4-20ma - RS232 Rx

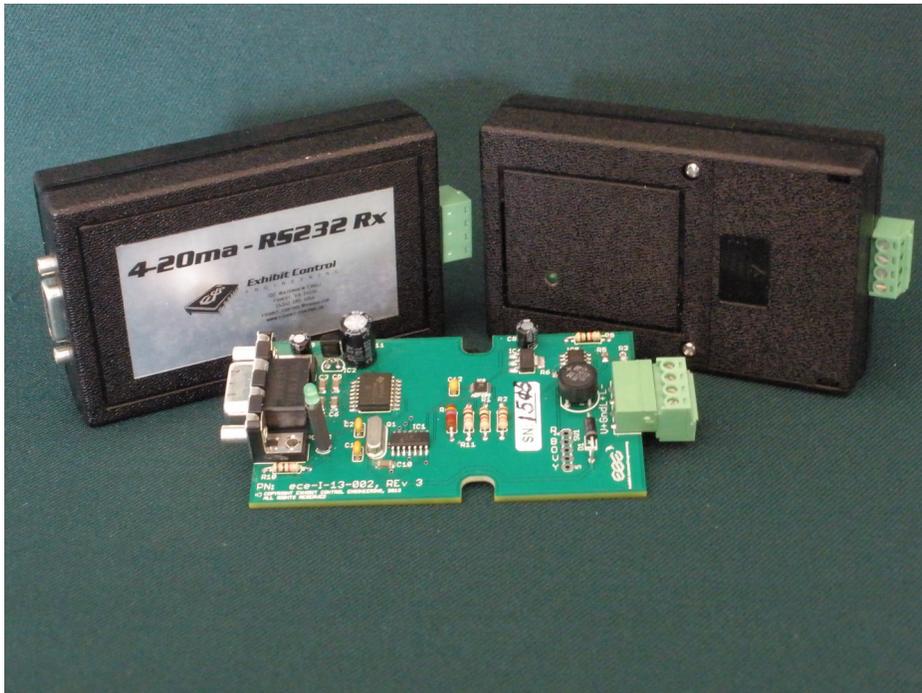**PN: ece-I-13-002, REv 3**
Firmware: Ver 1.0

Figure 1

Technical Details:

This product is for those instances where you need a non-PLC device, like a PC, NetLinx Master or Crestron Master with a serial port (UART or USB) or the like to communicate with a PLC by receiving a 4-20ma current loop signal. There are many sensors designed to provide a 4-20ma current and this device allows a device with a serial port to decipher that signal.

The protocol is quite simple. First it uses 9600 baud, no parity, 8-bits, 1 stop bit and no flow control. We provide this device in two basic versions. In both versions, the strings are simple ASCII. In one form, it provides the raw value from the ADC which results in a number from 0 to 65535 in ASCII format followed by a carriage return (0x0D). In the second configuration it returns the characters: "I = " and then the current value in milliamps followed by a carriage return (0x0D). Example: "I = 12.345<0x0D>". In the current reporting version, it is accurate to within +/- .02ma throughout the entire range of 2.00 to 20.00ma in an ambient temperature of 70ºF. In the range between 4.00 to

20.00ma at 20°C, it is generally more often as accurate as +/- 0.005ma in an ambient temperature of 70ºF. How or when the device will send out the current value depends on the mode it is in, described in the next paragraph. In the power up default mode, it will send out the current reading every time it senses a change in current at its most sensitive setting. These detections are very sensitive (0.002mA), hence the addition of the other modes of operation. If you send the device an ASCII "?<0x0D>", regardless of the mode, it will always respond with the current value.

With Ver 1.0 of the firmware, we have added additional operating modes that are user selected via the RS232 communications. The default milliamp current change reports are just as described in the preceding paragraph. ECE recommends this form of operation, as it has the least pit falls. We suggest that the logic in the report receiver's program take each report when it occurs. If the value is not one of interest, ignore it and wait for the next. This way you will always know exactly what the loop current is as soon as it changes. However, for the those instances where a loop current generator has excessive noise that results in a steady stream of current reports and one does not wish to deal with them in the report receiver, we now offer several other modes to ameliorate this problem. Regardless of the mode selected, you can always poll the device for the immediate current value.

The three modes are: Change in Current Reports with an optional value for plus-to-minus delta changes (the default mode uses a value of one for the delta change requirement), Polling Only Reports and Periodic Reports of a changed current reading rather than immediate change reports. The first mode is exactly as described in the paragraph above as the default except you can specify a delta change in milliamp current that must be exceeded, either negatively or positively, before the device reports a change. This option enables the device to integrate with "noisy" current loop generators that cause the device to issue a steady stream of reports of inconsequential changes in the loop current (you can filter out up to +/-0.099 milliamp changes). The second mode will only provide reports when polled. And, the last mode will use a periodic time (3-60 seconds) to see if there was a change in current from the last report and report it if a change was detected. This mode uses the same delta requirement as the default mode (+/-0.001ma). The following command strings are used for each mode:

1. Change in Current: This is the default mode upon power up of the device.   However, if you change the mode after power up and want to reset it to this mode later, or want to change the sensitivity, you will send the following command: "C??<0x0D>" where the "??" is a two numeral string from "01 to "99" which specifies the amount of milliamp current in thousandths of a milliamp required to prompt a report. The delta change in thousandths of a milliamp current needs to be a two-character string, so use a leading zero (0) for changes less than ten thousandths of a milliamp. The device will respond with the following string: "*C??<0x0D>". You should set this value as low as possible to ensure you do not miss a current value of interest. Remember this value is a plus and minus value so the swing around the last value reported is double this value.

2. Polling Mode Only: To set the device for this mode, send the following string: "P<0x0D>". When it takes, it will respond with "*P<0x0D>".

3. Periodic Reporting of Current Changes (Timed): To set the device for this mode, send this string: "T??<0x0D>" where the "??" are replaced with two numeral characters 03 to 60, for the time period 3 through 60 seconds. The time period needs to be two characters, so use a leading zero (0) for times less than 10 seconds. When it takes, it will respond with "*T??<0x0D>". The sensitivity to change is +/- 0.001ma.

Please remember that if other than the default mode (with its most sensitive setting of "01") is desired, the desired command will have to be issued every time the device is powered up. It always powers up in the default mode. Also, if the device receives an unknown or incorrect command string, it will respond with a "?<0x0D>". When you receive this string, you will know your last command didn't take and will have to re-issue it or fix it and re-issue it again to get the change you desire. If you would like to query the device for its current mode, you can send the following string Q<0x0D>. The first character of this response, regardless of mode, will be a dash (-) followed by the command string that put it in its current mode. Examples: -C32<0x0D>; -P<0x0D>; -T10<0x0D>.

The serial communication port is a D-SUB 9-pin female socket. It is configured as a DCE. That means you can use a straight through serial cable between it and a standard PC Comm port. In any case, the transmit pin is 2, the receive pin is 3 and ground is pin 5. By-the-by, the device's buffer is rather limited, so don't "hammer" it with lots of communications; pacing your serial strings is in order.

Early versions of the device are not protected from over current conditions, so the user must ensure the loop current never exceeds 80ma. Later versions have a 125ma fuse on the PCB. If it blows it will require soldering to replace. Best advice, don't let the loop current exceed the limits. Generally most over current conditions occur when there is a large capacitor in the loop circuit that causes a surge current when it first charges up upon application of power.

The loop current port is a four-pin Phoenix connector. The 9 to 18vdc power is applied to pins 1 (+V) and 2 (Gnd). Pin 3 is the positive loop connection and pin 4 is the negative loop connection. The polarity of the current loop

connection is not critical; it will work if reversed. Do not use the device's power supply to power the loop current generator as the common ground connection can lead to problems in the device's detection circuit.

There is one LED (green) on the device to indicate when it is powered.

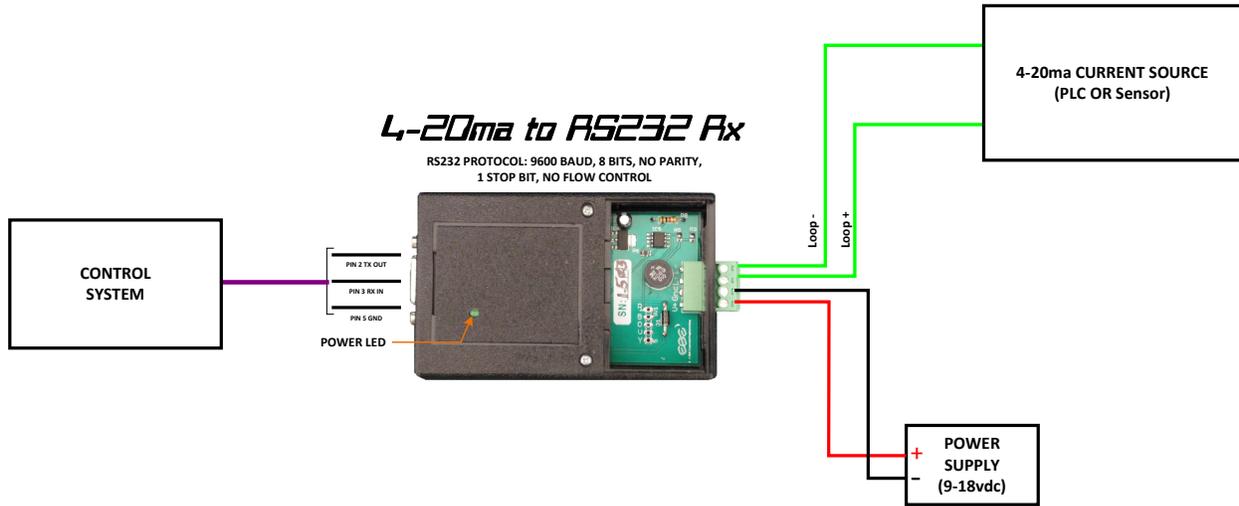See Figure 2 for a typical application:

Figure 2

Exhibit Control Engineering