide area network.

MAC Address

A Media Access Control (MAC) address is a hard-coded interface identification used by layer 2 devices (switch or bridge) for proper forwarding of frames between computers on a network.

NAT

Network Address Translation is an Internet standard that enables a local-area network (LAN) to use one set of IP addresses for internal traffic and a second set of addresses for external traffic. A NAT box located where the LAN meets the Internet makes all necessary IP address translations.

NAT serves three main purposes:

- Provides a type of firewall by hiding internal IP addresses
- Enables a company to use more internal IP addresses. Since they're used internally only, there's no possibility of conflict with IP addresses used by other companies and organizations.
- Allows a company to combine multiple ISDN connections into a single Internet connection.

Network interface

A network interface may be an Ethernet card, modem, ISDN card, etc. The computer sends and receives packets by means of the network interface.

Network Mask

A Network mask is used to group IP addresses together. Routers use a subnet mask to define the group (or IP subnet) to which an IP address belongs so that it can identify the correct interface from which it should forward an IP packet.

Packet

When data is transmitted over the network it is broken up into smaller pieces called packets and individually routed to their destination. This way if one packet is not properly received, the receiving party can request resubmission of the single packet, as opposed to the entire piece of data. Each packet contains headers, which are responsible for the successful transmission of the packet, and a data part, which contains a portion of the original data being transmitted over the network. The term packet is used when referring to layer 3 devices (i.e. a router). A frame is the term used when referring to layer two devices (i.e. a switch).

Peer-to-peer architecture

A type of network in which each workstation has equivalent capabilities and responsibilities. This differs from client/server architectures, in which some computers are dedicated to serving the others. Peer-to-peer networks are generally simpler, but they usually do not offer the same performance under heavy loads.

Port

A port, in terms of TCP/IP, is a 16-bit number (the allowed range being 1 through 65535) used by the protocols of the transport layer - the TCP and UDP protocols. Ports are used to address applications. In other words, when a packet is received by the computer, the operating system uses port information to determine which application will receive the data within the packet.

Port Mapping

Port mapping is an advanced feature of WinRoute that allows servers to be hosted securely behind NAT. When a packet is received by the WinRoute host it can be forwarded (by translating the destination information in the packet header) to another computer in the local network.

Protocol

Defines rules for the transmission of data.

RAS

Remote Access Service refers to the ability to dial into another computer or network remotely. In the context of WinRoute, RAS simply refers to a dial-up connection.

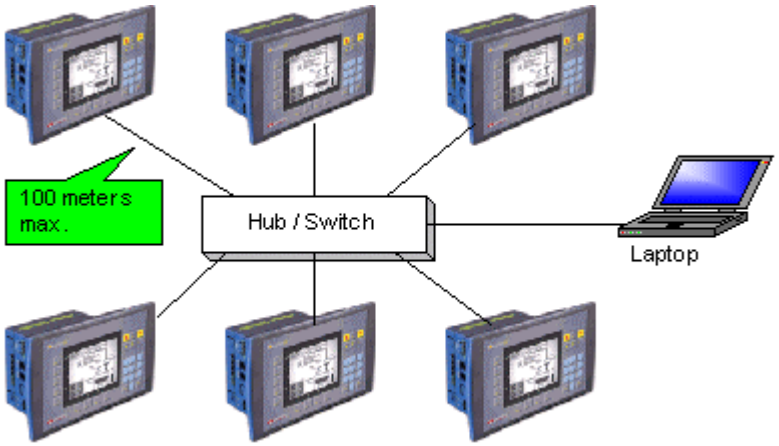
TCP/IP

TCP/IP is a suite of networking protocols used for communication across networks. It is the standard form of communication over the Internet. The two most significantly used Internet Protocols are TCP and UDP. Transmission Control Protocol (TCP) is a connection oriented protocol intended to provide reliability and to ensure that all data is transferred successfully from one computer to another. User Datagram Protocol (UDP) is a connectionless protocol that does not require any confirmation from the receiving party. UDP is more commonly used for multimedia and streaming applications.

Using Ethernet

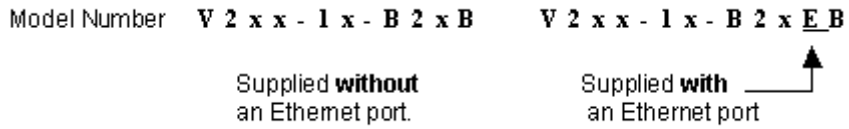
Unitronics currently supports both TCP and UDP protocols, as explained in the topic About Ethernet. This topic also contains general information about Ethernet, IP addressing, sockets, and ports.

Ethernet uses *star topology.*



In order to use Ethernet, your controller must comprise an *Ethernet port*.

V2xx Vision OPLCs can be ordered with or without an Ethernet port. The Ethernet port enables you to implement communications via TCP/IP, such as MODBUS over TCP. To check if your Vision controller was supplied with an installed Ethernet port, first check the device's model number. In addition, note that the Ethernet port is an RJ-45-type port that is lined with metal.

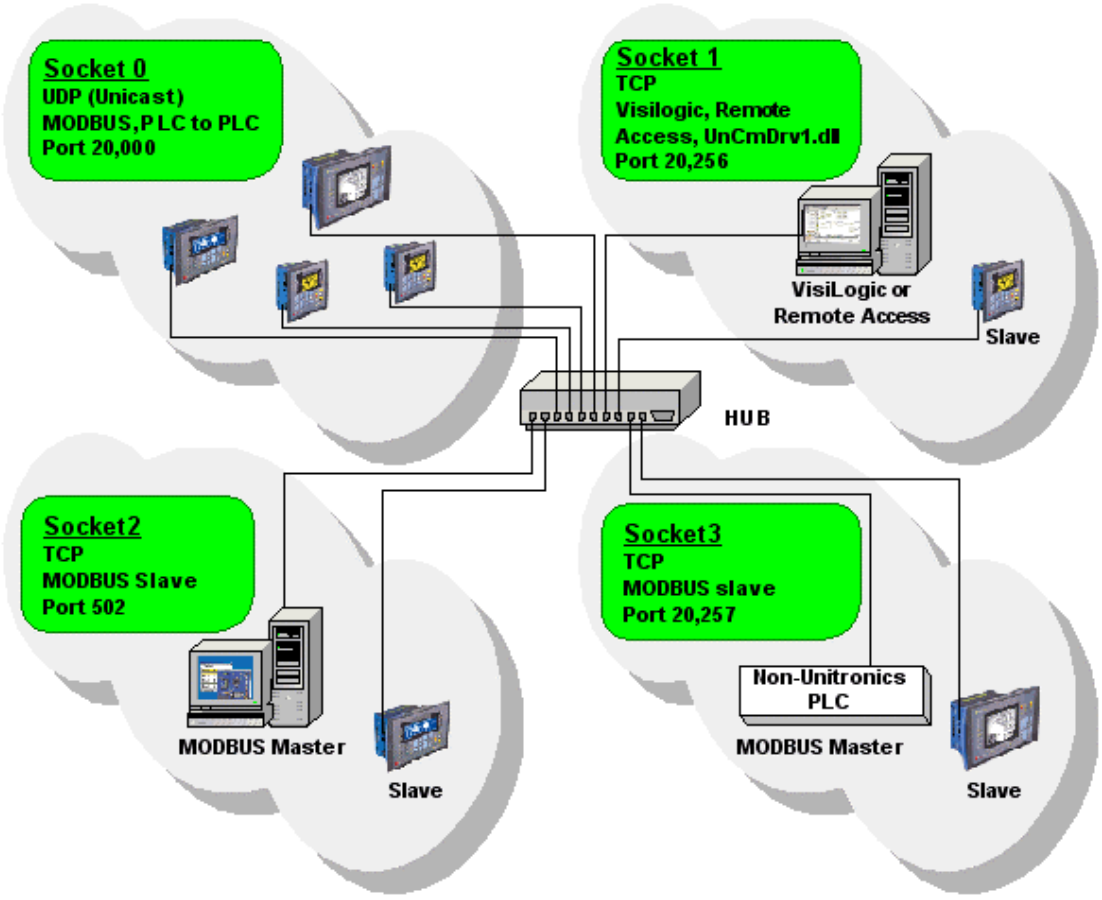


Via Ethernet, you can use the MODBUS IP FB to:

- Communicate data within a PLC network.
- Use a PC to access a PLC via MODBUS over TCP.
- Use MODBUS over TCP to enable non-Unitronics PLCs to access Unitronics PLCs, via MODBUS.

You can also use Ethernet to enable a PC running VisiLogic, Remote Access, or other communication .dll to access a networked PLC.

The **default** socket configuration enables you to implement these communication options as shown below:



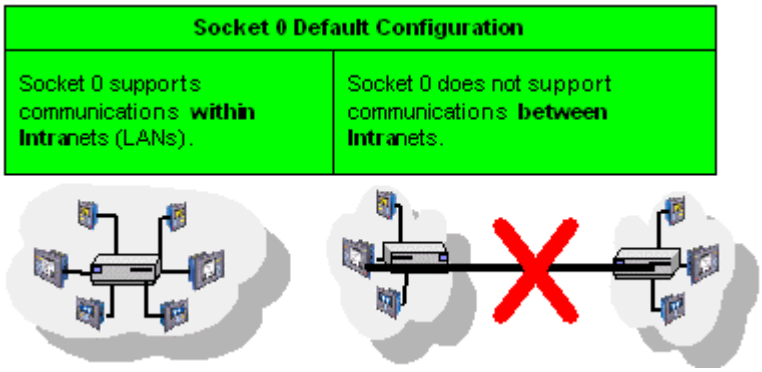
Default Socket Configuration

Vision controllers currently offer 4 sockets. The default socket configuration includes:

Socket	Protocol	Port Number	Function
0	UDP	20,000	Enables data to be both transmitted and received within a PLC network, via MODBUS. **Note♦** If you are using the default settings for Socket 0, note that data is sent via Unicast to IP: 255.255.255.255. port: 20,000 plus the last byte of the IP address originally assigned to the device. This is why Port numbers 20,000-20,255 are reserved for Socket 0.
1	TCP	20,256	Enables PC to PLC communication via UnCmDrv1.dll, including VisiLogic, Remote Access, and other Unitronics communication applications.
2	TCP	502	Set to 'listen' as slave (server), enables MODBUS applications such as OPC servers and SCADA systems which use MODBUS TCP over IP.
3	TCP	20,257	Set to 'listen' as slave (server), enables non-Unitronics PLCs to access Unitronics PLCs, via MODBUS.

Note♦ The default configuration means that, for most applications, you do not need to include a Socket Init FB in the ladder application. However, if, for example, your application requires 4 sockets for TCP, change the default configuration of Socket 0 from UDP to TCP via the Socket Init FBs.

♦ When using the default socket configuration, Socket 0 cannot be used to communicate data between routers, and therefore cannot transfer data between **Intra**nets as shown in the figure below. This is because the default configuration for Socket 0 uses Unicast.



♦ Note that when TCP is used, the formal 'handshake' required by the protocol means that, during each session occurring via a defined socket, other communications cannot flow through that socket until the current session has been terminated.

Such is not the case with UDP. Since there is no formal handshake, communications can continue to flow through a socket even when there are multiple requests.

General

When using Ethernet, use the MODBUS IP FBs. For detailed information regarding MODBUS IP commands, refer to the MODBUS IP help topics.

Note ♦ In order to implement Ethernet, a controller must be assigned an IP address. This is done via the Ethernet Card Init FB, which must be included in the Ladder applications of both master and slave controllers.
Class C-type addresses are recommended, as explained in the topic About Ethernet.

- When the Ethernet card finishes initialization, SB 142 rises. Use this as a condition before activating any Ethernet element, such as Socket Connect.
- An activating condition must be placed before the Ethernet Card Init FB. This may be assigned as a power-up task; however a one-shot transitional contact may also be used.
- Unitronics' proprietary COM Protocol FB, located on the FBs menu, which may ordinarily be used to access external slave devices, is not currently compatible with Ethernet.

Examples

PLC networks, PLC to PLC

Any controller within the network can be both master and slave. In order to be read by the master, a slave's application must contain the MODBUS IP Scan FB.

Using UDP to implement controller-to-controller communication

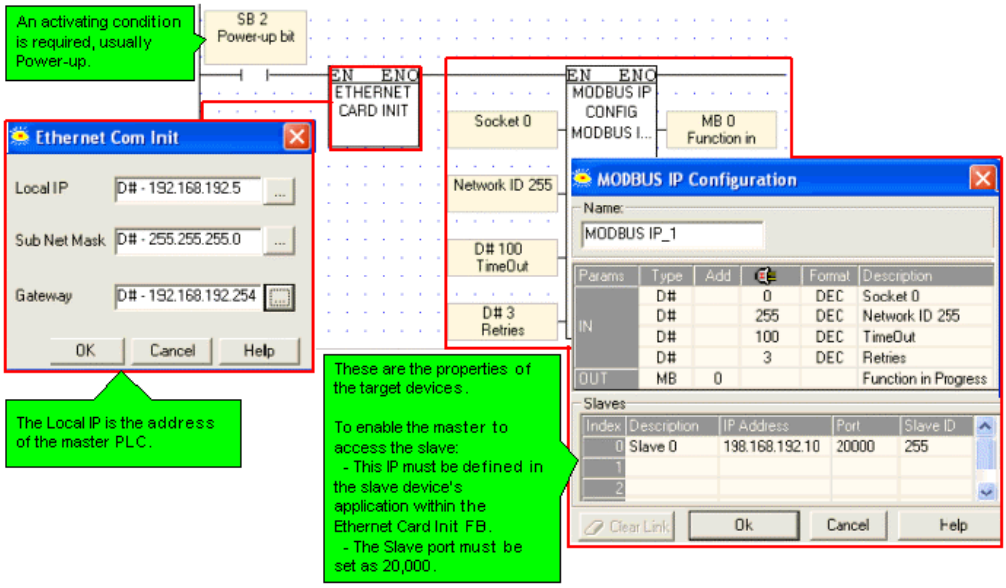
In order to communicate via Ethernet throughout your controller network, you must include an Ethernet Card Init FB in the ladder application of each networked controller. Remember that, when using UDP, **do not use the Socket: Connect or Socket: Close elements; these are only required by TCP applications.**

- ***Master***

  The master PLC Ladder application must include the elements shown below.

  Step 1: Initializing the Ethernet card and configuring MODBUS

  The MODBUS Configuration is linked to Socket 0, which is by default set to UDP.

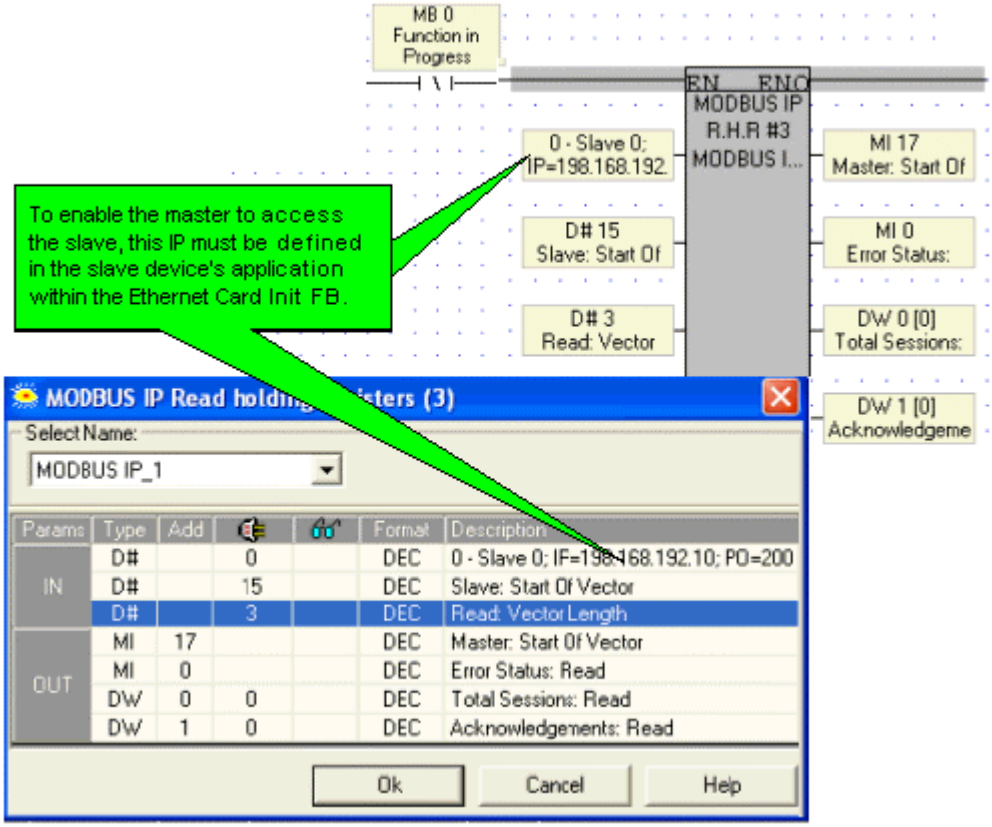


Note ♦ A PLC defined as a UDP master can communicate with a number of slave devices.

Step 2: Using MODBUS Commands

Note ♦ Note that the operand addresses in slave PLCs are indirect addresses (pointers). In the figure below,the Slave: Start of Vector parameter is 15. This means that the master will begin

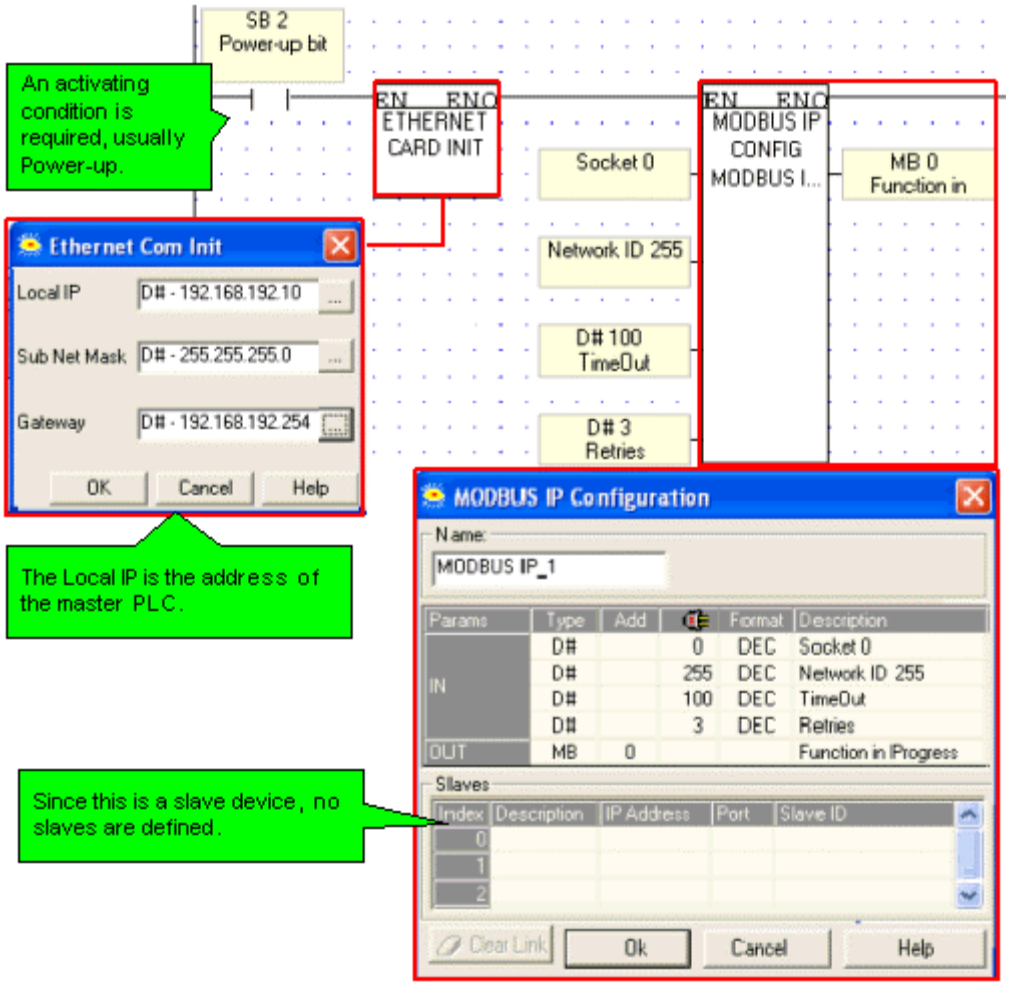
reading from MI 15 in the slave PLC. Since the Read: Vector Length parameter is 3, the function takes the values in MI 15, 16 and 17.
The Master: Start of Vector parameter is 17; therefore the values will be written into MI 17, 18, and 19 in the master device.



- ***Slave***

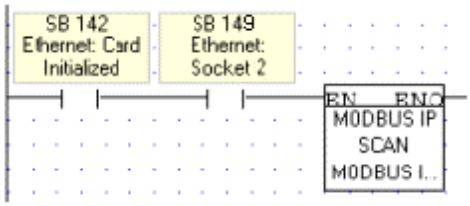
  The slave PLC Ladder application must include the elements shown below.

  Step 1: Initializing the Ethernet card and configuring MODBUS



Step 2: Scan

To enable the master PLC to access the slave, include a MODBUS Scan FB in the slave's application.



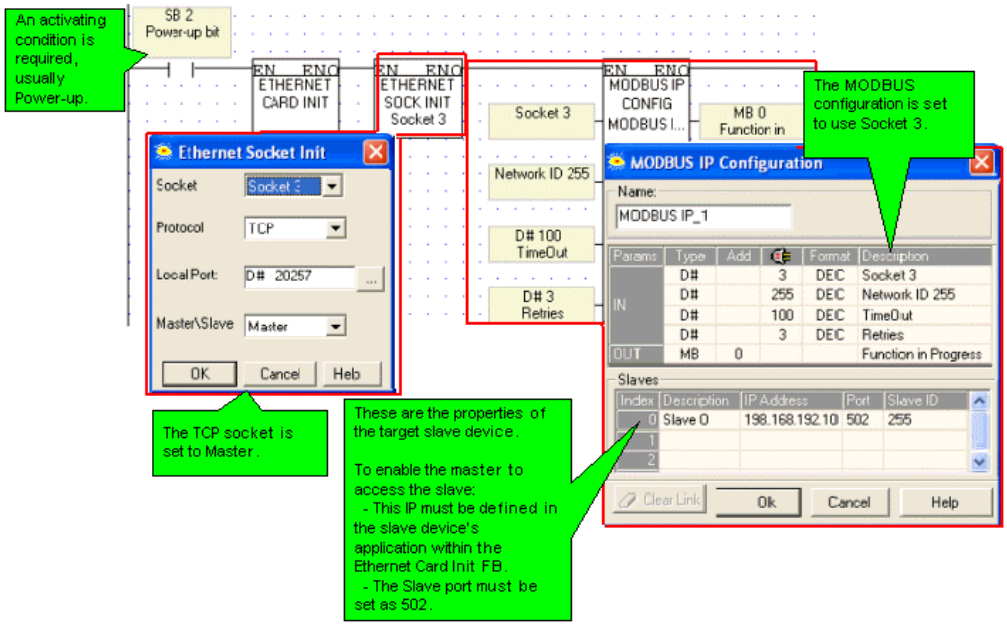
Using TCP to implement controller-to-controller communication

- ***Master***

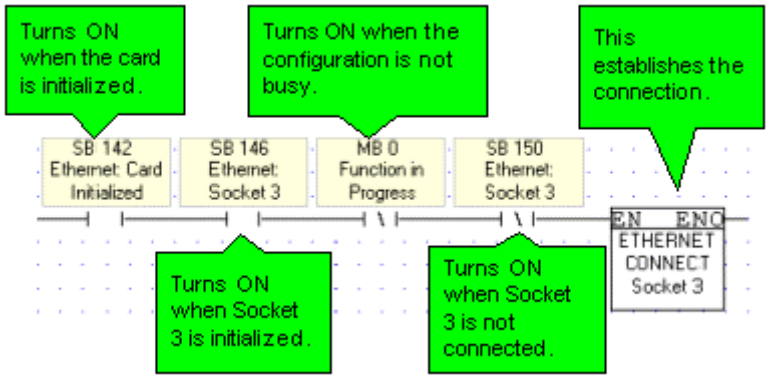
  The master PLC Ladder application must include the elements shown below.

  Step 1: Initializing the Ethernet card, Socket, and Configuring MODBUS

  In the figure below, the socket is configured to use TCP.



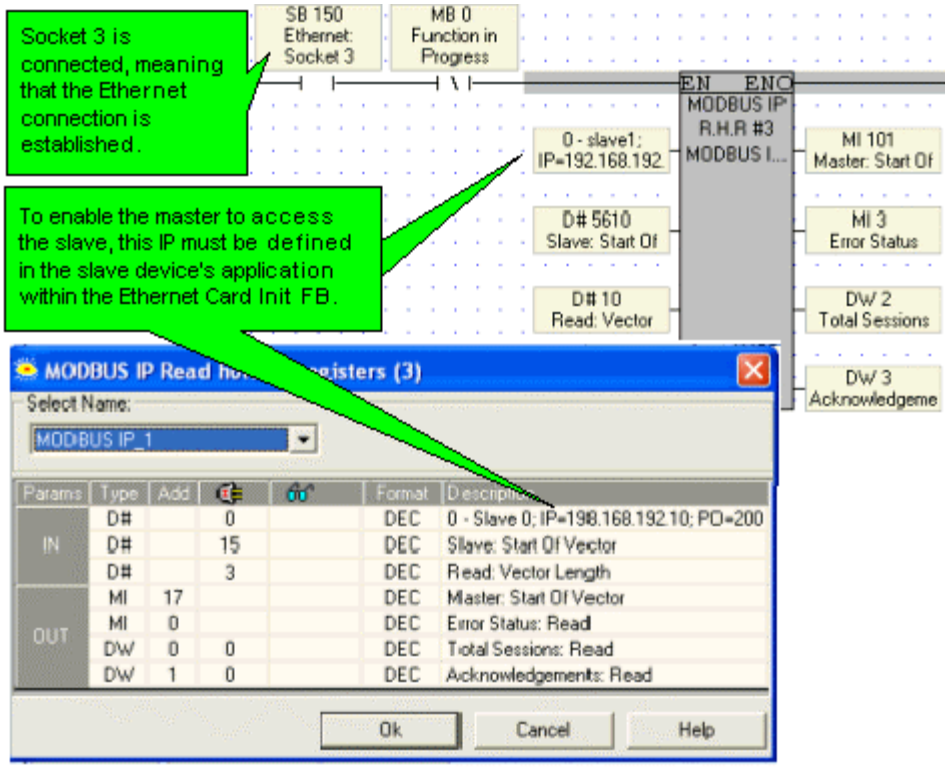
Step 2: Establishing the Ethernet Connection: Connect Socket



Note ♦ It is recommended that there be a time elapse of a few seconds after the Ethernet Card Initialization and before activating Socket Connect. A timer may be used for this purpose.

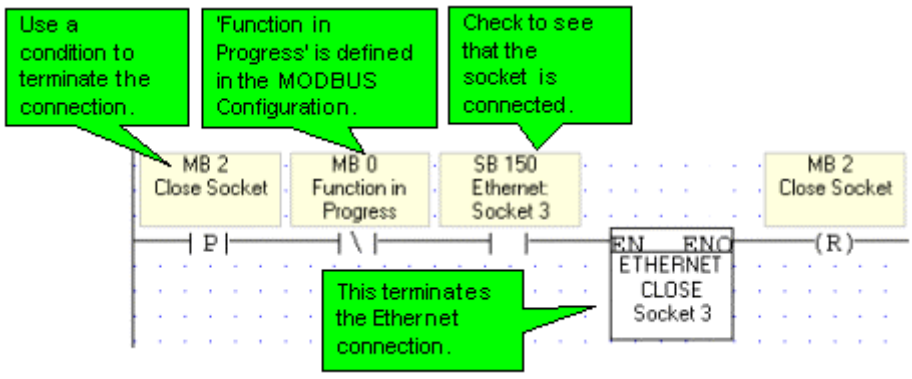
Step 3: Using MODBUS Commands

Note ♦ Note that the operand addresses in slave PLCs are indirect addresses (pointers). In the figure below, Below, the Slave: Start of Vector parameter is 15. This means that the master will begin reading from MI 15 in the slave PLC. Since the Read: Vector Length parameter is 3, the function takes the values in MI 15, 16 and 17.
The Master: Start of Vector parameter is 17; therefore the values will be written into MI 17, 18, and 19 in the master device.



Step 4: Terminating the Ethernet connection: Close Socket

When you terminate the connection, use the 'Function in Progress' MB to ensure that you do not terminate the connection while data is being communicated.

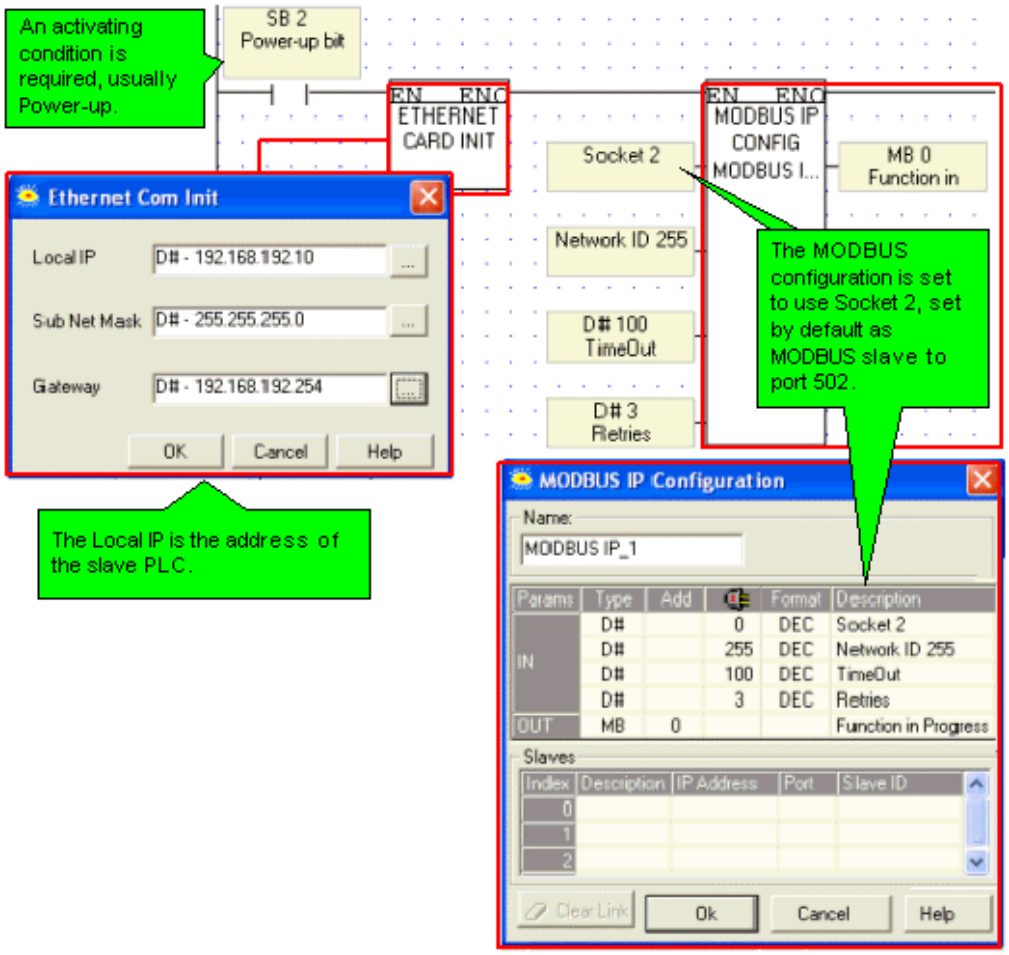


- ***Slave***

  The slave PLC Ladder application must include the elements shown below.

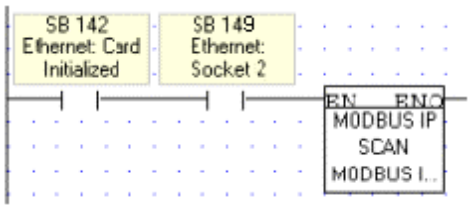
  Step 1: Initializing the Ethernet card, Socket, and Configuring MODBUS

  In the figure below, the socket is configured to use TCP.



Step 2: Scan

To enable the master PLC to access the slave, include a MODBUS Scan FB in the slave's application.

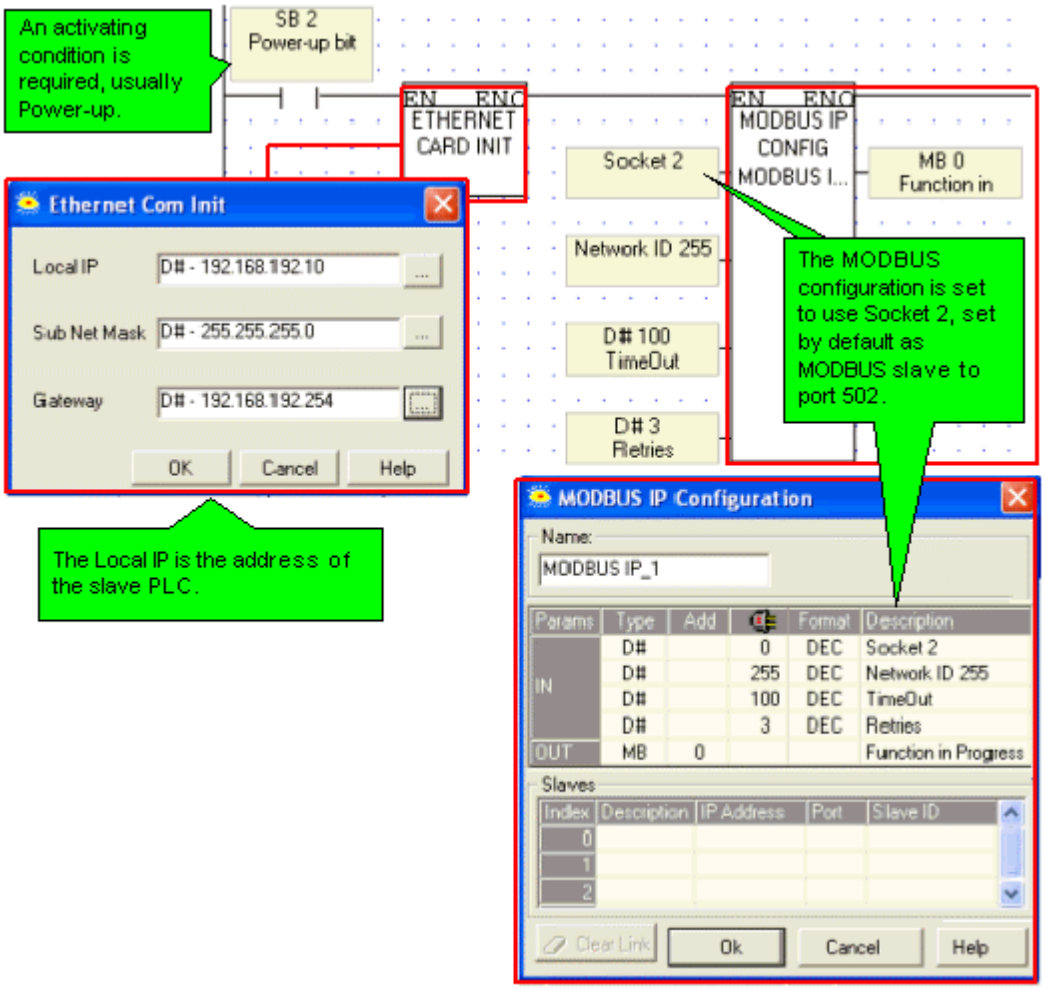


PC to PLC: Accessing PLC via SCADA

To enable the SCADA application to access the PLC, the PLC is defined as a slave device. The slave PLC Ladder application must include the elements shown below.

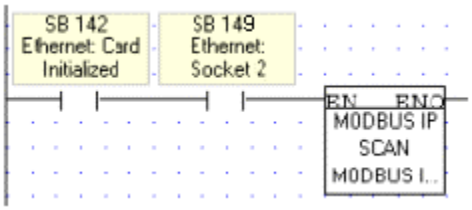
Step 1: Initializing the Ethernet card and configuring MODBUS

Port 502 is the well-known port for MODBUS applications.



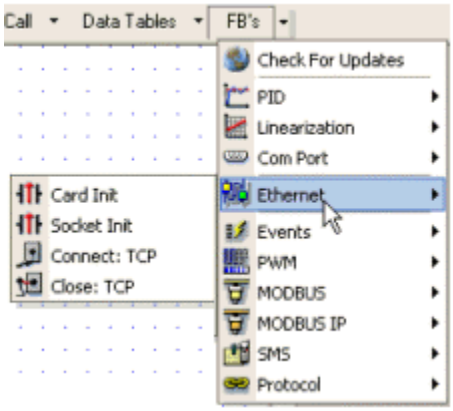
Step 2: Scan

To enable the SCADA application to access the slave, include a MODBUS Scan FB in the slave's application.



Ethernet Operations

The Ethernet FBs are grouped under *Ethernet on the FB's menu*.



Ethernet: Card Init

Ethernet: Socket Init

Ethernet: TCP Connect \ TCP Close

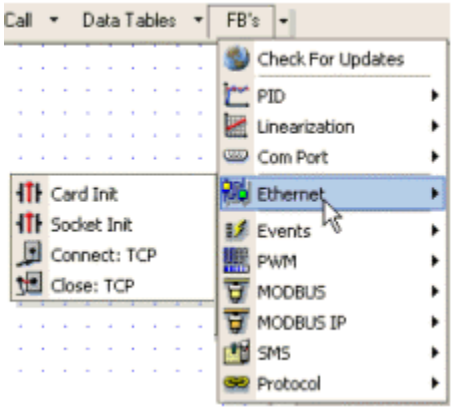
Ethernet FAQs

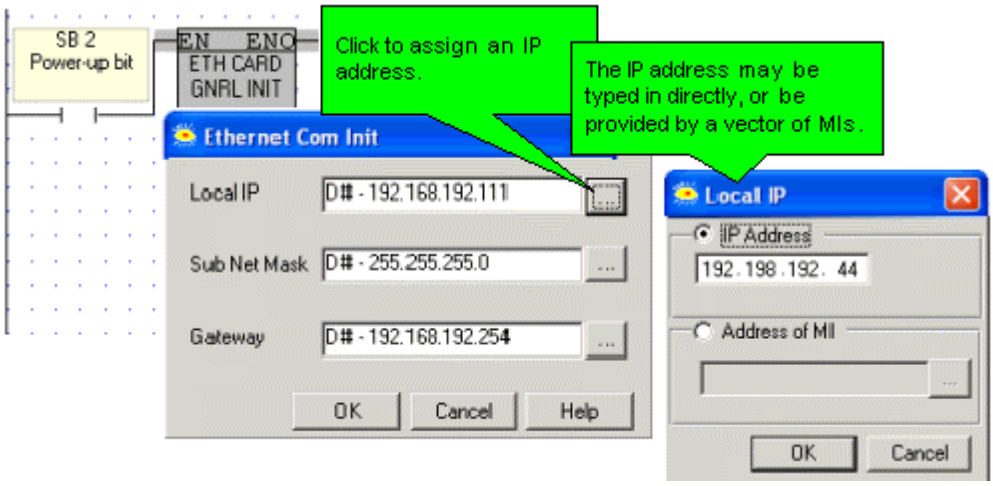
Using VisiLogic to access a remote controller via Ethernet

This feature will be supported in the next version of VisiLogic.

Ethernet: Card Init

This function is located under *Ethernet on the FB's menu*.





If you assign an IP address indirectly, via an MI vector, note that the vector is 4 MIs long. The low byte of each MI provides the number for an octet within the IP address.

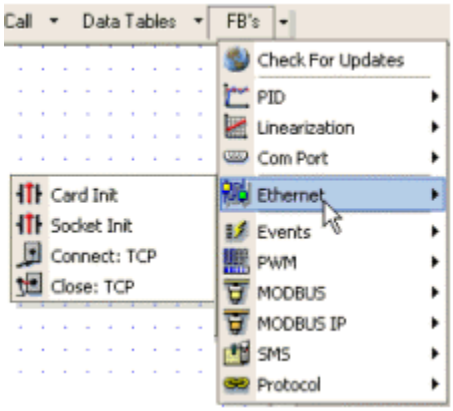
If, for example, the IP address is linked to MI 0, and the low bytes of MI 0 to MI 3 contain the values 192, 198, 192, 45, the IP address will be 192.198.192. 45.

Note ♦ In order to implement Ethernet, a controller must be assigned an IP address. This is done via the Ethernet Card Init FB, which must be included in the Ladder applications of both master and slave controllers. Information on IP addressing is given in the topic About Ethernet

- ♦ When the Ethernet card finishes initialization, SB 142 rises. Use this as a condition before activating any Ethernet element, such as Socket: Connect.
- ♦ An activating condition must be placed before the Ethernet Card Init FB. This may be assigned as a power-up task; however a one-shot transitional contact may also be used.
- ♦ If you have linked the IP address to a vector of MIs, and this condition is not activated, the IP address will not be assigned to the controller. Make sure, for example, that if you have used a power-up condition, that the controller does go through power-up.

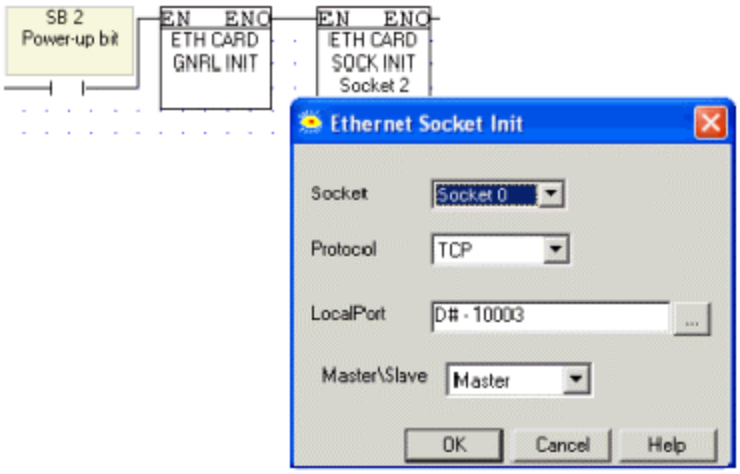
Ethernet: Socket Init

This function is located under *Ethernet on the FB's menu*.



Vision controllers currently offer 4 sockets.

The default configuration means that, for most applications, you do not need to include a Socket Init FB in the ladder application. However, if, for example, your application requires 4 sockets for TCP, change the default configuration of Socket 0 from UDP to TCP via the Socket Init FBs.



The default socket configuration includes:

Socket	Protocol	Port Number	Function
0	UDP	20,000	Enables data to be both transmitted and received within a PLC network, via MODBUS. **Note ♦** If you are using the default settings for Socket 0, note that data is sent via Unicast to IP: 255.255.255.255. port: 20,000 plus the last byte of the IP address originally assigned to the device. This is why Port numbers 20,000-20,255 are reserved for Socket 0.
1	TCP	20,256	Enables PC to PLC communication via UnCmDrv1.dll, including VisiLogic, Remote Access, and other Unitronics communication applications.
2	TCP	502	Set to 'listen' as slave (server), enables MODBUS applications such as OPC servers and SCADA systems which use MODBUS TCP over IP.
3	TCP	20,257	Set to 'listen' as slave (server), enables non-Unitronics PLCs to access Unitronics PLCs, via MODBUS.

Note ♦ When TCP is used, the formal 'handshake' required by the protocol means that during each session occurring via a defined socket, other communications cannot flow through any of the other sockets until the current session has been

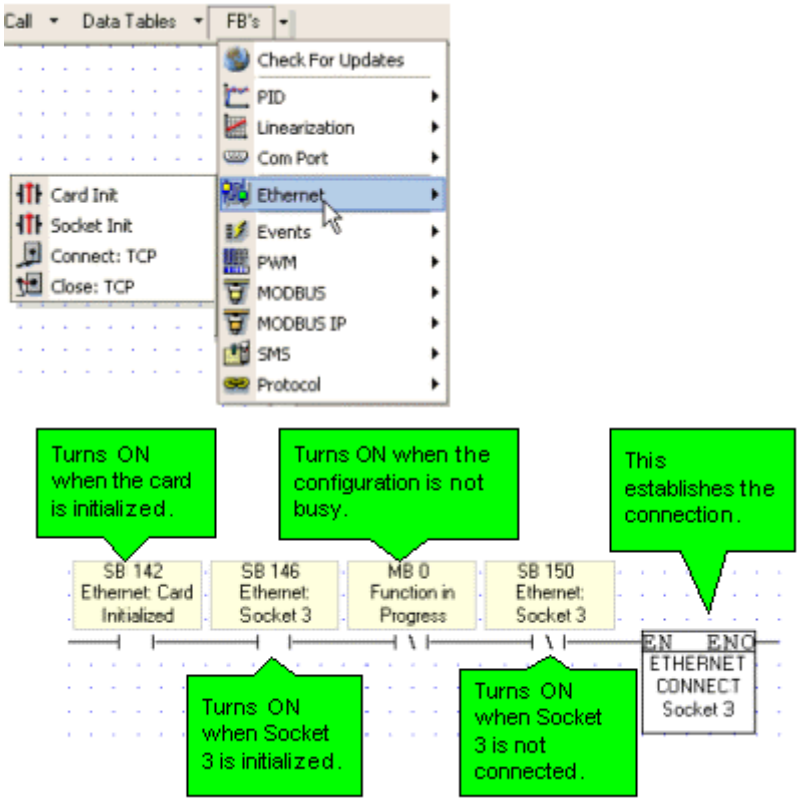
terminated.

Such is not the case with UDP. Since there is no formal handshake, communications can continue to flow through a socket even when there are multiple requests.

Ethernet: TCP Connect \ TCP Close

TCP applications require you to use a TCP: Connect FB to establish the Ethernet connection after the Ethernet card is initialized and before activating any of the MODBUS IP commands.

To terminate the session, use the TCP: Close FB. Both elements are located under ***Ethernet on the FB's menu.***



Ethernet: SBs & SIs

Parameter	Function
SB 141	Ethernet: Card Exists
SB 142	Ethernet: Card Initialized
SB 143	Ethernet: Socket 0 Initialized
SB 144	Ethernet: Socket 1 Initialized

SB 145	Ethernet: Socket 2 Initialized
SB 146	Ethernet: Socket 3 Initialized
SB 147	Ethernet: Socket 0 Connected
SB 148	Ethernet: Socket 1 Connected
SB 149	Ethernet: Socket 2 Connected
SB 150	Ethernet Status: Socket 3 Connected
SB 151	Ethernet Link: Communication is established
SB 152	Ethernet Link: 10baseT. When a 10baseT link is detected, turns ON/OFF during data transmit/ receive.
SB 153	Ethernet Link: 100baseT. When a 100baseT link is detected, turns ON/OFF during data transmit/ receive
SB 154	Ethernet: data collision
SB 155	Ethernet: Socket 0 is now transmitting data
SB 156	Ethernet: Socket 1 is now transmitting data
SB 157	Ethernet: Socket 2 is now transmitting data
SB 158	Ethernet: Socket 3 is now transmitting data

Parameter	Function
S1 145	Socket 0: Status
SI 146	Socket 1: Status
SI 147	Socket 2: Status
SI 148	Socket 3: Status

SI value	Message
0	Initialized to UDP, status: Closed
2	Initialized to TCP, status: Listen
14	Initialized to UDP, status: Ready
15	Initialized to UDP, status: Engaged in Transmit/Receive

Parameter	Function
SDW 14	Socket 0: Number of sent transmissions
SDW 15	Socket 1: Number of sent transmissions
SDW 16	Socket 2: Number of sent transmissions
SDW 17	Socket 3: Number of sent transmissions
SDW 18	Socket 0: Number of received transmissions
SDW 19	Socket 1: Number of received transmissions
SDW 20	Socket 2: Number of received transmissions
SDW 21	Socket 4: Number of received transmissions

Index