CORSAIR

~ Applications Note ~

Modbus Ethernet to Serial Conversion

The Corsair program can communicate with several types of devices using the Modbus protocol. It has drivers to do this over Ethernet or over a serial connection. The serial connection can be RS-232, RS-422, or RS-485 multi-drop.

Many times several inexpensive devices are hooked together to form an RS-485 multidrop system. They are Modbus slave devices that are monitored by the Corsair program. An inexpensive converter is used to hook them to an RS-232 port on the computer. The disadvantage of this approach is that only one Corsair computer can see the devices. A common answer to this problem is to install a Modbus Ethernet to serial converter. The converter has an Ethernet port and an RS-485 port to go to the devices. Each Corsair computer uses its Modbus Ethernet driver to talk to the converter. The converter sends a data packet down the RS-485 line, gets a reply from the slave device, and then replies to the computer over Ethernet.

There are some configuration items that need to be set into the Ethernet converter. One is the 4-byte IP address that the Corsair computer uses to locate the converter on the Ethernet network. Some of the characteristics of the RS-485 serial side that need to be configured include baud rate and parity.

Usually the configuration of the converter does not involve entering information about the individual multidrop slave devices. Each of the devices must be configured to have a unique slave address number ranging from 1 to 247. The Ethernet packet that the Corsair computer sends to the converter includes a one-byte ID number. This number is used for the slave address when the converter sends a Modbus packet to the slave devices. Each device only replies to Modbus queries that match its slave address number.

There are two serious implementation problems with Modbus Ethernet to serial converters that make them unusable in many systems. This application note is to discuss the Corsair approach to solving these problems.

Problem #1 - Socket Count

Ethernet TCP/IP devices can only establish a limited number of TCP socket connections at one time. This number is especially low with inexpensive devices like Modbus serial converters – it seldom is over 32 and frequently it is less than 10 and sometimes only one. This establishes the maximum number of computers that can talk to the converter at one time.

Unfortunately the socket count limitation presents even more problems. Many interface programs require a single unit ID number to apply on all the data tags on a PLC record. A separate PLC record must be entered into the database for each device on the multidrop

serial line. The problem is that the interface program thinks that each PLC is a separate device on the Ethernet network. It makes the assumption that it can't talk to all of them at the same time so it tries to open a separate socket to each of them even though they all have the same IP address. This results in disaster. If the converter only supports 10 sockets the system can only have 10 devices on the multidrop system. The problem is compounded when more than one computer tries to connect to the converter with the same software. The converter must handle a number of sockets equal to the number of computers multiplied by the number of serial devices. This is impossible for all except the smallest of systems.

A possible answer for this problem is some sort of specialized data buffering software running on a separate computer that takes the place of the converter. The small inexpensive converter has been replaced by a larger computer with a hard drive, some software, and an RS-232 to 485 converter. The software has to be reconfigured as devices are added to the multidrop system. This approach is an expensive nuisance that requires continuing support.

The basic Corsair Modbus Ethernet driver has this problem. It will try to open a separate socket for each device. A specialized driver that is also included with the program helps to solve the problem. It utilizes the fact that Corsair will only try to initiate one communication at a time with a PLC over Ethernet, although it will try to talk to several PLCs at once. The specialized driver incorporates the Unit ID address as a part of the address of each data tag, not globally for the PLC. The Corsair program thinks that all the slave devices are a single PLC. It only opens up one socket to the converter for all the slave devices and it only tries to talk to one at a time. This matches the capability of the serial port.

With the specialized Corsair driver a converter that allows 10 sockets can talk to 10 Corsair computers with no limit on the number of slave devices.

Problem #2 - Update Speed

The next problem with Modbus Ethernet to serial is the deterioration in throughput as computers are added to the system. The serial port is very slow compared to Ethernet. It frequently operates at 19,200 baud or less and inexpensive serial devices may be slow to respond. Only one device can be read at a time. Each computer has to wait its turn.

A system configured with one Corsair computer will have a definite update time for all the data to be read into the Corsair computer. When another computer is added the update time is doubled. It goes up for each computer that is added. Ten computers will operate at one-tenth of the speed of a single computer. The update speed becomes unusable for many systems with more than two computers.

The solution to this speed problem is for the converter to buffer the data. It reads in data from each slave device into memory in the converter. The converter then satisfies read

queries from the Corsair computers from its own memory at Ethernet speed. The throughput is no worse with ten Corsair computers than it is with one.

A standard converter will not perform this buffering function. It appears once again that the answer is a separate computer to replace the converter. It will require specialized (probably custom) software. The software will operate from a polling table that tells what slave addresses are present and what data needs to be read from each one. The polling table configuration will have to change as devices are added to and subtracted from the multidrop system. This is another expensive nuisance requiring continuing support.

The solution to this problem is the special Corsair converter. It is a polling converter that buffers data into its own memory for fast Ethernet update speeds. The thing that makes the Corsair converter unique is that it automatically 'learns' its polling table from data read requests that it receives over Ethernet. The Corsair converter requires very little configuration. It is smaller than a computer and has no hard drive. It costs far less than a computer with custom software.

A functional specification for the Corsair converter is available on the Corsair website.

Conclusion

The combination of the specialized driver contained within the Corsair software and the Corsair converter makes it possible for Ethernet to serial systems to perform as efficiently as the remote device serial port throughput will allow. This is done with a minimal amount of configuration work and without a separate computer to act as a buffering converter.

Corsair HMI Software
1-800-532-2477
www.AdvIndSys.com/CorsairSoftware.htm
Corsair@AdvIndSys.com