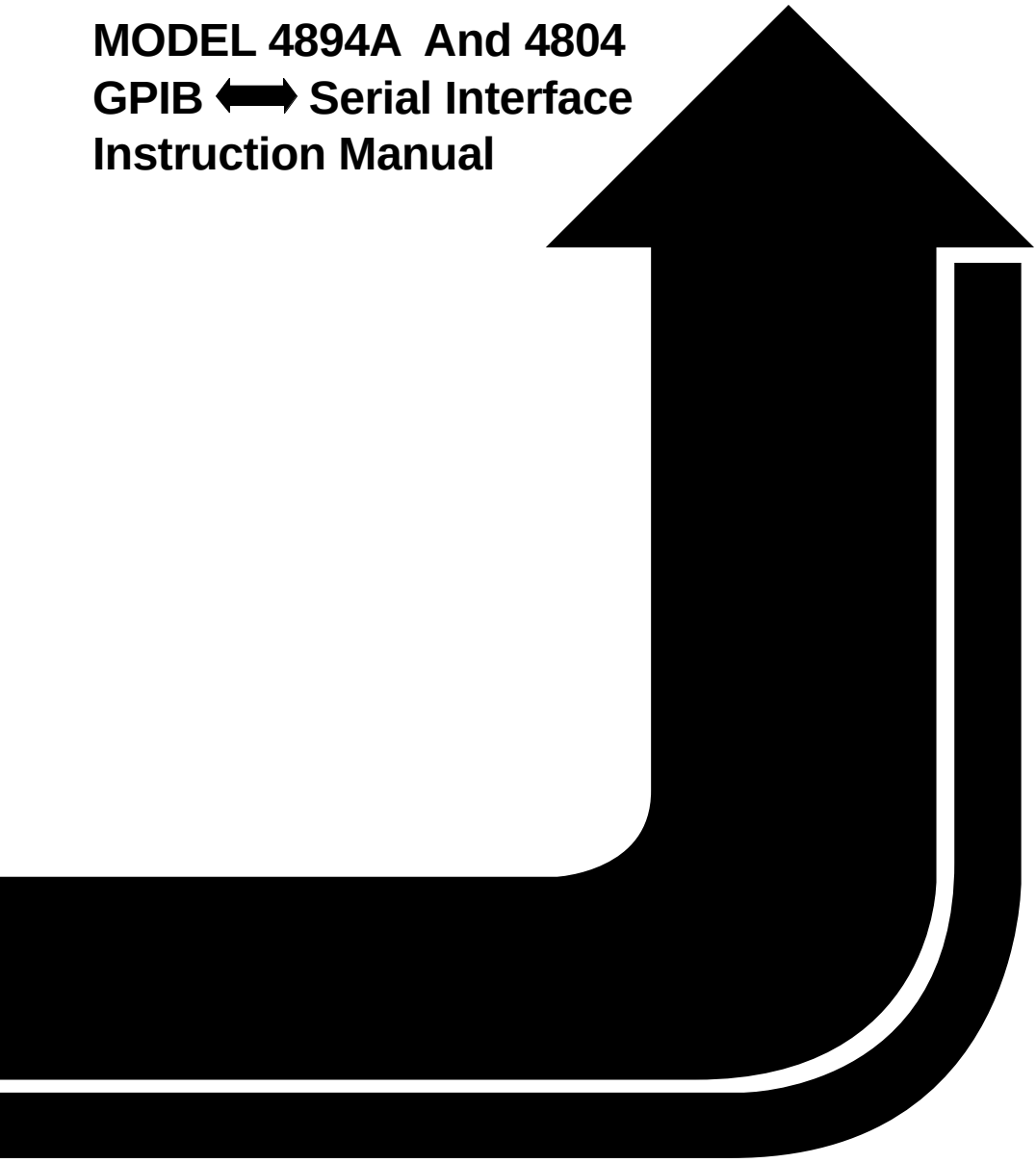
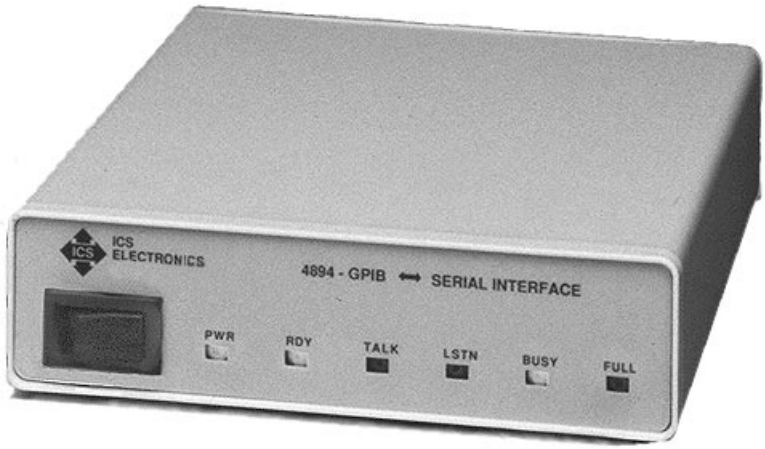
ICS
ELECTRONICS
a division of Systems West Inc.

MODEL 4894A And 4804
GPIB ⟷ Serial Interface
Instruction Manual



MODEL 4894A/4804
GPIB ⟷ Serial Interface
Instruction Manual



ICS ELECTRONICS
a division of Systems West Inc.
473 Los Coches Street
Milpitas, CA 95035
Phone (408) 263-5500
Fax (408) 263-5896

Publication Number 120117
June 1998 Edition Rev 5

LIMITED WARRANTY

Within 12 months of delivery (14 months for OEM customers), ICS Electronics will repair or replace this product, at our option, if any part is found to be defective in materials or workmanship (labor is included). Return this product to ICS Electronics, or other designated repair station, freight prepaid, for prompt repair or replacement. Contact ICS for a return material authorization (RMA) number prior to returning the product for repair.

CERTIFICATION

ICS Electronics certifies that this product was carefully inspected and tested at the factory prior to shipment and was found to meet all requirements of the specification under which it was furnished.

EMI/RFI WARNING

This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause interference to radio communications. The Model 4894A has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of the FCC Rules and to comply with the EEC Standards EN 55022 and EN 50082-1, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference. The Model 4804 should be tested for RFI/EMI compliance as a component in the user's equipment.

CE Certificate of Compliance reproduced in Figure 1-2.

TRADEMARKS

The following trademarks referred to in this manual are the property of the following companies:

HP is a trademark of Hewlett-Packard Corporation, Palo Alto, CA
ICS is a trademark of Systems West Inc, Milpitas, CA

Contents

General Information

1.1 INTRODUCTION

This section provides the specifications for ICS's Model 4894A and 4804 GPIB ↔ Serial Interfaces and their accessory items. The Model 4894A is an enclosed product designed for bench use with other equipment. The Model 4804 is a PC board product designed for placement inside another piece of equipment. Both products are functionally equivalent. Wherever the text refers to the Model 4894A, it also applies to the Model 4804 unless otherwise stated. Model 4804 differences are noted in parenthesis or in separate paragraphs.

1.2 DESCRIPTION

The Model 4894A/4804 GPIB ↔ Serial Interfaces are transparent interfaces that transfer data in a bidirectional manner from a GPIB bus to a device with an RS-232, RS-422 or an RS-485 serial interface. Typical applications are:

1) acting as a GPIB interface for devices with serial I/O ports such as modems, printers, plotters etc.

2) converting serial data into GPIB data to drive a printer or similar device with a GPIB interface or to control a single GPIB instrument interface.

When controlled from the GPIB bus, the units operate in the G mode. When controlled from a serial source, they operate in the S mode.

The 4894A contains a number of advanced features that increases its flexibility and simplifies its use in system applications. The 4894A includes a 256 Kbyte RAM that buffers up to 252K characters. (The Model 4804 has a 8 Kbyte RAM and buffers up to 4K characters) This buffering capability and the 4894A's high GPIB data transfer rate minimizes data transfer time and off-loads the GPIB controller for other tasks while the 4894A outputs the buffered data. The 4894A is completely programmable by SCPI commands from its GPIB interface, eliminating the need to open the unit to change or verify a function setting. This programmable capability makes the unit easy to use in a system application since its functions can be changed at any time or stored in its internal nonvolatile memory. The 4894A is also available with internal switches for setting its configurable functions. When the internal switch option is specified, the 4894A operates only a an IEEE-488.1 device and does not respond to the SCPI or to IEEE-488.2 common commands.

The GPIB interface for both units is IEEE-488.2 compliant and uses SCPI 1991.0 commands to set its configuration. At power turn-on, the units perform a self-test. At the end of the self-test, the unit momentarily displays its GPIB address and then turns the RDY LED on if the test was successful.

The Model 4894A is packaged in a small metal case that is less than 1U in height (1.6 inches) The front panel contains the power switch and LEDs which indicate the unit's status. The rear panel contains the GPIB and serial connectors and a DC power jack. The 4894A accepts a wide range of DC voltages and is shipped with an adapter for the local power lines.

The Model 4804 is a small, low-profile PC assembly that has flat ribbon connectors for its GPIB and serial signals. The 4804 has the same diagnostic LEDs as the 4894A but uses 5 volt power. The 4804's GPIB address can be set by the value stored in its E2PROM by connection to an external address switch or by a serial command.

1.3 MODEL 4894A/4804 SPECIFICATIONS

The following specifications apply to all models. Options for your unit may be found by comparing the list below to those listed on the serial label on your unit.

4894A - X — General Model Number

Option Codes

-1	Special Crystal
-3	Custom Paint
-5	Custom Front Panel
-7	Special Program
-8	Internal Switches or Custom Hardware Modification
-9	Factory Installed Rack Mount Kit
-A	Ship with Australian 230 Vac Adapter
-B	Ship with British 230 Vac Adapter
-E	Ship with European 230 Vac Adapter

Base Model Number

Note that the Model 4804 is a board level product that uses only + 5 Vdc. Options -3, -5, -9, and the AC adapters do not apply to the Model 4804.

1.4 OPERATIONAL MODES

The 4894A/4804 has two interfaces, a GPIB interface and a serial interface. The operational mode is determined by which interface is connected to the controlling device.

G-Mode	4894A controlled by GPIB inputs
S-Mode	4894A controlled by serial inputs

The 4894A/4804 has two sub-operational modes, one of which is selected at configuration time. They are:

IEEE-488.2 operation or
IEEE-488.1 operation which emulates National Instruments' GPIB-232CV or GPIB-422CV operation.

Model 4894A-8 units with internal switches operate only as IEEE-488.1 interfaces.

The Model 4804 units can be connected to an external address switch that overrides the GPIB address stored in the internal E2PROM. Model 4804s can be also sent a serial command that sets the GPIB address and saves the new value in the unit's E2PROM.

Refer to section 2.0 for factory setup and configuration instructions.

1.5 IEEE 488 INTERFACE

1.5.1 488.1 Capabilities

The 488 Bus interface meets the IEEE STD 488.1-1987 standard and has the following capabilities:

G Mode: SH1, AH1, T6, L3, SR1, PP0, DC1, RL0, DT0, C0 and E1/E2 drivers.

S Mode: SH1, AH1, T3, L2, SR0, RL0, PP0, DC0, C1, C2, C3 and C28

1.5.2 Address Ranges

Primary addresses 0 - 30, listen only (G-mode) or controller/talk-only (S-mode).

1.5.3 Data Transfer Rate

> 600,000 bytes/second from GPIB bus to memory (G-mode).

1.5.4 488.2 Common Commands

Standard 4894A/4804s conform to IEEE STD 488.2-1987. When addressed to listen in the command mode, the unit responds to the following 488.2 commands:

***CLS**, ***ESE**, ***ESE?**, ***ESR?**, ***IDN?**, ***OPC**, ***OPC?**, ***RCL**, ***RST**, ***SAV**, ***SRE**, ***SRE?**, ***STB**, ***TST?**, or ***WAI**.

1.5.5 SCPI Parser

Standard 4894A/4804s include an extended SCPI parser that complies with the SCPI Standard Version 1995.0.

1.6 SERIAL INTERFACE

The 4894A provides either RS-232 single ended or RS-485 (RS-422) differential signals on a rear panel DB-25S connector. (4804 uses a 26-pin flat ribbon connector) The choice of signals is selected by internal jumpers. Signal pinouts conform to EIA specification RS-530 and are listed in Tables 1-2 and 1-3.

1.6.1 Baud Rates:

Baud Rate: Any rate from 50 to 115,200 baud. Parser selects closest rate to specified rate when non-standard rate entered. Standard rates are: 50, 110, 300, 600, 1200, 2400, 4800, 7200, 9600, 14400, 19200, 28800, 38400, 57600, 76800, 92160 and 115200 baud. (The 4804 is limited to 38400 baud).

1.6.2 Data Character Formats:

Data bits	7 or 8 data bits per character
Parity	Odd, even, none
Type	Asynchronous character
Stop bits	1 or 2 stop bits per character

1.6.3 Data Buffers:

Mode		4894A	4804
G Mode:	GPIB input buffer -	220 K	2 Kbytes
	Serial input buffer -	32 K	2 Kbytes
S Mode:	Serial input buffer -	252 K	4 Kbytes

1.6.4 Parity Generation/Checking

Transmitted characters may be sent with odd, even or no parity. Received characters may be checked for parity errors if parity is generated. If the check bit is on, the 4894A substitutes an asterisk character for a character with a parity or over run error.

1.6.5 EOM Character Detection/Change

The 4894A can be set to detect any received serial character as the end-of-message (EOM) character. The 4894A can also be set to add an additional character after the EOM character and to assert EOI when outputting the last character of the message to the GPIB bus.

1.6.6 RS-232 SPECIFICATIONS

The 4894A has single-ended RS-232C drivers and receivers and is designed to operate as a DTE device with up to 50 feet of cable. Handshake lines and/or X-on/X-off protocol may be used to control data flow. The RS-232C signals may also be used to interface the 4894A to devices using the newer RS-423 single-ended drivers and receivers.

Signals	See Table 1-2
Transmit Levels	+10 Vdc = Logic "0" or On -10 Vdc = Logic "1" or Off
Receive	±1.5 Vdc minimum, ±25 Vdc Maximum

TABLE 1-2 RS-232 SIGNAL ASSIGNMENTS

Pin #	Signal	Direction In out
1	Chassis	
2	Transmit Data	→
3	Received Data	←
4	Request-to-send	→
5	Clear-to-send	←
7	Ground	
8	Signal Detected	←
20	Data Terminal Ready	→

1.6.7 RS-485/RS-422 Specifications

The 4894A has balanced RS-485 line drivers and receivers that are RS-422 compatible. The line drivers and receivers are designed to operate with up to 1200 meters of twisted-pair cable. Two handshake lines and/or X-on/X-off protocol can be used to control data flow. The 4894A disables Send Data and Request-to-Send signals when not transmitting for connection to two-wire, half-duplex systems. The 4804's Send Data and Request-to-Send signals are always enabled.

Signals	Differential See Table 1-3 for pin assignments 4894A - Half-duplex mode 4804 - Full-duplex mode
Transmit Levels	+5 Vdc differential for binary 0 or On -5 Vdc differential for binary 1 or Off
Receive Levels	±0.2 Vdc minimum, ±25 Vdc maximum, differential or single-ended input with other input line biased at mid-range.

TABLE 1-3 RS-485 (RS-422) SIGNAL ASSIGNMENTS

Pin #	Signal	Direction in out
1	Shield	
2/14	Send Data	→
3/16	Received Data	←
4/19	Request-to-send	→
5/13	Clear-to-send	←
8/10	Signal Detected	←
20/23	Data Terminal Ready	→

1.7 PROGRAMMABLE FUNCTIONS

Standard 4894A/4804s use IEEE 488.2 and SCPI commands to change their operating configuration. Table 1-4 lists the programmable functions and their default settings. Space is provided in the 4894A/4804's nonvolatile memory for storage of 10 configurations (power-on plus 9 user assigned configurations).

Note that the 4894As with internal switches are configured by the switch settings and do **not** respond to SCPI or IEEE-488.2 commands.

1.8 INDICATORS

The 4894A/4804s have six LEDs that display the following conditions:

PWR - Indicates power on
RDY - Indicates unit has passed self test
TALK - Indicates unit has recognized its talk address
LSTN - Indicates unit has recognized its listen address
BUSY - On when receiving serial data
FULL - On when either data buffer is full

When the 4894A is turned on, it performs an internal self test which takes about 0.5 seconds. Only the PWR indicator is on during self test. At the end of the self test the 4894A shows its current GPIB address by blinking the front panel LEDs for one-half second. The LED bit weights are:

RDY	TALK	LSTN	BUSY	FULL
16	8	4	2	1

Any errors found during self test are indicated by a repeated blinking of the error code pattern. Refer to paragraph 6.4 for a description of the errors and their possible causes.

The BUSY LED will also be on if a serial device is not connected to the unit. See Section 2 for a serial connections and Section 5 for troubleshooting tests.

TABLE 1-4 4894A CONFIGURATION COMMANDS (for standard 4894A/4804s only)

Command	Functions	Factory Setting
:EMULation	Selects ICS 488.2 mode or NI CV emulation mode	ICS
:MODE	Selects G or S mode operation (4894A)	S G (4804)
:OPERation	Selects data transmission operation	DATA
:ADDRess	Sets GPIB bus address	4
:BAUD	Sets transmit/receive baud rate	9600
:PARity	Sets parity type	NONE
:CHECK	Enables parity checking	OFF
:BITs	Sets number of data bits per character	8
:SBITs	Sets number of stop bits/per character	1
:PACE	Enables/disables X-on/X-off protocol	NONE
:EOMchr	Sets end of message character	13(CR)
:ADD :CHARacter	Sets the character which the 4894A can add to the end of the received serial message	10(LF)
:ADD :ENAB	Enable character addition	OFF
:EOI	Enables EOI on last character of the received serial message	ON
:SWAP	Selects how the 4894A addresses a GPIB device to talk in the S mode	TIME
***ESE**	Enables Standard Event Status Register bits	0
***SRE**	Enables Status Byte Register bits	0

1.9 4894A PHYSICAL

Size	-	7.45"L x 5.57"W x 1.52"H (18.92 cmL x 14.15 cmW x 3.86 cmH) (See Figure 1-2 on page 1-12)
Weight	-	3 lbs (1.4 kg) including adapter
Temperature	-	Operating -10° C to +55° C Storage -20° C to +70° C
Humidity	-	0-90% RH without condensation
Shock/Vibration	-	Normal handling only
Construction	-	All metal case
Power	-	9 to 32 Vdc @ 3.5 VA
Connectors	-	IEEE 488 Interface Amphenol 57-20240 with metric studs RS-232 Interface Cinch DB-25S with lock studs

1.10 4894A CERTIFICATIONS OR APPROVALS

EMI/RFI CE	Meets limits for part 15, Class A of US FCC Docket 20780 and complies with EEC Standards EN 55022 and 50082-1. CE Certificate of Compliance reproduced in Figure 1-2.
UL/CSA/VDE	AC Wall adapter has applicable UL/CSA/VDE approvals.

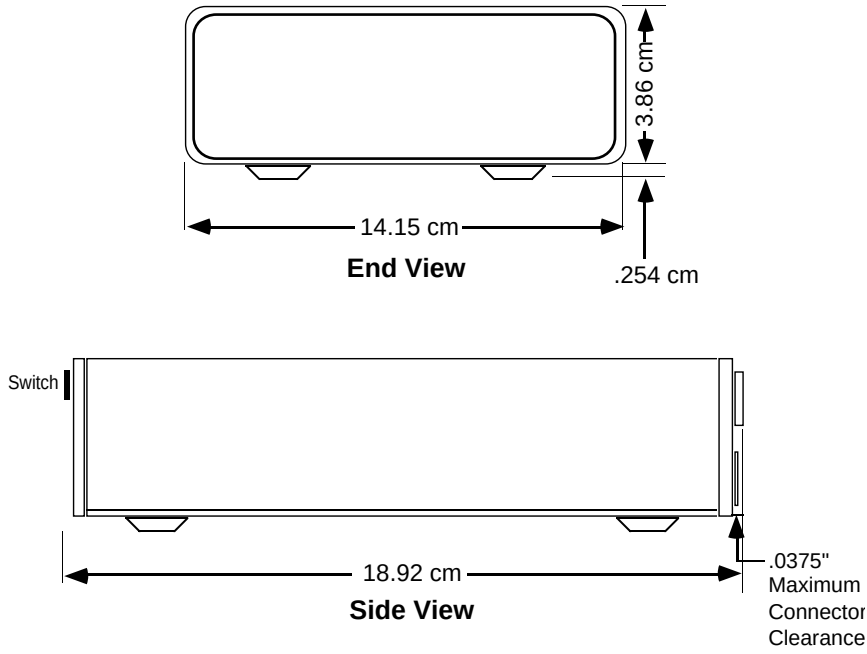


Figure 1-1 4894A Outline Dimensions

1.11 4894A INCLUDED ACCESSORIES

Instruction Manual
AC Wall Adapter, with applicable plug
3.5" Minibox Configuration Program Disk for PC and PC compatible computers.

Declaration of Conformity

Application of Council Directive(s).....89/336/EEC

Standard(s) to which Conformity is Declared.....EN 55022, EN 50082-1

Manufacturer's Name ICS ELECTRONICS CORPORATION

Manufacturer's Address 473 LOS COCHES STREET
MILPITAS, CA 95035-5422

Importer's Name

Importer's Address

Type of equipment GPIB TO SERIAL CONVERTER

Model No. 4894

Serial No. _____ Thru _____ Year of Manufacture 1995-1996

I, the undersigned, hereby declare that the equipment specified above conforms to the above Directive(s) and Standard(s)

Place Milpitas California USA

Date 11-05-95

(Signature)

G. IC. MERCOLA

(Full Name)

President

(Position)

Figure 1-2 4894 CE Certificate of Compliance

1.12 4804 PHYSICAL

1

Size	-	5.50"L x 4.50"W x 0.5"H (139.7 mmL x 114.3 mmW x 12.7 mmH) (See Figure 1-3)
Weight	-	6 oz. (0.17 kg)
Temperature	-	Operating -10° C to +70° C Storage -20° C to +85° C
Humidity	-	0-90% RH without condensation
Shock/Vibration	-	Normal handling only
Construction	-	Flame-retardant printed circuit board
Power	-	5 ± 0.25 Vdc @ 400 MA (typical)

Connectors GPIB - 24 pin male 3M 2524 connector
GPIB/Address Sw - 26 pin male 3M 2526 connector
Serial - 26 pin male 3M 2526 connector

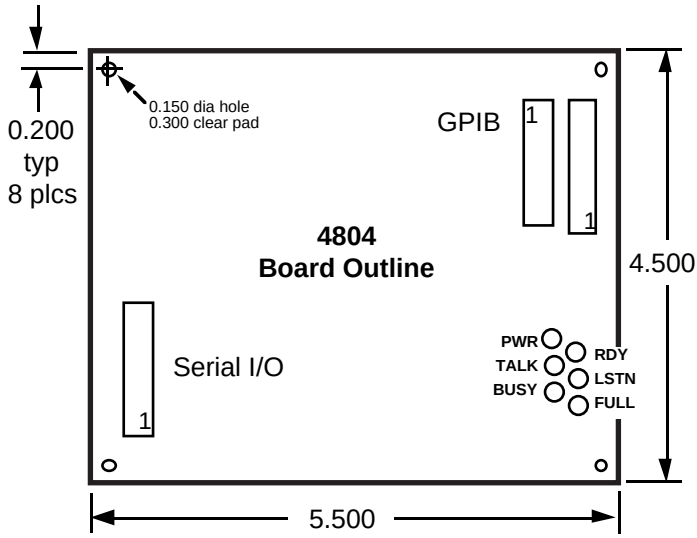


Figure 1-3 4804 Outline Dimensions

1.13 4804 INCLUDED ACCESSORIES

120117	4894A/4804 Instruction Manual
123045	Minibox GPIB Configuration Disk for PC and PC compatible computers.

1.14 OPTIONAL ACCESSORIES

120117	4894A/4804 Instruction Manual
123045	Minibox GPIB Configuration Disk
113640-L	Horizontal GPIB Connector/Address Switch Assembly (Dash number is cable length from 10 to 90 CM long. 90 CM standard)
113642-L	Vertical GPIB Connector/Address Switch Assembly (Dash number is cable length from 10 to 90 CM long. 90 CM standard)
114439-90	GPIB Flat Ribbon Extension Cable. (Dash number is cable length from 10 to 90 CM long. 90 CM standard)

INSTALLATION

2.1 UNPACKING

When unpacking, check the unit(s) for signs of shipping damage (damaged box, scratches, dents, etc.) If the unit is damaged or fails to meet specifications, notify ICS Electronics or your local sales representative immediately. Also, call the carrier immediately and retain the shipping carton and packing material for the carrier's inspection. ICS will make arrangements for the unit to be repaired or replaced without waiting for the claim against the carrier to be settled.

2.2 SHIPMENT VERIFICATION

Take a moment to verify the shipment. Each standard 4894A includes:

(1) Model 4894A or Model 4804 GPIB ↔ Serial Interface
(1) AC Power Adapter (Model 4894A only)
(1) 4894A/4804 Instruction Manual
(1) Minibox Configuration Program Disk

Model 4894-8s do not include a configuration disk.

Model 4804s do not include AC Power Adapters. Board only 4804 orders (Part# 114442) do not include manuals or configuration disks unless ordered separately.

2.3 4894A INSTALLATION GUIDE

The following steps should be used as a guide to setting up and using your 4894A.

1. If the 4894A is to be used with RS-422 or RS-485 signals, change the jumper settings as shown in Section 2.11.

2. See Section 2.10 to select and/or design the serial cable. CAUTION - "Standard" serial cables should not be used with the 4894A.

3. Review the factory settings in Table 1-4 to determine if your unit needs to be reconfigured. Note: 4894As are factory set to the S mode for applications without a GPIB Bus Controller.

4. Connect the AC adapter to the 4894A and to the AC power. Turn the unit on and verify that it passes its selftest and that it indicates the correct address. Turn the unit off.

5. If this is a standard 4894A and it needs to be reconfigured, follow the instructions in Section 2.5 to change its configuration. 4894A-8s with internal switches are configured by opening the unit and setting the internal switches as directed in Section 2.8.

6. If the unit is to go into a rack mounting kit, disconnect all cables from the unit. Follow the instructions in Section 2.12 to install it in the rack mounting kit.

7. Connect the unit to the GPIB bus, serial device and to the AC adapter. Turn the unit on and verify that it passes its selftest and that it indicates the correct address. Test the unit by passing data through it in both directions. See Section 5 for stand-alone test directions.

2.4 4804 INSTALLATION GUIDE

The following steps should be used as a guide to the 4804 installation.

1. If the 4804 is to be used with RS-422 or RS-485 signals, change the jumper settings as shown in Section 2.11.2.

2. Review Sections 2.9.2 and 2.10 to select and/or design the GPIB and serial interface cables.

3. Select a convenient location to mount the 4804. Do not mount it directly over a heat producing surface. Provide a 0.1 inch (2.5 mm) clearance underneath the 4804 or use an insulator if the 4804 is being mounted on a metal surface.

4. Use a twisted pair of #24 red/black wires to connect the 4804 to the host's +5 Vdc power supply. (Red is positive.) Connect directly to the power supply to avoid noise problems. Attach the wires into the terminal block on the 4804 PCB. Bypass diode CR1 if necessary so TP+5 = 5 ± 0.2 Vdc.

5. Plug in the GPIB and serial cables and connect the unit to the GPIB controller.

6. Apply power to the unit. Verify that it passes its selftest and that its indicates the correct GPIB address. (Address 4 if no external address switch was connected to the unit or if all rockers were off)

7. Review the factory configuration settings in Table 1-4 to see if the unit needs to be reconfigured. Follow the directions in Section 2.5 to configure the unit. In normal 4804 applications with a fixed serial device, it will not be necessary to change the settings again.

8. Test the unit by turning the unit off and back on. Send data through it in both directions to verify correct cable connections.

2.5 CONFIGURATION PROGRAMS

When shipped, Model 4894As and 4804s are configured as shown in Table 1-4. The configuration is stored in nonvolatile E2ROM and can be changed from any GPIB bus controller with SCPI commands. If your GPIB controller is a PC with DOS or Windows 3.1, and has one of the following GPIB controller cards, you can use the programs on the supplied Minibox Configuration Disk to walk you through a menu driven configuration procedure. Follow the instructions in Section 2.6 to install and use the configuration programs.

Program	**Supported GPIB Card**
mconfig.exe	ICS 488-PC2 Card National Inst. GPIB-PC2a Card (Set to address 2E1 and to 7210 emulation) or any NEC 7210 compatible GPIB Controller Card that is set to 2E1.
niconf.exe	National Instruments AT-GPIB Card (Set to address 2C0H)
hpconf.exe	Hewlett-Packard HP-IB Card (Set to address DC000)

If you are using a PC with Windows 95 and have one of the following cards installed in your computer, you can download the niconfig program from our web site at **www.icselect.com** and use it to configure the unit. Niconf_z.exe is a self-exploding zip file that installs a configuration program that makes National Instrument type calls to control the GPIB bus.

Program	**Supported GPIB Card**
niconfig	ICS 488-PCI or 488-PCMCIA Any National Inst Card Any Computer Boards Card

Follow the instructions in Section 2.7 to configure the 4894A/4804s from other computers.

2.6 USING THE CONFIGURATION PROGRAMS

Several Minibox configuration programs are available to walk the user through a menu driven program to configure the 4894A/4804's power-on settings. The configuration programs on the Configuration Disk run on any IBM type PC or compatible clone with DOS 6.0 or Windows 3.1 or later versions of teh operating system. A Windows 95 version of the configuration program can be downloaded from **www.icselect.com.**

2.6.1 Installing the Configuration Program from the Disk

Perform the following steps to install the configuration program on your hard disk.

1. Turn on the computer and select the directory where you want the configuration program.

2. Load the configuration disk into the floppy disk drive.

3. Read the README file to see if there are any changes to the program that may affect the configuration procedure.

4. Copy the selected configuration program from the floppy disk to your selected directory. Use the mouse to drag the desired .exe program to the directory on the hard disk or use the DOS copy command. Substitute the correct floppy drive letter for the letter **a** in the copy command.

   ```
   >cd c:/newdir              'go to new directory
   >copy a: mconfig.exe c:    'copy file
   ```

   When the installation is complete, remove the configuration disk from your floppy disk drive.

2.6.2 Installing a Downloaded Configuration Program

Perform the following steps to download and install the configuration program on your hard disk.

1. Use your internet browser to access the config.html page at www.icselect.com and to download the niconfig.exe file.

2. Place the file in a temporary directory and double click on it with the mouse to explode the file. Two of the exploded files will be a setup.exe and a readme file.

3. Follow the instructions in the included readme file to install the program in your computer.

2.6.3 Running the Configuration Program

The configuration programs support the standard configurable items. Special 4804 settings such as the user's IDN message will have to be entered with a live keyboard program or as part of the user's program. See sections 2.7 and 2.8.

1. Connect the 4894A to the GPIB controller card in the PC as shown in Figure 2-1. Plug the AC adapter plug into the DC jack on the 4894A's rear panel. Connect the AC adapter to an AC outlet.

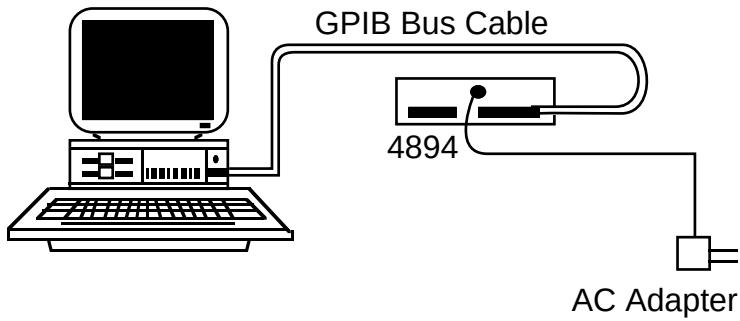


Figure 2-1 4894A Configuration Connections

2. Turn the 4894A's power switch on or apply power to the 4804. After 3 seconds, the unit should blink its GPIB bus address on the LEDs. The selftest ends with the PWR and RDY LEDs both on and the other LEDs off. (The BUSY LED may be on or blinking if nothing is connected to the serial port).

3. Run the configuration program. This may be done by double clicking on the program name or by typing the program's name at the DOS prompt or in the Windows Run command box.

> c:\new_directory\MCONFIG <return>

4. Product Selection

The program will display a list of model numbers. Enter or select the number that corresponds to the model that you are configuring and press return

e.g. 4894A <return> 'selects Models 4894A

The program will ask that you turn the unit off and back on. Press the Return key when the unit has finished its self test.

5. GPIB Address

The program branches to the selected product menu and asks for the unit's current GPIB address. Enter a one or two digit value ; i.e., 4, 04, 10. The factory default setting is 4.

Note - All units have a GPIB address between 0-30 for configuration purposes regardless of their operating mode or whether they have been set to address 31 for talk-only or listen-only operation. The unit will blink its GPIB address on the front panel LEDs at power turn-on. Add the bit weights to get the GPIB address.

RDY	TALK	LSTN	BUSY	FULL
16	8	4	2	1

6. Configuration Choices

The configuration program steps through each parameter and displays the current setting and configuration choices. The user should refer to the command definitions in Tables 3-2 and 3-3 to understand the command choices and there affect on the unit's operation. All setting changes are made by entering one of the displayed choices and pressing the Return key or clicking the Enter box in the Visual Basic versions. Pressing the Return key or clicking Enter without entering a new choice causes the program to skip to the next parameter leaving the current setting unchanged.

7. Saving the New Settings

After the last choice, the program will give you several configuration choices.

The program may give you the opportunity to set the SRE and ESE enable bit registers and to save the values so the unit can generate a SRQ at power turn-on. Enter **Y** to set PSC 0; **N** to set PSC 1 or click the appropriate boxes.

The program may ask if you want to lock the parameters so that they cannot be changed by the end user. The configuration program automatically unlocks the parameters whenever it is run. Enter **Y** to lock; **N** to continue or click the appropriate box.

The program will ask if you want to save the current configuration. Enter **Y** to save; **N** to continue or click the appropriate box.

8. Configuring other units

The program will ask if you want to configure another unit. Enter **Y** to configure another unit; **N** to exit.

2.7 CONFIGURING FROM OTHER CONTROLLERS

2.7.1 General Instructions

The 4894A/4804 can be configured from any GPIB bus controller by using the following procedure. The following example commands are shown in HP BASIC for easily conversion to another language.

1. Connect the unit to the bus controller as shown in Figure 2-1 Use an Abort, REN or a take control type command to have the bus controller assert REN. Then turn the unit on.

2. Determine the unit's GPIB address:

   a) For new units use the factory setting of 04.

   b) For other units, turn the unit off and back on. At the end of its self test, the unit blinks its GPIB address on the front panel LEDs.

RDY	TALK	LSTN	BUSY	FULL
16	8	4	2	1

3. Send the unit the escape sequence (**UNL LAD UNL LAD UNL)** to put it into the command mode. Provide a 30 ms delay after the escape sequence to allow the unit to set its internal logic. Use a **SEND** type command to output the GPIB commands in the escape sequence.

   i.e., SEND "UNL LISTEN 04 UNL LISTEN 04 UNL" or CMD "DD$"

   Note that the escape sequence in the above command is not an ASCII string of data characters. Instead it is a list of GPIB commands that should cause your GPIB Controller to output five GPIB bus command characters.

4. Verify the unit is in the command mode by sending it the *IDN? query and reading back the unit's IDN message. If the

IDN message comes back, the unit is in the command mode. If not, repeat steps 1 and 3.

5. Use Table 3-2 to put together the SCPI command for the parameter you want to change. Use an **OUTPUT** type statement to send the new configuration value to the unit. Follow each configuration statement with a query to verify that the unit accepted the new setting. The following example shows how to change the baud rate.

i.e., to change the baud rate

```
OUTPUT 704, "SYST:COMM:SER:BAUD  2400"
OUTPUT 704, "SYST:COMM:SER:BAUD?"
ENTER 704, B$
PRINT B$
```

6. Use caution when changing the unit's GPIB address. The change takes place immediately when the command is executed. Provide a 0.1 second delay before querying the new address setting.

i.e., to change the GPIB address to 20

```
OUTPUT 704, "SYST:COMM:GPIB:ADDR 20"
WAIT 0.1
' address the unit by its new address
OUTPUT 720, "SYST:COMM:GPIB:ADDR?
ENTER 720, A$
PRINT A$
```

Note: The 4894A responds to the SCPI **"SYST:COMM :GPIB:ADDR?"** query by outputting its current address setting. Units set to 31 for talk or listen-only operation are queried by addressing them at their prior primary address (0-30). The query response is 32 plus the prior primary address value.

```
e.g.  SYST:COMM:GPIB:ADDR?     'query
      04                       'response for address 4
```

If the unit is then set to address 31, its response would be:

36 '36 = 32 + 4

7. Use the ***SAV 0** command to save the new values in the unit's nonvolatile memory. The ***SAV 0** command stores the current configuration as the power-on values.

e.g. OUTPUT 704, "*SAV 0", END 'HP BASIC command

8. Return the unit to its data transfer mode either by turning it off and back on or by sending it the SCPI **SYST:OPER DATA** command. Wait 30 ms for the unit to set its internal logic before sending it any data.

e.g. **OUTPUT 704, "SYST:OPER DATA "**

NOTE: The **SYST:OPER DATA** command puts the 4894A into its transparent data mode. Any further SCPI or 488.2 commands will be treated as data until Step 3 above is repeated.

2.7.2 Labview Instructions

Labview users can execute the above sequence with the following commands:

SIC 'sends IFC and gets control of the bus
CMD ?$?$? 'sends escape sequence for address 04. (Change $ char to match the 4894A's current address. See Listen Address Group in Table A-5.)
wait 30 ms 'for 4894A to switch modes

GPIB Write 'configure unit
GPIB Read 'query unit

GPIB Write "*SAV 0" 'save configuration
GPIB Write "SYST:OPER DATA" 'switch to data mode

2.8 4894-8 SWITCH CONFIGURATION

Perform the following steps to change the 4894-8's configuration switch settings:

1. Disconnect any cables and power from the 4894-8

2. Unscrew the two screws holding the rear panel to the case.

3. Hold the rear of the case, tilt the front end up and the board assembly should slide back out into your hand.

4. Ground yourself before touching any components on the board to prevent damaging any of the CMOS components.

5. Locate the two dip switches. Switch U22 sets the GPIB parameters. Switch U20 sets the serial parameters. Review their setting against the switch rocker definitions in Tables 2-1 and 2-2. Mark the new settings along the bottom of the switches in Figures 2-2 and 2-3.

6. Use a tool like a small screwdriver to change the switches to the new settings.

6. While the unit is out of its case, connect the GPIB and serial cables. Apply power to the unit and verify it passes its self test routine. Send data through it to verify its operation. Make any changes to the switches as necessary. Turn power off and disconnect the cables.

7. Reassemble the unit by gently sliding the board assembly into its case. Use a rocking motion, not force, to guide the switch through the front panel opening. Install the two screws on the rear panel to secure the board assemble in the case.

2.8.1 GPIB Interface Settings- Switch U22

Switch U22 configures the parameters listed in Table 2-1.

TABLE 2-1 U22 ROCKER SETTINGS

Rocker	Position	Function	
1	OFF	Selects G mode operation	
	ON	Selects S mode operation	
		G Mode	**S Mode**
2-3	OFF OFF	Disables SRQ	Time termination
	OFF ON	Enables SRQ	CR Termination
	ON OFF	Disables SRQ	LF Termination
	ON ON	Enables SRQ	No swapping
4-8		Sets GPIB Address from 0 to 30 and listen-only. Rocker bit weights are: 16, 8, 4, 2 and 1 where rocker 4 has weight 16. All OFF = address 0. In G mode, all ON = listen-only operation. In S mode, all ON inhibits the unit from addressing the device on the GPIB bus.	

Standard 4894-8s are shipped from the factory with U22 set to G mode, SRQ disabled and address 04 as shown in Figure 2-2.

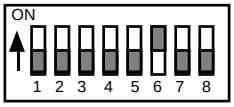


Figure 2-2 U22 Factory Setting

2.8.2 Serial Interface Settings - Switch U20

Switch U20 configures the parameters listed in Table 2-2..

TABLE 2-2 U20 ROCKER SETTING

Rocker	Position	Function
1	OFF	Disables X-on/X-off protocol
	ON	Enables X-on/X-off protocol
2	OFF	Selects 7 data bits
	ON	Selects 8 data bits
3	OFF	Selects one stop bit
	ON	Selects 2 stop bits
4	OFF	Disables parity generation/checking
	ON	Enables parity generation/checking
5	OFF	Selects odd parity if 4 is ON
	ON	Selects even parity if 4 is ON
6-8	-	Sets serial baud rate

6	7	8	Rate
OFF	OFF	OFF	300
OFF	OFF	ON	600
OFF	ON	OFF	1200
OFF	ON	ON	2400
ON	OFF	OFF	4800
ON	OFF	ON	9600
ON	ON	OFF	19200
ON	ON	ON	38400

Standard 4894-8s are shipped from the factory with U20 set to PACE off (no X-on/X-off protocol), 8 data bits, 1 stop bit, no parity and 9600 baud as shown in Figure 2-3..

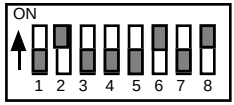


Figure 2-3 U20 Factory Setting

2.9 GPIB CONNECTIONS

2.9.1 4894A GPIB Connections

The 4894A has a standard GPIB 24 pin connector on its rear panel. Signal-pin assignments for the GPIB connector are shown in Figure A-2 in the Appendix. The GPIB connector mates with the standard IEEE 488 bus cables. If the 4894A or 4804 is being commanded by a GPIB controller (G-mode), it is connected as shown in Figure 2-4.

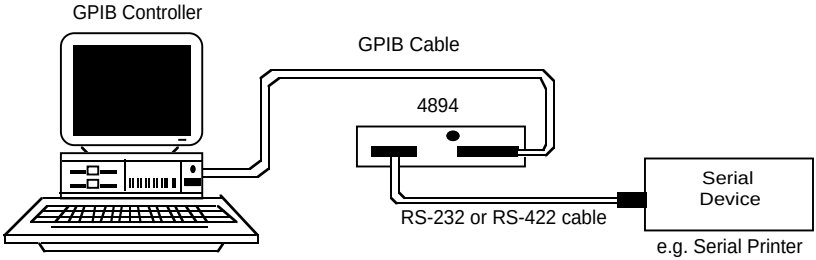


Figure 2-4 4894A G-Mode Connections

If the 4894A is being used to control a GPIB device (S mode), the unit is connected to the GPIB device as shown in Figure 2-5.

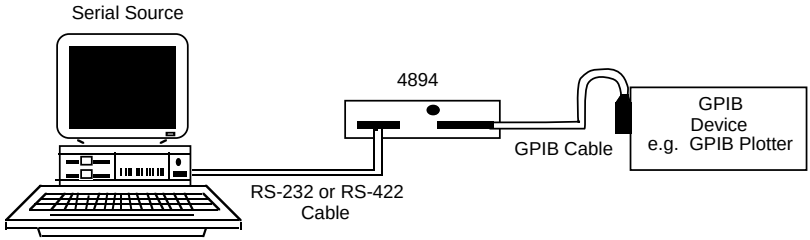


Figure 2-5 4894A S-Mode Connections

2.9.2 4804 GPIB Connections

The 4804 has two male ribbon connectors that can be used to connect the 4804 to the GPIB bus. Connector J1 is a 24-pin connector that is designed for direct connection to a GPIB bus connector. Connector J2 is a 26-pin connector that contains the address switch input signals as well as the GPIB bus signals. The 4804 only requires that one of the connectors be used to connect it to the GPIB bus. The unused connector can be left open.

2.9.2.1 GPIB Connector J1

The GPIB Signal-pin assignments for J1 are identical to the standard IEEE-488 connector shown in Figure A-2 of the Appendix. Connector J1 is laid out as shown in Figure 2-6 (a). Pin assignments are in Table 2-4. Use a flat ribbon cable with a 24-pin plug on one end and a GPIB connector on the other end. Mount the GPIB connector on the rear panel of the chassis.

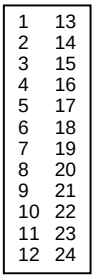
2.9.2.2 GPIB/Address Switch Connector J2

Connector J2 mates to one of ICS's GPIB Connector/Address Switch Assemblies. Signal-pin assignments for J2 and flat-ribbon wire colors are listed in Table 2-3. The connector layout is shown in Figure 2-6(b). The external address switch inputs are low true signals with pullup resistors on the 4804. At power turn-on, the 4804 checks the six address lines (ADSW1-ADSW5 and SISW) for a connection to ground. If they are all open, the 4804 defaults to its internal address setting. The T SW and L SW switch lines are not used by the 4804.

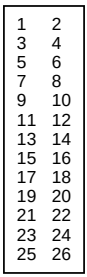
The GPIB Connector/Address Switch Assemblies are small, business card size, PC assemblies that mount a GPIB connector and an 8-bit rocker switch to the rear panel of a chassis. They have a 26 conductor, flat ribbon cable that plugs into the 4804. The assemblies are available in two layout styles. Refer to Appendix A3 for dimensions and installation instructions and silkscreen. Switch rockers functions are shown in Figure A-9. Figure 2-7 shows the switch set to address 4.

TABLE 2-3 4804 GPIB/Address Connector Signals (J2)

Signal	Pin Number	Wire Color	Bit Weights
GROUND	1	BRN 1	
ADSW5	2	RED 1	16 (MSB)
T SW	3	ORG 1	not used
L SW	4	YEL 1	not used
ADSW4	5	GRN 1	8
SI SW	6	BLU 1	0
ADSW1	7	VIO 1	1
ADSW3	8	GRY 1	4
ADSW2	9	WHT 1	2
NRFD	10	BLK 1	GPIB Signal
REN	11	BRN2	
DAV	12	RED 2	
IFC	13	ORG 2	
NDAC	14	YEL 2	
EOI	15	GRN 2	
ATN	16	BLU 2	
SRQ	17	VIO 2	
DIO1	18	GRY 2	
DIO2	19	WHT 2	
DIO3	20	BLK 2	
DIO4	21	BRN 3	
DIO5	22	RED 3	
DIO6	23	ORG 3	
DIO7	24	YEL 3	
DIO8	25	GRN 3	
GROUND	26	BLU 3	GPIB Signal

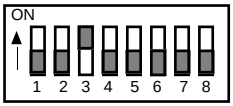


(a) J1 GPIB Layout



(b) J2 GPIB/Addr Sw Layout

Figure 2-6 4804 GPIB Connector Pin Layouts



Switch set to address 4

1 84 210 - - 6

Figure 2-7 Address Switch Rocker Assignments

2.10 4894A/4804 SERIAL INTERFACE AND CABLES

2.10.1 Signal Assignments

The 4894A's serial port is a DTE (Data Terminal Equipment) interface on a DB-25S female connector. The 4804 has a similar interface but uses a 26-pin male ribbon connector (Figure 2-8) that connects on a pin-to-pin basis to a 25-pin connector. The Both connectors contain RS-232 and RS-485 signals in accordance with EIA-STD - RS-530. RS-232 and RS-485 signal selection is made by setting jumpers inside the 4894A or on the 4804 board. Refer to section 2.11 for jumper setting instructions. Table 2-4 shows the signal-pin assignments and the signal directions.

TABLE 2-4 4894A/4804 SERIAL CONNECTOR PIN ASSIGNMENTS

Pin	RS-232	RS-422 RS-485	Signal	Direction In Out
1	AA	—	Chassis	
2	BA	SD(A)	Send Data (A)	→
3	BB	RD(A)	Receive Data (A)	←
4	CA	RS(A)	Request-to-Send (A)	→
5	CB	CS(A)	Clear-to-Send (A)	←
6			Data Set Ready	
7	AB		Ground	
8	CF	RR(A)	Signal Detected (A)	←
9				
10		RR(B)	Signal Detected (B)	←
11				
12				
13		CS(B)	Clear-to-send (B)	←
14		SD(B)	Send Data (B)	→
15				
16		RD(B)	Receive Data (B)	←
17				
18				
19		RS(B)	Request-to-send (B)	→
20	CD	TR(A)	Data Terminal Rdy (A)	→
21				
22				
23		TR(B)	Data Terminal Rdy (B)	→
24				
25				
26				

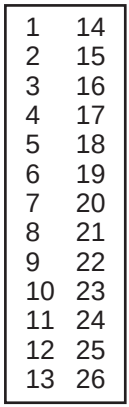


Figure 2-8 4804 Serial Connector Pin Layout

2.10.2 4894A Cable Selection

The user must determine what kind of a device that is being connected to the 4894A to select the proper cable. **Do not use off the shelf, pin-to-pin cables to connect the 4894A to other devices without first verifying all signal connections.** Table 2-5 lists several common serial devices, their interface types and suggested cables. To interface to other serial devices, consult their manual to determine the their interface type and their control lines requirements. Use the information in Table 2-4 and in the remainder of this section to design the serial cable. Refer to Appendix A2 for additional information on serial communication and on interfacing to serial devices.

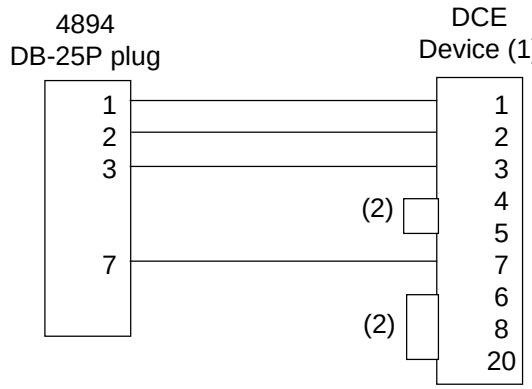
TABLE 2-5 SERIAL DEVICE INTERFACE TYPES

Serial Device	Interface Type	Refer To	Notes
PC Comm Port 1	DTE	2.10.4	*
PC Comm Port 2	DTE	2.10.4	
Terminal	DTE	2.10.4	
Modem	DCE	2.10.3	
Printer	DTE	2.10.4	
Mainframe	DCE	2.10.3	
4894A to 4894A	DTE	2.10.5	RS-422

*Use 9 to 25 pin adapter

2.10.3 Cables - 4894A to RS-232 DCE Devices

The 4894A may be connected to DCE devices by a very simple pin-to-pin cable that carries only 4 signals: Transmit Data, Receive Data, ground and shield. Data flow can be controlled by the X-on/X-off protocol (Configure PACE on for X-on/X-off protocol).

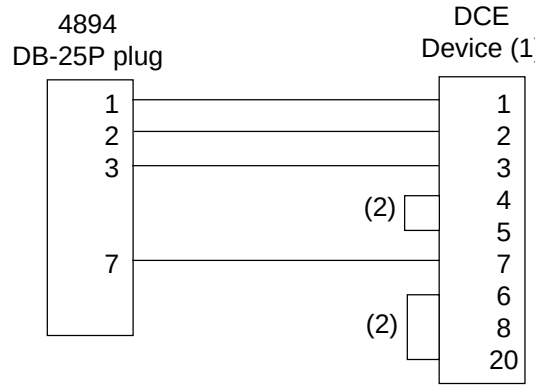


Notes (1) Select connector type to match serial device.

Figure 2-9 4894A to DCE Device Cable with Control Signals

2.10.4 Cables - 4894A to RS-232 DTE Device

To use hardware handshake signals to control serial data flow, add the Request-to-Send and Clear-to-Send lines to the cable as shown in Figure 2-10. Add Data Terminal Ready at the same time. Configure PACE OFF or XON.

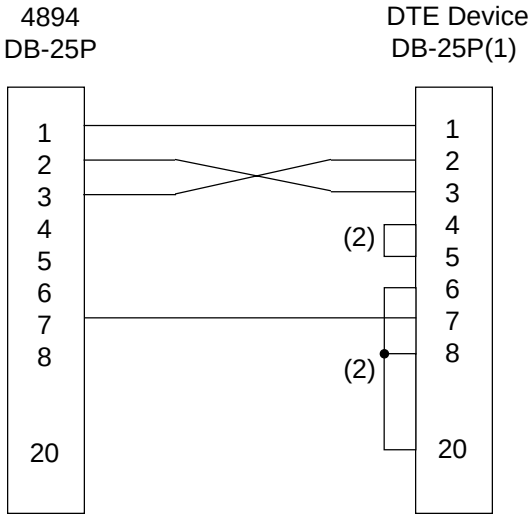


Notes (1) Select connector type to match serial device. (2) Add jumpers

Figure 2-10 4894A to DCE Device Cable with Control Signals

2.10.4 Cables - 4894A to RS-232 DTE Devices

The 4894A is connected to a DTE device by crossing over the signals in a 'null-modem' configuration cable. Figure 2-11 shows the minimum signal connections: Transmit Data, Receive Data, Signal Ground and Chassis Ground. . Configure **:PACE XON** or NONE.

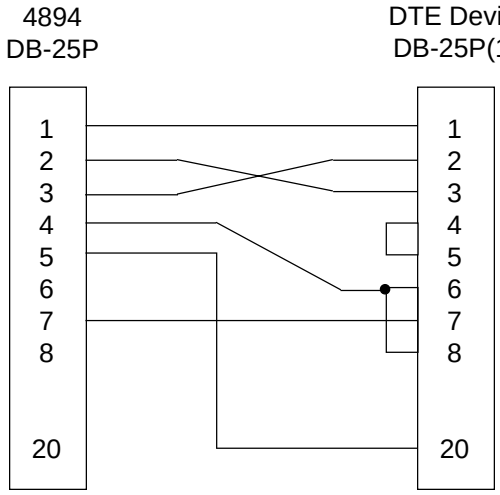


Notes (1) Select connector type to match DTE device.
(2) Add jumpers if required by the DTE device.

Figure 2-11 Minimum RS-232 4894A - DTE Device Cable

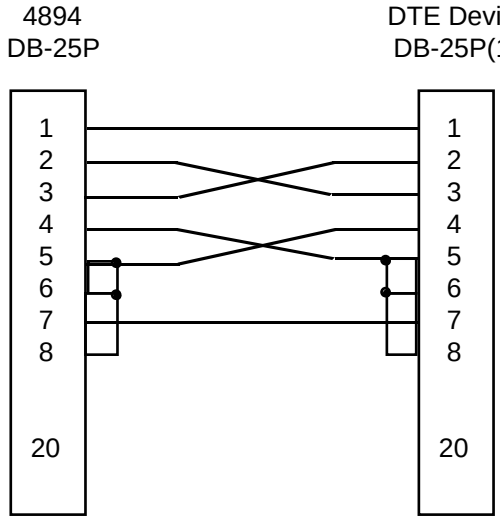
Many DTE devices use Data-Set-Ready and Data-Terminal-Ready to control the data flow. To add hardware flow control, add the Request-to-Send and Clear-to-Send control lines to the cable. Figure 2-12 shows how the 4894A control signals are added to a DTE cable.

When data flow is controlled by CTS (pin 5) and RTS (pin 4) in the DTE device (as the case is with the 4894A) connect the signals as shown in Figure 2-13. The cable in Figure 2-13 can be used to connect two 4894As in a back-to-back installation for distances up to 50 feet.



Notes (1) Select connector type to match DCE device.

Figure 2-12 4894A to DTE Device Cable with Control Signals type 1



Notes (1) Select connector type to match DTE device.

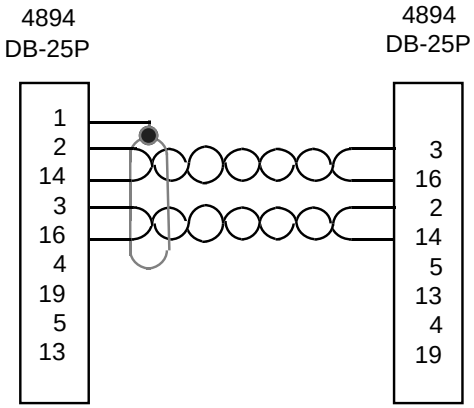
Figure 2-13 4894A-to-4894A Cable with Control Signals type 2 (ICS P/N 113553)

2.10.5 Cables - Full Duplex RS-422/RS-485 Devices

Cross connect the transmit and receiving signals for a minimum configuration full duplex cable. Configure PACE on if necessary to control the signal flow of long ASCII character strings. **Do not use PACE (X-on and X-off characters) to control the flow of binary data.** Note that the 4894A disables its Send Data lines when not transmitting. Pullup and pulldown resistors may have to be added to a 4894A cable as described in paragraph 2.10.6 if the receiving device has trouble distinguishing noise from true 4894A data messages. The minimum RS-422 connections are:

4894A/4804	**Other Devices**
Send Data (A)/(B)	Receive Data (A)/(B)
Receive Data (A)/(B)	Send Data (A)/(B)

The cable in Figure 2-14 can be used to connect two 4894As in a back-to-back installation for distances up to 1200 meters.



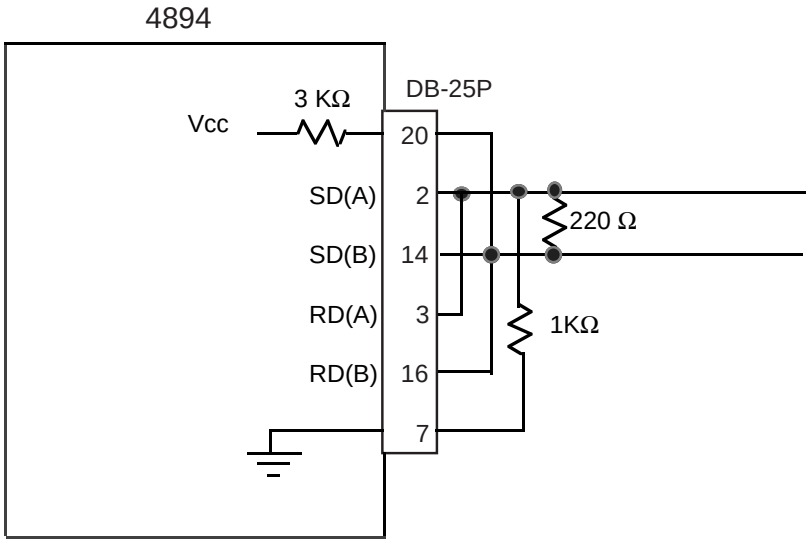
Notes
1. Dashed wires are for optional hardware flow control.
2. Use twisted shielded pairs for best noise immunity. Connect shields to pin 1 at one end.

Figure 2-14 RS-485 4894A-to-4894A Cable (ICS P/N 113554)

2.10.6 Cables - Half Duplex Connections

Some RS-485 devices operate in a half-duplex mode and use only two wires. These lines are often labeled "+" and "-". In a half-duplex 485 system, multiple devices can be connected to the pair of lines. One RS-485 device transmits while the other devices receive. The 4894A disables its transmitter when not transmitting so it can be connected to a two-wire, RS-485 system. 4804s cannot be connected to a two-wire network.

For a half-duplex connection, the 4894A's transmit and receive pairs need to be jumpered together in the cable plug. Because the 4894A always receives anything that is sent on the line, including its own messages, the user will have to read and discard his own message from the receive buffer after a transmission. The RS-485 lines will need to be biased to prevent other units from receiving noise when none of the transmitters are active. A suggested termination circuit that uses the 4894A's internal 3 KΩ resistor is shown in Figure 2-15. This circuit provides 0.26 volts across the load resistor when the transmitters are tristated.



Notes 1. Connect pins 10 and 13 to ground on pin 7

Figure 2-15 RS-485 Half-Duplex Connections

2.11 RS-232 - RS-485 SIGNAL SELECTION

2.11.1 4894A RS-232/RS485 SIGNAL SELECTION

The 4894A has six internal jumpers which are factory positioned as shown in Figure 2-16. The jumpers are accessed by removing the two screws on the rear panel and sliding the unit out from the rear of its case.

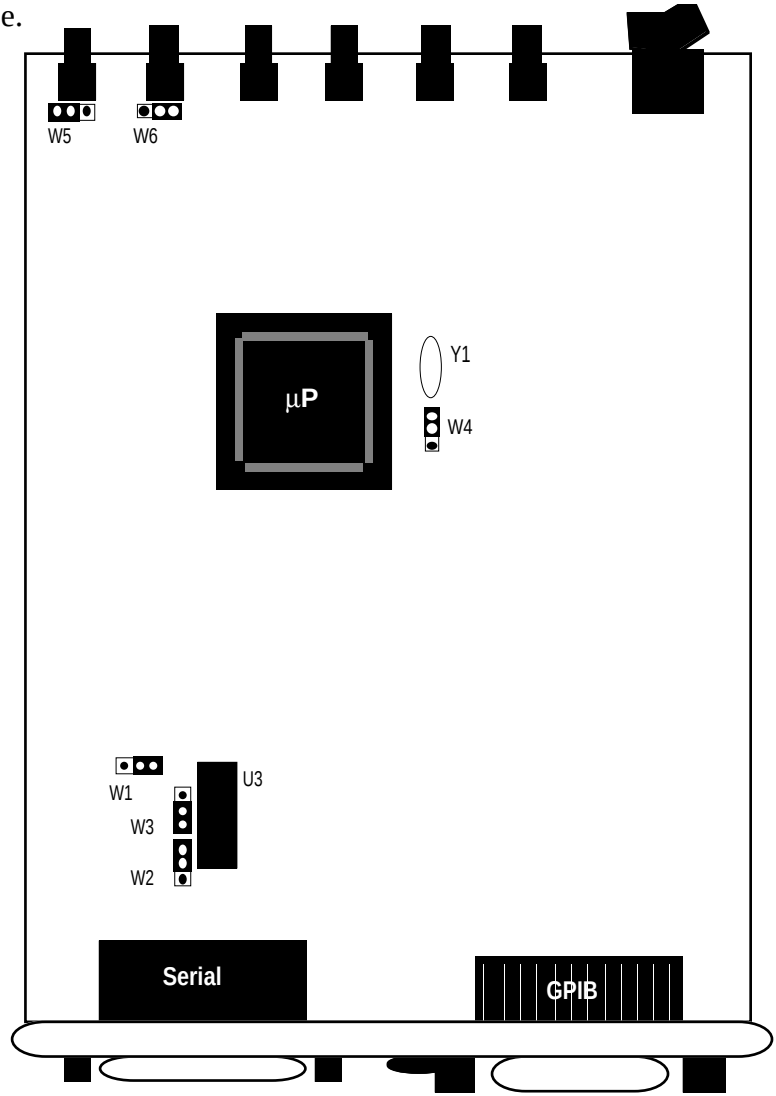


Figure 2-16 4894A Internal Jumpers with factory set positions (RS-232 serial signals)

Jumpers, W1, W2 and W3 select the type of signals for the 4894A's serial interface. For RS-232 serial signals, leave the jumpers positioned as shown in Figure 2-16. For RS-422 or RS-485 serial signals, set jumpers W1, W2 and W3 as shown in Figure 2-17. Do not change the other jumper positions. Carefully slide the unit back into its case, fitting the PC board into the tracks on each side of the case. Be sure the switch and LEDs are correctly aligned before pushing the switch through the front panel. Replace the two rear panel screws.

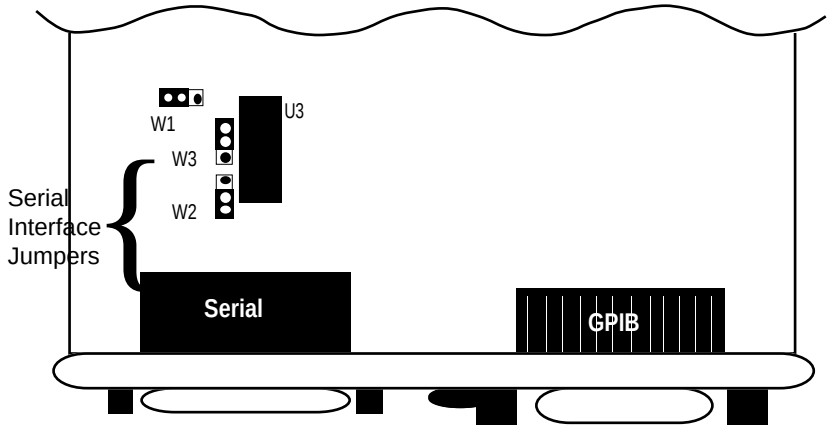
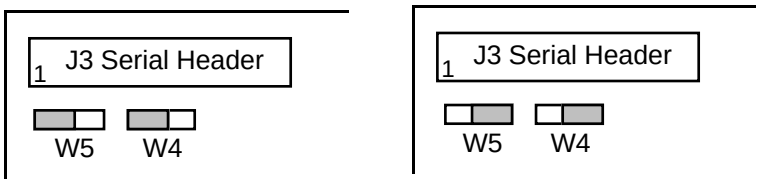


Figure 2-17 4894A Internal Jumpers set for RS-422/ RS-485 serial signals

2.11.2 4804 RS-232/RS-485 Signal Selection

The 4804 has two jumpers to select the type of serial output signals. The jumpers are factory set for RS-232 output signals as shown in Figure 2-18(a). For RS-485 signals, position both jumpers as shown in Figure 2-18(b).



(a) RS-232 Positions (b) RS-422/RS-485 PositionS

Figure 2-18 4804 W4 and W5 Jumper Positions

2.12 4894A Rack Mounting Instructions

The Model 4894A is held in its rack mounting kit with a winged-'U' shaped bracket. Perform the following steps to install a 4894A in a rack mounting kit:

1. Hold the 4894A at a 30 degree nose down angle and place the front bezel through the rack mount kit from the rear of the kit. Push it forward through the opening until the rubber feet line up with the holes in the rack mounting kit. Push the unit down until it rests flat on the kit and the feet are in the four holes.

2. Repeat step 1 for a second unit if two units are being held in one rack mounting kit.

3. Aline the unit(s) so the bezels are parallel with the front of the rack mount kit and protrude equally through the front panel of the rack mounting kit.

4. Set the bracket so its two holes line up with the holes in the rack mounting kit extrusion. Use the supplied 4-40 screws to hold the bracket to the extrusion. Do not overtighten.

5. Use the supplied 10-32 screws to bolt the rack mounting kit into the rack.

3

Operating From The GPIB Bus

3.1 INTRODUCTION

This section describes how the 4894A/4804 operates and transfers data when controlled from the GPIB bus. This section also describes the SCPI commands used to configure and control the 4894A/4804. Wherever the text refers to the Model 4894A, it applies equally to the Model 4804 unless otherwise noted.

3.2 G MODE OPERATION

When either unit is controlled from the GPIB bus it is operating in the G Mode. In the G Mode, data passes transparently through the unit in both directions but commands are only accepted from the GPIB bus. In the G Mode, the 4894A transfers data when addressed to talk or listen but will also accept data when configured for operation as a listen-only device.

Data received at the GPIB port is transferred directly to the GPIB buffer by the 4894A's high speed DMA controller. The buffered data is then outputted on the serial interface at the rate governed by the external device. Both data transfers take place simultaneously and without user interaction.

Any serial data that is received by the 4894A is transferred to the serial buffer and is later output to the GPIB bus when the unit is addressed to talk.

Data flow on the 4894A's serial interface is always controlled by hardware handshake lines and optionally by the software controlled X-on/X-off protocol. The 4894A uses its 'clear-to-send' input to sense the condition of the serial devices's input buffer. The 4894A sends when the input is high and stops outputting when the line is low. An open input appears as a high input to the 4894A. When the 4894A's serial buffer gets full, it drops its 'request-to-send' output to stop the serial device from sending any more data until there is space in the buffer.

The 4894A enables its RS-485 output signals only when transmitting so the RS-485 'request-to-send' output should not be used to control data flow. The 4804 keeps the RS-485 signals always enabled.

If the X-on/X-off protocol is enabled, the 4894A sends these characters to the serial device to control its serial output. The 4894A also senses the X-on/X-off characters from the serial device to control its own serial transmission. The 4894A starts in the X-on mode and assumes the remote device does likewise. X-on/X-off protocol should only be used with ASCII characters and never when binary data is being sent over the serial interface. Refer to appendix A2 for more information on serial data transmission and flow control.

At power turn-on, the 4894A defaults to its data transfer mode. In the data transfer mode, the unit treats everything sent to it as data and only responds to IEEE-488.1 commands i.e. Serial Poll, Device Clear etc. When addressed to talk, it only outputs data from its serial buffer.

The 4894A must be put into the command mode by sending it the escape sequence before it will recognize SCPI and IEEE-488.2 commands. Once in the command mode, the unit will accept configuration parameters, respond to queries, and save new configuration settings. The 4894A has ten configuration Save Areas. Save Area 0 stores the power turn-on default values. Save Areas 1 through 9 are available for the user to store custom setups. The 4894A must be put back into the data mode before transferring data through the unit.

The Model 4894-8 equipped with internal switches is always in the

data mode and has no response to IEEE-488.2 or to SCPI commands.

3.3 ADDRESSING THE UNITS

3.3.1 4894As and 4804s

The 4894A and 4804s can be set to any unused GPIB primary address between 0 and 30. The Bus Controller will use the primary address to address the unit as a talker or as a listener.

The 4894A can be set to a listen-only mode by setting its GPIB address to 31. In the listen only mode, the 4894A accepts every data character present on the bus and then outputs them on its serial interface. When set to address 31, the 4894A saves its prior primary address so it has an address to recognize when it is reconfigured. The user should record the 4894A's prior address for future reconfiguring.

If you have forgotten the GPIB address, momentarily turn the unit off and back on. At the end of the self test, the unit will blink its GPIB address on the front panel LEDs using the following bit weights:

PWR	RDY	TALK	LSTN	BUSY	FULL
-	16	8	4	2	1

If the 4894A is going to be used in listen-only application where it is connected to a bus controller, its prior primary address should not be used by another device in the system. If the prior primary address is being used, then the unit needs to be set to an unused bus address to prevent it from responding to the bus controller. First set the 4894A to the unused bus address and save the configuration. Next set the unit to address 31 and again save the configuration. Refer to Sections 2.6 and 2.7 for complete configuration instructions.

Because of the transparent nature of the 4894A, an escape sequence is used to put the 4894A into its command acceptance mode where it responds to the 488.2 and SCPI commands. The escape sequence is **UNL LAD UNL LAD UNL** followed by a short 30 millisecond pause before sending it data or addressing it to talk.

The 4894A will stay in the command acceptance until given the return-to-data command (**SYST:OPER DATA**) or is powered off. An example of the escape sequence before reading the ID message in HP BASIC is:

```
SEND "UNL LISTEN 04 UNL LISTEN 04 UNL"
WAIT 30
OUTPUT 704; "*IDN?"
ENTER 704; ID$
```

Where LISTEN 04 is the equivalent of LAD (listen address)

The unit is returned to its data transfer mode by sending it the SCPI **SYSTem:OPERation DATA**command or by turning it off and back on. An example of using the command before sending or receiving data is shown below.


```
OUTPUT 704, "SYST:OPER DATA"
WAIT 30
OUTPUT 704; "Serial message"
ENTER 704; Message$
```

3.3.2 Additional 4804 Address Setting Methods

In addition to the GPIB bus method described in 3.3.1, the 4804's GPIB address can also be set by connecting its GPIB/ Address Connector (J2) to an external rocker switch or by sending it a serial command.

If the 4804 senses an external address switch at power turn-on time, the value of the external switch overrides the value stored in the 4804's E^2PROM and becomes the 4804's current GPIB address. The switch value is only sampled at power turn-on time and power must be recycled if the external switch setting is changed. Address 0 is set by turning rocker S on.

If the 4804 is sent the serial address message it will change its GPIB address and save the new address in the unit's E^2PROM. The serial address message starts with @@@ followed by a two digit address from 00 to 30. An example is: **@@@07**

3.4 488.2 STATUS REPORTING STRUCTURE

The 4894A and 4804 include an expanded status reporting structure (shown in Figure 3.1) that conforms to the SCPI 1994 Specification and the IEEE 488.2 Standard. This status reporting structure is enabled when the unit's emulation is set to ICS (factory default setting) or when the unit is in the command mode.

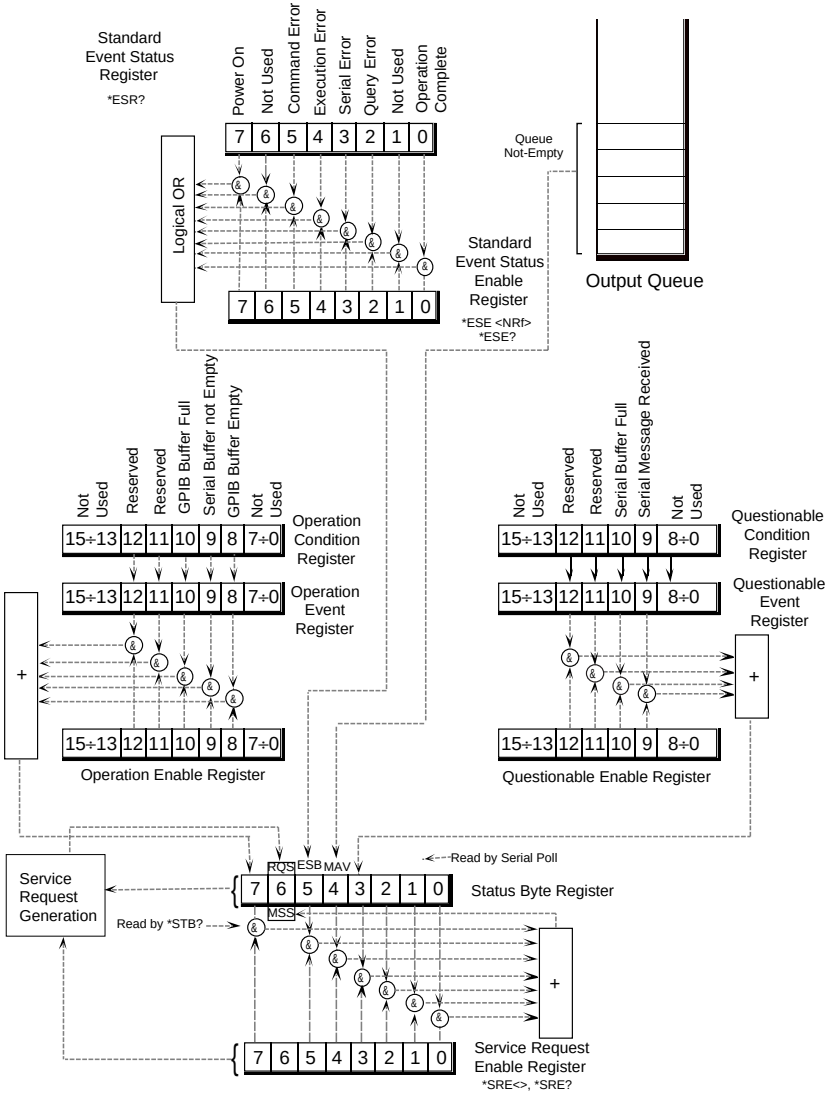


Figure 3-1 4894A Status Reporting Structure

As shown in Figure 3.1, SRQ generation is a multilevel function and is determined by the occurrence of an event that has its corresponding enable bit set to '1'. SRQs (SRQ line pulled low) are used by the 4894A to signal the bus controller that an event has occurred and/or that the 4894A needs service. There are four major sources of SRQs, each of which is summarized in a bit in the Status Byte Register, three of which are event registers with their own enabling bits and an output queue. The Standard Event Status Register reports events that are common to all 488.2 devices. The Operation register structure reports on the status of the GPIB and serial buffers. The Questionable register structure reports when serial messages are received or the serial buffer is full. The Output Queue is used by the 4894A to send messages back to the bus controller. (These messages are responses to queries sent by the controller.)

A Condition Register reflects the **real time** condition of its individual bits. To read a Condition Register use the following SCPI queries:

STAT:OPER:COND?
STAT:QUES:COND?

Reading a Condition Register doesn't change its contents.

An Event Register **captures 0 to 1 transitions** in its associated condition register. Each event bit has a corresponding bit in its condition register. An event bit becomes TRUE (1) when the associated condition bit makes **0 to 1 transition.** Once an event bit is set it **cannot be cleared** until the Event Register is read.

To read an event register use the following commands:

	STAT:OPER?	reads Operation Event Register
or	**STAT:OPER:EVEN?**	
	STAT:QUES?	reads Questionable Event Register
or	**STAT:QUES:EVEN?**	
	***ESR?**	reads Standard Event Status Register

Reading an Event Register clears its contents. The 488.2 ***CLS** command clears all Event Registers.

Each Event Register structure contains eight or sixteen bits and each bit represents a unique condition or problem. A binary '1' indicates presence of the condition. When the register is read, its binary equivalent corresponds to the state (1 or 0) of the bits in the register:

e.g. 23 decimal = 0001 0111 binary

Each Event Register bit has a corresponding enable bit. The enabling bits are ANDed with the state of the event bits to create the summary condition in the Status Byte Register. Unwanted conditions can be blocked from generating SRQs by setting their corresponding enabling bit to a '0'. The enabling bits are set by writing the equivalent decimal value to the desired enabled register.

The Service Request Enable Register is an 8 bit register that enables corresponding summary bits in the Status Byte Register.

Examples:

***ESE 52** Sets the Standard Event Status Enable Register to 00110100.

This pattern blocks serial error from generating SRQ while enables Query, Execution & Command errors to generate SRQs.

***SRE 40** Sets the Service Request Enable Register to 00101000.

This pattern blocks the Operation Status structure and the Output Queue from generating SRQs while enables the Standard & Questionable structures to generate SRQs.

The Output Queue is simpler structure in that it just reports a '1' in bit 4 of the Status Byte Register when it contains a message(s) to be read by the bus controller.

3.5 488.2 CONFORMANCE INFORMATION

The IEEE 488.2 Standard mandated a list of common commands that are common to all IEEE 488.2 compatible devices. The 4894A and the 4804 respond to all of the common commands and to some optional commands defined in IEEE-488.2. Table 3-1 lists the commands, the unit's response and describes the commands affect on the 4894A/4804 and the status reporting structure.

The Model 4894-8 with internal switches does not respond to the IEEE-488.2 common commands.

TABLE 3-1 IEEE-488.2 COMMON COMMANDS

COMMAND	NAME	DESCRIPTION
*** CLS**	Clear Status	Clears all event registers summarized in the status byte, except for "Message Available," which is cleared only if *CLS is the first message in the command line.
***ESE <value>**	Event Status Enable	Sets "Event Status Enable Register" to <value>, an integer between 0 and 255. <value> is an integer whose binary equivalent corresponds to the state (1 or 0) of bits in the register. If <value> is not between 0 and 255, an Execution Error is generated. EXAMPLE: decimal 16 converts to binary 00010000. Sets bit 4 (EXE) in ESE to 1.
***ESE?**	Event Status Enable Query	4894A returns the <value> of the "Event Status Enable Register" set by the *ESE command. <value> is an integer whose binary equivalent corresponds to the state (1 or 0) of bits in the register.
***ESR?**	Event Status Register Query	4894A returns the <value> of the "Event Status Register" and then clears it. <value> is an integer whose binary equivalent corresponds to the state (1 or 0) of bits in the register.
***IDN?**	Identification Query	4894A returns its identification code as four fields separated by commas. These fields are: manufacturer, model, six-digit serial number and version of firmware - e.g. **ICS Electronics Corp., 4894A, S/N 012123, Rev. 0.1 (2-04-92)**
***OPC**	Operation Complete Command	Causes the 4894A to generate the operation complete message in the Standard Event Status Register when all pending selected 4894A operations have been finished.

TABLE 3-1 IEEE-488.2 COMMON COMMANDS (CONTINUED)

COMMAND	NAME	DESCRIPTION
***OPC?**	Operation Complete Query	Places an ASCII character 1 into the 4894A's Output Queue when all pending selected 4894A operations have been finished.
***RCL<value>**	Recall	Restores the state of 4894A from a copy stored in its E^2PROM by *SAV command. *RCL 0 restores the power on setting. *RCL 1 through RCL 9 recalls setups saved by the corresponding *SAV command. <value> is an integer in the range 0-9.
***RST**	Reset	4894A restores its power-up state except that the state of IEEE-488 interface is unchanged, including: 1) instrument address, 2) Status Byte and, 3) Event Status Register.
***SAV <value>**	Save	Saves current 4894A configuration in the E^2 PROM. *SAV 0 saves the current setting as the new power on setting. *SAV 1 through *SAV 9 saves special setups. <value> is an integer in the range 0-9.
***SRE <value>**	Service Request Enable	Sets the "Service Request Enable Register" to <value>, an integer between 0 and 255. The value of bit six is ignored because it is not used by the Service Request Enable Register. <value> is an integer whose binary equivalent corresponds to the state (1 or 0) of bits in the register. If <value> is not between 0 and 255, an Execution Error is generated.
***SRE?**	Service Request Enable Query	4894A returns the <value> of the "Service Request Enable Register" (with bit six set to zero). <value> is an integer whose binary equivalent corresponds to the state (1

TABLE 3-1 IEEE-488.2 COMMON COMMANDS (CONTINUED)

COMMAND	NAME	DESCRIPTION
***STB?**	Read Status Byte	or 0) of bits in the register. 4894A returns the <value> of the "Status Byte" with bit six as the "Master Summary" bit. <value> is an integer whose binary equivalent corresponds to the state (1 or 0) of bits in the register.
***TST?**	Self-Test Query	Causes the 4894A to run internal self-test. Test takes about 9 seconds. Only the PWR LED is lit during the test. Blinking LEDs indicate a failure. No user interaction is required. The number returned corresponds to a state described in Section 6 which lists the error codes and faults. A zero response indicates no test failures. Other responses are listed in Table 5-2. Doing a self test clears all data buffers.
***WAI**	Wait-to-continue	Prevents the 4894A from executing any further commands or queries until the No-Operation-Pending flag is TRUE.

3.6 SCPI CONFORMANCE INFORMATION

The 4894A and 4804 accept SCPI commands and command extensions to setup their operating parameters. The commands conform to SCPI Standard 1994 and give the user a quick way of setting or learning the 4894A's configuration. At power turn on, the 4894A reads the last configuration values stored in its E^2PROM by ***SAV 0** as the current configuration. These values are used until a command is sent to the 4894A with a replacement value which then becomes part of the current configuration. When power is turned off, the current configuration values are lost. The ***SAV** command must be used to save the current configuration in the 4894A's E^2PROM.

Note that Model 4894-8s do not respond to SCPI commands.

The 4894A's Configuration commands conform to SCPI's hierarchal 'tree like' structure. Each command can be used as a query except where noted. Table 3-2 shows the 4894A's command tree. The letters in capitals are the abbreviated form of the keyword. Either the abbreviated or whole keyword may be used when entering a command:

e.g., **SYSTEM:COMMUNICATE:SERIAL:BAUD 1200**

is the same as

SYST:COMM:SER:BAUD 1200

and

SYST:COMM:SER:BAUD?
is the query form of the command

Table 3-3 lists the SCPI keywords and describes their functions in detail. Keywords other than those listed in the table will have no effect on the 4894A's operation and a command error will be reported.

TABLE 3-2 4894A SCPI COMMAND TREE

Keyword	Parameter Form	Notes
SYSTem		
:EMULation	ICS | NI	
:MODE	G | S	
:OPERation	COMMAND | DATA	
:COMMunicate		
:SERial		
[:RECeive]		
:BAUD	<numeric value>	
:PARity		
[:TYPE]	EVEN | ODD | NONE	
:CHECk	<Boolean>	
:BITS	<numeric value>	
:SBITs	<numeric value>	
:PACE	NONE | XON	
:EOMchr	<numeric value>	ASCII character
:ADD		
:CHARacter	<numeric value>	ASCII character
:ENABle	<Boolean>	
:EOI	<Boolean>	
:BUFFer?	[query only]	
:RS485	<Boolean>	
:GPIB		
:ADDRess	<numeric value>	
:SWAP	TIME | CR | LF | NONE	
:BUFFer?	[query only]	
:ERRor?	[query only]	
:VERSion?	[query only]	1991.0
STATus		
:OPERation		
[:EVENt]?	[query only]	
:CONDition?	[query only]	
:ENABle	<numeric value>	
:QUEStionable		
[:EVENt]?	[query only]	
:CONDition?	[query only]	
:ENABle	<numeric value>	
:PRESet	[no query]	

TABLE 3-3 SCPI COMMANDS AND QUERIES

Keyword	Default Value	Description
SYSTem	-	Starts 4894A SCPI command branch
:EMULation	ICS	Selects ICS 488.2 or NI-CV emulation. Values are: ICS | NI
:MODE	G	Sets Interface which controls the 4894A's operation. Assertion of REN and IFC force the 4894A to G Mode for programming. Values are G | S
:OPERation	DATA	Returns the 4894A to its transparent data transmission and ends command acceptance operation. Power on state is DATA operation. COMMAND has no effect on the unit's operation but is provided as the query message since there is no responses to SCPI commands in DATA operation.
:COMMunicate	-	Identifies communication subsystem commands
:SERial	-	Controls serial port settings. Setting changes are immediate after execution of the command
[:RECeive]	-	Optional node identifier
:BAUD(1)	9600	Sets baud rate to any value between 50 and 115200 baud. Standard rates are 50, 110, 300, 600, 1200, 2400, 4800, 7200, 9600, 14400, 19200, 28800, 38400, 57600, 76800 and 115200 baud. Non-standard rates will be the closest rate that can be created with an integer divider.
:PARity(1)	-	Identifies Parity subsystem
[:TYPE]	NONE	Optional identifier sets 4894A serial parity, values are:

Notes - (1) Setting changes are immediate after command execution.

TABLE 3-3 SCPI COMMANDS AND QUERIES
(CONTINUED)

Keyword	Default Value	Description
		EVEN | ODD | NONE
:CHECk[1]	0	Enables parity checking in accordance with PARity setting. Values are: 1 | 0 for on and off
:BITS[1]	8	Sets number of data bits per character. Values are: 7 | 8
:SBITs[1]	1	Sets number of stop bits. Values are: 1 | 2
:PACE[1]	NONE	Enables software (X-on/X-off) flow control of the serial data. Values are: XON | NONE
:EOMchr[1]	13(CR)	Sets end-of-message character. Numeric value is 0 to 255
:ADD[1]	-	Identifies add character subsystem
:CHARacter	10 (LF)	Sets the character which 4894A will add to the end of the received message. Values are: 0 - 255 for an ASCII character
:ENABle[1]	0	Enables character addition after EOM character is sensed. Values are: 1 | 0 for on and off
:EOI	1	Enables assertion of EOI when last character of the message (EOM or ADD char) is output on the GPIB bus. Values are 1/0 for

Notes - [1] Setting changes are immediate after command execution.

TABLE 3-3 SCPI COMMANDS AND QUERIES (CONTINUED)

Keyword	Default Value	Description
		on and off.
:BUFFer?	-	Returns number of bytes in serial buffer range is 0-32768
:RS485	OFF	Enables full or half-duplex RS-485 operation. OFF leaves transmitter enabled between data transmissions. ON causes transmitter to be disabled between data transmissions. Values are: ON | OFF
:GPIB	-	Controls GPIB (IEEE 488) port settings
:ADDRess[1]	04	Sets 4894A's GPIB primary address. In the G Mode, the 4894A will respond to the primary address. An address of 31, sets the 4894A into LISTEN ONLY. In the S Mode, this address is the address of the device that is connected to the 4894A's GPIB port. An address of 31 disables the S Mode addressing functions. Values are: 0 to 30 for GPIB Primary address 31 for G Mode 4894A becomes LISTEN-ONLY device 31 for S Mode disables device addressing capability
:ADDRess ?	-	Returns 0-30 for primary addresses or 32+ primary address if set to 31.
:SWAP	TIME	Sets when the 4894A will address the GPIB device to talk in the S Mode. Values are: TIME | CR | LF | NONE If NONE is selected, the 4894A operates as a talk-only device. If other values are selected, the 4894A uses the GPIB address setting as the address of the GPIB device
:BUFFer?	-	Returns number of bytes in GPIB buffer. Range is 0 to 225,280

Notes - [1] Setting changes are immediate after command execution.
Provide 20 msec. delay for internal reconfiguration.

TABLE 3-3 SCPI COMMANDS AND QUERIES (CONTINUED)

Keyword	Default Value	Description
:ERRor?	0, "No error"	Requests next entry in 4894A's error/ event queue. Error messages are: 0, "no error" -100, "command error" -200, "Execution error" 310, "Serial port input error" -400, "Query error"
:VERSion?	1994..0	4894A returns the <value> of the applicable SCPI version number
STATUS	-	Starts status reporting structure
:OPERation	-	Identifies Operation registers
:QUEStionable	-	Identifies Questionable registers
[:EVENt?]	-	Returns contents of the event register associated with the command
:CONDition?	-	Returns contents of the condition register associated with the command
:ENABle	0	Sets the enable mask which allows the true conditions in the associated event register to be reported in the summary bit
:PRESet	-	Sets all device dependent enable registers to all 1s so that device dependent events will propagate to their summary bits in the Status Byte Register

3.7 PROGRAMMING GUIDELINES

The following section provides information on how to program the 4894A and 4804 to receive and transmit serial data. When programming either unit, the user must follow the sequence outlined below:

Send IFC	'gets control, asserts REN and apply power to the unit.
Send Escape Sequence	'puts the 4894A/4804 into the command mode and enables the SCPI parser.
Wait 30 ms	'delay for above setup
Send SCPI commands and queries	'setup unit or query its status
Send "*SAV 0"	'saves current setup
Send "SYST:OPER DATA"	'returns unit to transparent data mode

Refer to section 2.5 for detailed instructions on programming the 4894A and 4804 from HP, National Instruments and other controllers.

3.7.1 Receiving ASCII Data

The easiest way to receive serial ASCII data is to have the 4894A buffer the received serial characters in the serial buffer and then generate an SRQ when it has received a complete message. The bus controller can then serial poll the 4894A to determine if there is a message and then read the message from the 4894A. This method minimizes the amount of time that the bus controller spends inputting the message. To enable SRQs for a received message, enable the Serial Message Received bit (bit 9) in the Questionable Enable Register and the Questionable summary bit (bit 3) in the Status Byte Register. Use the "**:EOMchr**" command to set the EOM character to detect the last character in the serial message. The following commands set the 4894A to sense a carriage return as the end of a serial message and to generate an SRQ.

STAT:QUES:ENAB 512	Enables serial message received bit.
***SRE 8**	Enables Questionable summary bit.
SYST:COMM:SER:EOM 13	Sets EOM character to carriage return.

The 4894A has the ability to add a character to the end of the serial message. This capability can be used to convert a serial message that ends in a carriage return (such as those from a terminal) into a message that ends with carriage return and line feed for easier entry into the bus controller. (IEEE-488.2 controllers may not sense a carriage return as the end-of-message character) The 4894A inserts the ADD character immediately after the EOM character if the ADD character ENABLE is set to a 1. The following commands enable the addition of line feed to any serial message that ends with a carriage return.

SYST:COMM:SER:EOM 13	Sets EOM to carriage return.
:ADD:CHAR 10	Sets Add character to line feed.
:ADD:ENAB 1	Enables Add character addition.

The 4894A can also assert EOI when it outputs the last character of the received serial message to the GPIB bus. The last character is either the EOM character or the ADD character. To enable EOI assertion send the command:.

SYST:COMM:SER:EOI 1	Enables EOI generation.

3.7.2 Receiving Binary Data

To receive serial binary data, SRQs from EOM character detection must be disabled since the EOM character could be a data character. EOI generation will also have to be inhibited. SRQs could be

generated by presence of the Serial Buffer not Empty bit, bit 9, in the Operation Condition Register. The programmer will have to do some form of byte counting or sense a character pattern to recognize when the complete binary message has been received. The commands are:

STAT:QUES:ENAB 0	Disables SRQs when EOM character detected.
SYST:COMM:SER:EOI 0	Disables EOI generation.
STAT:OPER:ENAB 512	Enables SRQs when serial receive buffer is not empty.
***SRE 128**	Enables Operation Summary bit (bit 7) in status byte Register.

An alternate method to byte counting is to let the serial data accumulate in the Serial Buffer and periodically query the buffer to determine the received byte count and transfer all the data at one time to the bus controller. The query command is:

SYST:COMM:SER:BUFF?	Queries Serial Buffer

3.7.3 Transmitting Large Data Blocks

When the amount of data being sent over the GPIB bus to the 4894A exceeds the 4894A's GPIB buffer capacity (225,000 bytes) the user can only output part of the data file at a time. The easiest way to do this is to enable the GPIB Buffer Empty bit (bit 9) in the Operation Enable Register and send the 4894A the first portion of the data. The 4894A will then generate an SRQ when it has output the data and the GPIB buffer is empty. Each time the bus controller receives an SRQ, it should output another block of the data file to the 4894A until the complete data file has been transmitted to the serial device. Use the following commands to enable SRQs when the GPIB Buffer is empty.

STAT:OPER:ENAB 256	Enables GPIB buffer empty bit 8.
***SRE 128**	Enables Operation Summary bit 7 in Status Byte Register.

Alternately, the programmer could query the GPIB Buffer to determine its status and capacity to accept more data. The query command is:

SYST:COMM:GPIB:BUFF?

3.7.4 Buffer Status Bits

There are two additional buffer status bits that can be used by the bus controller to send or receive data. The GPIB Buffer Full bit (bit 10) in Operation register will generate an SRQ when the GPIB Buffer is 98% full to warn the user to stop sending data to the 4894A. If the user does not stop, the 4894A will hold the GPIB bus when the GPIB buffer is full to prevent any data loss. The Serial Buffer Full bit (bit 10) in Questionable register can be used instead of the Message Received bit to generate SRQs when receiving large amounts of serial data. The Serial Buffer Full bit comes on whenever the Serial Buffer is 87% full (at approximately 28,000 characters). The SRQ enabling commands for these bits are:

STAT:OPER:ENAB 1024	Enables bit 10 in the Operation Enable Register.
STAT:QUES:ENAB 1024	Enables bit 10 in the Questionable Enable Register.
***SRE 136**	Enables bits 7 & 3 in the Service Request Enable Register.

3.7.5 Parity Generation and Checking

The 4894A can be programmed to generate serial characters with odd, even or no parity. Parity generation is automatic. Parity checking of received serial characters can be enabled separately. When the 4894A

senses a parity error on a received serial character it sets bit 3 in the Event Status Register. If checking is enabled, the 4894A substitutes an asterisk character, '*****', for the faulty character. The asterisk character has the hex code 2A. This substitution makes it easy for the programmer to develop a routine to locate and correct the faulty character. The relevant commands are:

SYST:COMM:SER:PAR type	Sets parity type.
:PAR:CHECK 1/0	Enables/disables parity checking.
***ESE 8**	Enables serial error bit 3 in the ESR.

Note that when using the ***ESE** and ***SRE** commands, the value sent should always be the sum of all the bits that are to be enabled.

When receiving serial messages, the programmer needs only to enable SRQs for the Serial Message Received bit in the Questionable Condition Register and summary bit 3 in the Status Byte Register. When the SRQ occurs, check for the presence of bit 3 and bit 5 in the Status Byte Register. If bit 3 is on, check bit 9 in the Questionable Condition Register for a serial message. Input the message if the bit is set on. If bit 5 is on, query the Event Status Register to determine whether there was a parity error with the serial data. The relevant commands are:

***STB?**	Reads Status Byte Register status.
STATUS:QUES:EVEN?	Reads Questionable Event Register status.
***ESR?**	Reads Event Status Register status.

If checking was enabled and a serial error had occurred, then the 4894A would have substituted an asterisk "*****" for each faulty character. Search the message for the faulty character(s) or ask for a retransmission.

3.8 4894-8 STATUS REPORTING STRUCTURE AND NI-CV EMULATION

The 4894-8 has the simple CV Status Reporting Structure shown in Figure 3-2. Table 2-3 shows how to set switch U22 to enable SRQ generation. If SRQs are enabled, bit 0 and/or bit 1 will generate an SRQ when their condition becomes true.

A standard 4894A/4804 can be programmed to emulate the CV Reporting Structure for compatibility with the National Instrument's GPIB-CV type GPIB to Serial Interface. The major change to the 4894A's operation is that the 488.2 Status Reporting and SRQ generation described in paragraph 3.4 is replaced with the much simpler status structure shown in Figure 3-2. This change lets the 4894A act as a plug in replacement for the older CV units without changing the user's programs.

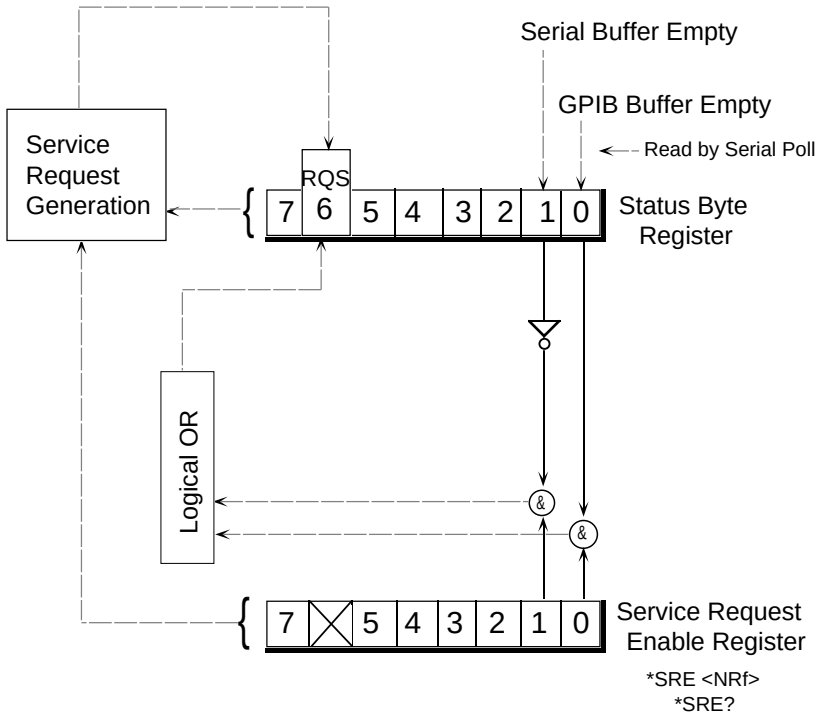


Figure 3-2 CV Emulation Status Reporting Structure

The selection of the NI-CV Emulation and SRQ generation is done during configuration. Use the following command to select the NI functions:

SYST:EMUL NI for NI-CV emulation

Use the "***SRE**" command to enable the desired SRQ enabling bits.

e.g. ***SRE 0** For no SRQ generation.

***SRE 3** For SRQs when GPIB buffer is empty or serial buffer is not empty.

Use the following commands to put the 4894A to the NI-CV's factory settings:

SYST:COMM:SER:BITS 7	Sets 7 data bits.
SYST:COMM:GPIB:ADDR 5	Sets primary address 05.
SYST:COMM:SER:EOI 0	Disable EOI assertion.

The remaining ICS factory settings remain the same:

Baud	9600
Parity	None
Check	0
Stop Bits	1
Mode	G
Swap	Time-out
Pace	None

The command ***SAV 0** saves the current setting. If NI emulation is active at power turn-on time, the saved SRE value will be restored and the 4894A will be ready for data transfer with no further settings required.

Model 4894-8s are configured by setting their internal switches as described in Section 2-9

4

Operating From The Serial Interface

4.1 INTRODUCTION

This section describes how the 4894A, 4894-8 and the 4804 operates and transfers data when controlled from the serial interface. Wherever the text refers to the Model 4894A, it applies equally to the Model 4804 unless otherwise noted.

4.2 S MODE OPERATION

When the 4894A is controlled from the serial interface it is operating in the S Mode. Mode selection is made when the unit is configured (For 4894As and 4804s refer to Section 2.6; for 4894-8s refer to Section
2-9)

In the S mode, the 4894A acts as a GPIB system controller for a single GPIB device. At power turn on, the 4894A asserts REN and pulses IFC on for a minimum of 100 microseconds. The 4894A then outputs a GPIB listen command to the GPIB device. The address of the GPIB device is the same as the GPIB address assigned to the 4894A when it was configured. If the 4894A's GPIB address was set to 31, it outputs all possible listen addresses and becomes a talk-only controller.

4.3 INCOMING SERIAL DATA

Incoming serial data is stored in the 4894A's serial-to-GPIB buffer and then transferred out to the GPIB device. This is the primary S Mode data transfer direction. The 4894A always transfers data in that direction if the SWAP parameter was set to 'none' or the GPIB address was set to 31 (talk only mode) when the 4894A was configured.

4.4 TRANSFERRING GPIB DATA TO THE SERIAL INTERFACE

There are three ways that the 4894A can be set to swap directions and transfer data from the GPIB device to the serial interface. They are:

1. Time out
2. Carriage Return Sensed
3. Line Feed Sensed

When time-out is enabled, the 4894A will swap directions and address the GPIB device to talk when its Serial-to-GPIB buffer is empty and no serial characters have been received for a minimum of 300 milliseconds. At this point the 4894A is ready to transfer data from the GPIB to the serial interface.

When either Carriage Return or Line Feed Termination is enabled, then the 4894A checks the Serial-to-GPIB output data for presence of either terminating character. If the selected terminator is found, the 4894A finishes handshaking the character on the GPIB bus and then addresses the GPIB device to talk. At this point the 4894A is ready to transfer data from the GPIB to the serial interface.

Any new serial character received by the 4894A will cause the 4894A to readdress itself as a talker and the GPIB device as a listener.

Theory of Operation

5.1 INTRODUCTION

This section describes the theory of operation of the 4894A/4804. Wherever the text refers to the Model 4894A, it applies equally to the Model 4804 unless otherwise noted.

5.2 BLOCK DIAGRAM DESCRIPTION

A block diagram of the 4894A is shown in Figure 5-1. The 4894A is a microprocessor based device that transparently passes data between the GPIB (IEEE 488) bus and a serial device. The 4894A is made up of seven major elements, most of which are interconnected to the microprocessor by a common data, address and control signal bus.

Incoming GPIB bus data and commands are received by the GPIB controller chip. Each received character interrupts the microprocessor to pass on the command characters. Data characters are automatically transferred to a memory buffer by a high speed DMA controller in the microprocessor. From the memory buffer, the data characters are placed in the microprocessor's UART where they are serialized, passed through the selected driver and outputted at the serial interface. Command characters are parsed and used to charge the 4894A's operational settings or invoke some action or response.

Incoming serial data is sensed and shifted into TTL levels and applied to the UART's input. Each received character is temporarily stored in

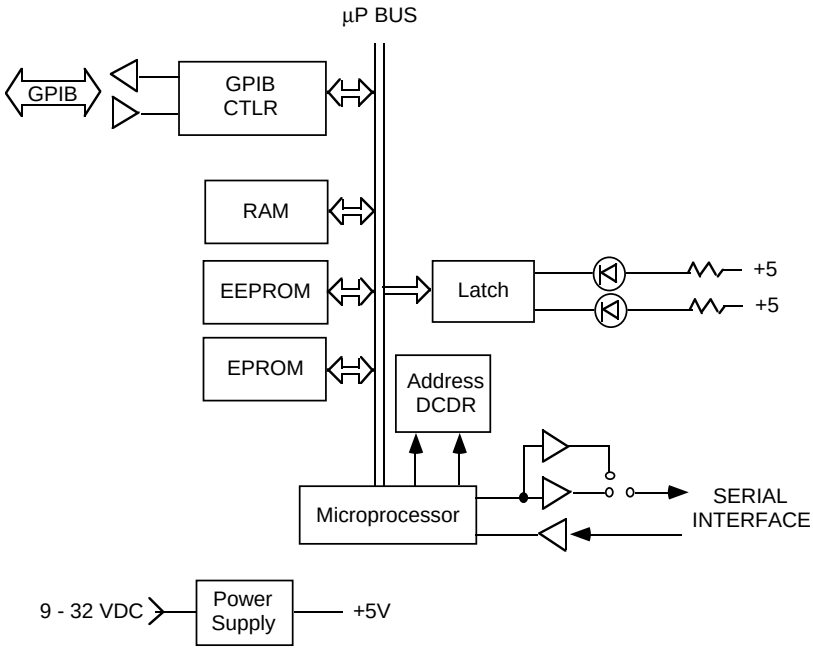


Figure 5-1 4894A Block Diagram

a received data buffer in memory until it can be transferred out onto the GPIB bus. A multilevel Status Byte Register and Event Register structure enables the 4894A to interrupt the GPIB bus controller on receipt of a character, on receipt of a complete serial message or when the receive data buffer is full.

The EPROM contains all of the 4894A's program instructions, command tables, and power turn-on/self test routines. At power turn-on, the 4894A performs a self test on each functional block to determine whether there is a gross system failure. Any self test error is displayed as a pattern of blinking LEDs on the front panel. The error pattern is repeated until the unit is turned off. Just after completing the self test routine, the 4894A displays its current GPIB address setting on the front panel LEDs. Bit weights are read from right to left with the least significant bit on the far right. In the 4894A, the RDY LED comes on to indicate a successful completion of the self test routine.

The E²PROM contains all of the 4894A's and 4804's configuration settings, serial number and other parameters that are subject to change. At power on time, the microprocessor copies the E²PROM configuration to RAM where it is used to operate the unit. Any changes made to the settings during run time are not stored in the E²PROM until the user sends the ***SAV** command.

For Model 4894-8s, the configuration values are saved in the internal switch settings. At power on time, the microprocessor copies the switch settings to RAM where they are used to operate the unit.

In the 4894A, the RAM is a 256K x 8 bit memory that is primarily used for data storage. The remaining area is used to hold the operating variables and configuration settings. The data buffers are mechanized as circular buffers with flags to prevent accidental overwrite of data when they are full. Serial control signals and X-on/X-off handshaking (if enabled) are both used to control the serial data flow and prevent loss of data. Any serial characters sent to the 4894A after it has issued a stop request (X-off) are saved until the buffer becomes full and then discarded. GPIB bus data is never lost since the 4894A simply inhibits further Bus handshakes until there is room in the GPIB buffer for more data.

In the 4804, the RAM is an 8K x 8 bit memory with reduced buffer sizes designed to handle the shorter serial messages required by an OEM interface.

The 4894A's power supply is a switching regulator that converts any unregulated DC input to +5 Vdc to run the 4894A's internal logic chips. Satisfactory input range is +9 to +32 Vdc. Another DC-DC converter makes ±10 Vdc to operate the RS-232 drivers.

The Model 4804 uses +5 to +5.3 Vdc power supplied from the host power supply. A diode in series with the power input pads reduces the input voltage by 0.3 volts. The voltage measured across the test points should be +5 ± 0.2 Vdc for proper operation.

6

Maintenance, Troubleshooting and Repair

6.1 INTRODUCTION

This section describes the maintenance testing, troubleshooting, and repair procedures for ICS's Model 4894A and 4804 GPIB↔ Serial Interfaces. Wherever the text refers to the Model 4894A, it applies equally to the Model 4804 unless otherwise noted.

6.2 MAINTENANCE

The units do not require periodic calibration and have no internal adjustments. However, if the 4894A is used in an application where the IEEE 488 bus cables are frequently changed, the 4894A's IEEE 488 Bus Connector may occasionally require cleaning to remove wax and dirt buildup. New bus connectors are shipped with a brightener (thin wax like film) on them. Depending upon usage, enough of the brightener may buildup on the 4894A's bus connector to cause intermittent operation.

The brightener is an organic compound and may be cleaned off by washing the connector with a mild detergent solution followed by an alcohol wash.

6.3 TROUBLESHOOTING

Troubleshooting consists of isolating the fault and then repairing it or returning the unit for repair. If the unit blinks its LEDs at power turn, refer to Table 5-1 for the 4894A and 4804's selftest error codes and the most probable faulty component. If the fault occurs after a successful selftest, refer to Table 5-2. Table 5-2 lists several common problems and suggests actions that can be done to either clear the fault or repair the situation.

WARNING

If the fault isolation procedure requires internal measurements, always remove power when disassembling or assembling the unit. Use extreme caution during troubleshooting, adjustments, or repair to prevent shorting components and causing further damage to the unit.

6.4 SELF TEST ERROR CODES

At power turn on, the 4894A conducts a self test of its major components. The test takes about 5 seconds and a successful test ends with the PWR and RDY LEDs both on. Test failures are indicated by the LED patterns shown in Table 5-1. Typically the failure is a pattern of blinking LEDs which lasts until the unit is turned off. In older units, if the 4894A did not have a fatal error, the self test would continue until it was complete but the RDY LED would remain off. If a self test failure occurs, turn the unit off for 10 seconds and turn power back on. If the failure persists, refer to paragraph 6.5 for repair instructions.

TABLE 5-1 4894A SELF TEST ERROR CODES

TST? Resp.	Front Panel LED PWR	RDY	TALK	LSTN	BUSY	FULL	Fault
–	⊕	–	–	–	–	–	fatal error (CPU, EPROM, RAM …etc.)
–	–	–	–	–	–	–	fatal error (power supply)
–	⊕	⊕	⊕	⊕	⊕	⊕	fatal error (CPU, EPROM, RAM …etc.)
1	⊕	B	–	–	–	–	EPROM U12, data or address bus
2	⊕	–	B	–	–	–	RAM-U25
4	⊕	B	B	–	–	–	RAM-U24
8	⊕	–	–	B	–	–	7210-U15
16	⊕	B	–	B	–	–	EEPROM U11
0	⊕	⊕	×	×	–(1)	–	self test passed successfully

led legend: ⊕ led is on B led is blinking
– led is off × led could be on or off

(1) if the serial input is not connected the BUSY LED will show high speed blinking.

TABLE 5-2 TROUBLESHOOTING GUIDE

Symptom	Possible Fault	Action or Check
Unit will not turn on	Power cord not plugged in	Push power cord into DC receptacle
	Power at AC outlet	Check outlet and power adapter
Unit shows blinking LEDs	Self test fault	Check Self Test Error Table
Unit will not switch into Command Mode	Wrong escape sequence	Verify correct escape sequence for current unit address. See Table A-1 in Appendix A1.
	Sequence sent too fast	PCs with clock > 200 MHz need a pause between characters. Add delay or use separate commands for each character.
Not sure unit is in Command Mode	Test with *IDN?	See paragraph 6.5
Unit does not respond to National Instruments TNT GPIB card	Risetime reflections breaking down data transceiver	Check data transceiver U9 type. Replace a TI 75160AN with a TI 75160BN. See paragraph 6.6
Unit configuration messed up or unconfigurable	May need to be reset to factory settings	Reset per paragraph 6.7

6.5 COMMAND MODE -LOOPBACK TEST

The following test can be used to verity that the unit is in the Command Mode. Remove any cable connected to the serial connector and insert a short jumper wire between pins 2 and 3. Send the unit the *IDN? query. Address the unit to talk and read back the serial message. If the unit is in the command mode, the response will be the unit's IDN message as described in Table 3-1. If the unit responds with "*IDN?", the unit is in the data mode. It treated the *IDN? query as a data, message, transmitted the query as a serial message and then received it back.

6.6 OPERATION WITH NATIONAL'S TNT CARD

National Instruments developed a high speed GPIB transmitter chip and tried to get the high-speed GPIB protocol approved as a standard. Hewlett-Packard and other companies voted against it as being a potential source of problems. Tests at ICS Electronics and at other companies have shown that the TNT chip's fast risetimes cause reflections that breakdown some GPIB data transceiver chips. This problem is most pronounced when the affected device is operated at the end of the GPIB cable. So far, this problem has only shown up on TI SN75160A series chips.

Possible solutions are:

1. Replace all TI 75160AN ICs with TI 75160BNs.
2. Shorten the cable length.
3. Rearrange the devices on the GPIB bus so that the affected units are not at the end of the cable.
4. Return the TNT card to National Instruments and replace it with one from ICS Electronics or Computer Boards.

6.7 RESETTING THE FACTORY DEFAULTS

To reset the 4894A to the factory default setting, turn power off. Jumper pin 7 to pin 25 on the serial I/O connector. Turn power on and let the unit perform a complete selftest. Turn power off and remove the jumper. The unit has now been returned to its factory setting.

6.8 RETURNING FOR REPAIR

Repair of the 4894A is best done by returning the unit to the factory or to your local distributor. Units in warranty should **always** be returned to the factory or distributor.

When returning a unit, a board assembly, or other products to ICS for repair, it is necessary to go through the following steps:

1. Contact the ICS customer service department and ask for a return material authorization (RMA) number. An ICS application engineer will want to discuss the problem at this time to verify that the unit needs to be returned, or to assist in correcting the problem. Many of the difficulties customers call about can be resolved over the phone as opposed to returning a unit for repair.

2. Write a description of the problem and attach it to the material being returned. Describe the installation, system failure symptoms, and how it was being used. If the item being returned is a board assembly, describe how you isolated the fault to it. Include your name and phone number so we can call you if we have any questions. Remember, we need to locate the problem in order to fix it.

3. Pack the item with the fault description in a box large enough to accommodate a minimum of two inches of packing material on all four sides, the top, and the bottom of the box. Securely seal the box.

4. Mark the shipping label to the attention of RMA#. The RMA number is very important since it is our way of identifying your unit in order to return it to you.

5. Ship the box to ICS freight prepaid. ICS does not pay freight to return the unit to ICS, but will prepay the freight to return the repaired item to you.

Appendix

A

A1 IEEE 488 BUS DESCRIPTION (IEEE 488.1, IEEE 488.2, SCPI)

A1.1 IEEE 488.1

The IEEE Std 488 Bus, or GPIB as it is commonly referred to, provides a means of transferring data and commands between devices. The physical portion of the bus is governed by IEEE -Std 488.1 - 1978. The interface functions for each device are contained within that device itself, so only passive cabling is needed to interconnect the devices. The cables connect all instruments, controllers and other components of the system in parallel to the signal lines as shown in Figure A-1. Eight of the lines (DIO1-DIO8) are reserved for the transfer of data and other messages in a byte-serial, bit-parallel manner. Data and message transfer is asynchronous, coordinated by the three handshake lines (DAV, NRFD, NDAC). The other five lines control Bus activity.

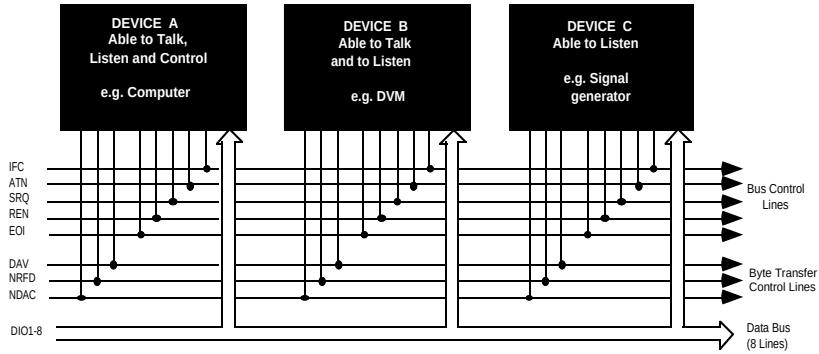


Figure A-1 IEEE 488 Bus

A1

Two types of messages are transferred over the bus:

Interface messages - for bus management

Device-dependent messages - for device control and data transfer

Devices connected to the bus may act as talkers, listeners, controllers, or combinations of the three functions, depending upon their internal capability. The system controller is a controller that becomes active at power turn-on. It is the Bus manager and the initial controller-in-charge.

A **controller** can send interface messages to manage the other devices, address devices to talk or listen and command specific actions within devices.

A **talker** sends device dependent messages, i.e., data, status.

A **listener** accepts interface messages, bus commands and device-dependent messages, i.e., setup commands, data.

Bus systems can be as simple as two devices; one a talker always sending data to a second device which listens to the data. Larger systems can have one or more controllers and many devices (the IEEE 488 driver specifications limit the total number of units on one bus system to 15). Only one controller can be the controller-in-charge at any given time. Control originates with the system controller and is passed back to other controller(s) as required. Control can be passes back to the system controller or to another controller after the completion of the task. The system controller has the capability of taking control back at any time and resetting all addressed devices to their unaddressed state.

Each bus device is identified by a five-bit binary address. There are 31 possible primary addresses 0 through 30. Address 31 is reserved as the 'untalk' or 'unlisten' command. Some devices contain subfunctions, or the devices themselves may be addressed by a secondary five-bit binary address immediately following the primary address, i.e. 1703. This secondary address capability expands the bus address range to 961 addresses. Most bus addresses are set at the time the system is configured by rocker switches which are typically located on each devices' rear panel. Devices that are SCPI 1991 compatible, can have their bus address set by a GPIB SYSTEM configuration command.

Information is transmitted on the data lines under sequential control of the three handshake lines. No step in the sequence can be initiated until the previous step is completed. Information transfer proceeds as fast as the devices respond (up to 1 Mbs), but no faster than that allowed by the slowest addressed device. This permits several devices to receive the same message byte at the same time. Although several devices can be addressed to listen simultaneously, only one device at a time can be addresses as a talker. When a talk address is put on the data lines, all other talkers are normally unaddressed.

ATN (attention) is one of the five control lines and is set true by the controller-in-charge while it is sending interface messages or device addresses. The messages are transmitted on the seven least significant data lines and are listed in the MSG columns in Table A-1. When a device is addressed as a talker, it is allowed to send device-dependent messages (e.g., data) when the controller-in-charge sets the ATN line false. The data messages are typically a series of ASCII characters ending in a CR, LF, or CR LF sequence. The data messages often consist of eight-bit binary characters and end on a predetermined count or when the talker asserts the EOI line simultaneously with the last data byte. The controller-in-charge must be programmed to correctly respond to each device's message termination sequence to avoid hanging-up the system or leaving characters that will be output when the device is addressed as a talker again.

IFC (interface clear) is sent by the system controller and places the interface system in a known quiescent state with all devices unaddressed.

REN (remote enable) is sent by the system controller and is used with other interface messages or device addresses to select either local or remote control of each device.

SRQ (service request) is sent by any device on the bus that wants service, such as counter that has just completed a time-interval measurement.

ASCII -- IEEE 488 BUS MESSAGES (COMMANDS AND ADDRESS) HEX CODES

MSD / LSD	0		1		2		3		4		5		6		7	
	ASCII	MSG	ASCII	MSG	ASCII	MSG1	ASCII	MSG1	ASCII	MSG1	ASCII	MSG1	ASCII	MSG	ASCII	MSG
0	NUL		DLE		SP	00	0	16	@	00	P	16	`	▲	p	▲
1	SOH	GTL	DC1	LLO	!	01	1	17	A	01	Q	17	a		q	
2	STX		DC2		"	02	2	18	B	02	R	18	b		r	
3	ETX		DC3		#	03	3	19	C	03	S	19	c		s	
4	EOT	SDC	DC4	DCL	$	04	4	20	D	04	T	20	d		t	
5	ENQ	PPC	NAK	PPU	%	05	5	21	E	05	U	21	e		u	
6	ACK		SYN		&	06	6	22	F	06	V	22	f		v	
7	BEL		ETB		'	07	7	23	G	07	W	23	g		w	
8	BS	GET	CAN	SPE	(	08	8	24	H	08	X	24	h		x	
9	HT	TCT	EM	SPD	)	09	9	25	I	09	Y	25	i		y	
A	LF		SUB		*	10	:	26	J	10	Z	26	j		z	
B	VT		ESC		+	11	;	27	K	11	[	27	k		(	
C	FF		FS		,	12	<	28	L	12	\	28	l		|	
D	CR		GS		-	13	=	29	M	13	]	29	m		)	
E	SO		RS		.	14	>	30	N	14	^	30	n		~	
F	SI		US		/	15	?	UNL	O	15	_	UNT	o	▼	DEL	▼

MEANING DEFINED BY PCG CODE

MEANING DEFINED BY PCG CODE

ADDRESSED COMMAND GROUP

UNIVERSAL COMMAND GROUP

LISTEN ADDRESS GROUP

TALK ADDRESS GROUP

SECONDARY COMMAND GROUP

PRIMARY COMMAND GROUP (PCG)

Notes:
1. Device Address messages shown in decimal
2. Message codes are:
   - DCL -- Devices Clear
   - GET -- Device Trigger
   - GTL -- Go to Local
   - LLO -- Local Lockout
   - PPC -- Parallel Poll Configure
   - PPU -- Parallel Poll Unconfigure
   - SDC -- Selected Device Clear
   - SPD -- Serial Poll Disable
   - SPE -- Serial Poll Enable
3. ATN off, Bus data is ASCII; ATN on, Bus data is an IEEE MSG.

EOI (end or identify) is used by a device to indicate the end of a multiple-byte transfer sequence. When a controller-in-charge sets both the ATN and EOI lines true, each device configured to respond to a parallel poll indicates its current status on the DIO line assigned to it.

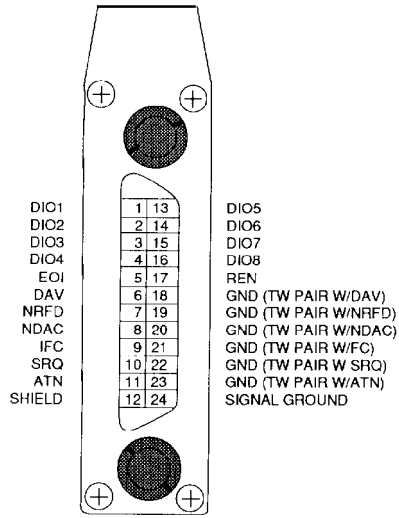


Figure A-2 GPIB Connector Signal Pinouts

A1.2 IEEE 488 MESSAGE FORMATS (IEEE 488.2)

IEEE Std 488.2 was established in 1987 to standardize message protocols, status reporting and define a set of common commands for use on the IEEE 488 bus. IEEE 488.2 devices are supposed to receive messages in a more flexible manner than they send. A message sent from GPIB controller to GPIB device is called: PROGRAM MESSAGE. A message sent from device to controller is called: RESPONSE MESSAGE. As part of the protocol standardization the following rules were generated:

(;) Semicolons are used to separate messages.
(:) Colons are used to separate command words.
(,) Commas are used to separate data fields.
<nl> Line feed and/or EOI on last character terminates a 'program message'. Line feed (ASCII 10) **end** EOI terminates RESPONSE MESSAGE.
(*) Asterisk defines a 488.2 common command.
(?) Ends a query where a reply is expected.

With 488.2, status reporting was enhanced from the simple serial poll response byte in 488.1 to the multiple register concept shown in Figure A-3. Each 488.2 device must implement a Status Byte Register, a Standard Event Status Register and an Output Message Queue. Both registers must have enabling registers that can control the generation of their reporting bits and ultimately SRQ generation. This standardized the bit assignments in the Status Byte Register, added eight more bits of information in the Event Status Register and introduced the concept of summary bits reporting to the Status Byte Register. A 488.2 device outputs the Status Byte Register contents plus the RQS bit in response to a serial poll. A device may include any number of condition registers, event registers and enabling registers providing they follow the model shown in Figure A-3.

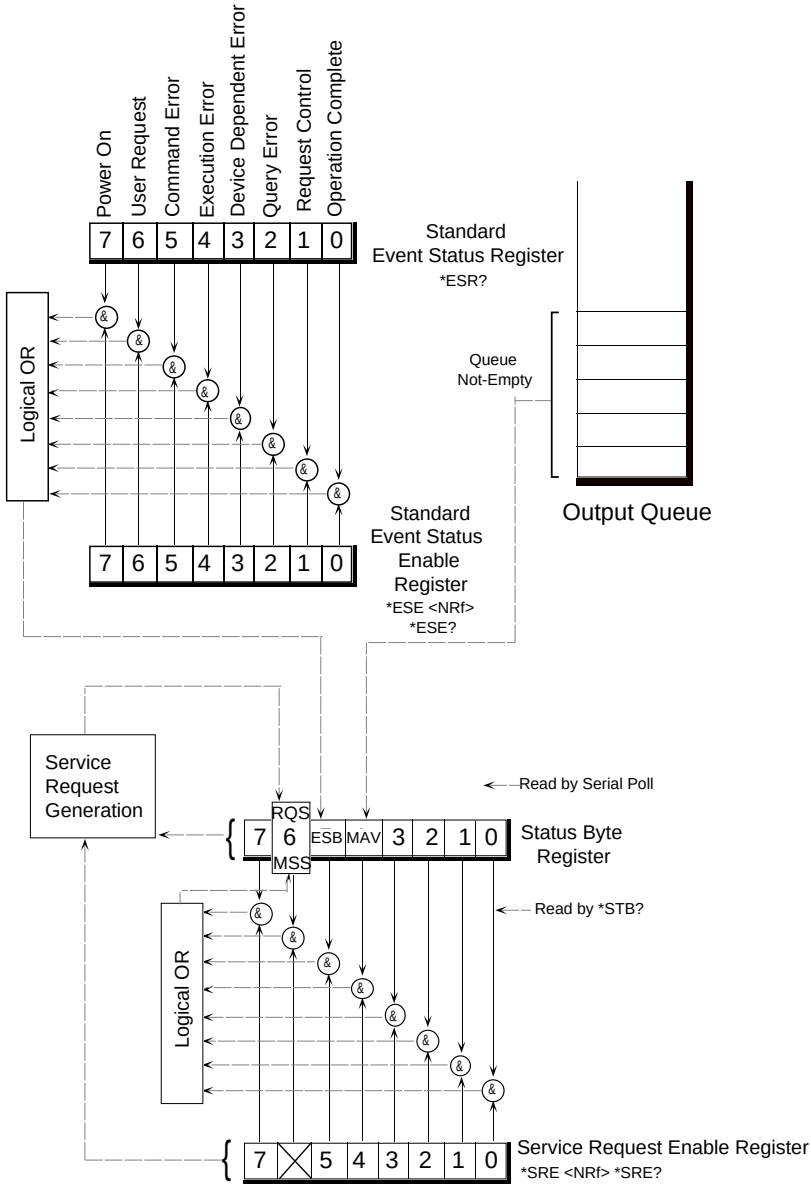


Figure A-3 488.2 Required Status Reporting Capabilities

The 488.2 specification also mandated a list of common commands that all devices will support. These commands are:

1	***CLS**	Clear Status Command
2	***ESE**	Standard Event Status Enable Command
3	***ESE?**	Standard Event Status Enable Query
4	***ESR?**	Standard Event Status Register Query
5	***IDN?**	Identification Query
6	***OPC**	Operation Complete Command
7	***OPC?**	Operation Complete Query
8	***RST**	Reset Command
9	***SRE**	Service Request Enable Command
10	***SRE?**	Service Request Enable Query
11	***STB?**	Status Byte Query
12	***TST?**	Self-Test Query
13	***WAI**	Wait-to-Continue Command

In addition to the above common commands, devices that support parallel polls must support the following three commands

***IST?**	Individual Status Query?
***PRE**	Parallel Poll Register Enable Command
***PRE?**	Parallel Poll Register Enable Query

Devices that support Device Trigger must support the following commands:

***TRG**	Trigger Command

Controllers must support the following command:

***PCB**	Pass Control Back Command

Devices that save and restore settings support the following commands:

***RCL**	Recall configuration
***SAV**	Save configuration

A1.3 SCPI COMMANDS

INTRODUCTION

SCPI (Standard Commands for Programmable Instruments) builds on the programming syntax of 488.2 to give the programmer the capability handling a wide variety of instrument functions in a common manner. This gives all instruments a common "look and feel".

SCPI commands use common command words defined in the SCPI specification. Control of any instrument capability that is described in SCPI shall be implemented exactly as specified. Guidelines are included for adding new defined commands in the future as new instruments are introduced without causing programming problems.

SCPI is designed to be laid on top of the hardware - independent portion of the IEEE 488.2 and operates with any language or graphic instrument program generators. The obvious benefits of SCPI for the ATE programmer is in reducing the learning time on how to program multiple SCPI instruments since they all use a common command language and syntax.

A second benefit of SCPI is that its English like structure and words are self documenting, eliminating the needs for comments explaining cryptic instrument commands. A third benefit is the reduction in programming effort to replace one manufacturer's instrument with one from another manufacturer, where both instruments have the same capabilities.

This consistent programming environment is achieved by the use of defined program messages, instrument responses and data formats for all SCPI devices, regardless of the manufacturer.

COMMANDS

SCPI commands are based on a hierarchical structure that eliminates the need for most multi-word mnemonics. Each key word in the command steps the device parser out along the decision branch - similar to a squirrel hopping from the tree trunk out on the branches to the leaves. Subsequent keywords are considered to be at the same branch level until a new complete command is sent to the device. SCPI commands may be abbreviated as shown by the capital letters in Figure A-4 or the whole key word may be used when entering a command. Figure A-4 shows some single SCPI commands for setting up and queuing a serial interface. Refer to Table 3-3 for a complete description of the SCPI commands used by this unit.

SYSTem:COMMunicate:SERial:BAUD 9600 <nl>
Sets the baud rate to 9600 baud

SYST:COMM:SER:BAUD? <nl>
queue the current baud setting

SYST:COMM:SER:BITS 8 <nl>
sets character format to 8 data bits

Figure A-4 SCPI Command Examples

Multiple SCPI commands may be concatenated together as a compound command using semi colons as command separators. The first command is always referenced to the root node. Subsequent commands are referenced to the same tree level as the previous command. Starting the subsequent command with a colon puts it back at the root node. Common commands and queries (start with *) can be freely mixed with SCPI messages in the same program message without affecting the above rules. Figure A-5 shows some compound command examples.

SYST:COMM:SER:BAUD 9600; BAUD? <nl>

SYST:COMM:SER:BAUD 9600; :SYST:COMM:SER: BITS 8 <nl>

SYST:COMM:SER:BAUD 9600; BAUD?; *ESR?; BIT 6; BIT?; PACE XON; PACE?; *ESR? <nl>

Figure A-5 Compound Command Examples

The 4894 response will be: **9600; 0; 8; XON; 16 <nl>**

The response includes five items because the command contains 5 queries. The first item is **9600** which is the baud rate, the second item is **ESR=0** which means no errors (so far). The third item is **8** (bit/word) which is the current setting. The BIT 6 command was not accepted because only 7 or 8 are valid for this command. The fourth item **XON** means that XON is active. The last item is **16** (ESR) which means execution error - caused by the BIT 6 command.

ERROR REPORTING

SCPI provides a means of reporting errors by responses to the **SYST:ERR?** query. If the SCPI error queue is empty, the unit responds with 0, "No error" message. The error queue is cleared at power turn-on, by a ***CLS** command or by reading all current error messages. The error messages and numbers are defined by the SCPI specification and are the same for all SCPI devices. Table 3-3 lists the SCPI errors reported by this unit.

A2 SERIAL DATA COMMUNICATIONS BACKGROUND

A2.1 INTRODUCTION

Serial data communication is the most common means of transmitting data from one point to another. In serial communication systems, the data word or character is sent bit by bit over some kind of transmission path. The receiving device recognizes each bit as they are received and reassembles them back into the original data word. Serial data communication systems are characterized by four primary factors:

1. Data speed or baud rate

2. Data format

3. Transmission medium

4. Clocking method

Serial data speed is referred to as Baud Rate. A baud is defined as a signaling bit, which includes data bits as well as start/stop framing, parity or any other bits that make up the data format. Typical computer baud rates and their uses are:

110 - for teletypes

300, 1200 - for low speed devices and telephone modems

4800, 9600, 19200 - for high speed devices and modems

Data format refers to the method or pattern the transmitter uses to send the data word or character as a series of bits so that the receiver will know how to recognize the pattern and reassemble the bits back into the original data word. The most common method and the one used in the 4894, is called asynchronous transmission because each character is sent one at a time with an undetermined amount of time between

characters. Each asynchronous character has a low going start bit, a number of data bits, an optional parity bit and 1 or 2 high stop bits. The transmitter automatically extends the stop bit when it has no more characters to transmit. The receiver uses the start bit to resynchronize its clock with the data at the start of each character as shown in Figure A-6.

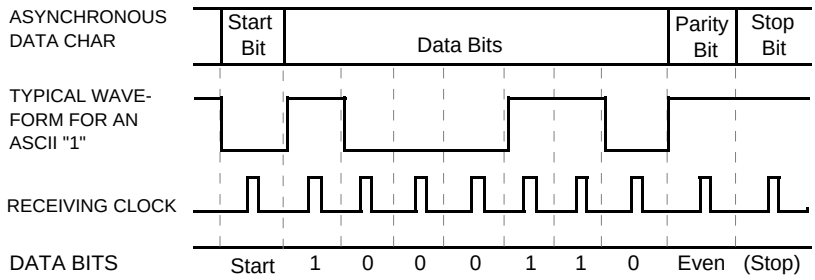


Figure A-6 Asynchronous Data Character Waveforms

Synchronous character do not have start/stop bits and are sent without spaces between characters. Voids between data characters are filled by predetermined sync characters which are discarded by the receiver.

The data portion of the serial character usually contains 5 to 8 bits and is transmitted least significant bit first. Today most of the computers and terminals use the 7 bit ASCII code to represent numbers and characters. Figure A-6 shows how the ASCII "1" is transmitted. Compare the binary code in Figure A-5 against the hex code for an ASCII '1' (HEX 31) and they will be the same. Binary data is usually sent in binary form as 8 bit characters or in hex form as the ASCII characters, 0 through 9 and A through F. Each Hex character represents 4 binary bits so two Hex characters are needed for each 8 bit binary byte.

A2

Parity bits are added after the data field if the user wants to detect transmission errors. When parity bits are used, the transmitter counts the number of high bits in the data field and makes the parity bit a 1 or 0 so the final count will be either even or odd. The receiver then validates the received characters by counting 1's in the data and parity bit fields. The 4894 detects parity errors along with data overrun and

framing errors, generates a Bus SRQ message for each data error and indicates the error by setting the bit 3 on the Standard Event Status Register.

Although serial data can be transmitted over any medium, most of today's computer systems use metallic cable. To ensure compatibility, the manufacturers have adopted interface standards so that they are electrically compatible. The more popular standards are:

RS-232	Most popular standard for office machines and computer systems.
RS-422 and RS-485	New high speed standard with noise improvements over RS-232 for longer distances.

Devices employing the same interface standard can usually be connected together but the user **must** verify each devices signal requirements before plugging them together.

When data must be transmitted over long distances, it is typically sent over the phone company's direct dial network (DDN) as shown in Figure A-7. Modems are used to convert the serial data bits into tones that will pass through the telephone system's 300 to 3000 Hz voice band. For low baud rates, up to 1200 Hz, the modems convert the bits into two tones (frequency switched keying) that the receiving modem recognizes and converts back to data bits. These low speed modems are referred to by the telephone company's designations, i.e.: Type 103 (300 baud) and Type 212 (1200 baud). Higher data rates require more complex modulation techniques and the modems are referred to by their CCITT specification i.e., V22.

With asynchronous characters, the receiver normally uses the start bit to synchronize its internal clock. However, some devices, such as the higher speed modems, require the data bits to be synchronized with their clock. These units are referred to as synchronous modems (not the same as synchronous data characters) and they will supply the clock signals to both the transmitting and receiving device.

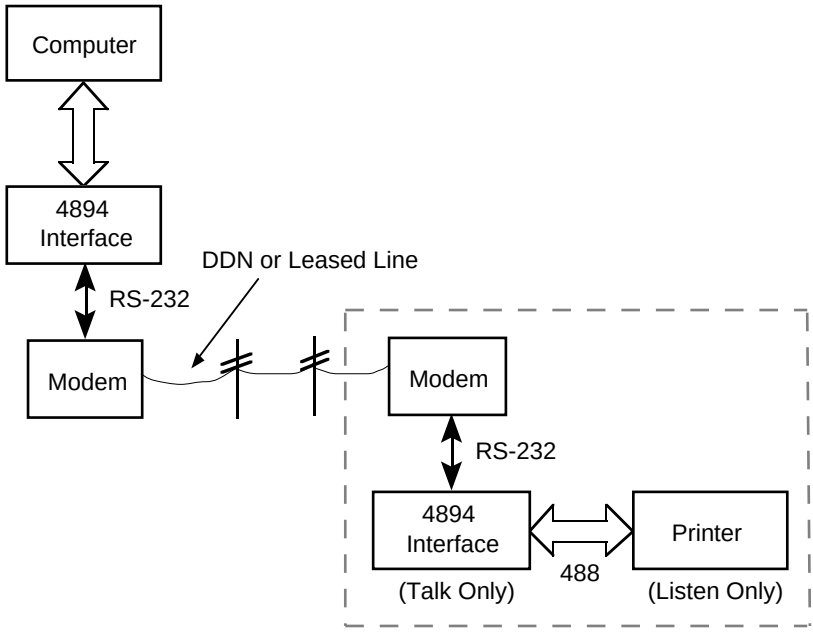


Figure A-7 Long Distance Transmission Using 4894's

Another aspect of timing is the control of data transmission to avoid data overrun. The two methods used are control signals and X-on/X-off characters.

For the control signal method, extra wires are provided in the cable for handshake signals that enable or inhibit data flow. The more common control signal pairs are:

Request-to-send / Clear-to-send

Data-terminal-ready / Data-set-ready

A2

All signals must be high to enable data transmission. Dropping any line normally means the receiving device's buffer is full or it is busy with the last message.

Another method of controlling the data flow is to imbed X-on/X-off characters in the data message. At turn on, both devices are initially in the X-on state. When one device becomes full, it sends the other an X-off character to inhibit future data transmission. X-on is then sent to restart the data transmission when there is room in the receiving device's buffer for additional data.

The 4894 Serial Interface normally uses asynchronous 8 bit data characters with no parity and single start and stop bits. The 4894 will also work with 7 bit data characters. The unused data bits are outputted on the 488 Bus as fixed zeros. The user can also add a parity bit and the second stop bit if required for his system.

A2.2 RS-232 STANDARD

In 1963, the Electronic Industry Association (EIA) established a standard to govern the interface between data terminal equipment and data communication equipment employing serial binary interchange. The latest revision of this standard (RS-232) has been in effect since 1969 and is known as RS-232C. It specifies:

- Mechanical characteristics of the interface
- Electrical characteristics of the interface
- A number of interchange circuits with descriptions of their functions
- The relationship of interchange circuits to standard interface types

The specification does not mean that two devices that are RS-232 compatible can be connected together with a standard cable and be expected to work.

Mechanically, RS-232 interfaces use a 25 pin male connector (DS-25P) with the data terminals and a 25 pin female connector (DS-25S) with the data communications units (modems).

Electrically, RS-232 signals are bi-polar and are referenced to a common ground (AB) on pin 7. Transmitted signals must be between $^{+}5$ and $^{+}15$V or -5 and -15V into 3000 to 7000 ohm loads. Maximum open circuit transmitter outputs is ±25V. Logic levels are:

	+5 to +15V	-5 to -15V
Data	0	1
Control	1 (On)	0 (Off)

Functionally, the specification established two types of devices, DCE and DTE, that would mate together by a pin-to-pin cable. The Data Communication Equipment (DCE) was designated as the device that connected to the communication line. An example of a DCE is a modem. The Data Terminal Equipment (DTE) was designated as the device that connected to the DCE. Examples of a DTE are a PC computer or a terminal. DTE devices can be mated to DTE devices by a special 'null-modem' cable that crosses the transmit signals of one device with the receive signals on the other device.

In Europe, the Comite Constultatif International Telephonique it Telegraphiqe (CCITT) has established standards that correspond to RS-232C. While these standards, CCITT V.24 and CCITT V.28, are very similar to RS-232C, they are not identical. The Model 4984 conforms to both RS-232 and CCITT V.24 standards, but does not contain or use all of the circuits allowed for in both standards.

A2.3 RS-422 AND RS-485 STANDARDS

In 1978, the EIA adopted the RS-422 standard to overcome the noise and distance problems associated with the single-ended RS-232 signals. The RS-422 standard specified a differential signal that used two lines per signal.

The RS-422 differential signals have the advantage of higher speed (up to 2Mbs) and longer distance capability (up to 1200M) over the single-ended RS-232 signals. The RS-422 differential signals require a differential receiver and are not referenced to Signal Ground. Differential transmitted signals applied to the interconnecting cable are +2 to +6V or -2 to -6V. Receivers are specified to have a ±0.2V sensitivity, 4Kohm minimum input impedance and be capable of withstanding a maximum input of ±10V. Cable terminators and transmitter wave shaping may be required to minimize cross talk. Logic levels are:

	+2 to +6V	-2 to -6V
Data A/B	0	1
Control A/B	1 (On)	0 (Off)

The differential transmitter output terminal that is positive with respect to the other terminal for the Control On Signal is designated the A terminal. The negative terminal is designated the B terminal. All voltage measurements are made by connecting a voltmeter between the A and B terminals.

RS-485 signals are similar to RS-422 signals except their transmitters are capable of driving up to 32 receivers and their protocol addresses individual devices.

A2.4 RS-530 PINOUTS

In 1987, the EIA released a new standard, EIA-530, for high speed signals on a 25 pin connector. This new standard combined the older RS-232 single-ended signals and the newer RS-422 differential signals on one connector. The advantage of the RS-530 specification is that it established a pin out standard for RS-422 signals on a 25 pin connector and at the same time provided for the presence of both signals on the same connector.

The 4894 serial interface conforms to the EIA-530 Standard and uses an internal jumpers to select the active signal levels on its serial interface. The 4894 is designed so that it receives either single ended RS-232 or differential RS-422/RS-485 signals.

A2.5 SERIAL INTERFACE PROBLEMS

Most of the problems that arise when connecting serial devices can be avoided if the user will compare the signals on both devices' interfaces before plugging them together. The obvious things to look for are:

1. Verify transmit and receive data direction and pin numbers. DTE devices mate directly with DCE devices while DTE and DTE connections need to be crossed.

2. Check needed control lines. Some devices need signal inputs, others can function with open inputs. All inputs need a valid signal level. If in doubt add jumpers to a known 'on' signal such as the devices's DTR or DSR output signal.

3. Same baud rates. Different baud rates result in garbled data.

   i.e.,*!1-

4. Same character formats. It may be obvious but often the character formats and parity settings are incorrect. A typical parity setting symptom is half good- half bad characters.

   i.e.,'1', '2', '4' good
   '3' and '5' bad

A3 GPIB CONNECTOR/SWITCH BOARDASSEMBLIES

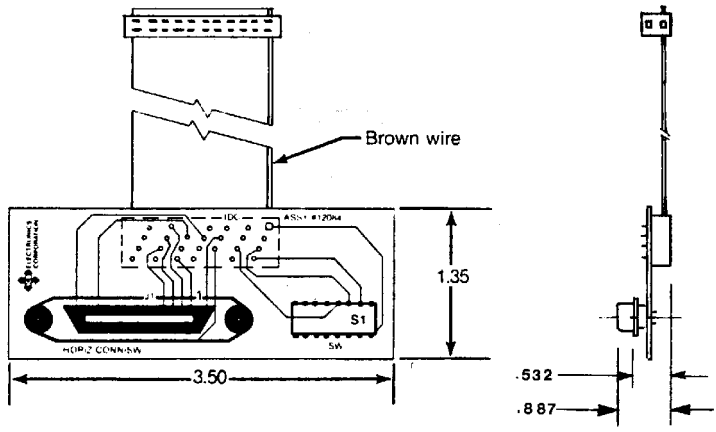
A3.1 INTRODUCTION

The Connector/Switch Board Assemblies are small printed circuit boards that provide a convenient way to mount an IEEE-488 Connector and an Address Switch on the rear of the host unit. They connect to the 4804 with a flat ribbon cable that plugs into the GPIB/ Address header (J2).

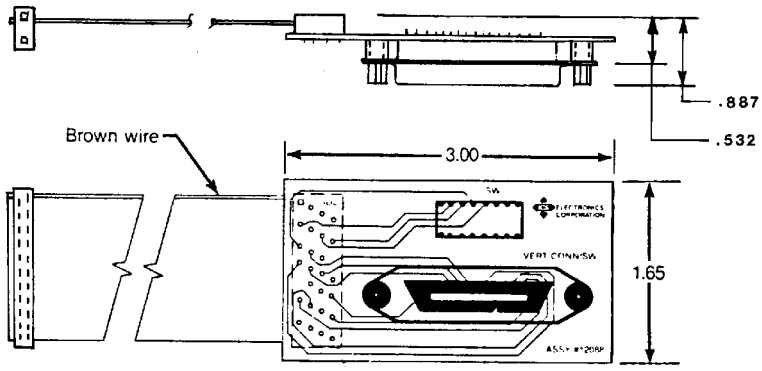
The Connector/Switch Board Assemblies are available in two layout styles. The Horizontal Connector/Switch Board Assembly has the Address Switch in line with the IEEE-488 connector as shown in Figure A-8(a). The Vertical Connector/Switch Board Assembly has the Address Switch located on top of the IEEE 488 connector as shown in Figure A-8(b).

The assemblies may be ordered with any length flat ribbon cable, from 10 to 90 cm long. The dash number specifies the cable length. Order as:

Type	**Part Number**
Horizontal Conn./Sw Assy with 90 cm long cable	113640-90
Vertical Conn./Sw Assy with 90 cm long cable	113642-90



(a) Horizontal Board Assembly



(b) Vertical Board Assembly

A3

Figure A-8 GPIB Connector/Switch Board Assemblies

A3.2 INSTALLATION

Both the Horizontal and the Vertical Connector/Switch Board Assemblies are designed to be mounted to the rear panel of the host equipment's rear panel by the included metric studs. The following are the suggested installation steps:

1. Select the appropriate cutout from Figure A-10

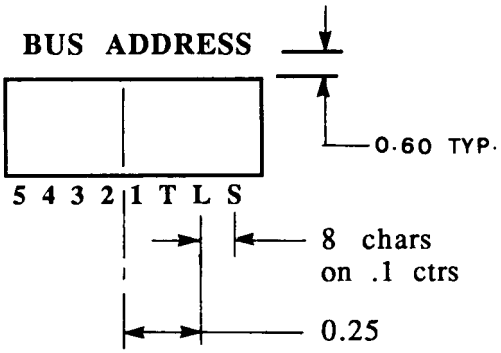
2. Locate a blank area or the host's rear panel. Leave enough room for the flat ribbon cable bend radius.

3. Machine the cutouts.

4. Install the Connector/Switch Assembly from the inside. Use the metric studs and two thin lock washers to hold the assembly to the panel.

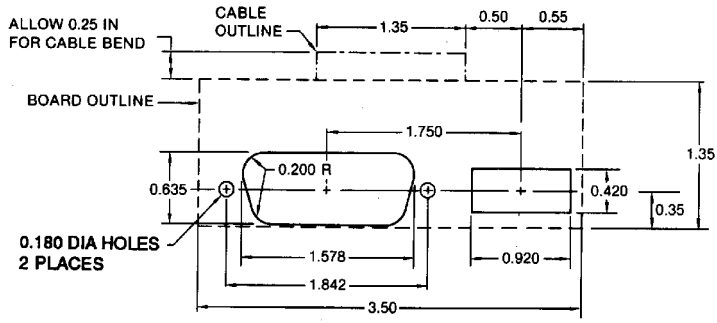
5. Route the flat ribbon cable to the 4804 so it avoids any high RFI or electrical noise area. Plug the cable into J2.

6. Mark or silkscreen the switch functions onto the rear panel as shown in Figure A-9

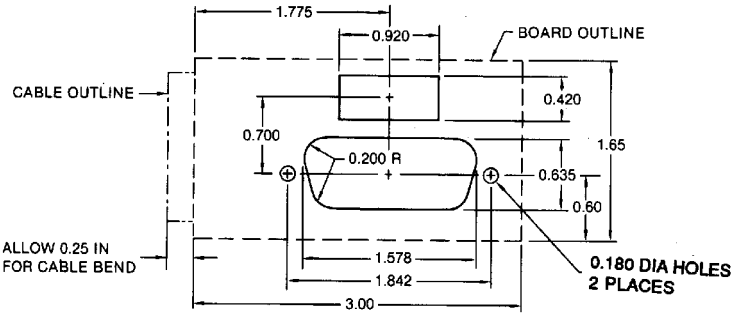


SILKCREEN DETAIL
Center titles over opening
All letters are helvetica medium

Figure A-9 Address Switch Silkscreen Detail



Horizontal Connector/Switch Board Cutouts



Vertical Connector/Switch Board Cutouts

A3

Note: The supplied metric stud is designed to be used with an 0.022 thick lock washer on a 0.048 thick panel. Thicker panels will need to be countersunk.

Figure A-10 GPIB Connector/Switch Board Cutouts

Index

Symbols

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U

V

4894A/4804 Errata

E.1 INTRODUCTION

This Errata contains additional information and changes to the 4894/ 4804 Instruction Manual since Revision 4 was printed.

E.2 GENERAL INFORMATION

An enhanced version of the Model 4894, called the Model 4894A, was released in December 1997. The 4894A has a faster processor higher baud rates. The 4894A and 4804 have been upgraded to handle RS-485 half-duplex data transmission and binary formats such as the Modbus.

E.3 EXPANDED BAUD RATES

Paragraph 1.6.1 on Page 1-6 should read

1.6.1 Baud Rate and Character Format

Baud Rate	4894A - Any rate from 50 to 230,400 baud. Unit sets baud rate divider to closest integer to the entered rate. Standard rates are: 50, 110, 300, 600, 1200, 2400, 4800, 7200, 9600, 14400, 19200, 28800, 38400, 57600, 92160, 115200 and 230400 baud 4804 - 300, 600, 1200, 2400, 4800, 9600, 19200 and 38400 baud.

Rev10/98

In Table 3-3, change :BAUD to:

:BAUD	9600	Sets baud rate to any value between 50 and 115200 baud. See paragraph 1.6.1 for a list of standard rates.

E.4 HALF-DUPLEX RS-485 CONNECTIONS

Add + and - designations to Figure 2-16 to simplify the connections to an RS-485 half-duplex device.

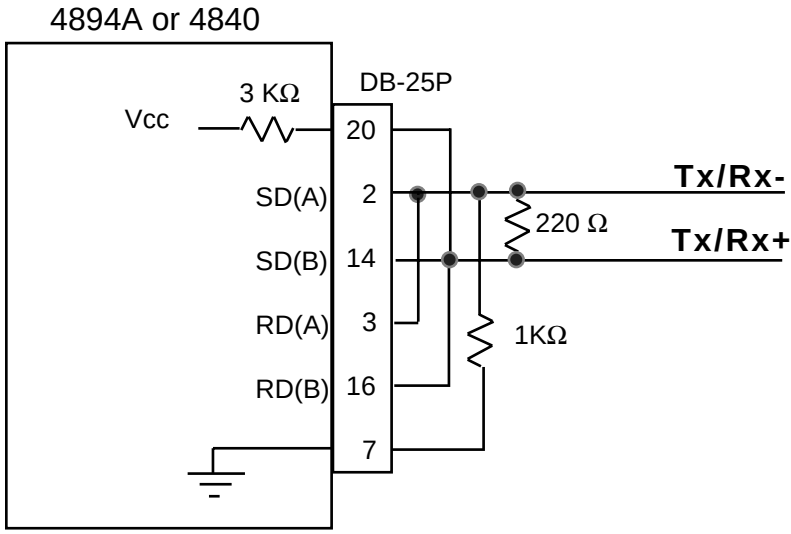


Figure 2-16 RS-485 Half-Duplex Connections

E.5 SELECTION OF HALF OR FULL-DUPLEX RS-485 OPERATION

Add a new command to the 4894A/4804 SCPI Command Tree in Table 3-2. This new command tristates the transmitter between messages and blocks the receiver from receiving the transmitted message. This saves the user from having to read and discard the transmitted message.

SYSTem:COMMunicate:SERial:RS485 <Boolean>

Add RS485 to Table 3-3

RS485	OFF	Enables full or half-duplex RS-485 operation. Off selects full-duplex RS-422 operation and the transmitter holds the line in the mark condition between messages. On selects half-duplex, RS-485 operation and the transmitter is tristated to free the line between messages. ON also blocks the receiver from receiving the transmitted message. Value is 1 or 0 or ON and OFF. The default for 4894A is ON, The default for the 4804 is OFF. The 4804 has the following delays between the last character and the time the line is tristated:

Baud Rate	Delay
1200	3.2 ms
2400	1.6 ms
9600	0.5 ms
38400	0.19 ms

E.6 GENERATING EOI ON LAST CHARACTER FROM THE RX BUFFER

4894As with Revision 20 and 4804s with Revision 7 or later firmware can be set to generate EOI when the last character in the Receive Serial Buffer is talked out on the GPIB bus regardless of the character's value. This capability makes it possible for a GPIB Controller to receive a string of binary data characters without knowing the byte count or timing out.

The conditions to enable this function are:

1. EOI generation to be ON
2. EOM Character be set to FF HEX
3. Add character addition be OFF

The commands for the above conditions are:

```
SYST:COMM:SER:EOI ON
SYST:COMM:SER:EOM 255
SYST:COMM:SER:ADD:CHAR OFF
```

Note: The user should ignore the Serial Message Received bit in the Questionable Register when receiving binary data.

E.7 RECOVERING FROM FINDLSTN COMMAND

The 488.2 FindLstn Protocol duplicates a part of the 4894A's and 4804's escape sequence. When the FindLstn Protocol is executed, the 4894A or 4804 will be left in the command mode. Use the SYSTem:OPER DATA command to put the unit back into its transparent data pass-thru mode.

If the FindLstn Protocol is used, it is normally done in the beginning of the program to create a list of devices on the bus. The list is saved and used for the AllSerialPoll and FindRQS Protocols.