Introduction

Remote ACCES Distributed I/O solutions provide a quick and simple solution for your data acquisition and control needs. Simply follow the few steps outlined below and you'll be talking to the Pod! This tutorial guide assumes you're using Windows 98, Windows Me, Windows 2000, or Windows XP. Its purpose is to outline the basic steps necessary to establish communications with the Pod, and verify that it is functioning properly.

START HERE

1. Verify your RS-485 Port is Installed and Working
2. Apply Power to the Pod
3. Connect the RS-485
4. Install the Software
5. Talk to the Pod
6. Exercise the I/O

An additional section is provided describing briefly how to use the Pods in operating systems other than Windows.

1. VERIFY YOUR RS-485 PORT IS INSTALLED AND WORKING

   The Remote ACCES Pods use RS-485 communication by default, so it is important your system has an RS-485 port installed and working. If you purchased your RS-485 port solution from us, check your manual or Quick-Start guide for instructions on verifying your port is installed and working.

   The simplest possible port to use is our CONV232/485 converter. This device plugs into one of your computer’s existing RS-232 ports and turns it into an RS-485 port. Refer to the CONV232/485's Quick-Start guide for instructions on verifying the installation and operation of this device.

2. APPLY POWER TO THE POD

   The Remote ACCES Distributed I/O Pods require DC Power to operate. The quickest way to power a Pod during setup and development is from the Computer’s 12Volts. The standard power supplies in computers use a Molex 4-pin connector as shown in Figure 1, below.
The color coding is industry standard. The middle two wires are ground, the yellow wire (shown on the right) is +12 Volts DC power. Make sure the power to your computer is disconnected, then connect the +12 Volts to the screw terminal on the Pod labeled PWR+. Also connect one of the Ground wires to the GND screw terminal. Refer to Figure 1 for details.

We offer several solutions for powering your Pod that don’t require this kind of connection. The **PS120-12** provides a clean 12 Volts DC from standard 120 Volt AC power sources. Alternately, we also offer the **PC-POWER-OUT** cable-bracket kit to bring fused +5 and +12 voltages to a 9-pin (DB-9) female connector installed in a spare slot in your chassis.

3. CONNECT THE RS-485

Now connect the RS-485 port to your Pod.

Pin 3 of the 9-pin connector on the RS-485 port’s 9-pin connector is TRX-. Connect it to the screw terminal on the Pod labeled RX-/TX- or TRx-. Similarly, connect pin 2 of the 9-pin connector to the screw terminal labeled RX+/TX+ or TRx+. If your Pod also has screw terminals labeled TX+ and TX- as shown in the picture below, connect as shown.
4. INSTALL THE SOFTWARE

1. Power up your computer (and if you're using an external supply for the Pod, power it on as well).
2. Insert the Software Master CD into the CD-ROM drive.
3. If the Setup program does not automatically run within a few seconds, double click the “Setup.exe” file in the root of the CD.
4. Install the software package for the Remote Pod you’re using.

These instructions assume you accept all the default selections as shown by the Setup program.

5. TALK TO THE POD

Now that the Remote ACCES Pod is connected and powered, it’s ready to say hello.

1. Run WinRISC.
   A. Select WinRISC from your “Start” “Programs” “[name of Pod]” menu
   B. Make sure the “Local Echo” checkbox is checked
   C. Make sure the current Baud rate says “9600” (in the drop list at the top)
   D. Make sure the current COM port is set to the number of your RS-485 port (ex: COM2)
   E. Click the “Connect” button
   F. Click in the large text area so you can type to the Pod
2. Get the Pod to tell you about itself
   A. Press [ENTER] a few times. You should see an error message from the pod, similar to “Error use ‘?’ for command list, unrecognized command:” This is a
normal response to an [ENTER] message

B. Press “?” then [ENTER]. You should see a long message appear describing the commands available.

C. Type “Hello?” then [ENTER]. The Pod should respond with information about its model number and revision, and any firmware options it has installed.

Congratulations, you have confirmed the power and serial communication links are working perfectly.

6. EXERCISE THE I/O

Provided on the Software CD-ROM are sample programs for use with the Pod. During the software install process these programs were copied to your hard disk, and shortcuts were placed in your START Menu.

Run the sample for your Pod by clicking “Start” “Programs” “[name of pod]” “Windows Samples”, and selecting a language to use. Double click the executable file to run the sample, and configure a few options as shown onscreen.

Once you start the sample, real data is being taken and displayed, and communication is happening! Congratulations on successfully installing your Remote ACCES Distributed I/O Pod!

DIFFERENCES IN OTHER OPERATING SYSTEMS

Each operating system has its own challenges when working with independent hardware, but we’re going to describe the major differences when using the Pods with DOS and Linux.

DOS: There is no “Start” menu, you’ll need to navigate to the directory specified as destination during the install process. Instead of running WinRISC, use RISCTERM.EXE. The Window sample mentioned in step 6 won’t work of course, but we provide DOS samples in the [destination directory]/DOS directory; these should work fine.

LINUX: Every Linux distribution we’ve tested comes with a program called “minicom” which can be used instead of WinRISC or RISCTERM. Just make sure you configure the program for 7 data bits, 1 stop bit, even parity. (RISCTERM and WinRISC are configured this way by default.) We don’t provide a Linux install program, nor any samples pre-compiled for use in Linux. However, the “C” language samples provided in the [destination directory]/DOS/csamples directory can be easily modified to compile under Linux. The Linux.html file on the CD-ROM is a document detailing how to convert our DOS samples to use with Linux.