

## What is Serial to Ethernet?

Devices that convert data between the serial and Ethernet interfaces allow engineers to take advantage of the best of two worlds. The "serial world" is the world of sensors, actuators, modems, and basic RS-485 networks that transmit data between devices and computers. The "Ethernet world" is the world of NICs (Network Interface Cards), the Internet, and open protocols used to whisk information from host to host.

The transformation between the serial and Ethernet interfaces takes place at the electronic signal and network protocol levels, such as in the transformation of data from the RS-232 format into a format suitable for a TCP/IP network. Carrying out such a transformation requires Device Server technology, in which a Device Server is a smart, standalone device with an embedded operating system and CPU that is large enough to contain its own operating system and the requisite software protocols, such as the TCP/IP stack. A Device Server also comes equipped with the required hardware interfaces, such as RS-232, RS-422, and RS-485 ports. The device server can transfer, and even process data between the serial and Ethernet interfaces to carry out pre-defined tasks.

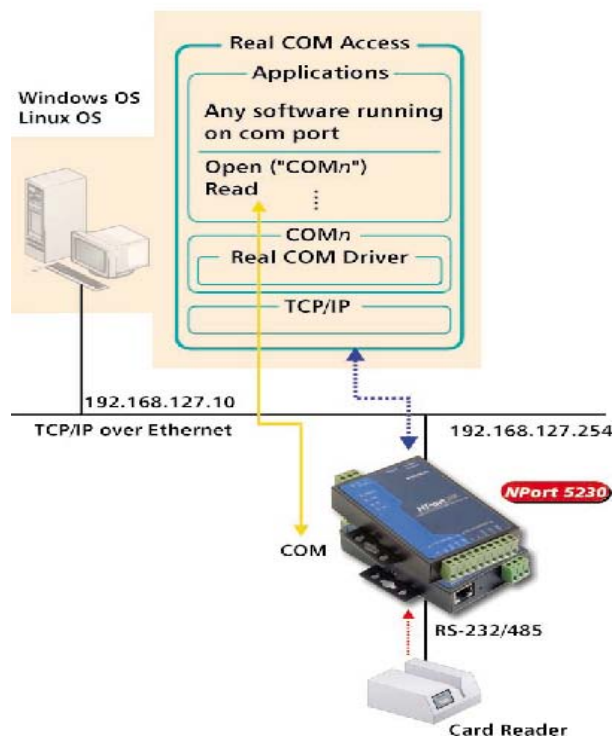
Ethernet, and in a broader context, the Internet, have been adopted by the automation industry as its main communication backbone. By using open technologies, such as TCP/IP and Ethernet, control devices can be networked more quickly and with greater ease. Since most LANs are also connected to the Internet, this type of serial-to-Ethernet solution gives devices a data transmission outlet to any Internet host in the world.

## How does Serial-to-Ethernet device work?

A Serial-to-Ethernet device, also called Serial device server, can use various operation modes to communicate. The following section discusses the options include operation modes that use as driver installed on the host computer, operation modes that rely on TCP/IP socket programming concepts, and the operation modes that typically involve communication between pairs of appropriately configured device servers.

### Real COM Mode

Most serial device server manufacturers provide native Real drivers, called port redirector, that work with Windows operating systems, and many also provide fixed tty and real tty drivers for the Linux and Unix operating systems. The driver establishes a transparent connection between host and serial device by creating a virtual local COM/tty port on the host computer for each of the serial device server's serial ports. Its primary function is to intercept what the computer thinks data intended for its serial port, and redirect the data through the computer's Ethernet card.



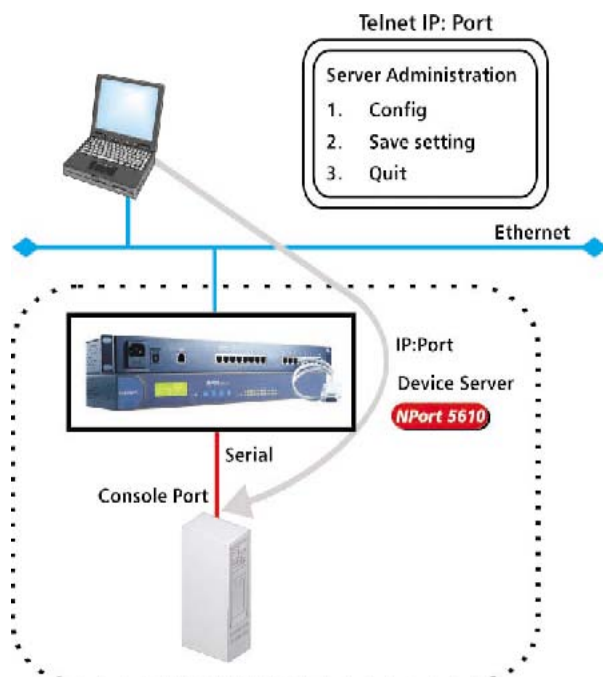
The figure shown here nicely illustrates a typical scenario of a device server configured for Single-Host application.

## Socket Mode

The second class of operation mode, referred to as Socket Mode, provides a way to directly access serial device servers over TCP/IP networks, but without first installing a driver. To control serial device servers directly over the TCP/IP network layer, knowledge of basic concepts related to TCP/IP networks, such as TCP, UDP, IP, Netmask, and Routing is a must.

## TCP Server

In TCP Server Mode, the serial device server acts as a network agent for the serial device. For example, when a serial device (e.g., a file server) which has a serial console port connects to the serial device server under TCP Server mode, the console port becomes a network-accessible point by way of the serial device server. In TCP Server Mode, the serial device server can accept a TCP connection from the control host, and then provide dual direction transmission between control host and serial device.



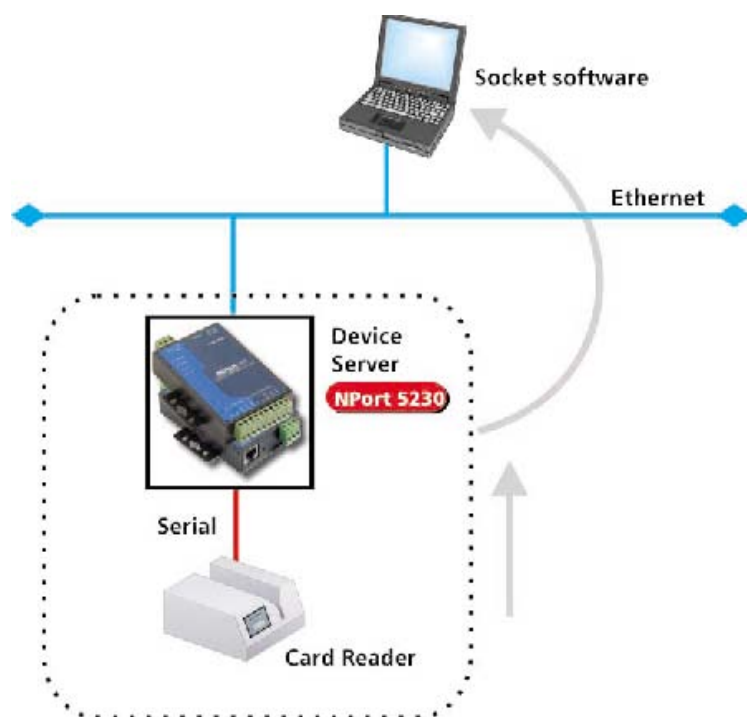
from a host located on the same LAN, or on the Internet if the LAN is

Consider the example illustrated in the left figure, in which a serial device server is connected to the console port of a file server. In this case, the file server could be part of a large server farm, or housed in a remote location not easily accessed by the system administrator. By connecting the serial device server's Ethernet port to a LAN, the system administrator gains access to the PC's console management features

connected to a public network.

## TCP Client

The TCP Client Mode of operation is designed for serial devices that need to actively establish communication with a server program located on a different network device. In this case, the serial device server running TCP Client actively establishes a TCP connection with the server software. After data from the serial device is transferred to the server software, the serial device server automatically disconnects the TCP connection. While the connection is active, the server software is also able to send data to the serial device via the serial device server. You could say that TCP Client achieves the goal of *Connect-on-Demand*, a useful feature for hosts that must deal with so many serial devices that the number exceeds the maximum simultaneous TCP connections allowed.



As an example, consider the card reader shown in the above figure.

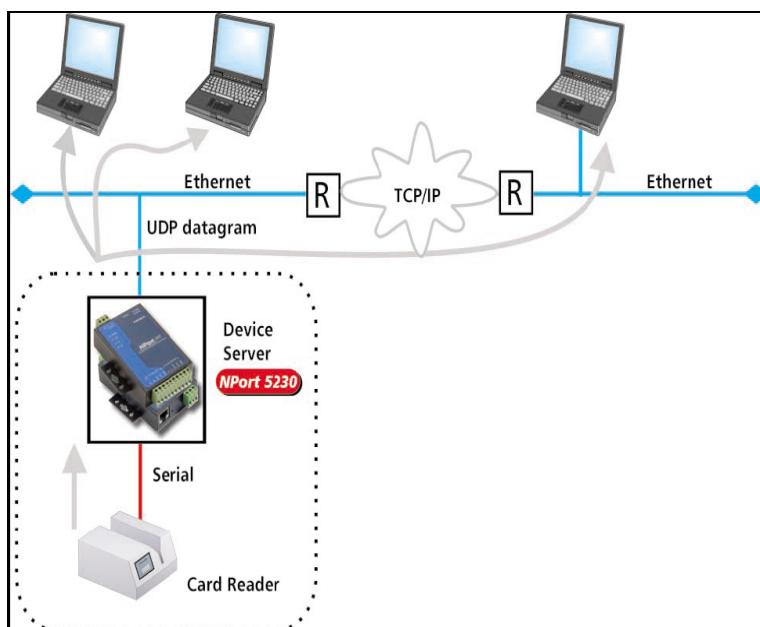
Depending on its specific function, a card reader is generally used sporadically over any 24-hour period. Once a card is swiped, the stored information is transformed into the appropriate serial signal, which is then sent to the device server via the device server's serial port. The

device server, which would have already been set up to send data to a

particular server application on a remote PC or device, forms the serial data into a TCP packet, requests a connection with the remote application, and then sends the packet across the Ethernet to the remote host.

## UDP Server/Client

The UDP Server/Client Mode of operation is designed for applications that require speedy data transmission over UDP protocol layer. By using serial device servers with the UDP Server/Client feature; your serial device can deliver data to multiple destinations at almost the same time. Thanks to the speedy nature of UDP and the Data Packing feature,



the serial device server turns traditional serial devices into powerful network-enabled devices. It is suitable for connecting input devices, such as scanners, card readers, fingerprint readers, and optical scanners.

As an example, consider the situation illustrated in the above figure. In this

case, the application dictates that card reader data should be sent to more than one host, all of which are connected to the Internet. The way to enforce such a requirement is to configure the device server connected to the card reader for UDP Server/Client mode, and then set up the system so that data are sent to each of the hosts.