



## APPLICATION BULLETIN

### USING NATIONAL INSTRUMENTS GPIB HARDWARE AND SOFTWARE WITH THE ESS, INC. RC900 CONTROLLER

#### INTRODUCTION

This application note will help solve communication problems encountered when using National Instruments (NI) hardware and software with the ESS, Inc. RC900 controller with the GPIB communications interface. NI hardware refers to GPIB PC cards and GPIB cables. NI software refers to LabVIEW. There are several common problems encountered when using NI GPIB products. This application note will deal with the most common problems and offer potential solutions.

#### SOFTWARE

##### CARRIAGE RETURN TERMINATION CHARACTER

The ESS, Inc. RC900 controller with GPIB interface uses text strings to send data between itself and the GPIB host computer. In addition to the text characters that make up the data string a carriage return <cr> (13) **must** terminate the data string. In many programs such as HP BASIC, the carriage return <cr> (13) is automatically added at the end of the data string when you press the **ENTER** key. However, this is NOT the case when using LabVIEW. The carriage return will have to be added to the string manually with special characters. See the example below:

##### Typical string in HP BASIC:

=<sp>SP1<sp>100<cr>      \* note that <sp> indicates a space and <cr> indicates a carriage return

##### Same string in LabVIEW:

=\sSP1\s100\r      \*note that \s indicates a space and you must add the \r carriage return

In LabVIEW, pressing the space bar will get you a space character in your string. However, pressing the **ENTER** key will NOT get you a carriage return at the end of your string. You must add the carriage return by typing \r at the end of the string. Then when the string is sent to the RC900 it will contain the necessary <cr> (13) termination character.



## Enabling EOI

When the RC900 outputs a data string over the GPIB interface, a carriage return <cr> (13) and line feed <lf> (10) are automatically attached as termination characters at the end of the data string. If you need the RC900 GPIB interface to assert EOI to avoid a timeout in your program then EOI will need to be enabled. Factory default value for EOI is 0 (disabled). To check the status or to enable EOI use the following SCPI commands:

**Note: On early RC900 controllers using the 4804 series of GPIB interface it is necessary to put the interface into COMMAND mode before sending SCPI commands. This can be done using the “Escape Sequence” or the IEEE-488.2 “Find Listeners” command. RC900s that are fitted with the 4804 GPIB interface can be identified by an external 8-position slide switch located on the rear of the chassis next to the GPIB connector. This switch is used to set the unit address in hardware overriding the address stored in software. Later RC900s using the 4806 series of GPIB interface have smart firmware that allows the interface to distinguish between COMMAND and DATA strings. 4806 GPIB equipped RC900s can be identified by the lack of an external hardware address switch. Unit address can only be set in software.**

Sample HP Basic “Escape Sequence” string to put 4804 GPIB interface into COMMAND mode

**SEND “UNL LISTEN 04 UNL LISTEN 04 UNL” \* 04 is the 4804 GPIB address**

SCPI command to query EOI status

**SYST:COMM:SER:EOI?**

Response: 0 = EOI disabled 1 = EOI enabled

SCPI command to enable EOI status

**SYST:COMM:SER:EOI 1 \* SYST:COMM:SER:EOI<sp>1** (space necessary)

To query new EOI status

**SYST:COMM:SER:EOI?**

Response should be: 1

Command to save new EOI value into non-volatile memory

**\*SAV 0 \* \*SAV<sp>0** space is necessary - last character is a zero

To put the 4804 back into DATA mode (this is not necessary with the 4806) send the following command:

**SYST:OPER DATA \* SYST:OPER<sp>DATA** space is necessary



## **USING “FIND” UTILITIES OR \*IDN? TO IDENTIFY GPIB BUS DEVICES**

Early RC900s use the type 4804 GPIB interface. (See “Note” above under EOI) The 4804 GPIB interface has two operational modes; DATA and COMMAND. DATA mode allows GPIB data strings to flow bi-directly through the 4804 interface between the RC900 controller and the host GPIB computer. COMMAND mode is used to change software settings such as the GPIB address and store them in non-volatile memory. At power up and during regular operation the 4804 remains in DATA mode. However if a parameter such as the bus address is to be changed and saved remotely via the GPIB interface then the 4804 must first be placed in COMMAND mode. This is done using an escape sequence (see above or operation manual for more details). While in COMMAND mode, changes can be made to the software settings but no data strings can be passed through. Once the software change has been made and saved the 4804 must be placed back into DATA mode again. This can be done in software by using the SCPI command “SYST:OPER DATA” or by powering the RC900 OFF then back ON. The RC900 4804 GPIB interface always powers up in DATA mode.

Using “FIND” type utilities or the \*IDN? command to identify devices on the GPIB bus has the same effect as an escape sequence and will cause the 4804 to go into COMMAND mode. Unless the RC900 is cycled off and on or the “SYST:OPER DATA” command is sent then the 4804 will remain in COMMAND mode and no data can pass between the RC900 and host GPIB computer.

## **HARWARE**

### **NI PC CARDS AND GPIB CABLES**

IEEE-488.1 deals primarily with hardware standards for GPIB devices. The standard recommends that GPIB devices operate at a maximum speed of 1 Mbyte/s. Some NI PC GPIB cards can operate at speeds as fast as 8 Mbyte/s and use TNT drivers with extremely fast rise times. IEEE-488.1 also limits GPIB cable length to 2 meters (a little under 7 feet). NI and other vendors sell GPIB cables that are much longer than the standard calls for. Either of these deviations from the standard can cause GPIB communications failures and combining fast PC GPIB cards with long GPIB cables can significantly increase the risk of communication failure. Lastly, GPIB cable quality can vary significantly from vendor to vendor. Shielding, grounding and the quality of the wire cable and connectors all play a role in their reliability. If you are encountering problems with your GPIB interface, be sure you are using a high quality cable of no more than 2 meters. If you must use a longer cable then using a GPIB PC card that operates at the recommended 1 Mbyte/s speed may be necessary.

## **FIRMWARE UPGRADES**

### **UPGRADE YOUR 4804 FIRMWARE**

The 4804 GPIB interface option to the RC900 uses an EPROM for permanent storage of firmware and default parameters and a separate E<sup>2</sup>PROM for storing user configurable parameters. If the hardware watchdog suspects that the data on the E<sup>2</sup>PROM has become corrupted it will re-write the default parameter data from the EPROM. This re-write usually occurs due to a voltage transient or “brown out” in the power supplied to the 4804 interface card. If a re-write occurs then incorrect



parameter data may be written to the E<sup>2</sup>PROM and the GPIB interface will not function. This problem can be solved by replacing early an firmware version EPROM with the newest version (currently version R8). The new EPROM parameters will exactly match the E<sup>2</sup>PROM parameters (other than those changed by the user such as GPIB address) so if a re-write occurs the parameters are always correct. The new firmware, version F30143R8, is available at no charge as an EPROM upgrade to the 4804 GPIB interface. You can send your 4804 GPIB interface board in and we will install the new EPROM and test the interface at no charge.

## **UPGRADE YOUR 4804 TO A 4809**

If you are using NI GPIB PC cards and or LabVIEW software and you have made sure that all of the common problems associated with GPIB communications have been solved and you are still having problems, you may want to consider a firmware upgrade. The upgrade can be accomplished by replacing the 4804 EPROM. The new EEPROM will convert the 4804 into a 4809. The 4809 uses smart firmware and has only one mode of operation and therefore can be easily used with "FIND" type utilities and the "\*IDN?" command without problems. The EPROM to convert a 4804 to a 4809 is part number F30160R0.

The command protocol in the RC900 will also have to be changed in the COMM menu. The new command protocol will require a change in the text strings that are used to communicate with the RC900. If you have a lot of code already written in the 4804 protocol this will be something to consider. Sample protocol strings are below:

### **4804 PROTOCOL STRINGS**

**= SP1 100** \* write a new set point of 100 degrees

**? C1** \* read the current channel 1 temperature

### **4809 PROTOCOL STRINGS**

**W 7,100** \* write a new set point of 100 degrees

**R 1,1** \* read the current channel 1 temperature

For more information on solving problems with the GPIB interface option please contact ESS, Inc.

Last Revised: Tuesday, April 30, 2002