

Introduction to DVT's Ethernet Communications

Introduction

One of the major enhancements incorporated into DVT's new hardware platform, the Series 600, was Ethernet. During set up time, this new feature allows faster image update rates to the PC running FrameWork's user interface. During service, the Ethernet connection allows DVT to share information with other devices on the factory floor without the limitations of serial communications.

Protocol

The protocol supported by DVT's Ethernet connection is TCP/IP. Each DVT SmartImage SoftSensor has an IP address that identifies it uniquely on a network. For details on how to configure it and the physical media used, see the Series 600 *Getting Started Guide*.

A DVT SmartImage Sensor acts as a TCP server, which means that another application ('the client') must establish the connection and request data from it. This does not mean that the communication is one way. Once the connection is established, the server and client exchange data in both directions. In this case, the client application is usually FrameWork's user interface which runs on a PC equipped with an Ethernet card. It connects to the SmartImage Sensor and initiates a Framework session. Notice that a Framework session implies two way communications; the PC receives images and inspection data, while the SmartImage sensor receives system commands and parameters such as exposure time and threshold. However, as will be seen later, other applications different from Framework's user interface can act as the client.

IP Addresses, Ports and Sockets

In order to establish a connection, a client application must be directed to the correct IP address, but also to the correct TCP *port*. In TCP/IP protocol, a TCP port number is used in conjunction with the *IP address* to identify a specific application running on a network node (computer or Ethernet capable device). A unique communications 'path' or 'pipe' is defined by the combination of an IP address and a TCP port number. This is called a *socket*. A *socket* tells the client application exactly where to look for its information. Consider that a network node identified with one IP address can use multiple TCP ports at the same time, that is, several sockets can be established between it and other nodes. Some common network applications that use TCP/IP have 'well-known' port numbers assigned to them. For instance, FTP uses port 32 and Telnet uses port 23.

The DVT SmartImage Sensor uses two TCP ports: 5000 and 5001. The function and data types are listed on the following table:

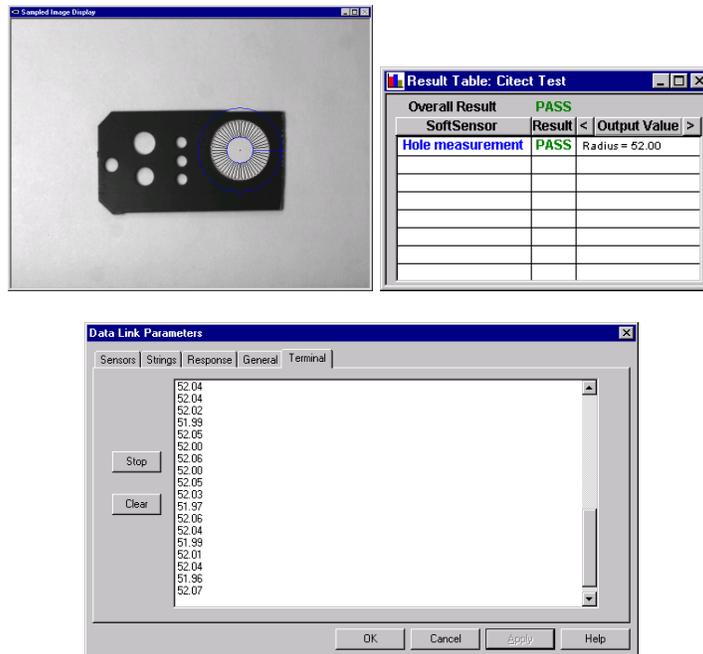
Port Definitions			
Port	Type	Data	Description
5000	TCP	ASCII + Binary	Used for system-specific communications between the SmartImage Sensor and Framework's User interface on the PC.
5001	TCP	ASCII	Used to transmit DataLink inspection Data

Figure 1 TCP Port Definitions.

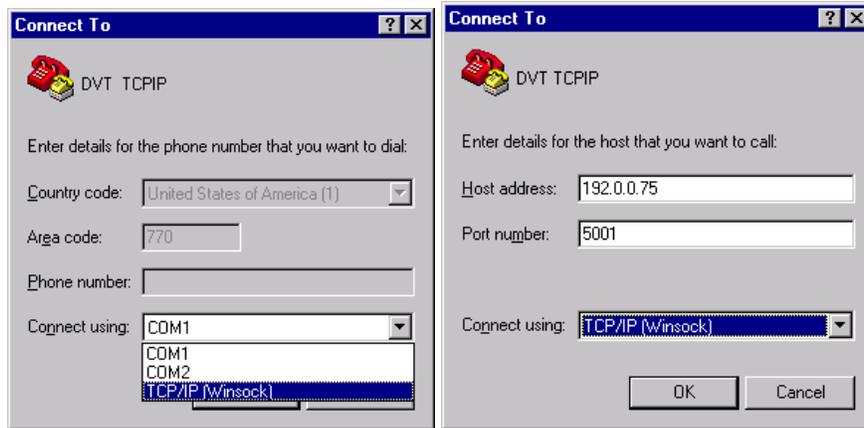
When Framework's user interface connects to the SmartImage Sensor, the system commands and images are exchanged through TCP port 5000. When a particular SoftSensor is configured in DataLink to output inspection results as ASCII strings, these are transmitted through TCP port 5001. The ASCII strings are placed inside the TCP/IP packets without any additional protocols just as if they were being transmitted over a serial line.

Before the Series 600, it was common to use Windows HyperTerminal to verify the serial connection and correct DataLink configuration. The new version of HyperTerminal that comes standard with Windows 98 supports TCP/IP and can be used for the same purpose. The only difference is that instead of specifying a COM port and serial parameters such as Baud Rate and parity, the user specifies an IP address and a TCP port. The following example shows how easy it is to use this tool.

1. Set up a precision measurement inspection to output the result over DataLink.



- Open HyperTerminal and start a new connection. Choose **Connect using:** TCP/IP Winsock. Next click **Ok** and write the IP address of the SmartImage Sensor as the **Host Address**. Also write 5001 as the **port number** (default is port is 23 corresponding to Telnet). Click Ok and connect to see your data.



This example shows how a client application (different from Framework) can connect to SmartImage Sensor and obtain data over the Ethernet TCP/IP connection.

Protocols on top of TCP/IP?

The combination of Ethernet's physical media and TCP/IP protocol is responsible for transporting information packets between nodes in a network. However, the information inside those packets can be formatted in many different ways. This means that two devices supporting TCP/IP over Ethernet will not necessarily communicate with each other. DVT's uses straight ASCII characters inside the TCP/IP packets. But other devices such as Ethernet PLC's or I/O modules may use other data types and formats to present their information; for instance, they may use binary data with special structures to identify source and destination information inside a PLC's memory.

DVT	IP	TCP	ASCII DATA	TCP	IP
Some systems	IP	TCP	OTHER Protocols	TCP	IP

Figure 2 Schematic of TCP/IP packets carrying information with different formats.

To ensure successful communication with DVT over Ethernet TCP/IP the client device must be able to interpret ASCII data. DVT is studying the possibility of supporting other common protocols over TCP/IP to allow seamless communications with even more devices and software packages present on the factory floor.