TeleBUS Protocols

User and Reference Manual

CONTROL MICROSYSTEMS

SCADA products... for the distance

48 Steacie Drive	Telephone:	613-591-1943
Kanata, Ontario	Facsimile:	613-591-1022
K2K 2A9	Technical Support:	888-226-6876
Canada		888-2CONTROL

TeleBUS Protocols User and Reference Manual

Trademarks

Table of Contents

TeleBUS Protocols Overview

The TeleBUS communication protocols provide a standard communication interface to SCADAPack and TeleSAFE controllers. The TeleBUS protocols are compatible with the widely used Modbus RTU and ASCII protocols. Additional TeleBUS commands provide remote programming and diagnostics capability.

The TeleBUS protocols operate on a wide variety of serial data links. These include RS-232 serial ports, RS-485 serial ports, radios, leased line modems, and dial up modems. The protocols are generally independent of the communication parameters of the link, with a few exceptions.

TeleBUS protocol commands may be directed to a specific device, identified by its station number, or broadcast to all devices. Using extended addressing up to 65534 devices may connect to one communication network.

The TeleBUS protocols provide full access to the I/O database in the controller. The I/O database contains user-assigned registers and general purpose registers. Assigned registers map directly to the I/O hardware or system parameter in the controller. General purpose registers can be used by ladder logic and C application programs to store processed information, and to receive information from a remote device.

Application programs can initiate communication with remote devices. A multiple port controller can be a data concentrator for remote devices, by polling remote devices on one port and responding as a slave on another port.

The protocol type, communication parameters and station address are configured separately for each serial port on a controller. One controller can appear as different stations on different communication networks. The port configuration can be set from an application program, from the TelePACE programming software, or from another Modbus compatible device.

Compatibility

There are two TeleBUS protocols. The TeleBUS RTU protocol is compatible with the Modbus RTU protocol. The TeleBUS ASCII protocol is compatible with the Modbus ASCII protocol.

Compatibility refers to communication only. The protocol defines communication aspects such as commands, syntax, message framing, error handling and addressing. The controllers do not mimic the internal functioning of any programmable controller. Device specific functions – those that relate to the hardware or programming of a specific programmable controller – are not implemented.

Serial Port Configuration

Communication Parameters

The TeleBUS protocols are, in general, independent of the serial communication parameters. The baud rate, word length and parity may be chosen to suit the host computer and the characteristics of the data link.

The port configuration can be set in four ways:

- using the TelePACE program;
- using the **set_port** function from a C application program;
- writing to the I/O database from a C or ladder logic application program; or
- writing to the I/O database remotely from a Modbus compatible device.

To configure a serial port through the I/O database, add the module, CNFG Serial port settings, to the Register Assignment Table.

RTU Protocol Parameters

The TeleBUS RTU protocol is an eight bit binary protocol. The table below shows possible and recommended communication parameters.

Parameter	***Possible Settings***	***Recommended Setting***
Baud Rate	see Baud Rate section below	see Baud Rate section below
Data Bits	8 data bits	8 data bits
Parity	None Even Odd	none
Stop bits	1 stop bit 2 stop bits[1]	1 stop bit
Flow control	Disabled	disabled
Duplex	see Duplex section below	see Duplex section below

[1] Not applicable to SCADAPack 350

ASCII Protocol Parameters

The TeleBUS ASCII protocol is an seven bit character based protocol. The table below shows possible and recommended communication parameters.

Parameter	***Possible Settings***	***Recommended Setting***
Baud Rate	see Baud Rate section below	see Baud Rate section below
Data Bits	7 data bits 8 data bits	7 data bits
Parity	None Even Odd	none
Stop Bits	1 stop bit 2 stop bits[1]	1 stop bit

Parameter	*Possible Settings*	*Recommended Setting*
Flow Control	Enabled Disabled	disabled
Duplex	see Duplex section below	see Duplex section below

[1] Not applicable to SCADAPack 350

NOTE: Flow control should never be enabled with modems or in noisy environments. Noise can result in the accidental detection of an XOFF character, which shuts down communication. Flow control is not recommended for any environment, but can be used on high quality, full duplex, direct wiring where speeds greater than 4800 baud are required.

Baud Rate

The baud rate sets the communication speed. The type of serial data link used determines the possible settings. The table below shows the possible settings for SCADAPack and TeleSAFE controllers. Note that not all port types and baud rates are available on all controller ports.

Port Type	*Possible Settings*	*Recommended Setting*
RS-232 or RS-232 Dial-up modem	75 baud 110 baud 150 baud 300 baud 600 baud 1200 baud 2400 baud 4800 baud 9600 baud 19200 baud 38400 baud 57600 baud 115200 baud	Use the highest rate supported by all devices on the network.
RS-485	75 baud 110 baud 150 baud 300 baud 600 baud 1200 baud 2400 baud 4800 baud 9600 baud 19200 baud 38400 baud 57600 baud 115200 baud	Use the highest rate supported by all devices on the network.

Duplex

The TeleBUS protocols communicate in one direction at a time. However the type of serial data link used determines the duplex setting. The table below shows the possible settings for SCADAPack and TeleSAFE controllers. Note that not all port types are available on all controllers.

Port Type	***Possible Settings***	***Recommended Setting***
RS-232 or RS-232 Dial-up modem	half duplex full duplex	Use full duplex wherever possible. Use half duplex for most external modems.
RS-485	half duplex full duplex	Slave stations always use half duplex. Master stations can use full duplex only on 4 wire systems.

Protocol Parameters

The TeleBUS protocols operate independently on each serial port. Each port may set the protocol type, station number, protocol task priority and store-and-forward messaging options.

The port configuration can be set in four ways:

- using the TelePACE or ISaGRAF programs;
- using the **set_protocol** function from a C or C++ application program;
- writing to the I/O database from a C, C++, ISaGRAF or ladder logic application program;
- writing to the I/O database remotely from a Modbus compatible device.

To configure protocol settings through the I/O database, add the module, CNFG Protocol settings, to the Register Assignment for TelePACE applications or use the setprot function in ISaGRAF applications.

Protocol Type

The protocol type may be set to emulate the Modbus ASCII and Modbus RTU protocols, or it may be disabled. When the protocol is disabled, the port functions as a normal serial port.

Station Number

The TeleBUS protocol allows up to 254 devices on a network using standard addressing and up to 65534 devices using extended addressing. Station numbers identify each device. A device responds to commands addressed to it, or to commands broadcast to all stations.

The station number is in the range 1 to 254 for standard addressing and 1 to 65534 for extended addressing. Address 0 indicates a command broadcast to all stations, and cannot be used as a station number. Each serial port may have a unique station number.

Task Priority

A task is responsible for monitoring each serial port for messages. The real time operating system (RTOS) schedules the tasks with the application program tasks according to the task priority. The priority can be changed only with the **set_protocol** function from an application program.

The default task priority is 3. Changing the priority is not recommended.

Store and Forward Messaging

Store and forward messaging re-transmits messages received by a controller. Messages may be re-transmitted on any serial port, with or without station address translation. A user-defined translation table determines actions performed for each message. The ***Store and Forward Messaging*** section below describes this feature in detail.

Store and forward messaging may be enabled or disabled on each port. It is disabled by default.

I/O Database

The TeleBUS protocols read and write information from the I/O database. The I/O database contains user-assigned registers and general purpose registers.

User-assigned registers map directly to the I/O hardware or system parameter in the controller. Assigned registers are initialized to the default hardware state or system parameter when the controller is reset. Assigned output registers do not maintain their values during power failures. However, output registers do retain their values during application program loading.

General purpose registers are used by ladder logic and C application programs to store processed information, and to receive information from remote devices. General purpose registers retain their values during power failures and application program loading. The values change only when written by an application program or a communication protocol.

The I/O database is divided into four sections.

- Coil registers are single bits which the protocols can read and write. Coil registers are located in the digital output section of the I/O database. The number of registers depends on the controller. Coil registers are numbered from 1 to the maximum for the controller.
- Status registers are single bits which the protocol can read. Status registers are located in the digital input section of the I/O database. The number of registers depends on the controller. Status registers are numbered from 10001 to the maximum for the controller.
- Input registers are 16 bit registers which the protocol can read. Input registers are located in the analog input section of the I/O database. The number of registers depends on the controller. Input registers are numbered from 30001 to the maximum for the controller.
- Holding registers are 16 bit registers that the protocol can read and write. Holding registers are located in the analog output section of the I/O database. The number of registers depends on the controller. Holding registers are numbered from 40001 to the maximum for the controller.

Accessing the I/O Database

TelePACE ladder logic programs access the I/O database through function blocks. All function blocks can access the I/O database. Refer to the ***TelePACE Ladder Logic Reference and User Manual*** for details.

ISaGRAF applications access the I/O database through dictionary variables with assigned network addresses or using Permanent Non-Volatile Modbus registers. See the ***ISaGRAF User and Reference Manual*** for details.

C language programs access the I/O database with two functions. The **dbase** function reads a value from the I/O database. The **setdbase** function writes a value to the I/O database. Refer to the ***TelePACE C Tools Reference and User Manual*** for full details on these functions.

Coil and Status Registers

Coil and status registers contain one bit of information, that is, whether a signal is off or on.

Writing any non-zero value to the register turns the bit on. Writing zero to the register turns the bit off. If the register is assigned to an I/O module, the bit status is written to the module output hardware or parameter.

Reading a coil or status register returns –1 if the bit is on, or 0 if the bit is off. The stored value is returned from general purpose registers. The I/O module point status is returned from assigned registers.

Input and Holding Registers

Input and holding registers contain 16 bit values.

Writing any value to a general purpose register stores the value in the register. Writing a value to an assigned register, writes the value to the assigned I/O module.

Reading a general purpose register returns the value stored in the register. Reading an assigned register returns the value read from the I/O module.

Exception Status

The exception status is a single byte containing controller specific status information. It is returned in response to the Read Exception Status function (see the ***Slave Mode*** section).

A C language application program can define the status information. The **modbusExceptionStatus** function sets the status information. Ladder logic programs cannot set this information.

Slave ID

The slave ID is a variable length message containing controller specific information. It is returned in response to the Report Slave ID function (see the ***Slave Mode*** section).

A C language application program can define the information and the length of the message. The **modbusSlaveID** function sets the information. Ladder logic programs cannot set this information.

Extended Station Addressing

The TeleBUS RTU and ASCII protocols support two type of Modbus station addressing. Standard Modbus addressing allows a maximum of 255 stations and is compatible with standard Modbus devices.

Extended Modbus addressing allows a maximum of 65534 stations. Extended Modbus addressing is fully compatible with standard Modbus addressing for addresses between 0 and 254.

Theory of Operation

The address field of a Modbus message is a single byte. Address 0 is a broadcast address; messages sent to this address are sent to all stations. Addresses 1 to 255 are station addresses. Figure 1 shows the format of a standard Modbus message.

Field	Address	Function	...
Size	1	1	N

Figure 1: Standard Modbus Message

The address field extension adds a two-byte extended address field to the message. Figure 2 shows the format of an extended address Modbus message.

Field	Address = 255	Extended Address (high)	Extended Address (low)	Function	...
Size	1	1	1	1	n

Figure 2: Extended Address Modbus Message

Messages for addresses 0 to 254 use the standard format message. The station address is stored in the address byte.

Messages for stations 255 to 65534 use the extended address format message. The address byte is set to 255. This indicates the extended address format is used. The actual address is stored in the two extended address bytes.

Station address 65535 is reserved and cannot be used as a station number. This station address is used in store-and-forward tables to indicate a disabled station.

Slave, master and store-and-forward stations treat the addresses in the same manner. The application program controls the use of the extended addressing format. It may enable or disable the extended addressing.

Slave Mode

The TeleBUS protocols operate in slave and master modes simultaneously. In slave mode the controller responds to commands sent by another device. Commands may be sent to a specific device or broadcast to all devices.

The TeleBUS protocols emulate the Modbus protocol functions required for communication with a host device. These functions are described below. It also implements functions for programming and remote diagnostics. These functions are not required for host communication, so are not described here.

A technical specification for the TeleBUS protocol is available from Control Microsystems. It describes all the functions in detail. In most cases knowledge of the actual commands is not required.

Broadcast Messages

A broadcast message is sent to all devices on a network. Each device executes the command. No device responds to a broadcast command. The device sending the command must query each device to determine if the command was received and processed. Broadcast messages are supported for some function codes that write information.

A broadcast message is sent to station number 0.

Function Codes

The table summarizes the implemented function codes. The maximum number of registers that can be read or written with one message is shown in the maximum column.

Function	*Name*	*Description*	*Maximum*
01	Read Coil Status	Reads digital output registers.	2000
02	Read Input Status	Reads digital input registers.	2000
03	Read Holding Register	Reads analog output registers.	125
04	Read Input Register	Reads analog input registers.	125
05	Force Single Coil	Writes digital output register.	1
06	Preset Single Register	Writes analog output registers.	1
07	Read Exception Status	Reads special information.	N/A
15	Force Multiple Coils	Writes digital output registers.	880
16	Preset Multiple Registers	Writes analog output registers.	60
17	Report Slave ID	Reads controller type information	N/A

Functions 5, 6, 15, and 16 support broadcast messages. The functions are described in detail below.

Read Coil Status

The Read Coil Status function reads data from the digital output section of the I/O database. Any number of registers may be read up to the maximum number. The read may start at any address, provided the entire block is within the valid register range. Each register is one bit.

Read Input Status

The Read Input Status function reads data from the digital input section of the I/O database. Any number of registers may be read up to the maximum number. The read may start at any address, provided the entire block is within the valid register range. Each register is one bit.

Read Holding Register

The Read Holding Register function reads data from the analog output section of the I/O database. Any number of registers may be read up to the maximum number. The read may start at any address, provided the entire block is within the valid register range. Each register is 16 bits.

Read Input Register

The Read Input Register function reads data from the analog input section of the I/O database. Any number of registers may be read up to the maximum number. The read may start at any address, provided the entire block is within the valid register range. Each register is 16 bits.

Force Single Coil

The Force Single Coil function writes one bit into the digital output section of the I/O database. The write may specify any valid register.

Preset Single Register

The Preset Single Register function writes one 16 bit value into the analog output section of the I/O database. The write may specify any valid register.

Read Exception Status

The Read Exception Status function reads a single byte containing controller specific status information. The information is defined by the application program. This function is included for compatibility with devices expecting to communicate with a Modicon PLC.

Force Multiple Coils

The Force Multiple Coils function writes single bit values into the digital output section of the I/O database. Any number of registers may be written up to the maximum number. The write may start at any address, provided the entire block is within the valid register range. Each register is 1 bit.

Preset Multiple Registers

The Preset Multiple Register function writes 16 bit values into the analog output section of the I/O database. Any number of registers may be written up to the maximum number. The write may start at any address, provided the entire block is within the valid register range. Each register is 16 bits.

Report Slave ID

The Report Slave ID function reads a variable length message containing controller specific information. The information and the length of the message is defined by the application program. This function is included for compatibility with devices expecting to communicate with a Modicon PLC.

Modbus Master Mode

The TeleBUS protocol may act as a communication master on any serial port. In master mode, the controller sends commands to other devices on the network. Simultaneous master messages may be active on all ports.

The protocol cannot support master mode and store-and-forward mode simultaneously on a serial port. Enabling store and forward messaging disables processing of responses to master mode commands. Master mode may be used on one port and store-and-forward mode on another port.

Modbus Function Codes

The table shows the implemented function codes. The maximum number of registers that can be read or written with one message is shown in the maximum column. The slave device may support fewer registers than shown; consult the manual for the device for details.

Function	***Name***	***Description***	***Maximum***
01	Read Coil Status	Reads digital output registers.	2000
02	Read Input Status	Reads digital input registers.	2000
03	Read Holding Register	Reads analog output registers.	125
04	Read Input Register	Reads analog input registers.	125
05	Force Single Coil	Writes digital output register.	1
06	Preset Single Register	Writes analog output registers.	1
15	Force Multiple Coils	Writes digital output registers.	880
16	Preset Multiple Registers	Writes analog output registers.	60

Read Coil Status

The Read Coil Status function reads data from coil registers in the remote device. Data can be written into the digital input or the digital output sections of the I/O database.

Any number of registers may be read up to the maximum number supported by the slave device or the maximum number above, whichever is less. The read may start at any address, provided the entire block is within the valid register range. Each register is one bit.

Read Input Status

The Read Input Status function reads data from input registers in the remote device. Data can be written into the digital input or the digital output sections of the I/O database.

Any number of registers may be read up to the maximum number supported by the slave device or the maximum number above, whichever is less. The read may start at any address, provided the entire block is within the valid register range. Each register is one bit.

Read Holding Register

The Read Holding Register function reads data from holding registers in the remote device. Data can be written into the analog input or the analog output sections of the I/O database.

Any number of registers may be read up to the maximum number supported by the slave device or the maximum number above, whichever is less. The read may start at any address, provided the entire block is within the valid register range. Each register is 16 bits.

Read Input Register

The Read Input Register function reads data from input registers in the remote device. Data can be written into the analog input or the analog output sections of the I/O database.

Any number of registers may be read up to the maximum number supported by the slave device or the maximum number above, whichever is less. The read may start at any address, provided the entire block is within the valid register range. Each register is 16 bits.

Force Single Coil

The Force Single Coil function writes one bit into a coil register in the remote device. The data may come from the digital input or digital output sections of the I/O database.

The write may specify any valid coil register in the remote device.

Preset Single Register

The Preset Single Register function writes one 16 bit value into a holding register in the remote device. The data may come from the analog input or output sections of the I/O database.

The write may specify any valid holding register in the remote device.

Force Multiple Coils

The Force Multiple Coils function writes single bit values coil registers in the remote device. The data may come from the digital input or digital output sections of the I/O database.

Any number of registers may be written up to the maximum number supported by the slave device or the maximum number above, which ever is less. The write may start at any address, provided the entire block is within the valid register range of the remote device. Each register is 1 bit.

Preset Multiple Registers

The Preset Multiple Register function writes 16 bit values into holding registers of the remote device. The data may come from the analog input or output sections of the I/O database.

Any number of registers may be written up to the maximum number supported by the slave device or the maximum number above, which ever is less. The write may start at any address, provided the entire block is within the valid register range of the remote device. Each register is 16 bits.

Enron Modbus Master Mode

The Enron Modbus protocol is based on the Modbus ASCII and RTU protocols. Message framing is identical to the Modbus protocols. However, there are many differences in message formatting and register numbering, at both the logical and protocol levels.

The document ***Specifications and Requirements for an Electronic Flow Measurement Remote Terminal Unit*** describes the Enron Modbus protocol.

Variable Types

There are ranges of Enron registers to hold short integers, long integers and single precision floats. The ranges are as follows.

Range	***Data Type***
1001 - 1999	Boolean
3001 - 3999	Short integer
5001 - 5999	Long integer
7000 - 9999	Float

In general, both Numeric and Boolean function codes can be used to read and write all types of registers. Consult the Enron Modbus specification for details.

Boolean Registers

Enron Modbus Boolean registers are usually numbered 1001 to 1999.

Boolean registers are read using Modbus command 1. Boolean registers are written using Modbus command 5 for single registers and 15 for multiple registers.

The address offset in the message is equal to the register number.

The number of Modbus registers is equal to the number of Enron registers.

The response format is identical to the Modbus response format.

Short Integer Registers

Enron Modbus Short Integer registers are usually numbered 3001 to 3999.

Short Integer registers are read using Modbus command 3. Short Integer registers are written using Modbus command 6 for single registers and 16 for multiple registers.

The address offset in the message is equal to the register number.

The number of Modbus registers is equal to the number of Enron registers.

The response format is identical to the Modbus response format.

Long Integer Registers

Enron Modbus Long Integer registers are usually numbered 5001 to 5999.

Long Integer registers are read using Modbus command 3. Long Integer registers are written using Modbus command 6 for single registers and 16 for multiple registers.

The address offset in the message is equal to the register number.

The number of Modbus registers requested is equal to the number of Enron registers.

The number of Modbus registers expected in the response is equal to two times the number of Enron registers.

Floating Point Registers

Enron Modbus Floating-point registers are usually numbered 7001 to 7999.

Floating-point registers are read using Modbus command 3. Floating-point registers are written using Modbus command 6 for single registers and 16 for multiple registers.

The address offset in the message is equal to the register number.

The number of Modbus registers requested is equal to the number of Enron registers.

The number of Modbus registers expected in the response is equal to two times the number of Enron registers.

Enron Modbus Function Codes

The following table shows the implemented function codes for Enron Modbus. The maximum number of registers that can be read or written with one message is shown in the maximum column. The slave device may support fewer registers than shown; consult the manual for the device for details.

Functions 129, 130, 132, 133, 135, 136, 138, and 139 may be broadcast, but some Enron Modbus slave devices may not support broadcast messages. Consult the manual for the device for details.

Function	*Name*	*Description*	*Maximum*
128	Read Enron Boolean	Read Enron Boolean registers	2000
129	Write Enron Boolean	Write Enron Boolean register	1
130	Write Enron Multiple Boolean	Write Enron Boolean registers	880
131	Read Enron Short Integer	Read Enron short integer register	125
132	Write Enron Short Integer	Write Enron short integer register	1
133	Write Enron Multiple Short Integer	Write Enron short integer registers	60
134	Read Enron Long Integer	Read Enron long integer register	62
135	Write Enron Long Integer	Write Enron long integer register	1
136	Write Enron Multiple Long Integer	Write Enron long integer registers	30
137	Read Enron Floating Point	Read Enron floating-point register	62
138	Write Enron Floating Point	Write Enron floating-point register	1
139	Write Enron Multiple Floating Point	Write Enron floating-point registers	30

Sending Messages

A master message is initiated in one of five ways:

- using the **master_message** function from a C or C++ application program; or
- using the **MSTR** function block from a TelePACE ladder logic program; or
- using the **MSIP** function block from a TelePACE ladder logic program; or
- using the **master** function in an ISaGRAF program; or
- using the **masterip** function in an ISaGRAF program.

These functions specify the port on which to issue the command, the function code, the type of station addressing, the slave station number, and the location and size of the data in the slave and master devices. The protocol driver, independent of the application program receives the response to the command.

The application program detects the completion of the transaction by:

- calling the **get_protocol_status** function in a C application program; or
- using the output of the **MSTR** function block in a TelePACE ladder logic program; or
- using the output of the **master** function in an ISaGRAF program.

A communication error has occurred if the slave does not respond within the expected maximum time for the complete command and response. The application program is responsible for detecting this condition. When errors occur, it is recommended that the application program retry several times before indicating a communication failure.

The completion time depends on the length of the message, the length of the response, the number of transmitted bits per character, the transmission baud rate, and the maximum message turn-around time. One to three seconds is usually sufficient. Radio systems may require longer delays.

Store and Forward Messaging

Store and forward messaging is required on systems where there is no direct link between a host computer and all the remote sites. This occurs on radio systems where the host computer transmission cannot be heard by all remote sites. It occurs on systems where one controller is used as a data concentrator for several remote units. With store and forward messaging, a request to a controller that cannot be directly accessed by a host is routed through an intermediate controller, which can communicate with both the host and the remote controller.

The TeleBUS protocol provides store and forward messaging through address translation. A controller configured for store and forward operation receives messages destined for a remote station, re-addresses them according to translation table, and forwards the message to the remote station. Responses from the remote station are processed in the same manner.

The TeleBUS protocol allows messages to be re-transmitted on the same port with address translation. This is used with radio systems. The radio at the intermediate site is used as a type of repeater. The protocol allows messages to be re-transmitted on a different port, with or without address translation. This is used where the intermediate controller is a bridge between two networks.

The TeleBUS protocol driver maintains diagnostics counters at the store and forward site on the number of messages received and transmitted to aid in the diagnosing of communication problems.

The protocol cannot support master mode and store-and-forward mode simultaneously on a serial port. Enabling store and forward messaging disables processing of responses to master mode commands. Master mode may be used on one port and store-and-forward mode on another port. Applications requiring both modes on a single port must switch the modes under control of the application program.

Translation Table

The translation table specifies address and communication port translation. The translation table differs for SCADAPack and SCADAPack 32 controllers. Each entry in the translation table for SCADAPack controllers has four components, as shown in the table entry below.

Port A	Station Address A	Port B	Station Address B

The entry defines a bi-directional transfer. A message (poll or reply) received for station A on port A is re-transmitted to station B on port B. A message received for station B on port B is re-transmitted to station A on port A.

Each entry in the translation table for SCADAPack 32 controllers has five components, as shown in the table entry below.

Slave Interface	Slave Station	Forward Interface	Forward Station	Forward IP Address

The Slave Interface entry contains the receiving slave interface the message is received from for each translation.

The Slave Station entry contains the Modbus station address of the slave message.

The Forward Interface entry contains the interface the message is forwarded from. When forwarding to a TCP or UDP network, the protocol type is selected for the Forward Interface.

The IP Stack automatically determines the exact interface (e.g. Ethernet1) to use when it searches the network for the Forward IP Address.

The Forward Station entry contains the Modbus station address of the forwarded message.

The Forward IP Address entry contains the IP address of the Forward Station. This field is blank unless a TCP or UDP network is selected for Forward Interface.

Table Size

The translation table holds 128 translation entries. This is sufficient to re-transmit one-half of 256 possible addresses. On a single port controller only 128 translations are required since each address must translate to a different address for re-transmission on the same port see Invalid Translations.

Invalid Translations

The following translations are not valid. The described action is taken when these translations are encountered.

- Re-transmission on the same port with the same address is not valid, except for broadcast messages. This restriction is required because many message responses are identical to the command. It is impossible for the master station to distinguish between the re-transmitted message and the response from the slave. The re-transmitted message would appear to be the response.
- The protocol re-transmits broadcast messages on the same port. Some stations will receive the broadcast message twice. The master station will also receive the message and may execute it if it is able to operate as a slave. The user must bear these consequences in mind when forwarding broadcast messages.
- The store and forward controller also processes broadcast messages.
- Translations where either of the station addresses are the same as the controller station address for the port, are not valid. The protocol processes these messages as if they were directed to the controller. It does not look up the address in the translation table.
- Translations with non-existent port numbers or invalid addresses are not valid.
- Multiple translations for a port and station address combination are not valid.
- Translations where one station is DISABLED and the other station is not, are not valid. A DISABLED translation is a valid translation.

Store and Forward Configuration

The Store and Forward configuration varies depending on the controller you are configuring. The configuration for each type of controller is described in the following sections.

SCADAPack Controller

An application program, written in TelePACE Ladder Logic or TelePACE C Tools and ISaGRAF IEC61131 or ISaGRAF IEC61131 C Tools programming, enables and configures store and forward messaging. A HMI host may enable and configure store and forward messaging through the controller I/O database.

TelePACE Ladder Logic

1. To enable the use of store and forward messaging on one or more serial ports the Configuration I/O Module ***CNFG Protocol Settings Method 1, 2 or 3*** must be added to the register assignment. The store and forward enable register must be set to enable.
2. Add the Configuration I/O Module ***CNFG Store and Forward*** to the register assignment to configure the translation table.
3. Configure the translation table by writing the necessary translation table entries to the registers defined in the CNFG Store and Forward I/O module.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when TelePACE programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

TelePACE C Tools

The TelePACE C language application program interface provides the following functions. Refer to the ***TelePACE C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED (station 256).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

ISaGRAF IEC61131

1. To enable the use of store and forward messaging on one or more serial ports the Custom Function **setprot** or **setprot2** must be added to the project. The SandFEnabled input must be set to TRUE.
2. Configure the translation table by using the **setsf** function to write the necessary translation table entries.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when ISaGRAF IEC61131 programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

ISaGRAF IEC61131 C Tools

The ISaGRAF C language application program interface provides the following functions. Refer to the ***ISaGRAF C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.

- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

SCADAPack Light Controller

An application program, written in TelePACE Ladder Logic or TelePACE C Tools and ISaGRAF IEC61131 or ISaGRAF IEC61131 C Tools programming, enables and configures store and forward messaging. A HMI host may enable and configure store and forward messaging through the controller I/O database.

TelePACE Ladder Logic

1. To enable the use of store and forward messaging on one or more serial ports the Configuration I/O Module ***CNFG Protocol Settings Method 1, 2 or 3*** must be added to the register assignment. The store and forward enable register must be set to enable.
2. Add the Configuration I/O Module ***CNFG Store and Forward*** to the register assignment to configure the translation table.
3. Configure the translation table by writing the necessary translation table entries to the registers defined in the CNFG Store and Forward I/O module.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when TelePACE programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

TelePACE C Tools

The TelePACE C language application program interface provides the following functions. Refer to the ***TelePACE C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).

- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

ISaGRAF IEC61131

1. To enable the use of store and forward messaging on one or more serial ports the Custom Function **setprot** or **setprot2** must be added to the project. The SandFEnabled input must be set to TRUE.
2. Configure the translation table by using the **setsf** function to write the necessary translation table entries.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when ISaGRAF IEC61131 programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

ISaGRAF IEC61131 C Tools

The ISaGRAF C language application program interface provides the following functions. Refer to the ***ISaGRAF C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

SCADAPack Plus Controller

An application program, written in TelePACE Ladder Logic or TelePACE C Tools and ISaGRAF IEC61131 or ISaGRAF IEC61131 C Tools programming, enables and configures store and forward messaging. A HMI host may enable and configure store and forward messaging through the controller I/O database.

TelePACE Ladder Logic

1. To enable the use of store and forward messaging on one or more serial ports the Configuration I/O Module ***CNFG Protocol Settings Method 1, 2 or 3*** must be added to the register assignment. The store and forward enable register must be set to enable.
2. Add the Configuration I/O Module ***CNFG Store and Forward*** to the register assignment to configure the translation table.
3. Configure the translation table by writing the necessary translation table entries to the registers defined in the CNFG Store and Forward I/O module.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when TelePACE programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

TelePACE C Tools

The TelePACE C language application program interface provides the following functions. Refer to the ***TelePACE C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

ISaGRAF IEC61131

1. To enable the use of store and forward messaging on one or more serial ports the Custom Function **setprot** or **setprot2** must be added to the project. The SandFEnabled input must be set to TRUE.
2. Configure the translation table by using the **setsf** function to write the necessary translation table entries.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when ISaGRAF IEC61131 programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

ISaGRAF IEC61131 C Tools

The ISaGRAF C language application program interface provides the following functions. Refer to the ***ISaGRAF C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).

- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

SCADAPack LP Controller

An application program, written in TelePACE Ladder Logic or TelePACE C Tools and ISaGRAF IEC61131 or ISaGRAF IEC61131 C Tools programming, enables and configures store and forward messaging. A HMI host may enable and configure store and forward messaging through the controller I/O database.

TelePACE Ladder Logic

1. To enable the use of store and forward messaging on one or more serial ports the Configuration I/O Module ***CNFG Protocol Settings Method 1, 2 or 3*** must be added to the register assignment. The store and forward enable register must be set to enable.
2. Add the Configuration I/O Module ***CNFG Store and Forward*** to the register assignment to configure the translation table.
3. Configure the translation table by writing the necessary translation table entries to the registers defined in the CNFG Store and Forward I/O module.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when TelePACE programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

TelePACE C Tools

The TelePACE C language application program interface provides the following functions. Refer to the ***TelePACE C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

ISaGRAF IEC61131

1. To enable the use of store and forward messaging on one or more serial ports the Custom Function **setprot** or **setprot2** must be added to the project. The SandFEnabled input must be set to TRUE.
2. Configure the translation table by using the **setsf** function to write the necessary translation table entries.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when ISaGRAF IEC61131 programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

ISaGRAF IEC61131 C Tools

The ISaGRAF C language application program interface provides the following functions. Refer to the ***ISaGRAF C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

SCADAPack 100 Controller

An application program, written in TelePACE Ladder Logic or TelePACE C Tools and ISaGRAF IEC61131 or ISaGRAF IEC61131 C Tools programming, enables and configures store and forward messaging. A HMI host may enable and configure store and forward messaging through the controller I/O database.

TelePACE Ladder Logic

1. To enable the use of store and forward messaging on one or more serial ports the Configuration I/O Module ***CNFG Protocol Settings Method 1, 2 or 3*** must be added to the register assignment. The store and forward enable register must be set to enable.
2. Add the Configuration I/O Module ***CNFG Store and Forward*** to the register assignment to configure the translation table.
3. Configure the translation table by writing the necessary translation table entries to the registers defined in the CNFG Store and Forward I/O module.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when TelePACE programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

TelePACE C Tools

The TelePACE C language application program interface provides the following functions. Refer to the ***TelePACE C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.

- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

ISaGRAF IEC61131

1. To enable the use of store and forward messaging on one or more serial ports the Custom Function **setprot** or **setprot2** must be added to the project. The SandFEnabled input must be set to TRUE.
2. Configure the translation table by using the **setsf** function to write the necessary translation table entries.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when ISaGRAF IEC61131 programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

ISaGRAF IEC61131 C Tools

The ISaGRAF C language application program interface provides the following functions. Refer to the ***ISaGRAF C Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

SCADAPack 350, SCADAPack 32 and 32P Controller

An application program, written in TelePACE Ladder Logic or TelePACE C++ Tools and ISaGRAF IEC61131 or ISaGRAF IEC61131 C++ Tools programming, enables and configures store and forward messaging. A HMI host may enable and configure store and forward messaging through the controller I/O database.

TelePACE Ladder Logic

When a SCADAPack 350, SCADAPack 32 or SCADAPack 32P controllers are used the store and forward translation table is configured using an **Element Configuration** dialog. From the **Controller** menu select the **Store and Forward** command to access the element configuration. Refer to the TelePACE Ladder Logic Program Reference Manual for complete information on using the Store and Forward element configuration.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when TelePACE programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

TelePACE C++ Tools

The SCADAPack 32 C++ language application program interface provides the following functions. Refer to the ***SCADAPack 32 C++ Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).
- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

ISaGRAF IEC61131

1. To enable the use of store and forward messaging on one or more serial ports the Custom Function **setprot** or **setprot2** must be added to the project. The SandFEnabled input must be set to TRUE.
2. Configure the translation table by using the **setsfip2** function to write the necessary translation table entries.

The translation table must be initialized before store and forward messaging is enabled. Forwarding of messages is disabled when ISaGRAF IEC61131 programming software or a SERVICE boot initializes the controller. This prevents inadvertent forwarding of messages when new controllers are installed on networks.

ISaGRAF IEC61131 C++ Tools

The SCADAPack 32 C++ language application program interface provides the following functions. Refer to the ***SCADAPack 32 C++ Tools Reference and User Manual*** for details.

- The **getSFTranslation** function returns an entry from the store and forward translation table. The entry consists of two port and station address pairs.
- The **setSFTranslation** function writes an entry into the store and forward translation table. The entry consists of two port and station address pairs. The function checks for invalid translations; if the translation is not valid it is not stored. The function returns a

status code indicating success or an error if the translation is not valid. A translation is cleared from the table by writing a translation with both stations set to DISABLED_STATION (65535).

- The **clearSFTranslationTable** function clears all entries in the translation table. A cleared entry has the port set to 0 (com1) and the station set to DISABLED_STATION (65535).
- The **checkSFTranslationTable** function checks the translation table for invalid entries. It returns a status structure indicating if the table is valid and the location and type of the first error if it is not valid.

Diagnostics Counters

The TeleBUS protocol provides diagnostics counters for each serial port. The counters aid in determining the source of communication errors. Store and forward messaging provides the following counters for each communication port. All counters have a maximum count of 65535. Counters roll back to zero on the next event.

- **Stored Message Counter**: the number of messages received, which qualified for forwarding. A message qualifies for forwarding if a valid translation is found for the port and station in the translation table.
- **Forwarded Message Counter**: the number of messages forwarded (transmitted) on this port.

Refer to the user manual for the controller and programming environment you are using for information on the diagnostics counters.

Point–To–Point Protocol (PPP)

SCADAPack 32 and SCADAPack 32P controllers support Point-to-Point Protocol (PPP) on the serial ports. Any serial port may be configured for the PPP protocol. Once a PPP connection is established the serial port has access to all IP protocol servers enabled on the controller.

A serial port configured for PPP supports an auto answer mode when dialed up through a modem. After answering the modem the serial port performs the login steps according to the authentication option selected for the port.

PPP provides two authentication protocols, which automates logins - PAP (Password Authentication Protocol) and CHAP (Challenge-Handshake Authentication Protocol).

PPP settings are configurable for each serial port on the SCADAPack 32 or SCADAPack 32P controller.

An inactivity timeout closes the PPP connection and hangs up the modem when the connection becomes idle. The timeout may also be disabled. Timeout range is 1 to 65535 minutes (~1092 hours maximum).

When the PPP protocol is selected for a serial port, the serial port must be assigned a unique IP address, different from the IP address assigned to Ethernet or any other active PPP connection.

The remote end of a PPP connection may request an IP address from the controller PPP Server. The PPP Server will provide this IP address if requested.

Only one default gateway may be assigned to the controller. A PPP connection may be configured as the gateway.

PPP Client Setup in Windows 2000

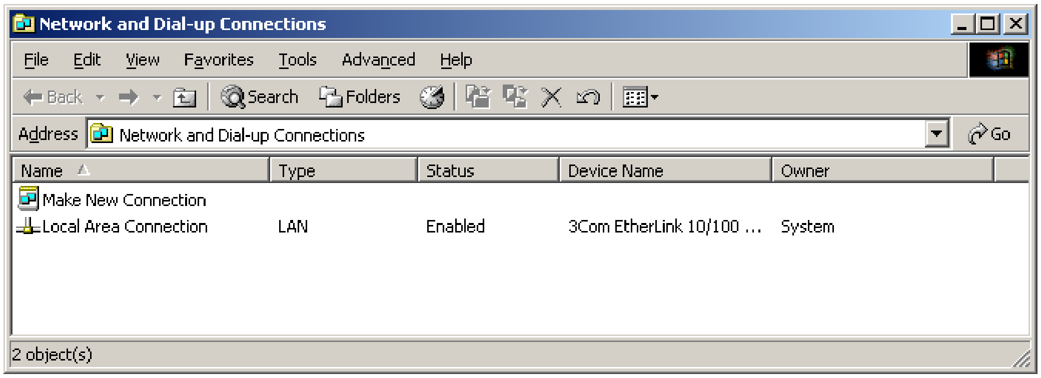
This section describes the procedure for setting up a PPP client from a Windows 2000 PC. Client setup for a dialup PPP connection and a direct serial PPP connection are presented.

Direct Serial PPP Connection using Windows 2000

Connection Setup

Use this connection when an only serial cable is used to establish a PPP connection between a Windows 2000 PC and a SCADAPack 32, without a dialup modem.

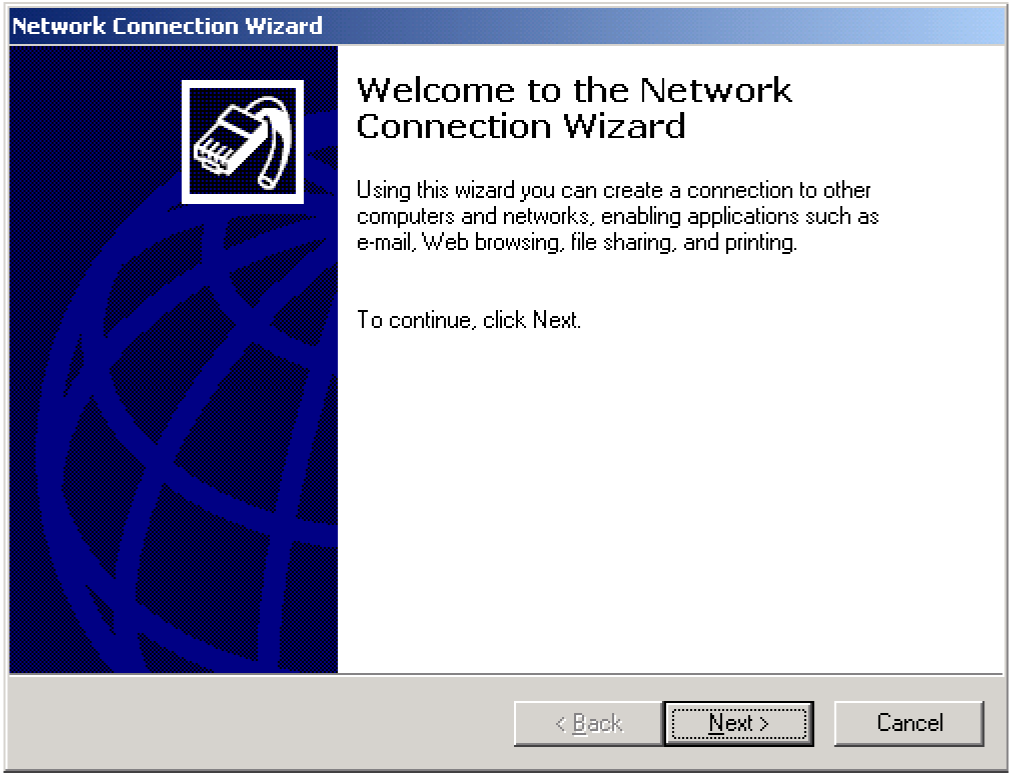
1. From the ***Start*** menu, right click ***Network and Dial-up Connections*** from the ***Settings*** group, and select **Open**. The *Network and Dial-up Connections* dialog is displayed.


Network and Dial-up Connections
File Edit View Favorites Tools Advanced Help
Back Search Folders
Address Network and Dial-up Connections
Go

Name	Type	Status	Device Name	Owner
Make New Connection				
Local Area Connection	LAN	Enabled	3Com EtherLink 10/100 ...	System

2 object(s)

2. Double click the item ***Make New Connection*** from the *Network and Dial-up Connections dialog.* The connection wizard dialog is displayed.

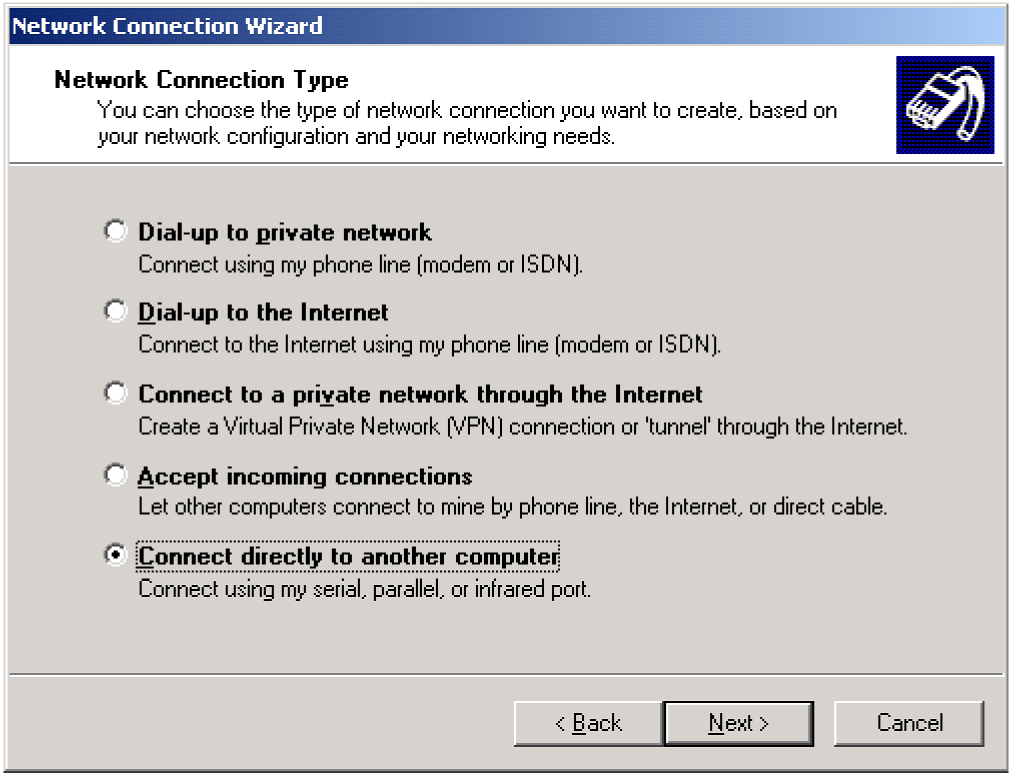


Welcome to the Network Connection Wizard

Using this wizard you can create a connection to other computers and networks, enabling applications such as e-mail, Web browsing, file sharing, and printing.

To continue, click Next.

3. Select the ***Next*** button to display the connection type options dialog.



Network Connection Wizard

Network Connection Type

You can choose the type of network connection you want to create, based on your network configuration and your networking needs.

○ **Dial-up to private network**
Connect using my phone line (modem or ISDN).

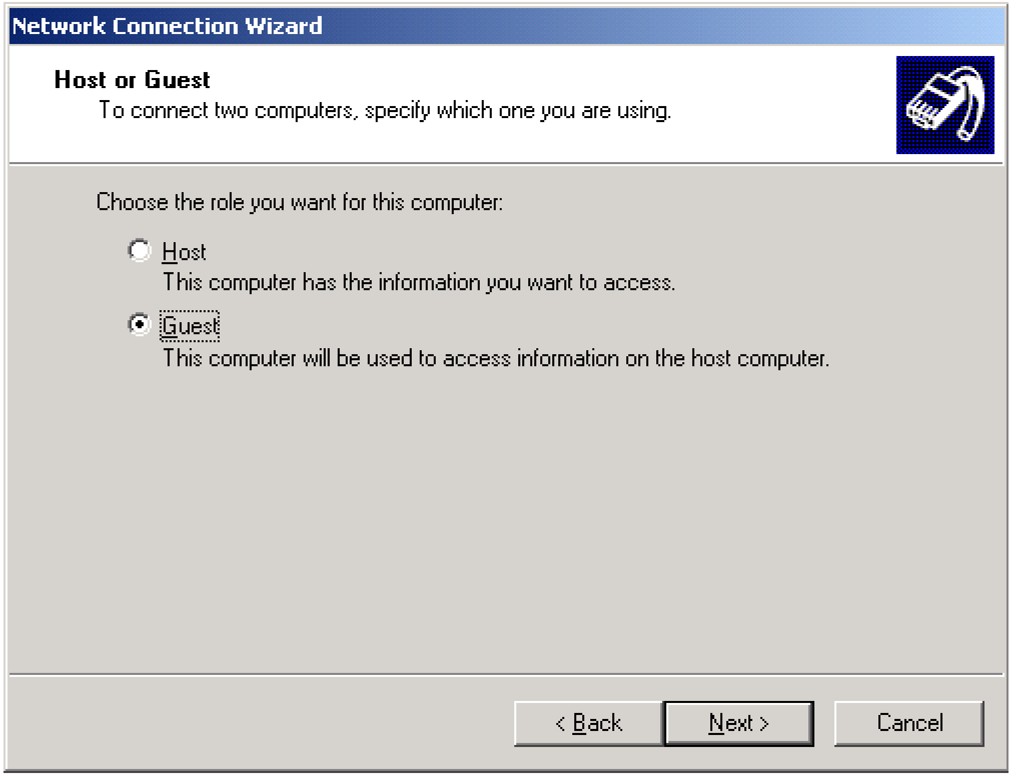
○ **Dial-up to the Internet**
Connect to the Internet using my phone line (modem or ISDN).

○ **Connect to a private network through the Internet**
Create a Virtual Private Network (VPN) connection or 'tunnel' through the Internet.

○ **Accept incoming connections**
Let other computers connect to mine by phone line, the Internet, or direct cable.

● **Connect directly to another computer**
Connect using my serial, parallel, or infrared port.

4. For Network Connection Type select the type ***Connect directly to another computer*** and select the ***Next*** button. The Host or Guest options dialog is displayed.



Network Connection Wizard

Host or Guest

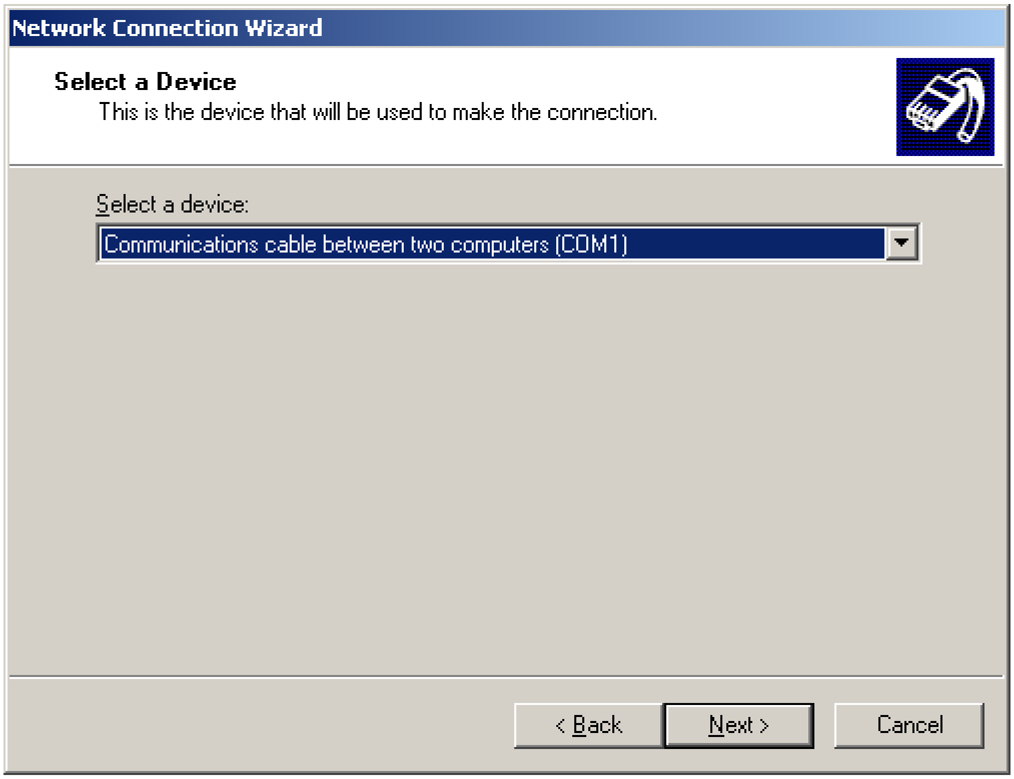
To connect two computers, specify which one you are using.

Choose the role you want for this computer:

○ Host
This computer has the information you want to access.

● Guest
This computer will be used to access information on the host computer.

5. Select the ***Guest*** option and the ***Next*** button. The *Select a Device* dialog is displayed.



Network Connection Wizard

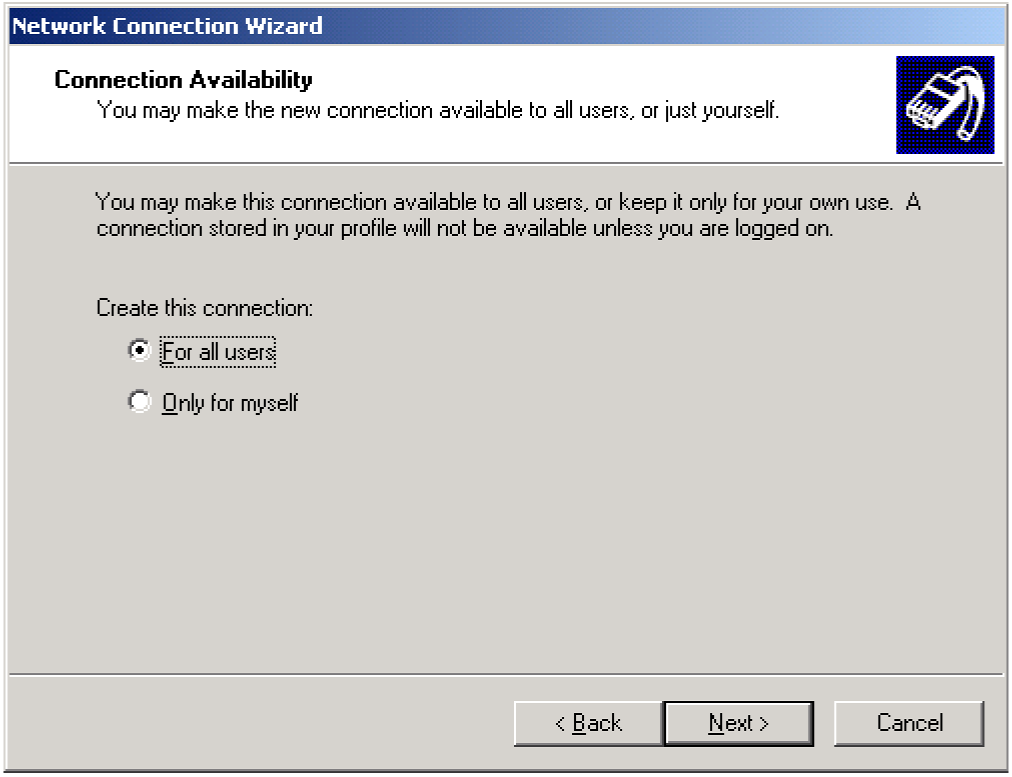
Select a Device

This is the device that will be used to make the connection.

Select a device:

Communications cable between two computers (COM1)

6. From the menu select the serial port on your PC that will be used to connect to the SCADAPack 32. Select the ***Next*** button. The *Connection Availability* dialog is displayed.



Network Connection Wizard

Connection Availability

You may make the new connection available to all users, or just yourself.

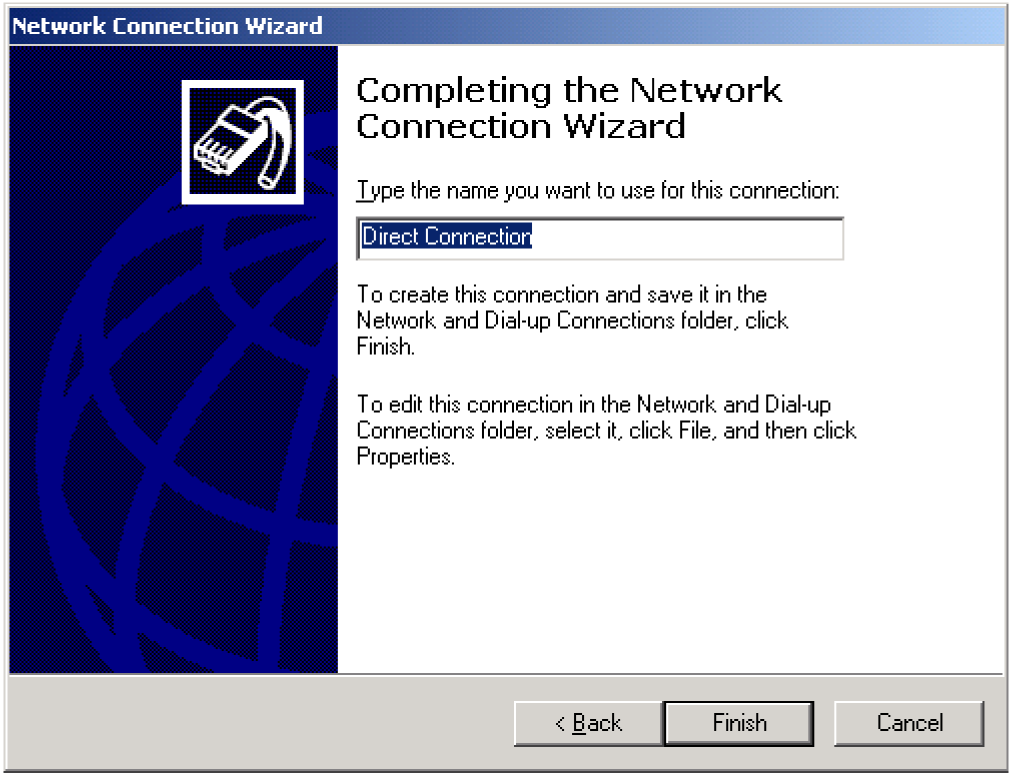
You may make this connection available to all users, or keep it only for your own use. A connection stored in your profile will not be available unless you are logged on.

Create this connection:

● For all users

○ Only for myself

7. Select either option and then select the ***Next*** button. The *Connection Name* dialog is displayed.


Network Connection Wizard

Completing the Network Connection Wizard

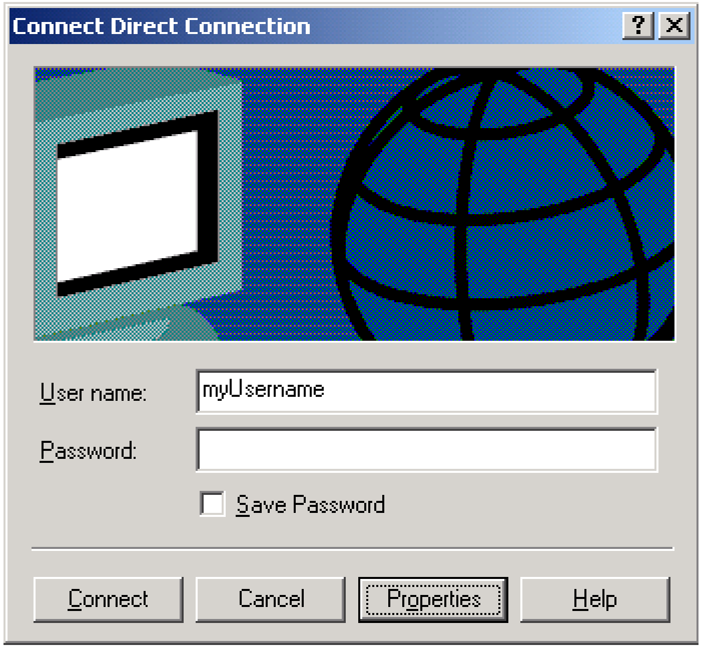
Type the name you want to use for this connection:

Direct Connection

To create this connection and save it in the Network and Dial-up Connections folder, click Finish.

To edit this connection in the Network and Dial-up Connections folder, select it, click File, and then click Properties.

8. Enter a name for the connection and select the ***Finish*** button. The username and password prompt is displayed.

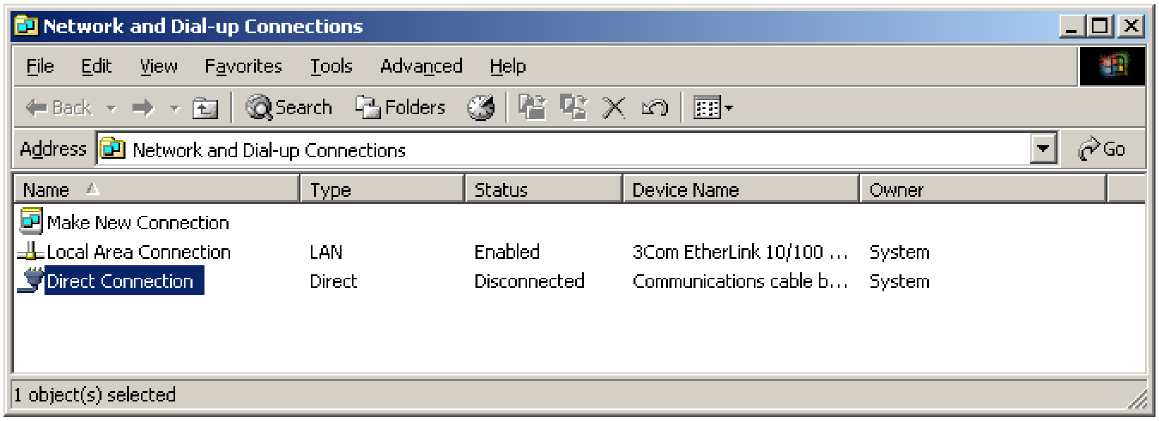

Connect Direct Connection

User name: myUsername

Password:

□ Save Password

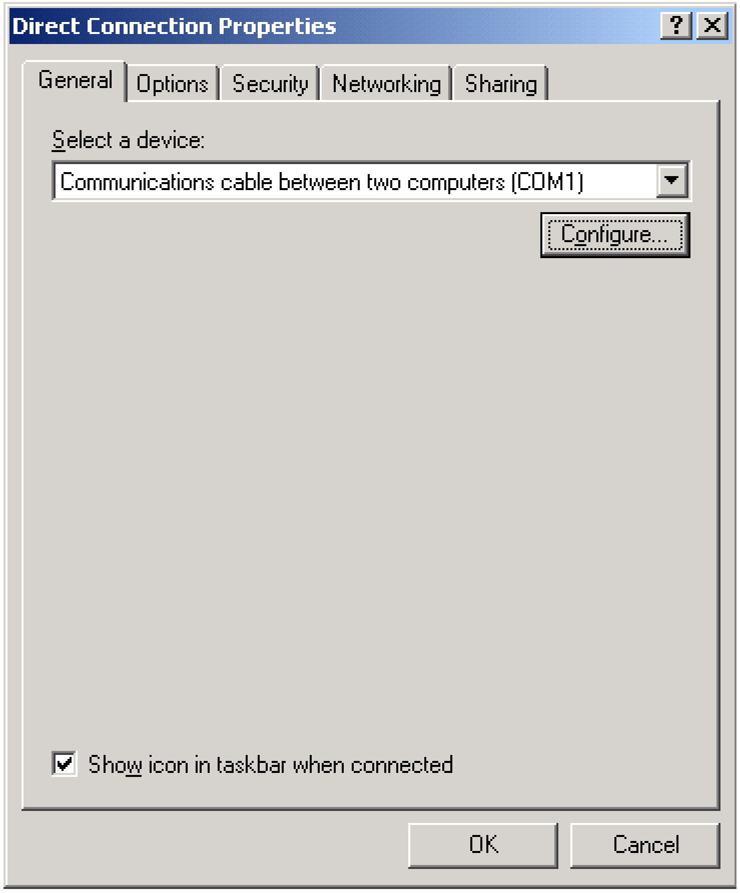
9. Select the ***Cancel*** button. The *Network and Dial-up Connections* dialog should be visible again.


Network and Dial-up Connections
File Edit View Favorites Tools Advanced Help
Back Search Folders
Address Network and Dial-up Connections
Go

Name	Type	Status	Device Name	Owner
Make New Connection				
Local Area Connection	LAN	Enabled	3Com EtherLink 10/100 ...	System
Direct Connection	Direct	Disconnected	Communications cable b...	System

1 object(s) selected

10. Right click your new *Direct Connection* icon from the *Network and Dial-up Connections* dialog and select ***Properties*** from the list. The *Properties* dialog is displayed.

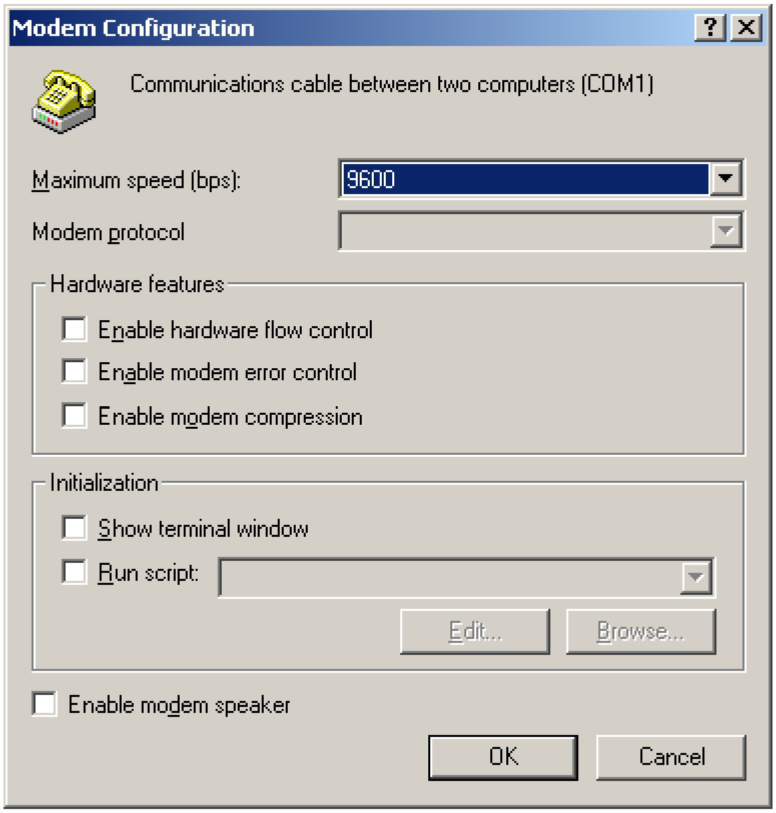


Select a device:

Communications cable between two computers (COM1)

☑ Show icon in taskbar when connected

11. Select the ***Configure*** button from the *General* page. The *Modem Configuration* dialog is displayed.



Modem Configuration

Communications cable between two computers (COM1)

Maximum speed (bps): 9600

Modem protocol

Hardware features

□ Enable hardware flow control

□ Enable modem error control

□ Enable modem compression

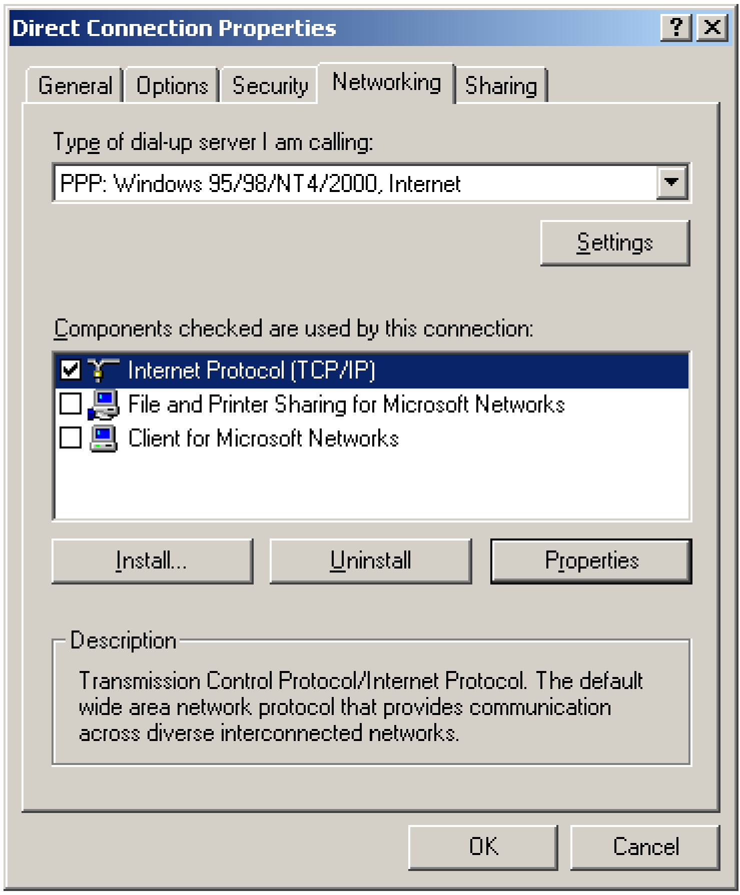
Initialization

□ Show terminal window

□ Run script:

□ Enable modem speaker

12. There is no modem in this direct serial connection so uncheck all items including *hardware flow control.* Select the baud rate you intend to use (e.g. 9600 bps). Select ***OK*** to return to the *Properties* dialog.
13. From the *Properties* dialog select the ***Networking*** page.



Type of dial-up server I am calling:

PPP: Windows 95/98/NT4/2000, Internet

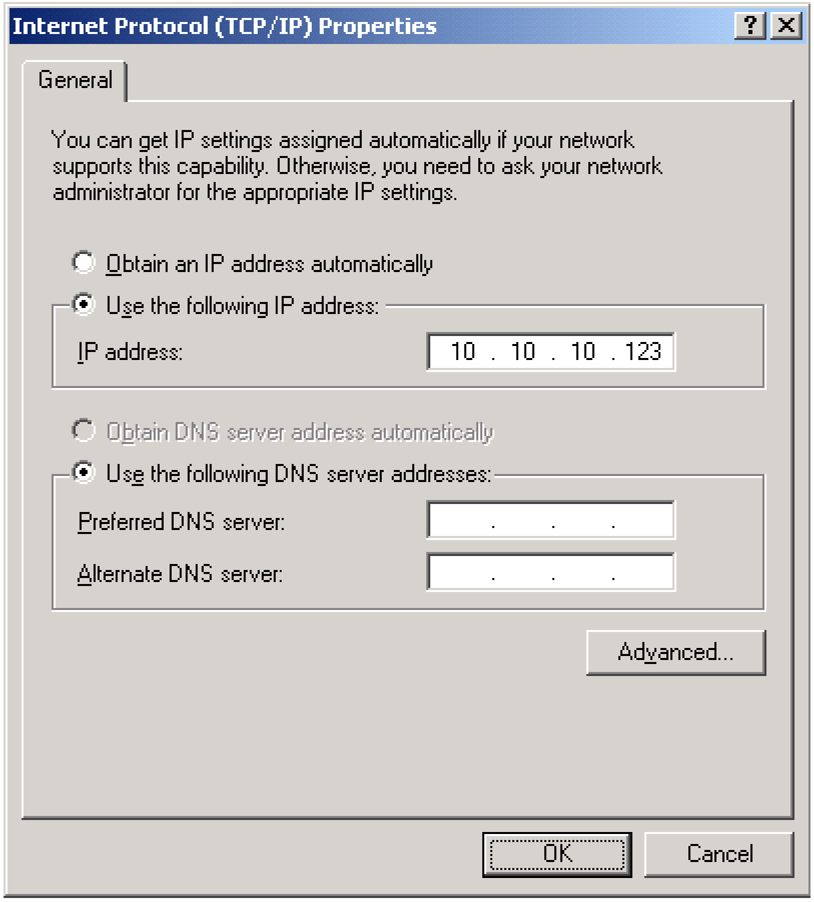
Components checked are used by this connection:

☑ Internet Protocol (TCP/IP)
□ File and Printer Sharing for Microsoft Networks
□ Client for Microsoft Networks

Description

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

Uncheck all components except the component ***Internet Protocol (TCP/IP***). Select the component ***Internet Protocol (TCP/IP)*** and select the ***Properties*** button. The *Internet Protocol (TCP/IP) Properties* dialog is displayed.



You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 10 . 10 . 10 . 123

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

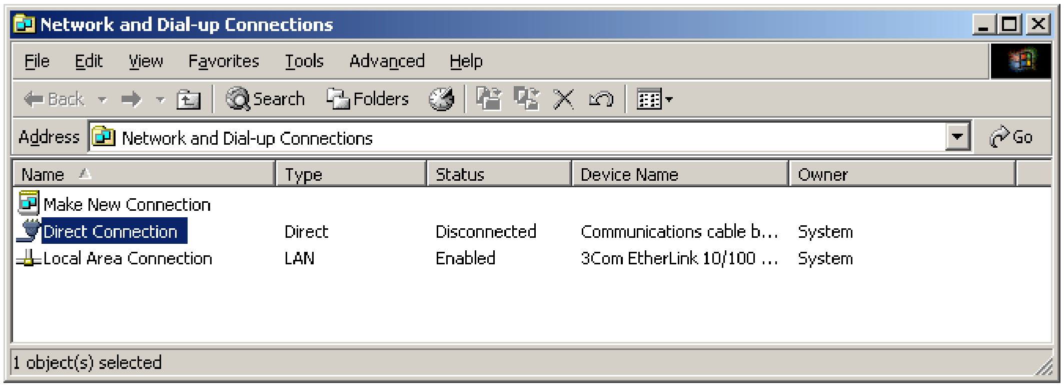
14. The SCADAPack 32 does not have a DHCP server to automatically provide an IP address. Instead the PC's serial port must be given a fixed IP address to use for PPP connections. Select the option ***Use the following IP address.*** Enter an IP address to assign to your PC's serial port. Obtain this IP address from your Network Administrator. Then select ***OK*** to return to the *Properties* dialog.

15. Select ***OK*** again to close the dialog.

Making a PPP Connection to the SCADAPack 32

A connection can only be made after successfully setting up a Direct Connection icon as described in the section *Connection Setup* above. Also, a serial port on the SCADAPack 32 must already be configured for the PPP protocol using the *Controller IP Configuration* dialog and must be downloaded to the SCADAPack 32.

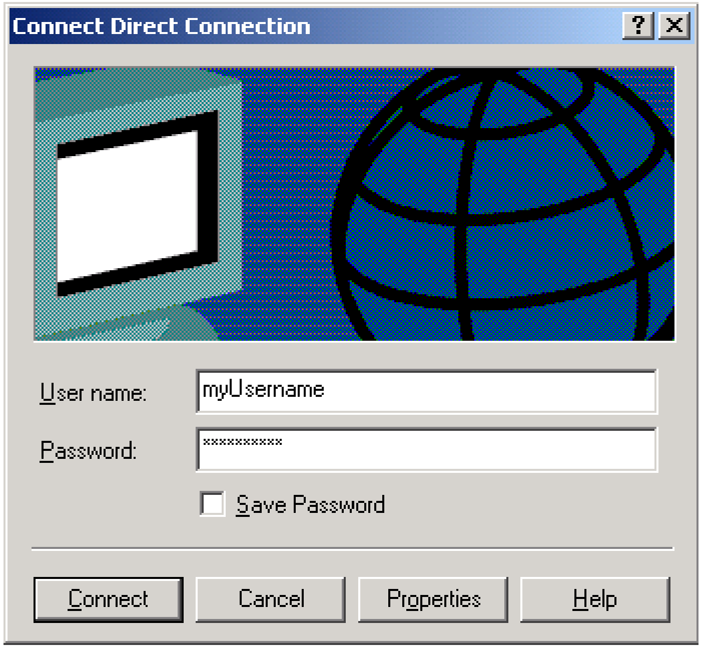
1. From the ***Start*** menu, double click ***Network and Dial-up Connections*** from the ***Settings*** group. The *Network and Dial-up Connections* dialog is displayed.


Network and Dial-up Connections
File Edit View Favorites Tools Advanced Help
Back Search Folders
Address Network and Dial-up Connections
Go

Name	Type	Status	Device Name	Owner
Make New Connection				
Direct Connection	Direct	Disconnected	Communications cable b...	System
Local Area Connection	LAN	Enabled	3Com EtherLink 10/100 ...	System

1 object(s) selected

2. Right click your *Direct Connection* icon that was setup in the previous section and select ***Connect*** from the list. A prompt for username and password is displayed.



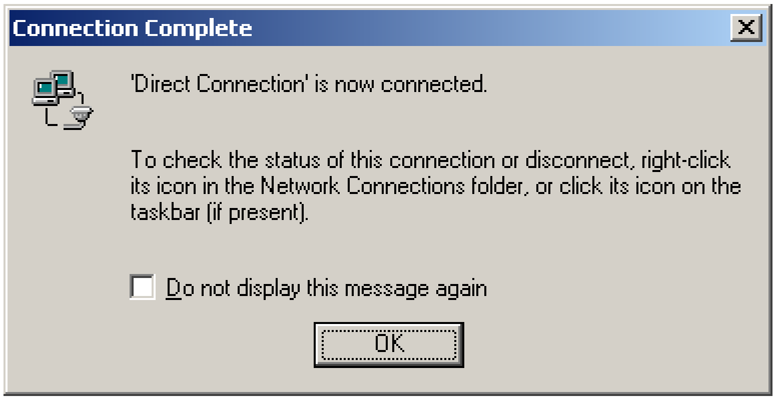
User name: myUsername

Password: ***********

□ Save Password

3. Enter a valid PAP or CHAP username and password. Valid usernames and passwords are configured on the *PPP Login* page of the *Controller IP Configuration* dialog and must be downloaded to the SCADAPack 32. Then select the ***Connect*** button. If neither PAP nor CHAP is being used, ignore the prompt and just select the ***Connect*** button.

4. A progress message is displayed. If the connection is successful the following message is displayed.

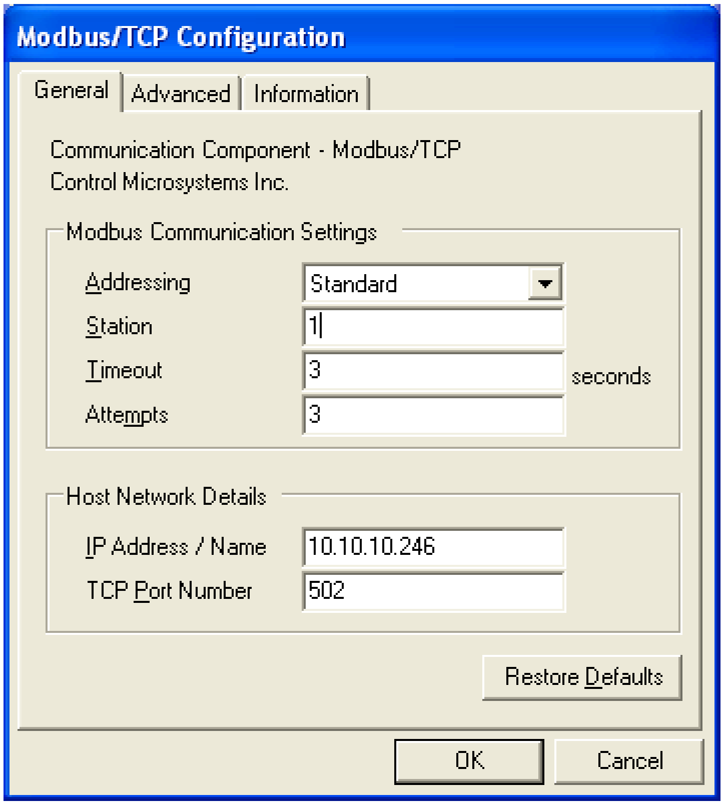

Connection Complete

'Direct Connection' is now connected.

To check the status of this connection or disconnect, right-click its icon in the Network Connections folder, or click its icon on the taskbar (if present).

□ Do not display this message again

5. You may now connect to the IP address assigned to SCADAPack 32 PPP serial port using an appropriate application and a supported protocol (e.g. Modbus/TCP). In the example below, **Firmware Loader** is used to connect over PPP to the SCADAPack 32. From the *PC Communication Settings* dialog, the IP address assigned to the SCADAPack 32 PPP serial port is selected as the ***Connect to Host***.


Modbus/TCP Configuration

Communication Component - Modbus/TCP
Control Microsystems Inc.

Modbus Communication Settings

Addressing: Standard
Station: 1
Timeout: 3 seconds
Attempts: 3

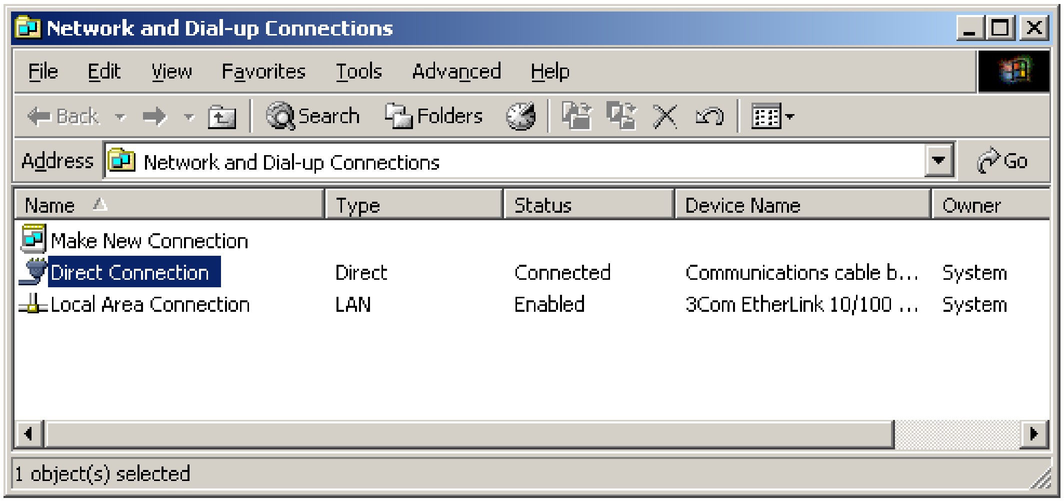
Host Network Details

IP Address / Name: 10.10.10.246
TCP Port Number: 502

Disconnecting a PPP Connection

To disconnect a PPP connection made using the Windows PPP Client, do the following:

1. From the ***Start*** menu, double click ***Network and Dial-up Connections*** from the ***Settings*** group. The *Network and Dial-up Connections* dialog is displayed.


Network and Dial-up Connections
File Edit View Favorites Tools Advanced Help
Back
Search
Folders
Address
Network and Dial-up Connections
Go

Name	Type	Status	Device Name	Owner
Make New Connection				
Direct Connection	Direct	Connected	Communications cable b...	System
Local Area Connection	LAN	Enabled	3Com EtherLink 10/100 ...	System

1 object(s) selected

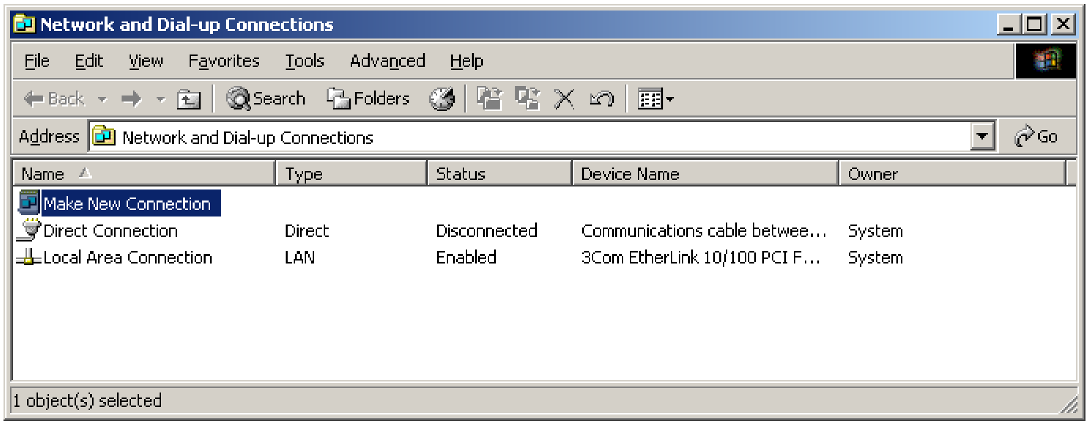
2. Your *Direct Connection* icon should display the word *Connected* in the Status column. To disconnect, right click your *Direct Connection* icon and select ***Disconnect*** from the list.

Dial-up PPP Connection using Windows 2000

Connection Setup using Windows 2000

Use this connection when a dial-up modem is used to establish a PPP connection between a Windows 2000 PC and a SCADAPack 32.

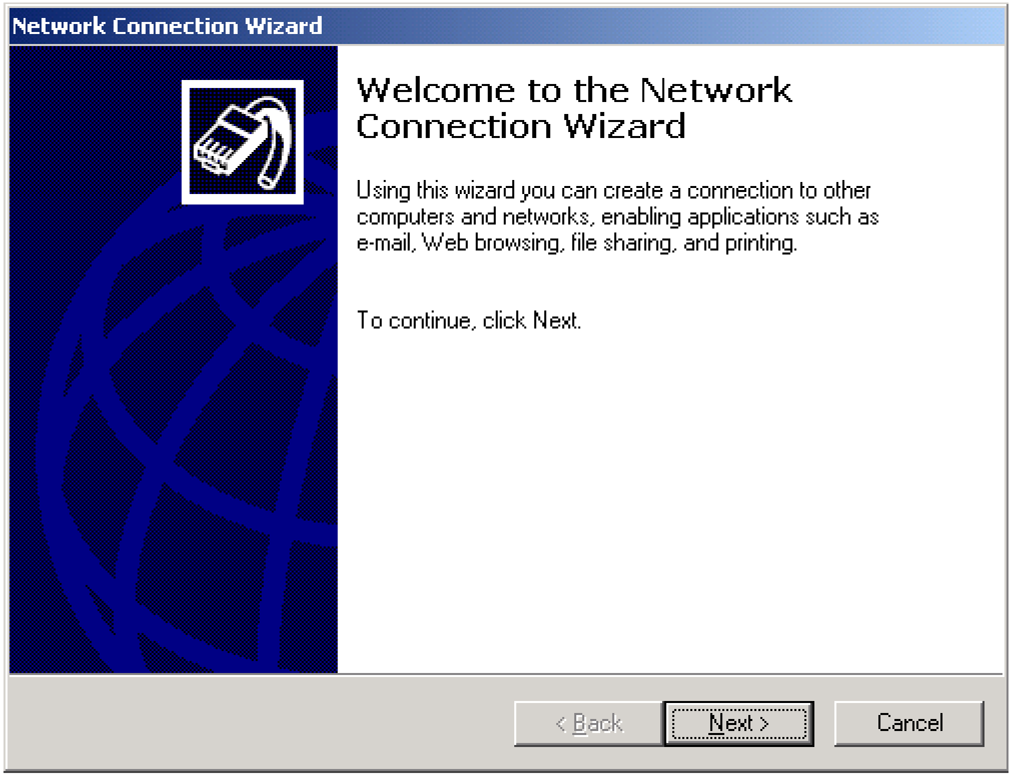
1. From the ***Start*** menu, right click ***Network and Dial-up Connections*** from the ***Settings*** group, and select **Open**. The *Network and Dial-up Connections* dialog is displayed.


Network and Dial-up Connections
File Edit View Favorites Tools Advanced Help
Back
Search
Folders
Address
Network and Dial-up Connections
Go

Name	Type	Status	Device Name	Owner
Make New Connection				
Direct Connection	Direct	Disconnected	Communications cable betwee...	System
Local Area Connection	LAN	Enabled	3Com EtherLink 10/100 PCI F...	System

1 object(s) selected

2. Double click the item ***Make New Connection*** from the *Network and Dial-up Connections dialog.* The connection wizard dialog is displayed.



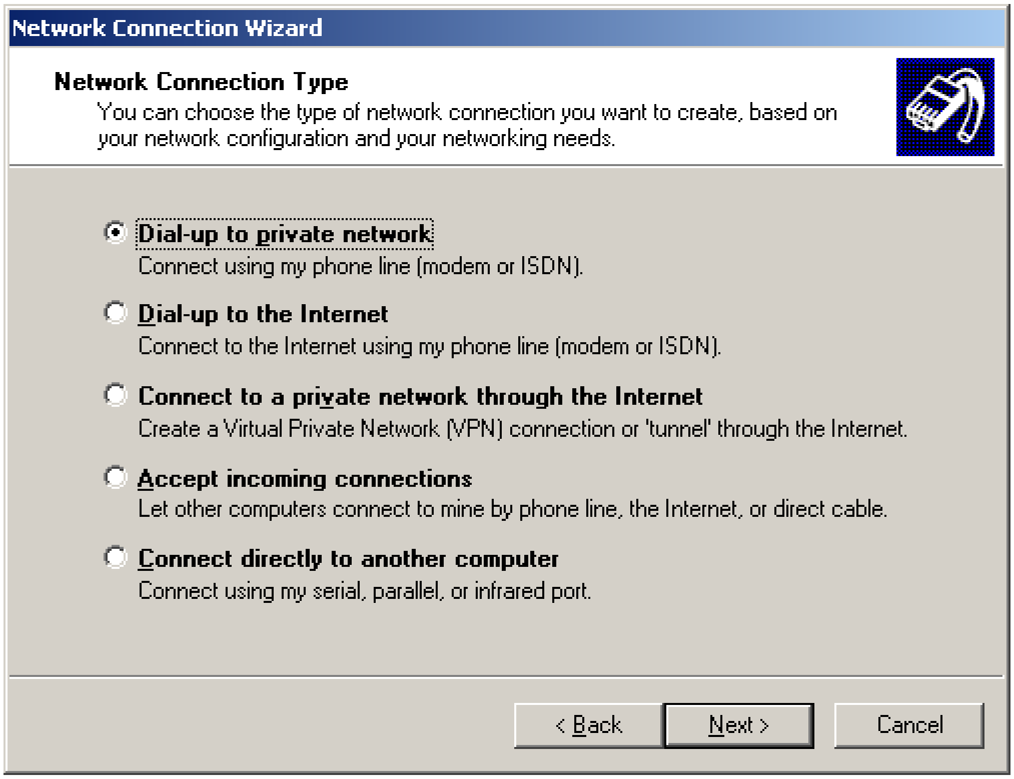
Network Connection Wizard

Welcome to the Network Connection Wizard

Using this wizard you can create a connection to other computers and networks, enabling applications such as e-mail, Web browsing, file sharing, and printing.

To continue, click Next.

3. Select the ***Next*** button to display the connection type options dialog.



Network Connection Wizard

Network Connection Type

You can choose the type of network connection you want to create, based on your network configuration and your networking needs.

● **Dial-up to private network**
Connect using my phone line (modem or ISDN).

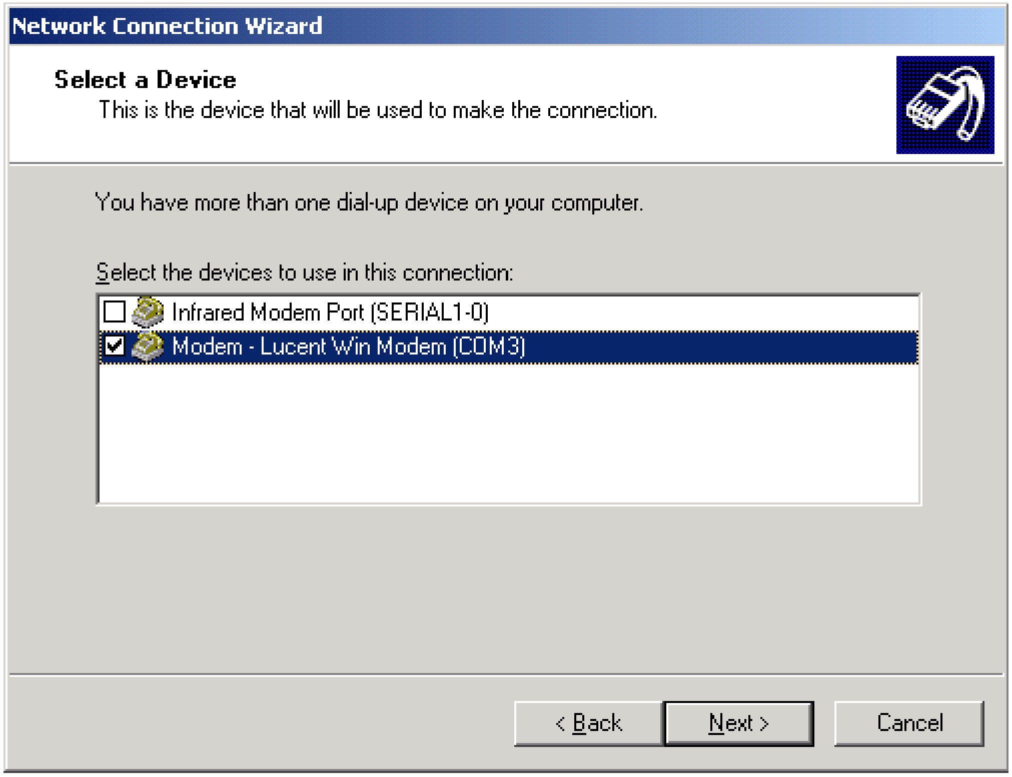
○ **Dial-up to the Internet**
Connect to the Internet using my phone line (modem or ISDN).

○ **Connect to a private network through the Internet**
Create a Virtual Private Network (VPN) connection or 'tunnel' through the Internet.

○ **Accept incoming connections**
Let other computers connect to mine by phone line, the Internet, or direct cable.

○ **Connect directly to another computer**
Connect using my serial, parallel, or infrared port.

4. For Network Connection Type select the type ***Dial-up to private network*** and select the ***Next*** button. If there is more than one modem installed on the PC, the *Select a Device* dialog is displayed. If not, proceed to the next step.



Network Connection Wizard

Select a Device

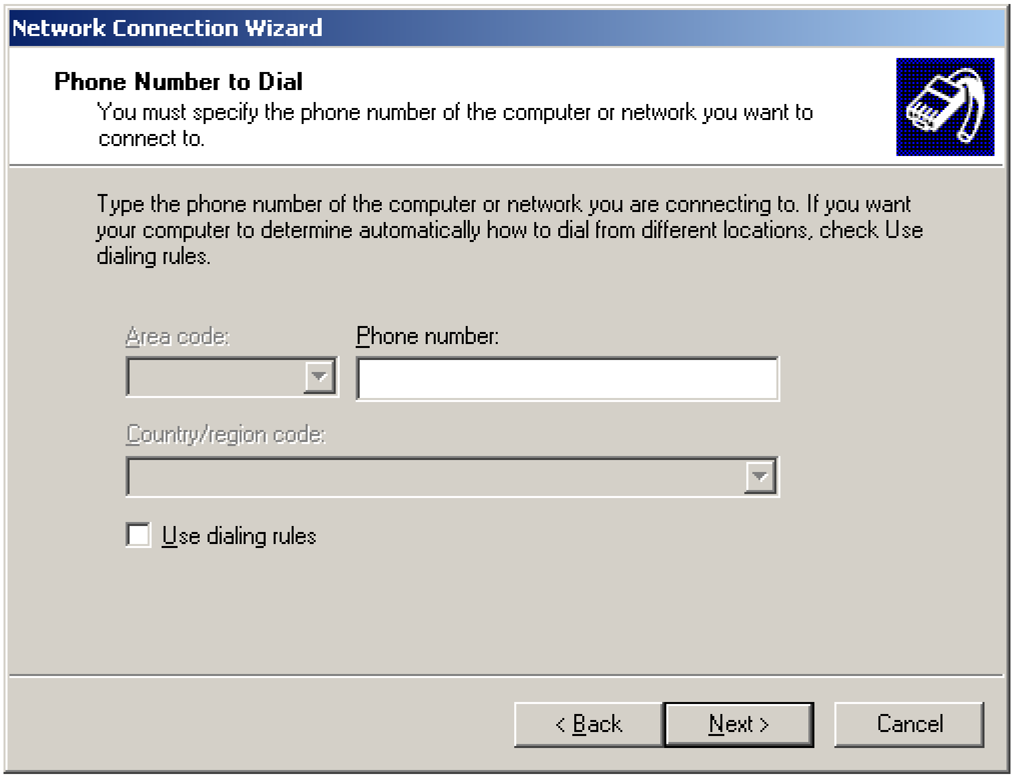
This is the device that will be used to make the connection.

You have more than one dial-up device on your computer.

Select the devices to use in this connection:

- □ Infrared Modem Port (SERIAL1-0)
- ☑ Modem - Lucent Win Modem (COM3)

5. From the menu select the modem installed on your PC that will be used to connect to the SCADAPack 32. Select the ***Next*** button. The *Phone Number to Dial* dialog is displayed.



Network Connection Wizard

Phone Number to Dial

You must specify the phone number of the computer or network you want to connect to.

Type the phone number of the computer or network you are connecting to. If you want your computer to determine automatically how to dial from different locations, check Use dialing rules.

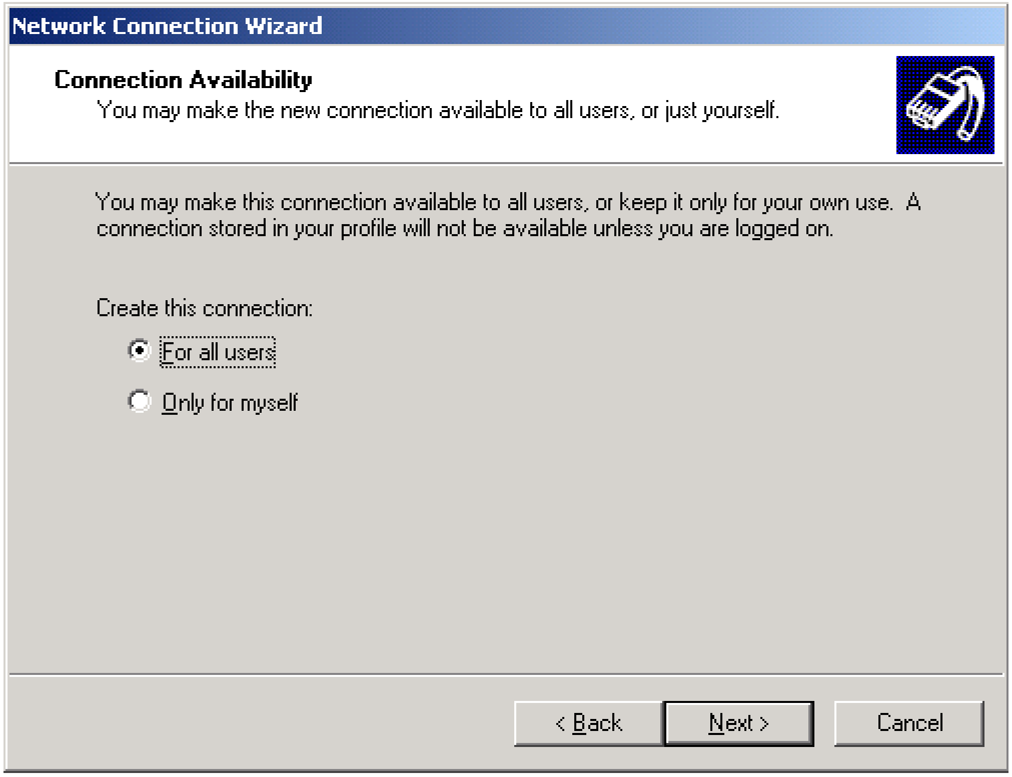
Area code:

Phone number:

Country/region code:

□ Use dialing rules

6. Enter the phone number to dial (this can be changed later) and select the ***Next*** button. The *Connection Availability* dialog is displayed.



Network Connection Wizard

Connection Availability

You may make the new connection available to all users, or just yourself.

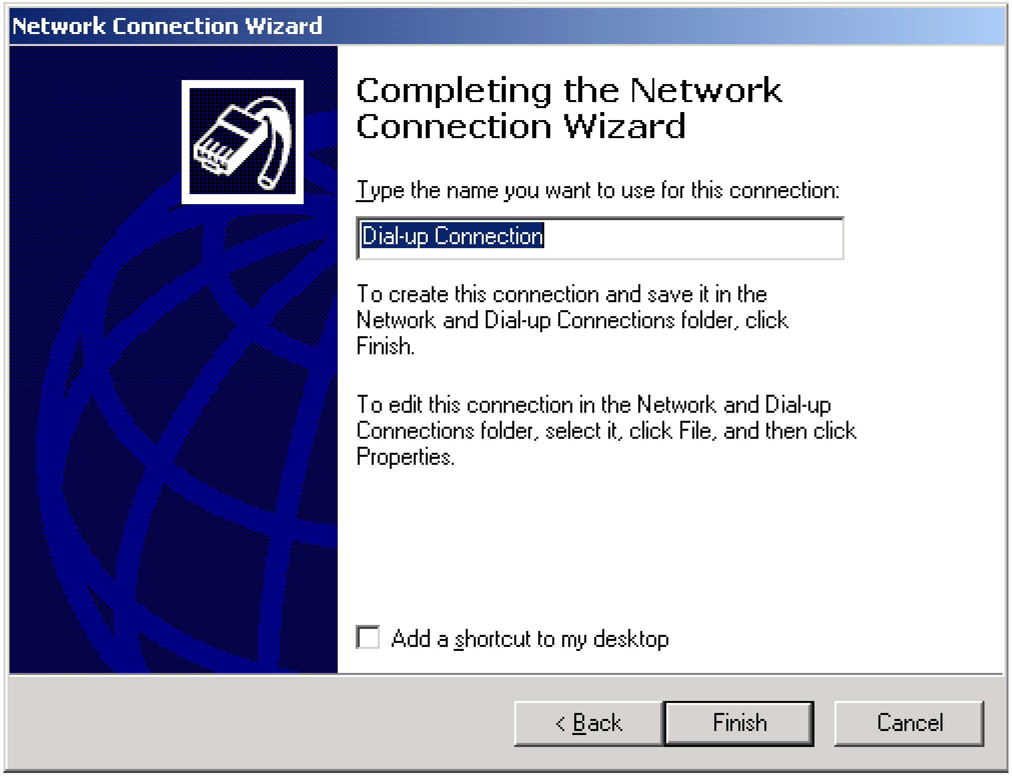
You may make this connection available to all users, or keep it only for your own use. A connection stored in your profile will not be available unless you are logged on.

Create this connection:

● For all users

○ Only for myself

7. Select either option and then select the ***Next*** button. The *Connection Name* dialog is displayed.



Network Connection Wizard

Completing the Network Connection Wizard

Type the name you want to use for this connection:

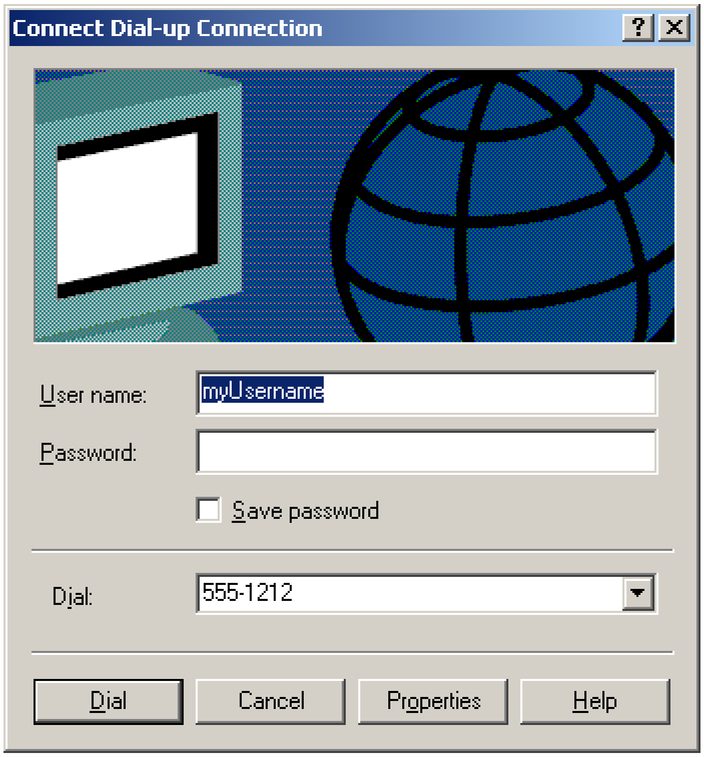
Dial-up Connection

To create this connection and save it in the Network and Dial-up Connections folder, click Finish.

To edit this connection in the Network and Dial-up Connections folder, select it, click File, and then click Properties.

□ Add a shortcut to my desktop

8. Enter a name for the connection and select the ***Finish*** button. The username and password prompt is displayed.



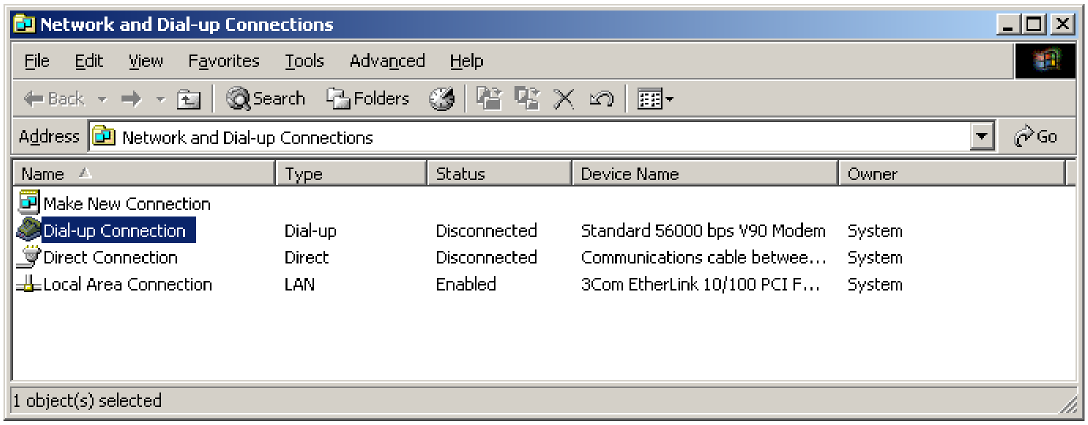
User name: myUsername

Password:

□ Save password

Dial: 555-1212

9. Select the ***Cancel*** button. The *Network and Dial-up Connections* dialog should be visible again.

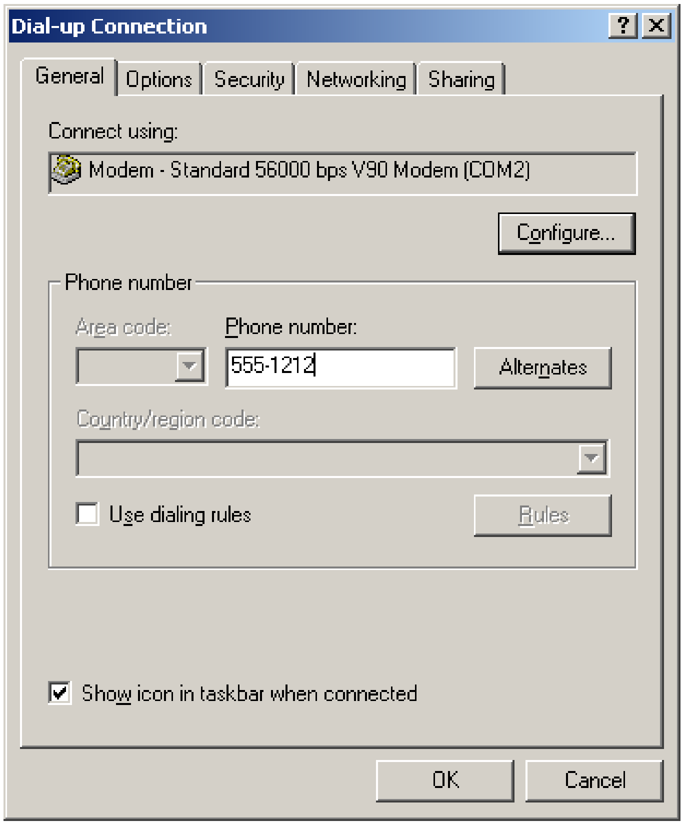


Address: Network and Dial-up Connections

Name	Type	Status	Device Name	Owner
Make New Connection				
Dial-up Connection	Dial-up	Disconnected	Standard 56000 bps V90 Modem	System
Direct Connection	Direct	Disconnected	Communications cable betwee...	System
Local Area Connection	LAN	Enabled	3Com EtherLink 10/100 PCI F...	System

1 object(s) selected

10. Right click your new *Dial-up Connection* icon from the *Network and Dial-up Connections* dialog and select ***Properties*** from the list. The *Properties* dialog is displayed.



Connect using:

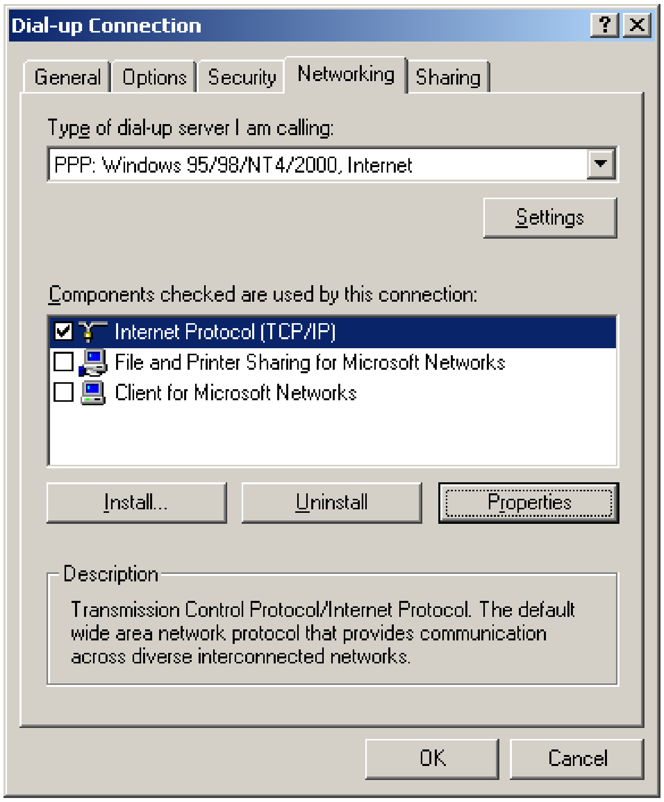
Modem - Standard 56000 bps V90 Modem (COM2)

Phone number: 555-1212

☐ Use dialing rules

☑ Show icon in taskbar when connected

11. From the *Properties* dialog select the ***Networking*** page.



Type of dial-up server I am calling:

PPP: Windows 95/98/NT4/2000, Internet

Components checked are used by this connection:

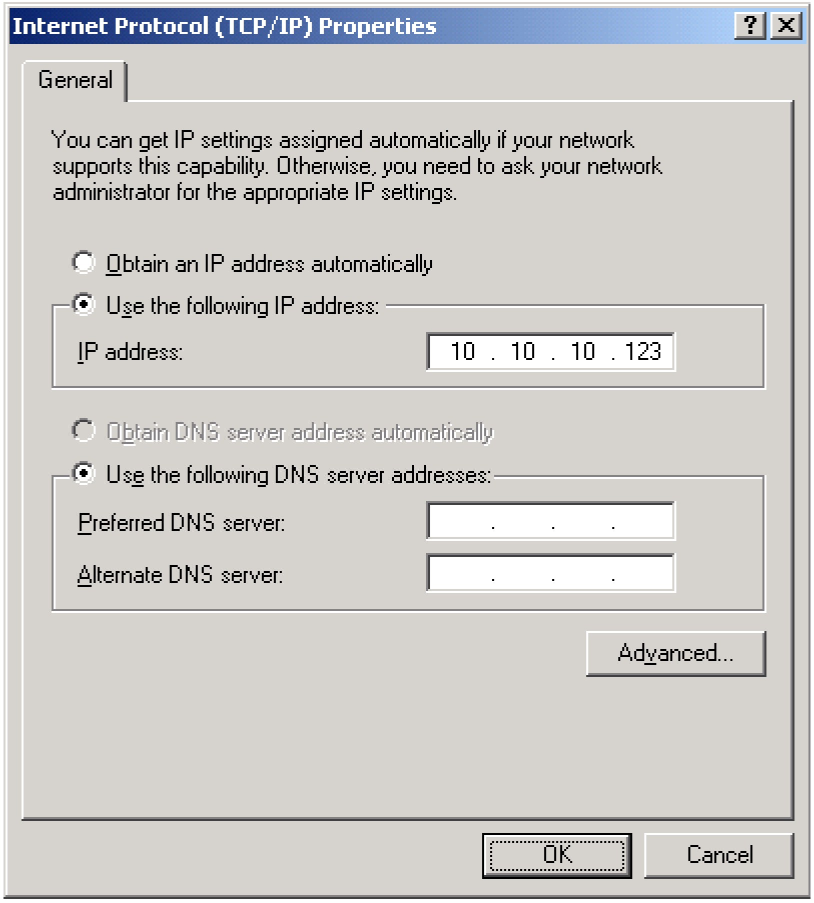
☑ Internet Protocol (TCP/IP)

☐ File and Printer Sharing for Microsoft Networks

☐ Client for Microsoft Networks

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

12. Uncheck all components except the component ***Internet Protocol (TCP/IP***). Select the component ***Internet Protocol (TCP/IP)*** and select the ***Properties*** button. The *Internet Protocol (TCP/IP) Properties* dialog is displayed.



Internet Protocol (TCP/IP) Properties

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 10 . 10 . 10 . 123

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

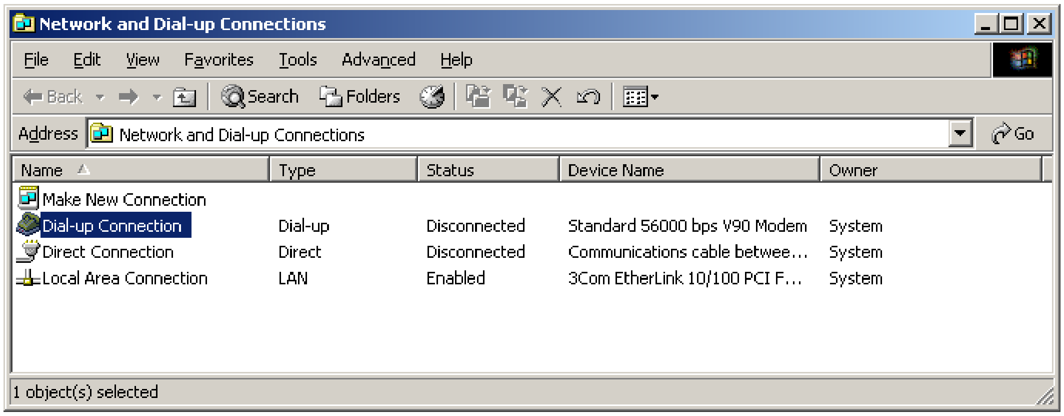
13. The SCADAPack 32 does not have a DHCP server to automatically provide an IP address. Instead the PC’s serial port must be given a fixed IP address to use for PPP connections. Select the option ***Use the following IP address.*** Enter an IP address to assign to your PC’s serial port. Obtain this IP address from your Network Administrator. Then select ***OK*** to return to the *Properties* dialog.

14. Select ***OK*** again to close the dialog.

Making a PPP Dial-up Connection to the SCADAPack 32 using Windows 2000

A connection can only be made after successfully setting up a Dial-up Connection icon as described in the section *Connection Setup* above. Also, a serial port on the SCADAPack 32 must already be configured for the PPP protocol using the *Controller IP Configuration* dialog and must be downloaded to the SCADAPack 32.

1. From the ***Start*** menu, right click ***Network and Dial-up Connections*** from the ***Settings*** group, and select **Open**. The *Network and Dial-up Connections* dialog is displayed.

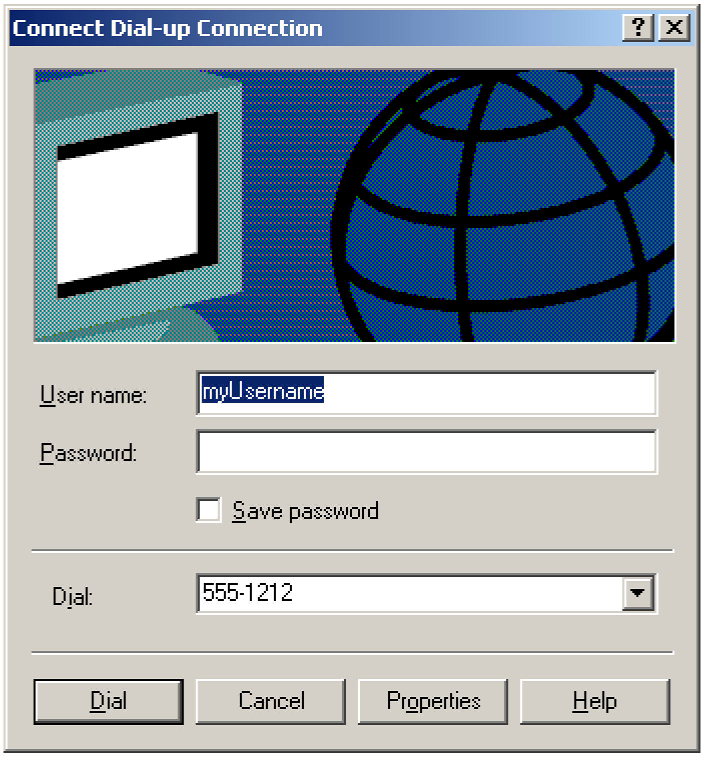


Network and Dial-up Connections
File Edit View Favorites Tools Advanced Help
Back Search Folders
Address Network and Dial-up Connections
Go

Name	Type	Status	Device Name	Owner
Make New Connection				
Dial-up Connection	Dial-up	Disconnected	Standard 56000 bps V90 Modem	System
Direct Connection	Direct	Disconnected	Communications cable betwee…	System
Local Area Connection	LAN	Enabled	3Com EtherLink 10/100 PCI F…	System

1 object(s) selected

2. Right click your *Dial-up Connection* icon that was setup in the previous section and select ***Connect*** from the list. A prompt for username and password is displayed.



User name: myUsername

Password:

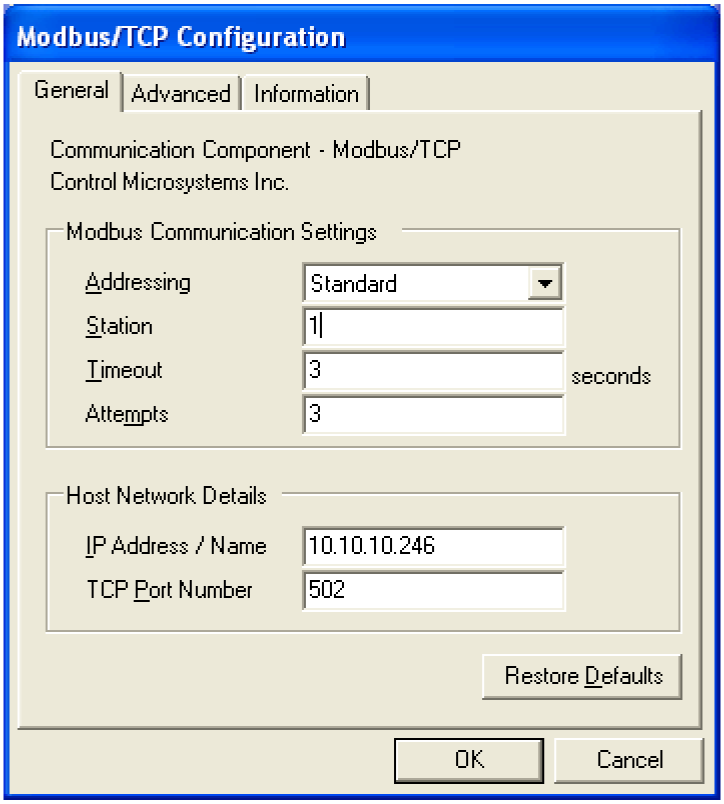
□ Save password

Dial: 555-1212

3. Enter a valid PAP or CHAP username and password. Valid usernames and passwords are configured on the *PPP Login* page of the *Controller IP Configuration* dialog and must be downloaded to the SCADAPack 32. Then select the ***Dial*** button. If neither PAP nor CHAP is being used, ignore the prompt and just select the ***Dial*** button.

4. A progress message is displayed. If the connection is successful your *Dial-up Connection* icon should display the word *Connected* in the Status column.

5. You may now connect to the IP address assigned to SCADAPack 32 PPP serial port using an appropriate application and a supported protocol (e.g. Modbus/TCP). In the example below, **Firmware Loader** is used to connect over PPP to the SCADAPack 32. From the *PC Communication Setting*s dialog, the IP address assigned to the SCADAPack 32 PPP serial port is selected as the ***Connect to Host***.



Communication Component - Modbus/TCP
Control Microsystems Inc.

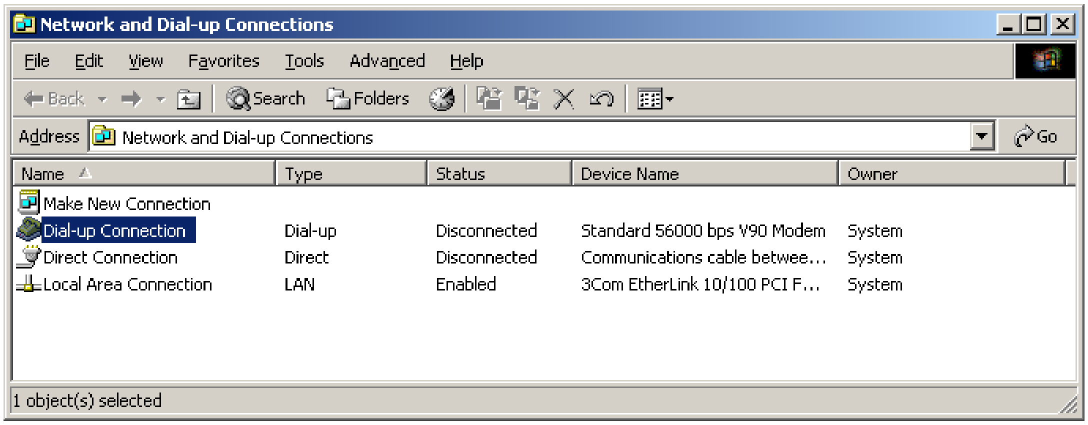
Addressing: Standard
Station: 1
Timeout: 3 seconds
Attempts: 3

IP Address / Name: 10.10.10.246
TCP Port Number: 502

Disconnecting a PPP Connection using Windows 2000

To disconnect a PPP connection made using the Windows PPP Client, do the following:

1. From the ***Start*** menu, right click ***Network and Dial-up Connections*** from the ***Settings*** group, and select **Open**. The *Network and Dial-up Connections* dialog is displayed.



Name	Type	Status	Device Name	Owner
Make New Connection				
Dial-up Connection	Dial-up	Disconnected	Standard 56000 bps V90 Modem	System
Direct Connection	Direct	Disconnected	Communications cable betwee…	System
Local Area Connection	LAN	Enabled	3Com EtherLink 10/100 PCI F…	System

1 object(s) selected

2. Your *Dial-up Connection* icon should display the word *Connected* in the Status column. To disconnect, right click your *Dial-up Connection* icon and select ***Disconnect*** from the list.