

IEEE 488

APPLICATION BULLETIN

CONTROLLING ICS'S SERIAL INTERFACE MODULES and BOARDS FROM A PC COM PORT

INTRODUCTION

ICS Electronics has introduced a family of minibox modules and OEM interface cards with serial interfaces that can be operated directly from a PC's COM port. These new modules (referred to here as 236x series) and boards offer the user all of the functions of their GPIB counterparts but with RS-232 and RS-485 serial interfaces so that they can be used in point-to-point serial links or on a serial network. This Application Bulletin describes how these new modules can be connected to a PC or to another serial source and how to control the modules from the user's program, with a terminal or with ICS's SERkybd control program. Any reference to a 236x module will in most cases also apply to ICS's serial interface cards.

BACKGROUND

The serial interface in the 236x series modules provides the user with a simple interface that does away with the need for a GPIB interface card in the PC. For a single module applications, the PC to module connection can be a standard RS-232 pin-to-pin cable. The 236x modules are designed as DCE interfaces so 'null-modem' adapters are not required. Multiple module can be connected together on an RS-485 network. The benefits of using a serial interface are cost savings (no GPIB controller card and less expensive cables), the ability to locate the modules at greater distances from the computer without using bus extenders (over 1,200 meters with RS-485 links).

While the concept of serial communication is simple, there are certain steps that the user should take to send and receive data through the PC's COM ports. Refer to Application Bulletin AB23-1 for a tutorial on serial communication.

PC CONNECTIONS

The connection between the computer and the 236x series serial modules depends upon the number of modules to be connected to the PC and the distance between the computer and the module(s). The 236x interface family gives the user a choice of two types of interfaces, RS-232 and two-wire RS-485. ICS's serial interface boards have RS-232 and 4-wire RS-422/RS-485 interfaces.

The 236x's RS-232 interface is a basic DCE type interface that can be connected directly to a PC COM Port with a standard pin-to-pin RS-232 cable as shown in Figures 1 or 2. Newer PCs with 9-pin COM port connectors will use the cable shown in Figure 2.

DCE means that the 236x's interface is wired as a Data Communications Equipment interface which is the same as a modem. In the 236x, the serial handshake control lines are looped back or are held in the ON state to satisfy the PC's handshaking signals.

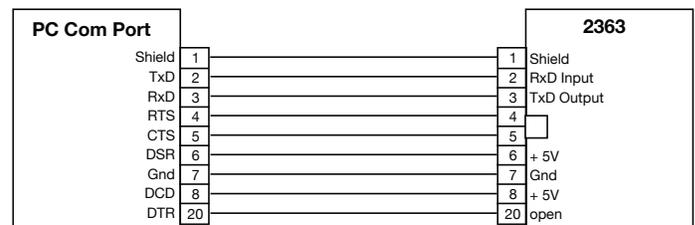


Figure 1 25-Pin RS-232 Cable Wiring Diagram

RS-232 connections can be used for short, single module to PC COM port connections. The 236x family operates at baud rates up to 34,800 bits/second which should work fine up to the RS-232 specified cable limit of 50 feet. For longer distances, the baud rate will have to be reduced to compensate for the added cable capacitance. Typically, 19,200 baud should work up to 100 feet and 9,600 baud will work up to 200 feet. The serial cable should be shielded to prevent noise pickup and to minimize cross talk. Do not run unshielded RS-232 cables next to fluorescent lights or other noise sources.

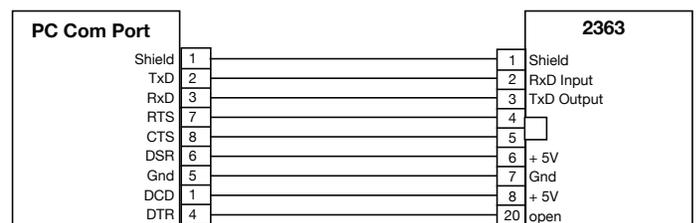


Figure 2 9-Pin to 25-Pin RS-232 Cable Wiring Diagram

CONNECTING TO MULTIPLE SERIAL MODULES

When using RS-232 signals, each PC COM port can only drive one module. Driving multiple modules can only be accomplished by adding additional COM ports to the PC. This is done by adding a second serial card with two more COM ports and/or by changing the existing serial card to a multi-port card. A suggested multi-port serial card is the ME9000 series cards from ICSDDataCom. An ME9000 card can add 4, 8 or 16 additional COM ports to the PC.

Longer cable distances and multiple modules are best handled by changing to an RS-485 transmission system. RS-485 differential signals provide greater noise immunity and longer transmission distance than do the RS-232's single-ended signals. The RS-485 system used in the 236x series family is a two wire, half-duplex circuit that transmits data in one direction at a time. This is the 236x's normal mode of operation since the modules have to be sent a command or query before they respond by sending a message to the computer.

Inside the 236x series modules, the two RS-485 lines are driven by a transmitter that is tristated when not active. The lines also go to a differential receiver and to a termination network. The termination network holds the signal lines in the mark state when all of the transmitters in the system are tristated and nothing is being sent over the network.

At the PC end, use an inexpensive RS-232 to RS-485 Converter such as the MA485F25 or MA485F9 converters from ICSDDataCom to convert the PC's RS-232 signals to RS-485. An alternate method is to add a PCI Card with RS-422/RS-485 serial ports to the PC. See the ME9000/45 and ME9000/8 cards on ICSDDataCom's website. The card costs more but gives you multiple RS-485 ports. Optical isolation is also available with the ME9000i series cards.

When using the MSA485 converters, the RS-485 network wires connect to a plug-in terminal strip with screw terminals that is wired as shown in Figure 3. The MA485 series converters automatically tristate the RS-422/RS-485 transmitter when the PC is not transmitting to prevent blocking the line. The ME9000 series serial cards include a breakout cable with separate DE-9 connectors for each serial channel. The user wires his cable to a mating DE plug. Drivers are supplied for all Windows, Linux and other operating systems.

The serial interfaces on the MA485 Converters and the ME9000 Serial cards are designed for 4-wire full-duplex connections. The conversion to a 2-wire, half-duplex connection is done by jumpering the TX and RX lines together. Figure 3 shows how this is done with the MA485 series Converters. The pair of wires going to the remote device can be connected to either signal pair.

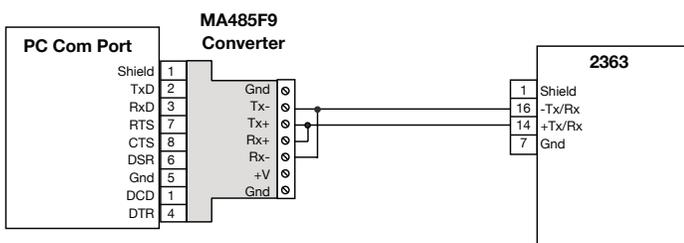


Figure 3 RS-485 Connections using a MA485 Converter

Figure 4 shows how you jumper the TX and RX signals on the DE-9 connector from a multi-port Serial Interface Card. The breakout cable or optional breakout panel is not shown to simplify the figure.

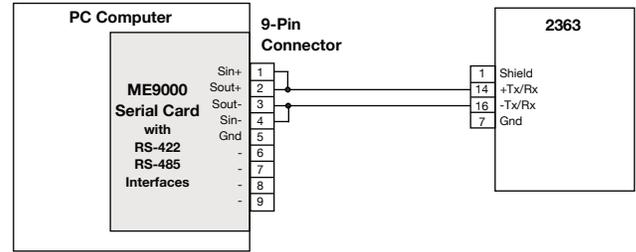


Figure 4 236x Connections to RS-485 PC Plug-in Card

RS-485 NETWORK CONNECTIONS

Multiple 236x Interfaces can be connected together to form a RS-485 network by simply connecting them in parallel as shown in Figure 5. The network wires are normally a twisted shielded pair to minimize noise pickup. Ground the shield at the PC or Serial Controller end. The values of the 236x's internal termination network are such that a 236x module can drive up to 16 other modules. If more than 16 modules are put on the same network, then the termination resistors should be removed from all but the end modules.

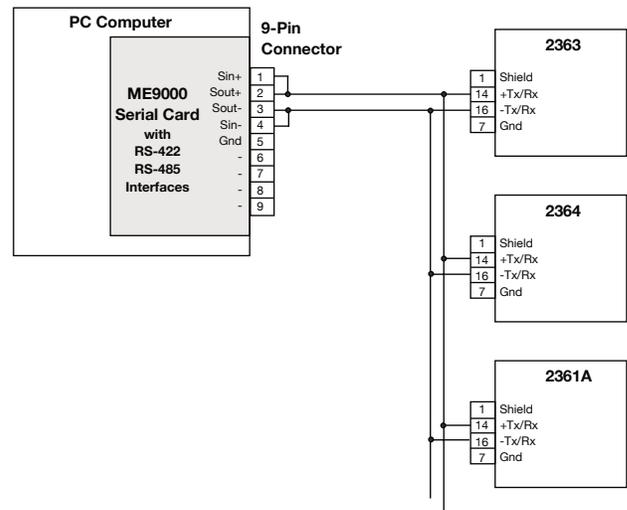


Figure 5 Multi-module RS-485 Network

In a multi-module network, the 236x modules should be configured with their network address function enabled so that they will respond to individual addresses. Only the module that recognizes its address will respond to commands sent over the network. The other modules will ignore the command. Caution- Set each module to a unique address before enabling the network address function.

Older 236x modules and the OEM serial interfaces had only two settings for the serial network: OFF and ON. Newer units have three choices: OFF, ADDRESS and PACKET. When the network setting is off, the 236xs only handle standard ASCII strings. When the network setting is ON or ADDRESS is selected, then the 236x interfaces recognize a two character sequence that is prefixed to the command. The two character sequence is the STX character (02)

followed by a ASCII number character with a value between 0 and 15 (0-9 ;;<=> and ?). i.e. the default address is STX 4. Up to sixteen 236x interfaces can share the same RS-485 network.

When the network setting is set to PACKet then all commands are set to the 236x modules as packets. The addressed module responds to each packet by returning an acknowledgement or a response packet. Each packet starts with the STX character so the interface addressing is the same as described above for the ADDRESS setting. Each packet ends with a ETX character and a checksum to verify the packet's integrity.

ICS's 236x modules can also operate on networks with other RS-485 modules as long as the other modules do not conflict with the 236x series addresses. ICS's address sequence and serial protocols are compatible with the Profibus™ system and with some other fieldbus systems.

TESTING THE CONNECTION

RS-232 connections can be tested on a Windows based PC with HyperTerminal. HyperTerminal is a utility program that is found in the Programs-Accessories group. Select the right most icon and set it for the 236x's default serial settings of 9600 baud, 8 data bits, no parity and 1 stop bit. Select a direct connection without dialing.

Put the cursor on the HyperTerminal screen and send CNTL-E to the module enable echo and a cursor after every message. Next, send *IDN? to verify data communication with the module. The module or interface board should respond with its identification string "ICS Electronics, 23xx.....". Note that the module's TX and RX LEDs blink for each message.

When hand typing commands, it is advisable to check the 236x's front panel LEDs after each command. If the red ERR LED comes on, the module detected an error in the last command. Use an *ESR? query to read the error and to clear the ERR LED. Refer to the module's Status Structure diagram in Section 3 of the module's instruction manual for a description of the ESR register bits.

ICS's SERkybd PROGRAM

HyperTerminal can only send and receive data strings. As such it can only communicate to the 236x interfaces when they are in an non-addressed mode. ICS supplies an interactive utility program, called SERkybd, that enables the user to easily communicate with 236x interfaces in any network mode.

ICS's SERkybd program lets a user send commands to a serial device and read back the device's response in a manner similar to the way ICS's GPIBkybd program works. The program lets you select string terminators, count bytes and displays error messages. SERkybd also contains buttons for sending the Cntl-E and Cntl-F commands and can run diagnostic loops. You can set the interface's address and select various communication protocols.

SERkybd also serves as a programming example for how to control ICS's 236x interfaces. SERkybd is a Visual Basic program and can be easily converted into a C language or other language programs.

The SERkybd program and source files can be downloaded from ICS's website.

RUNNING SERkybd

SERkybd starts with the Main Form shown in Figure 6. The Main Form contains all of the controls for sending and receiving serial messages. When starting the program, the user must click on the "Initialize COM Port" button to select and initialize a COM port. All of the other buttons are grayed out until the COM port is initialized. Clicking on the "Initialize.." button brings up the Port Form shown in Figure 7.



Figure 6 SERkybd Main Form

The Port Form lets the user set the baud rate and select a COM port to be used as the serial port for the rest of the program. 9600 baud is the power on default. Clicking on the "Return to Main Form" button completes the initialization of the serial port and closes the Port Form. If you are monitoring the serial signals with a LED Troubleshooting box, the control lines will switch to their 'ON' state with the red LEDs lit. You can always return to the Port Form to change the baud rate by clicking the "Initialize..." button.



Figure 7 Initialize Ports Form

When the COM port has been initialized, the remainder of the Main Form is enabled. DTR and RTS are defaulted to the 'ON' condition.

The CNTL-E and CNTL-F buttons send control characters to the 236x interfaces. CNTL-E enables echo back mode from the interface for operation with a terminal or with a terminal emulation program. CNTL-F disables the echo and so you only see a prompt character and linefeed when the 236x is ready for another command. Press CNTL-F to start as the program does not need the echo back.

Enter the desired 236x command into the Device Command textbox. Then click the Send button to transmit the message. The program defaults to adding a linefeed when transmitting the command so you do not have to put a linefeed in the Device Command window. Press the Read Device response button to check the receive buffer for a data string. If your device command was a standard query like "*IDN?", SERkybd will read the buffer for you. (AutoQuery defaults to On). You can set the Device Response options to display all characters and to display the received character count.

The IDN Message Loop buttons enable an *IDN? query which repeats twice a second. The Device Command Loop outputs the current command in the Device Command window at a twice a second rate. Both loops are good for measuring device response times, for testing the device and for troubleshooting serial connections.

The Network Protocol frame lets you switch to the ADDRESS or PACKET protocol. Multiple modules can be controlled by entering their address in the Device Address box. First put the current 236x into the desired network mode. ICS's 236x interfaces use the SYST:COMM:SER:NET ADDR or SYST:COMM:SER:NET PACK command to enable a network mode followed with the SYST:COMM:SER:UP command to update the UART and make the protocol switch. Then click the appropriate SERkybd Protocol button and enter a number in Device Address box. Press Set Address to enter the address in the program. SERkybd will convert your device commands into the correct protocol and will decode the responses for you.

PC PROGRAMMING

Most Windows compatible program languages now have functions for direct serial port communication. In a point-to-point application, the 236x interface's network function is disabled and the user simply sends the module the same command strings he would use over the GPIB bus. Some examples are:

```
For a 2361, "SOUR:VOLT 5.12"
For a 2363, "CONF:OUT (@ 3,4)
For a 2303, "ROUT:CLOS 3,1"
```

All of ICS's 236x modules and interface boards respond to all of the IEEE-488.2 Common Commands.

CONVERTING GPIB PROGRAMS

If the user has a working program for GPIB modules, it can be quickly converted into one for ICS's serial modules by changing the 'Output' and 'Enter' functions. A suggested way is to create a flag to show that the program is being used with serial or GPIB modules. Put the 'Output' and 'Enter' functions in a subroutines that can be called from the original "Output" and "Enter" commands. This way,

the same program can be used with GPIB or Serial modules. An example is:

If the original program used the Send command
`CALL Send(Bd%, Device%, OutStr$, NLEnd)`
 replace it with
`CALL newOutput(Device%, OutStr$)`

Add a flag to select the interface
`Port_Out = 0 for GPIB, 1 or 2 for Serial COM port`

```
Create the subroutine newOutput
SUB newOutput(Device%, OutStr$)
  IF Port_Out = 0 then
    CALL SEND(Bd%, Device%, OutStr$, NLEND) 'GPIB
  ELSE
    OutStr$ = OutStr$ + CHR$(10)           'add LF
    MSComm1.Output = OutStr$              'Serial
  END IF
RETURN
```

Notes:

- SEND is an NI GPIB output command.
- MSComm1.Output is a Microsoft serial output command.
- Error tests were eliminated for clarity.
- OutStr\$ is terminated with a linefeed for 236x interfaces.

There are a couple of differences between ICS's Serial interfaces and the equivalent GPIB Interface that should be noted. The serial interfaces do not generate a SRQ. Instead they can generate a Service Request Message (SRM). If SRQs are important to you, write the serial program to handle SRMs as asynchronous serial messages. Enable the SRMs the same way you would SRQs. Secondary addresses and transparent data transfers using a second GPIB address on the 4863 do not work with the 2363 modules.

CONTROLLING MULTIPLE DEVICES

When controlling devices on a multi-device network in the Addressed Protocol, the devices' address sequence is prefixed to the command string before calling the output function. ICS's 236x modules use a two character address sequence that start with the STX character (decimal 02). The second character is a number between 0 and 15 that identifies the interface. The recommendation is to use the ASCII number characters (0 to 9) for clarity (decimal 48 to decimal 63). i.e.

```
OutStr$ = CHR$(02) + CHR$(48+Device_num) + OutStr$
```

SUMMARY

This application note has described how to connect ICS new 236x serial interfaces to a PC with a Windows operating system. The application note provided directions for construction of a RS-485 network, testing methods and programming suggestions for the user's program. Also included are directions for using ICS's Serial Keyboard program, SERkybd, to control devices on a RS-232 or RS-485 serial link from the PC. The complete SERkybd program including source files can be downloaded from ICS's web site at <http://www.icselect.com>.