

PLC Communication Options for Electronic LED Displays

Introduction

This information document has been prepared to help you understand the different PLC communication options that are available with LED displays. Primarily this document will focus on the LED displays manufactured by Electronic Displays Inc. We'll also discuss, with some examples, on how to communicate to these displays.

Each display can either be communicated to by two general methods - parallel methods or serial methods. These are described in detail below.

Parallel Methods

These methods use discrete outputs to drive the information on the displays. Typically these methods are used for simple ANDON systems with cell lights, or numeric-type displays. Alpha-numerics can be used, but are usually setup with "predefined" messages that are triggered by these discrete outputs.

Direct PLC Outputs

Using several PLC discrete outputs directly from the PLC (usually 24vdc source outputs, but other combinations are possible, including 120vac outputs and 24vdc sink), you can control both the andon cell lights and numeric displays.

The benefits using this method are that simple PLC's can be used, and programming is quite simplified. The downfalls are that wiring could become quite extensive and that distance may have a limiting factor.

Andon Cell Lights

Since these lights are simple indicators (red, green, amber, or similar), a PLC output that is turned on simply lights up that cell. Quite easy to program by using output coils in ladder logic type programming.

Numeric Displays

These require several inputs to drive the numbers, usually as a BCD (binary-coded decimal). Using BCD requires only 4 inputs. Each input would have a "weight" - 8, 4, 2, and 1. Summing these weights when each input is "on" will display that number. Since a display could contain many digits, there a couple of varieties to drive these

digits:

- BCD Inputs, NO Strobe - This type of input simply has the 4 weight-inputs assigned to each digit. So a display with 4 digits would require 16 outputs from a PLC (4 for each digit).
- BCD Inputs, With Strobe - This type of input add another discrete input per digit of display. This extra input is a strobe. This allows you to "common" the weight-inputs all together, and then use the strobe input to show that digit. For a display with 4 digits, this would require 8 outputs from a PLC (4 for the weights, and one strobe for each input $\times 4 = 8$). Operation is simple - turn on the weight outputs to select the desired number to display, then apply output to the strobe for that particular digit.

PLC programming for above would usually consist of a register with BCD numbers that are simply "written" (MOV) out to the output coils that are mapped in a BCD fashion (4 coils aligned properly and contiguous). Also, discrete coils can be used along with DIV/MOD instructions, but usually PLC MOV instructions are easier.

Remote I/O Outputs

This is an additional option that basically extends the outputs of a PLC through remote i/o devices. These devices have a network/serial type interface on one end with the outputs on the another. Typically these devices can be located right inside the LED display - and are available as several different types. This include DeviceNet, Profibus, and ModBus.

The PLC would have to be setup typically as a master and include appropriate network module needed. The PLC would also have to map internal output coils to each of the "slave" devices attached to the network.

The benefits of using this method are that less wiring is needed (only for the network), and that many more outputs are available and not required to be directly installed on the PLC. Not to mention that in some cases, the network can be made wireless. The downfalls are that setup is a bit more complicated within the PLC and extra cost associated with modules and devices.

Serial Methods

These methods use a communications protocol over either a RS-232, RS-422/485, or Ethernet transport. These methods are used for mostly numeric and alpha-numeric displays. They allow for the most control of any of the methods.

Wireless may also be used. In that case, it is usually taking one of the 3 methods above and selecting devices that overlay that method over wireless. There are

several different manufacturers of wireless equipment to fit that need. Real-time response is not usually required with LED displays, so simple wireless equipment is all that is needed.

RS-232 Communications

This specific serial method requires the use of a serial port located on the PLC. This can either be a serial module, or can be shared with the programming port, if possible. Typically this port is setup to 1200 baud, no parity, and 1 stop bit. The setup is usually done with configuration registers/coils within the programming of the PLC, or via a specific instruction.

With RS-232 communication, usually it's a point-to-point communication, that is - only one plc can drive one display. Also, you are limited as to the distance you can run the wires - typically 50ft. The best cable to use is 18ga 3-conductor, shielded wire.

The benefits using this method is easy of use. The downfalls are that the distance is limited, and there is some additional programming required in the PLC.

The protocol used is a simple, ASCII-type protocol. ASCII simply means any character with a code from 0 to 127 (refer to an ASCII chart for details). The protocol may also contain "addressing" - in which each display is given a unique address of 01 to 99 or 001 to 255. This allows for the opportunity to drive several displays from one serial port from the PLC (although this is not recommended for RS-232, RS-422/485 is better suited - see below). For numeric displays, the protocol is <STX>AANN<ETX> - where <STX> is the ASCII code of 2 (like pressing Ctrl-B on your keyboard)- AA is the ASCII code representation for the address - NN is the ASCII representation for the number to display, and <ETX> is the ASCII code of 3 (Ctrl-C on your keyboard). Since these are ASCII codes - to send the number 1234 to address one - you need to send the codes (in decimal notation) - 2, 48, 49, 49, 50, 51, 52, 3 - Note that the numbers are represented by ASCII codes (48, 49, 50).

Additional protocols are documented and supplied with your display - for displays that contain cell lights or alpha-numeric LEDs.

Programming PLC using ladder logic: You'll need to setup a range of registers for your message. Each register may contain 2 characters, so the first one may be setup with the start code (2) followed by the first digit. Typically you'll have an instruction to convert numeric data to "string" type data into a series of registers. They you will have to move these, in the proper order for the protocol above, to your message registers. Then there will be a special instruction to send the data out the serial port.

If you have trouble communicating from your PLC, the first step is to make sure the display works by using a laptop with Hyperterminal - and manually type the message in (Ctrl-B 011234 Ctrl-C as in the example above). If that works, you can then hook up Hyperterminal to the PLC and have the PLC send the string. You can verify it by looking at the result. Most problems occur due to issues of the proper sequence of characters - or by trying to send out actual numbers, instead of the ASCII codes that they represent.

RS-422/485 Communications

This specific serial method requires the use of a serial port located on the PLC specifically for RS-422/485. This can either be a serial module, or can be shared with the programming port, if possible. Typically this port is setup to 1200 baud, no parity, and 1 stop bit. The setup is usually done with configuration registers/coils within the programming of the PLC, or via a specific instruction.

With RS-422/485, you can control multiple signs (usually up to 32). The distance is useful out to 4000ft. It's also very important that each end of the link be terminated correctly. The best cable is 18ga twisted pair (1 or 2 pairs, depending if you need to receive data), shielded.

RS-485 sometimes differs in that it may only use 2-wires to communicate both back and forth. LED displays by Electronic Displays support both 422/485. Also, RS-485 may require "bias" to have each wire go to a known state. Please check your serial module for details. Many issues have occurred due to the result of not having bias enabled on a serial device.

Other than the physical transport, there is no difference in the protocol or programming from the RS-232 method above.

Ethernet Communications

This specific method requires the use of a general networking module, or functionality built-in to the PLC. Please note that the network module must support general TCP/IP communications on top of Ethernet. LED displays do not currently support proprietary protocols from A/B (blue hose), modbus plus, etc. LED display also only support 10/100 mb/s interfaces via RJ-45 connectors.

The benefits of using this method is that you can use your own Ethernet network (bandwidth is very low). The downfalls are that it may require even more programming and setup.

With Ethernet, the protocol used in RS-232 above is overlaid upon a network connection that is made from the PLC to the display. Each display not only must have it's own address as described in the RS-232 section above, but must also have it's own network address (it's IP address). Using the combination of these two, you will be able to select the correct display. There is a utility that is provided to set the ip addresses on the LED displays.

The displays listen to a specific "port" on the network (usually port 23). Once they receive a connection, then they will listen to the protocol. Then the connection can be terminated.

Typically, with a PLC - you'll have an instruction that allows you to connect and send a string of data out. The format of this data is the same protocol/series of registers as you would use in the RS-232 section above. With this instruction, you'll have to identify the IP address, what transport protocol (TCP), and the port number (23), along with additionally timeout values. There also may only be one instruction to open a connection, of which when successful, the output drives another instruction to

send the data.

If you have trouble communicating with the PLC, you can verify the display by using a laptop, and running telnet - connect to the ip address of the sign and type in the protocol as indicated above. If you cannot connect still, try running ping on that ip address. You may have to verify your network settings are correct with your IT department.

Conclusion

As you can see, there are a variety of ways to have your PLC communicate to the LED displays. You can make your selection of interface when you go to order any of the displays and have confidence that you will be able to talk to them. As we encounter different PLC's, we'll update this document with even more detailed examples. There have been many people using the methods above to successfully communicate with their displays.

Please feel free to contact us for any additional feedback.