Application Note AN2013-02

Using the RTAC as a Modern Serial Data Line Analyzer

Chris Bontje

INTRODUCTION

Remote retrieval of industrial process control information (digital contacts, analog data, metering, and so on) has been accomplished by predefined supervisory control and data acquisition (SCADA) protocols since the late 1970s. A typical SCADA protocol is formatted to use a standardized 8-bit serial exchange (e.g., 8 data bits, no parity, 1 stop bit) that can operate over a leased line, dial-up modem, dedicated multiplexer serial circuit, and a variety of other communications methods. As Ethernet technology has become more prevalent in the industrial control systems world, several SCADA protocols have been adapted to operate under Ethernet environments, in addition to their serial-based equivalents.

Many of these SCADA protocols are traditionally proprietary and manufacturer-specific. This requires customers to use compatible equipment for data exchange, which is typically only available from one or two manufacturers. However, in the last 10 years, many manufacturer-neutral protocols have gained widespread use and have somewhat standardized SCADA data exchange between equipment from a wide variety of manufacturers. Common manufacturer-neutral protocols today include DNP3, Modbus®, and IEC 61850.

Whether dealing with a standardized or proprietary SCADA protocol, the need has always arisen for some form of in-line debugging capabilities that offer a programmer or commissioning engineer a look into the protocol data stream between the SCADA system and the end device. For newer Ethernet connections, a solution is an available class of computer software known as Ethernet packet capture utilities (Wireshark® is an example), but a different solution is needed for serial data streams. The actual device that offers this capability for serial channels is referred to as a serial data line analyzer, and this application note outlines both a traditional and a modern solution for this type of device.

TRADITIONAL SOLUTION

Many hardware manufacturers produced serial data line analyzers in the past decades and had great success marketing and selling them. A commonly available analyzer was the Hewlett-Packard 1640B Serial Data Analyzer. A device such as this offered a wide variety of features for allowing the capture and analysis of serial data streams, such as the following:

- In-line connection capabilities provided by dual EIA-232 connection points.
- Selection of a variety of serial data rates and bit formatting.
- Display of state of serial control lines (RTS, CTS, DTR, and so on).
- Receive and transmit data output on the provided internal monitor.

Even with all the capabilities offered by these devices, they still proved cumbersome to use in practice due to the difficulties in manually recording and decoding the data stream provided by the monitor output. As SCADA protocols gained in features and complexity over time, the actual process of debugging even a simple data exchange could take several hours as an engineer or technician manually stepped through each recorded byte and interpreted its meaning. Debugging an entire data stream of multiple exchanges could take days or weeks. In addition, if a non-EIA-232 connection was needed for the intelligent electronic device (IED) connection (e.g., EIA-485), an in-line converter had to be used, adding extra complexity and setup.

MODERN SEL SOLUTION

The SEL-3530 Real-Time Automation Controller (RTAC) and related RTAC products offer a modern solution for a serial data analyzer with many added benefits. The RTAC family of controllers (see Figure 1 below) can be ordered with various options, allowing for 4, 17, or 33 EIA-232 or EIA-485 dual-mode serial ports. Each serial port is completely configurable for direct communication with various protocols (SEL, DNP3, Modbus, and so on), as well as for access point functionality that normally allows tie-in of external connections (serial or Ethernet) for engineering access or other similar functionality. The RTAC allows for a web configuration interface for basic parameters such as Ethernet addresses and security accounts, but the majority of all RTAC configuration settings are adjusted via the ACSELERATOR RTAC® SEL-5033 Software.

Figure 1 RTAC Family

Normally, any configured access points in the RTAC are used along with an access point router to provide a maintenance connection to a downstream-connected SEL protocol client device, such as a relay. However, by using a pair of access points (configured for EIA-232 serial) along with an interconnecting access point router, the RTAC provides an in-line connection between two external devices, while adding the ability to monitor each byte that passes through the access point(s). Figure 2 shows an example RTAC configuration with the pair of serial access points connected by a single access point router.

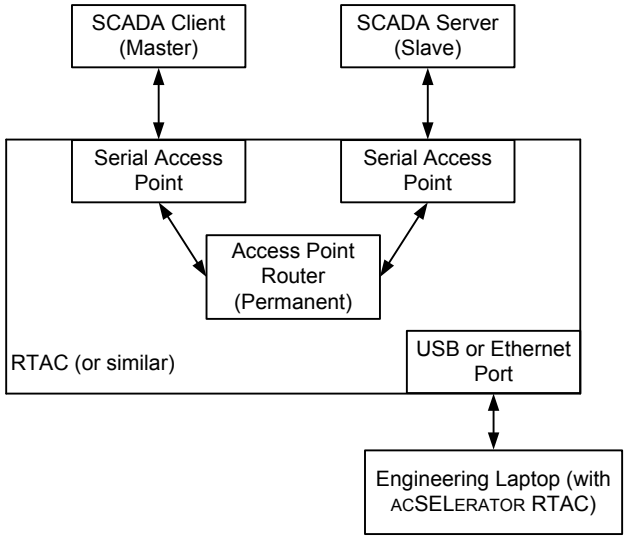

Figure 2 Configuration of RTAC Access Points

Once the connection has been established between the SCADA host (master or client) and IED (slave or server) with the RTAC in-line with access point pass-through configured, the communications monitoring tool that is built into the ACSELERATOR RTAC software can be used. This communications monitor will provide a raw output of the data bytes across the access point channel with the state of the various serial control lines, and the entire stream can be saved into a common file format known as pcap (packet capture). Figure 3 shows an example of the ACSELERATOR RTAC communications monitoring tool.

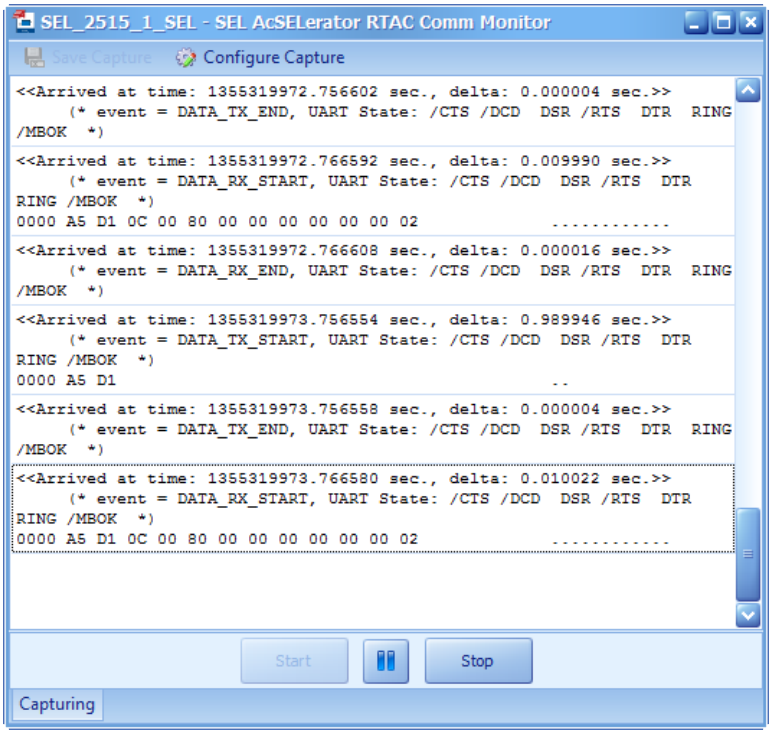

Figure 3 ACSELERATOR RTAC Comm Monitor

Once the pcap file is saved on the local hard disk drive, an external pcap analysis tool, such as the Wireshark utility, can be used to rapidly display all transmitted and received data in raw byte form with relative time stamps and decode the standard SCADA protocol data (if a compatible

decoder is present). A tool such as this allows for the rapid analysis of an entire serial data exchange, as well as quick debugging of faulty protocol exchanges by providing automatic decoding of supported data protocols. Wireshark supports a variety of standardized protocols such as DNP3, serial Modbus RTU and Modbus TCP, and IEC 61850 Generic Object-Oriented Substation Event (GOOSE) and Manufacturing Message Specification (MMS). Figure 4 shows an example of Wireshark decoding a DNP3 serial data exchange between a server and client.

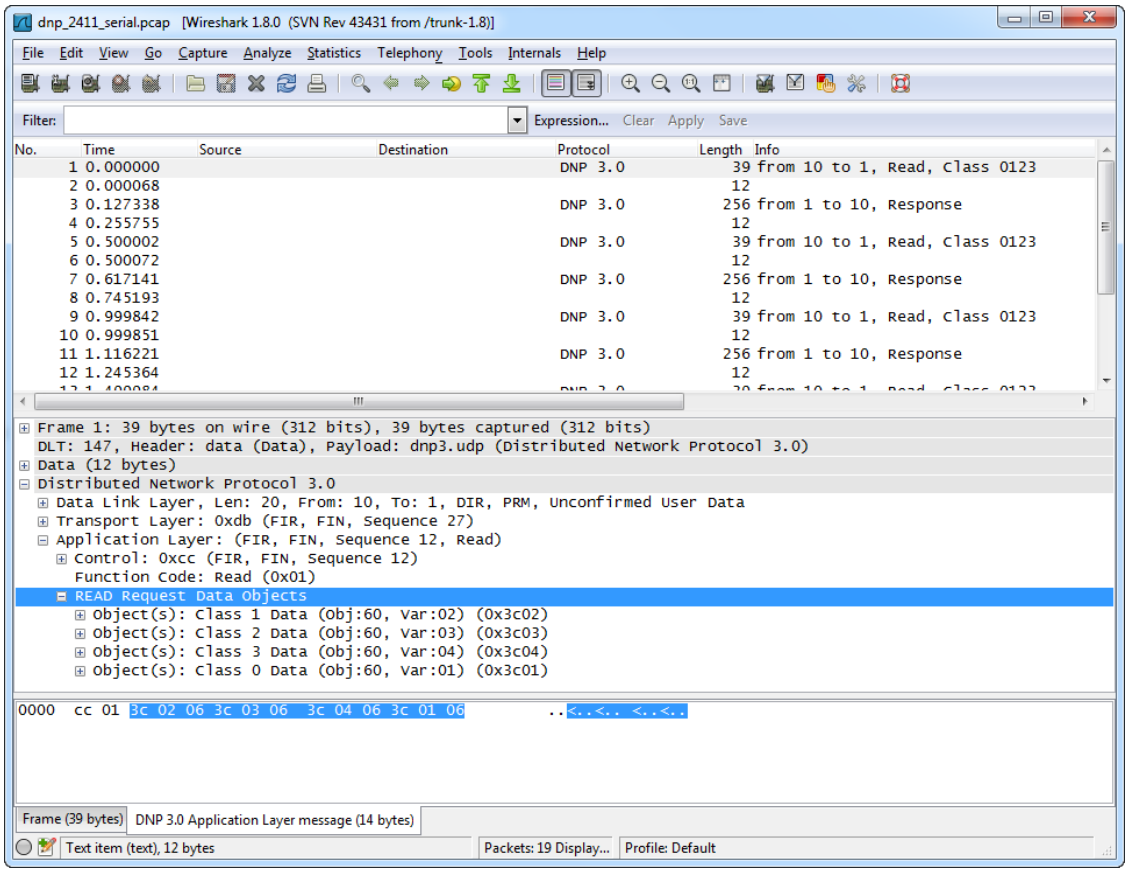

Figure 4 Captured DNP3 Protocol Data Wireshark Display

CONCLUSION

The RTAC provides a number of built-in functions that facilitate easy debugging of external devices communicating via serial data streams.

LAN2013-02

SCHWEITZER ENGINEERING LABORATORIES, INC.
2350 NE Hopkins Court • Pullman, WA 99163-5603 USA
Tel: +1.509.332.1890 • Fax: +1.509.332.7990
www.selinc.com • info@selinc.com