E5900A Option 300 Emulation for the ARM7/ARM9—
At a Glance

The emulator provides a low-cost way to debug embedded software for a variety of ARM target systems.

**Emulation Probe**

The emulation probe is a stand-alone emulator. You can connect the emulation probe to a debug port on the target system through the provided target interface module (TIM). The emulation probe can be controlled by a debugger on a host computer or by the Emulation Control Interface on an Agilent Technologies 16600A/700A-series logic analysis system.

The emulation probe lets you use the target processor's built-in background debugging features, including run control and access to registers and memory.
**Emulation Module and Target Interface Module**

The emulation module plugs into your Agilent Technologies 16600A/700A-series logic analysis system frame. The emulation module lets you use the target processor’s built-in background debugging features, including run control and access to registers and memory. A high-level source debugger can use the emulation module to debug code running on the target system. You can connect the emulation module to a debug port on the target system through the provided target interface module (TIM).

**Emulation Migration**

The emulation migration includes a target interface module and firmware. Use the emulation migration if you already have an emulation probe or an emulation module for another processor.
In This Book

This book documents the following products:

**Emulator**

<table>
<thead>
<tr>
<th>Processors supported</th>
<th>Product ordered</th>
<th>Includes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARM7, ARM 9</td>
<td>Agilent Technologies E5900A Option #300</td>
<td>Agilent Technologies E3459A emulation probe, Agilent Technologies E3459-66501 target interface module (TIM)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Processors supported</th>
<th>Product ordered</th>
<th>Includes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARM7, ARM9</td>
<td>Agilent Technologies E5901A Option #300</td>
<td>Agilent Technologies 16610A emulation module, Agilent Technologies E3459-66501 target interface module (TIM)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Processors supported</th>
<th>Product ordered</th>
<th>Includes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARM7, ARM9</td>
<td>Agilent Technologies E5902A Option #300</td>
<td>Agilent Technologies E3459-66501 target interface module (TIM)</td>
</tr>
</tbody>
</table>
Contents

1 Overview

Setup Flowchart 13

Emulation Probe 14
Equipment supplied 14
Minimum equipment required 16

Power-On/Power-Off Sequence for the Emulation Probe 17
To power on the system 17
To power off the system 17
To connect the emulation probe to a power source 17

Connection Sequence (Emulation Probe only) 20

Emulation Module 21
Equipment supplied 21
Minimum equipment required 23

Emulation Migration 24
Equipment supplied 24
Minimum equipment required 25

Additional Information Sources 26
Contents

2 Connecting the Emulation Probe to a LAN
   Setting Up a LAN Connection to a PC or Workstation 29
   To obtain an IP address 30
   To configure LAN parameters using the built-in terminal interface 32
   To configure LAN parameters using BOOTP 35
   To set the 10BASE-T configuration switches 38
   To verify LAN communications 38

   Setting Up a Serial Connection (Emulation probe only) 40
   To set the serial configuration switches 40
   To connect a serial cable 42
   To verify serial communications 44

3 Installing the Emulation Module
   To install the emulation module in a 16700A-series logic analysis system or a
   16701A expansion frame 47
   To install the emulation module in a 16600A-series logic analysis system 49

4 Installing Software on a 16600A/700A
   To install the software from CD-ROM (16600A/700A) 53

5 Connecting the Emulator to the Target System
   To test the Agilent Technologies emulator 56
   To connect to a target system via the debug port 57
   Signal descriptions (14-pin connectors) 60
   ARM 14-pin connector 60
   TI 14-pin connector 60
   VLSI 14-pin connector 60
Contents

6 Configuring the Emulator

What can be configured 64
To configure using the Emulation Control Interface 65
To configure using the built-in commands 66
To configure using a debugger 68

Configuration Items 69
To set the type of ARM processor 69
To configure the availability of DBGACK and DBGREQ 70
To configure restriction to real-time runs 71
To configure the JTAG clock speed (communication speed) 72
To configure the endian mode of the target system 73
To configure the coprocessor register RAM address 73
To configure debug communications channel polling 74
To set up the debug communication channel 75
To configure the emulator to work with a coprocessor 76
To configure the reset behavior 78
To configure the vector catch register (ARM920T/940T/966ES only) 79
To configure the Trigger Out BNC (Emulation Probe only) 80
To configure the Break In BNC (Emulation Probe only) 80

Testing the emulator and target system 81
To test memory accesses 81
To test with a running program 82

7 Designing a Target System for the Emulator

Designing the JTAG connector 84
Signal descriptions (20-pin connector) 85
Electrical requirements for the TDO, TDI, TCK, TMS and nTRST signals 86
8 Using the Emulator with a Debugger

Setting up Debugger Software 90
To connect the logic analysis system to the LAN 91
To change the port number of an emulator 92
To verify communication with the emulator 92
To export the logic analysis system's display to a workstation 93
To export the logic analysis system's display to a PC 94

Using the ARM Debugger for Windows 95

9 Specifications and Characteristics

Processor compatibility 102
Emulation probe and emulation module—electrical characteristics 103
Emulation probe—electrical characteristics 104
Emulation probe—environmental characteristics 105
Emulation module—environmental characteristics 105

10 Updating Firmware

Emulation Probe Firmware 109
To display current firmware version information 109
To update firmware for an emulation probe 109

Emulation Module Firmware 110
To display current firmware version information 110
To update firmware for an emulation module using the Emulation Control Interface 110
To update firmware for an emulation module using the Setup Assistant 111
11 Solving Problems

Troubleshooting Guide 115
Emulation Probe Status Lights 116
Emulation Module Status Lights 118
Built-in Commands 118

Problems with the Target System 121
What to check first 121
To check the debug port connector signals 123
To interpret the initial prompt 124
If you see memory-related problems 127

Problems with the LAN Interface (Probe Only) 129
If you cannot verify LAN communication 129
If you have LAN connection problems 130
If the "POL" LED is lit 130
If it takes a long time to connect to the network 131

Problems with the Serial Interface (Emulation probe only) 132
If you cannot verify RS-232 communication 132
If you have RS-232 connection problems with the MS Windows Terminal program 133

Problems with the Emulation Probe Itself 134
To run the power up self test 134
To execute the built-in performance verification tests 137
To perform complete PV tests for an emulation probe 138
If a performance verification test fails 140

Returning the Parts to Agilent Technologies for Service 143
To return a part to Agilent Technologies 143
To obtain replacement parts 144
To clean the emulator 145
Contents

Glossary

Index
Overview
Overview

This chapter describes:

• Setup flowchart
• Equipment used with the emulation probe
• Connection sequences for the emulation probe
• Equipment used with the emulation module
• Additional information sources
Overview

Setup Flowchart

**Setup Flowchart**

1. **Emulation module** E5901A
   - Install emulation module (if necessary)
   - Install software on logic analysis system
   - Connect emulator to target interface module
   - Connect target interface module to target
   - Connect emulator
     - Connect emulator to target interface module
     - Connect target interface module to target
   - Update emulator firmware (if necessary)
   - Connect power supply
   - Connect to LAN

2. **Emulation probe** E5900A
   - Installation done. Begin making measurements.
Emulation Probe

Equipment supplied

- An emulation probe.
- A 12V power supply for the emulation probe.
- A power cord.
- A target interface module (TIM) circuit board.
- An emulation probe loopback test board (Agilent part number E3496-66502).
- A 50-pin ribbon cable (connects the emulation probe to the target interface module).
- A 14-pin low density ribbon cable (connects the target interface module to your target system).
- A 20-pin low density ribbon cable (connects the target interface module to your target system).
- A 20-pin high density ribbon cable (connects the target interface module to your target system).
- This User's Guide.
Overview
Emulation Probe

Minimum equipment required

The following equipment is required to use the emulation probe:

- A target system with the appropriate connector. The target system must meet the criteria on page 83.
- A host computer, such as a PC or workstation. You can also connect the emulation probe to an Agilent Technologies 16600A or 16700A logic analysis system.
- A LAN (local area network) to connect the emulation probe to the host computer.
- A user interface on the host computer, such as a high-level source debugger or the logic analysis system’s Emulation Control Interface.
Power-On/Power-Off Sequence for the Emulation Probe

To power on the system
With all components connected, power on your system as follows:
1 Logic analyzer, if you are using one.
2 Emulation probe.
3 Your target system.

To power off the system
Power off your system as follows:
1 Your target system.
2 Emulation probe.
3 Logic analyzer, if you are using one.

To connect the emulation probe to a power source
The emulation probe does not have an On/Off switch. To power on the emulation probe:
Overview

Power-On/Power-Off Sequence for the Emulation Probe

1. Connect the power cord to the power supply and to a socket outlet.

![Diagram of power cord connection]

2. Connect the 12V power cord to the back of the emulation probe.

![Diagram of power cord connection to emulation probe]

The power light on the target side of the emulation probe will be illuminated.
Overview

Power-On/Power-Off Sequence for the Emulation Probe

The emulation probe is shipped from the factory with a power supply and cord appropriate for your country. If the cord you received is not appropriate for your electrical power outlet type, contact your Agilent Technologies sales and service office.

**WARNING:**

Use only the supplied Agilent Technologies power supply and cord. Failure to use the proper power supply could result in electric shock.

**CAUTION:**

Use only the supplied Agilent Technologies power supply and cord. Failure to use the proper power supply could result in equipment damage.
Connection Sequence (Emulation Probe only)

Disconnect power from the target system, emulation probe, and logic analyzer before you make or break connections.

1. Connect the emulation probe to a LAN (page 28).
2. Connect the emulation probe to your target system (page 55).
3. Configure the emulation probe (page 63).
Emulation Module

Equipment supplied

The equipment supplied with your emulation module includes:

- An Agilent Technologies 16610A emulation module. If you ordered an emulation module as part of your Agilent Technologies 16600A or 16700A-series logic analysis system, it is already installed in the frame.
- A target interface module (TIM) circuit board.
- An emulation module loopback test board (Agilent part number E3496-66502).
- Firmware for the emulation module and software for the Emulation Control Interface on a CD-ROM.
- A 50-pin ribbon cable for connecting the emulation module to the target interface module.
- A 14-pin low density ribbon cable (connects the target interface module to your target system).
- A 20-pin low density ribbon cable (connects the target interface module to your target system).
- A 20-pin high density ribbon cable (connects the target interface module to your target system).
- One Torx T-10, on T-8, and one Torx T-15 screwdriver (if the emulation module was not installed at the factory).
- This User's Guide.
Minimum equipment required

The following equipment is required to use the emulation module:

• An Agilent Technologies 16600A or 16700A logic analysis system.
• A user interface, such as a high-level source debugger or the logic analysis system's Emulation Control Interface.
Emulation Migration

This section lists equipment supplied with the emulation migration and lists the minimum equipment required to use the emulation migration.

Equipment supplied

The equipment supplied with your emulation migration includes:

- A target interface module (TIM) circuit board.
- Firmware for the emulation module and/or updated software for the Emulation Control Interface on a CD-ROM.
- A 14-pin low density ribbon cable (connects the target interface module to your target system).
- A 20-pin low density ribbon cable (connects the target interface module to your target system).
- A 20-pin high density ribbon cable (connects the target interface module to your target system).
- This User’s Guide.
Minimum equipment required

The following equipment is required to use the emulation migration:

- An emulation module or emulation probe.
- A 50-pin data cable (supplied with the emulation module or probe).
- A method for connecting to the target system.
- A host computer such as a PC, a workstation, or an Agilent Technologies 16600A or 16700A logic analysis system.
- A user interface, such as a high-level source debugger or the logic analysis system's Emulation Control Interface.
Additional Information Sources

Additional or updated information can be found in the following places:

Newer editions of this manual may be available. Contact your local Agilent Technologies representative.

- Application notes may be available from your local Agilent Technologies representative or on the World Wide Web at:
  http://www.agilent.com/find/logicanalyzer
- If you have an Agilent Technologies 16600A or 16700A logic analysis system, the online help for the Emulation Control Interface has additional information on using the emulation module.
- The measurement examples include valuable tips for making emulation and analysis measurements. You can find the measurement examples under the system help in your Agilent Technologies 16600A/700A-series logic analysis system.
Connecting the Emulation Probe to a LAN
Connecting the Emulation Probe to a LAN

You can connect your PC, workstation, or logic analysis system to the emulation probe via a serial or LAN connection.

**Serial connection**

A serial connection allows you to complete all of the performance verification tests and set LAN parameters. Other use of the serial port is not supported.

**LAN connection**

A LAN connection will allow you to make your measurements quickly and easily. A few of the performance verification tests cannot be run over a LAN.

**Recommended connection**

Use a LAN connection for routine use, and a serial connection for LAN configuration and for troubleshooting.
Connecting the Emulation Probe to a LAN

Setting Up a LAN Connection to a PC or Workstation

The emulation probe has two LAN connectors:

- A BNC connector that can be directly connected to an IEEE 802.3 Type 10BASE2 cable (ThinLAN). When using this connector, the emulation probe provides the functional equivalent of a Medium Attachment Unit (MAU) for ThinLAN.

- An IEEE 802.3 Type 10BASE-T (StarLAN) connector.

Use either the 10BASE2 or the 10BASE-T connector. Do not use both. The emulation probe will not work with both connected at the same time.

You must assign an IP address (Internet address) to the emulation probe before it can operate on the LAN. You can also set other network parameters such as a gateway address. The IP address and other network parameters are stored in nonvolatile memory within the emulation probe.

The emulation probe automatically sets a subnet mask based on the subnet mask used by other devices on the network.

You can configure LAN parameters in any of the following ways:

- Using the built-in terminal interface over a serial connection. This is the most reliable method.
- Using BOOTP, which is part of the HP-UX, SunOS, and Solaris operating systems.
Connecting the Emulation Probe to a LAN

To obtain an IP address

1 Obtain the following information from your local network administrator or system administrator:

   • An IP address for the emulation probe.

   You can also use a "LAN name" for the emulation probe, but you must configure it using the integer dot notation (such as 127.0.0.1).

   • The gateway address.

   The gateway address is an IP address and is entered in integer dot notation. The default gateway address is 0.0.0.0, which allows all connections on the local network or subnet. If connections are to be made to workstations on other networks or subnets, this address must be set to the address of the gateway machine.

2 Find out whether port numbers 6470 and 6471 are already in use on your network and if that use constitutes a conflict.

   The host computer interfaces communicate with the emulation probe through two TCP service ports. The default base port number is 6470. The second port has the next higher number (default 6471).

   In almost all cases, the default numbers (6470, 6471) can be used without change. If necessary the default numbers can be changed if they conflict with some other product on your network.

   To change the port numbers, see page 32. If you have already set the IP address, you can use a telnet connection instead of a serial connection to connect to the emulation probe.
3 Write down the link-level address of the emulation probe.

You will need this address if you use BOOTP to set the IP address.

The link-level address (LLA) is printed on a label above the LAN connectors on the emulation probe. This address is configured in each emulation probe shipped from the factory and cannot be changed.

IP Address of Emulation probe __________________________
LAN Name of Emulation Probe __________________________
Gateway Address __________________________
Link-Level Address of Emulation Probe __________________________
To configure LAN parameters using the built-in terminal interface

1 Set configuration switches S1 through S4 to ON, and set the other switches as appropriate for your serial interface.

Switch settings are printed on the bottom of the emulation probe. If you will use a baud rate of 9600 baud, set the switches like this:

2 Connect an ASCII terminal (or terminal emulator) to the emulation probe’s RS-232 port with a 9-pin RS-232 cable.

Complete instructions for setting up a serial connection begin on page 40.

3 Plug in the emulation probe’s power cord. Press the terminal’s <RETURN> key a couple times. You should see a prompt such as "p>", "?>", or "R>".

At this point, you are communicating with the emulation probe’s built-in terminal interface.
Connecting the Emulation Probe to a LAN

4 Display the current LAN configuration values by entering the lan command:

R>
lan lan is disabled
lan -i 0.0.0.0
lan -g 0.0.0.0
lan -p 6470
Ethernet Address : 08000903212f

The "lan -i" line shows the current IP address (IP address) of the emulation probe.

The Ethernet address, also known as the link level address, is preassigned at the factory, and is printed on a label above the LAN connectors.

5 Enter the following command:

lan -i > [-g >] [-p >]

The lan command parameters are:

- i <internet> The IP address which you obtained from your network administrator.
- g <gateway> The gateway address. Setting the gateway address allows access outside your local network or subnet.
- p <port> This changes the base TCP service port number.
Connecting the Emulation Probe to a LAN

The default numbers (6470, 6471) can be changed if they conflict with some other product on your network. TCP service port numbers must be greater than 1024. If you change the base port, the new value must also be entered in the /etc/services file on the host computer. For example, you could modify the line:

```
hp64700 6470/tcp
```

The IP address and any other LAN parameters you change are stored in nonvolatile memory and will take effect the next time the emulation probe is powered off and back on again.

6 Disconnect the power cord from the emulation probe, and connect the emulation probe to your network.

This connection can be made by using either the 10BASE-T connector or the 10BASE2 (BNC) connector on the emulation probe. Do not use both connectors at the same time.

7 Set the configuration switches to indicate the type of connection that is to be made.

Switch S1 must be set to OFF, indicating that a LAN connection is being made. Switch S5 should be ON if you are connecting to the BNC connector:

```
1 2 3 4 5 6 7 8
0 1
```

Switch S5 should be OFF if you are connecting to the 10BASE-T connector:

```
1 2 3 4 5 6 7 8
0 1
```

Set all other switches to ON.

8 Connect the power cord to the emulation probe.

9 Verify your emulation probe is now active and on the network. See "To verify LAN communications" on page 38.

Once you have set a valid IP address, you can use the telnet utility to connect
Connecting the Emulation Probe to a LAN

to the emulation probe, and use the lan command to change LAN parameters.

**Example**

To assign an IP address of 192.6.94.2 to the emulation probe, enter the following command:

```bash
R>lan -i 192.6.94.2
```

Now, cycle power on the emulation probe so that the new address will take effect.

**See Also**

"Solving Problems," page 113, if you have problems verifying LAN communication.

---

**To configure LAN parameters using BOOTP**

Use this method only on a workstation which is running bootpd, the BOOTP daemon.

1. Make sure that BOOTP is enabled on your host computer.

   If the following commands yield the results shown below, the BOOTP protocol is enabled:

   ```bash
   $ grep bootp /etc/services
   bootps   67/udp
   bootpc   68/udp
   $ grep bootp /etc/inetd.conf
   bootps dgram udp wait root /etc/bootpd bootpd
   ```

   If the commands did not yield the results shown, you must either add BOOTP support to your workstation or use a different method to configure the emulation probe LAN parameters.
Connecting the Emulation Probe to a LAN

2 Add an entry to the host BOOTP database file, /etc/bootptab. For example:

```
# Global template for options common to all Agilent 64700 emulators and Emulation Probes.
# Use a different gateway addresses if necessary.
hp64700.global:\
    :gw=0.0.0.0:\
    :vm=auto:\
    :hn:\
    :bs=auto:\
    :ht=ether

# Specific emulator entry specifying hardware address (link-level address) and IP address.
hpprobe.div.hp.com:\
    :tc=hp64700.global:\
    :ha=080009090B0E:\
    :ip=192.6.29.31
```

In this example, the "ha=080009090B0E" identifies the link-level address of the emulation probe. The "ip=192.6.29.31" specifies the IP address that is assigned to the emulation probe. The node name is "hpprobe.div.hp.com".

3 Connect the emulation probe to your network.

This connection can be made by using either LAN connector on the emulation probe.

4 Set the configuration switches to indicate the type of connection that is to be made.

Switch S1 must be set to OFF, indicating that a LAN connection is being made.
Switch S6 must be set to OFF to enable BOOTP mode.
Switch S5 should be set to OFF if you are connecting to the BNC connector.

Switch S5 should be set to ON if you are connecting to the 10BASE-T.
Connecting the Emulation Probe to a LAN

Set all other switches to ON.

5 Connect the power cord to the emulation probe.

6 Verify that the power light stays on after 10 seconds.
   The IP address will be stored in EEPROM.

7 Set switch S6 back to ON.
   Do this so that the emulation probe does not request its IP address each time
   power is cycled. The IP address is stored in EEPROM, so BOOTP does not
   need to be run again. Leaving this switch on will result in slower performance,
   increased LAN traffic, and even failure to power up (if the BOOTP server
   becomes inactive).

8 Verify your emulation probe is now active and on the network. See "To
   verify LAN communications" on page 38.

See Also

For additional information about using bootpd, refer to the bootpd (1M) man
page.
Connecting the Emulation Probe to a LAN

To set the 10BASE-T configuration switches

Set switches S7 and S8 to ON unless one of the following conditions is true:

- If the LAN cable exceeds the standard length, set switch S7 to OFF.

The emulation probe has a switch-selectable, twisted-pair receiver threshold. With switch S7 set to OFF, the twisted-pair receiver threshold is lowered by 4.5 dB. This should allow you to use cable lengths of up to about 200 meters. If you use a long cable, you should consult with your LAN cabling installer to ensure that:

- The device at the other end of the cable has long cable capability, and
- The cable is high-grade, low-crosstalk cable with crosstalk attenuation of greater than 27.5 dB.

When switch S7 is set to ON, the LAN port operates at standard 10BASE-T levels. A maximum of 100 meters of UTP cable can be used.

- If your network doesn’t support Link Beat integrity checking or if the emulation probe is connected to a non 10BASE-T network (such as StarLAN) set this switch to LINK BEAT OFF (0 or OPEN).

In normal mode (switch S8 set to ON), a link integrity pulse is transmitted every 15 milliseconds in the absence of transmitted data. It expects to receive a similar pulse from the remote MAU. This is the standard link integrity test for 10BASE-T networks. If your network doesn’t support the Link Beat integrity checking or if the emulation probe is used on a non 10BASE-T network (such as StarLAN) set this switch to LINK BEAT OFF (OFF).

**NOTE:** Setting switch S8 to OFF when Link Beat integrity checking is required by your network will cause the remote MAU to disable communications.

To verify LAN communications

1. Verify your emulation probe is now active and on the network by issuing a telnet to the IP address.

   This connection will give you access to the emulation probe’s built-in terminal interface.
To view the LAN parameters, enter the lan command at the terminal interface prompt.

To exit from this telnet session, type <CTRL>D at the prompt.

The best way to change the emulation probe’s IP address, once it has already been set, is to telnet to the emulation probe and use the terminal interface lan command to make the change. Remember, after making your changes, you must cycle power or enter a terminal interface init -p command before the changes take effect. Doing this will break the connection and end the telnet session.

If you encounter problems, refer to the "Solving Problems" chapter (page 113).

Example

$ telnet 192.35.12.6
R>lan
lan is enabled
lan -i 192.35.12.6
lan -g 0.0.0.0
lan -p 6470
Ethernet Address : 08000F09B30
Connecting the Emulation Probe to a LAN

Setting Up a Serial Connection (Emulation probe only)

Setting Up a Serial Connection (Emulation probe only)

To set up a serial connection, you will need to:

- Set the serial configuration switches
- Connect a serial cable between the host computer and the emulation probe
- Verify communications

Serial connections on a workstation

If you are using a UNIX workstation as the host computer, you need to use a serial device file. If a serial device file does not already exist on your host, you need to create one. Once it exists, you need to ensure that it has the appropriate permissions so that you can access it. See the system documentation for your workstation for help with setting up a serial device.

Serial connections on a PC

Serial connections are supported on PCs. You must use hardware handshaking if you will use the serial connection for anything other than setting LAN parameters.

If you are using a PC as the host computer, you do not need to set up any special files.

To set the serial configuration switches

1. Set switch S1 to ON (RS-232).
2. Set switches S2-S4 to ON.
3. Set switch S5 to ON (HW HANDSHAKE ON) if your serial interface uses the DSR:CTS/RTS lines for flow control. Set S5 to OFF (HW HANDSHAKE OFF) if your serial interface uses software flow control (XON/XOFF).

If your serial interface supports hardware handshaking, you should use it (set switch S5 to ON). Hardware handshaking will make the serial connection
4 Set switches S6-S8 for the baud rate you will use. These switch settings are listed on the bottom of the emulation probe.

The higher baud rates may not work reliably with all hosts and user interfaces. Make sure the baud rate you choose is supported by your host and user interface.

Example

To use a baud rate of 9600 baud, set the switches as follows:
Connecting the Emulation Probe to a LAN
Setting Up a Serial Connection (Emulation probe only)

To connect a serial cable

**CAUTION:** Use a grounded, shielded cable. If the cable is not shielded, or if the cable is not grounded at the serial controller, the emulation probe may be damaged by electrostatic discharge.

Connect an RS-232C modem cable from the host computer to the emulation probe. The recommended cable is HP part number C2932A. This is a 9-pin cable with one-to-one pin connections.

If you want to build your own RS-232 cable, follow the pinout shown in the following figure:

![Pinout Diagram](image-url)
### Serial Cable Pinout

<table>
<thead>
<tr>
<th>Pin Number</th>
<th>Signal</th>
<th>Signal Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DCD</td>
<td>Data Carrier Detect (not used)</td>
</tr>
<tr>
<td>2</td>
<td>TD</td>
<td>Transmit Data (data coming from Agilent Technologies emulation probe)</td>
</tr>
<tr>
<td>3</td>
<td>RD</td>
<td>Receive Data (data going to Agilent Technologies emulation probe)</td>
</tr>
<tr>
<td>4</td>
<td>DTR</td>
<td>Data Terminal Ready (not used)</td>
</tr>
<tr>
<td>5</td>
<td>GND</td>
<td>Signal Ground</td>
</tr>
<tr>
<td>6</td>
<td>DSR</td>
<td>Data Set Ready (Output from Agilent Technologies emulation probe)</td>
</tr>
<tr>
<td>7</td>
<td>RTS</td>
<td>Request to Send (Input to Agilent Technologies emulation probe)</td>
</tr>
<tr>
<td>8</td>
<td>CTS</td>
<td>Clear to Send (connected to pin 6)</td>
</tr>
<tr>
<td>9</td>
<td>RING</td>
<td>Ring Indicator (not used)</td>
</tr>
</tbody>
</table>
To verify serial communications

1 Start a terminal emulator program on the host computer.

   If you are using a PC, the Terminal application in Microsoft Windows will work fine.

   If you are using a UNIX workstation, you can use a terminal emulator such as cu or kermit.

2 Plug the power cord into the emulation probe.

   When the emulation probe powers up, it sends a message (similar to the one that follows) to the serial port and then displays a prompt:

   Copyright (c) Agilent Technologies 1987
   All Rights Reserved. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under copyright laws.

   E3499B Series Emulation System
   Version:  A.07.57 04Mar99
   Location:  Generics

   E3459B ARM7/9 JTAG Emulator
   Version:  B.02.00 17Jun99 10:42
   R>

   The version numbers may be different for your emulation probe.

3 Press the Return or Enter key a few times.

   You should see a prompt such as "p>", "R>", or "?>".

   For information about the commands you can use, enter ? or help at the prompt.

See Also

"Problems with the Serial Interface," page 132.
Installing the Emulation Module
Installing the Emulation Module

This chapter shows you how to install an emulation module in your Agilent Technologies 16600A/700A-series logic analysis system.

If your emulation module is already installed in your logic analysis system frame, you may skip this chapter.

**CAUTION:** Electrostatic discharge can damage electronic components. Use grounded wrist straps and mats when you handle modules.
To install the emulation module in a 16700A-series logic analysis system or a 16701A expansion frame

You will need T-10 and T-15 Torx screw drivers.

1. Turn off the logic analysis system and REMOVE THE POWER CORD.
   Remove any other cables (including mouse or video monitor cables).

2. Turn the logic analysis system frame upside-down.

3. Remove the bottom cover.

4. Remove the slot cover.
   You may use either slot.
Installing the Emulation Module

5 Install the emulation module.

6 Connect the cable and re-install the screws.

You may connect the cable to either of the two connectors. If you have two emulation modules, note that many debuggers will work only with the "first" module: the one toward the top of the frame ("Slot 1"), plugged into the connector nearest the back of the frame.

7 Reinstall the bottom cover, then turn the frame right-side-up.

8 Plug in the power cord, reconnect the other cables, and turn on the logic analysis system.

The new emulation module will be shown in the system window.
To install the emulation module in a 16600A-series logic analysis system

You will need T-8, T-10, and T-15 Torx screw drivers (supplied with the modules).

1 Turn off the logic analysis system and REMOVE THE POWER CORD.
   Remove any other cables (such as probes, mouse, or video monitor).

2 Slide the cover back.

3 Remove the slot cover.

4 Install the emulation module.
Installing the Emulation Module

5 Connect the cable and re-install the screws.

6 Reinstall the cover.
   Tighten the screws snugly (2 N·m or 18 inch-pounds).

7 Plug in the power cord, reconnect the other cables, and turn on the logic analysis system.
   The new emulation module will be shown in the system window.
Installing Software on a 16600A/700A
This chapter explains how to install the software you will need for your emulator.

**Installing and loading**

Installing the software will copy the files to the hard disk of your logic analysis system. Later, you will need to load some of the files into the appropriate hardware module.

**What needs to be installed**

If you ordered an emulation module with your logic analysis system, the software was installed at the factory.

The following files are installed from the CD-ROM:

- Emulation module firmware
- Emulation Control Interface
- Other files which support logic analysis
To install the software from CD-ROM (16600A/700A)

Installing a processor support package from a CD-ROM will take just a few minutes. If the processor support package requires an update to the Agilent Technologies 16600A/700A operating system, installation may take approximately 15 minutes.

If the CD-ROM drive is not connected, see the instructions printed on the CD-ROM package.

1 Turn on the CD-ROM drive first and then turn on the logic analysis system.

   If the CD-ROM and analysis system are already turned on, be sure to save any acquired data. The installation process may reboot the logic analysis system.

2 Insert the CD-ROM in the drive.

3 Click the System Admin icon.

4 Click the Software Install tab.

5 Click Install....

   Change the media type to "CD-ROM" if necessary.

6 Click Apply.

7 From the list of types of packages, double-click "PROC-SUPPORT."

   A list of the processor support packages on the CD-ROM will be displayed.

8 Click on the "ARM" package.

   If you are unsure whether this is the correct package, click Details for information about the contents of the package.

9 Click Install.

   The Continue dialog box will appear.

10 Select Continue.

   The Software Install dialog will display "Progress: completed successfully" when the installation is complete.
If required, the system will automatically reboot. Otherwise, close the software installation windows.

**See Also**

The instructions printed on the CD-ROM package for a summary of the installation instructions.

The online help for more information on installing, licensing, and removing software.
Connecting the Emulator to the Target System
Connecting the Emulator to the Target System

The Agilent Technologies emulator is connected to a target system through one of the IEEE 1149.1 (JTAG) Port connectors. The included connectors are a 14-pin ARM Embedded Ice connector, a 14-pin VLSI connector, a 14-pin TI connector, a 20-pin low density ARM/Agilent connector, and a 20-pin high density ARM/Agilent connector.

To test the Agilent Technologies emulator

If this is the first time that you have used the emulator, you should run the built-in performance verification test before you connect to a target system. Refer to the "Solving Problems" chapter (beginning on page 113) for information on performance verification.
To connect to a target system via the debug port

In order to connect the emulator to the microprocessor, one of the supported connectors must be available on the target system (see page 83 for information on designing a target system for use with the emulator).

1. Remove power from the target system and the emulator.
2. Plug one end of the 50-pin cable into the emulator.
3. Plug the other end of the 50-pin cable into the target interface module.
4. Choose the appropriate ribbon cable for the connector that is on your target system.
5. For a target system using either of the 20-pin headers, connect one end of the ribbon cable to the target interface module.
Connecting the Emulator to the Target System

**ARM/Agilent 20-pin Connector***

*Recommended for new designs.

For a target system using the 14-pin header, match the connector on the target system with one of the following diagrams.

**ARM Embedded ICE 14-pin Connector**
Connecting the Emulator to the Target System

### TI XDS510 14-pin Connector

<table>
<thead>
<tr>
<th>Pin 1: TMS</th>
<th>Pin 2: nTRST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pin 3: TDI</td>
<td>Pin 4: GND</td>
</tr>
<tr>
<td>Pin 5: TVCC</td>
<td>No pin (key)</td>
</tr>
<tr>
<td>Pin 7: TDO</td>
<td>Pin 8: GND</td>
</tr>
<tr>
<td>Pin 9: TCK RET</td>
<td>Pin 10: GND</td>
</tr>
<tr>
<td>Pin 11: TCK</td>
<td>Pin 12: nICERST</td>
</tr>
<tr>
<td>Pin 13: EMU0</td>
<td>Pin 14: EMU1</td>
</tr>
</tbody>
</table>

* = PN[2:0] and PT are used for the VLSI MDP board and are grounded on other VLSI targets.

### VLSI Jumpstart 14-pin Connector

**CAUTION:**

Connect only one ribbon cable to the target interface module at any one time. Connecting several cables may result in damage to the target system.

6. Plug the ribbon cable into the appropriate connector on the target interface module.

7. Plug the other end of the selected ribbon cable into the debug port on the target system.

8. Turn on the power to the emulator, then turn on the power to the target system.
Connecting the Emulator to the Target System

Signal descriptions (14-pin connectors)

This section describes signals which differ from the descriptions of the signals for the Agilent/ARM 20-pin connector described on page 85.

ARM 14-pin connector

TDO, TDI, TMS, TCK: Same as Agilent/ARM 20 pin

Vref signal. The Vref signal is used to sense when the target system is connected to the emulator. This signal is also used as a reference voltage for all signals driven by the emulator. The emulator does not use this pin for power.

nTRST: Input: The nTRST signal is the JTAG reset signal. Most target systems implement a reset by holding TMS high for 5 consecutive clock cycles. nTRST can be used as a substitute for that operation.

nSRST: Input: target reset.

TI 14-pin connector

TDO, TDI, TMS, TCK: Same as Agilent/ARM 20 pin.

TVCC: Output from target system; 0 to 1 volt → system is not powered. The emulator will not drive TDO, TDI, TMS, TCK. 1 to 5 V → the target system is powered. The emulator will drive TDO, TDI, TMS, TCK.

EMU0, EMU1: No connection on emulator.

nI CerST: Input: target reset.

VLSI 14-pin connector

TDO, TDI, TMS, TCK: Same as Agilent/ARM 20 pin.

+5V: Output. No connection on emulator.

3.3V: Output. 0 to 1 volt → system is not powered. The emulator will not drive TDO, TDI, TMS, TCK. 1 to 5 V → the target system is powered. The
emulator will drive TDO, DTI, TMS, TCK.

**nRST:** Input: target reset.

**PN0/GND, PN1/GND, PN2/GND, PT/GND:** Target input. On most VLSI targets these pins are grounded. On the VLSI MDP board these pins are inputs to the target which select which processor is being addressed. The emulator drives these pins with a high on PT and a low on PN0, PN1, PN2.
Connecting the Emulator to the Target System
Configuring the Emulator
Configuring the Emulator

The emulator has several user-configurable options. These options may be customized for specific target systems and saved in configuration files for future use.

| The easiest way to configure the emulator is through the Emulation Control Interface in an Agilent Technologies 16600A or 16700A logic analysis system. |

If you use the Emulation Control Interface, please refer to the online help in the Configuration window for information on each of the configuration options.

Other ways to configure the emulator are by using:

- the emulator's built-in terminal interface
- your debugger, if it provides an “emulator configuration” window which can be used with this Agilent Technologies emulator

**What can be configured**

The following options can be configured using the Emulation Control Interface or using built-in commands:

- Processor type
- Availability of DBGACK and DBGRQ signals
- Restriction to real-time runs
- JTAG clock speed
- Endian mode
- Debug communications channel
- Coprocessor register RAM address
- Reset behavior
- Vector catch register
- Trigger out behavior
- Break in behavior
To configure using the Emulation Control Interface

The easiest way to configure the emulator is to use the Emulation Control Interface.

1 Start an Emulation Control Interface session.

   In the system window, click the Emulation Control Interface icon, and then select **Start Session...**.

2 Open a Configuration window.

   Select **Configuration...** from the Emulation Control Interface icon or from the Navigate menu in any Emulation Control Interface window.

3 Set the configuration options, as needed.

   Configuration changes will take effect when you close the configuration window or when you move the mouse pointer outside the window.

4 Save the configuration settings.

   To save the configuration settings, open the File Manager window and click **Save...**.
To configure using the built-in commands

If you are unable to configure the emulator with the Emulation Control Interface or a debugger interface, you can configure the emulator using the built-in “terminal interface” commands.

1 Connect a telnet session to the emulator over the LAN.

   For example, on a UNIX system, for an emulator in Slot 1 enter:

   \texttt{telnet LAN\_address 6472}

2 Enter \texttt{cf} to see the current configuration settings.

3 Use the \texttt{cf} command to change the configuration settings.

\textbf{See Also}

Enter \texttt{help cf} for help on the configuration commands.

For information on connecting using telnet, and for information on other built-in commands, see 118.
Configuring the Emulator

Example

To see a complete list of configuration items, type `help cf`. This command displays:

```
cf - display or set emulation configuration
    cf - display current settings for all config items
    cf <item> - display current setting for specified <item>
    cf <item>=<value> - set new <value> for specified <item>
    cf <item>=<value> <item> - set and display can be combined
    help cf <item> - display long help for specified <item>

--- VALID CONFIGURATION <item> NAMES ---
    proc - Set type of Arm Processor
    reset - RESET actions
    rrt - Enable or Disable Restriction to Real-Time Runs
    breakin - Select BNC break input option
    trigout - Select BNC trigger output option
    dbgack - Toggle availability of DBGACK
    dbgreq - Toggle availability of DBGREQ
    speed - Set JTAG Clock Divisor
    endian - Specify Big or Little Endian
    cparea - RAM region used by cpreg command
    comms - Enable debug communications channel polling

M>
```

To see a more detailed description of any configuration item, use the command “help cf <item>”. For example:

```
M>help cf rrt
    Enable or Disable Restriction of Real-Time Runs
    cf rrt=no
    cf rrt=yes

When enabled and while the processor is running the user program, any
command that requires a break to the monitor will be rejected except ‘rst’,
‘b’, ‘z’ or ‘s’.

When disabled, commands that require a break to the monitor will alway be
accepted.

M>
```
Configuring the Emulator

To see a list of the current configuration settings, use “cf”:

M>cf
  cf proc=Arm7TDMI
  cf reset=auto
  cf rrt=no
  cf breakin=rising
  cf trigout=monhigh
  cf dbgack=no
  cf dbgreq=no
  cf speed=3
  cf endian=little
  cf cparea=20
  cf comms=off

To configure using a debugger

Because the Agilent Technologies emulator can be used with several third-party debuggers, specific details for sending the configuration commands from the debugger to the emulator cannot be given here. However, all debuggers should provide a way of directly entering terminal mode commands to the emulator. Ideally, you would create a file that contains the modified configuration entries to be sent to the emulator at the beginning of each debugger session.

See Also

Information about specific debuggers in Chapter 8, “Using the Emulator with a Debugger,” beginning on page 87.

Your debugger manual.
Configuring the Emulator

Configuration Items

The configuration items must be set correctly for each target system.

To set the type of ARM processor

<table>
<thead>
<tr>
<th>Processor</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arm7TDMI (default)</td>
<td>cf proc=Arm7TDMI</td>
</tr>
<tr>
<td>Arm7DI</td>
<td>cf proc=Arm7DI</td>
</tr>
<tr>
<td>Arm710T</td>
<td>cf proc=Arm710T</td>
</tr>
<tr>
<td>Arm720T</td>
<td>cf proc=Arm720T</td>
</tr>
<tr>
<td>Arm740T</td>
<td>cf proc=Arm740T</td>
</tr>
<tr>
<td>Arm 920T</td>
<td>cf proc=Arm920T</td>
</tr>
<tr>
<td>Arm 940T</td>
<td>cf proc=Arm940T</td>
</tr>
<tr>
<td>Arm 966ES</td>
<td>cf proc=Arm966ES</td>
</tr>
</tbody>
</table>

The `cfsave -s` command will store the processor type configuration in the emulator’s flash memory. The `cfsave -r` command will restore this configuration.
To configure the availability of DBGACK and DBGRQ

The Agilent Technologies emulator will make use of the DBGACK and DBGRQ signals if they are available on the debug port connector.

The DBGACK signal allows the emulator to quickly detect entry or exit from debug mode. Also, the emulator is able to start or stop the logic analyzer through the "TRIGGER OUT" of the emulator.

The DBGRQ signal is used to quickly enter debug mode after receiving a "BREAK IN" signal from the logic analyzer. This allows the logic analyzer triggering capability to be used for complex breakpoints.

To make use of these signals, the emulator must be configured correctly. The following cf commands allow the specification of whether each signal is connected or not.

<table>
<thead>
<tr>
<th>Emulator configuration</th>
<th>Value</th>
<th>Emulator configured for</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>yes</td>
<td>The corresponding signal is connected and will be used by the emulator.</td>
<td>cf dbgack=yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>cf dbgreq=yes</td>
</tr>
<tr>
<td></td>
<td>no</td>
<td>The corresponding signal is not connected and will not be used by the emulator. (Default)</td>
<td>cf dbgack=no</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>cf dbgreq=no</td>
</tr>
</tbody>
</table>
To configure restriction to real-time runs

Real-time runs configuration

<table>
<thead>
<tr>
<th>Value</th>
<th>Emulator configured for</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td>Allows commands which break to the monitor. Examples include commands which display memory or registers. These commands break to the monitor to access the target processor, then resume the user program. (Default)</td>
<td>cf rrt=no</td>
</tr>
<tr>
<td>yes</td>
<td>No commands are allowed which break to the monitor, except “break,” “reset,” “run,” or “step.” The processor must be explicitly stopped before these commands can be performed.</td>
<td>cf rrt=yes</td>
</tr>
</tbody>
</table>

If your debugger allows displaying or modifying memory or registers while the processor is running, you must set rrt=no in order to use this feature.
To configure the JTAG clock speed (communication speed)

The Agilent Technologies emulator needs to be configured to communicate at a rate which is compatible with your target processor. The JTAG Clock speed is independent of processor clock speed. In general, speed=1 can always be used and provides the best performance. With some target systems that have additional loads on the JTAG lines or with target systems that do not quite meet the requirements described in the "Designing a Target System" chapter (page 83), setting speed to a slower setting may enable the probe to work. Some targets also have problems if the JTAG clock speed is set too slow, particularly at the 156 kHz setting.

**TCK clock speed configuration**

<table>
<thead>
<tr>
<th>Value</th>
<th>TCK clock is at least</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10 MHz (default)</td>
<td>cf speed=1</td>
</tr>
<tr>
<td>2</td>
<td>5 MHz</td>
<td>cf speed=2</td>
</tr>
<tr>
<td>3</td>
<td>2.5 MHz</td>
<td>cf speed=3</td>
</tr>
<tr>
<td>4</td>
<td>1.25 MHz</td>
<td>cf speed=4</td>
</tr>
<tr>
<td>5</td>
<td>625 kHz</td>
<td>cf speed=5</td>
</tr>
<tr>
<td>6</td>
<td>312 kHz</td>
<td>cf speed=6</td>
</tr>
<tr>
<td>7</td>
<td>156 kHz</td>
<td>cf speed=7</td>
</tr>
</tbody>
</table>
To configure the endian mode of the target system

Because the ARM7 processor is capable of being configured as either big-endian or little-endian, the emulator must know which configuration the target system is configured for.

**Processor endian configuration**

<table>
<thead>
<tr>
<th>Value</th>
<th>Emulator configured for</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td>little</td>
<td>Little endian (Default)</td>
<td>cf endian=little</td>
</tr>
<tr>
<td>big</td>
<td>Big endian</td>
<td>cf endian=big</td>
</tr>
</tbody>
</table>

To configure the coprocessor register RAM address

Reading and writing coprocessor registers requires a small area of target system RAM. The contents of this RAM is saved before the RAM is used and the original contents is restored after it is used. The `cparea` configuration item specifies the starting address of an 80 byte block of target system memory that must be readable and writable by the emulator. The default address is 20.

For example:

```
cfcparea=40
```

will start the `cparea` at address 40.
To configure debug communications channel polling

### Debug communication channel configuration

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td>off</td>
<td>Disables polling for the debug communications channel.</td>
<td>cf comms=off</td>
</tr>
<tr>
<td>on</td>
<td>Enables polling for the debug communications channel</td>
<td>cf comms=on</td>
</tr>
</tbody>
</table>
To set up the debug communication channel

Use the following commands to setup and use the debug communication channel.

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
</table>
| comm -c | Displays comms control register status.  
Example Output:  
control=0 Write and read registers are clear.  
control=1 Data written to the comms data read register.  
control=2 Data available in the comms data write register.  
control=3 Data available in both registers. |
| comm -d | Display comms data buffer (max=64 values)  
Example output:  
Count=2  
xxxxxxxxx  
xxxxxxxxx |
| comm -m <count>, <value0>, <value1>... | Modify comms data read register. |
| comm -m <count> | Number of values to be entered (max=64). |
| comm -r | Clear the write buffer. |
| comm -v | Enables verbose messages for data in comms data buffer. |
| comm -v enable | Enables verbose messages for data in comms data buffer.  
(Same as comm -v.) |
| comm -v disable | Disables verbose messages for data in comms data buffer. |
| comm -w | Display the number of words in the comms write buffer |

**See Also**
The ARM7 data sheet for more information on the debug communications channel.
To configure the emulator to work with a coprocessor

The firmware supports the description and access of coprocessors for the ARM core. Use the following commands to describe coprocessors.

```
cpdesc -d <CP#> <type> <reg> <size> <nbit> <rdbits>
cpdesc -r <CP#> R <reg> <size> <read0> <read1>
cpdesc -r <CP#> W <reg> <size> <write0> <write1>
cpdesc -r <CP#> B <reg> <size> <read0> <read1> <write0> <write1>
```

cpreg [-r <read0>] <CP#> <reg> - displays coprocessor registration

cpreg -s [-w <write0>] <CP#> <reg> <value1> [<value2>] - sets the processor register

cpdesc <CP#> - displays the description for the specified CP

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;CP#&gt;</td>
<td>The coprocessor number</td>
</tr>
<tr>
<td>&lt;type&gt;</td>
<td>R, W, or B for read, write, or both</td>
</tr>
<tr>
<td>&lt;reg&gt;</td>
<td>A single register number or range (eg: 1 or 3..6)</td>
</tr>
<tr>
<td>&lt;size&gt;</td>
<td>The size of register in bytes (hex)</td>
</tr>
<tr>
<td>&lt;nbit&gt;</td>
<td>Bit 22 in an LDC or SDC instruction</td>
</tr>
<tr>
<td>&lt;rdbits&gt;</td>
<td>Bits 15..12 in an LDC or SDC instruction</td>
</tr>
<tr>
<td>&lt;read0&gt;</td>
<td>Bits 7..0 in an MRC instruction</td>
</tr>
<tr>
<td>&lt;read1&gt;</td>
<td>Bits 23..16 in an MRC instruction</td>
</tr>
<tr>
<td>&lt;write0&gt;</td>
<td>Bits 7..0 in an MCR instruction</td>
</tr>
<tr>
<td>&lt;write1&gt;</td>
<td>Bits 23..16 in an MCR instruction</td>
</tr>
</tbody>
</table>

**NOTE:**

cpdesc for a register already specified will overwrite previous data.
Example usage of cpdesc and cpreg

To setup a description to read coprocessor 15 register 0 (ID register) for the ARM720T.

cpdesc -r 15 R 0 4 00 00  #ID register, read only
creg 15 0               #Reads CP15 c0
Configuring the Emulator

Configuring the Emulator
Configuration Items

To configure the reset behavior

The reset configuration item controls what happens when a reset occurs.

**Reset configuration**

<table>
<thead>
<tr>
<th>Value</th>
<th>Emulator configured for</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td>auto</td>
<td>A RESET will break into monitor and restore watchpoint registers only when watchpoint registers are being used. (Default)</td>
<td>cf reset=auto</td>
</tr>
<tr>
<td>tomon (break)</td>
<td>A RESET will break into monitor. Watchpoint registers will be restored. The processor will remain in monitor until another command causes it to exit.</td>
<td>cf reset=tomon</td>
</tr>
<tr>
<td>touser (continue)</td>
<td>A RESET will not break into monitor. Watchpoint registers will not be restored.</td>
<td>cf reset=touser</td>
</tr>
<tr>
<td>restart</td>
<td>A RESET will break into monitor. Watchpoint registers will be restored and the user program will begin running.</td>
<td>cf reset=restart</td>
</tr>
</tbody>
</table>

Note that these actions occur only when the RESET signal is deasserted. Selecting the Reset button in the Emulation Control Interface (or using the built-in rst command) asserts RESET continuously until another command causes RESET to be deasserted.
To configure the vector catch register (ARM920T/940T/966ES only)

This configuration item sets and displays the vector catch register, which defines which conditions will cause the processor to enter debug mode.

**Vector catch register configuration**

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>FRQ</td>
<td>cf vcatch=value</td>
</tr>
<tr>
<td>6</td>
<td>IRQ</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>D_Abort</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>P_Abort</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>SWI</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Undef</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>Reset</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

To set the vector catch register for SWI and RESET, set it to binary 00000101:

```
cf vcatch=0x5
```
To configure the Trigger Out BNC (Emulation Probe only)

With an emulation module, this configuration item is always set to the default setting and cannot be changed with a cf command. The Intermodule window of the logic analyzer must be used instead.

**Trigger out configuration**

<table>
<thead>
<tr>
<th>Value</th>
<th>The Trigger Out BNC will</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td>fixhigh</td>
<td>Always be high</td>
<td>cf trigout=fixhigh</td>
</tr>
<tr>
<td>fixlow</td>
<td>Always be low</td>
<td>cf trigout=fixlow</td>
</tr>
<tr>
<td>monhigh</td>
<td>Go high when the processor is running in background (Default)</td>
<td>cf trigout=monhigh</td>
</tr>
<tr>
<td>monlow</td>
<td>Go low when the processor is running in background</td>
<td>cf trigout=monlow</td>
</tr>
</tbody>
</table>

To configure the Break In BNC (Emulation Probe only)

With an emulation module, this configuration item is always set to the default setting and cannot be changed with a cf command. The Intermodule window of the logic analyzer must be used instead.

**Break in configuration**

<table>
<thead>
<tr>
<th>Value</th>
<th>The Break In BNC will</th>
<th>Built-in command</th>
</tr>
</thead>
<tbody>
<tr>
<td>off</td>
<td>Inputs to the Break In BNC will be ignored.</td>
<td>cf breakin=off</td>
</tr>
<tr>
<td>rising</td>
<td>The emulator will cause a break on a rising edge. (Default)</td>
<td>cf breakin=rising</td>
</tr>
<tr>
<td>falling</td>
<td>The emulator will cause a break on a falling edge.</td>
<td>cf breakin=falling</td>
</tr>
</tbody>
</table>
Testing the emulator and target system

After you have connected and configured the emulator, you should perform some simple tests to verify that everything is working.

See Also

Chapter 11, “Solving Problems,” beginning on page 113, for information on testing the emulator hardware.

To test memory accesses

1. Start the Emulation Control Interface and configure the emulator, if necessary.

2. Open the Memory window.

3. Write individual locations or fill blocks of memory with patterns of your choosing.

   The access size is the size of memory access that will be used to write or read the memory values.

4. Use the Memory I/O window to stimulate I/O locations by reading and writing individual memory locations.
To test with a running program

To more fully test your target, you can load simple programs and execute them.

1. Compile or assemble a small program and store it in a Motorola S-Record or Intel Hex file.

2. Use the Load Executable window to download the program into RAM or flash memory.

3. Use the Breakpoints window to set breakpoints. Use the Registers window to initialize register values.

   The new register or breakpoint values are sent to the processor when you press the Enter key or when you move the cursor out of the selected register field.

4. In the Run Control window, click Run.

5. Use the Memory Mnemonic window to view the program and use the Memory window to view any output which has been written to memory.
Designing a Target System for the Emulator
Designing a Target System for the Emulator

This chapter will help you design a target system that will work with the Agilent Technologies emulator.

Designing the JTAG connector

There are four different “standards” for ARM JTAG target system headers:

- ARM Embedded ICE 14-pin (0.100 inch pin spacing)
- VLSI Jumpstart 14-pin (0.100 inch pin spacing)
- TI XDS510 14-pin (0.100 inch pin spacing)
- Agilent/ARM 20-pin low density (0.100 inch pin spacing)

The 14-pin connectors are supported by the emulator, but should not be designed into new systems due to their susceptibility to noise and cross-talk. New designs should use the Agilent/ARM 20-pin connector described in this chapter.

![Diagram of JTAG connector pins](image_url)
Signal descriptions (20-pin connector)

**Required Signals**

**Vtref signal.** The Vtref signal is used to sense when the target system is connected to the emulator. This signal is also used as a reference voltage for all signals driven by the emulator. The emulator does not use this pin for power.

**TDI signal.** The TDI signal is a core JTAG signal and should be connected to TDI on the target.

**TDO signal.** The TDO signal is a core JTAG signal and should be connected to TDO on the target.

**TCK signal.** The TCK signal is a core JTAG signal.

**TMS signal.** The TMS signal is a core JTAG signal.

**Strongly recommended signals**

**DBGRQ signal.** The DBGRQ signal is used to quickly enter debug mode after receiving a "BREAK IN" signal from the logic analyzer. This allows the logic analyzer triggering capability to be used for complex breakpoints. Without DBGRQ, the processor must be stopped by using the JTAG scan chains, and the processor may not stop until several hundred instructions have been executed. This may cause complications when attempting to correlate the state of the logic analyzer and the emulator after the breakpoint has been reached. In general, DBGRQ is much more useful than DBGACK for system debug.

**DBGACK signal.** The DBGACK signal is used to quickly detect entry or exit from debug mode. If DBGACK is available, then the "TRIGGER OUT" signal from the emulator can be used to start or stop the logic analyzer. Without DBGACK, the emulator must do polling over the JTAG scan chains to determine when the processor is in debug mode. Without the use of DBGACK, there will be a substantial and variable delay between the processor stopping and "TRIGGER OUT" being driven. This may cause some difficulty in correlating the trace of the
logic analyzer with the exact time that the processor entered debug mode.

**Optional signals**

**VSupply signal.** The VSupply signal is used by some JTAG controllers as a power source. The emulator does not use this pin for power. This pin should be left unconnected if it is not available. If it is available, then the emulator can accept 0V - 5V at this pin.

**nTRST signal.** The nTRST signal is the JTAG reset signal. Most target systems implement a reset by holding TMS high for 5 consecutive clock cycles. nTRST can be used as a substitute for that operation.

**nSRST signal.** The nSRST signal is a current limited, open collector signal that is driven and monitored by the emulator. nSRST is connected to the ARM signal nRESET. If this signal is not available, then the emulator will be unable to reset the target system.

**Unused signals**

**RTCK signal.** The emulation probe does not use the RTCK signal.

---

**Electrical requirements for the TDO, TDI, TCK, TMS and nTRST signals**

TDO, TDI, TCK, TMS and nTRST signal traces between the JTAG connector and the ARM target must be less than 3 inches long. If these signals are connected to other nodes, the other nodes must be daisy chained between the JTAG connector at one end and the ARM7 microprocessor at the other end. These signals are sensitive to crosstalk and must not be routed along active signals such as clock lines on the target board.

The TDI, TCK, TMS and nTRST signals must not be actively driven by the target system when the debug port is being used.
Using the Emulator with a Debugger
Using the Emulator with a Debugger

Several prominent companies design and sell state-of-the-art source debuggers which work with the Agilent Technologies emulation module and emulation probe.

Benefits of using a debugger

The debugger will enable you to control the execution of your processor from the familiar environment of your debugger. Using a debugger lets you step through your code at the source-code level.

With a debugger connection, you can set breakpoints, single-step through source code, examine variables, and modify source code variables from the debugger interface. The debugger can also be used to download executable code to your target system.

Using a debugger to connect to the emulator allows the entire design team to have a consistent interface from software development to hardware/software integration.

Debugger interfaces must be ordered directly from the debugger vendor.

Compatibility with other logic analysis system tools

You can use your logic analysis system to collect and analyze trace data while you use your debugger. If you are using an X windows workstation or a PC with an X terminal emulator, you can display the logic analyzer windows right next to your debugger.
Using the Emulator with a Debugger

Minimum requirements
To use a debugger with the emulator, you will need:
- A debugger which is compatible with the emulator
- A LAN connection to the PC or workstation that is running the debugger
- X windows or an X terminal emulator, such as Reflection X on a PC. This is required only if you wish to have the logic analysis system user interface displayed on your PC or workstation screen, along with the debugger.

Is your debugger compatible with the emulator?
Ask your debugger vendor whether the debugger can be used with an Agilent Technologies emulation module or Agilent Technologies emulation probe (also known as a "processor probe" or "software probe").

LAN connection
You will use a LAN connection to allow the debugger to communicate with the emulator.

Compatibility with the Emulation Control Interface
Do not use the logic analysis system's Emulation Control Interface and your debugger at the same time.
Setting up Debugger Software

The instructions in this manual assume that your PC or workstation is already connected to the LAN, and that you have already installed the debugger software according to the debugger vendor’s documentation.

To use your debugger with the emulator, follow these general steps:

- Connect the emulator to your target system (see page 56).
- Connect the emulator or logic analysis system to the LAN (page 91).
- Export the logic analysis system's display to your PC or workstation (page 93).
- Configure the emulator (page 64).
- Begin using your debugger.

If you use the Emulation Control Interface to configure the emulator, remember to end the Emulation Control Interface session before you start the debugger.

**CAUTION:**

Do not use the Emulation Control Interface at the same time as a debugger. The Emulation Control Interface and debuggers do not keep track of commands issued by other tools. If you use both at the same time, the tools may display incorrect information about the state of the processor, possibly resulting in lost data.

**See Also**

Refer to the documentation for your debugger for more information on connecting the debugger to the emulator.
To connect the logic analysis system to the LAN

Information on setting up a LAN connection is provided in the online help or installation manual for your logic analysis system.

Your debugger will require some information about the LAN connection before it can connect to the emulator. This information may include:

- IP address (Internet address) or LAN name of the logic analysis system.
- Gateway address of the logic analysis system.
- Port number of the emulator.

**Port numbers for emulators**

<table>
<thead>
<tr>
<th>Port number</th>
<th>Use for</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Debugger connections</strong></td>
<td></td>
</tr>
<tr>
<td>6470</td>
<td>Slot 1 (First emulator in an Agilent Technologies 1660A/700A-series logic analysis system)</td>
</tr>
<tr>
<td>6474</td>
<td>Slot 2 (Second emulator in an Agilent Technologies 16700A-series system)</td>
</tr>
<tr>
<td>6478</td>
<td>Slot 3 (Third emulator in an expansion frame)</td>
</tr>
<tr>
<td>6482</td>
<td>Slot 4 (Fourth emulator in an expansion frame)</td>
</tr>
<tr>
<td><strong>Telnet connections</strong></td>
<td></td>
</tr>
<tr>
<td>6472</td>
<td>Slot 1 (First emulator)</td>
</tr>
<tr>
<td>6476</td>
<td>Slot 2 (Second emulator)</td>
</tr>
<tr>
<td>6480</td>
<td>Slot 3 (Third emulator)</td>
</tr>
<tr>
<td>6484</td>
<td>Slot 4 (Fourth emulator)</td>
</tr>
</tbody>
</table>

Write the information here for future reference:

- IP Address of Logic Analysis System: ___________________________
- LAN Name of Logic Analysis System: ___________________________
- Gateway Address: ___________________________
- Port Number of Emulator: ___________________________
Using the Emulator with a Debugger

Setting up Debugger Software

To change the port number of an emulator

Some debuggers do not provide a means to specify a port number. In that case, the debugger will always connect to port 6470 (the first emulator). If you need to connect to another module, or if the port number of the first module has been changed, you must change the port number to be 6470.

To view or change the port number:

1 Click on the emulation module icon in the system window of the logic analysis system, then select Update Firmware.

2 Select Modify Lan Port...

3 If necessary, enter the new port number in the Lan Port Address field.

The new port number must be greater than 1024 and must not already be assigned to another emulator.

To verify communication with the emulator

1 Telnet to the IP address.

   For example, on a UNIX system, enter “telnet <IP_address> 6472”. This connection will give you access to the emulator's built-in terminal interface. You should see a prompt, such as “M>”.

2 At the prompt, type:

   ver

   You should then see information about the emulator and firmware version.

3 To exit from this telnet session, type <CTRL>D at the prompt.

See Also

The online help or manual for your logic analysis system, for information on physically connecting the system to the LAN and configuring LAN parameters. If you have problems verifying LAN communication, see page 129.
To export the logic analysis system’s display to a workstation

By exporting the logic analyzer’s display, you can see and use the logic analysis system’s windows on the screen of your workstation. To do this, you must have telnet software and X window installed on your computer.

1. On the workstation, add the host name of the logic analysis system to the list of systems allowed to make connections:

   \texttt{xhost +<IP\_address>}

2. Use telnet to connect to the logic analysis system.

   \texttt{telnet <IP\_address>}

3. Log in as “hplogic”.

   The logic analysis system will open a Session Manager window on your display.

4. In the Session Manager window, click Start Session on This Display.

   \textbf{Example} 

   On a UNIX workstation, you could use the following commands to export the display of a logic analysis system named “mylogic”:

   \begin{verbatim}
   $ xhost +mylogic
   $ telnet mylogic
   Trying...
   Connected to mylogic.mycompany.com.
   Escape character is `^]’.
   Local flow control on
   Telnet TERMINAL-SPEED option ON
   
   Agilent Technologies Logic Analysis System
   
   Please Log in as: hplogic [displayname:0]
   
   login: hplogic
   Connection closed by foreign host.
   \end{verbatim}
Using the Emulator with a Debugger

Setting up Debugger Software

To export the logic analysis system's display to a PC

By exporting the logic analyzer's display, you can see and use the logic analysis system's windows on the screen of your PC. To do this, you must have telnet software and an X terminal emulator installed on your computer. The following instructions use the Reflection X emulator from WRQ, running on Windows 95, as an example.

1. On the PC, start the X terminal emulator software.
   
   To start Reflection X, click the Reflection X Client Startup icon.

2. Start a telnet connection to the logic analysis system.
   
   Log in as “hplogic”.
   
   For Reflection X, enter the following values in the Reflection X Client Startup dialog:
   
   a. In the Host field, enter the LAN name or IP address of the logic analysis system.
   
   b. In the User Name field, enter “hplogic”.
   
   c. Leave the Password field blank.
   
   d. Leave the Command field blank.
   
   e. Click Run to start the connection.
   
   The logic analysis system will open a Session Manager window on your display.

3. In the Session Manager window, click Start Session on This Display.
Using the ARM Debugger for Windows

Compatibility

The ARM Debugger for Windows (ADW) and ARM Developer Suite v1.0 or greater are two of the debuggers that connect to the Agilent Technologies emulator. ADW is part of the Software Development Toolkit (SDT), available from ARM Ltd. Only versions of the SDT greater than v2.11a support the use of an Agilent Technologies emulator.

This section provides information that is specific to using ADW with the emulator. It should be used in conjunction with the more detailed SDT user documentation provided by ARM. In the case of any conflict between these two sources, the SDT documentation should be considered definitive.

Overview
Using the Emulator with a Debugger

Using the ARM Debugger for Windows

Getting Started

1 Connect and set up the emulation probe or logic analysis system as described in the Agilent Technologies documentation.

Ensure that the file ‘gateway.dll’ is placed in the ‘bin’ sub-directory of the ADW installation directory. Depending on the intended target system, one or more Gateway Configuration ‘.gcf’ files should also be placed in this same directory. The relationship between target and configuration file is given in the following table:

<table>
<thead>
<tr>
<th>Target</th>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARM 7TDMD</td>
<td>gate_arm7tdmi.gcf</td>
</tr>
<tr>
<td>ARM 710T</td>
<td>gate_arm710t.gcf</td>
</tr>
<tr>
<td>ARM 720T</td>
<td>gate_arm720t.gcf</td>
</tr>
<tr>
<td>ARM 740T</td>
<td>gate_arm740t.gcf</td>
</tr>
<tr>
<td>ARM 920T</td>
<td>gate_arm920t.gcf</td>
</tr>
<tr>
<td>ARM 940T</td>
<td>gate_arm940t.gcf</td>
</tr>
<tr>
<td>ARM 966ES</td>
<td>gate_arm966es.gcf</td>
</tr>
</tbody>
</table>

The list given in this table may not be exhaustive.

2 Configure the ARM Debugger for Windows.

The ARM Debugger for Windows must be configured prior to use with the emulator.

a Start ADW and allow it to initialize normally.

If no errors occur, choose Options->Configure Debugger from the ADW menu to display the configuration dialog for ADW.

If an initialization error does occur, then choose Cancel and enter Armulator mode (SDT versions prior to v2.50) then choose Options->Configure Debugger again, or simply click on the Configure button when this is presented as an option (SDT v2.50 and higher).
The configuration dialog actually consists of a number of tabbed pages, a particular page being chosen by clicking on the corresponding tab. Select the **Target** tab of the dialog.

b **Select gateway.**

If gateway has been used previously, then click on the target environment combo box to display the list of known environments and ensure that gateway is selected.

If a version of gateway has not been installed previously, it must be added to the list of target environments supported by ADW. Do so by clicking on the **Add** button on the dialog and then opening the gateway.dll file. This automatically makes gateway the selected environment.

c **Configure gateway.**

Click on the **Configure** button to display the configuration parameters specific to gateway. These include the currently used network location of the emulation probe, the target processor type, and a list of channel viewers. Selecting a channel viewer involves checking the Enabled checkbox and highlighting the particular viewer in the list. There are also Add and Remove buttons to enable the list to be manipulated. Please see the relevant ARM documentation for details of channel viewers and their usage.

Click on the **Set** button to alter the network location and target type parameters. A further dialog is then displayed where the new address can be entered and a target selected from those supported by the emulation probe. Note that there may be a short delay before display of this dialog as a connection is made to the emulation probe.

Click **OK** to dismiss the gateway configuration dialog. Change parameters listed on other pages of the configuration dialog as appropriate. In particular, make sure that the target endianness is set up correctly on the Debugger page. Finally click **OK** again to dismiss the ADW configuration dialog. ADW should restart without errors using the new configuration.
To Perform Common Debugger Tasks

To download an image, choose File->Load Image from the ADW command menu. The Load Image file dialog is displayed and can be used to locate and select the image file. During a download the number of bytes written and read is displayed as a progress indicator. Once the operation is complete, the execution window of ADW will show the start of the disassembled code. If the image contains debug information and has been compiled from 'C' sources a breakpoint is automatically inserted by ADW at the start of the main() procedure.

To start execution of a downloaded image, choose Execute->Go from the ADW menu. Note that the initial value of the PC will have been set appropriately as a result of the download.

To stop the target, choose Execute->Stop from the ADW menu. A subsequent Execute->Go will start the target running from the current value of the PC.

To set or clear a breakpoint, select the line in the execution window and choose Execute->Toggle Breakpoint. If no breakpoint exists at this location it is added, while an existing breakpoint will be cleared. This can only be done while the target is stopped.

To set a watchpoint, select View->Variables and choose either local or global. This displays respectively the variables that are local to the current procedure or global within the application. Selecting the value of interest and choosing Execute->Toggle Watchpoint sets the watchpoint.

To display or modify registers, choose View->Registers and select the desired register bank (that is, the set of registers for a particular ARM processor mode). All of the registers in that bank are then displayed in a separate window. To change the value of a register, double-click on the displayed value in this window. A further dialog box is then displayed from which a new value for that register can be set. These operations can only be carried out when the target is stopped.

To step through code, choose Execute->Step. Various other stepping options are also available from the execute menu such as Step In, Step Out and Run to Cursor.
Note that the above operations can be carried out in a variety of other ways, for example shortcut keys or toolbar buttons. See ARM documentation for further details.

**Error Conditions**

The emulator can detect and report to ADW that the target power is switched off. However, if the emulator is not connected or powered down, ADW will appear to hang for approximately 1 minute during start-up. This is simply a network time-out. ADW will then report that it could not initialize the communications link. Under these circumstances, check all connections and ensure that the address of the emulator has been correctly specified. The behavior is similar if the TIM has been disconnected from the emulator. In this case, the error reported will be failure to enter monitor mode.

Starting up ADW after the target has been power cycled may result in repeated warnings that the probe cannot read memory. This is not a problem and occurs as a result of the debugger attempting to access memory around a non-valid value in the pc register. The debugger's execution window may be all or partially blank as a result of these errors.

If certain sensitive operations, such as downloading memory, are interrupted, subsequent attempts to start up ADW may result in an error message stating that the probe could not enter monitor mode. Under these circumstances it may be necessary to cycle power to the probe and target.

**See Also**

The ARM Web Site: [http://www.arm.com](http://www.arm.com)

The *Software Development Toolkit User Guide*. 
Using the Emulator with a Debugger

Using the ARM Debugger for Windows
Specifications and Characteristics
Specifications and Characteristics

---

**Processor compatibility**

The Agilent Technologies emulation probe supports the following processors:

- ARM7TDMI
- ARM7DI
- ARM710T
- ARM720T
- ARM740T
- ARM920T
- ARM940T
- ARM966ES

The emulation probe supports a target voltage reference from 1.2V to 5V.
### Emulation probe and emulation module—electrical characteristics

**Specifications and Characteristics**

#### Maximum Ratings

<table>
<thead>
<tr>
<th>Characteristics for Agilent Technologies Emulation Probe</th>
<th>Symbol</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDO, DBGACK</td>
<td>$V_{ih}$</td>
<td>$0.8 \times V_{cc}$</td>
<td>$0.2 \times V_{cc}$</td>
</tr>
<tr>
<td></td>
<td>$V_{il}$</td>
<td>$0.2 \times V_{cc}$</td>
<td>$\pm 1 \mu A$</td>
</tr>
<tr>
<td></td>
<td>$I_i$</td>
<td>$15 , \mu A$</td>
<td>$15 , \mu A$</td>
</tr>
<tr>
<td></td>
<td>$C_{in}$</td>
<td>$15 , \mu A$</td>
<td>$15 , \mu A$</td>
</tr>
<tr>
<td>TDI, TCK, TMS, TRST, DBGQR</td>
<td>$V_{oh} @ I_{oh} = -32 , mA$</td>
<td>$0.8 \times V_{cc}$</td>
<td>$0.2 \times V_{cc}$</td>
</tr>
<tr>
<td></td>
<td>$V_{oh} @ I_{oh} = 64 , mA$; $V_{cc} = 4.5 , V$</td>
<td>$0.8 \times V_{cc}$</td>
<td>$0.2 \times V_{cc}$</td>
</tr>
<tr>
<td>TDI, TMS, TRST, DBGQR</td>
<td>$C_o$</td>
<td>$25 , pF$</td>
<td>$25 , pF$</td>
</tr>
<tr>
<td>TCK</td>
<td>$C_o$</td>
<td>$45 , pF$</td>
<td>$45 , pF$</td>
</tr>
<tr>
<td>Vref&lt;sup&gt;1&lt;/sup&gt;</td>
<td>$V_{ih}$</td>
<td>$1.2 , V$</td>
<td>$5 , V$</td>
</tr>
<tr>
<td></td>
<td>$V_{il}$</td>
<td>$0.8 , V$</td>
<td>$0.8 , V$</td>
</tr>
<tr>
<td>SRST&lt;sup&gt;2&lt;/sup&gt;</td>
<td>$V_{oh} @ 24 , mA$</td>
<td>$0.5 , V$</td>
<td>$25 , pF$</td>
</tr>
<tr>
<td></td>
<td>$C_o$</td>
<td>$25 , pF$</td>
<td>$25 , pF$</td>
</tr>
</tbody>
</table>

<sup>1</sup> Vref is used only to determine target powered status. The emulation probe does not draw power from this source.

<sup>2</sup> Open collector output, pulled up to a generated voltage equivalent to the Vref voltage with a 2.61 K pullup resistor.
Specifications and Characteristics

Emulation probe—electrical characteristics

BNC, labeled TRIGGER OUT

Output Drive. 2.0 V. Logic low level with 50-ohm load <= 0.4 V. Output function is selectable, see the configuration chapter (page 63).

BNC, labeled BREAK IN

Input. Edge-triggered TTL level input, 20 pF, with 2k ohms to ground in parallel. Maximum input: 5V above VCC; 5 V below ground. Input function is selectable, see the configuration chapter (page 63).

Communications

Serial Port. 9-pin female type D subminiature connector. RS-232 DCE to 115.2 kbaud.

10BASE-T LAN Port. RJ-45 connector. IEEE 802.3 10BASE-T (StarLAN).

10BASE 2 LAN Port. 50-ohm BNC connector. IEEE 802.3 10BASE2 (ThinLAN). When using this connector, the Agilent Technologies emulator provides the functional equivalent of a Medium Attachment Unit (MAU) for ThinLAN.

Accessory Power Out

12 V, 3.0A, center negative

Power Supply

Input. 100-240 V, 9.75 A, 50/60 Hz, IEC 320 connector.

Output. 12 V, 3.3 A

CAT I (Mains isolated)

Pollution degree 2
Specifications and Characteristics

Emulation probe—environmental characteristics

**Temperature**
Operating, +5 C to +40 C (+41 to +104 F); nonoperating, -40 to +70 C (-40 to +158 F).

**Altitude**
Operating/nonoperating 4600 m (15 000 ft).

**Relative Humidity**
15% to 95%.

Emulation module—environmental characteristics

The Agilent Technologies 16610A emulation module meets the environmental characteristics of the logic analysis system in which it is installed.

For indoor use only.
Specifications and Characteristics
Updating Firmware

Firmware gives your emulator a “personality” for a particular processor or processor family.

After you have connected the emulator to your target system, you may need to update the firmware to give it the right personality for your processor.

You must update the firmware if:

- You have an emulation module which was not shipped already installed in the logic analysis system.
- You need to change the personality of the emulator for a new processor.
- You have an updated version of the firmware from Agilent Technologies.

The procedure for updating firmware for an emulation probe is different from the procedure for updating firmware for an emulation module.
Emulation Probe Firmware

To display current firmware version information

- Use telnet or a terminal emulator to access the built-in "terminal interface" and use the ver command to view the version information for firmware currently in the emulation probe.

To update firmware for an emulation probe

To update the firmware, you must have access to the World Wide Web and a PC or a workstation connected to your emulation probe.

1. Download the new firmware from the following World Wide Web site:

   http://www.agilent.com/find/emulator

   The firmware will be in the “Technical Support Information” section of this web site.

2. Follow the instructions on the web site for installing the firmware.

   If Agilent Technologies sends you firmware on a floppy disk, install the firmware from the floppy disk. The README file on the floppy disk contains instructions for installing the firmware using a PC or workstation.
Updating Firmware

Emulation Module Firmware

Always update firmware by installing a processor support package. This will ensure that the version of the Emulation Control Interface software is compatible with the version of the emulator firmware.

To display current firmware version information

1. In the Update Firmware window, click **Display Current Version**.

   There are usually two firmware version numbers: one for “Generics” and one for the personality of your processor.

To update firmware for an emulation module using the Emulation Control Interface

1. End any run control sessions which may be running.
2. In the Workspace window, remove any Emulator icons from the workspace.
3. Install the processor support package from the CD-ROM, if necessary.
   - In the system window, click the emulation module and select **Update Firmware**.
4. **Update Firmware**.

   ![Update Firmware Window]

5. In the Update Firmware window, select the firmware to load into the emulation module.
6 Click **Update Firmware**.

In about 20 seconds, the firmware will be installed and the screen will update to show the current firmware version.

**See also**

“Installing Software” beginning on page 52 for instructions on how to install the processor support package from the CD-ROM.

---

**To update firmware for an emulation module using the Setup Assistant**

The Setup Assistant is an online tool for connecting and configuring your logic analysis system for microprocessor and bus analysis. The Setup Assistant is available on the Agilent Technologies 16600A and 16700A-series logic analysis systems.

This menu-driven tool will guide you through the connection procedures for connecting the logic analyzer to an analysis probe, an emulation module, or other supported equipment. It will also guide you through connecting an analysis probe to the target system.

1 Install the processor support package from the CD-ROM.

2 Start the Setup Assistant by clicking its icon in the system window.

3 Follow the instructions displayed by the Setup Assistant.

**See also**

Page 53 for instructions on how to install a the processor support package from the CD-ROM.
Updating Firmware

Emulation Module Firmware
Solving Problems
If you have problems with the emulator, your first task is to determine the source of the problem. Problems may originate in any of the following places:

- The connection between the emulator and your workstation, PC, or logic analyzer
- The emulator itself
- The connection between the emulator and the target interface module
- The connection between the target interface module and the target system
- The target system

You can use several means to determine the source of the problem:

- The troubleshooting guide on the next page
- The status lights on the emulator
- The emulator "performance verification" tests
- The emulator's built-in "terminal interface" commands
Troubleshooting Guide

For troubleshooting the emulation module, refer to the online help in the logic analysis system.

**Common problems and what to do about them (Emulation probe)**

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Action</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host computer reports LAN connection problems</td>
<td>Follow the checklist in the “If you have LAN problems” section.</td>
<td>129</td>
</tr>
</tbody>
</table>
| Commands from the Emulation Control Interface or debugger have no effect | 1 Verify LAN communication.  
2 Check that you are using the correct firmware for your chip.  
3 Use a telnet connection (or serial connection) to try a few built-in commands. If this works, your debugger may not be configured properly. If this does not work, continue with the next procedure... | 38, 110, 118 |
| Emulator built-in commands do not work       | 1 Check that the emulator has been properly configured for your target system.  
2 Run the emulator performance verification tests.  
3 If the performance verification tests pass, then there is an electrical problem with the connection to the target processor OR the target system may not have been designed according to “Designing a Target System.” | 63, 137, 83 |
| “Slow or missing clock” message after a logic analyzer run | Check that the target system is running user code or is in reset. (This message can appear if the processor is in background mode.) |      |
| “Slow clock” message in the Emulation Control Interface or “c>” prompt in the built-in “terminal interface” | Check that the clock rate is properly configured. | 72   |
| Some commands fail                           | Check the “restrict to real-time runs” configuration                    | 71   |
The emulation probe communicates various modes and error conditions via the status lights.

The following table gives information about the meaning of the power and target status lights.

○ = LED is off
● = LED is on
✻ = Not applicable (LED is off or on)

### Power/Target Status Lights

<table>
<thead>
<tr>
<th>Pwr/Target LEDs</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>○○</td>
<td>Emulation probe is not connected to power supply</td>
</tr>
<tr>
<td>○○</td>
<td></td>
</tr>
<tr>
<td>○●</td>
<td>No target system power, or emulator is not connected to the target system</td>
</tr>
<tr>
<td>○●</td>
<td></td>
</tr>
<tr>
<td>●●</td>
<td>Target system is in a reset state</td>
</tr>
<tr>
<td>●●</td>
<td></td>
</tr>
<tr>
<td>●●</td>
<td>Only boot firmware is good (other firmware has been corrupted)</td>
</tr>
<tr>
<td>●●</td>
<td></td>
</tr>
<tr>
<td>●●</td>
<td>The target processor is in debug mode</td>
</tr>
<tr>
<td>●●</td>
<td></td>
</tr>
<tr>
<td>●●</td>
<td>The target processor is executing user code</td>
</tr>
<tr>
<td>●●</td>
<td></td>
</tr>
<tr>
<td>●●</td>
<td>The target processor is in an unknown state</td>
</tr>
</tbody>
</table>
Emulation probe status lights

The following illustration shows the status lights on both sides of the emulation probe and what they mean:

- **Lit when the power supply is properly connected to the emulation probe**
- **Lit when the target processor is running in debug mode**
- **Lit when the target system is in a reset state**
- **Lit when the target processor is running in normal (user program) mode**
- **Lit when LAN data is being transmitted**
- **Lit when 10BASE-T connection has a good link; not used for 10BASE2**
- **Lit when the polarity on the receive twisted pair is reversed for a 10BASE-T connection**
- **Lit when LAN data is being received**

**Status Lights**
Solving Problems

Emulation Module Status Lights

The emulation module communicates various modes and error conditions via the status lights.

The following table gives information about the meaning of the power and target status lights.

❖ = LED is off
● = LED is on
✻ = Not applicable (LED is off or on)

### Power/Target Status Lights

<table>
<thead>
<tr>
<th>Pwr/Target LEDs</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>❖ Reset ❖ Break ❖ Run</td>
<td>No target system power, or emulation module is not connected to the target system</td>
</tr>
<tr>
<td>● Reset ❖ Break ❖ Run</td>
<td>Target system is in a reset state</td>
</tr>
<tr>
<td>❖ Reset ● Break ❖ Run</td>
<td>The target processor is executing in Debug Mode</td>
</tr>
<tr>
<td>❖ Reset ❖ Break ● Run</td>
<td>The target processor is executing user code</td>
</tr>
<tr>
<td>❖ Reset ● Break ● Run</td>
<td>Only boot firmware is good (other firmware has been corrupted)</td>
</tr>
</tbody>
</table>

Built-in Commands

The emulator has some built-in "terminal interface" commands which you can use for troubleshooting.
Solving Problems

You can access the terminal interface via:

- A serial connection (Emulation probe only. See page 40.)
- A telnet (LAN) connection
- The Command Line window in the Emulation Control Interface
- A "debugger command" window in your debugger

Here are a few useful commands:

**Useful built-in commands**

<table>
<thead>
<tr>
<th>Commands</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>Break—go into debug mode</td>
</tr>
<tr>
<td>cf</td>
<td>Configuration—read or write configuration options</td>
</tr>
<tr>
<td>help</td>
<td>Help—display online help for built-in commands</td>
</tr>
<tr>
<td>init</td>
<td>Initialize—init-c reinitializes everything in the emulator except for the LAN software; init-p is the equivalent of cycling power (it will break LAN connections)</td>
</tr>
<tr>
<td>lan</td>
<td>configure LAN address</td>
</tr>
<tr>
<td>m</td>
<td>Memory—read or write memory</td>
</tr>
<tr>
<td>reg</td>
<td>Register—read or write register</td>
</tr>
<tr>
<td>r</td>
<td>Run—start running user code</td>
</tr>
<tr>
<td>rep</td>
<td>Repeat—repeat a command or group of commands</td>
</tr>
<tr>
<td>rst</td>
<td>Reset—reset the target processor</td>
</tr>
<tr>
<td>s</td>
<td>Step—do a low-level single step</td>
</tr>
<tr>
<td>ver</td>
<td>Version—display the product number and firmware version of the emulator</td>
</tr>
</tbody>
</table>
The prompt indicates the status of the emulator:

**Emulator prompts**

<table>
<thead>
<tr>
<th>Prompts</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>Emulation reset</td>
</tr>
<tr>
<td>U</td>
<td>Running user program</td>
</tr>
<tr>
<td>M</td>
<td>Running in background monitor</td>
</tr>
<tr>
<td>r</td>
<td>Target system reset active</td>
</tr>
<tr>
<td>p</td>
<td>No target power</td>
</tr>
<tr>
<td>?</td>
<td>Unknown state</td>
</tr>
</tbody>
</table>

**Examples**

To set register R0, then view R0 to verify that it was set, enter:

```
R>rst -m
M>reg r0=ffff
M>reg r0
   reg R0=0000ffff
```

To break execution then step a single instruction, enter:

```
M>b
M>s
   PC=xxxxxxxx
M>
```

To determine what firmware version is installed in the emulation module, enter:

```
M>ver
```

**See Also**

Use the `help` command for more information on these and other commands. Note that some of commands listed in the help screens may not be available.

If you are writing your own debugger, contact Agilent Technologies for more information.
Problems with the Target System

What to check first

1. Try some basic built-in commands (using a serial or telnet connection):

   U> **rst**
   R>
   This should reset the target and display an "R>" prompt.

   R>b
   M>
   This should stop the target and display an "M>" prompt.

   M> **reg r1**
   reg r1=007fd6a0
   M>
Solving Problems

Problems with the Target System

This should read the value of the r1 register (the value will probably be different on your target system).

\[ \text{M} \text{> m 0...} \]

<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000</td>
<td>7c3043a6</td>
</tr>
<tr>
<td>00000010</td>
<td>00000000</td>
</tr>
<tr>
<td>00000020</td>
<td>00000000</td>
</tr>
<tr>
<td>00000030</td>
<td>00000000</td>
</tr>
<tr>
<td>00000040</td>
<td>00000000</td>
</tr>
<tr>
<td>00000050</td>
<td>00000000</td>
</tr>
<tr>
<td>00000060</td>
<td>00000000</td>
</tr>
<tr>
<td>00000070</td>
<td>00000000</td>
</tr>
</tbody>
</table>

\[ \text{M} \text{> s} \]

This should display memory values starting at address 0.

\[ \text{M} \text{> s} \]

This should execute one instruction at the current program counter.

**Note that stepping can fail if memory at the current PC does not contain a valid instruction.**

If any of these commands don’t work, there may be a problem with the design of your target system, a problem with the revision of the processor you are using, or a problem with the configuration of the emulator. The following steps will help you identify the problem.

2. Check that the emulator firmware matches your processor. To do this, enter:

\[ \text{M} \text{> ver} \]

**See Also**  
Page 118 for information on entering built-in commands.
To check the debug port connector signals

- Check for the following logic levels on the target debug port. The signal names are for the ARM7.

**Signal Levels with the emulation probe in default\(^1\) reset state**

<table>
<thead>
<tr>
<th>Signal Name</th>
<th>ARM 14-pin</th>
<th>VLSI 14-pin</th>
<th>TI 14-pin</th>
<th>ARM/Agilent pin</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vcc/Vref/SPU</td>
<td>1,13</td>
<td>13</td>
<td>5</td>
<td>1</td>
<td>1.2V - 5.0V(^2)</td>
</tr>
<tr>
<td>TDI</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>5</td>
<td>Toggling</td>
</tr>
<tr>
<td>TDO</td>
<td>11</td>
<td>11</td>
<td>7</td>
<td>13</td>
<td>Toggling</td>
</tr>
<tr>
<td>TCK</td>
<td>9</td>
<td>9</td>
<td>11</td>
<td>9</td>
<td>~10MHz signal</td>
</tr>
<tr>
<td>TMS</td>
<td>7</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>Toggling</td>
</tr>
<tr>
<td>nSYSRST</td>
<td>12</td>
<td>12</td>
<td>N/A</td>
<td>15</td>
<td>Low</td>
</tr>
<tr>
<td>DBGACK</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>19</td>
<td>Low</td>
</tr>
<tr>
<td>DBGREQ</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>17</td>
<td>Low</td>
</tr>
<tr>
<td>nTRST</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>High</td>
</tr>
<tr>
<td>PT</td>
<td>N/A</td>
<td>10</td>
<td>N/A</td>
<td>N/A</td>
<td>High</td>
</tr>
<tr>
<td>PN[3:0]</td>
<td>N/A</td>
<td>4,6,8</td>
<td>N/A</td>
<td>N/A</td>
<td>Low</td>
</tr>
</tbody>
</table>

All other pins should be low.
\(^1\)Default cf settings
\(^2\)Target dependent
To interpret the initial prompt

The initial prompt can be used to diagnose several common problems.

1. Connect the emulator to your target system.
2. Set the default configuration settings. Enter:

   `M>init -c`

   You can enter this command at any prompt. The emulator will respond with the same information as printed by the "ver" command.

   **If the response is "!ERROR 905! Driver firmware is incompatible with ID of attached device"

   Make sure the target interface module is connected to the cable of the emulator, then try the "init -c" command again.

   **If the initial prompt is "p>"

   Check pin 1 on header (if using the TI connector check pin 5 instead) 1.2V - 5V (VDD).

   **If the initial prompt is "M>"

   The processor entered debug mode without the help of the emulator. Is another debugger connected?

   **If the initial prompt is "?>"

   Check TCK, TDO, TDI, TMS, and TRST_L signals. Check the firmware revision.
If the initial prompt is "R>

The emulator is scanning the JTAG correctly. Now you can do some more tests:

3 Enter the reset command:

```
R> rst
R>
```

The "R>" prompt is a good response that indicates that nRESET is working. Continue with "If the prompt after rst is R>

4 Enter the break command:

```
R>b
M>
```

If the prompt after b is "U>" with error messages

If you see: "$ERROR 608! Unable to break", try resetting the target and/or cycling power on the target.

If the problem still persists, then try the sequence:

```
U>rst
R>r
M>b
```

Some ARM chips have problems if JTAG commands are issued while the chip is reset. Using the “rst”, “r”, “b” sequence avoids this problem.

If the prompt after b is "M>" with no error messages

Everything is still working correctly. Continue with more tests:
Solving Problems
Problems with the Target System

If you can get to the "M>" prompt

5 At the "M>" prompt, check register and memory access:

\begin{verbatim}
M>reg r0
  reg r0=xxxxxxxx
M>reg r0=12345678
M>reg r0
  reg r0=12345678
M>
\end{verbatim}

If the returned value is equal to the written value, then the voltage level of the chip is probably correct.

Now enter:

\begin{verbatim}
M>m -d4 -a4 0=11111111,22222222,33333333,44444444
M>m -d4 -a4 0..
  00000000   11111111 22222222 33333333 44444444
  00000010   00000000 00000000 00000000 00000000
  00000020   00000000 00000000 00000000 00000000
  00000030   00000000 00000000 00000000 00000000
  00000040   00000000 00000000 00000000 00000000
  00000050   00000000 00000000 00000000 00000000
  00000060   00000000 00000000 00000000 00000000
  00000070   00000000 00000000 00000000 00000000
M>
\end{verbatim}

- Returned value is equal to the written value implies that memory is working.
- Returned value is not equal to the written value implies that the memory controller may not be initialized. Be sure the memory controller for your target is setup correctly.
If you see memory-related problems

1. Set caches and translation off. Refer to your target documentation for doing this.

2. Now enter:

```
M> m -d4 -a4 0=11111111,22222222,33333333,44444444
M> m -d4 -a4 0..
00000000 11111111 02222222 33333333 44444444
00000010 00000000 00000000 00000000 00000000
00000020 00000000 00000000 00000000 00000000
00000030 00000000 00000000 00000000 00000000
00000040 00000000 00000000 00000000 00000000
00000050 00000000 00000000 00000000 00000000
00000060 00000000 00000000 00000000 00000000
00000070 00000000 00000000 00000000 00000000
```

- Read value not equal to the written value implies that the memory controller is not setup correctly.
- Read value equal to the written value implies that the emulator firmware might not be working with cache.

3. Hand load a little program:

```
M> m -d4 -a4 100=e2811001,00000000,60000000,eafffffc
M> reg r1=0
M>
This means: Add r1, r1, #1, NOP, b 100

Set the PC to this program:

```
M> reg pc=100
M>
Step, then check the register:
```

```
M> s
  pc=00000104
M> reg r1
  reg r1=0000 0001
M>
This should return"reg r1=00000001".
```
Solving Problems

Problems with the Target System

Step some more and verify that r1 increments after every three steps:

M> s 3
   pc=00000104
M> reg r1
   reg r1=00000002
M>

If this does not work as described, make sure "cf endian" is set correctly for your target, the memory at 100 is read/write memory, and that the memory controller is programmed correctly.
Problems with the LAN Interface (Probe Only)

NOTE:
For LAN interface problems with the emulation module, refer to the online help in the logic analysis system.

If you cannot verify LAN communication

If you cannot verify connection using the procedure in "To verify LAN communication", or if the commands are not accepted by the emulation probe:

- Make sure that you have connected the emulation probe to the proper power source and that the power light is lit.
- Make sure that you wait for the power-on self test to complete before connecting.
- Make sure that the LAN cable is connected. Watch the LAN LED's to see whether the emulation probe is seeing LAN activity. Refer to your LAN documentation for testing connectivity.
- Make sure that only one of the LAN ports is connected.
- Make sure the emulation probe communication configuration switches are set correctly. Unplug the emulation probe power cord, then plug it in again to make sure the switch settings are read correctly by the emulation probe.
- Check that the Emulation Control Interface or debugger was configured with the correct LAN address. If the emulation probe is on a different subnet than the host computer, check that the gateway address is correct.
- Make sure that the emulation probe's IP address is set up correctly. Use the RS-232 port to verify this that the IP address is set up correctly. When you are connected to the RS-232 port, run performance verification on the emulation probe's LAN interface with the "pv" command.
- It's also possible for there to be a problem with the emulation probe firmware while the LAN interface is still up and running. In this case, you must reboot the emulation probe by disconnecting power to the emulation probe and reconnecting it again.
Solving Problems

Problems with the LAN Interface (Probe Only)

- Use a serial connection to run the LAN performance verification tests (see page 137).

---

If you have LAN connection problems

- If the emulation probe does not accept commands from the Agilent Technologies 16600A/700A:
  1. Check that switch S1 is "0" (attached to LAN, not RS-232).
  2. Check that switch S5 is in the correct position for your LAN interface (either 10BASE2 or 10BASE-T).
     (Remember: if you change any switch settings, the changes do not take effect until you cycle power.)

- If the emulation probe still does not respond, you need to verify the IP address and gateway mask of the emulation probe. To do this, connect the emulation probe to a terminal or terminal emulator (see page 40), change the switch settings so it is connected to RS-232, and enter the "lan" command. The output looks something like this:

```
lan -i 15.5.24.116
lan -g 15.5.23.1
lan -p 6470
Ethernet Address : 08000909BAC1
```

"lan -i" shows the internet address is 15.5.24.116 in this case. If the Internet address (IP) is not what you expect, you can change it with the 'lan -i >' command.

"lan -g" shows the gateway address. Make sure it is the address of your gateway if you are connecting from another subnet, or 0.0.0.0 if you are connecting from the local subnet.

"lan -p" shows the port is 6470.

---

If the "POL" LED is lit

The "POL" LED indicates that the polarity is reversed on the receive pair if you are using a 10BASE-T connection. The emulator should still work properly in this situation, but other LAN devices may not work.
If it takes a long time to connect to the network

- Check the subnet masks on the other LAN devices connected to your network. All of the devices should be configured to use the same subnet mask.

Subnet mask error messages do not indicate a major problem. You can continue using the emulation probe.

The emulation probe automatically sets its subnet mask based on the first subnet mask it detects on the network. If it then detects other subnet masks, it will generate error messages.

If there are many subnet masks in use on the local subnet, the emulation probe may take a very long time to connect to the network after it is turned on.

To "clean up" the network, connect a terminal to the emulation probe. You can then see error messages which will help you identify which devices on the network are using the wrong subnet masks.
Problems with the Serial Interface (Emulation probe only)

If you cannot verify RS-232 communication

If the emulation probe prompt does not appear in the terminal emulator window:

- Make sure that you have connected the emulation probe to the proper power source and that the power light is lit.
- Make sure that you have properly configured the data communications switches on the emulation probe and the data communications parameters on the host computer. You should also verify that you are using the correct cable.

The most common type of data communications configuration problem involves the configuration of the emulator as a DTE device instead of as a DCE device. If you are using the wrong type of cable, no prompt will be displayed.

A cable with one-to-one connections will work with a PC or an HP 9000 Series 700 workstation.
If you have RS-232 connection problems with the MS Windows Terminal program

- Remember that Windows 3.1 only allows two active RS-232 connections at a time. To be warned when you violate this restriction, choose Always Warn in the Device Contention group box under 386 Enhanced in the Control Panel.

- Use the "Terminal" program (usually found in the Accessories windows program group) and set up the "Communications..." settings as follows:
  
  Baud Rate: 9600 (or whatever you have chosen for the emulation probe)
  Data Bits: 8
  Parity: None
  Flow Control: hardware
  Stop Bits: 1

  When you are connected, hit the Enter key. You should get a prompt back. If nothing echos back, check the switch settings on the emulation probe.

- If the switches are in the correct position and you still do not get a prompt when you hit return, try turning OFF the power to the emulation probe and turning it ON again.

- If you still don't get a prompt, make sure the RS-232 cable is connected to the correct port on your PC, and that the cable is appropriate for connecting the PC to a DCE device.

With certain RS-232 cards, connecting to an RS-232 port where the emulation probe is turned OFF (or is not connected) will hang the PC. The only way to get control back is to reboot the PC. Therefore, we recommend that you always turn ON the emulation probe before attempting to connect via RS-232.
Problems with the Emulation Probe Itself

To run the power up self test

1. Unplug the emulation probe, then plug it in.

2. Watch the status lights. They should show the following pattern:

   ❍ = LED is off
   ● = LED is on
   ✻ = Not applicable (LED is off or on)

Normal sequence during power up self test

<table>
<thead>
<tr>
<th>Pwr/Target LEDs</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>○● ○● Initial power up, system reset</td>
</tr>
<tr>
<td>2</td>
<td>○● ○● XILINX array initialized successfully</td>
</tr>
<tr>
<td>3</td>
<td>○● ●● XILINX array tested successfully</td>
</tr>
<tr>
<td>4</td>
<td>●● ○● BOOT ROM space tested successfully</td>
</tr>
<tr>
<td>5</td>
<td>○● ●● GENERIC ROM space tested successfully</td>
</tr>
<tr>
<td>6</td>
<td>●● ○● DRIVER ROM space tested successfully</td>
</tr>
<tr>
<td>7</td>
<td>○● ●● RESERVED ROM space tested successfully</td>
</tr>
<tr>
<td>8</td>
<td>●● ○● RAM tested successfully</td>
</tr>
<tr>
<td>9</td>
<td>○● ●● LAN internal feedback tested successfully</td>
</tr>
<tr>
<td>10</td>
<td>○● ○● Boundary scan master (BSM) test begun</td>
</tr>
<tr>
<td>11</td>
<td>●● ○● BMS test completed, start system, load drivers, initialize LAN</td>
</tr>
</tbody>
</table>
Solving Problems

Problems with the Emulation Probe Itself

If the power up self test fails, the RESET LED will flash the number of the test, then stay lit.

If any of the LEDs fail to change, or all of them remain on, there is a system failure.
Solving Problems

Problems with the Emulation Probe Itself

Following power up, the LEDs will enter one of the following states:

<table>
<thead>
<tr>
<th>Pwr/Target LEDs</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>○●</td>
<td>No target system power, or the emulation probe is not connected to the target system, or remove</td>
</tr>
<tr>
<td>○○</td>
<td>Target system is in a reset state</td>
</tr>
<tr>
<td>○●</td>
<td>Target processor is executing user code</td>
</tr>
<tr>
<td>●●</td>
<td>Target processor is in an unknown state</td>
</tr>
<tr>
<td>●●</td>
<td>Target processor is in debug mode</td>
</tr>
<tr>
<td>●●</td>
<td>Only the boot ROM was used; other firmware in the Flash EPROM has been corrupted</td>
</tr>
</tbody>
</table>

Starting a user interface will change the pattern to the one requested by the interface.

If the power up self tests fail, try the following:

- Check and reset the LAN address. LAN powerup failures will occur if the emulation probe does not have a valid Link Level Address and IP Address.
- Disconnect all external connections, including the LAN, serial (RS-232), and BNC Break and Trigger cables, then cycle power.
- To ensure that the firmware is working as it should, reprogram the firmware, then cycle power.
Solving Problems

Problems with the Emulation Probe Itself

To execute the built-in performance verification tests

In addition to the powerup tests, there are several additional performance verification (PV) tests available.

Some of these tests can be performed through the 16700A/700A. The LAN tests for an emulation probe can only be executed through the RS-232 port.

To fully test the emulation probe, you will need to run the PV test with several hardware configurations:

- For the BREAK IN, TRIGGER OUT BNC FEEDBACK TEST, connect a coaxial cable between BREAK IN and TRIGGER OUT.

To run the built-in performance verification test using the logic analysis system

1 End any Emulation Control Interface or debugger sessions.

2 Disconnect the 50-pin cable from the emulation module, and plug the loopback test board (Agilent part number E3496-66502) into the emulation module.

3 In the system window, click the emulation module and select Performance Verification.

4 Click Start PV.

The results will appear onscreen.
To perform complete PV tests for an emulation probe

The LAN tests can only be executed through the RS-232 port. The remainder of this section assumes that the tests are being run from a terminal emulator connected to the RS-232 port.

1 Disconnect the 50-pin cable from the emulation probe, and plug the emulator loopback test board (Agilent part number E3496-66502) directly into the emulator. Do not plug anything into the other end of the emulator loopback test board.

On a good system, the RESET LED will light and the BKG and USER LEDs will be out.

2 Connect a coaxial cable between BREAK IN and TRIGGER OUT.

3 Set all of the switches to CLOSED.

This is standard RS-232 at 9600 baud which can be connected directly to a 9 pin RS-232 interface that conforms to the IBM PC-AT 9 pin standard.

4 Use a terminal emulator to connect to the emulation probe.

5 Enter the pv 1 command.

See Also

Options available for the "pv" command are explained in the help screen displayed by typing "help pv" or "? pv" at the prompt.
Solving Problems

Problems with the Emulation Probe Itself

Examples

Here are some examples of ways to use the pv command.

To execute both tests one time:

```
pv 1
```

To execute test 2 with maximum debug output repeatedly until a ^C is entered:

```
pv -t2 -v9 0
```

To execute tests 3, 4, and 5 only for 2 cycles:

```
pv -t3-5 2
```

The results on a good system, with the BNC’s connected, and with the loopback test board connected, are as follows:

```
c>pv 1
Testing: E3499B Series Emulation System passed!
Test # 1: Powerup PV Resultspassed!
Test # 2: LAN 10Base2 Feedback Testpassed!
Test # 3: LAN 10BaseT Feedback Testpassed!
Test # 4: Break In and Trigger Out BNC Feedback Test passed!
Test # 5: Target Probe Feedback Testpassed!
Test # 6: Boundary Scan Master Testpassed!
Test # 7: 12C Testpassed!
Test # 8: Data Lines Testpassed!
PASSED Number of tests: 1Number of failures: 0
```

Copyright (c) Agilent Technologies 1987
All Rights Reserved. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under copyright laws.

E3499B Series Emulation System
Version: A.07.56 28Sept98
Location: Generics

E3459A ARM7 JTAG Emulator
Version: A.01.50 13Oct98
Solving Problems

Problems with the Emulation Probe Itself

If a performance verification test fails

There are some things you can do if a failure is found on one of these tests. Details of the failure can be obtained through using a verbose level of 2 or more.

If the particular failure you see is not listed below, contact Agilent Technologies for assistance.

TEST 2: LAN 10BASE2 Feedback Test failed

For LAN 10BASE2 test, the following is an example of a failure which is not caused by a broken emulator.

R>pv -t2 -v2 1

Testing: E3499B Series Emulation System
Test # 2: LAN 10Base2 Feedback Test failed!
FAILED - no lan connection (LAN probably not terminated
FAILED Number of tests: 1 Number of failures: 1

Check to see that the port under test has a good cable connected to it and that the cable is properly terminated with a 50 ohm terminator on each end of the overall cable.

R>pv -t2 -v2 1

Testing: E3499B Series Emulation System
Test # 2: LAN 10Base2 Feedback Test failed!
FAILED due to excessive collisions
FAILED Number of tests: 1 Number of failures: 1

The most common cause of this problem is poor termination of the cable or failure to remove the port under test from the LAN before performing the test. Check to see that the terminators are good (50 Ohms) and that you are isolated from any traffic on a system LAN.

R>pv -t2 -v2 1

Testing: E3499B Series Emulation System
Test # 2: LAN 10Base2 Feedback Test failed!
FAILED - invalid Ethernet address in EEPROM
FAILED Number of tests: 1 Number of failures: 1

First check to see that a correct LLA and IP address have been set in the virtual EEPROM through the "lan" command. If the "lan" command shows bad information for the LLA and IP, then try to set them to correct values. If you
are unable to set them to correct values, there is a failure in the FLASH ROM which requires service from Agilent Technologies.

**Test 3: 10BaseT Feedback Test failed here**

```
R>pv -t3 -v2 1
Testing: E3499B Series Emulation System
Test # 3: LAN 10BaseT Feedback Test failed!
FAILED Number of tests: 1 Number of failures: 1
```

In addition to the internal checks performed in Test 2, this test also checks for shorts on the cable connected to the network. If this test fails, disconnect the cable and run the test again. If if then passes, the cable is faulty. If it still fails, it requires service from Agilent Technologies.

If the emulator passes this "pv" test, additional testing can be performed through exercising the connection to the network. To run this test, set configuration switch 1 and switch 5 to OPEN, all other configuration switches CLOSED (this enables LAN using 10BaseT). Cycle power and wait for 15 to 30 seconds. Then "ping" the emulator from your host computer or PC. See the LAN documentation for your host computer for the location and action of the "ping" utility. If the emulator fails to respond to the "ping" request, verify that the lan parameters (IP address and gateway address) are set correctly and that your host computer recognizes the IP address of the emulator. If all else is good, then failure to respond to ping indicates a faulty emulator.

**Test 4: Break In and Trigger Out BNC Feedback Test**

```
R>pv -t4 -v2 1
Testing: E3499B Series Emulation System
Test # 4: Break In and Trigger Out BNC Feedback Test failed!
Break In not receiving Break Out HIGH
FAILED Number of tests: 1 Number of failures: 1
```

Before returning to Agilent Technologies, check to ensure that you have connected a good Coaxial cable between the two BNCs. If the cable is good, the emulator is bad.
Solving Problems

Problems with the Emulation Probe Itself

TEST 5: Target Probe Feedback Test
TEST 6: Boundary Scan Master Test
TEST 7: I2C Test
TEST 8: Data Line Test

If these tests are not executed, check that you have connected the emulator loopback test board.

If these tests fail, return the emulator to Agilent Technologies for replacement.
Returning the Parts to Agilent Technologies for Service

To return a part to Agilent Technologies

1. Follow the procedures in this chapter to make sure that the problem is caused by a failure in the emulator or power supply, not by configuration or communication problems.

2. In the U.S., call 1-800-403-0801. Outside the U.S., call your nearest Agilent Technologies sales office. Ask them for the address of the nearest Agilent Technologies service center.

   A list of Agilent Technologies sales offices is included at the back of this manual.

3. Package the part and send it to the Agilent Technologies service center.

   Keep any parts which you know are working. For example, if only the power supply is broken, keep the target interface module, emulator and cables.

4. When the part has been replaced, it will be sent back to you.

   The unit returned to you will have the same serial number as the unit you sent to Agilent Technologies.

   The Agilent Technologies service center can also troubleshoot the hardware and replace the failed part. To do this, send your entire measurement system to the service center, including the logic analysis system, target interface module, and cables.

   In some parts of the world, on-site repair service is available. Ask an Agilent Technologies sales or service representative for details.
To obtain replacement parts

The repair strategy for this emulation solution is board replacement. The following table lists some parts that may be replaced if they are damaged or lost. Contact your nearest Agilent Technologies Sales Office for further information on servicing the board.

Exchange assemblies are available when a repairable assembly is returned to Agilent Technologies. These assemblies have been set up on the "Exchange Assembly" program. This allows you to exchange a faulty assembly with one that has been repaired, calibrated, and performance verified by the factory. The cost is significantly less than that of a new assembly.

**Exchange and Replacement Assemblies Part Numbers**

<table>
<thead>
<tr>
<th>Exchange Assemblies Part Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>E3459-69401</td>
<td>Programmed emulation probe</td>
</tr>
<tr>
<td>16600-69515</td>
<td>Emulation module</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Replacement Assemblies Part Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0950-3043</td>
<td>Power supply (for emulation module only)</td>
</tr>
<tr>
<td>E3459-61601</td>
<td>20-pin cable</td>
</tr>
<tr>
<td>E3459-61602</td>
<td>14-pin cable</td>
</tr>
<tr>
<td>E3459-61603</td>
<td>20-pin cable</td>
</tr>
<tr>
<td>E3496-66502</td>
<td>Processor probe loopback test board</td>
</tr>
<tr>
<td>E3459-66502</td>
<td>Target interface module (ARM7TDMI JTAG board)</td>
</tr>
</tbody>
</table>
To clean the emulator

If the emulator requires cleaning:

1. Remove power from the instrument.
2. Clean the instrument using a soft cloth that has been moistened in a mixture of mild detergent and water.
3. Make sure that the instrument is completely dry before reconnecting it to a power source.
Solving Problems

Returning the Parts to Agilent Technologies for Service
Glossary

**Analysis Probe** A probing solution connected to the target microprocessor. It provides an interface between the signals of the target microprocessor and the inputs of the logic analyzer. Formerly called a “preprocessor.”

**Elastomeric Probe Adapter** A connector that is fastened on top of a target microprocessor using a retainer and knurled nut. The conductive elastomer on the bottom of the probe adapter makes contact with pins of the target microprocessor and delivers their signals to connection points on top of the probe adapter.

**Emulation Module** An emulation module is installed within the mainframe of a logic analyzer. It provides run control within an emulation and analysis test setup. See Emulation Probe.

**Emulation Probe** An emulation probe is a standalone instrument connected via LAN to the mainframe of a logic analyzer or to a host computer. It provides run control within an emulation and analysis test setup. Formerly called a “processor probe” or “software probe.” See Emulation Module.

**Extender** A part whose only function is to provide connections from one location to another. One or more extenders might be stacked to raise a probe above a target microprocessor to avoid mechanical contact with other components installed close to the target microprocessor. Sometimes called a “connector board.”

**Flexible Adapter** Two connection devices coupled with a flexible cable. Used for connecting probing hardware on the target microprocessor to the analysis probe.

**General-Purpose Flexible Adapter** A cable assembly that connects the signals from an elastomeric probe adapter to an analysis probe. Normally, a male-to-male header or transition board makes the connections from the general-purpose flexible adapter to the analysis probe.

**High-Density Adapter Cable** A cable assembly that delivers signals from an analysis probe hardware interface to the logic analyzer pod cables. A high-density adapter cable has a single Mictor connector that is installed into the analysis probe, and two cables that are connected to corresponding odd and even logic analyzer pod cables.
Glossary

**High-Density Termination Adapter Cable** Same as a High-Density Adapter Cable, except it has a termination in the Mictor connector.

**Jumper** Moveable direct electrical connection between two points.

**Mainframe Logic Analyzer** A logic analyzer that resides on one or more board assemblies installed in an Agilent Technologies 16500, or 16600A/700A-series mainframe.

**Male-to-male Header** A board assembly that makes point-to-point connections between the female pins of a flexible adapter or transition board and the female pins of an analysis probe.

**Preprocessor** See Analysis Probe.

**Preprocessor Interface** See Analysis Probe.

**Probe adapter** See Elastomeric Probe Adapter.

**Processor Probe** See Emulation Probe.

**Prototype Analyzer** The Agilent Technologies 16505A prototype analyzer acts as an analysis and display processor for the Agilent Technologies 16500B/C logic analysis system. It provides a windowed interface and powerful analysis capabilities. Replaced by Agilent Technologies 16600A/700A-series logic analysis systems.

**Run Control Probe** See Emulation Probe and Emulation Module.

**Setup Assistant** A software program that guides a user through the process of connecting and configuring a logic analyzer to make measurements on a specific microprocessor.

**Shunt Connector.** See Jumper.

**Software Probe** See Emulation Probe.

**Solution** Agilent Technologies' term for a set of tools for debugging your target system. A solution includes probing, inverse assembly, the Agilent Technologies B4620B Source Correlation Tool Set, and an emulation module.

**Stand-alone Logic Analyzer** A standalone logic analyzer has a predefined set of hardware components which provide a specific set of capabilities. It is designed to perform logic analysis. A standalone logic analyzer differs from a
mainframe logic analyzer in that it does not offer card slots for installation of additional capabilities, and its specifications are not modified based upon selection from a set of optional hardware boards that might be installed within its frame.

**Target Control Port** An 8-bit, TTL port on a logic analysis system that you can use to send signals to your target system. It does not function like a pattern generator or emulation module, but more like a remote control for the target's switches.

**Target Interface Module** A small circuit board which connects the 50-pin cable from an emulation module or emulation probe to signals from the debug port on a target system.

**TIM** See Target Interface Module.

**Trigger Specification** A set of conditions that must be true before the instrument triggers. See the printed or online documentation for your logic analyzer for details.

**Transition Board** A board assembly that obtains signals connected to one side and rearranges them in a different order for delivery at the other side of the board.

**1/4-Flexible Adapter** An adapter that obtains one-quarter of the signals from an elastomeric probe adapter (one side of a target microprocessor) and makes them available for probing.
Glossary
Index

A
address, IP
  IP address, 30
altitude specifications, 105
analysis probe
  definition, 147
assistant
  See setup assistant

B
BNC
  break in, 80
  LAN, 29
  trigger out, 80
  BNC, LAN, 104
  bootp, 36
built-in commands
  configuration, 66
  LAN configuration, 32
  list of commands, 118

C
cable
  LAN, 38
  replacing, 144
  serial, 42
  CD-ROM, installing software from, 53
  cf commands, 66
  cleaning, 145
  clocks
    specifications, 104
    configuration, 63
    emulation module, 64, 68
    connecting, 17
    connecting to the target system, 55
    connection, 20, 130, 133
    host workstation, 27

connector
  10BASE2, 29, 104
  10BASE-T, 29, 104
  JTAG, 86
  JTAG, designing, 84
  JTAG, levels, 123
  serial, 104
  connector board, 147

D
data communications
  specifications, 104
  DCE or DTE selection and RS-232
    cable, 132
descriptor port
  connecting to, 57
descriptor
  writing, 120
descriptors
  configuration, 68
driver firmware error, 124

E
elastomeric probe adapter
  definition, 147
Emulation Control Interface
  configuration, 65
emulation module
  configuration, 68
definition, 147
description of, 3
  using debugger, 68
description
  definition, 147
environmental characteristics
  emulation module, 105
equipment supplied
  emulation migration, 24
  ethernet address, 33
  extender, 147

F
firmware
  version, 109, 110
flexible adapter
  definition, 147

G
gateway address, 33, 130
general-purpose flexible adapter
  definition, 147

H
high-density adapter cable
  definition, 147
high-density termination adapter
  definition, 148
host computer
  connecting to, 27

I
IEEE 802.3, 29
init command, 124
internet address, 30
IP address, 29, 30, 33, 129, 130
IPC
  connecting to, 27

J
JTAG
  designing connector, 84
  jumper, definition, 148

L
lan command, 32
LAN connection problems, 130, 140, 141
LAN interface, 29
LAN parameters, configuring
  BOOTP, 35
  methods, 29
  terminal interface, 32
Index

link beat, 38
link level address, 33, 36

M
mainframe logic analyzer
   definition, 148
male-to-male header
   definition, 148
mask, subnet, 131
MAU, 29, 104
memory
   testing, 127
minimum equipment, 16
monitor, 71

N
noise, reducing, 86

O
operating characteristics
   emulation module, 105
overview, 11, 64

P
performance verification test, 137
ping command, 130
POL light, 117, 130
port number, 33
power on/off sequence, 17
power supply, 17
power up self test, 134
preprocessor
   See analysis probe
problems, 113
problems, LAN, 130
problems, RS-232, 133
processor support package, 53
prompts, 120
   list of, 120
   troubleshooting, 124
prototype analyzer
   definition, 148

R
real-time runs, configuring, 70, 71
register commands, 121
requirements, 16
RESET
   configuring, 78
   light, 117
run control tool
   See emulation control interface

S
self test, 134
sequence, 20
serial connection
   DCE or DTE selection, 132
   number of connections, 133
   problems, 133
   setting up, 40
   verifying, 44
   service ports, 33
   service, how to obtain, 143
   setup
   See configuration
   setup assistant, 111
   definition, 148
signals
   JTAG, 85
   signals, expected levels, 123
slow clock, 115
software probe
   See emulation probe
solution
   definition, 148
   specifications
   altitude, 105
   clock, 104
   data communications, 104
   temperature, 105
   trigger in/out, 104
StarLAN, 29, 38
status lights, 116, 118, 134
subnet mask, 29, 130, 131
switches, 36
   bootp, 36
   LAN configuration, 32, 38
   serial configuration, 40

T
target control port
   definition, 149
target interface module, 57
target interface module (TIM)
   definition, 149
target system
   connecting to, 56
   problems with, 121
TCP, 33
TDO, TDI signals, 86
telnet, 38, 129
temperature specifications, 105
terminal (MS Windows program)
   133
terminal interface, 38
   built-in commands, 118
   LAN parameters, setting, 32
ThinLAN, 29, 104
trace length, 86
transition board
   definition, 149
   trigger
   in/out specifications, 104
troubleshooting, 115
TX light, 117

U
Updating Firmware, 108

V
versions
   emulation module firmware, 110
   emulation probe firmware, 109
Index

W
wizard
  See setup assistant
workstation
  connecting to, 27
# DECLARATION OF CONFORMITY

according to ISO/IEC Guide 22 and EN 45014

<table>
<thead>
<tr>
<th>Manufacturer's Name:</th>
<th>Agilent Technologies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manufacturer's Address:</td>
<td>Digital Design Product Generation Unit 1900 Garden of the Gods Road Colorado Springs, CO 80907 USA</td>
</tr>
</tbody>
</table>

declares, that the product

<table>
<thead>
<tr>
<th>Product Name:</th>
<th>Processor Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Number(s):</td>
<td>E3459A</td>
</tr>
<tr>
<td>Product Option(s):</td>
<td>All</td>
</tr>
</tbody>
</table>

conforms to the following Product Specifications:

**Safety:**
- UL3111
- CSA-C22.2 No. 1010.1:1993

**EMC:**
- CISPR 11:1990 / EN 55011:1991  Group 1 Class A
- IEC 801-2:1991 / EN 50082-1:1992  4 kV CD, 8 kV AD
- IEC 801-3:1984 / EN 50082-1:1992  3 V/m, {1kHz 80% AM, 27-1000 MHz}
- IEC 801-4:1998 / EN 50082-1:1992  0.5 kV Sig. Lines, 1 kV Power Lines

**Supplementary Information:**


This product was tested in a typical configuration with Agilent Technologies test systems.

Colorado Springs, 02/18/98

Ken Wyatt / Product Regulations Manager

European Contact: Your local Agilent Technologies Sales and Service Office or Agilent Technologies GmbH, Department ZQ / Standards Europe, Herrenberger Strasse 130, D-71034 Böblingen Germany (FAX: +49-7031-14-3143)
**Product Regulations**

**Safety**  
UL3111  
CSA-C22.2 No. 1010.1:1993

**EMC**  
This Product meets the requirement of the European Communities (EC) EMC Directive 89/336/EEC.

**Emissions**  
EN55011/CISPR 11 (ISM, Group 1, Class A equipment)  
IEC 555-2 and IEC 55-3

**Immunity**  
<table>
<thead>
<tr>
<th>EN50082-1</th>
<th>Code</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEC 801-2 (ESD) 8kV AD</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>IEC 801-3 (Rad.) 3V/m</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>IEC 801-4 (EFT) 1kV</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

1Performance Codes:  
1 PASS - Normal operation, no effect.  
2 PASS - Temporary degradation, self recoverable.  
3 PASS - Temporary degradation, operator intervention required.  
4 FAIL - Not recoverable, component damage.

2Notes:  
1 The target cable assembly is sensitive to ESD events. Use standard ESD preventative practices to avoid component damage.

**Sound Pressure Level**  
N/A
Safety
This apparatus has been designed and tested in accordance with IEC Publication 1010, Safety Requirements for Measuring Apparatus, and has been supplied in a safe condition. This is a Safety Class I instrument (provided with terminal for protective earthing). Before applying power, verify that the correct safety precautions are taken (see the following warnings). In addition, note the external markings on the instrument that are described under "Safety Symbols."

Warning
• Before turning on the instrument, you must connect the protective earth terminal of the instrument to the protective conductor of the (mains) power cord. The mains plug shall only be inserted in a socket outlet provided with a protective earth connection. You must not negate the protective action by using an extension cord (power cable) without a protective conductor (grounding). Grounding one conductor of a two-conductor outlet is not sufficient protection.
• Only fuses with the required rated current, voltage, and specified type (normal blow, time delay, etc.) should be used. Do not use repaired fuses or short-circuited fuse holders. To do so could cause a shock or fire hazard.
• Service instructions are for trained service personnel. To avoid dangerous electric shock, do not perform any service unless qualified to do so. Do not attempt internal service or adjustment unless another person, capable of rendering first aid and resuscitation, is present.
• If you energize this instrument by an auto transformer (for voltage reduction), make sure the common terminal is connected to the earth terminal of the power source.
• Whenever it is likely that the ground protection is impaired, you must make the instrument inoperative and secure it against any unintended operation.

Safety Symbols
Instruction manual symbol: The product is marked with this symbol when it is necessary for you to refer to the instruction manual in order to protect against damage to the product.

Hazardous voltage symbol.

Earth terminal symbol: Used to indicate a circuit common connected to grounded chassis.

CAUTION
The Caution sign denotes a hazard. It calls attention to a procedure, practice, or the like, which, if not correctly performed or adhered to, could result in personal injury. Do not proceed beyond a Caution sign until the indicated conditions are fully understood and met.

Agilent Technologies
P.O. Box 2197
1900 Garden of the Gods Road
Colorado Springs, CO 80901-2197, U.S.A.
Product Warranty
This Agilent Technologies product has a warranty against defects in material and workmanship for a period of one year from date of shipment. During the warranty period, Agilent Technologies Company will, at its option, either repair or replace products that prove to be defective.

For warranty service or repair, this product must be returned to a service facility designated by Agilent Technologies.

For products returned to Agilent Technologies for warranty service, the Buyer shall prepay shipping charges to Agilent Technologies and Agilent Technologies shall pay shipping charges to return the product to the Buyer. However, the Buyer shall pay all shipping charges, duties, and taxes for products returned to Agilent Technologies from another country.

Agilent Technologies warrants that its software and firmware designated by Agilent Technologies for use with an instrument will execute its programming instructions when properly installed on that instrument. Agilent Technologies does not warrant that the operation of the instrument software, or firmware will be uninterrupted or error free.

Limitation of Warranty
The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance by the Buyer, Buyer-supplied software or interfacing, unauthorized modification or misuse, operation outside of the environmental specifications for the product, or improper site preparation or maintenance.

No other warranty is expressed or implied. Agilent Technologies specifically disclaims the implied warranties of merchantability or fitness for a particular purpose.

Exclusive Remedies
The remedies provided herein are the buyer’s sole and exclusive remedies. Agilent Technologies shall not be liable for any direct, indirect, special, incidental, or consequential damages, whether based on contract, tort, or any other legal theory.

Assistance
Product maintenance agreements and other customer assistance agreements are available for Agilent Technologies products. For any assistance, contact your nearest Agilent Technologies Sales Office.

Certification
Agilent Technologies certifies that this product met its published specifications at the time of shipment from the factory. Agilent Technologies further certifies that its calibration measurements are traceable to the United States National Institute of Standards and Technology, to the extent allowed by the Institute’s calibration facility, and to the calibration facilities of other International Standards Organization members.