How to connect your computer to OI electronics.

Ref : ITC family, IPS family, ILM family, SMC family, Lambda controller, IDR/IGH, AST Controller

July 2008

Oxford Instruments
Tubney Woods, Abingdon,
Oxon, OX13 5QX, England
Telephone: +44 (0)1865 393 200
Fax: +44 (0)1865 393 333

Table of Contents

Warnings

Before you attempt to install or operate this equipment for the first time, please make sure that you are aware of the precautions that you must take to ensure your own safety.

1 Introduction to remote operation

The instruments from Oxford Instruments all have a similar method of remote control. Some of the commands are common for all the Oxford Instruments devices. For example “C3” puts the instrument into remote control, “V” gives you a description of the instrument connected to the computer and X gives you a status string of the instrument (which needs to be interpreted).

1.1 General guidelines.

1) The instrument can not be put into remote control from the front panel of the actual electronic device. An instrument can only be put into remote control by a command sent from the computer e.g. “C3” (whether it is via RS232, ISOBUS or GPIB)

2) You can always read a parameter from an instrument, whether it is in remote or local mode.

3) If you want the instrument to perform an action (e.g. IPS - go to set point, ITC - go into auto temperature control) then the instrument must be in remote control mode (an instrument can be put into remote control by issuing a “C3” command).

4) For every command you write to the instrument from a computer, you MUST read the reply. There are one or two exceptions to this rule. However, by default, you must read the reply from the instrument for every command that is sent to the computer (even if you are not going to use the reply or you are not expecting a reply). This can be thought of as a simple form of handshaking i.e. if you do not receive a reply to the command you sent, then the command was not executed and hence there is a problem.

5) The command set used to communicate via RS232 is exactly the same as the command set for GPIB. The only difference is when you want to use the ISOBUS address in some instances (GPIB and GATEWAY adaptor)

6) When you first turn on an Oxford Instruments device, the number displayed on the front panel is the address of the instrument. If no GPIB card is fitted then the address is the ISOBUS address of the instrument e.g. “S 05” is ISOBUS address 5. If a GPIB card is fitted, then the address is the GPIB address of the instrument e.g. “G 24” is GPIB address 24. In this latter instance, the instrument does have an ISOBUS address, it is just not displayed.

7) An instrument fitted with a GPIB card can communicate via RS232 or GPIB. If you use GPIB communication initially, then the instrument must be turned off and then on again to revert back to being able to use RS232.

8) You can not communicate with an instrument using both RS232 and GPIB communication simultaneously. You must use one method of communication or the other.

9) The ISOBUS address of an instrument can only be set using software commands. It is not possible to change the ISOBUS address from the front panel of the instrument.

10) The GPIB address of an instrument can not be changed via software. To change the GPIB address of a device, you must use the front panel of the instrument to enter the test modes. Please consult the instrument manual for these instructions.

1.2 Command construction

Within the manual for your instrument you will find a list of valid commands that are available along with a description. The commands that must be sent to the instrument can have different formats.

1) For direct RS232 communication (1 instrument connected to a single com port of the computer) simply send the command to the instrument (remember to terminate the command with a carriage return)

2) When using an ISOBUS cable connected to the com port of the PC and multiple instruments (e.g. ITC, IPS, ILM), sending a command through the com port requires an additional address term. This is called the ISOBUS address of the device and each device will have an address ranging from 1 to 9. The command structure then takes the following form:

@(ISOBUS ADDRESS)(COMMAND)

For example @1V

@2C3 (Each command must be terminated with a carriage return)

@3X

Each instrument on the ISOBUS cable must have a different ISOBUS address. All the instruments on the ISOBUS cable will receive the command. However, only the instrument with the correct ISOBUS address will act and respond to the command. If two instruments on the ISOBUS cable have the same ISOBUS address, they will both respond and corrupt data will be received by the computer.

3) When using GPIB communication, as with simple RS232, simply send the command you want to the instrument (to the relevant GPIB address). No other address need be considered.

4) When using a GATEWAY plug with GPIB communication, you need to combine the GPIB address with the ISOBUS address. The GPIB protocol must be used to send the command from the computer to the GATEWAY instrument. An ISOBUS address must be included in the command so that the GATEWAY instrument can relay the command to the ISOBUS cable and hence onto the instrument with the correct ISOBUS address.

For example “@4C3”sent via GPIB to the GATEWAY instrument (GPIB address say 25) will relay the command to the ISOBUS cable connected to the GATEWAY instrument (via a GATEWAY plug) and the instrument on the ISOBUS cable with ISOBUS address 4 will be put into remote control.

2 RS232/ISOBUS/Com Port communication

2.1 Cable connections

2.1.1 Computer (9 pin D type) to instrument (25 pin D type)

Pin # (25pin D)	Pin # (9pin D)	Signal Name	Direction	Function
1	Shell	FG	Common	Frame Ground
2	3	TD	To DCE	Transmitted Data
3	2	RD	To DTE	Received Data
4	7	RTS	To DCE	Request to Send
5	8	CTS	To DTE	Clear to Send
6	6	DSR	To DTE	Data Set (i.e. the DCE) Ready
7	5	SG	Common	Signal Ground
8	1	DCD	To DTE	Data Carrier Detect
20	4	DTR	To DCE	Data Terminal Ready

2.1.2 Computer (25 pin D type) to instrument (25 pin D type)

This is a 1 to cable i.e.

Pin # (25pin D)	Pin # (25pin D)	Signal Name	Direction	Function
1	1	FG	Common	Frame Ground
2	2	TD	To DCE	Transmitted Data
3	3	RD	To DTE	Received Data
4	4	RTS	To DCE	Request to Send
5	5	CTS	To DTE	Clear to Send
6	6	DSR	To DTE	Data Set (i.e. the DCE) Ready
7	7	SG	Common	Signal Ground
8	8	DCD	To DTE	Data Carrier Detect
20	20	DTR	To DCE	Data Terminal Ready

2.2 RS232/serial port /com port initialization for software

Baud Rate	9600
Stop Bits	2
Data bits	8
Parity	None
Flow Control	None

2.3 Writing an RS232/ISOBUS communication program.

2.3.1 Simple RS232 communication

1) Initialize the port as per the setup parameters.

2) Write the command you want to send (terminated with a carriage return (ASCII 13)) to the com port e.g. “C3”

3) Read characters received at the com port until you receive a carriage return (which marks the end of the reply). Put all the characters you received together to get the complete reply.

4) Interpret the reply.

2.3.2 Advanced, robust RS232/ISOBUS communication

1) Initialize the port as per the setup parameters. This must be done once initially, or if there is an error reading or writing or there is a timeout from the previous command.

2) Read any parameters present on the com port. If there are characters present, wait for 5 milliseconds, then read any characters present again until there are no characters present. This ensures that there are no characters present on the com port which could be miss interpreted.

3) Write the command you want to send (terminated with a carriage return (ASCII 13)) to the com port eg “C3”

4) Read characters from the com port – store them in a variable. Does the reply contain a carriage return (ASCII 13)? If it does, you have the complete reply from the instrument. If not, read from the com port again until you receive a reply which contains a carriage return. Check when waiting for the reply if the response from the instrument takes longer than a given time e.g. 5sec. If the instrument does not respond within say 5 seconds, then there is a problem.

5) For complete robustness, once you have a complete reply check that each character in the reply is a true character and not corrupted.

6) If the reply is OK, interpret the response as necessary for your program.

7) This is only a suggestion. If there is a problem (command timed out, or you did not receive a reply, or the reply was corrupt, or you were not able to write or read to/from the com port), display a dialog on the screen informing you that you have an error, and a message telling you what the error was. Display this dialog for say 5sec, then re-initialise the com port and re-issue the command to try again.

8) If the reply was OK and everything worked with no errors, start the next command at step 2 i.e. do not initialize the com port (it is not necessary).

3 GPIB/IEEE communication

Oxford Instruments devices are IEEE-488 compatible. For GPIB communication with an instrument, you firstly must have a GPIB card in your computer, and the National Instruments driver software installed and working correctly. If the GPIB card or driver has a problem, then trying to communicate with the Oxford Instruments device will possibly not succeed.

The driver for a National Instruments card can be downloaded from their web site. When the Oxford Instruments device is first turned on, if a GPIB card is fitted, the GPIB address will be displayed on the front panel e.g. “G 24” is GPIB address 24

3.1 Cable

The cable for GPIB connection can be bought from National Instruments and other similar electronic suppliers. National Instruments claim that the cable should not be more than 2 meters long.

3.2 National Instruments driver calls.

If you are using individual GPIB calls to the National Instruments driver to establish communication with the IPS120, then the following calls need to be done in sequence to achieve communication:

ibfind GPIB0	This finds your computer GPIB card
ibrsc 1	Request system control
ibdev 0,25,0,13,1,0x140D	Open and initialize an unused device (GPIB address 25)
ibclr	Clear specified device
ibrsp	Return serial poll byte
ibwrt "@0V\r"	Write data from string ("\r" is a carriage return)
ibsrp	Return serial poll byte
ibrd 80	Read data from a device to a string
ibrsp	Return serial poll byte

3.3 The National Instruments “Scan for Instruments” software

National Instruments do provide software which allows you to “Scan for devices” connected to the GPIB card in your computer. Unfortunately, with Oxford Instruments electronic devices such as the ITC, IPS etc., this National Instruments routine will only identify the GPIB address of the instrument in question, but will not give you a description of the instrument. This is due to most instruments being able to respond to the “*idn?” command. Unfortunately, Oxford Instruments devices do not respond to the “*idn?” command. To find out the identity of an Oxford Instruments device, you must issue the “V” command instead.

3.4 Writing a GPIB communication program.

3.4.1 Simple GPIB communication

1) If this is the first command to be sent, or the previous command was an error, initialize the GPIB board (send interface clear and set remote enable, set the DMA mode or programmed I/O mode to 0, assert remote enable (1)).

2) Clear the device you want to talk to (device clear). Wait for 100ms.

3) Write the command to the instrument. This command follows the normal command structure (append a carriage return (CR) to the string and send end of instruction (EOI) with carriage return (CR)). Wait for 50msec.

4) Read from the device (end of string (EOS) character is carriage return (CR), read terminated on end of instruction (EOI), requested byte count, or carriage return (CR))

3.4.2 Advanced robust GPIB communication

1) If this is the first command to be sent, or the previous command was an error, initialize the GPIB board (send interface clear and set remote enable, set the DMA mode or programmed I/O mode to 0, assert remote enable).

2) Clear the device you want to talk to (device clear). Wait for 100ms.

3) Read the Serial Poll Byte.

4) Write the command to the instrument. This command follows the normal command structure. Append a carriage return (CR) to the string and send end of instruction (EOI) with carriage return (CR).

5) Repeat reading of the serial poll byte until bit 4 (message available(MAV)) of the byte is high i.e. 1 (make sure this times out after n sec)

6) Read from the device (end of string (EOS) character is carriage return (CR), read terminated on end of instruction (EOI), requested byte count, or carriage return (CR))

7) For completeness repeatedly read the Serial Poll Byte until bit 4 of the byte is low (0).

8) Take the reply from the instrument and check that all the individual characters are not corrupted.

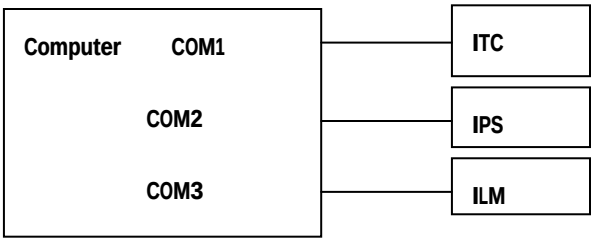
9) If at any point an error has occurred display a dialog informing you that there is a problem for say 5sec. Then reinitialize the GPIB card, clear the device and issue the command from the beginning of this sequence once more.

10) If the reply is OK, for the next command there is no need to reinitialize the GPIB card or clear the device. Start from step 3 by reading the Serial Poll Byte.

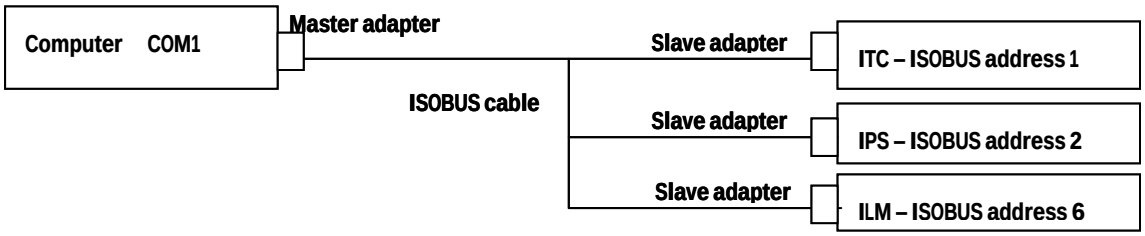
4 General Computer Connection

Below is a diagram that illustrates how our electronic devices can be connected to a computer using RS232, ISOBUS, GPIB and GATEWAY cables and adapters.

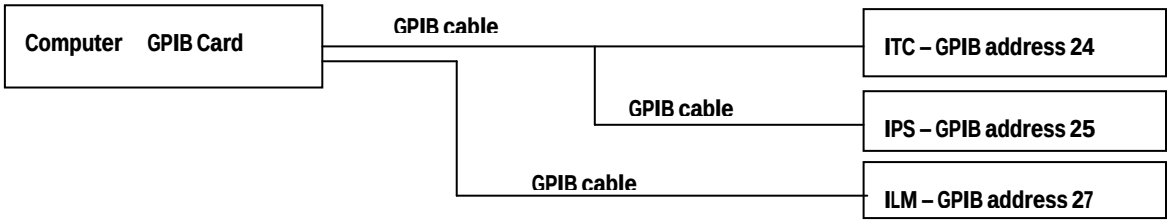
RS232 Communication



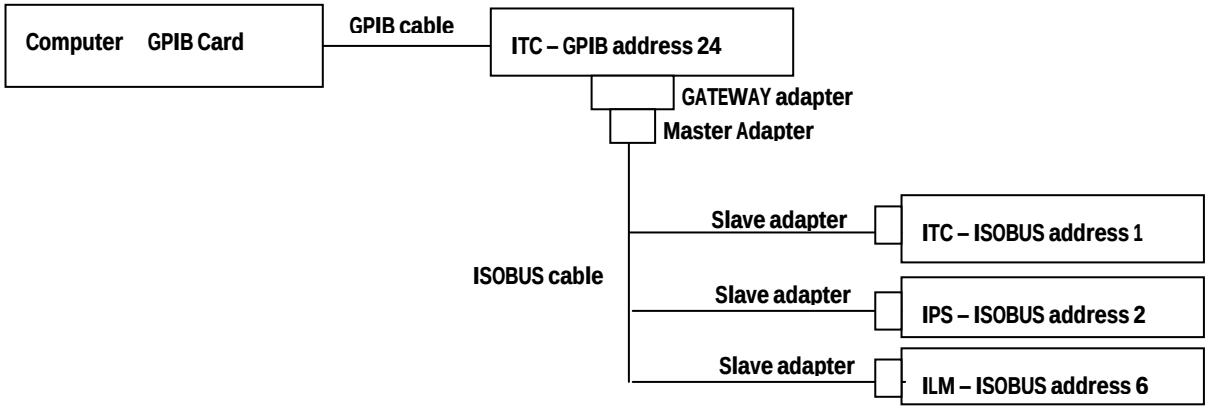
ISOBUS Communication



GPIB Communication



GATEWAY Communication



5 Fault finding

1) Please check the cable connected from the instrument to the computer. If it is RS232, check the pin connections in the RS232 section (if you are using an ISOBUS cable, you will not be able to do continuity checks since ISOBUS is an optically isolated cable). If you are using GPIB communication, make sure the cable is less than 2 meters long and that the cable is plugged in properly. If possible, use a GPIB cable that you have used successfully with other equipment. This will eliminate the cable as the source of the problem.

2) If you are using an ISOBUS communication and you are receiving corrupt characters back from the instruments, then there are a couple of things you can try. Firstly, make sure that the instruments all have a different ISOBUS address. Second, initialise the com port correctly. Thirdly, read any residual characters on the com port first before you send your command to the instrument.

3) If you are using GPIB communication, please check that all the instruments have a different GPIB address.