Ether-Serial Link™ Advanced Features

This document describes the advanced configuration features of the Ether-Serial Link™. It covers:

1. Ether-Serial Link™ port modes
2. Driver Mode (Default mode)
3. Raw Server Mode
4. Raw Client Mode
5. RFC 2217 Mode
6. Data Connect Mode
7. Ethernet Modem Mode
8. RAS Server Mode
9. Disabled Mode
10. WAN Port configuration
11. Port reset
12. Device reboot
13. Subnet configuration with the Ether-Serial Link
14. Re-establishing a TCP connection with the Ether-Serial Link
15. Configuring the Ether-Serial Link using Telnet
16. Jumpering Pin #10 on the RJ-45 serial port connector

1. Ether-Serial Link™ port modes

The Ether-Serial Link™ converts serial port data to and from TCP/IP packets over Ethernet. Each Ether-Serial Link™ port exchanges TCP/IP packets on its own unique TCP port (called the "Local Port" in the Properties dialog box). Each serial port on an Ether-Serial Link™ device can be configured independently to perform this conversion in one of eight modes:

2. Driver Mode (Default mode)

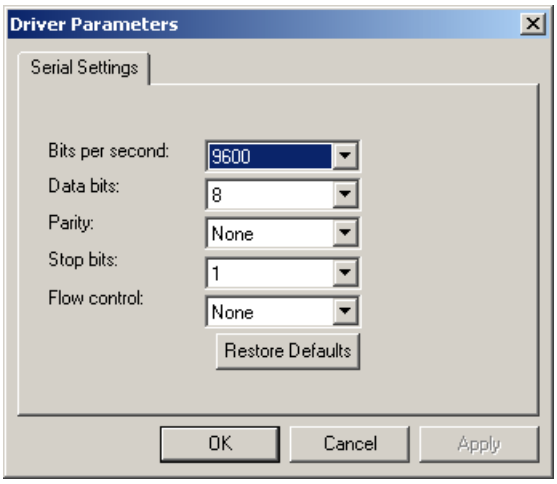
This mode enables the physical port of an Ether-Serial Link™ to communicate directly with the serial port API of a supported operating system as a local COM port. In this mode, authorized users can set bits per second, data bits, parity, stop bits, and flow control port-by-port, or the port will accept its serial port settings from the application/operating system that is accessing it. Driver mode is the factory default mode when an Ether-Serial Link™ is installed.

When installed on a LAN, an Ether-Serial Link™ in Driver Mode has all the functionality of an internally installed serial port.

When installed on a WAN or the Internet, the serial port of an Ether-Serial Link™ gains enormous flexibility and power. To use an Ether-Serial Link™ port on a remote network:

1. configure the IP address and TCP port of the WAN connection in the *Ether-Link Manager*™ to match the gateway IP address of the remote network,
2. map the IP address and TCP port of the remote gateway to the IP address of the Ether-Serial Link™, and
3. configure the Ether-Serial Link™ itself for Driver Mode.

*Applications***:** General serial port access from applications running on a PC.



Driver Mode operation with Host PC security and firewalls:

When first installing *Ether-Link Manager*™ and driver onto the host PC, the security and firewall settings may have to be adjusted or turned off to allow the application and driver to use COM port and network resources in order to load the software.

The *Ether-Link Manager*™ (and firmware upgrade utilities) use UDP communications, which includes UDP broadcast packets. Note: some LAN routers may block some UDP packets. In such instances an Ethernet crossover cable can be used from the host PC to connect directly to the device that is to be configured.

The *Ether-Link Manager*™ utility can be used to locate any LAVA Ether-Serial Link™ type device on the LAN, regardless of the IP address or DHCP setting. This is very convenient if the IP and other settings of the unit are unknown.

Ether-Serial Link Driver Mode interaction with Microsoft's "Verifier" in Windows 2000/XP

General: This note applies Ether-Serial Link serial ports operating in DRIVER mode.

EVENT: Interaction between the Ether-Serial Link and Microsoft's driver verifier software may cause a blue screen system crash. This event will be encountered when accessing or sending data to a serial port enumerated under an Ether-Serial Link.

COMMENTS: The Windows 2000/XP Verifier utility may have settings enabled to verify signed/unsigned drivers. The serial drivers used by the Ether-Serial Link are compatible with Windows 2000/XP, but are not “signed” by Microsoft. Therefore, to prevent system crashes, disable the Verifier settings, and re-boot the PC for the new settings take effect.

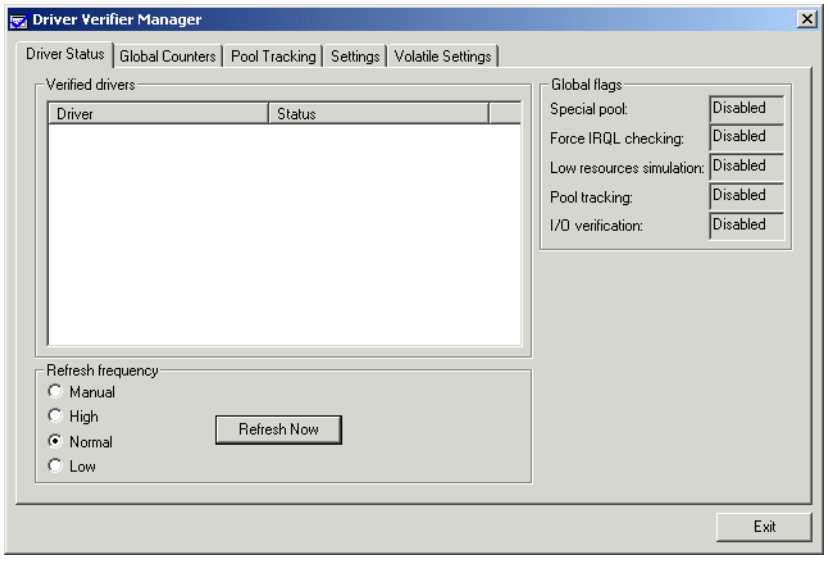
CORRECTIVE ACTION: Disable the Verifier settings as follows:

1. Go to START. In the "Run..." dialog, enter “verifier” and select "OK". The Verifier application will appear with a selection of options.

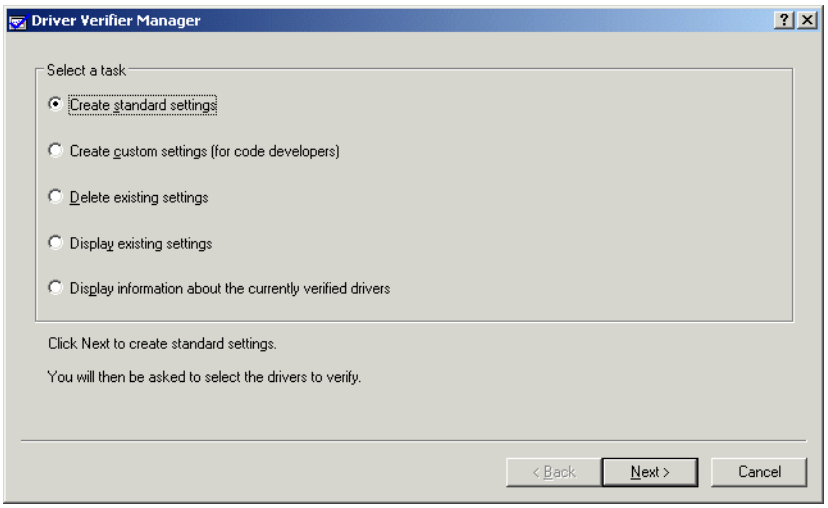
2. Disable settings, and exit from the Verifier. You can confirm on-screen that the settings have been disabled. HOWEVER, you MUST reboot the PC for the new settings to take effect.

3. Reboot the system, then re-check that the Verifier settings have been disabled.

Windows 2000 Verifier

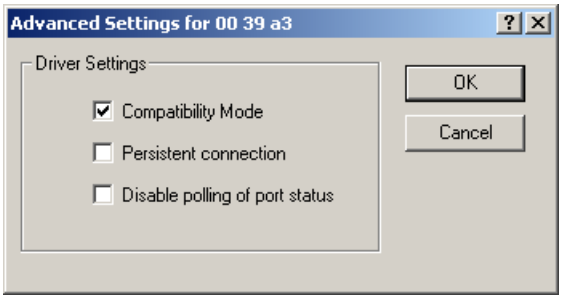


Windows XP Verifier



Virtual port (driver) settings

The Ether-Serial Link™ virtual port allows a serial port on an Ether-Serial Link™ to be used in same manner as a local serial port when the port is configured in Driver Mode (described above). In this state, the driver for the virtual port converts local serial port accesses into TCP/IP packets to be exchanged with Ether-Serial Link™ ports.



Compatibility Mode. The Ether-Serial Link™ virtual port normally operates in high performance mode. Compatibility mode supports applications that require a more accurate simulation of a local serial port. The need for Compatibility mode is indicated when data being sent to the remote serial port is lost or sent repeatedly.

When this option is enabled, the Ether-Serial Link™ Windows driver sends a "write request" with the data packet to the Ether-Serial Link™. The Ether-Serial Link™ signals the completion of the "write request" only after the last data character has been output to the UART. The "write request" is handled in the same way as a "write request" under Windows. This ensures that all the characters that were sent to the Ether-Serial Link™ serial port are in the UART and are output, before the next data character is sent by the driver from the host PC.

Persistent connection. This setting maintains a persistent connection when the client station's connection is broken to the unit, then re-established. This setting is accessed through Window's Device Manager | Ports (COM & LPT).

When this option is enabled the communication link from physical serial port on the Ether-Serial Link™ to the COM port that is enumerated on the PC will be re-established automatically, in event that the power is cycled, the Ether-Serial Link™ is rebooted, or the port is reset.

To most effectively use this feature, match ALL the default serial operating parameters to the parameters that are used in the communications at the serial port and COM ports. Doing so will minimize delay in reconnection as the reconnection procedure uses default parameters first, then the settings that are actually in use.

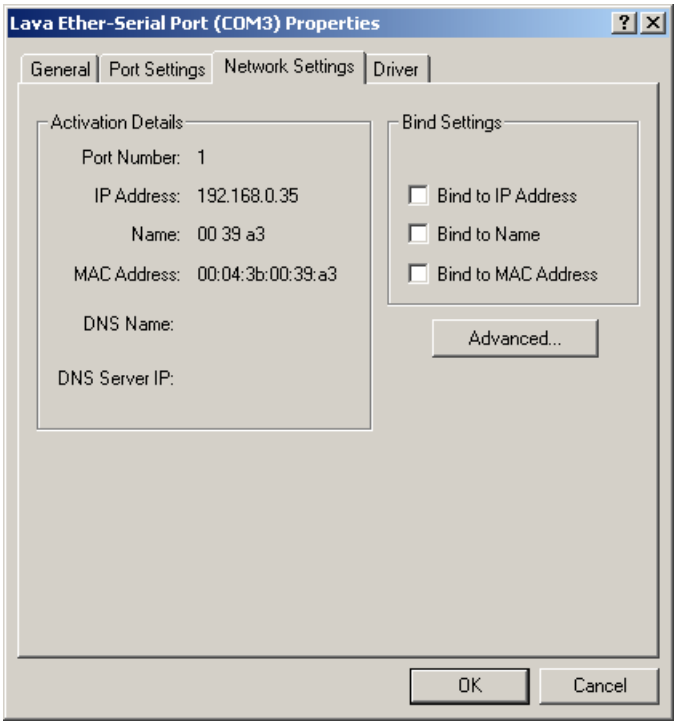
Disable polling of port status. Polling of port status can be disabled. This is useful when a software application accessing the remote serial port repeatedly polls to check port status, slowing access to the remote port. This setting is accessed through Window's Device Manager | Ports (COM & LPT).

COM port applications may send multiple-repeated status polls to the COM port UART. Status polls are passed to the Ether-Serial Link™ serial port UART firmware. These status polls will show up as TCP/IP packets that are exchanged on the LAN between the Windows driver and the Ether-Serial Link™. When the option to suppress the status polls is enabled, the Windows driver will reply to the application status polls using the last set of stored parameters in the driver. Status polls will not be sent to the Ether-Serial Link™ over the LAN. The COM port status changes that are made in the Ether-Serial Link™ are updated and stored in the driver memory, as and when they occur.

Virtual port (driver) bindings

Background. Activating an Ether-Serial Link™ port in effect creates a "virtual" driver port that is associated with a particular physical serial port on a particular Ether-Serial Link™. When a driver virtual port is opened, a connection must be established to the appropriate Ether-Serial Link™. In normal operation, that connection is made to the IP address assigned to the Ether-Serial Link™ at the time the activation was created. However, on many Ethernet/IP networks, the IP address of individual devices may change (for example, when a DHCP server assigns addresses). This address reassignment will cause virtual ports to fail or make unintended connections to unintended Ether-Serial Links™.

Therefore, Ether-Serial Links™ are designed with the ability to "find" an Ether-Serial Link™ port based on a "binding". A port binding is a network identification of the target Ether-Serial Link™ that must match before a driver port will initiate a connection. Bindings may be made to the MAC address, Name, or IP address of the associated device.



Binding to MAC Address. MAC addresses are unique and permanent identifiers for Ethernet devices, and are assigned to devices when they are manufactured. Binding to an Ether-Serial Link™'s MAC address ensures that a virtual port always makes a connection to that same physical device, even if the device's IP address or Name have changed. Binding to MAC address provides security and reliability. Virtual ports will only connect to that single particular physical device.

Binding to Name. Device Name is a 15-character field assigned by the Ether-Serial Link™ system administrator. Binding a virtual port to a device's Name will ensure that a connection will be made to an actual port if a device can be found with the name specified in the "Bindings" dialog box. This port may be on a different physical device from the one with which the virtual port was initially communicating (that is, a device with a different MAC address), or it may be a port on the same physical device but with a changed IP address, or both. Binding to a device's Name provides reliability and dynamic connections. Virtual ports will connect to the device with a name that matches the binding. If a particular device is replaced with a new one with the same assigned Name, the virtual port will establish the connection. Also, Names can be reassigned at any time to cause client PCs to connect to new devices without being reconfigured.

Binding a virtual port to a Name allows that virtual port to be identified on the network with a more representative mnemonic than an IP address or MAC address. For example, a security system might name its virtual port targets "Camera1", "Camera2", "CardReader6", and so forth.

Binding to IP Address. The IP address of a network device is assigned by either the Ether-Serial Link™ system administrator or through a DHCP server. Binding a virtual port to an IP address will force that virtual port to attempt a connection to any device with that particular IP address only. Binding to IP address provides dynamic connections. Device IP addresses can be reassigned at any time to cause client PCs to form new connections without being reconfigured. Note: when no bindings have been stipulated, a virtual port is implicitly bound to an IP address because such an address is required to make a connection.

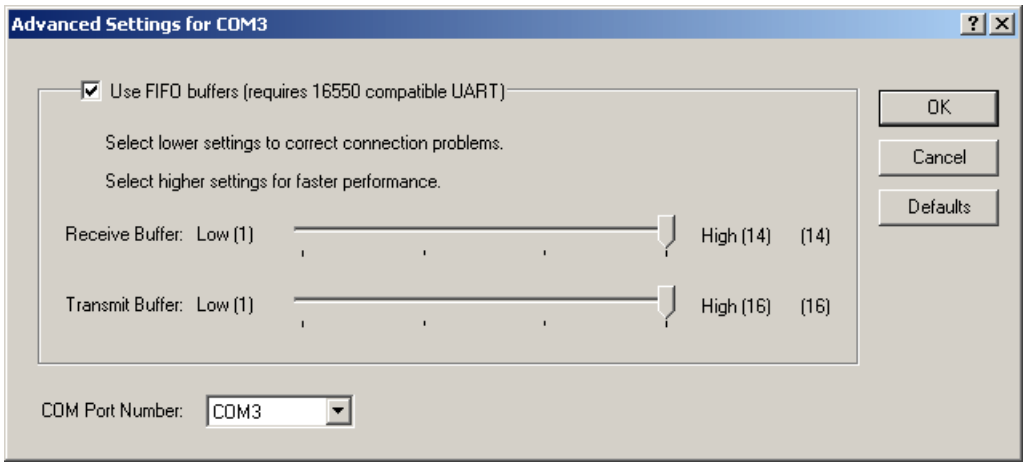
Binding to more than one parameter. Usually, only one degree of binding is sufficient for an individual virtual port to achieve the goals of robust communication and flexibility. It is however also possible to bind a virtual port to more than one parameter. Doing so requires careful consideration of the implications of maintaining connectivity. If multiple bindings are selected for one port, *all* individual binding parameters must match for a connection to be made. For example, enabling both IP address and MAC address bindings for a virtual port would cause that virtual port to find a particular physical device by MAC address, but make a connection only if the device's IP address also matched the binding setting.

Resetting registry information in Windows

Under Windows you may need to use the line command "arp -d *" to clear the OS registry of the previous IP and MAC relationship. This may become apparent when first using and setting up a device with the factory default IP settings.

Virtual port COM number assignment

By default, an Ether-Serial Link™ virtual serial port is assigned to the next available COM number when it is activated. This COM number assignment can be reassigned to any unused COM number from 1 to 256.



Transmit FIFO buffer configuration

Each serial port of an Ether-Serial Link™ has a first-in first-out (FIFO) buffer for transmitting data and a FIFO buffer for receiving data.

Transmit FIFO buffers are configurable and can buffer from 1 to 16 characters in the UART, for output to the serial port. Receive FIFO buffer levels are fixed at 8 characters.

Some serial devices, when they are not able to accept data from the serial port, will send an XOFF character to that port. Upon receipt of the XOFF character, the Ether-Serial Link™ will output the remaining characters in the port's transmit FIFO buffer. The Ether-Serial Link™ will then only accept more characters from the PC driver after an XON has been issued by the serial device.

If the nature of a serial device is such that it cannot accept buffered characters after sending an XOFF character, the user of an Ether-Serial Link™ can set the transmit buffer level to some low value such as 1 or 2 characters, to overcome the situation where data is sent to a device that is unable to accept it. This may be necessary for POS printers and some CNC devices or PLCs, for example.

The transmit FIFO buffer level can be set using either the *Ether-Link Manager*™ (Port Settings |Advanced), or the Telnet configuration interface.

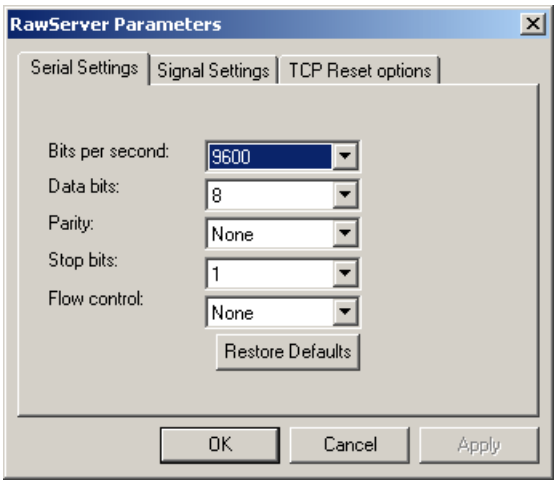
3. Raw Server Mode

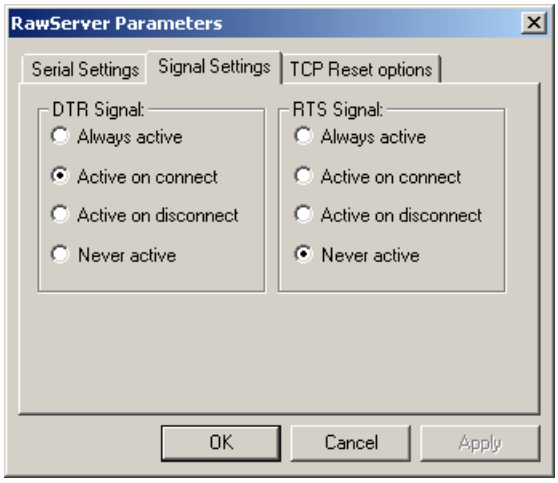
This mode is used to establish a raw TCP connection to an Ether-Serial Link™ port. The physical port on the Ether-Serial Link™ is made available as a network resource with a specified IP address and port number. In this mode, authorized users can set bits per second, data bits, parity, stop bits, and flow control port-by-port, to match the requirements of the Data Terminal Equipment (DTE) connected to the physical Ether-Serial Link™ port.

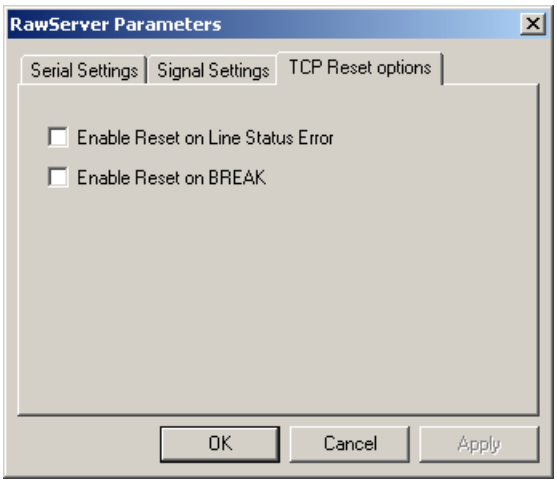
Control lines can be triggered by the TCP connection being established. Under the "Signal Settings" tab shown below, the user can set how the control lines are to react when a TCP connection is established.

The TCP connection can be set to close on two conditions: a status error at the serial port (such as a parity error, buffer overrun, cable disconnection, etc.), or when a BREAK is sent to t he serial port. By default these conditions are disabled.

Applications: Remote monitoring, security systems.





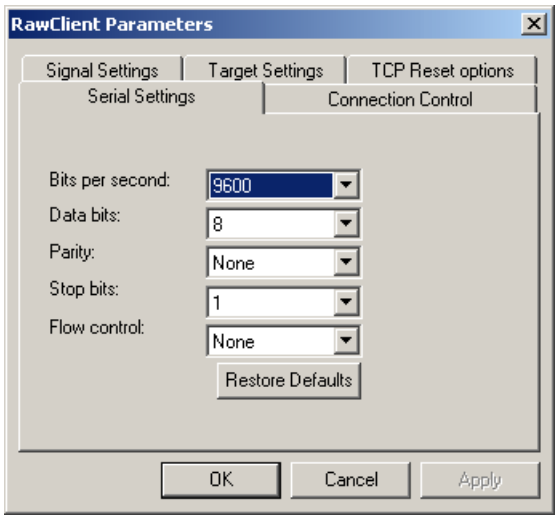


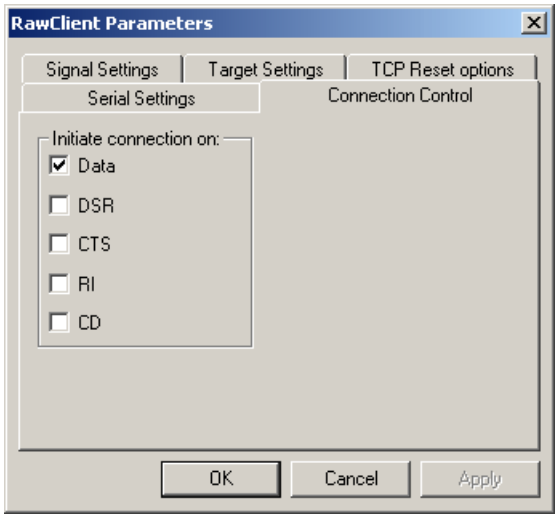
4. Raw Client Mode

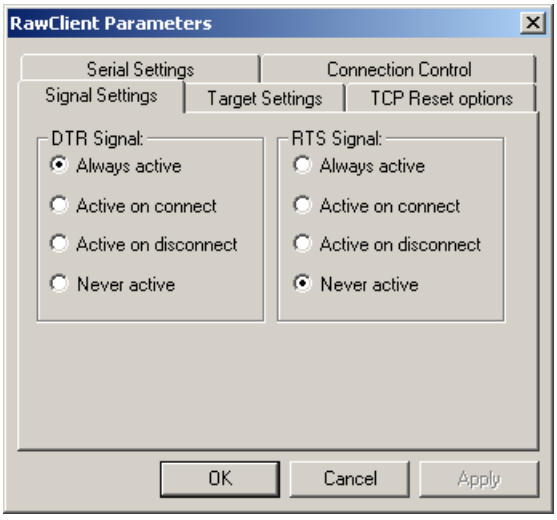
This mode is used to establish a raw connection to the Ether-Serial Link™. The physical port on the Ether-Serial Link™ is configured to initiate a connection to a pre-defined IP address and port number, specified by an authorized user of that port in the "Target Settings" dialog box. In this mode, authorized users set bits per second, data bits, parity, stop bits, and flow control port-by-port, to match the requirements of the Data Terminal Equipment (DTE) connected to the physical Ether-Serial Link™ port. Operating as a client, the Ether-Serial Link™ will open a connection to the target IP address and port when it detects one or more of the following line conditions: data, DSR, CTS, RI, or CD. Users can configure which of these triggers will be in effect for any Ether-Serial Link™ port set in this mode in the "Connection Control" dialog box. "Signal Settings" are also settable by the user, as described in Raw Server mode.

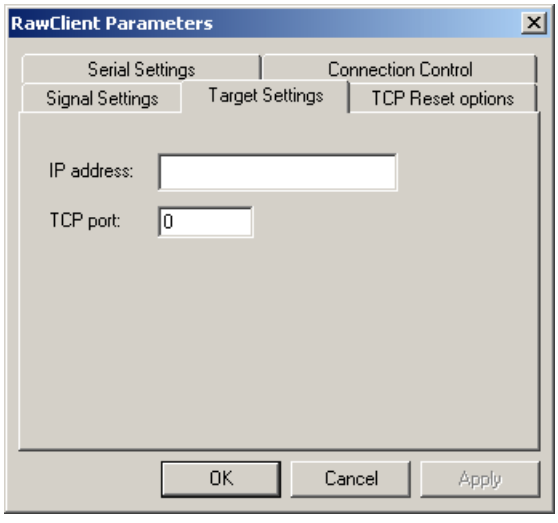
The TCP connection can be set to close on two conditions: a status error at the serial port (such as a parity error, buffer overrun, cable disconnection, etc.), or when a BREAK is sent to t he serial port. By default these conditions are disabled.

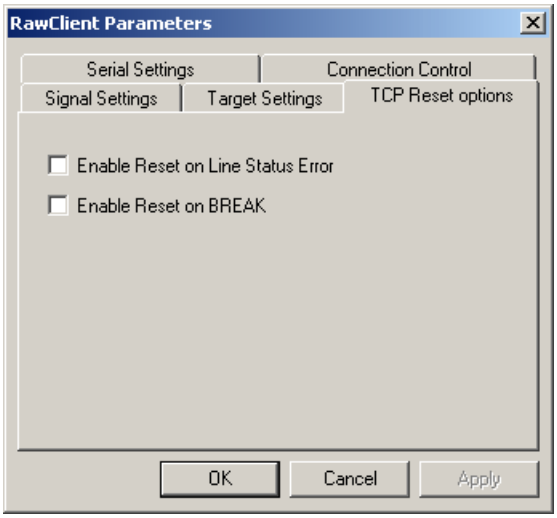
Application: Remote control.









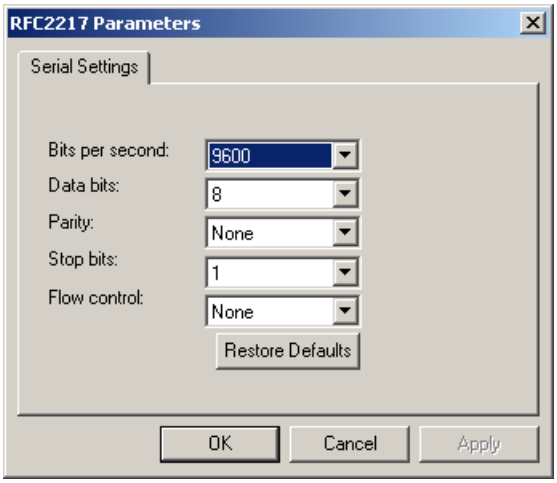

RawClient Parameters
Serial Settings
Connection Control
Signal Settings
Target Settings
TCP Reset options
Enable Reset on Line Status Error
Enable Reset on BREAK
OK
Cancel
Apply

5. RFC2217 Mode

Users who want to communicate with an Ether-Serial Link™ across gateways or routers can configure a port on an Ether-Serial Link™ to operate in RFC 2217 mode, allowing serial port configuration commands and serial data to be sent to an Ether-Serial Link™ using the RFC 2217 framework for serial port control over Telnet. For those not familiar with RFC 2217 (the Telnet COM Port Control Option), the text of the actual RFC (Request For Comments) is available at www.faqs.org/rfcs/rfc2217.html or at any number of other sites.

This mode opens an Ether-Serial Link™ port when a TCP connection is established to the "local port". Port settings are embedded in data stream in accordance with RFC 2217.

Applications: UNIX systems and other platforms that have RFC 2217 Telnet capability can access and control the serial COM port of the Ether-Serial Link™.



6. Data Connect Mode

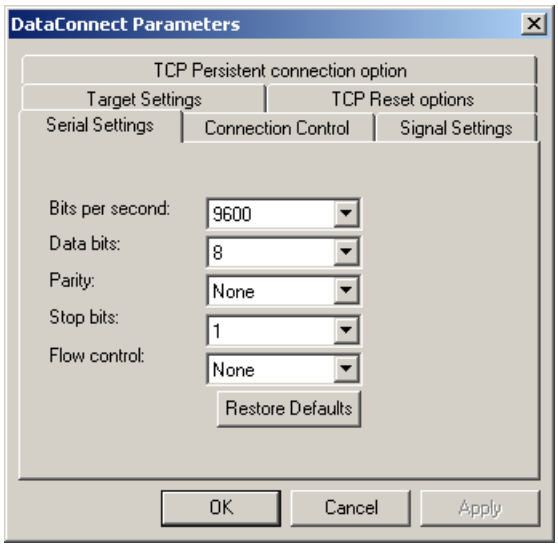
Data Connect mode combines Raw Client mode and Raw Server Mode. In this mode, the Ether-Serial Link™ will either initiate a TCP connection when activity is detected at the serial port, or it will receive TCP packetized serial data from the network port when an outside client connects to it.

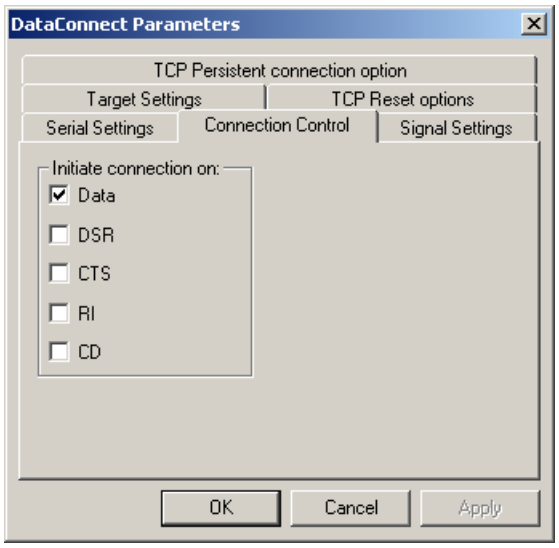
The configuration parameters for this mode combine all the parameters found in Raw Client and Raw Server modes (see above for details).

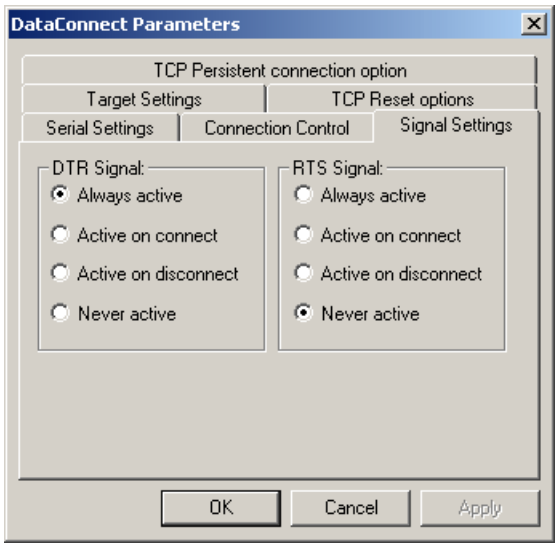
With two Ether-Serial Links™ in this mode placed "back to back," the connection between them becomes in effect a cable extender over an Ethernet connection.

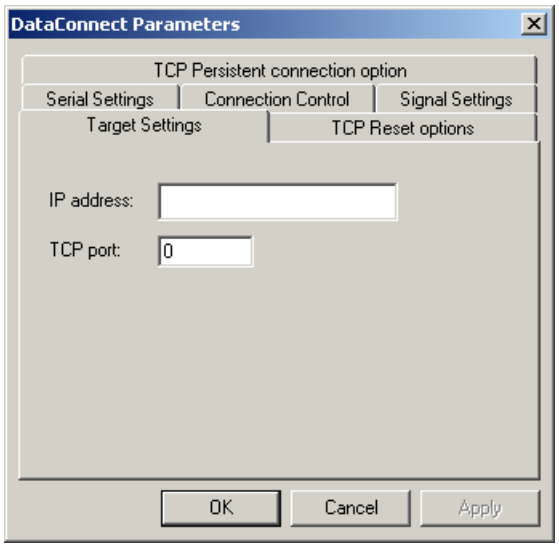
The TCP connection can be set to close on two conditions: a status error at the serial port (such as a parity error, buffer overrun, cable disconnection, etc.), or when a BREAK is sent to t he serial port. By default these conditions are disabled.

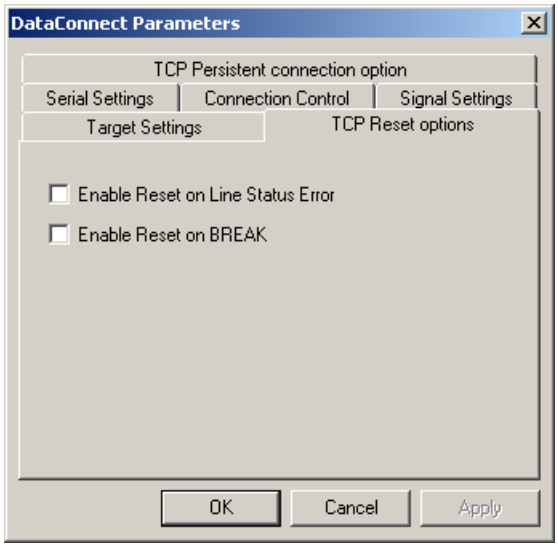
Applications: Data connect mode provides a serial-to-serial communication link that would not be practicable to implement as a directly-wired link; it can replace serial cables with an Ethernet connection.

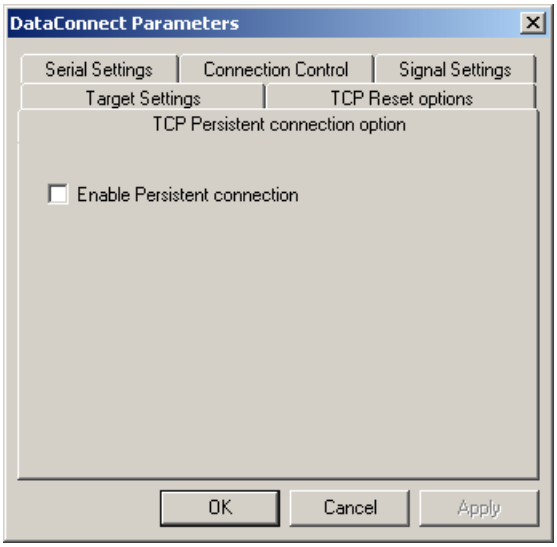









DataConnect Parameters
TCP Persistent connection option
Serial Settings
Connection Control
Signal Settings
Target Settings
TCP Reset options
Enable Reset on Line Status Error
Enable Reset on BREAK
OK
Cancel
Apply


DataConnect Parameters
Serial Settings
Connection Control
Signal Settings
Target Settings
TCP Reset options
TCP Persistent connection option
Enable Persistent connection
OK
Cancel
Apply

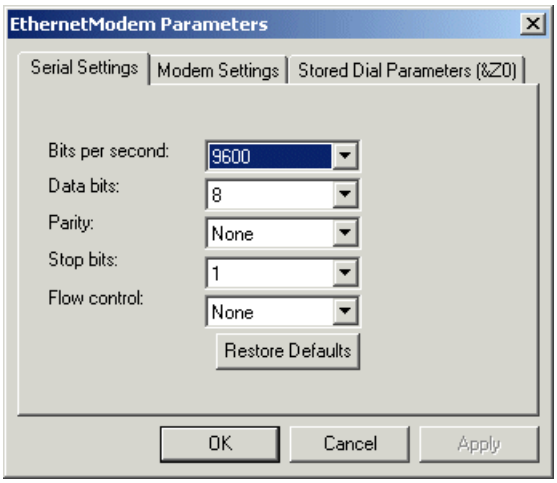
7. Ethernet Modem Mode

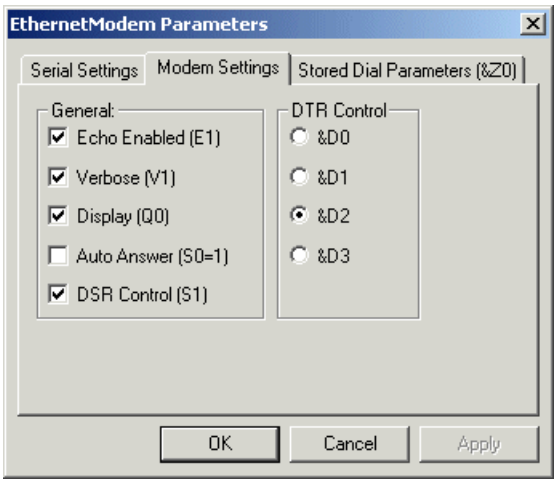
This mode allows Data Terminal Equipment (DTE) to control the Ether-Serial Link™ using the conventional modem "AT" command set. In this mode, a DTE user is able to "dial" an IP address and TCP port. Incoming TCP connections are handled under AT command set rules.

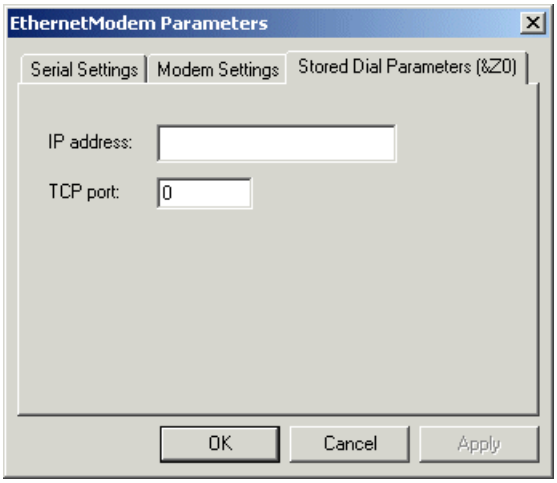
The standard serial port settings for bits per second, data bits, parity, stop bits, and flow control are available for configuration, and should be set to match the DTE.

Modem settings include ECHO Enabled (E1), Result Codes Format Verbose (V1), Result Codes Display Return (Q0), Auto answer (S0=1), DSR Control (&S1). Modem DTR control can be set to &D0, &D1, &D2, or &D3.

Applications: To provide the standard "AT" command interface to user applications that wish to communicate with devices in an Ethernet environment. Remote console management that looks for AT interfacing can use this application.







Ethernet Modem mode operation: Ether-Serial Link™ port as a modem

Procedure:

1. Place the serial COM port into Ethernet Modem mode (via Telnet IP:23, Web page IP:80, or in the *Ether-Link Manager*™). Note: the factory default IP for the Ether-Serial Link™ is 192.168.0.35.
2. The serial port behaves like a "universal" HAYES-type modem to the device connected at the serial port. The device receives the expected modem replies to the "AT commands" sent, when communicating with the serial (modem) port.

General "AT" Commands:

The "AT" commands direct a modem to dial, answer, hang up, and perform other communication tasks. The **commands** are preceded by **AT**, and followed by <**cr**> (ie. ATcmdCR). The only exception is for the +++ sequence, which is used to **place modem into command mode**.

The modem **must be in command mode to accept AT commands**. Any command sent to the modem while it is in the transmission (CONNECTed) mode **is passed as data**.

The following command set is supported by the serial (modem) port of the Ether-Serial Link™:

Command	Description	Parameters
AT	The sequence AT followed by <CR>, returns OK, indicating that the serial port is in modem mode.	
AT**A**	**Answer**. After this command has been entered, the modem is placed into the **answer mode**: if an incoming RING is received, the modem will CONNECT with the incoming client. The ATA command is used to "answer" an incoming RING, which results in a CONNECT. If ATA is entered, and there is no incoming RING within 5 seconds, a timeout will occur, a NO CARRIER message will be displayed, and the modem will remain in command mode. NOTE: To place the modem into "**auto-answer**" mode, set the S0 register for a non-zero value (ie. ATS0=1 and AT&W to save).	
AT**Dstring**	**Dial**. Modem makes a connection to the IP address and port indicated by the string.	string = aaa.bbb.ccc.ddd:pppp
AT**DS=n** **DSn**	**Dial Stored**. Modem makes a connection to the stored IP and port address. To store an IP:Port, use AT&Z0="IP:Port" Note re **ATD**… commands: if the destination IP:Port does not accept the connection within 25-30 seconds, a NO CARRIER message will be displayed, and the modem will remain in command mode.	n = 0 remote location
AT**En**	**Echo**. Host commands are echoed.	n = 0 disable echo n = 1 enable echo (default)
AT**H** or AT**H0**	**(GO) on HOOK**. Close the connection. Enter the COMMAND mode using "+++", then enter ATH to close the connection.	n = 0 close the connection
AT**In**	**Inquiry**. Displays information about the modem.	n = 0 device name n = 1 details of unit n = 2 more detailed + IP n = 3 all info plus profile
AT**O**	**Go Online**. Returns from command mode to an active connection (if one was previously established).	
AT**Qn**	**Display Result Codes**. A "result" can be output after each command. Also see ATVn for format of result.	n = 0 display result n = 1 do not display result

<table>
<tr><th>Command</th><th>Description</th><th>Parameters</th></tr>
<tr><td>ATSr=n</td><td>Set Register. Set value of register r to n.
Note: only “S0” can be set for 0 or non zero. All others are fixed.

<table>
<tr><th>“Sr” Register</th><th>“n” Value (default)</th><th>Purpose</th></tr>
<tr><td>S0</td><td>0</td><td>auto-answer disabled = 0
auto-answer enabled = non-zero</td></tr>
<tr><td>S1</td><td>0</td><td>not used</td></tr>
<tr><td>S2</td><td>43</td><td>escape to command mode char “+”</td></tr>
<tr><td>S3</td><td>13</td><td><CR> character. Fixed.</td></tr>
<tr><td>S4</td><td>10</td><td><LF> character. Fixed.</td></tr>
<tr><td>S5</td><td>8</td><td>backspace character defined</td></tr>
<tr><td>S6</td><td>3</td><td>not used</td></tr>
<tr><td>S7</td><td>60</td><td>not used</td></tr>
<tr><td>S8</td><td>2</td><td>not used</td></tr>
<tr><td>S9</td><td>6</td><td>not used</td></tr>
<tr><td>S10</td><td>7</td><td>not used</td></tr>
<tr><td>S11</td><td>70</td><td>not used</td></tr>
<tr><td>S12</td><td>50</td><td>guard time in 20 ms increments</td></tr>
<tr><td>S13</td><td>0</td><td>not used</td></tr>
</table></td><td>r = register number [r < 14]
n = value to assign</td></tr>
<tr><td>ATSr?</td><td>Display Register. Value of register r.
Individual “S” register values can be obtained.</td><td>r = register number</td></tr>
<tr><td>ATVn</td><td>Format of Result Codes.

Note: text output is: <CR><LF>text<CR><LF>
Note: numeric output is: number<CR>

<table>
<tr><th>Text Output</th><th>Numeric</th><th>Meaning</th></tr>
<tr><td>OK</td><td>0</td><td>command executed</td></tr>
<tr><td>CONNECT</td><td>1</td><td>connection established</td></tr>
<tr><td>RING</td><td>2</td><td>connection is being requested</td></tr>
<tr><td>NO CARRIER</td><td>3</td><td>connection broken or not made</td></tr>
<tr><td>ERROR</td><td>4</td><td>illegal command</td></tr>
</table></td><td>n = 0 numeric form
n = 1 text form</td></tr>
<tr><td>ATZ</td><td>Reset. Close any open connections, and reset the S registers to the saved values.</td><td></td></tr>
<tr><td>AT&Cn</td><td>DCD Control. [as Output].</td><td>n = 0 always active
n = 1 follows connect status</td></tr>
<tr><td>AT&Dn</td><td>DTR Control. [as Input].</td><td>n = 0 ignore
n = 1 on-to-off: go to command mode & maintain the connection.
n = 2 on-to-off: go to command mode & close the connection.
n = 3 on-to-off: go to command mode, close connection & do Reset.</td></tr>
</table>

Command	Description	Parameters
AT**&F**	**Load Factory Settings**.	Default parameters for commands and parameters are set: Serial Port: **9600 8-data no parity 1-stop bit no flow control** Address of remote host: **0.0.0.0:0** Auto-Answer is disabled: **S0 register = 0** Command line echo is enabled. **ATE1** Result codes are displayed as text. **ATQ1, ATV1** DCD line follows connection status [active if connected]. **AT&C1** DTR control: if on-to-off, then go to command mode and close connection. **AT&D2** DSR line is always active. **AT&S1**
AT**&Sn**	**DSR Control**.	n = 0 set DSR always active n = 1 follows connection status.
AT**&V**	**View Profile Settings**. Displays the S-register values, stored IP:Port of remote host, serial port and control line settings.	
AT**&W**	**Save Configuration Settings**. Saves the current settings into memory for re-use in subsequent operations.	
AT**&Z?**	**Display Address Settings**. Displays the stored IP:Port for the remote host.	
AT**&Z0=s**	**Store Address Settings**. Stores the IP:Port number for the remote host.	
+++	**Escape from the active connection to the modem command line mode**. Three consecutive "+" characters will place the modem into command mode. The first two "+" characters will be transmitted to the remote host on the link, third "+" character will place the modem into command mode – the third character is not transmitted to the remote host. The connection to the remote host is placed "on-hold" when the third "+" character is keyed in. No further data will be transmitted unless the transmission mode is re-enabled.	The follow-on options are: – use the ATO command to re-enable transmission to remote host – use the ATH command to terminate the connection – use the ATZ command to terminate the connection, and reset parameters to stored settings.

Extended AT commands:

EXTENDED "AT" commands are used for re-configuring the network parameters of the Ether-Serial Link™: password, IP address, port TCP socket number, net mask address, gateway address, DHCP enabled/disabled. Extended AT cammmands can also save parameters, get password/IP/TCP port/net mask/gateway, and reboot the Ether-Serial Link™.

To use any of the extended AT commands, the user must log in using the password that has been assigned to the Ether-Serial Link™ (using one of Telnet, web browser, or *Ether-Link Manager*™ utility).

"AT%NLOGIN=password" must be entered before starting a session using the extended AT commands.:

Command	Description	Parameters
AT %**NLOGIN=password**	**Login.** "Password" is the password that has been assigned to the Ether-Serial Link™. If no password is in use, do not enter any characters after the "=". Press <CR>, then proceed to use the extended AT commands. NOTE: AT command session log-in passwords, whether entered in upper-case or lower-case, are converted to all upper-case before being compared for validity.	
AT **%NPPASSWORD=password**	**Network Put Password.** Assigns a new password to the Ether-Serial Link™. The new password is entered into a temporary memory register. Other commands may follow this command, but a SAVE command is required to submit this number for use. The SAVE command must be followed by a REBOOT command to implement the new password. NOTE: To undo any changes before issuing a REBOOT command, invoke the ATZ command. NOTE: AT command session log-in passwords, whether entered in upper-case or lower-case, are converted to all upper-case before being compared for validity.	
AT **%NPTCPPORTx=nn...n**	**Network Put TCP Socket Number for serial port *x*.** Assigns a new TCP socket number (nn...n) to serial port x, where *x* is a port on the Ether-Serial Link™. By default Port 1 is assigned 4098, Port 2 is assigned 4097, and so on. The number "nn...n" is entered into a temporary memory register. Other commands may follow this command, but a SAVE command is required to submit this number for use. The SAVE command must be followed by a REBOOT command to implement the new socket number for the port. NOTE: To undo any changes before issuing a REBOOT command, invoke the ATZ command.	
AT **%NPIP=xxx.xxx.xxx.xxx**	**Network Put IP.** Enters a new IP address for the Ether-Serial Link™. This number is entered into a temporary memory register. Other commands may follow after this command, but a SAVE command is required to submit this IP address for use. The SAVE must be followed by a REBOOT command to implement the new IP address for the Ether-Serial Link™. NOTE: To undo any changes before issuing a REBOOT command, invoke the ATZ command.	
AT **%NPMASK=xxx.xxx.xxx.xxx**	**Network Put Netmask.** Enters a new netmask address for the Ether-Serial Link™. This number is entered into a temporary memory register. Other commands may follow after this command, but a SAVE command is required to submit this netmask address for use. The SAVE must be followed by a REBOOT command to implement the new netmask address for the Ether-Serial Link™. NOTE: To undo any changes before issuing a REBOOT command, invoke the ATZ command.	

Command	Description	Parameters
AT **%NPGATE=xxx.xxx.xxx.xxx**	**Network Put Gateway.** Enters a new gateway address for the Ether-Serial Link™. This number is entered into a temporary memory register. Other commands may follow after this command, but a SAVE command is required to submit this gateway address for use. The SAVE must be followed by a REBOOT command to implement the new gateway address for the Ether-Serial Link™. NOTE: To undo any changes before issuing a REBOOT command, invoke the ATZ command.	
AT **%NPDHCP=ENABLED**	**Network Put DHCP Enabled.** Sets the Ether-Serial Link™ for acquiring its IP address from a network DHCP Server. If DHCP is enabled the Ether-Serial Link™ will not operate until a valid IP address is obtained. To implement this command, invoke SAVE, then invoke REBOOT. NOTE: Use *Ether-Link Manager*™ to locate the Ether-Serial Link™ on the network to make any changes if the Ether-Serial Link™ has not yet received a valid IP address.	
AT **%NPDHCP=DISABLED**	**Network Put DHCP Disabled.** Disables the DHCP mode of operation. A set of valid/useable IP/netmask/gateway addresses will need to be entered for Ether-Serial Link™ operation on the network. To implement this command, invoke SAVE, then invoke REBOOT.	
AT **%NSAVE**	**Network Save.** Stores the parameters that have been entered into the Ether-Serial Link™ in anticipation of making the changes permanent. If the SAVE command is not invoked before rebooting, the previous parameters will remain valid after the reboot.	
AT **%NREBOOT**	**Network Reboot.** Reboots the Ether-Serial Link™. If a SAVE command has been issued before rebooting, then the parameters that were stored by the SAVE command will be used in the reboot of the Ether-Serial Link™.	
AT **%NGPASSWORD**	**Network Get Password.** Returns the password that is stored in the register. The password can be the current valid password, or the last password that was entered (using the AT%NPPASSWORD cmd) prior to a SAVE and REBOOT sequence.	
AT **%NGIP**	**Network Get IP.** Gets the IP address that is stored in the temporary memory register of the Ether-Serial Link™. This may also be the current address being used, if a change command has not been invoked. If the ATZ command is issued prior to the Network Get IP command, then the returned address is the current address in use for the Ether-Serial Link™.	
AT **%NGMASK**	**Network Get Netmask.** Gets the netmask address that is stored in the temporary memory register of the Ether-Serial Link™. This is the current address being used, if a change command has not been invoked. If the ATZ command is issued prior to the Network Get Netmask command, then the returned address is the current netmask in use for the Ether-Serial Link™.	

Command	Description	Parameters
AT **%NGGATE**	**Network Get Gateway.** Gets the gateway address that is stored in the temporary memory register of the Ether-Serial Link™. This can also be the current address being used, if a change command has not been invoked. If the ATZ command is issued prior to the Network Get Gateway command, then the returned address is the current address in use for the Ether-Serial Link™.	
AT **%NGTCPPORT=x**	**Network Get TCP Socket Number for serial port *x*.** Returns the current value of the TCP port socket number for serial port *x*.	

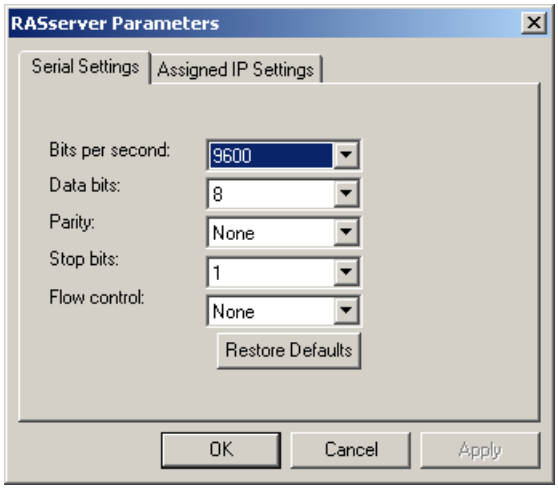
8. RAS Server Mode

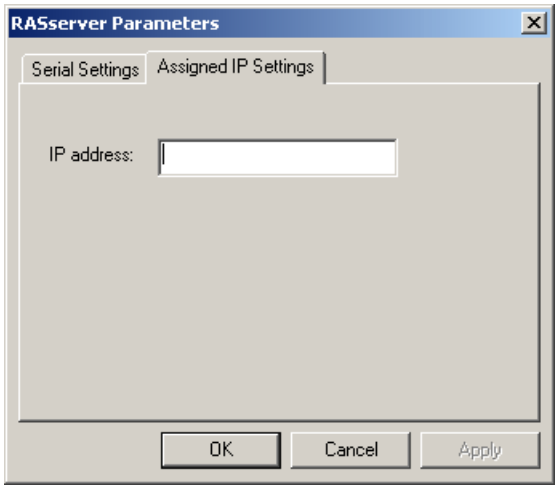
Windows-based devices that have Remote Access Services (RAS) client capability but are not equipped with a direct network interface can access the network through a serial port using the Point-to-Point Protocol (PPP).

The serial port of the RAS client device is attached to the serial port of the Ether-Serial Link™, configured in RAS Server mode. The RAS negotiations are conducted between the RAS client and the Ether-Serial Link™ RAS server.

In the initial negotiations between the RAS client and the Ether-Serial Link™, an IP address (configured by user) is assigned to the RAS client, which is used for all Ethernet communications related to the RAS client.

Applications: Windows CE embedded systems, Palm type units, or other portable data acquisition devices that may need access to a TCP/IP-Ethernet environment, and have PPP capability, but do not have a Ethernet port, can be connected to the serial port of an Ether-Serial Link™.





9. Disabled Mode

This mode simply disables a port; it has no configuration settings.

10. WAN Port configuration

Enter the IP address and TCP port number for WAN port configuration using this feature. Ensure that the IP address does not conflict with an IP address also in use on the local area network.

The WAN activation is normally used to enumerate a remote serial port as a local COM port on the PC, where the remote serial port is located on a separate network (i.e., different LAN, over the Internet).

To create a WAN COM port enumeration, the target COM port must not be on-line on the LAN during enumeration. After enumeration, the Ether-Serial Link™ can be placed on-line on the LAN. WAN-enumerated COM ports will communicate with the remote COM ports on the LAN.

11. Port reset

To reset a serial port, when in the *Ether-Link Manager*™, right-click on the serial port and select RESET from drop down menu.

Each serial port can be reset without disturbing the other serial ports on the Ether-Serial Link™. The set/stored operating settings for the serial port are not lost/changed after a reset. A serial port reset will also occur if operating parameter settings are changed. This is the case if after entering or viewing the settings the OK or Submit function is used. If the serial port is in use, the communications at that port will be interrupted; therefore, the user should keep this in mind when resetting the port. Options to restrict port resetting are provided.

12. Device reboot

To reboot the Ether-Serial Link™, when in the *Ether-Link Manager*™, right-click on the device and select Advanced|Reboot.

This function will fully reboot the Ether-Serial Link™, including a reset of all serial ports. The set/stored operating settings for the network and serial ports are not lost/changed after a reboot. A reboot will occur if the network parameter settings are changed. This is the case if, after entering or viewing the settings, the OK or Submit function is used. If serial ports are in use, the communications at those ports will be interrupted; therefore, the user should keep this in mind when rebooting the device.

13. Subnet configuration with the Ether-Serial Link

If the Ether-Serial Link is on a different subnet from a client station that is attempting to access it, the Ether-Serial Link will be visible in an instance of the *Ether-Link Manager*™ running on the the client station, and its ports can be activated on that client station, but applications running on the client station will not be able to open a serial port on the Ether-Serial Link.

For example, if the IP address of the Ether-Serial Link is 192.168.0.35 and the IP address of the client station is 192.168.1.1, the two devices are on different subnets (as indicated by the difference between the "0" and the "1" in the third portion of the two IP addresses).

Two solutions exist:

1. change the subnet of the either the Ether-Serial Link or the client station so that they are both on the same subnet or,
2. change the subnet mask of the Ether-Serial Link to encompass both subnets. By default, the Ether-Serial Link has its subnet mask set to 255.255.255.0. This setting excludes all but the fourth portion of the IP address when assessing IP addresses. In the example above, a subnet mask of 255.255.0.0 will include the Ether-Serial Link in both subnets.

14. Re-establishing a TCP connection with the Ether-Serial Link

The nature of TCP socket connections means that a TCP socket established with a host will hold that TCP socket "open" even when active communication has been broken. This situation might occur, for example, when an Ether-Serial Link serial port is connected in Raw Server mode with a host PC that is rebooted without the TCP connection to the Ether-Serial Link being properly closed. While the TCP connection continues to be held open, further TCP connections cannot be established. This "lockout" is normal TCP behaviour designed to allow communications to be smoothly re-established between a TCP client and server.

In the circumstance where connection to an Ether-Serial Link is broken, it may be desireable to allow another client to access the Ether-Serial Link port. To prevent a "lock-out" from persisting, the Ether-Serial Link monitors the TCP port for an active and valid connection every time that a TCP connection request is attempted.

If the connection between the Ether-Serial Link and a client has been "broken," then the next client attempting to connect to the serial port will need to make ***two*** attempts to connect. The first attempt will cause the Ether-Serial Link firmware to check for the validity of the previously-established TCP connection. If the connection is determined to be "broken," then the Ether-Serial Link will "properly" close the TCP socket. At that point a second connection attempt can be accepted and set up as a new TCP socket. The time interval between connection attempts should be in the order of about 10 seconds.

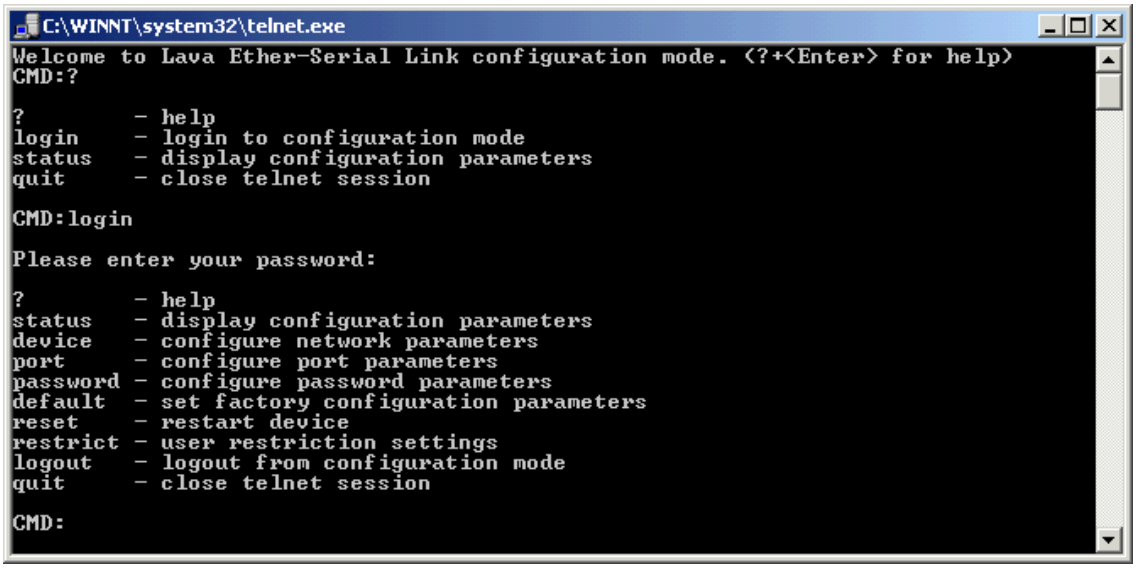
Note for application programmers: When writing connection code, make the attempt to connect several times, in the event that the previous TCP connection with the Ether-Serial Link serial port was not closed properly under TCP rules.

15. Configuring the Ether-Serial Link using Telnet

1. Open a Telnet session with local echo on and carriage return/line feed on.

Windows example

1. Go to Start|Run, type "telnet", and click on "OK".
2. A Telnet console opens. At the Telnet prompt, set local echo on by typing "set local_echo" and pressing <Enter>. Set carriage return and line feed on by typing "set crlf" and pressing <Enter>. NOTE: These commands may differ slightly depending on the Telnet client you are using.
3. Open a Telnet session to the Ether-Serial Link. At the Telnet prompt, type ""open [IP address]" and press <Enter>, where [IP address] is the address of the Ether-Serial Link with which you wish to communicate. The Telnet configuration screen will open.
4. In the Telnet console, you can configure device, port, and password settings, restore factory defaults, restart the device, or query the device to display its current status.



2. Configure the Ether-Serial Link.

The following network and port settings can configured in a Telnet session:

Device settings (for the Ether-Serial Link device):

IP address
Net mask
Gateway
NetBios name (Device Name)
DHCP mode
Device password

Port settings (for each Ether-Serial Link port individually):

Port mode (see ADVANCED readme file for description of the various port modes)
TCP port number
Port mode settings (see ADVANCED readme file for details on port mode settings)
Port access
Port passwords
Port reset
Port status

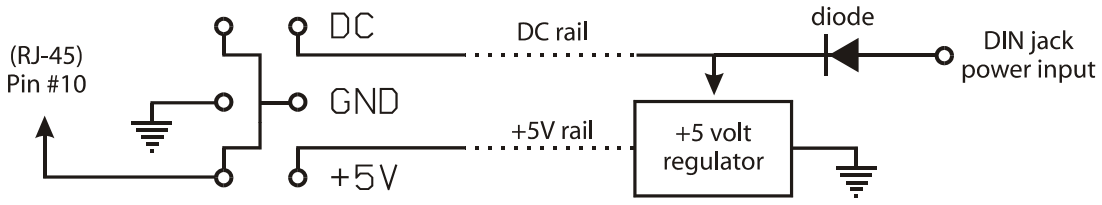
After changing device settings, the Ether-Serial Link will be reset. Changes to port settings are implemented without resetting the Ether-Serial Link.

16. Jumpering Pin #10 on the RJ-45 serial port connector

General:

Devices with an RJ-45 10-pin connector have a jumper-selectable power input/output option at Pin #10. A jumper is located on the interface board (I/F board), to the right of the RJ-45 10-pin connector, to configure this pin.

Jumper pin connections (viewed looking onto the component side)



Pin #10 can be used to supply power to serial devices attached to the RJ-45 serial port, or to supply power to the unit through the serial port.

Pin #10 is separately jumpered for each RJ-45 connector. Consequently, each port can be configured independently of the others.

Jumper settings:

Four jumpering options are possible:

[no jumper]

This is the factory default jumper setting. When the connector is unjumpered, Pin #10 is unconnected.

[DC]

When the jumper is in this position, Pin #10 is connected to the power input pcb trace. Three scenarios for configuring Pin #10 for DC power exist:

1. Power can be supplied to the unit via the DIN connector. In this case, the input voltage (after the diode) is present at Pin #10. Voltage can be drawn from Pin #10 to power some external equipment that is connected to the RJ-45 10-pin connector. When drawing power from a multi-port unit, consider the total possible power draw of the external equipment being connected and powered.

Note: The DIN power input has a protective input diode connected in series with the center pin of its connector (as diagrammed above). The cathode of the diode is connected to the power input pcb trace.

2. Power can be supplied to the unit via Pin #10 of the RJ-45 connector. If no voltage is supplied to the DIN connector, power can be supplied via Pin #10 to the unit. **Careful consideration must be given to the implications of**

supplying power to the system in this manner. Consult Technical Support before attempting this configuration.

Note: Pin #10 *does not* have a protective input diode connected to its connector.

3. Power can be supplied to the unit via both Pin #10 of the RJ-45 connector and the DIN connector at the same time. If voltage is supplied both at the DIN connector and at Pin #10 (i.e.: battery back-up), current to power the unit (150-200ma) will be drawn from the higher-voltage source. **Careful consideration must be given to the implications of supplying power to the system in this manner. Consult Technical Support before attempting this configuration.**

Note: Pin #10 *does not* have a protective input diode connected to its connector.

[GND]

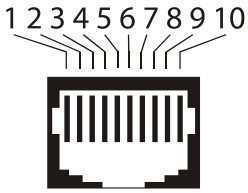
When the jumper is in this position, Pin #10 is connected to the unit's common ground.

[+5V]

When the jumper is in this position, Pin #10 is connected to the output of the +5 volt internal power regulator circuit. Voltage of +5 VDC can be accessed by an external device (current limited to ____ ma. max.)

If the internal power regulator circuit is not installed, then a +5 regulated voltage can be supplied to the unit at Pin #10.

RJ-45 Pinout



#1 Ring Indicator (RI)
#2 Data Carrier Detect (DCD)
#3 Request to Send (RTS)
#4 DCE Ready/Data Set Ready (DSR)
#5 Transmit Data (TD)
#6 Receive Data (RD)
#7 Signal Ground/Common (GND)
#8 Clear to Send (CTS)
#9 DTE Ready/Data Terminal Ready (DTR)
#10 DC Power I/O (jumper select)