Agilent Technologies

Dear Customer:

You have probably heard from news reports and from your sales representative that as of November 1, 1999, four of Hewlett-Packard's businesses became a new company -- Agilent Technologies. The new company includes the following former HP businesses: test and measurement, semiconductor products, healthcare solutions and chemical analysis."

We at Agilent Technologies are working diligently to make this transition as seamless as possible for you, however, we are not able to make all changes immediately. As a result, the products and related documentation may be labeled with either the Hewlett-Packard name and logo or the Agilent Technologies name and logo. Rest assured that whatever logo you see, the information, products and services come from the same reliable source.

In addition, it is our sincere intent that the transition from Hewlett Packard to Agilent Technologies should have no impact on your warranties, service levels, or purchase volume credits.

For more information about this transition, please visit our website at: http://www.agilent.com, or contact your local sales representative. It has been our pleasure to work with you for the past 60 years as part of Hewlett-Packard. We look forward to continuing to serve you as Agilent Technologies for years to come.

User's Guide

Publication number E3454-97001
December 1998

For Safety information, Warranties, and Regulatory information, see the pages behind the index.

Emulation for the PowerPC 700

Embedded PowerPC 700 Emulation—At a Glance

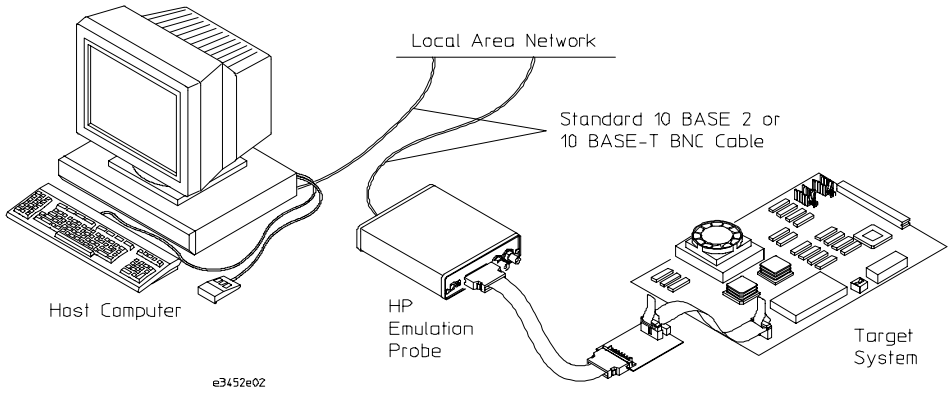
This manual describes how to set up several HP emulation products: an emulation probe, an emulation module, and an emulation migration.

These emulators provide a low-cost way to debug embedded software for Embedded PowerPC 700 microprocessors. The emulator 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 emulator to debug code running on the target system.

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

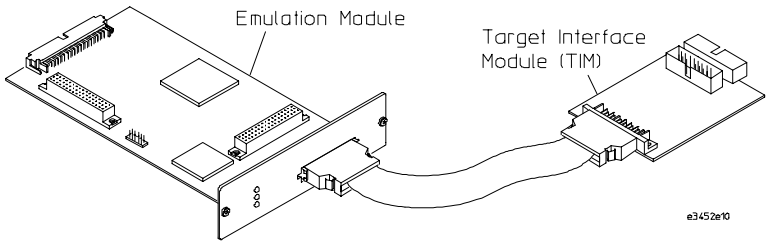
Emulation Probe

The emulation probe is a stand-alone emulator.



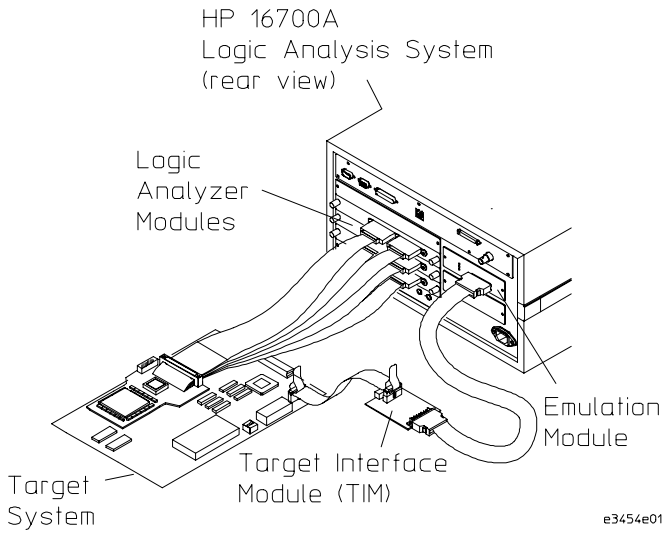
Emulation Module

The emulation module plugs into your HP 16600A/700A-series logic analysis system frame.



Emulation Migration

The emulation migration includes a TIM and firmware. Use the emulation migration if you already have an emulation probe or an emulation module for another processor and you wish to migrate to support of a different processor.



HP does not offer an analysis probe for PowerPC 7xx products. If you design logic analysis connections into your target system, you can connect the logic analyzer to perform state analysis of processor activity.

In This Book

This book documents the following products:

Emulation Probe

Processors supported	Product ordered	Includes
PowerPC 740 and PowerPC 750	HP E5900A Option #070	HP E3454A emulation probe, HPE3452A target interface module (TIM)

Emulation Module

Processors supported	Product ordered	Includes
PowerPC 740 and PowerPC 750	HP E5901A Option #070	HP 16610A emulation module, HP E3452A target interface module (TIM)

Emulation Migration

Processors supported	Product ordered	Includes
PowerPC 740 and PowerPC 750	HP E5902A Option #070	HP E3452A target interface module (TIM)

Contents

1

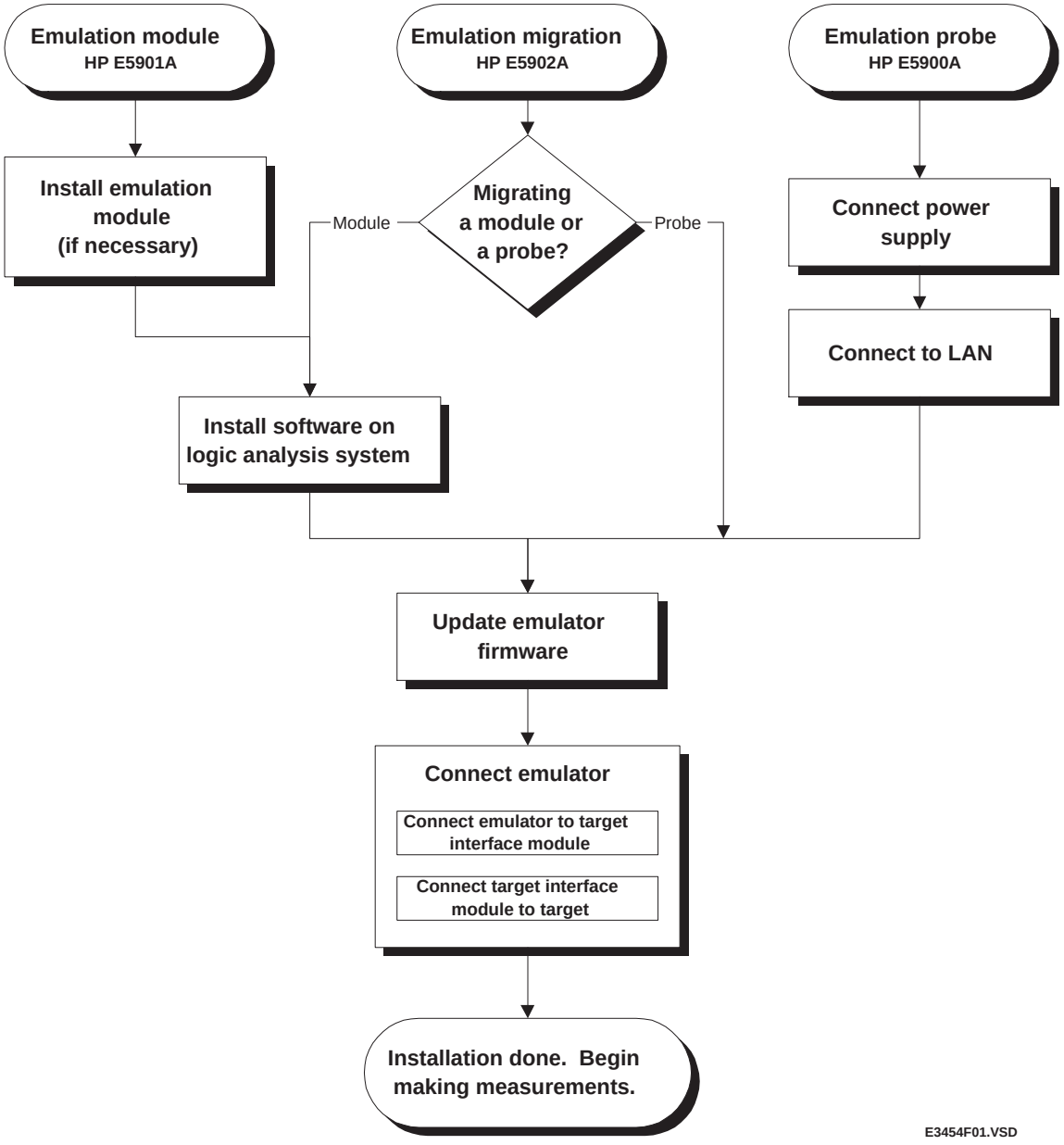
Overview

Overview

This chapter describes:

- Setup Checklist
- Equipment used with the emulation probe
- Connection sequences for the emulation probe
- Equipment used with the emulation module
- Additional information sources

Setup Flowchart



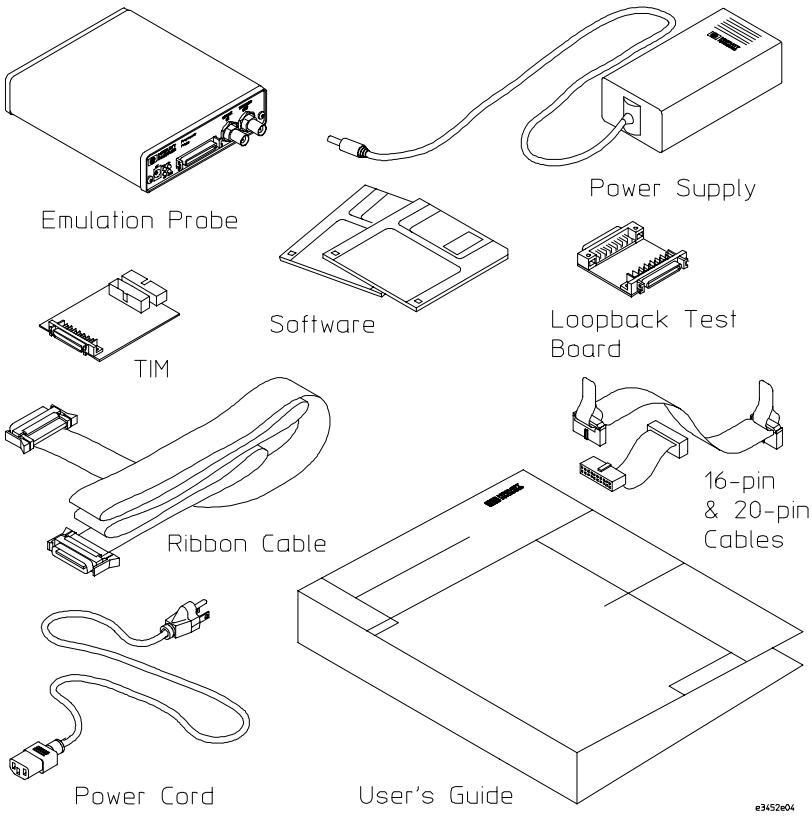
Emulation Probe

This section lists equipment supplied with the emulation probe and equipment requirements for using the emulation probe.

Equipment supplied

The equipment supplied with the emulation probe is shown in the illustration on the next page. It is listed below:

- An emulation probe.
- A 12V power supply for the emulation probe.
- A power cord.
- A target interface module (TIM) circuit board.
- An emulator loopback test board (HP part number E3496-66502).
- Firmware for the emulation probe on 3.5-inch disks.
- A 50-pin ribbon cable for connecting the emulation probe to the target interface module.
- A 16-pin ribbon cable for connecting the target interface module to the target system.
- A 20-pin ribbon cable (for use by certain third-party products).
- This User's Guide.



Equipment Supplied with the Emulation Probe

Minimum equipment required

The following equipment is required to use the emulation probe:

- A method for connecting to the target system. You can design a debug port connector on the target system.
- A host computer, such as a PC or workstation. You can also connect the emulation probe to an HP 16600A or HP 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.

To connect the emulation probe to a power source

The emulation probe does not have an On/Off switch. To turn the emulation probe on or off, plug or unplug it from the power supply.

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 Hewlett-Packard sales and service office.

Warning

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

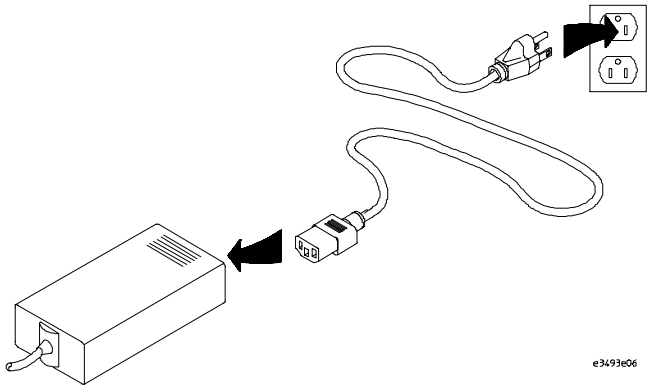
Caution

Use only the supplied HP power supply and cord.
Failure to use the proper power supply could result in equipment damage.

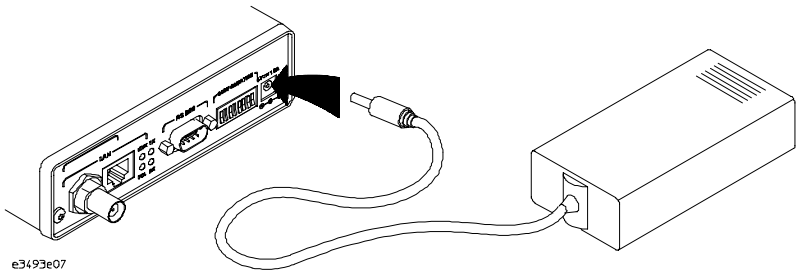
With all components connected, power on your system in the following order:

1. Logic analyzer, if you are using one.
2. Emulation probe.
3. Your target system.

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



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



The power lamp on the target side of the emulation probe will light. The emulation probe does not have an On/Off switch.

Power off your system in the following order:

1. Your target system
2. Emulation probe.
3. Logic analyzer, if you are using one.

Emulation probe connection sequence

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 27).
2. Connect the emulation probe to your target system (page 44).
3. Connect power and turn on your emulation probe.
4. Connect power and turn on your target system.

Important Note

When it powers up, the emulation probe tries to read the PVR register in the target processor. If it could read the PVR register, it would configure itself correctly. It can't read the PVR register because your target system is powered up last. In order to get the emulation probe to read the PVR register in the target processor and configure itself correctly, simply cycle power on the emulation probe (by disconnecting and then reconnecting the power supply cable) after your system is powered up. Failure to make the emulation probe read the PVR register in the target microprocessor may cause the emulation probe to be configured incorrectly.

5. Configure the emulation probe (page 67).

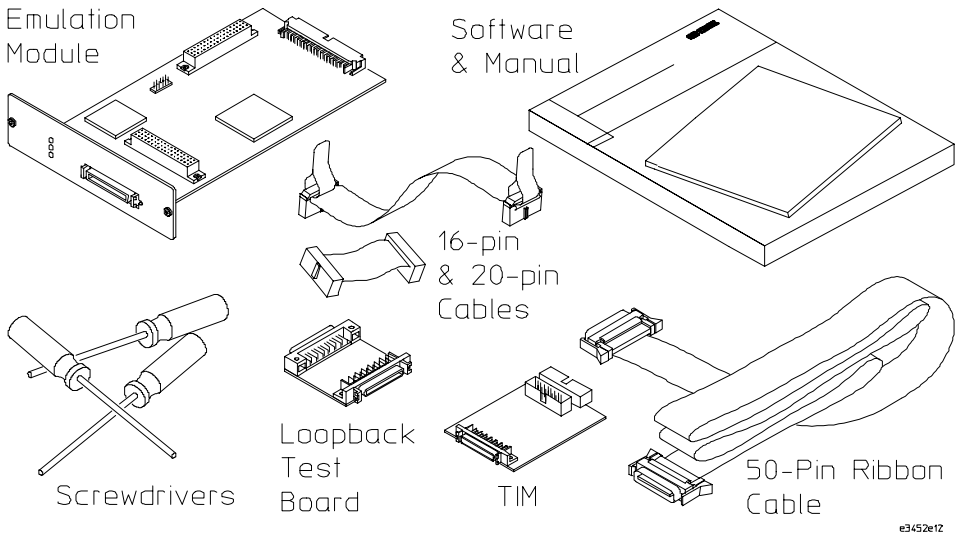
Emulation Module

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

Equipment supplied

The equipment supplied with your emulation module includes:

- An HP 16610A emulation module. If you ordered an emulation module as part of your HP 16600A or HP 16700A logic analysis system, it is already installed in the frame.
- A target interface module (TIM) circuit board.
- A emulation module loopback test board (HP part number E3496-66502).
- Firmware for the emulation module and/or updated 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 16-pin ribbon cable for connecting the target interface module to the target system.
- A 20-pin ribbon cable (for use with certain third-party products).
- One Torx T-8, one Torx T-10, and one Torx T-15 screwdriver (if the emulation module was not installed at the factory).
- This User's Guide.



Equipment Supplied with the HP E3497A Emulation Module

Minimum equipment required

The following equipment is required to use the emulation module:

- A method for connecting to the target system. You can design a debug port connector on the target system. Refer to Chapter 5 for information on designing a debug port connector on a target system.
- An HP 16600A or HP 16700A logic analysis system.
- A user interface, such as a high-level source debugger or the logic analysis system's Emulation Control Interface.

Power Up Configuration Requirement

When it powers up, the emulation module tries to read the PVR register in the target processor. If it could read the PVR register, it would configure itself correctly. It can't read the PVR register because your target system is powered up last. In order to get the emulation module to read the PVR register in the target processor and configure itself correctly, simply select the emulation module icon in the interface and click Update Firmware. Then in the Update Firmware window, click Display Current Version.

Failure to make the emulation module read the PVR register in the target microprocessor may cause the emulation module to be configured incorrectly.

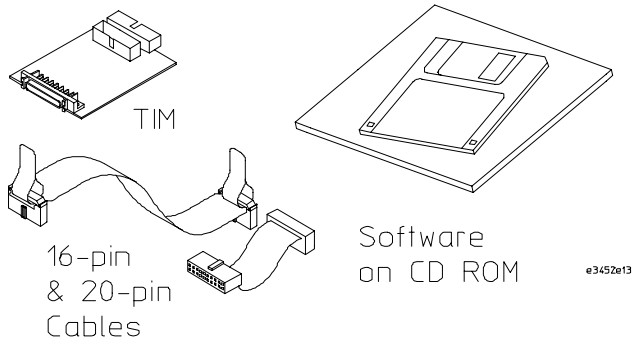
Emulation Migration

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.
- Firmware for the emulation probe on a floppy disk.
- A 16-pin ribbon cable for connecting the target interface module to the target system.
- A 20-pin ribbon cable for use by certain third-party products.
- 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. You can design a debug port connector on the target system. See Chapter 5 provides information on designing a debug port on the target system.
- A host computer such as a PC, a workstation, or an HP 16600A or HP 16700A logic analysis system.
- A user interface, such as a high-level source debugger or the logic analysis system's Emulation Control Interface.

Where to find emulation migration firmware

To change the personality of your emulation probe or emulation module for a new processor, you need to install new firmware.

If you have an emulation probe

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.

If you have an emulation module

Use the CD-ROM to install the appropriate processor support package (see page 49). This package installs the firmware on the hard disk of your HP 16600A/700A-series logic analysis system.

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 HP representative.

If you have a probing adapter, the instructions for connecting the probe to your target microprocessor are in the Probing Adapter documentation.

Application notes may be available from your local HP representative or on the World Wide Web at:

http://www.hp.com/go/logicanalyzer

If you have an HP 16600A or HP 16700A logic analysis system, the **online help** for the Emulation Control Interface has additional information on using the emulator.

The **measurement examples** include valuable tips for making emulation and analysis measurements. You can find the measurement examples under the system help in your HP 16600A/700A logic analysis system.

If you cannot easily find the information you need, send email to documentation@col.hp.com. Your comments will help HP improve future manuals. (This address is for comments only; contact your local HP representative if you need technical support.)

Connecting the Emulation Probe to a LAN

Connecting the Emulation Probe to a LAN

You can connect your PC or workstation 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. Other use of the serial port is not supported. Performance over a serial connection, especially if you are downloading code, may be unacceptably slow.

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

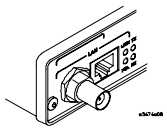
Use a LAN connection for routine use, and a serial connection for LAN configuration and for troubleshooting.

See Also

For information on LAN connections to an emulation *module*, see "Using the Emulator with a Debugger" beginning on page 83.

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 emulator 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. BOOTP is part of the HP-UX, SunOS, and Solaris operating systems.

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.**

   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).

   The default numbers (6470, 6471) can be changed if they conflict with some other product on your network.

   To change the port numbers, see page 31. 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.

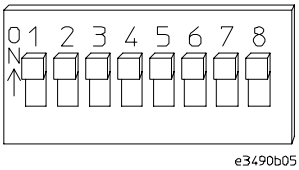
   The link-level address (LLA) is printed on a label above the LAN connectors on the emulation probe. This address is configured in each emulation probe shipped from the factory and cannot be changed.

IP Address of emulation probe
LAN Name of emulation probe
Gateway Address
Link-Level Address of emulation probe

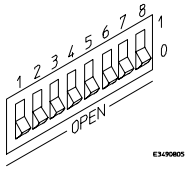
To configure LAN parameters using the built-in terminal interface

1. **Set configuration switches S1 through S4 to ON/CLOSED, 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:



-OR-



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 38.

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 "c>".**

   At this point, you are communicating with the emulation probe's built-in terminal interface.

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 <internet> [-g <gateway>] [-p <port>]
```

   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.

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.

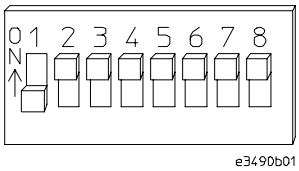
6 Disconnect the power cord from the emulation probe, and connect the emulation probe to your network.

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.

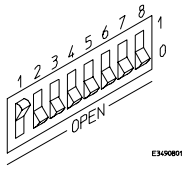
7 Set the configuration switches to indicate the type of connection that is to be made.

Set Switch S1 to OFF/OPEN, indicating that a LAN connection is being made.

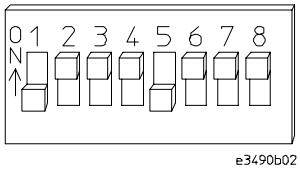
Set Switch S5 to ON/CLOSED if connecting to the BNC connector:



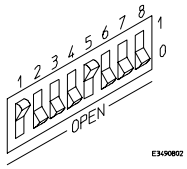
-OR-



Set Switch S5 to OFF/OPEN if connecting to the 10BASE-T connector:



-OR-



Set all other switches to ON/CLOSED.

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 37.

Once you have set a valid IP address, you can use the telnet utility to connect 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:

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

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

See Also

"Troubleshooting," page 133, 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:

```
$ 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.

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

```
# Global template for options common to all HP 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.

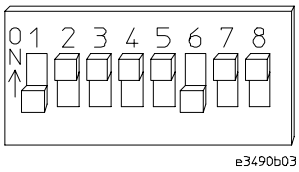
This connection can be made by using either LAN connector on the emulation probe.

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

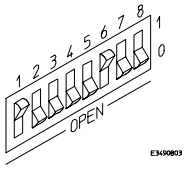
Set Switch S1to OFF/OPEN, indicating a LAN connection is being made.

Set Switch S6 to OFF/OPEN to enable BOOTP mode.

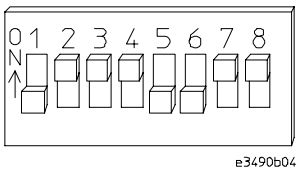
Set Switch S5 to ON/CLOSED if connecting to the BNC connector



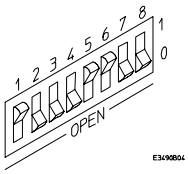
-OR-



Set Switch S5 to OFF/OPEN if connecting to the 10BASE-T connector.



-OR-



Set all other switches to ON/CLOSED.

5 Connect the power cord to the emulation probe.

Verify that the power light stays on after 10 seconds.

The IP address will be stored in EEPROM.

6 Set Switch S6 back to ON/CLOSED.

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/CLOSED will result in slower performance, increased LAN traffic, and even failure to power up (if the BOOTP server becomes inactive).

7 Verify your emulation probe is now active and on the network. See "To verify LAN communications" on page 37.

See Also

For additional information about using bootpd, refer to the bootpd (1M) man page.

To set the 10BASE-T configuration switches

Set Switches S7 and S8 to ON/CLOSED unless one of the following conditions is true:

- **If the LAN cable exceeds the standard length, set Switch S7 to OFF/OPEN.**

  The emulation probe has a switch-selectable, twisted-pair receiver threshold. With Switch S7 set to OFF/OPEN, 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/CLOSED, 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 (OFF/OPEN).**

  In normal mode (Switch S8 set to ON/CLOSED), 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/OPEN).

Note

Setting Switch S8 to OFF/OPEN 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.
2. To view the LAN parameters, enter the **lan** command at the terminal interface prompt.
3. 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 Have Problems

If you encounter problems, refer to the "troubleshooting" chapter (page 133).

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 : 08000F090B30
```

Setting Up a Serial Connection

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

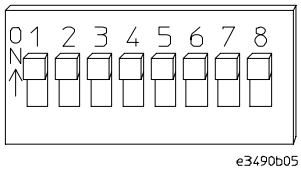
1. Set Switch S1 to ON/CLOSED (RS-232).
2. Set Switches S2-S4 to ON/CLOSED.
3. Set Switch S5 to ON/CLOSED (HW HANDSHAKE ON) if your serial interface uses the DSR:CTS/RTS lines for flow control. Set Switch S5 to OFF/OPEN (HW HANDSHAKE OFF) if your serial interface uses software flow control (XON/XOFF).

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

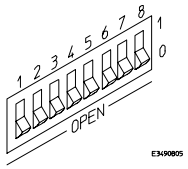
   The higher baud rates may not work reliably with all hosts and user interfaces. Make sure the baud rate you choose is supported by your host and user interface.

Example

To use a baud rate of 9600 baud, set the switches as follows:



-OR-



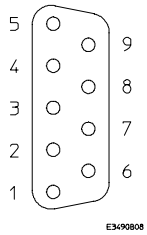
To connect a serial cable

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:



Serial Cable Pinout

Pin Number	Signal	Signal Description
1	DCD	Data Carrier Detect (not used)
2	TD	Transmit Data (data coming from HP emulation probe)
3	RD	Receive Data (data going to HP emulation probe)
4	DTR	Data Terminal Ready (not used)
5	GND	Signal Ground
6	DSR	Data Set Ready (Output from HP emulation probe)
7	RTS	Request to Send (Input to HP emulation probe)
8	CTS	Clear to Send (connected to pin 6)
9	RING	Ring Indicator (not used)

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) Hewlett-Packard Co. 1987
All Rights Reserved.  Reproduction, adaptation, or translation without prior
written permission is prohibited, except as allowed under copyright laws.

  HPE3499A Series Emulation System
    Version:   A.07.06 06May97
    Location:  Generics

 HPE3454A PowerPC 700 JTAG Emulator
    Version:   A.01.01 28Aug97


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>", "C>", or "?>".

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

See Also "Problems with the Serial Interface," page 145.

Setting up Debugger Software

Before you can use a debugger with the emulator, you may need to configure some communication parameters, including the LAN address you assigned to the emulation probe.

Use the Emulation Control Interface to configure the emulation probe. End the Emulation Control Interface session before you start the debugger.

> Do not use the Run Control tool at the same time as a debugger.

See Also

Refer to the documentation for your debugger for more information on connecting the debugger to the emulator.

3

Installing the Emulation Module

Installing the Emulation Module

This chapter shows you how to install an emulation module in your HP 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

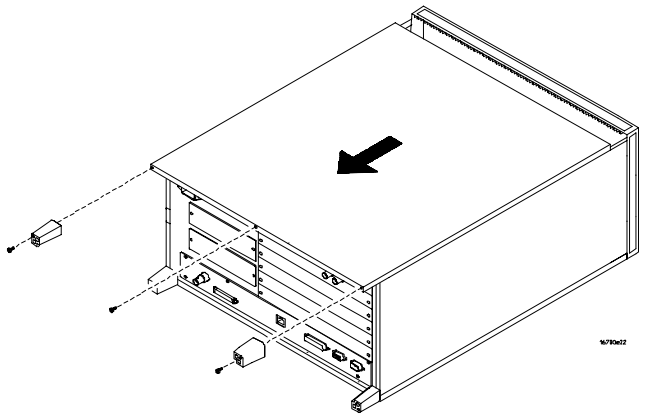
These instructions are for trained service personnel. To avoid dangerous electric shock, do not perform any service unless qualified to do so. Do not attempt internal service or adjustment unless another person, capable of rendering first aid and resuscitation, is present.

Electrostatic discharge can damage electronic components. Use grounded wriststraps and mats when you handle modules.

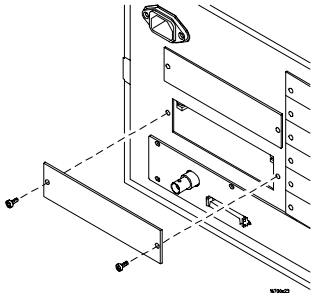
To install the emulation module in an HP 16700A-series logic analysis system or an HP 16701A expansion frame

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

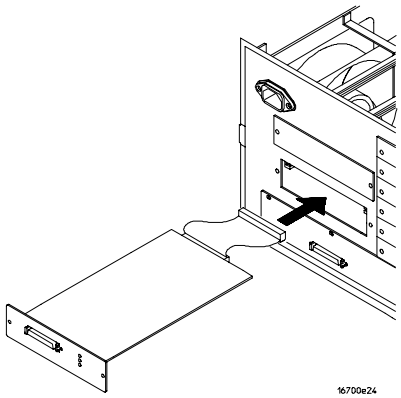
1. Turn off the logic analysis system and REMOVE THE POWER CORD.
   Remove any other cables (such as probes, mouse, or video monitor).
2. Turn the logic analysis system frame upside-down.
3. Remove the bottom cover.



4. Remove the slot cover.
   You may use either slot.

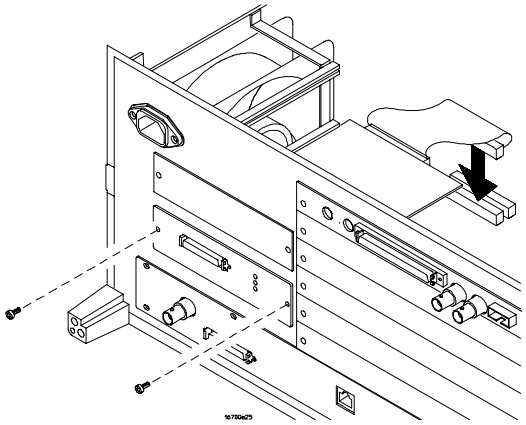


5 Install the emulation module.



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, and 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.

See Also

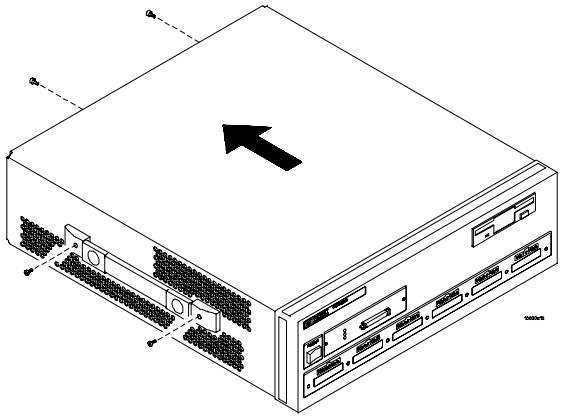
See page 111 for information on giving the emulation module a "personality" for your target processor.

To install the emulation module in an HP 16600A-series logic analysis system

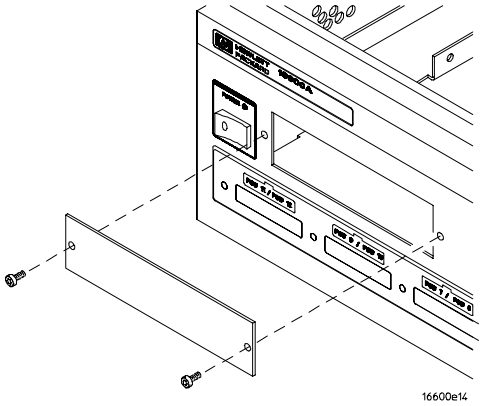
You will need T-8, T-10, and T-15 Torx screw drivers.

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.

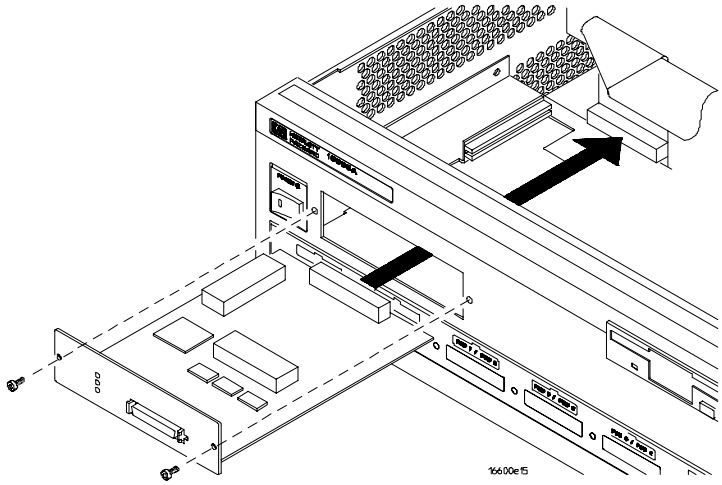


3 Remove the slot cover.



4 Install the emulation module.

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.

See Also

See page 111 for information on giving the emulation module a "personality" for your target processor.

4

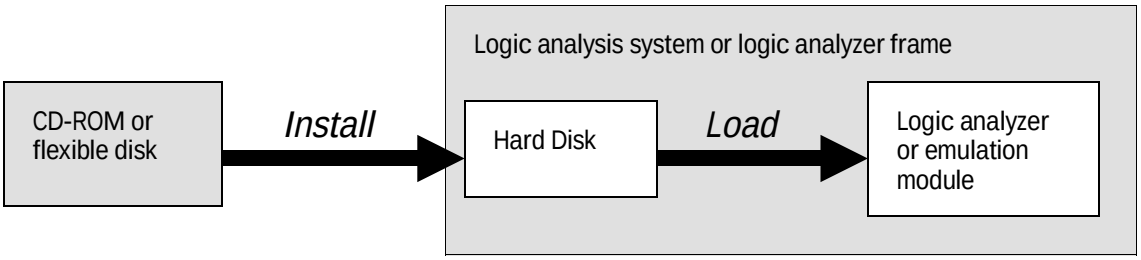
Installing Software on an HP 16600A/700A

Installing Software on an HP 16600A/700A

This chapter explains how to install the software you will need for your logic analyzer or emulation module.

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

HP 16600A/700A-series logic analysis systems

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

The following files are installed when you install a processor support package from the CD-ROM:

- Logic analysis system configuration files
- Inverse assembler (automatically loaded with the configuration files)
- Personality files for the Setup Assistant
- Emulation module firmware
- Emulation Control Interface

The HP B4620B Source Correlation Tool Set is installed with the logic analysis system's operating system.

The following files can be installed from the floppy disk that contains your inverse assembler.

- Logic analysis system configuration file
- Inverse assembler (automatically loaded with the configuration file)

To list software packages which are installed (HP 16600A/700A)

- In the System Administration Tools window, click **List...** .

To install the software from CD-ROM (HP 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 HP 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.
2. Insert the CD-ROM in the drive.
3. Click the **System Admin** icon.
4. Click **Install...** .

   Change the media type to "CD-ROM" if necessary.
5. Click **Apply**.
6. From the list of types of packages, select "PROC-SUPPORT."

   A list of the processor support packages on the CD-ROM will be displayed.
7. Click on the "POWERPC7XX" package.

   If you are unsure if this is the correct package, click Details for information on what the package contains.
8. Click **Install...** .

   The dialog box will display "Progress: completed successfully" when the installation is complete.
9. Click **Close**.

The configuration files are stored in /hplogic/configs/hp/*ppc7xx*.

The inverse assemblers are stored in /hplogic/ia.

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.

To load an inverse assembler from the floppy disk (HP 16600A/700A)

The preferred method is to install this functionality from the CD-ROM onto the hard disk and load from the hard disk.

To install a configuration and inverse assembler file from the floppy disk that was shipped with your HP inverse assembler product:

1. Install the floppy disk in the floppy drive on the HP 16600A/16700A-series logic analysis system mainframe.
2. In the Logic Analysis System window, click the **File Manager** icon.
3. In the File Manager window:
   - Set Current Disk to Flexible Disk.
   - Set Target to the analyzer you wish to configure.
   - Click the name of the desired configuration file in the Contents frame. The Contents frame lists the configuration files and inverse assembler files available on the floppy disk. These may be either DOS or LIF format files. Either format can be loaded directly into the appropriate logic analyzers.

   Note that the logic analyzers read both DOS and LIF formats. However, only DOS formatted floppy disks can be used to store configurations and data. LIF format floppy disks are read-only.
4. Click **Load**.

   The configuration file you choose will set up the logic analyzer and associated tools. You may see Information, Error, and Warning dialogs that say your configuration has been loaded, and advise you about making proper connections.
5. Click the **Workspace window** icon to see the arrangement of analysis tools in your configuration.
6. Right-click the logic analyzer icon in your configuration and choose its **Setup** button to see the way your configuration file defined the Config, Format, and Trigger options.

   Under the Format tab, buses are labeled, and bits included in each bus are identified by an asterisk "*".

   This procedure restores the configuration that was in effect when the configuration file was saved. Because the file was not saved using your system, you may receive error messages about loading the enhanced inverse assembler or about pods that were truncated. Click the Config, Format, and

Trigger tabs and modify the configuration to satisfy your measurement desires. Then you can save your customized configuration to DOS format using the File→Save Configuration selection in any of your tool windows, or clicking the Save tab in the File Manager. For details about how to save configuration files, open the Help window.

5

Connecting and Configuring the Emulator

Connecting and Configuring the Emulator

This chapter shows you how to connect the emulator to the target system and how to configure the emulator and target processor.

Overview

Here is a summary of the steps for connecting and configuring the emulator:

1. Make sure the target system is designed to work properly with the emulator. (Page 60.)
2. Install the emulation module in your logic analysis system, if necessary. (Page 43.)

   If you are connecting an emulation module to an HP 16600A/700A-series logic analysis system, use the Setup Assistant to guide you through steps 3-6. Use this manual for additional information, if desired.
3. Connect the emulator to your target system using the 50-pin cable and the TIM. (Page 65.)
4. Update the firmware of the emulator, if necessary. (Page 111.)
5. Verify communication between the emulator and the target.
6. Configure the emulator. (Page 67.)
7. Test the connection between the emulator and the target. (Page 81.)
8. Connect a debugger to the emulator, if applicable. (Page 83.)

See Also

"Using the Emulator with a Debugger" beginning on page 83 for information on configuring the emulator with a debugger, and for information on configuring LAN port numbers.

Using the Emulation Control Interface

The Emulation Control Interface in your HP 16600A/700A-series logic analysis system allows you to control an emulator (an emulation module or an emulation probe).

As you set up the emulator, you will use the Emulation Control Interface to:

- Update firmware (which reloads or changes the processor-specific personality of the emulator).
- Change the LAN port assignment (rarely necessary).
- Run performance verification tests on the emulator.

The Emulation Control Interface allows you to:

- Run, break, reset, and step the target processor.
- Set and clear breakpoints.
- Read and write registers.
- Read and write memory.
- Read and write I/O memory.
- View memory in mnemonic form.
- Read and write the emulator configuration.
- Download programs (in Motorola S-Record or Intel Hex format) to the target system RAM or ROM.
- View emulator status and errors.
- Write and play back emulator command script files .

If you have an emulation probe, this interface also allows you to configure the LAN address of the emulation probe.

Using the logic analysis system's intermodule bus does not require the Emulation Control Interface to be running. If the emulation module icon is in the Intermodule window, then it will be able to send and receive signals. Therefore if you are using a debugger, you can use an analyzer to cause a break.

Using a debugger with the Emulation Control Interface is not recommended because:

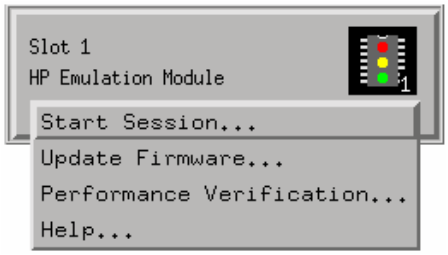
- The interfaces can get out of synchronization when commands are issued from both interfaces. This causes windows to be out-of-date and can cause confusion.
- Most debuggers cannot tolerate another interface issuing commands and may not start properly if another interface is running.

See Also

All of the Emulation Control Interface windows provide online help with a **Help** button or a **Help→On this window** menu selection. Refer to the online help for complete details about how to use a particular window.

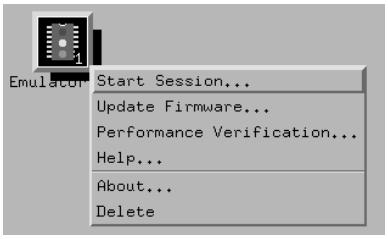
To start the Emulation Control Interface from the main System window (emulation module)

1 In the System window, click the emulation module icon.
2 Select **Start Session...**.



To start the Emulation Control Interface from the Workspace window (emulation module)

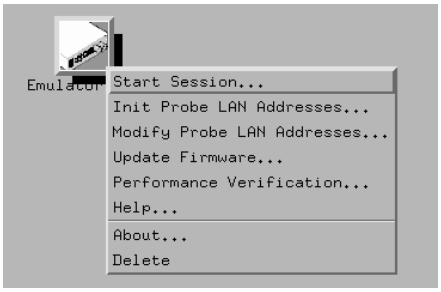
1 Open the Workspace window.
2 Drag the Emulator icon onto the workspace.
3 Right-click on the Emulator icon, and then select **Start Session...**.



To start the Emulation Control Interface from the Workspace window (emulation probe)

If you have a stand-alone emulation probe connected to the logic analysis system via LAN, use the Emulation Probe icon instead of the Emulation Module icon.

1 Open the Workspace window.
2 Drag the Emulation Probe icon onto the workspace.
3 Right-click the Emulation Probe icon, and select **Start Session...**.



4 In the Session window, enter the IP address or LAN name of the emulation probe; then click **Start Session**.

Designing a Target System for the Emulator

The following sections describe design considerations for your target system to operate properly with the emulator.

Target System Requirements for PowerPC 740/750

MMU Support

When the MMU is enabled in the PowerPC hardware, and the HP Emulator is configured for effective addresses, all memory addresses given to the emulator are assumed to be effective addresses (logical addresses). The emulator uses the MMU block address translation (BAT) registers, segment registers, hash tables, and other special-purpose MMU registers to compute each corresponding physical address. The requested memory operation is then performed using the physical address.

Operational notes:

- The emulator attempts to perform address translation only if the MSR[IR] and/or the MSR[DR] bits are set (=1) AND the emulator is configured to do translation (cf address=effective). The emulator configuration may be changed using the cf command:
  - cf address=effective (power up default value)
  - cf address=physical
- If both the MSR[IR] and MSR[DR] are set, the emulator will perform address translations by first searching the IBATs and then the DBATS, if no match is found in the IBATs. Note that the PowerPC silicon allows the IBAT and DBAT registers to specify overlapping effective address ranges. Avoid defining overlapping ranges. These make debugging more difficult because the emulator can use the IBATs to translate addresses intended for the DBATs.

- If an effective address is not found in the MMU translation tables, the emulator will return an error and will not perform the requested operation.
- Cache coherency is maintained during emulator MMU translations.
- Be sure the translation enable/disable condition is the same when you set and clear breakpoints. If a breakpoint is set while translation is enabled and then cleared while translation is disabled, the result will be erroneous and unpredictable. This is also true if a breakpoint is set while translation is disabled and then cleared while translation is enabled.
- The emulator ignores read-only restrictions defined in the MMU. (i.e. The emulator may attempt to write to memory that has been defined by the MMU as read-only.)
- MMU translation is automatic and transparent to debuggers connected to the emulator.

Unsupported modes

Target systems which use any of the following modes of operation are not currently supported:

- Address parity is not generated on external address bus operations. Accesses to devices that check parity will fail.

$\overline{\text{QACK}}$ signal

If the target development board does not use the $\overline{\text{QACK}}$ signal , the board must have a pull down resistor to pull this signal low. This allows the PowerPC to enter the debug state. Recommended value: 1KΩ or less.

TDO, TDI, TCK, TMS and $\overline{\text{TRST}}$ signals

TDO, TDI, TCK, TMS and $\overline{\text{TRST}}$ signal traces between the JTAG connector and the PPC740/750 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 PowerPC 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 $\overline{\text{TRST}}$ signals must not be actively driven by the target system when the JTAG port is being used.

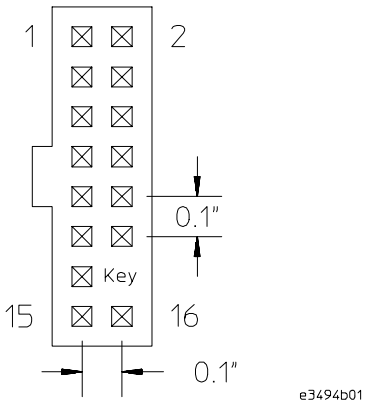
Reset signals

The $\overline{\text{HRESET}}$, $\overline{\text{SRESET}}$ and $\overline{\text{TRST}}$ signals from the JTAG connector must be logically ORed with the $\overline{\text{HRESET}}$, $\overline{\text{SRESET}}$ and $\overline{\text{TRST}}$ signals that connect to the processor on the target system. They cannot be "dotted" or "wire-ORed" on the board. The ORed signals should only reset the processor and no other devices on the target system.

The HP emulation module adds capacitance to all target system signals routed to the JTAG connector. This added capacitance may reduce the rise time of the $\overline{\text{SRESET}}$ or the $\overline{\text{HRESET}}$ signal beyond the processor specifications. If so, the target may need to increase the pull-up current on these signal lines.

PowerPC JTAG interface connections and resistors

The target system must have a 16-pin male 2x8 header connector with the following dimensions:



JTAG Header Connector (top view)

Position 14 of the connector on the target system must not contain a pin. The cable supplied with the emulator can only be installed if pin 14 has been removed from the header.

Place the connector as close as possible to the processor to ensure signal integrity.

PowerPC 7xx Connections

Header Pin Number	Signal Name	I/O	Board Resistor
1	TDO	Out	
2	Not connected		
3	TDI	In	1KΩ pulldown
4	$\overline{\text{TRST}}$	In	10KΩ pullup
5	Not connected		
6	+POWER[1]		1KΩ series[2]
7	TCK	In	10KΩ pullup
8	Not connected		
9	TMS	In	10KΩ pullup
10	Not connected		
11	$\overline{\text{SRESET}}$	In	10KΩ pullup
12	Not connected		
13	$\overline{\text{HRESET}}$	In	10KΩ pullup
14	KEY		
15	$\overline{\text{CSTP\_OUT}}$	Out	1KΩ pullup
16	GND		
	$\overline{\text{QACK}}$ [3]	In	1KΩ pulldown
	L2_TEST_CLK	In	10KΩ pullup
	L1_TEST_CLK	In	10KΩ pullup
	$\overline{\text{LSSD\_MODE}}$	In	10KΩ pullup
	ARRAY_WR	In	10KΩ pullup

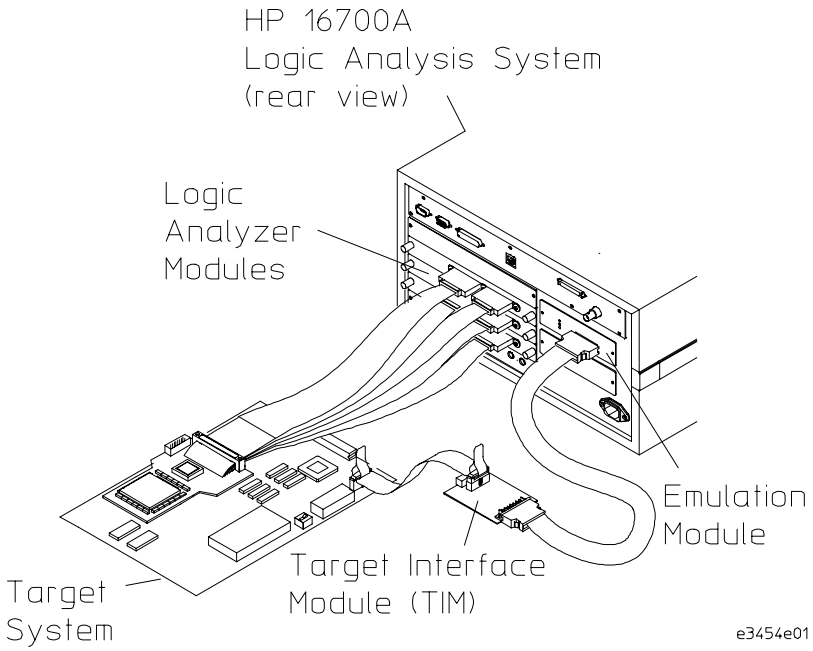
[1] The +POWER signal is sourced from the target system and is used as a reference signal. It should be the power signal being supplied to the processor (either +3.3V or +5V). It does not supply power to the HP emulator.

[2] This 1KΩ series resistor provides short circuit current limiting protection only. If the resistor is present, it should be 1KΩ or less.

[3] If the target system does not use this signal, the board must have a 1KΩ pulldown resistor connected to this pin. This signal allows the HP emulator to force the processor into soft stop mode. If the target system does use this signal, it should provide logic so that $\overline{\text{QACK}}$ goes low in response to a $\overline{\text{QREQ}}$.

Connecting the Emulator to the Target System

Connect the emulator and TIM to a target system directly through a JTAG connector on the target board. Use the procedure on the following page.



After you have connected the emulator to your target system, you may need to update the firmware in the emulator.
If you have designed logic analysis connections into your target system, you can use the logic analysis system to trace execution.

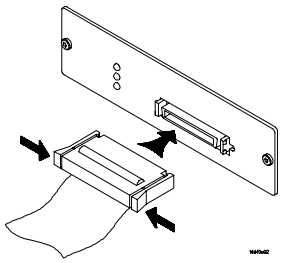
See Also

For information on designing a JTAG port on your target board, see page 60.
For a list of the parts supplied with the emulator, see page 21.

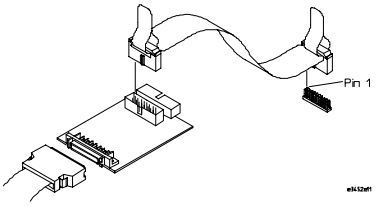
To connect to a target system using a JTAG port

The emulator can be connected to a target system through a 16-pin JTAG port connector. The emulator should be connected to a 16-pin male 2x8 header connector on the target system using the 16-conductor cable assembly provided.

1. Turn off the target system and disconnect it from all power sources.
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. Plug one end of the 16-pin cable into the target interface module.
5. Plug the other end of the 16-pin cable into the JTAG port on the target system.



6. Turn on the power to the logic analysis system and then the target system.

See Also "Designing a Target System" (page 60) for information on designing a target system for use with 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 HP 16600A or HP 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 HP emulator

What can be configured

There are two categories of configuration items: general configuration and cache configuration.

The default powerup configuration will generally work with many target systems if the cache is turned off.

If the instruction and data caches are both turned off, the cache configuration items are meaningless and can be ignored.

The following option can be configured using built-in commands:

- Restriction to real-time runs.

The built-in "help cf" command also lists the following options, which are provided only for compatibility with standalone emulation probes:

- BNC break in behavior.
- BNC trigger out behavior.

General Configuration

- JTAG clock speed
- Reset operation
- Memory read delays
- Memory write delays
- Parity bit information

Cache Configuration

- Memory read operation
- Data memory write operations
- Instruction memory write operations

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.

For an emulation module:

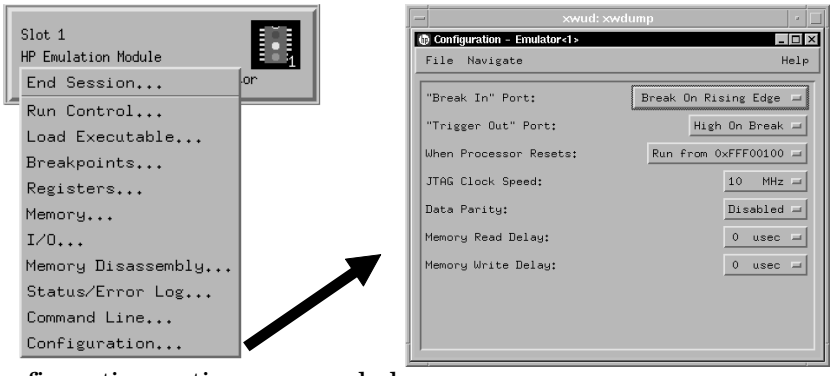
- In the system window, click the Emulation Control Interface icon, and then select "Start Session...".

For an emulation probe:

- In the workspace window, drag the Emulation Probe icon onto the workspace, 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.

The configuration selections 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...**.

See Also

Help →Help on this window in the Configuration window for information on each of the configuration options.

Help in the Emulation Control Interface menu for help on starting an Emulation Control session.

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 emulation module in Slot 1 enter:

   `telnet` *`LAN_address`* `6472`

2. Enter `cf` to see the current configuration settings.
3. Use the `cf` command to change the configuration settings.

See Also

Enter `help cf` for help on the configuration commands.

For information on connecting using telnet, and for information on other built-in commands, see page 139.

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> <item>=<value> <item> - set and display can be combined

  help cf <item>    - display long help for specified <item>

  --- VALID CONFIGURATION <item> NAMES ---
    rrt       - Restrict to real-time runs
    reset     - Configure reset actions
    speed     - Set JTAG clock
    mrdop     - Configure mem read operation
    dmwrop    - Configure D mem write operation
    imwrop    - Configure I mem write operation
    mrddel    - Set memory read delay
    mwrdel    - Set memory write delay
    breakin   - Select BNC break input option
    trigout   - Select BNC trigger output option
    parity    - Enable/disable data parity
M>
```

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

```
M>help cf rrt

  Restrict to real-time runs

    cf rrt=yes
    cf rrt=no

  If yes (and while the processor is running the user program), any
  command that requires the processor to be stopped will be rejected.
  For example 'reg' and 'm'.

  If no, commands that require the processor to be stopped will
  actually stop the processor, execute then resume running the
  processor.

M>
```

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

```
M>cf
  cf rrt=yes
  cf reset=runrom
  cf speed=1
  cf mrdop=mm
  cf dmwrop=mm
  cf imwrop=upd_dcu
  cf mrddel=0
  cf mwrdel=0
  cf breakin=off
  cf trigout=fixhigh
  cf parity=off
M>
```

To configure using a debugger

Because the HP 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 the "Using the Emulator with a Debugger" chapter (page 83).

Your debugger manual.

To configure restriction to real-time runs

Real-time runs configuration

Value	Emulator configured for	Built-in command
no	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.	`cf rrt=no`
yes	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. (Default)	`cf rrt=yes`

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 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 analysis system must be used instead.

Trigger out configuration

Value	The Trigger Out BNC will	Built-in command
fixhigh	Always be high	`cf trigout=fixhigh`
fixlow	Always be low	`cf trigout=fixlow`
monhigh	Go high when the processor is running in background (Default)	`cf trigout=monhigh`
monlow	Go low when the processor is running in background	`cf trigout=monlow`

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 analysis system must be used instead.

Break in configuration

Value	Meaning	Built-in command
off	Inputs to the Break In BNC will be ignored.	`cf breakin=off`
rising	The emulation probe will cause a break on a rising edge. (Default)	`cf breakin=rising`
falling	The emulation probe will cause a break on a falling edge.	`cf breakin=falling`

There is a delay of about 400 usec between receiving the edge and stopping the processor.

To configure the JTAG clock speed (communication speed)

The HP 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" paragraph (page 60), setting speed to a slower setting may enable the module to work.

Processor clock speed configuration

Value	Processor clock (TCK) is at least	Built-in command
1	10 MHz (default)	`cf speed=1`
2	5 MHz	`cf speed=2`
3	2.5 MHz	`cf speed=3`
4	1.25 MHz	`cf speed=4`
5	625 kHz	`cf speed=5`
6	312 kHz	`cf speed=6`
7	156 kHz	`cf speed=7`

To configure reset operation

The reset configuration item controls what kind of reset is performed and what state the processor will be in after the reset.

Reset configuration

Value	Effect of a reset from the emulator	Built-in command
runrom	Reset the processor and cause it to start running user code at address FFF00100H.(Default)	`cf reset=runrom`
rom	Reset the processor and cause it to stop at address 0FFF00100H.	`cf reset=rom`
runram	Reset the processor and cause it to start running user code at address 00000100H.	`cf reset=runram`
ram	Reset the processor and cause it to stop at address 00000100H.	`cf reset=ram`
jtag	Just reset the JTAG interface on the processor. The processor itself will not be reset. This may help in some cases where communications are lost, however all the other reset settings reset the JTAG interface as part of the reset sequence so this setting will only rarely be useful.	`cf reset=jtag`

To set memory read delays

The memory read delay setting delays the number of microseconds specified during memory reads. It is provided for accessing slow devices like memory mapped IO.

- To set the memory read delay using the built-in terminal interface, use the `cf mrddel=<`*delay in usec*`>` command.

  The <*delay in usec*> must be in the range 0-10000000. This should be set to the smallest number possible for best performance since it delays all reads by the number of microseconds specified.

  Default: cf mrddel=0

To set memory write delays

The memory write delay setting delays memory writes by the number of microseconds specified. It is provided for accessing slow devices like memory mapped IO.

- To set the memory write delay using the built-in terminal interface, use the `cf mwrdel=<`*delay in usec*`>` command.

  The <*delay in usec*> must be in the range 0-10000000. This should be set to the smallest number possible for best performance.

  Default: cf mwrdel=0

To generate parity bits on memory operations

The PowerPC processor generates parity bits on both address and data lines when running user code. When used in debug mode these bits must be generated separately slowing down memory operations. Since memory operations on the PowerPC are slow as it is and many target systems do not check parity, parity is only generated if requested.

Parity configuration

Value	Emulator configured for	Built-in command
off	Do not generate the parity bits for memory operations from the emulator. This provides better performance, but will not work correctly when accessing devices that check the parity bits.(Default)	`cf parity=off`
on	Generate the parity bits for memory operations. Currently, only parity bits for the memory data lines are generated. Parity bits on the address lines are not. This may change in future firmware versions.	`cf parity=on`

To configure the memory read operation

The memory read operation configuration entry defines how the memory and cache interact during a memory read operation. If both instruction and data caches are turned off (bits ICE and DCE in the register HID0 are zero), this configuration setting has no effect and a memory read will always return the contents of physical memory.

Memory read configuration

Value	emulator configured for	Built-in command
mm	A memory read from an address that is valid in either the data or instruction cache will return the contents of the cache. Memory reads from addresses not valid in either cache will return the contents of the physical memory.(Default)	`cf mrdop=mm`
phys	A memory read will always return the contents of physical memory.	`cf mrdop=phys`

Using the mrdop=phys setting with the cache enabled may show data that is no longer valid. Use this setting only for solving cache problems where you really need to see the contents of physical memory. For general operation, the "mm" setting should always be used.

The instruction cache in PPC740 and PPC750 is encoded. The emulator will decode the content of the instruction cache before displaying it. However, the emulator will only decode valid instructions. Invalid instructions in the cache will be displayed in coded form, which might not match the content of memory.

To configure data memory write operations

Although the PowerPC processor has one contiguous physical memory address space that can hold both data and instructions, it has separate caches for instructions and data. These separate caches must be considered in order to keep the caches and memory coherent during memory write operations. These settings are only used for memory write operations. Code download always writes to physical memory and disables any cache entries containing addresses written for improved performance. Some host interfaces use the code download mode for all memory write operations so this setting may or may not have any effect on your debugger.

Only the memory write command allows specifying instruction or data memory operations. This may not be provided by your debugger interface. If not specified, memory write operations are always instruction memory.

If the data cache is disabled, a data memory write will always write to physical memory and this configuration setting is ignored.

Memory write configuration

Value	emulator configured for	Built-in command
mm	Data writes to addresses that are valid in the data cache will write the value only to the cache and mark the cache line modified as "dirty", which will indicate to the cpu that the cache line must be written to memory. A data write that is not valid in the data cache will only be written to physical memory.(Default)	`cf dmwrop=mm`
thru	A data memory write to an address that is valid in the data cache will write to both cache and physical memory. If the address is not valid in the cache, only physical memory will be modified.	`cf dmwrop=thru`
bypass	A data memory write will only be written to physical memory, ignoring the cache.	`cf dmwrop=bypass`

The **cf dmwrop=bypass** setting should be used with extreme caution because dirty cache entries may be written by the processor over the new data value written to memory by the emulator.

To configure instruction memory write operations

Although the PowerPC processor has one contiguous physical memory address space that can hold both data and instructions, it has separate caches for instructions and data. These separate caches must be considered in order to keep the caches and memory coherent during memory write operations. Code download always writes to physical memory and disables any cache entries containing addresses written for improved performance. Some host interfaces use the code download mode for all memory write operations so this setting may or may not have any effect on your debugger.

Only the memory write command allows specifying instruction or data memory operations. Access to this may not be provided by your debugger interface. If not specified, memory write operations are always instruction memory.

If the instruction and data caches are both disabled, an instruction memory write will always write to physical memory and this configuration setting is ignored. If the instruction cache is disabled, instruction memory writes will always write to physical memory and the data cache will be either updated or bypassed, depending on this configuration setting.

This configuration setting controls the behavior of both caches when doing instruction memory writes so that instruction memory writes can be used for all memory operations, if desired.

Instruction memory write configuration

Value	Emulator configured for	Built-in command
upd_dcb	This stands for instruction cache update, data cache bypass. An instruction memory write to an address that is valid in the instruction cache will write the value to both the instruction cache and memory. The data cache will be bypassed even if the address is valid in the data cache.	`cf imwrop=upd_dcb`
upd_dcu	This stands for update instruction cache and update data cache. An instruction memory write to an address that is valid in both caches will write the value to both caches and physical memory. (Default)	`cf imwrop=upd_dcu`

Value	Emulator configured for	Built-in command
inv_dcb	This stands for instruction cache invalidate and data cache bypass. An instruction memory write will invalidate the instruction cache if valid and write only to physical memory. The data cache is not modified even if valid.	`imwrop=inv_dcb`
inv_dcu	This stands for instruction cache invalidate and data cache update. An instruction memory write will invalidate the instruction cache if valid and write to physical memory. The data cache will also be updated if the address is valid in the data cache	`imwrop=inv_dcu`

Setting imwrop to upd_dcb or inv_dcb should be used with caution since dirty cache entrys in the data cache may overwrite the memory just modified by the HP emulator.

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 "Troubleshooting the Emulator" on page 133 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.

6

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 HP 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.

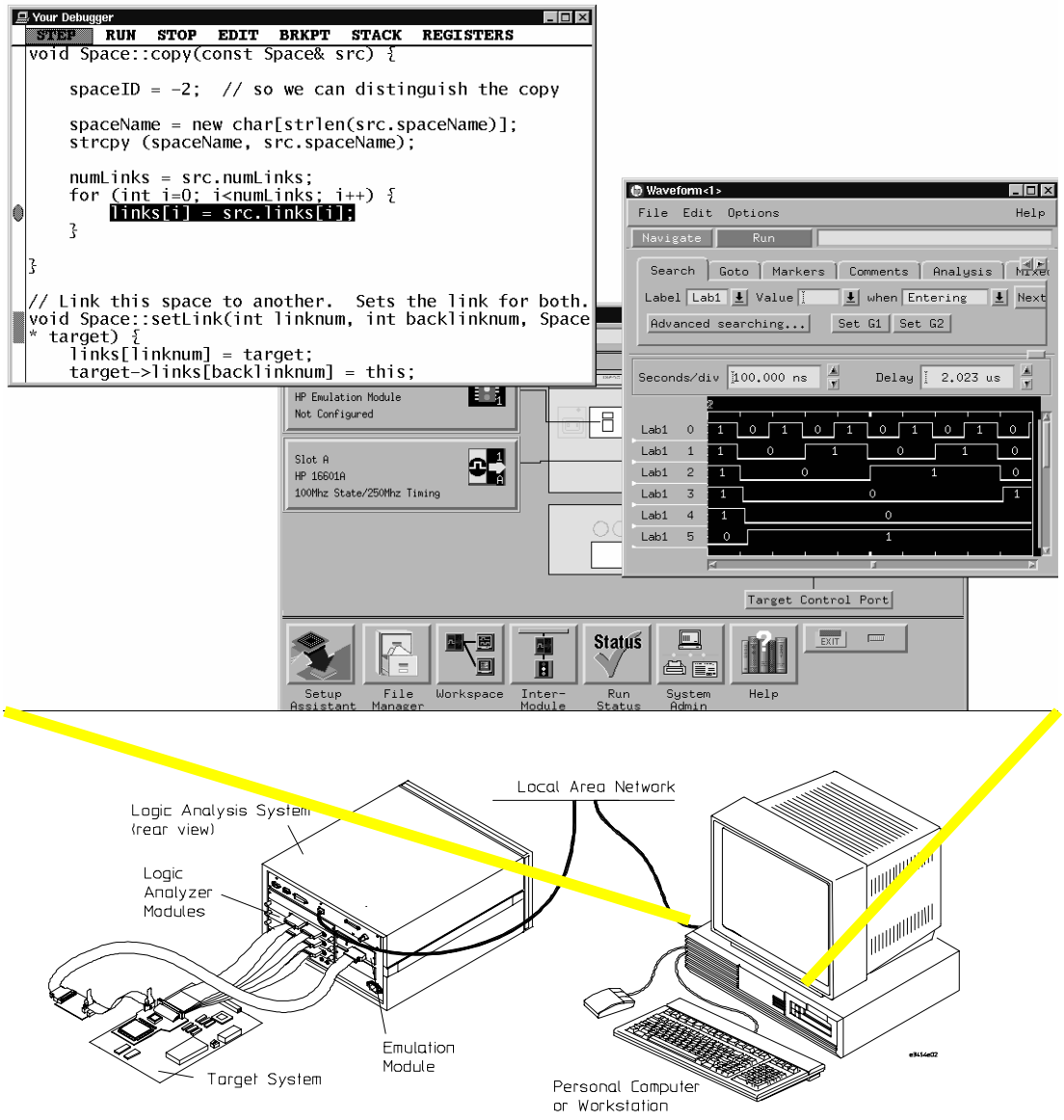
Using a debugger to connect the emulation module allows the entire design team to have a consistent interface from software development to hardware/software integration.

Debugger interfaces must be ordered directly from the debugger vendor.

Compatibility with other logic analysis system tools

If you design logic analysis connections into your target system, you can connect 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.

Here is an example of what the display on your PC or workstation might look like:



Minimum requirements

To use a debugger with the emulator, you will need:

- A debugger which is compatible with the emulator
- A LAN connection between the PC or workstation that is running the debugger, and the HP 16600A or HP 16700A logic analysis system
- 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 HP emulation module or HP 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 (page 44).
- Connect the logic analysis system to the LAN (page 88).
- Export the logic analysis system's display to your PC or workstation (page 91).
- Configure the emulator (page 67).
- 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

Port number	Use for
Debugger connections	
6470	Slot 1 (First emulation module in an HP 16600A/700A-series logic analysis system)
6474	Slot 2 (Second emulation module in an HP 16700A-series system)
6478	Slot 3 (Third emulation module in an expansion frame)
6482	Slot 4 (Fourth emulation module in an expansion frame)
Telnet connections	
23	Emulation probe (standard telnet port number)
6472	Slot 1 (First emulation module)
6476	Slot 2 (Second emulation module)
6480	Slot 3 (Third emulation module)
6484	Slot 4 (Fourth emulation module)

Write the information here for future reference:

IP Address of Logic Analysis System _____

LAN Name of Logic Analysis System _____

Gateway Address _____

Port Number of Emulation Module _____

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 emulation module). 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, and 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 emulation module.
4. For an emulation probe, ccycle power on the emulation probe.

To change the port number using built-in commands:

1. Telnet to the IP address of the emulation probe.

   For example, on a UNIX system, enter "telnet <*IP_address*>".
2. Enter the "lan -p" command:

   lan -p <*new port number*>
3. For an emulation probe, cycle power on the emulation probe.

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.

"Troubleshooting," page 142, if you have problems verifying LAN communication.

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 windows 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:

   `xhost +<IP_address>`

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

   `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**.

Example

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

```
$ xhost +mylogic
$ telnet mylogic
Trying...
Connected to mylogic.mycompany.com.
Escape character is '^]'.
Local flow control on
Telnet TERMINAL-SPEED option ON

HP Logic Analysis System

Please Log in as: hplogic [displayname:0]

login: hplogic
Connection closed by foreign host.
$
```

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**.

To enable or disable processor caches

The Power PC 7xx processors have instruction and data caches. Debugging using an third party debugger will have the greatest performance if the caches are disabled during debugging. There are three ways to disable the caches prior to a debug session:

- Set bits 16 and 17 of register HID0 to zero (bit 0 being the MSB). This will turn off I and D caches. Also turn off the L2 Cache, by setting L2CR to zero.

  Ensure that your startup code does not reset the HID0 or L2CR registers because this could re-enable the caches.

- Issue the following probe commands:

  "cf reset=rom"
  "rst" ("rst" will turn off all caches)

  Ensure that your startup code does not reset the HID0 register after the "rst" command because this could re-enable the caches.

- Keep the caches enabled but tell the HP emulator to bypass them. To do this, issue the probe commands:

  "cf mrdop=phys" (so only physical memory is read)
  "cf dmwrop=bypass" (to bypass the updating of the data cache)
  reference all addresses with the @dmem modifier.

  Example:

  ```
  M> cf mrdop=phys
  M> cf dmwrop=bypass
  M> m -d4 -a4 0..                    (this will read physical memory only)
  M> m -d4 -a4 0@dmem=12345678 (this will write physical memory only)
  ```

  When caches are bypassed, all memory accesses occur out of physical memory and the cache information is ignored. **This means that cache coherency is not maintained.**

> If cache handling is not modified using one of the above three methods, execution with the third party debugger may be slower due to the HP emulator making sure the cache information stays coherent with physical memory.

7

Using Logic Analysis and the Emulation Module Together

Using Logic Analysis and the Emulation Module Together

This chapter describes how to use an inverse assembler, an emulation module, and other features of your HP 16600A or HP 16700A logic analysis system to gain insight into your target system.

This chapter assumes you designed logic analysis connections into your target system so that you can connect the logic analyzer to perform state analysis of processor activity.

What are some of the tools I can use?

You can use a combination of all of the following tools to control and measure the behavior of your target system:

- The HP E2489A Inverse Assembler, to acquire an assembly language view of the data from the processor bus while it was running full-speed.
- Your emulation module, to control the execution of your target processor and to examine the state of the processor and of the target system.
- The Emulation Control Interface, to control and configure the emulation module, and to display or change target registers and memory.
- Display tools including the Listing tool, Chart tool, and System Performance Analyzer tool to make sense of the data collected by the logic analysis system.
- Your debugger, to control your target system using the emulation module. Do not use the debugger at the same time as the Emulation Control Interface.
- The HP B4620B Source Correlation Tool Set, to relate the analysis trace to your high-level source code.

Which assembly-level listing should I use?

Several windows display assembly language instructions. Be careful to use the correct window for your purposes:

- The Listing tool shows processor states that were captured during a "Run" of the logic analyzer. Those states are disassembled and displayed in the Listing window.
- The Emulation Control Interface shows the disassembled contents of a section of memory in the Memory Disassembly window.
- Your debugger shows your program as it was actually assembled, and (if it supports the emulation module) shows which line of assembly code corresponds to the value of the program counter on your target system.

Which source-level listing should I use?

Different tools display source code for different uses:

- The Source Viewer window allows you to follow how the processor executed code as the analyzer captured a trace. Use the Source Viewer to set analyzer triggers. The Source Viewer window is available only if you have licensed the HP B4620B Source Correlation Tool Set.
- Your debugger shows which line of code corresponds to the current value of the program counter on your target system. Use your debugger to set breakpoints.

Where can I find practical examples of measurements?

The Measurement Examples section in the online help contains examples of measurements that will save you time throughout the phases of system development: hardware turn-on, firmware development, software development, and system integration.

A few of the many things you can learn from the measurement examples are:

- How to find glitches.
- How to find NULL pointer de-references.
- How to profile system performance.

To find the measurement examples, click the Help icon in the logic analysis system window, and then click "Measurement Examples."

Triggering the Emulation Module from the Analyzer

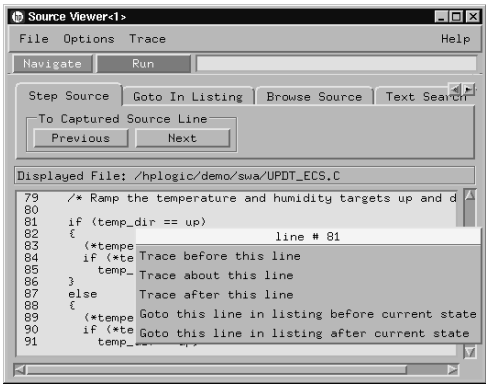
You can trigger the emulation module from the logic analyzer using either the Source Viewer window or the Intermodule window. If you are using the HP B4620B Source Correlation Tool Set, using the Source Viewer window is the easiest method.

To stop the processor when the logic analyzer triggers on a line of source code (Source Viewer window)

If you have the HP B4620B Source Correlation Tool Set, you can easily stop the processor when a particular line of code is reached.

1. In the Source window, click the line of source code where you want to set the trigger, and then select **Trace about this line**.

   The logic analyzer trigger is now set.



2. Select **Trace→Enable - Break Emulator On Trigger.**

   The emulation module is now set to halt the processor after receiving a trigger from the logic analyzer.

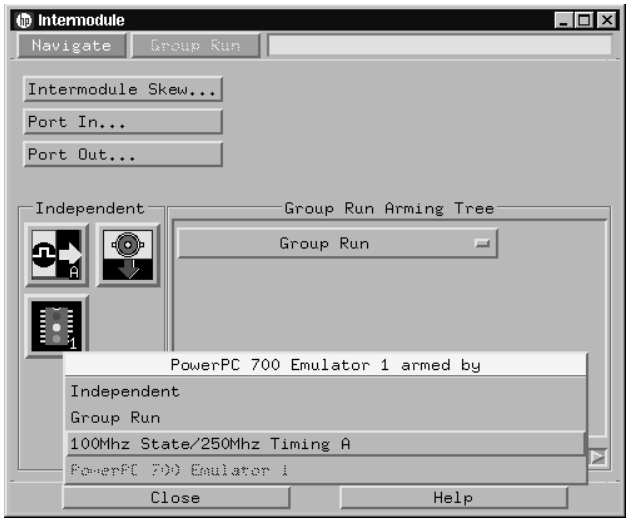
   To disable the processor stop on trigger, select **Trace→Disable - Break Emulator On Trigger.**

3 Click **Group Run** in the Source window (or other logic analyzer window).
4 If your target system is not already running, click **Group Run** in the emulation Run Control window to start your target.

To stop the processor when the logic analyzer triggers (Intermodule window)

Use the Intermodule window if you do not have the HP B4620B Source Correlation Tool Set or if you need to use a more sophisticated trigger than is possible in the Source Viewer window.

1 Create a logic analyzer trigger.
2 In the Intermodule window, click the emulation module icon, and then select the analyzer that is intended to trigger it.



The emulation module is now set to stop the processor when the logic analyzer triggers.

3 Click **Run** in the Source window (or other logic analyzer window).
4 If your target system is not already running, click **Group Run** in the emulation Run Control window to start your target.

See Also

See the online help for your logic analysis system for more information on setting triggers.

To minimize the "skid" effect

There is a finite amount of time between when the logic analyzer triggers, and when the processor actually stops. During this time, the processor will continue to execute instructions. This latency is referred to as the skid effect.

To minimize the skid effect:

1. In the Emulation Control Interface, open the Configuration window.
2. Set processor clock speed to the maximum value your target can support.

   The amount of skid will depend on the processor's execution speed and whether code is executing from the cache. See page 74 for information on how to configure the clock speed.

To stop the analyzer and view a measurement

- To view an analysis measurement, you may have to click **Stop** after the trigger occurs.

  When the target processor stops, it may cause the analyzer qualified clock to stop. Therefore, most intermodule measurements will have to be stopped to see the measurement.

Example

An intermodule measurement has been set up where the analyzer is triggering the emulation module. The following sequence could occur:

1. The analyzer triggers.
2. The trigger ("Break In") is sent to the emulation module.
3. The emulation module stops the user program which is running on the target processor. The processor enters a background debug monitor.
4. Because the processor has stopped, the analyzer stops receiving a qualified clock signal.
5. If the trigger position is "End", the measurement will be completed.

   If the trigger position is not "End", the analyzer may continue waiting for more states.
6. The user clicks **Stop** in a logic analyzer window, which tells the logic analyzer to stop waiting, and to display the trace.

Tracing until the processor halts

If you are using a state analyzer, you can begin a trace, run the processor, and then manually end the trace when the processor has halted.

To halt the processor, you can set a breakpoint using the Emulation Control Interface or a debugger.

Some possible uses for this measurement are:

- To store and display processor bus activity leading up to a system crash.
- To capture processor activity before a breakpoint.
- To determine why a function is being called. To do this, you could set a breakpoint at the start of the function and use this measurement to see how the function is getting called.

This kind of measurement is easier than setting up an intermodule measurement trigger.

To capture a trace before the processor halts

1. Set the logic analyzer to trigger on **nostate**.
2. Set the trigger point (position) to **End**.
3. In a logic analyzer window, click **Run**.
4. In the Emulation Control Interface or debugger, click **Run**.
5. When the emulation module halts, click **Stop** in the logic analyzer window to complete the measurement.

   This is the recommended method to do state analysis of the processor bus when the processor halts.

   If you need to capture the interaction of another bus when the processor halts, or you need to make a timing or oscilloscope measurement, you will need to trigger the logic analyzer from the emulation module (described in the next section).

Triggering the Logic Analyzer from the Emulation Module

You can create an intermodule measurement that will allow the emulation module to trigger another module such as a timing analyzer or oscilloscope.

If you are only using a state analyzer to capture the processor bus, it will be much simpler to use "Tracing until processor halts" as described on page 102.

Before you trigger a logic analyzer (or another module) from the emulation module, you should understand a few things about the emulation module trigger:

The emulation module trigger signal

The trigger signal coming from the emulation module is an "In Background Debug Monitor" ("In Monitor") signal. This may cause confusion because a variety of conditions can cause this signal and falsely trigger your analyzer.

The "In Monitor" trigger signal can be caused by:

- The most common method to generate the signal is to click **Run** and then click **Break** in the Emulation Control Interface. Going from "Run" (Running User Program) to "Break" ("In Monitor") generates the trigger signal.
- Another method to generate the "In Monitor" signal is to click **Reset** and then click **Break**. Going from the reset state of the processor to the "In Monitor" state will generate the signal.
- In addition, an "In Monitor" signal is generated any time a debugger or other user interface reads a register, reads memory, sets breakpoints or steps. Care must be taken to not falsely trigger the logic analyzers listening to the "In Monitor" signal.

Group Run

The intermodule bus signals can still be active even without a Group Run.

The following setups can operate independently of Group Run:

- Port In connected to an emulation module
- Emulation modules connected in series
- Emulation module connected to Port Out

Here are some examples:

- If "Group Run" is armed from "Port In" and an emulation module is connected to Group Run, any "Port In" signal will cause the emulation module to go into monitor. The Group Run button does not have to be pressed for this to operate.
- If two emulation modules are connected together so that one triggers another, the first one going into monitor will cause the second one to go into monitor.
- If an emulation module is connected to Port Out, the state of the emulation module will be sent out the Port Out without regard to "Group Run".

The current emulation module state (Running or In Monitor) should be monitored closely when they are part of a Group Run measurement so that valid measurements are obtained.

Group Run into an emulation module does not mean that the Group Run will Run the emulation module.

The emulation module Run, Break, Step, and Reset are independent of the Group Run of the Analyzers.

For example, suppose you have the following IMB measurement setup:

Clicking the **Group Run** button (at the very top of the Intermodule window or a logic analyzer window) will start the analyzer running. The analyzer will then wait for an arm signal. Now when the emulation module transitions into "Monitor" from "Running" (or from "Reset"), it will send the arm signal to the analyzer. If the emulation module is "In Monitor" when you click **Group Run**, you will then have to go to the emulation module or your debugger interface and manually start it running.

Debuggers can cause triggers

Emulation module user interfaces may introduce additional states into your analysis measurement and in some cases falsely trigger your analysis measurement.

When a debugger causes your target to break into monitor, it will typically read memory around the program stack and around the current program counter. This will generate additional states which appear in the listing.

You can often distinguish these additional states because the time tags will be in the μs and ms range. You can use the time tag information to determine when the processor went into monitor. Typically the

time between states will be in nanoseconds while the processor is running and will be in μs or ms when the debugger has halted the processor and is reading memory.

Note also that some debugger commands may cause the processor to break temporarily to read registers and memory. These states that the debugger introduces will also show up in your trace listing.

If you define a trigger on some state and the debugger happens to read the same state, you may falsely trigger your analyzer measurement. In summary, when you are making an analysis measurement, be aware that the debugger could be impacting your measurement.

To trigger the analyzer when the processor halts

Remember: if you are only using a state analyzer to capture the processor bus, it will be much simpler to use "Tracing until processor halts" as described on page 102.

1. Set the logic analyzer to trigger on **anystate**.
2. Set the trigger point to **center** or **end**.
3. In the Intermodule window, click the logic analyzer you want to trigger and select the emulation module.

   The logic analyzer is now set to trigger on a processor halt.
4. Click **Group Run** to start the analyzer(s).
5. Click **Run** in the Emulation Control Interface or use your debugger to start the target processor running.

   Clicking **Group Run** will *not* start the emulation module. The emulation module run, break, step, reset are independent of the Group Run of the analyzers.
6. Wait for the Run Control window in the Emulation Control Interface or the status display in your debugger to show that the processor has stopped.

   The logic analyzer will store states until the processor stops, but may continue running.

   You may or may not see a "slow clock" error message. In fact, if you are using a state analyzer on the processor bus, the status may never change upon receiving the emulation module trigger (analysis arm). This occurs because the qualified processor clock needed to switch the state analyzer to the next state is stopped. For example, the state analyzer before the arm event may have a status of "Occurrences Remaining in Level 1: 1" and after the arm event it may have the same status, "Occurrences Remaining in Level 1: 1"
7. If necessary, in the logic analyzer window, click **Stop** to complete the measurement.

   If you are using a timing analyzer or oscilloscope, the measurement should complete automatically when the processor halts. If you are using a state logic analyzer, click **Stop** if needed to complete the measurement.

To trigger the analyzer when the processor reaches a breakpoint

This measurement is exactly like the one on the previous page, but with the one additional complexity of setting breakpoints. Be aware that setting breakpoints may cause a false trigger and that the breakpoints set may not be valid after a reset.

Remember: if you are only using a state analyzer to capture the processor bus, it will be much simpler to use "Tracing until processor halts" as described on page 102.

1. Set the logic analyzer to trigger on **anystate**.
2. Set the trigger point to **center** or **end**.
3. In the Intermodule window, click the logic analyzer you want to trigger and select the emulation module.

   The logic analyzer is now set to trigger on a processor halt.
4. Set the breakpoint.

   If you are going to run the emulation module from Reset, you must do a **Reset** followed by **Break** to properly set the breakpoints. The Reset will clear all on-chip hardware breakpoint registers. The Break command will then reinitialize the breakpoint registers. If you are using software breakpoints which insert an illegal instruction into your program at the breakpoint location, you will not need to do the Reset, Break sequence. Instead you must take care to properly insert your software breakpoint in your RAM program location.
5. Click **Group Run** to start the analyzer(s).
6. Click **Run** in the Emulation Control Interface or use your debugger to start the target processor running.

   Clicking **Group Run** will *not* start the emulation module. The emulation module run, break, step, reset are independent of the Group Run of the analyzers.
7. Wait for the Run Control window in the Emulation Control Interface or the status display in your debugger to show that the processor has stopped.

   The logic analyzer will store states until the processor stops, but may continue running.

You may or may not see a "slow clock" error message. In fact, if you are using a state analyzer on the processor bus, the status may never change upon receiving the emulation module trigger (analysis arm). This occurs because the qualified processor clock needed to switch the state analyzer to the next state is stopped. For example, the state analyzer before the arm event may have a status of "Occurrences Remaining in Level 1: 1" and after the arm event it may have the same status, "Occurrences Remaining in Level 1: 1"

8 If necessary, in the logic analyzer window, click **Stop** to complete the measurement.

If you are using a timing analyzer or oscilloscope, the measurement should complete automatically when the processor halts. If you are using a state logic analyzer, click **Stop** if needed to complete the measurement.

8

Updating Firmware

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 that 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 HP.

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.hp.com/go/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 HP 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.

If there is a power failure during a firmware update

If there is a power glitch during a firmware update, some bits may be lost during the download process, possibly resulting in an emulation probe that will not boot up. To correct a partial firmware update:

1. **Set switch S4 to OFF/OPEN; then cycle power. This tells the emulation probe to ignore everything in the Flash EPROM except the boot code.**
2. **Repeat the firmware update process.**
3. **Set switch S4 to ON/CLOSED; then cycle power. This restores the emulation probe to its normal mode.**

Emulation Module Firmware

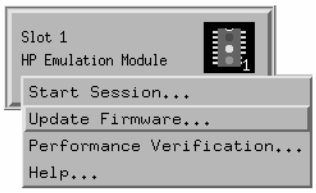
Always update firmware by installing a processor support package. This will ensure that the version of the Emulation Control Interface software is compatible with the version of the emulator firmware.

To display current firmware version information

- 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.
4. In the system window, click the emulation module and select **Update Firmware...**.



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 49 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 HP 16600A and HP 16700A-series logic analysis systems.

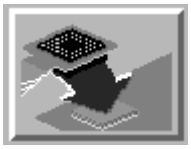
This menu-driven tool will guide you through the connection procedures for connecting the logic analyzer to an analysis probe, an emulation module, or other supported equipment. It will also guide you through connecting an analysis probe to the target system.

Do not use the Setup Assistant to connect an emulation probe if you already have an emulation module installed.

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 49 for instructions on how to install the processor support package from the CD-ROM.

Specifications and Characteristics

Operating characteristics

The following operating characteristics are not specifications, but are typical operating characteristics for the HP 16610A emulation module, emulation probe, and HP E2498A PowerPC Inverse Assembler.

Operating Characteristics	
Microprocessor Compatibility	PowerPC 740 and PowerPC 750
Microprocessor Clock Speed	70 MHz for the HP 1670A and HP 16554A Logic Analyzers 100 MHz for the HP 16600A, HP 16601A, HP 16550A, HP 16554D, HP 16555/56/57, HP 1660, and HP 1670D Logic Analyzers
Logic Analyzers Supported	HP 1660A/AS/C/CP/CS, HP 1670A/D, HP 16550A (two cards), HP 16554A/55A/56A (two or three cards), HP 16555D/56D/57D (two or three cards), HP 16600A, HP 16601A.
Probes Required	Eight 16-channel probes are required for disassembly. Two additional 16-channel probes are available.
Signal Line Loading	Typically 100 kOhm plus 10 pf.
Setup/Hold Requirement	For all signals, the logic analyzers require a minimum combined setup/hold window. For the HP 16600-series logic analysis system, the combined setup/hold must be at least 4.5 ns (such as 0/4.5, 1.0/3.5. etc). For all other logic analyzers, the combined window must be at least 3.5 ns.

Emulation Probe Electrical Characteristics

BNC, labeled TRIGGER OUT

Output Drive Logic high level with 50-ohm load >= 2.0 V. Logic low level with 50-ohm load <= 0.4 V. Output function is selectable. Refer to the configuration chapter.

BNC, labeled BREAK IN

Input Edge-triggered TTL level input (active high), 20 pf, with 2K ohms to ground in parallel. Maximum input: 5 V above $V_{CC;}$ 5 V below ground. Input function is selectable. Refer to the configuration chapter. Refer to Online Help for more information. The BNC introduces approximately 2.5 ms skid after break-in at 25 MHz.

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 emulation probe 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

Emulation module and emulation probe electrical characteristics

Maximum Ratings

Characteristics for the PowerPC 7xx emulation module and emulation probe	Symbol	Min	Max
TDO, $\overline{\text{CKSTP\_OUT}}$	V_{ih}	2.0 V	5.5 V
	V_{il}		0.8 V
	I_i		±1 µA
	C_{in}		15 pF
TDI, TCK, TMS, $\overline{\text{TRST}}$ [1]	V_{oh} @ I_{oh} = -32 mA	2.0 V	2.8 V
	V_{ol} @ I_{ol} = 64 mA; V_{CC}=4.5V		0.55 V
TDI, TMS, $\overline{\text{TRST}}$	C_o		25 pF
TCK	C_o		45 pF
+3.3V Power Sense [2]	V_{ih}	2.0 V	5.3 V
	V_{il}	-0.3 V	0.8 V
$\overline{\text{SRESET}}$, $\overline{\text{HRESET}}$ [3]	V_{ol} @ I_{ol} = 12 ma		0.5 V
	C_o		25 pF
TS0 - TS6, SYSCLK	C_{in}		10 pF
	V_{ih}	2.0 V	5.5 V
	V_{il}		0.8 V
	I_i		±1 µA

1 These signals must not be actively driven by the target system when the debug port is being used.

2 Power Sense is used only to determine target powered status. The emulation module and emulation probe do not draw power from this source.

3 Open collector outputs, pulled up to a generated voltage equivalent to the Power Sense voltage with a 2.61 K pullup resistor

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

For indoor use only.

Inverse assembler—signal-to-connector mapping

The following tables show the electrical signal-to-connector mapping required by the HP E2498A Inverse Assembler Software.

If you are using the 2x19 AMP Mictor connectors, you must allocate the odd and even pods according to the tables in this section. (Note that the odd pods have even pin numbers, and the even pods have odd pin numbers.) The connectors and the high-density termination cables are keyed so they will fit together only the correct way.

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J1 Odd

2x19 pin	LA bit	signal name	analyzer label
6	CLK1	SYSCLK	SYSCLK
8	15	A16	ADDR
10	14	A17	ADDR
12	13	A18	ADDR
14	12	A19	ADDR
16	11	A20	ADDR
18	10	A21	ADDR
20	9	A22	ADDR
22	8	A23	ADDR
24	7	A24	ADDR
26	6	A25	ADDR
28	5	A26	ADDR
30	4	A27	ADDR
32	3	A28	ADDR
34	2	A29	ADDR
36	1	A30	ADDR
38	0	A31	ADDR

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J1 Even

2x19 pin	LA bit	signal name	analyzer label
5	CLK1	--	
7	15	A0	ADDR
9	14	A1	ADDR
11	13	A2	ADDR
13	12	A3	ADDR
15	11	A4	ADDR
17	10	A5	ADDR
19	9	A6	ADDR
21	8	A7	ADDR
23	7	A8	ADDR
25	6	A9	ADDR
27	5	A10	ADDR
29	4	A11	ADDR
31	3	A12	ADDR
33	2	A13	ADDR
35	1	A14	ADDR
37	0	A15	ADDR

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J2 Odd

2x19 pin	LA bit	signal name	analyzer labels		
6	CLK1	QACK			QACK-
8	15	HRESET	STAT		HRESET-
10	14	CKSTP_IN	STAT		CKSTP-
12	13	CKSTP_OUT	STAT		CHKOUT
14	12	BR	STAT		BR-
16	11	--			
18	10	--			
20	9	WT	STAT		WT-
22	8	CI	STAT		CI-
24	7	GBL	STAT		GBL-
26	6	DBWO	STAT		DBWO-
28	5	DBG	STAT		DBG-
30	4	BG	STAT		BG-
32	3	AACK	STAT	acks	AACK-
34	2	QREQ	STAT		QREQ-
36	1	ARTRY	STAT	acks	ARTRY-
38	0	ABB	STAT		ABB-

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J2 Even

2x19 pin	LA bit	signal name	analyzer labels		
5	CLK1	--			
7	15	TSIZ0	STAT	TSIZ	
9	14	TSIZ1	STAT	TSIZ	
11	13	TSIZ2	STAT	TSIZ	
13	12	TBST	STAT	TSIZ	TBST-
15	11	TT0	STAT	TT	Atomic
17	10	TT1	STAT	TT	R/W-
19	9	TT2	STAT	TT	Invldt
21	8	TT3	STAT	TT	A Only
23	7	TT4	STAT	TT	
25	6	INT	STAT		INT-
27	5	DRTRY	STAT	acks	DRTRY
29	4	TA	STAT	acks	TA-
31	3	TEA	STAT		TEA-
33	2	SRESET-	STAT		SRESET-
35	1	TS	STAT		TS-
37	0	DBB	STAT		DBB-

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J3 Odd

2x19 pin	LA bit	signal name	analyzer label
6	CLK1	--	
8	15	DL16	DATA_B
10	14	DL17	DATA_B
12	13	DL18	DATA_B
14	12	DL19	DATA_B
16	11	DL20	DATA_B
18	10	DL21	DATA_B
20	9	DL22	DATA_B
22	8	DL23	DATA_B
24	7	DL24	DATA_B
26	6	DL25	DATA_B
28	5	DL26	DATA_B
30	4	DL27	DATA_B
32	3	DL28	DATA_B
34	2	DL29	DATA_B
36	1	DL30	DATA_B
38	0	DL31	DATA_B

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J3 Even

2x19 pin	LA bit	signal name	analyzer label
5	CLK1	DBDIS	DBDIS-
7	15	DL0	DATA_B
9	14	DL1	DATA_B
11	13	DL2	DATA_B
13	12	DL3	DATA_B
15	11	DL4	DATA_B
17	10	DL5	DATA_B
19	9	DL6	DATA_B
21	8	DL7	DATA_B
23	7	DL8	DATA_B
25	6	DL9	DATA_B
27	5	DL10	DATA_B
29	4	DL11	DATA_B
31	3	DL12	DATA_B
33	2	DL13	DATA_B
35	1	DL14	DATA_B
37	0	DL15	DATA_B

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J4 Odd

2x19 pin	LA bit	signal name	analyzer label
6	CLK1	--	
8	15	DH16	DATA
10	14	DH17	DATA
12	13	DH18	DATA
14	12	DH19	DATA
16	11	DH20	DATA
18	10	DH21	DATA
20	9	DH22	DATA
22	8	DH23	DATA
24	7	DH24	DATA
26	6	DH25	DATA
28	5	DH26	DATA
30	4	DH27	DATA
32	3	DH28	DATA
34	2	DH29	DATA
36	1	DH30	DATA
38	0	DH31	DATA

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J4 Even

2x19 pin	LA bit	signal name	analyzer label
5	CLK1	--	
7	15	DH0	DATA
9	14	DH1	DATA
11	13	DH2	DATA
13	12	DH3	DATA
15	11	DH4	DATA
17	10	DH5	DATA
19	9	DH6	DATA
21	8	DH7	DATA
23	7	DH8	DATA
25	6	DH9	DATA
27	5	DH10	DATA
29	4	DH11	DATA
31	3	DH12	DATA
33	2	DH13	DATA
35	1	DH14	DATA
37	0	DH15	DATA

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J5 Odd

2x19 pin	LA bit	signal name	analyzer labels	
6	CLK1	--		
8	15	AP0	AP	
10	14	AP1	AP	
12	13	AP2	AP	
14	12	AP3	AP	
16	11	MCP		MCP-
18	10	SMI		SMI-
20	9	--		
22	8	--		
24	7	--		
26	6	--		
28	5	--		
30	4	LSSDMODE		LSSDMO
32	3	PLLCF0	PLLCFG	
34	2	PLLCF1	PLLCFG	
36	1	PLLCF2	PLLCFG	
38	0	PLLCF3	PLLCFG	

PowerPC 740/750 Logic Analyzer Interface Signal List - Connector J5 Even

2x19 pin	LA bit	signal name	analyzer labels	
5	CLK1	--		
7	15	L1TSTCLK		L1Tclk
9	14	L2TSTCLK		L2Tclk
11	13	--		
13	12	--		
15	11	--		
17	10	RSRV		RSRV-
19	9	TBEN		TBEN
21	8	TBLISYNC		TBLISY
23	7	DP0	DP	
25	6	DP1	DP	
27	5	DP2	DP	
29	4	DP3	DP	
31	3	DP4	DP	
33	2	DP5	DP	
35	1	DP6	DP	
37	0	DP7	DP	

10

Troubleshooting the Emulator

Troubleshooting the Emulator

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 debugger
- The emulation module or emulation probe 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 emulation probe or emulation module
- The emulator "performance verification" tests
- The emulator’s built-in "terminal interface" commands

Troubleshooting Guide

Common problems and what to do about them

Symptom	What to do	See also
Commands from the Emulation Control Interface have no effect	Check that you are using the correct firmware.	
Commands from debugger have no effect	Use the Emulation Control Interface 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 steps for the next symptom....	pages 20, 23, 139
Emulator built-in commands do not work	**1** Check that the emulator has been properly configured for your target system.	pages 20, 23, 67
	2 Run the emulator performance verification tests.	page 160
	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."	page 60, page 147
"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.)	page 108
"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.	page 74
Some commands fail	Check the "restrict to real-time runs" configuration	page 72
Host computer reports LAN connection problems	Follow the checklist in the "If you have LAN problems" section.	page 142
Commands from the Run Control tool or debugger have no effect	Verify LAN communication.	page 37

Status Lights

Emulation Module Status Lights

The emulation module uses status lights to communicate various modes and error conditions.

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

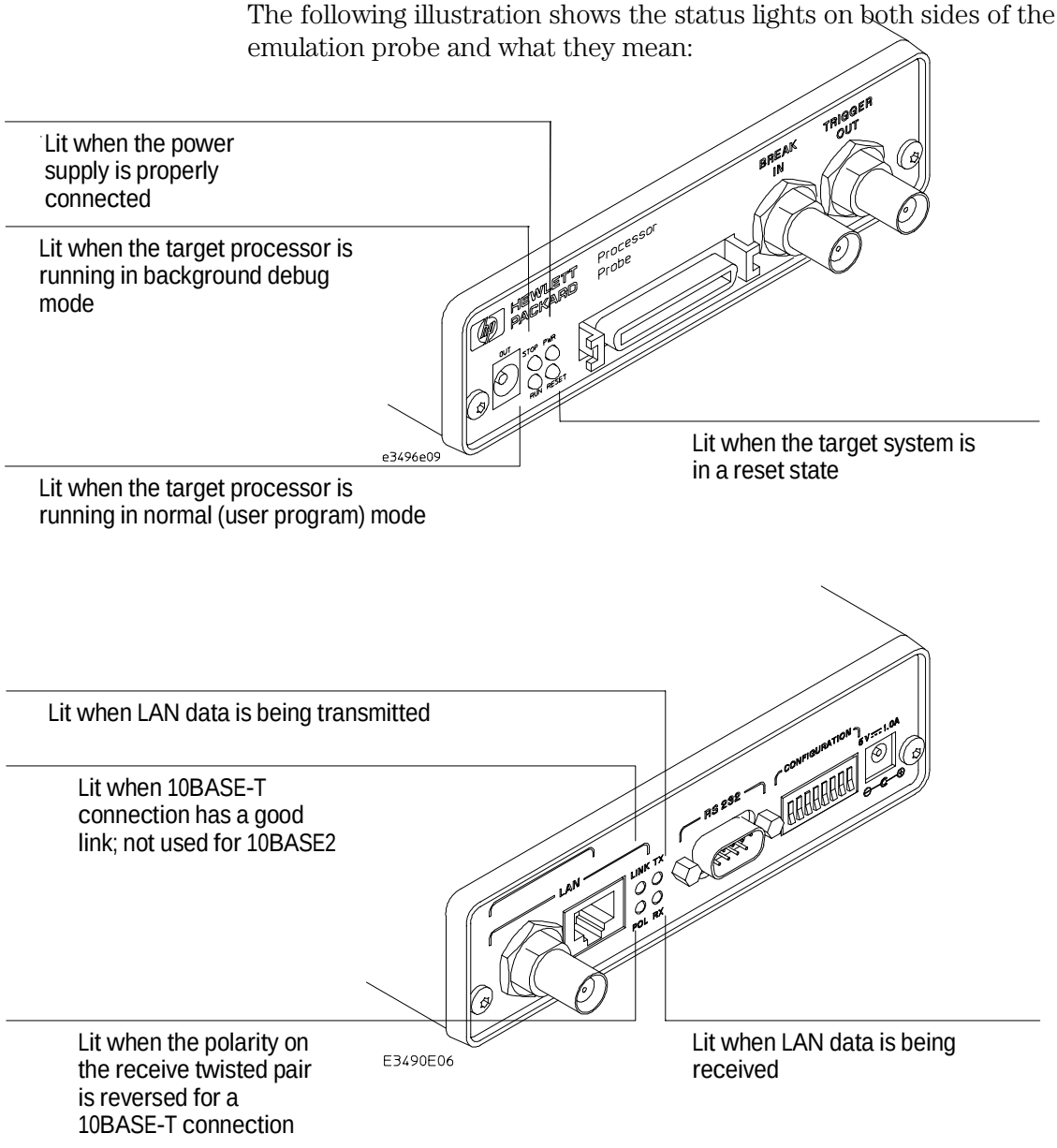
❍ = LED is off
● = LED is on
✻ = Not applicable (LED is off or on)

Power/Target Status Lights

Pwr/Target LEDs	Meaning
❍ Reset ❍ Break ❍ Run	No target system power, or emulation module is not connected to the target system
● Reset ❍ Break ❍ Run	Target system is in a reset state
❍ Reset ● Break ❍ Run	The target processor is executing in background mode
❍ Reset ❍ Break ● Run	The target processor is executing user code
❍ Reset ● Break ● Run	Only boot firmware is good (other firmware has been corrupted)

Emulation Probe Status lights

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



The emulation probe communicates various modes and error conditions via the status lights. The meanings of the status lights are shown on the previous page.

The following table gives more 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

Pwr/Target LEDs	Meaning
❍❍ ❍❍	emulation probe is not connected to power supply
❍● ❍❍	No target system power, or emulation probe is not connected to the target system
❍● ❍●	Target system is in a reset state
●● ●❍	Only boot firmware is good (other firmware has been corrupted)
●● ❍❍	The target processor is executing in Debug Mode
❍● ●❍	The target processor is executing user code

Emulator Built-in Commands

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

You can access the terminal interface using:

- A telnet (LAN) connection
- The Command Line window in the Emulation Control Interface
- A "debugger command" window in your debugger
- A serial connection (see page 38)

To telnet to the emulator

You can establish a telnet connection to the emulator if:

- A host computer and the logic analysis system are both connected to a local-area network (LAN), and
- The host computer has the telnet program (often part of the operating system or an internet software package).

To establish a telnet connection:

1. **Find out the port number of the emulator.**

   The default port number of an emulation probe or the first emulation module in an HP 16600A/700A series logic analysis system is 6472. The default port of a second emulation module in an HP 16600A-series system is 6476. The default port numbers of third and fourth emulation modules in an expansion frame are 6480 and 6484. These port numbers can be changed, but that is rarely necessary.
2. **Find out the LAN address or LAN name of the logic analysis system.**
3. **Start the telnet program.**

   If the LAN name of the logic analysis system is "test2" and you have only one emulation module installed, the command might look like this:

   ```
   telnet test2 6472
   ```

4 If you do not see a prompt, press the <Return> key a few times.
To exit from this telnet session, type <CTRL>D at the prompt.

To use the built-in commands

Here are a few commonly used built-in commands:

Useful built-in commands

b	Break—go into the background monitor state
cf	Configuration—read or write configuration options
help	Help—display online help for built-in commands
init	Initialize—init -c re-initializes everything in the emulator except for the LAN software; init -p is the equivalent of cycling power (it will break LAN connections)
lan	configure LAN address (emulation probes only)
m	Memory—read or write memory
reg	Register—read or write a register
r	Run—start running user code
rep	Repeat—repeat a command or group of commands
rst	Reset—reset the target processor (the emulator will wait for you to press the target's RESET button)
s	Step—do a low-level single step
ver	Version—display the product number and firmware version of the emulator

The prompt indicates the status of the emulator:

Emulator prompts

U	Running user program
M	Running in background monitor
c	Target is checkstopped
p	No target power
d	No target interface module connected to emulator
?	Unknown state

Examples

To set register GPR0, and then view GPR0 to verify that it was set, enter:

```
R>rst -m
M>reg GPR0=ffff
M>reg GPR0
  reg GPR0=0000ffff
```

To break execution and then step a single instruction, enter:

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

To determine what firmware version is installed in the emulator, 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 are generic commands for HP emulators and may not be available for your product.

If you are writing your own debugger, contact HP for more information.

Problems with the LAN Interface (Emulation Probe Only)

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 you have connected the emulation probe to the proper power source and the power light is lit.
- ☐ Make sure you wait for the power-on self test to complete before connecting.
- ☐ Make sure the LAN cable is connected. Watch the LAN LED's to see whether the emulation probe is detecting LAN activity. Refer to your LAN documentation for testing connectivity.
- ☐ Make sure 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, and then plug it in again to make sure the switch settings are read correctly by the emulation probe.
- ☐ Check that the Run Control Tool or debugger was configured with the correct LAN address. If the emulation probe is on a different subnet from the host computer, check that the gateway address is correct.
- ☐ Make sure the emulation probe's IP address is set up correctly. Use the RS-232 port to verify 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 from the emulation probe and reconnecting it again.

- Use a serial connection to run the LAN performance verification tests (see page 160).

If you have LAN connection problems

- If the emulation probe does not accept commands from the logic analysis system:

  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 38), 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 <new IP>' 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 port is not 6470, you must change it with the "lan -p 6470" command (unless you have deliberately set the port number to a different value because of a conflict).

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 emulation probe 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 you have connected the emulation probe to the proper power source and the power light is lit.

- ☐ Make sure 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 emulation probe 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 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 emulator)
  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, HP recommends you always turn ON the emulation probe before attempting to connect via RS-232.

Problems with the Target System

This section describes how to determine whether your target system is causing problems with the operation of the emulator.

What to check first

1 Try some basic built-in commands using the Command Line window or a 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 GPR1
  reg GPR1=00000000
M>
```

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

```
M>m 0..
  00000000   7c3043a6 7c2802a6 7c3143a6 4bf04111
  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>
```

This should display memory values starting at address 0.

```
M>s
```

This should execute one instruction at the current program counter.

If any of these commands do not 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.

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

```
M>ver
```

See Also Page 139 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.

Levels with the emulator not connected

Header Pin	Signal Name	Level
3	TDI	Low
4	$\overline{\text{TRST}}$	High
6	+POWER	V_{DD}
7	TCK	High
9	TMS	High
11	$\overline{\text{SRESET}}$	High
13	$\overline{\text{HRESET}}$	High
15	$\overline{\text{CHECKSTOP}}$	High
16	GND	Low

Levels with the emulator connected

Header Pin	Signal Name	I/O
1	TDO	Toggle with "es" command
3	TDI	Toggle with "es" command
4	$\overline{\text{TRST}}$	Low pulse with "rst" command
6	+POWER	V_{DD}
7	TCK	10+ MHz clock (default)
9	TMS	Low, pulse with "es" command
11	$\overline{\text{SRESET}}$	High, pulse low with "rst" command
13	$\overline{\text{HRESET}}$	High, pulse low with "rst" command
15	$\overline{\text{CHECKSTOP}}$	High
16	GND	Low

To interpret the initial prompt

The initial prompt can be used to diagnose several common problems. To get the most information from the prompt, follow this procedure:

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 6 on header, 3.3V (V_{DD}).

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 "c>"

Processor is checkstopped. Something caused a machine exception before the emulator connected or $\overline{\text{CHECKSTOP}}$ is being pulled or held low.

If the initial prompt is "?>" with "ERROR 171!"

A bad status code (0xXX) was received from the processor. Valid status is 0x01 or 0x05. Any other status indicates a bad scan of the instruction register. Check TCK, TDO, TDI, TMS, and $\overline{\text{TRST}}$_L signals. Check the firmware revision.

If the initial prompt is "U>"

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

3 Enter the reset command:

```
U>rst
U>
```

The "U>" prompt is a good response that indicates $\overline{\text{SRESET}}$ and $\overline{\text{HRESET}}$ are working. Continue with "If the prompt after rst is U>".

If the prompt after rst is "?>" with "ERROR 171!"

A bad status code (0xXX) was received from the processor. Valid status is 0x01. Any other status indicates bad scan of IR or failure of the reset signals. Verify TCK, TDO, TDI,TMS, and $\overline{\text{TRST}}$ are all changing state on an $\overline{\text{HRESET}}$.

If the rst command fails

Set "cf reset=rom" (no external bus cycles used in this mode). Then enter the "rst" command again:

```
*>cf reset=rom
*>rst
M>
```

You can enter these commands at any prompt, shown here as "*>".

- If the prompt is "M>" with no error messages, all scans worked. We have control as long as we don't try to run code. Continue with "If you can get to the "M>" prompt.
- If an error message is displayed, verify that $\overline{\text{HRESET}}$ and $\overline{\text{SRESET}}$ are being driven.
- If the prompt is "c>", there was bad scanning of the data scan chain. Check processor mask revision.
- If the prompt is "U>", the processor failed to stop soft or hard. Check reset lines, mask revision, processor type and firmware version.

If the prompt after rst is "U>"

The $\overline{\text{HRESET}}$ and $\overline{\text{SRESET}}$ lines are working. Continue with more tests:

4 Enter the break command:

```
U>b
M>
```

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

If you see: "!ERROR 145! Unable to soft stop - freezing the processor clocks" the processor is hard stopped. Check the mask revision, processor type, and firmware version. If all of these look good, the target may not be terminating cycles (pending external bus cycles). Successive run ("r") and step ("s") commands will fail. The processor may have fetched an invalid instruction.

Check the value of the PC (IAR):

```
M>reg PC
  reg PC=xxxxxxxx
M>
```

If the value is fff00100, the processor had a problem accessing the boot ROM and crashed during boot.

Processor and/or board level reset is required to recover from "freezing processor clocks" -- register and memory commands should still work.

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

Everything is still working correctly. Continue with more tests:

If you can get to the "M>" prompt

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

```
M>reg GPR0
  reg GPR0=xxxxxxxx
M>reg GPR0=12345678
M>reg GPR0
   reg GPR0=12345678
M>
```

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

Now enter:

```
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>
```

- Returned value is equal to the written value implies that memory is working.
- Returned value is not equal to the written value implies that memory control may not be initialized. Try to initialize by:

```
M>cf reset=runrom;rst;w 5
  #waiting for 5 seconds...
U>b
M>
```

Repeat above memory test.

6 At the "M>" prompt , check the processor's revision level:

The target must support burst cache fill from where PC is pointing.

Set the PC to a location in RAM. For example:

```
M>reg PC=100
M>
```

Now enter:

```
M>reg PVR
  reg PVR=xxxxxxxx
M>
```

The returned value is in the form VVVVRRrr where VVVV is the processor's design architecture family, and RRrr is mask revision level.

VVVV:

0008 -> 740/750

For example reg PVR=00080202 means 740/750 Mask Revision 2.2.

If you see memory-related problems

1 Set caches and translation off:

```
M>reg HID0=0
M>reg MSR=0
M>
```

If these commands fail, just try again.

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
M>
```

- If you do not see correct values written in memory, try increasing memory delay (page 75).
- If the read value is not equal to the written value, the memory controller may not be set up correctly.
- If the read value is equal to the written value, but you still suspect memory problems, the emulator firmware might not be working with cache.

3 Enter:

```
M>cf reset=rom
M>rst
M>m -d4 -a4 0..
```

- Read value not equal to the written value implies that reset is tied to memory controller. Check $\overline{\text{HRESET}}$ and $\overline{\text{SRESET}}$ for correct connections.

4 If you have memory problems running Windows NT, you may have this problem:

- System normally runs in little endian mode
- "rst" returns processor to big endian. Memory controller on target still little endian, so memory access doesn't work.

5 Hand load a little program:

```
M>m -d4 -a4 100=38210001,60000000,60000000,4bfffff4
M>reg GPR1=0
M>
```

This means: Add 1, GPR1, NOP, NOP, JMP .-4

Set the PC to this program:

```
M>reg PC=100
M>
```

Step, and then check the register:

```
M>s
  PC=00000104
M>reg GPR1
  reg GPR1=00000001
M>
```

This should return"reg GPR1=00000001" .

Step some more and verify that GPR1 increments after every four steps:

```
M>s 4
  PC=00000104
M>reg GPR1
  reg GPR1=00000002
M>
```

Problems with the LAN Interface

If LAN communication does not work

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

- ☐ If using an emulation probe, make sure you have connected it to the proper power source and the power light is lit.
- ☐ Make sure you wait for the power-on self test to complete before connecting.
- ☐ Make sure the LAN cable is connected. Watch the LAN LED's on the back of the logic analysis system to see whether the system is detecting LAN activity. Refer to your LAN documentation for testing connectivity.
- ☐ If using an emulation probe, make sure its communication configuration switches are set correctly. Unplug emulation probe power and plug it in again to make sure the switch settings are read correctly by the emulation probe.
- ☐ Check that the host computer or debugger was configured with the correct LAN address. If the logic analysis system is on a different subnet from the host computer, check that the gateway address is correct.
- ☐ Make sure the logic analysis system's IP address is set up correctly.

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 emulator.

  The subnet mask is set in the logic analysis system's System Admin window. 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 logic analysis system may take a very long time to connect to the network after it is turned on.

Problems with the Emulation Probe

To run the power up self test

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

	Pwr/Target LEDs	Meaning
1	❍● ❍❍	Initial power up, system reset
2	❍● ❍❍	XILINX array initialized successfully
3	❍● ●❍	XILINX array tested successfully
4	●● ❍❍	BOOT ROM space tested successfully
5	❍● ●❍	GENERIC ROM space tested successfully
6	●● ❍❍	DRIVER ROM space tested successfully
7	❍● ●❍	RESERVED ROM space tested successfully
8	●● ❍❍	RAM tested successfully
9	❍● ●❍	LAN internal feedback tested successfully
10	❍● ❍❍	Boundary scan master (BSM) test begun
11	●● ●❍	BSM test completed, start system, load drivers, initialize LAN

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

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

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

❍● ❍❍	No target system power, or the emulation probe is not connected to the target system, or
❍● ❍●	PowerPC is checkstopped
❍● ●❍	PowerPC is running user code
●● ●●	PowerPC is in an unknown state
●● ●❍	Only the boot ROM was used; other firmware in the Flash EPROM has been corrupted

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 as shown in the "Connecting the Emulation Probe to a LAN" chapter. 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, and then cycle power.
- To ensure that the firmware is working as it should, reprogram the firmware, and then cycle power.

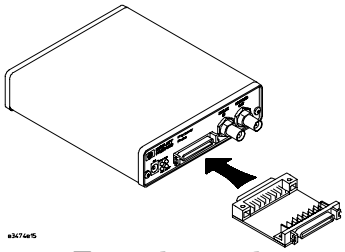
To execute the built-in performance verification test (emulation probe only)

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

Some of these tests can be performed through the logic analysis system. The LAN tests can only be executed through the RS-232 port.

To perform the PV tests through the logic analysis system

1. End any Emulation Control Interface sessions.
2. Disconnect the 50-pin cable from the emulation probe, and plug the loopback test board into the emulation probe.



3. From the emulation probe icon menu, open the Performance Verification window.
4. Enter the LAN address of the emulation probe.
5. Select the number of iterations to perform.
6. Click **Start PV**.

   The results will appear on screen.

Additional PV Tests

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.

For the BREAK IN, TRIGGER OUT BNC FEEDBACK TEST, connect a coaxial cable between BREAK IN and TRIGGER OUT
For the TARGET PROBE FEEDBACK TEST, connect the self-test board (HP part number E3496-66502).

1. Set all of the switches to ON/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.

2. Use a terminal emulator to connect to the emulation probe.
3. Enter the **pv** command.

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

Examples:

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
```

On a good system, when the feedback connector is plugged into the target connector, the RESET LED will light and the BKG and USER LEDs will be out.

The results on a good system, with the BNC's connected, and with the self-test board plugged in, are as follows:

```
R>pv 1

  Testing: HPE3499A Series Emulation System
    Test # 1: Powerup PV Results                              Passed!
    Test # 2: LAN 10Base2 Feedback Test                       Passed!
    Test # 3: LAN 10BaseT Feedback Test                       Passed!
    Test # 4: Break In and Trigger Out BNC Feedback Test      Passed!
    Test # 5: Target Probe Feedback Test                      Passed!
    Test # 6: Boundary Scan Master Test                       Passed!
    Test # 7: I2C                                             Passed!
    Test # 8: Data Lines Test                                 Passed!
  PASSED  Number of tests: 1           Number of failures: 0

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

  HPE3499B Series Emulation System
    Version:   A.07.54 22Apr98
```

```
    Location:  Generics

  HPE3454A PowerPC 700 JTAG Emulator
    Version:   A.01.07 26May98
M>
```

You may get an error like "!ERROR 172! Bad status code (0xff) from the hard reset sequence" just before the prompt. This is because the selftest loopback connector is installed instead of being connected to a real PowerPC target system. You will also get either a "c>" or "?>" prompt for the same reason, this is normal and expected. Any errors after the "PASSED Number of tests: 1 Number of failures: 0" line can be ignored.

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

If the particular failure you see is not listed below, contact HP 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 emulation probe.

```
R>pv -t2 -v2 1

  Testing: HPE3499A 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: HPE3499A 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: HPE3499A 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 correct LLA and IP addresses have been set in the virtual EEPROM through the "lan" command. If the "lan" command shows bad information for the LLA and IP, 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 that requires service from HP.

Test 3: 10BaseT Feedback Test failed

```
R>pv -t3 -v2 1

  Testing: HPE3499A 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 it then passes, the cable is faulty. If it still fails, contact HP for service.

If the emulation probe 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 OFF/OPEN, and all other configuration switches to ON/CLOSED (this enables LAN using 10BaseT). Cycle power and wait for 15 to 30 seconds. Then "ping" the emulation probe 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 emulation probe 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 emulation probe. If all else is good, then failure to respond to ping indicates a faulty emulation probe.

HPE3499A TEST 4: Break In and Trigger Out BNC Feedback Test

```
R>pv -t4 -v2 1

  Testing: HPE3499A 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 HP, ensure you have connected a good Coaxial cable between the two BNCs. If the cable is good, the emulation probe is bad.

TEST 5: Target Probe Feedback Test

A verbose output on this test can be extensive. For example, the following is the output of this test if you forget to plug in the self-test board.

```
p>pv -t5 -v2 1

  Testing: HPE3499A Series Emulation System
    Test # 5: Target Probe Feedback Test                    failed!
      Bad 20 Pin Status Read when unconnected = 0x7fb7
                               Expected Value = 0xffb7
      Bad 20 Pin Status Read when connected= 7fb7
                              Expected Value = 0x7fb7
      Output 19 Low not received on Input 11
      Output 11 Low not received on Input 19
      Output 13 Low not received on Input 1
      Output 12 High not received on Input 6
      Output 12 and Input 6 not pulled high on release
      Output 8 Low not received on Input 10
      Output 7 Low not received on Input 20
      Output 4 Low not received on Input 14
      Output 2 Low not received on Input 18
  FAILED  Number of tests: 1          Number of failures: 1
```

If you get a verbose output like this, check to make sure that the loopback test board was connected properly.

TEST 6: Boundary Scan Master Test
TEST 7: I2C Test

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

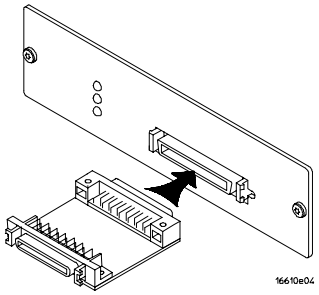
If these tests fail, return the processor probe to HP for replacement.

Problems with the Emulation Module

Occasionally you may suspect a hardware problem with the emulation module or target interface module. The procedures in this section describe how to test the hardware, and if a problem is found, how to repair or replace the broken component.

To run the built-in performance verification test using the logic analysis system (emulation module only)

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 (HP 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 run complete performance verification tests using a telnet connection (emulation module only)

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

   On a good system, the RESET LED will light and the BKG and USER LEDs will be out.
2. telnet to the emulation module.
3. 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. Note, however, that some of the options listed may not apply to your emulator.

Examples:

If you are using a UNIX system to telnet to a logic analysis system named "mylogic", enter:

```
telnet mylogic 6472
```

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 loopback test board connected are as follows:

```
M>pv 1

  Testing: HPE3499C Series Emulation System
    Test  1: Powerup PV Results                              Passed!
    Test  2: Target Probe Feedback Test                      Passed!
    Test  3: Boundary Scan Master Test                       Passed!
    Test  4: I2C Test                                        Passed!
    Test  5: Data Lines Test                                 Passed!
  PASSED  Number of tests: 1           Number of failures: 0

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

  HPE3499C Series Emulation System
    Version:   A.07.54 22Apr98
    Location:  Generics

  HPE3454A PowerPC 700 JTAG Emulator
    Version:   A.01.07 26May98
M>
```

You may get an error like "!ERROR 172! Bad status code (0xff) from the hard reset sequence" just before the prompt. This is because the selftest loopback connector is installed instead of being connected to a real PowerPC target system. You may also get a "?>" prompt for the same reason, and this is normal and expected. Any errors after the "PASSED Number of tests: 1 Number of failures: 0" line can be ignored.

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 -v option ("verbose" level) of 2 or more.

> If the particular failure you see is not listed below, contact HP for assistance.

TEST 3: Boundary Scan Master Test
TEST 4: I2C Test

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

If these tests fail, return the emulation module to HP for replacement.

Returning Parts to Hewlett-Packard for Service

The repair strategy for this emulator is board replacement.

Exchange assemblies are available when a repairable assembly is returned to Hewlett-Packard. These assemblies have been set up on the "Exchange Assembly" program. This lets you 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.

To return a part to Hewlett-Packard

1. **Follow the procedures in this chapter to make sure the problem is caused by a hardware failure, not by configuration or cabling problems.**
2. **In the U.S., call 1-800-403-0801. Outside the U.S., call your nearest HP sales office. Ask them for the address of the nearest HP service center.**
3. **Package the part and send it to the HP service center.**

   Keep any parts which you know are working. For example, if only the target interface module is broken, keep the emulation module 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 HP.

The HP 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 HP sales or service representative for details.

To obtain replacement parts

The following table lists some parts that may be replaced if they are damaged or lost. Contact your nearest Hewlett-Packard Sales Office for further information.

Part numbers

Exchange Assemblies

Part Number	Description
16600-69515	Emulation module
E3454-69401	Programmed emulation probe assembly

Replacement Assemblies

Part number	Description
16700-61608	Expansion cable
E3494-61604	16-pin cable
E3496-61601	50-pin cable
E3496-66502	loopback test board
E3481-61601	20-pin cable
E3452-66501	Target interface module (PPC JTAG board)
0950-3043	Power supply for emulation probe

Cleaning the Instrument

If the instrument requires cleaning:

1. Remove power from the instrument.
2. Clean the instrument with a mild detergent and water.
3. Make sure the instrument is completely dry before reconnecting it to a power source.

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 Migration By loading new firmware and connecting a different TIM, your emulator migrates from support of one PowerPC model to support of another PowerPC model.

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.

Emulator As used in this manual, the term Emulator applies equally to both the Emulation Module and the Emulation Probe.

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.

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 HP 16500 or HP 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 HP 16505A prototype analyzer acts as an analysis and display processor for the HP 16500B/C logic analysis system. It provides a windowed interface and powerful analysis capabilities. Replaced by HP 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 HP's term for a set of tools for debugging your target system. A solution includes probing, inverse assembly, the HP 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.

DECLARATION OF CONFORMITY

according to ISO/IEC Guide 22 and EN 45014

Manufacturer's Name: Hewlett-Packard Company

Manufacturer's Address: Colorado Springs Division
1900 Garden of the Gods Road
Colorado Springs, CO 80907 USA

declares that the product

Product Name: Logic Analyzer

Model Number(s): HP 16600A, HP 16601A, HP 16602A, HP 16603A

Product Option(s): All

conforms to the following Product Specifications:

Safety:	IEC 1010-1:1990+A1 / EN 61010-1:1993	
	UL 3111	
	CSA-C22.2 No. 1010.1:1993	
EMC:	CISPR 11:1990 / EN 55011:1991	Group 1, Class A
	IEC 555-2:1982 + A1:1985 / EN 60555-2:1987	
	IEC 555-3:1982 + A1:1990 / EN 60555-3:1987 + A1:1991	
	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 801-4:1988 / EN 50082-1:1992	0.5 kV Sig. Lines, 1 kV Power Lines

Supplementary Information:

The product herewith complies with the requirements of the Low Voltage Directive 73/23/EEC and the EMC Directive 89/336/EEC and carries the CE marking accordingly.

This product was tested in a typical configuration with Hewlett-Packard test systems.

Colorado Springs, 08/18/97

John Strathman, Quality Manager

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

Product Regulations

Safety

IEC 1010-1:1990+A1 / EN 61010-1:1993
UL 3111
CSA-C22.2 No.1010.1:1993

EMC

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

CE
ISM 1-A

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

N279

Immunity

EN50082-1	Code[1]	Notes[2]
IEC 801-2 (ESD) 4kV CD, 8kV AD	3	
IEC 801-3 (Rad.) 3 V/m	1	
IEC 801-4 (EFT) 0.5 kV, 1kV	3	

[1] Performance 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.

[2] Notes: (none)

Sound Pressure Level <60 dBA

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in subparagraph (C) (1) (ii) of the Rights in Technical Data and Computer Software Clause in DFARS 252.227-7013.
Hewlett-Packard Company, 3000 Hanover Street, Palo Alto, CA 94304 U.S.A.
Rights for non-DOD U.S. Government Departments and Agencies are set forth in FAR 52.227-19(c)(1,2).

Document Warranty

The information contained in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose.

Hewlett-Packard shall not be liable for errors contained herein or for damages in connection with the furnishing, performance, or use of this material.

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 contact. 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 fuseholders. 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.
- Do not operate the instrument in the presence of flammable gasses or fumes. Operation of any electrical instrument in such an environment constitutes a definite safety hazard.
- Do not install substitute parts or perform any unauthorized modification to the instrument.
- Capacitors inside the instrument may retain a charge even if the instrument is disconnected from its source of supply.
- Use caution when exposing or handling the CRT. Handling or replacing the CRT shall be done only by qualified maintenance personnel.

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.

WARNING

The Warning 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 Warning sign until the indicated conditions are fully understood and met.

CAUTION

The Caution sign denotes a hazard. It calls attention to an operating procedure, practice, or the like, which, if not correctly performed or adhered to, could result in damage to or destruction of part or all of the product. Do not proceed beyond a Caution symbol until the indicated conditions are fully understood or met.

Hewlett-Packard
P.O. Box 2197
1900 Garden of the Gods Road
Colorado Springs, CO 80901

Product Warranty

This Hewlett-Packard product has a warranty against defects in material and workmanship for a period of one year from date of shipment. During the warranty period, Hewlett-Packard 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 Hewlett-Packard.

For products returned to Hewlett-Packard for warranty service, the Buyer shall prepay shipping charges to Hewlett-Packard and Hewlett-Packard 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 Hewlett-Packard from another country.

Hewlett-Packard warrants that its software and firmware designated by Hewlett-Packard for use with an instrument will execute its programming instructions when properly installed on that instrument. Hewlett-Packard 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. Hewlett-Packard 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. Hewlett-Packard 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 Hewlett-Packard products.

For any assistance, contact your nearest Hewlett-Packard Sales Office.

Certification

Hewlett-Packard Company certifies that this product met its published specifications at the time of shipment from the factory. Hewlett-Packard 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.

About this edition

This is the *Emulation for the PowerPC 700 User's Guide*.

Publication number E3454-97001, December 1998
Printed in USA.

The information in this manual previously appeared in:

E3454-97000, Sept. 1997

New editions are complete revisions of the manual. Many product updates do not require manual changes; and, conversely, manual corrections may be done without accompanying product changes. Therefore, do not expect a one-to-one correspondence between product updates and manual updates.

Comments welcome!

Send your comments or suggestions regarding this manual to:
documentation@col.hp.com