**ICS
ELECTRONICS**
division of Systems West Inc.

IEEE 488

APPLICATION BULLETIN

Troubleshooting ICS's 48x9A Interfaces
Solving Modbus Serial Communication Problems

INTRODUCTION

This application note describes how to troubleshoot communication problems with ICS's 48x9A Interfaces and the Modbus slave device they are controlling. In general, most problems with new applications end up being a baud rate or wiring issue. Problems with existing systems can be a hardware failure, a damaged cable or settings that have been inadvertently changed. The steps in this application note should help get your system running again. The following directions are given for a 4899A but apply to all ICS 48x9A series interfaces.

PC and PROGRAM

You will need to run an interactive GPIB control program on your PC that will lets you enter commands as you go through the following steps. If you are using an National Instruments GPIB card, you can use NI's Measurement and Automation Explorer (MAX) or ICS's GPIBkybd program. ICS's GPIBkybd program works will all GPIB cards that use a GPIB-32.dll. (NI, Keithley, Measurement Computing, Adlink etc.) If you have an HP, Agilent or Keysight card, use the Keysight IO Connection expert.

4899A and GPIB CONNECTION

Launch the interactive GPIB control program and use it to find the 4899A. If the 4899A is not found, power cycle the 4899A and watch the front panel LEDs. At the end of its internal self test, the 4899A will blink its GPIB address on the front panel LEDs. The LED bit weights are:

RDY	TALK	LSTN	SRQ	ERR
16	8	4	2	1

If none of the LEDs blink, the 4899A's GPIB address is 0. The GPIB controller will have to be set to a non-zero address to talk to it. If using the GPIBkybd, set the Controller address to 21 and retry finding the 4899A. Alternately, you can reset the 4899A to its factory setting which is GPIB address 4 by following the procedure in paragraph 5.4 in the 4899A manual.

If none of the 4899A's LEDs come on, it may not be getting any power. Check the power supply for 12 volts and the outlet for AC power. If the 4899A's LEDs are frozen in an odd pattern, it has an internal failure and will need to be repaired.

If the red ERR LED comes on when the unit is power cycled, that indicates the 4899A has lost its settings and has reset itself to the factory default settings. It will need to be re-configured to work properly in your system which we will do in a later step.

Use the 4899A's GPIB address to setup the interactive GPIB control program to communicate with the 4899A. Send it an IDN query and verify that you can read back an IDN message.

```
Send:     *IDN?
Response :  ICS Electronics, 4899A, S/N xxxx, rev 02.17
```

4899A's delivered with a system or a test chamber may have the system manufacturer's IDN message. The important thing is that you get a response. If you did not get a response, look at the 4899A's front panel to see if the red ERR LED came on. If it is on, you may have sent the 4899A a bad query, the GPIB cable maybe damaged or not connected properly. GPIB cable length should be no more that 2 meters between devices and 20 meters overall. Swap cables, check connections to fix the problem and repeat until you can read the IDN message.

Next read the 4899A's ESR and Modbus Error Registers to clear them.

```
Send:     *ESR?
Response: 128          'power turn-on

Send:     *ESR?
Response: 0            'ESR register cleared.  No errors

Send:     E?
Response: 0            'Modbus Error Register.  No errors
```

At this time you have good GPIB communication with the 4899A and have learned to read the ESR and Modbus Error Registers.

If the red ERR led comes on at power turn-on time, the unit lost its settings and must be re-configured by entering the settings for your system. Hopefully they were written down or you can obtain them from your system supplier. Refer to the SCPI Command Tree on page 3-15 of the 4899A Manual for the configuration commands and see the command examples below.

When done reentering the settings, send the 4899A a CAL:DATE command using today date. This will save the settings and keep the ERR LED from coming on at power turn-on time.

```
Send:    CAL:DATE 02-28-28         'use todays date
```

Power cycle the 4899A before proceeding.

MODBUS COMMUNICATION

Successful modbus communication requires that the 4899A and Modbus device serial settings match, that the cable connection is correct and that the 4899A jumpers are correctly set for the serial signals. Ignore the jumper settings for working systems that have failed. Turn the Modbus Slave Device on so it is running normally. Check its Modbus Device's address to see it it matches the address set in the 4899A. Use the C command to query the 4899A's setting.

```
Send:    C?
Response: 1
```

If the addresses are different, set them so they are both the same. Save any changes to the 4899A. Use the SYSTem:COMMunicate:SERial branch to query and set the 4899A's serial parameters. An example for setting baud rate is:

```
Send:    SYST:COMM:SER:BAUD 19200
Send:    SYST:COMM:SER:BAUD?
Response: 19200
```

4899A ESR Register Bits

Bit	Bit Weight	Event	Description
7	128	PON	The Power-on event occurs at power turn-on and can be used to signal a power off-on occurrence.
6	64	Modbus	Modbus Error detected. Reading the Modbus Errror Register
		Error	clears this bit. See the E? query in Table 3-5.
5	32	Cmd	Command Error
4.	16	Exc	Execution Error includes EDR not set and missing listen handshake.
3	8	Flash	Flash data corrupted.
2	4	Query	Query error, data not read or read attempt with no data.
1	2	not used	
0	1	OPC	Operation Complete has no meaning in the 48x9As.

Always query your setting to be sure it is correct. If the ERR LED comes on, the 4899A had a problem with the command. Do an *ESR? query to clear the error and resend the correct command. The ESR reading is the sum of the active bits. Use the table below to determine the exact error.

Try sending it a query to read a known register such as model number, chamber temperature etc.

```
Send:    R? 0,1
Response: 5720          'Watlow F4 model number
```

If you get a GPIB timeout (NI EABO error), the 4899A's ERR LED will be on. Send *ESR? to read the ESR register to get the error value. The 4899A ESR register buts and their meanings are:

The response value is the sum of the set bits. If bit 6 is set, read the Modus Error Register. The Modbus Error Register has the following responses:

4899A Modbus Error Register Responses

Response	Meaning
0	No errors present
1	Exception Code 1
2	Exception Code 2
3	Exception Code 3
100	CRC Error set when the Modbus Slave Device response did not checksum correctly.
101	Timeout Error indicates no characters received in the response message.
2nn	Partial or corrupted message received. nn is the number of received bytes.

The exception codes are from the Modbus Device and their meanings are described in its manual. They are a good sign as it means you have good Modbus communication. Responses 100-200 are from the 4899A and indicate a communication problem.

The typical response is 101. This is either a serial setting mismatch such as baud rates or a bad cable connection. Compare the serial settings in the 4899A (baud, parity, bits and stop bits) to the settings in your Modbus Device. They must be set the same for successful communication. If you change the 4899A's settings, save the settings with *SAV 0 (there is a space before the 0) and power cycle the 4899A so the changes can take affect.

Modbus errors 100, 101 and 200 can also be due to the Modbus device needing more time to respond than the 4899A's timeout setting. Check and change the timeout setting with the D command.

```
Send:    D?            'queries the serial timeout
Response: 100          '100 milliseconds
```

Lengthen the timeout setting to 500 milliseconds.

```
Send:    D 500         'sets timeout to 500 milliseconds
Send:    D?            'queries serial communication timeout
Response: 500          '500 milliseconds

Send:    *SAV 0        'saves the new setting
```

Power cycle the 4899A and try reading the Modbus device register again. Any 2nn error should be gone and the ERR LED should stay off. If the error persists, consult the Modbus device manufacturer for the correct communication timeout value. Add 100 milliseconds to the given value when setting the Modbus communication timeout.

If the 500 ms timeout works, try backing the timeout down in steps of 100 ms. When communication fails, add 100 ms to the timeout value and save it as your new setting. Run a register query loop for 10-20 minutes to be sure the timeout setting is correct.

SERIAL CONNECTIONS

RS-232 connection problems on new systems are typically reversed transmit and receive signals. Pins 2 and 3 are the TX and RX signals and pin 7 is ground on a 25-pin connector (pin 5 on a 9-pin connector).

Use an LED Signal Module or a voltmeter to determine the signal wiring. With the LED Signal Tracker, the LEDs for pins 2 and 3 must both be the same color. With the voltmeter, disconnect the serial cable from the 4899A and measure between pin 7 and pins 2 and 3 on the plug. The pin with a negative voltage of -5 to -12 V is the TX signal. The other signal should be 0 ± 1 volt. The TX signal must be on pin 3 to mate with the 4899A. If the signals are not correct, use a null-modem adapter or rewire the connector so the Modbus device TX signal is on pin 3 and RX is on pin 2.

Older systems may have a break in the cable or the signals are not reaching the Modbus device, another hardware error or the settings were inadvertently changed. Check the cable continuity or check the signals at the terminals on the Modbus device to find the fault. When properly mated, both the TX and RX terminals signals should read -6 to -12 volts. The following figure shows the 4899A connected to a Watlow F4 Controller. Measure from terminals 16 to terminals 14 and 15. Other Modbus devices will have different terminal s o connectors where you can measure the signals.
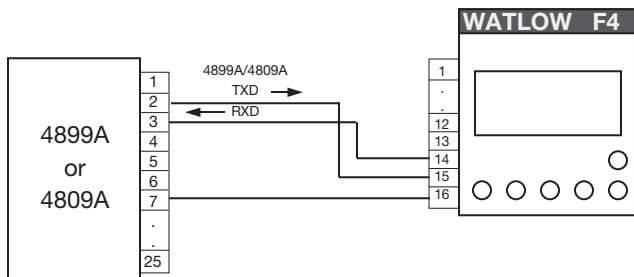

Figure 1   4899A RS-232 connection to a Watlow F4

New two-wire, RS-485 connection problems are possible reversed connections, floating signals and a configuration error. Two-wire

RS-485 networks are floating unless a device is transmitting and should have a terminating resistor or a bias network to minimize noise into the receiver. ICS recommends using a three resistor network as shown in Figure 3. The 4899A provides an internal pullup resistor to +5 V so you only need to add an external load and pulldown resistors as shown in Figure 2. This biases the lines and keeps 260 mv across the signal pair while they are not being driven. Unfortunately there is not an easy way to determine RS-485 signal polarity so it is just trial and error.
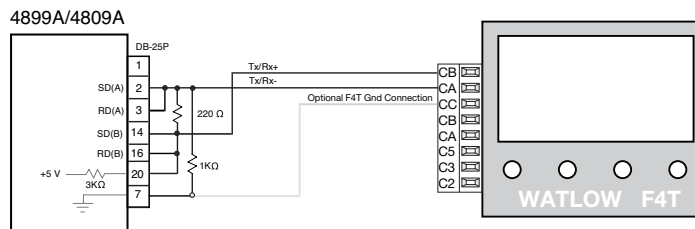

Figure 2   4899A RS-485 connections to a Watlow F4T

When the 4899A is used with two-wire RS-485 networks, the RS485 variable must be set on. Leave it off or RS-232 and four-wire RS-422 and RS-485 communication. Send the 4899A the following commands to set the RS485 variable.

```
Send:      SYST:COMM:SER:RS485 1
Send:      SYST:COMM:SER:RS485?
Response: 1

Send:      *SAV 0
```

Power cycle the 4899A to update the unit after making serial configuration changes.

SUMMARY

This Application Note has described how to troubleshoot Modbus communication with ICS's 48x9A series GPIB to Modbus interfaces. The goal is to provide the user with some knowledge of the 4899A and how to go about troubleshooting communication problems. Contact ICS Support at support@icselect.com.

REFERENCES

The following references provide more information on the subjects discussed in this Application Note:

[1] 4809A_19A_99A_manual. Download from icselect.com
[2] Application Note AB48-48, AB48-48 F4T Connections. Download from www.icselect.com
[3] RS-232 Signal Tracker. Download a data sheet from www.icsdatacom.com