Contents

HP E1468A/E1469A Relay Matrix Switch Module User's Manual

Edition 4

Warranty . . . 3
WARNINGS . . . 4
Safety Symbols . . . 4
Declaration of Conformity . . . 5
Reader Comment Sheet . . . 7

Chapter 1. Getting Started . . . 9

Using This Chapter . . . 9
Relay Matrix Switch Module Description . . . 9
Instrument Definition . . . 12
Programming the Relay Matrix Switch Module . . . 12
Specifying SCPI Commands . . . 12
Initial Operation . . . 14
Configuring, Wiring, and Installing the Relay Matrix Switch Modules . . . 15
Setting the Logical Address Switch . . . 16
Setting the Status Register Switch . . . 17
Selecting the Interrupt Priority . . . 18
Installing the Relay Matrix Switch Module in a Mainframe . . . 19
Wiring the Terminal Module . . . 20
Attaching the Terminal Module to the Relay Matrix Switch Module . . . 22
Connector Pin-Out . . . 23
Configuring the HP E1468A Terminal Module . . . 24
Configuring the HP E1469A Terminal Module . . . 25
Creating Larger Matrices with the HP E1468A/69A Modules . . . 27
Additional Cables . . . 30

Chapter 2. Using the Relay Matrix Switch Module . . . 31

Using This Chapter . . . 31
Relay Matrix Switch Module Commands . . . 31
Power-on and Reset Conditions . . . 32
Module Identification . . . 32
Switching Channels . . . 33
Query Open/Closed Channels . . . 34
Recalling and Saving States . . . 35
Detecting Error Conditions . . . 36
Synchronizing the Relay Matrix Switch Module . . . 37
Querying the Relay Matrix Switch Module . . . 37

Chapter 3. Relay Matrix Switch Command Reference . . . 39

About This Chapter . . . 39
Command Types . . . 39
Common Command Format . . . 39
SCPI Command Format . . . 39
Linking Commands . . . 41
SCPI Command Reference . . . 42
ABORt . . . 43
ARM . . . 44
:COUNt . . . 44
:COUNt? . . . 45

INITiate . . . 46
:CONTinuous . . . 46
:CONTinuous? . . . 47
[:IMMediate] . . . 47
OUTPut . . . 48
:ECLTrg[:STATe] . . . 48
:ECLTrg[:STATe]? . . . 49
[:EXTernal][:STATe] . . . 49
[:EXTernal][:STATe]? . . . 50
:TTLTrg[:STATe] . . . 50
:TTLTrg[:STATe]? . . . 51
[ROUTe:] . . . 52
CLOSe . . . 52
CLOSe? . . . 53
OPEN . . . 54
OPEN? . . . 55
SCAN . . . 55
STATus . . . 57
:OPERation :CONDition? . . . 59
:OPERation:ENABle . . . 59
:OPERation:ENABle? . . . 59
:OPERation[:EVENt]? . . . 60
:PRESet . . . 60
SYSTem . . . 61
:CDEScription? . . . 61
:CPON . . . 62
:CTYPe? . . . 62
:ERRor? . . . 63
TRIGger . . . 64
[:IMMediate] . . . 64
:SOURce . . . 65
:SOURce? . . . 66
IEEE 488.2 Common Command Reference . . . 67
SCPI Command Quick Reference . . . 68

Appendix A. HP E1468A/E1469A Specifications . . . 69

Appendix B. Relay Matrix Registers . . . 71

Register Definitions . . . 71
Addressing the Registers . . . 71
Reading the Registers . . . 73
ID and Device Type Registers . . . 73
Status/Control Register . . . 73
Relay Control Registers . . . 73
Writing to the Registers . . . 74
Status/Control Register . . . 74
Relay Control Registers . . . 74

Appendix C. Relay Matrix Error Messages . . . 79

Index . . . 81

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 (formerly National Bureau of Standards), to the extent allowed by that organization's calibration facility, and to the calibration facilities of other International Standards Organization members.

Warranty

This Hewlett-Packard product is warranted against defects in materials and workmanship for a period of three years from date of shipment. Duration and conditions of warranty for this product may be superseded when the product is integrated into (becomes a part of) other HP products. During the warranty period, Hewlett-Packard Company will, at its option, either repair or replace products which prove to be defective.

For warranty service or repair, this product must be returned to a service facility designated by Hewlett-Packard (HP). Buyer shall prepay shipping charges to HP and HP shall pay shipping charges to return the product to Buyer. However, Buyer shall pay all shipping charges, duties, and taxes for products returned to HP from another country.

HP warrants that its software and firmware designated by HP for use with a product will execute its programming instructions when properly installed on that product. HP does not warrant that the operation of the product, or 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 Buyer, Buyer-supplied products or interfacing, unauthorized modification or misuse, operation outside of the environmental specifications for the product, or improper site preparation or maintenance.

The design and implementation of any circuit on this product is the sole responsibility of the Buyer. HP does not warrant the Buyer's circuitry or malfunctions of HP products that result from the Buyer's circuitry. In addition, HP does not warrant any damage that occurs as a result of the Buyer's circuit or any defects that result from Buyer-supplied products.

NO OTHER WARRANTY IS EXPRESSED OR IMPLIED. HP SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Exclusive Remedies

THE REMEDIES PROVIDED HEREIN ARE BUYER'S SOLE AND EXCLUSIVE REMEDIES. HP SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER BASED ON CONTRACT, TORT, OR ANY OTHER LEGAL THEORY.

Notice

The information contained in this document is subject to change without notice. HEWLETT-PACKARD (HP) MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. HP shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material. This document contains proprietary information which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced, or translated to another language without the prior written consent of Hewlett-Packard Company. HP assumes no responsibility for the use or reliability of its software on equipment that is not furnished by HP.

U.S. Government Restricted Rights

The Software and Documentation have been developed entirely at private expense. They are delivered and licensed as "commercial computer software" as defined in DFARS 252.227-7013 (Oct 1988), DFARS 252.211-7015 (May 1991) or DFARS 252.227-7014 (Jun 1995), as a "commercial item" as defined in FAR 2.101(a), or as "Restricted computer software" as defined in FAR 52.227-19 (Jun 1987) (or any equivalent agency regulation or contract clause), whichever is applicable. You have only those rights provided for such Software and Documentation by the applicable FAR or DFARS clause or the HP standard software agreement for the product involved.

HP E1468A/E1469A Relay Matrix Switch Module User's Manual
Edition 4

Documentation History

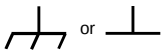
All Editions and Updates of this manual and their creation date are listed below. The first Edition of the manual is Edition 1. The Edition number increments by 1 whenever the manual is revised. Updates, which are issued between Editions, contain replacement pages to correct or add additional information to the current Edition of the manual. Whenever a new Edition is created, it will contain all of the Update information for the previous Edition. Each new Edition or Update also includes a revised copy of this documentation history page.

Edition 4 (E1468-90004) . . . . . February 1996

Safety Symbols

Instruction manual symbol affixed to product. Indicates that the user must refer to the manual for specific WARNING or CAUTION information to avoid personal injury or damage to the product.

Indicates the field wiring terminal that must be connected to earth ground before operating the equipment—protects against electrical shock in case of fault.



Frame or chassis ground terminal—typically connects to the equipment's metal frame.

Alternating current (AC).

Direct current (DC).

Indicates hazardous voltages.

WARNING Calls attention to a procedure, practice, or condition that could cause bodily injury or death.

CAUTION Calls attention to a procedure, practice, or condition that could possibly cause damage to equipment or permanent loss of data.

WARNINGS

The following general safety precautions must be observed during all phases of operation, service, and repair of this product. Failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the product. Hewlett-Packard Company assumes no liability for the customer's failure to comply with these requirements.

Ground the equipment: For Safety Class 1 equipment (equipment having a protective earth terminal), an uninterruptible safety earth ground must be provided from the mains power source to the product input wiring terminals or supplied power cable.

DO NOT operate the product in an explosive atmosphere or in the presence of flammable gases or fumes.

For continued protection against fire, replace the line fuse(s) only with fuse(s) of the same voltage and current rating and type. DO NOT use repaired fuses or short-circuited fuse holders.

Keep away from live circuits: Operating personnel must not remove equipment covers or shields. Procedures involving the removal of covers or shields are for use by service-trained personnel only. Under certain conditions, dangerous voltages may exist even with the equipment switched off. To avoid dangerous electrical shock, DO NOT perform procedures involving cover or shield removal unless you are qualified to do so.

DO NOT operate damaged equipment: Whenever it is possible that the safety protection features built into this product have been impaired, either through physical damage, excessive moisture, or any other reason, REMOVE POWER and do not use the product until safe operation can be verified by service-trained personnel. If necessary, return the product to a Hewlett-Packard Sales and Service Office for service and repair to ensure that safety features are maintained.

DO NOT service or adjust alone: Do not attempt internal service or adjustment unless another person, capable of rendering first aid and resuscitation, is present.

DO NOT substitute parts or modify equipment: Because of the danger of introducing additional hazards, do not install substitute parts or perform any unauthorized modification to the product. Return the product to a Hewlett-Packard Sales and Service Office for service and repair to ensure that safety features are maintained.

Declaration of Conformity

according to ISO/IEC Guide 22 and EN 45014

Manufacturer's Name: Hewlett-Packard Company
Loveland Manufacturing Center

Manufacturer's Address: 815 14th Street S.W.
Loveland, Colorado 80537

declares, that the product:

Product Name: Matrix Relay Switch Module

Model Number: HP E1468A/E1469A

Product Options: All

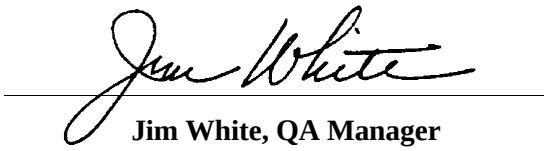
conforms to the following Product Specifications:

Safety: IEC 1010-1 (1990) Incl. Amend 1 (1992)/EN61010-1 (1993)
CSA C22.2 #1010.1 (1992)
UL 1244

EMC: CISPR 11:1990/EN55011 (1991): Group1, Class A
IEC 801-2:1991/EN50082-1 (1992): 4kVCD
IEC 801-3:1984/EN50082-1 (1992): 3 V/m
IEC 801-4:1988/EN50082-1 (1992): 1kV Power Line, .5kV Signal Lines
IEC 801-5:1992(DRAFT) 3kV Low Energy, 1kV High Energy

Supplementary Information: The product herewith complies with the requirements of the Low Voltage Directive 73/23/EEC and the EMC Directive 89/336/EEC.

Tested in a typical configuration in an HP C-size VXI mainframe.



August 20, 1995 **Jim White, QA Manager**

European contact: Your local Hewlett-Packard Sales and Service Office or Hewlett-Packard GmbH, Department HQ-TRE, Herrenberger Straße 130, D-71034 Böblingen, Germany (FAX +49-7031-14-3143).

Notes

Please fold and tape for mailing

Reader Comment Sheet

HP E1468A/E1469A 8x8 & 4x16 Relay Matrix Switch Modules User's Manual
Edition 4

You can help us improve our manuals by sharing your comments and suggestions. **In appreciation of your time, we will enter you in a quarterly drawing for a Hewlett-Packard Palmtop Personal Computer** (U.S. government employees are not eligible for the drawing).

Your Name	City, State/Province
Company Name	Country
Job Title	Zip/Postal Code
Address	Telephone Number with Area Code

Please list the system controller, operating system, programming language, and plug-in modules you are using.

fold here

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 37 LOVELAND, CO

POSTAGE WILL BE PAID BY ADDRESSEE

HEWLETT-PACKARD COMPANY
Measurement Systems Division
Learning Products Department
P.O. Box 301
Loveland, CO 80539-9984

cut along this line

fold here

Please pencil-in one circle for each statement below:

	Disagree ←				→ Agree
• The documentation is well organized.	O	O	O	O	O
• Instructions are easy to understand.	O	O	O	O	O
• The documentation is clearly written.	O	O	O	O	O
• Examples are clear and useful.	O	O	O	O	O
• Illustrations are clear and helpful.	O	O	O	O	O
• The documentation meets my overall expectations.	O	O	O	O	O

Please write any comments or suggestions below--be specific.

Chapter 1
Getting Started

Using This Chapter

This chapter describes the E1468A and E1469A Relay Matrix Switch Modules and provides programming, configuration and wiring information. This chapter contains the following sections:

- Relay Matrix Switch Module Description . . . . . Page 9
- Instrument Definition . . . . . Page 12
- Programming the Relay Matrix Switch Module . . . . . Page 12
- Initial Operation . . . . . Page 14
- Setting the Logical Address Switch . . . . . Page 16
- Setting the Status Register Switch . . . . . Page 17
- Selecting the Interrupt Priority . . . . . Page 18
- Installing the Relay Matrix Switch Module in a Mainframe . . . . . Page 19
- Wiring the Terminal Module . . . . . Page 20
- Attaching a Terminal Module to the Relay Matrix Switch Module . . . . . Page 22
- Connector Pin-Out . . . . . Page 23
- Configuring the HP E1468A Terminal Module . . . . . Page 24
- Configuring the HP E1469A Terminal Module . . . . . Page 25
- Creating Larger Matrices with the HP E1468A/69A Modules . . . . . Page 27
- Additional Cables . . . . . Page 29

Relay Matrix Switch Module Description

Both the HP E1468A and E1469A Relay Matrix Switch Modules are VXIbus C-size register-based modules and operate with an HP E1406A Command Module. Both relay modules consist of 64 two-wire relays.

The HP E1468A Relay Matrix Switch Module provides an 8x8 two-wire crosspoint matrix. Multiple modules can be easily wired together creating 8x16 (two modules), 16x16 (four modules), 8x24 (three modules), or larger matrices. Figure 1-1 shows a simplified schematic of the E1468A module and terminal module.

The HP E1469A Relay Matrix Switch Module provides a 4x16 two-wire crosspoint matrix. Multiple modules can easily be wired together creating 4x32 (two modules), 8x16 (two modules), 4x48 (three modules), or larger matrices. Figure 1-2 shows a simplified schematic of the E1469A module and terminal module.

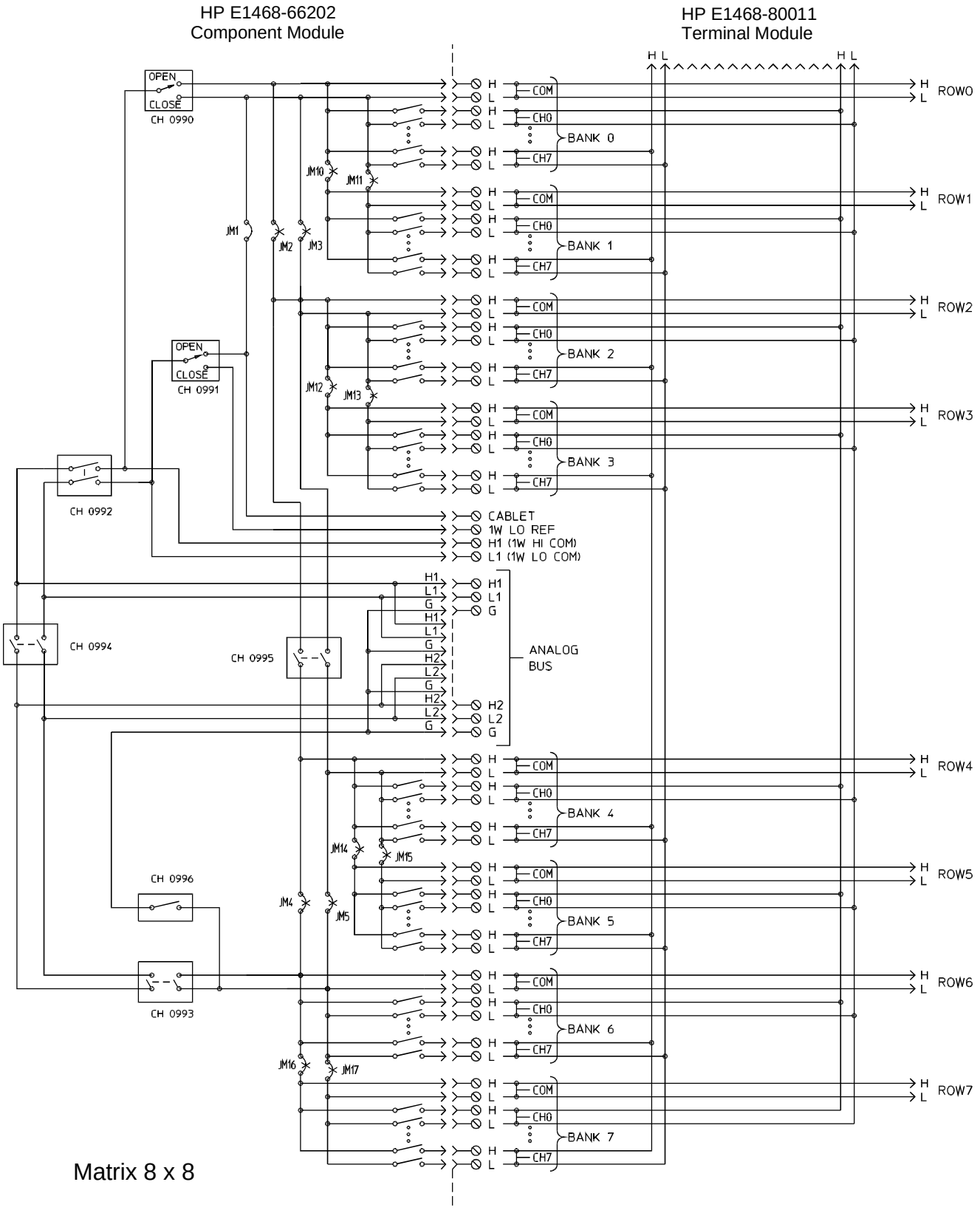


Figure 1-1. HP E1468A Simplified Schematic

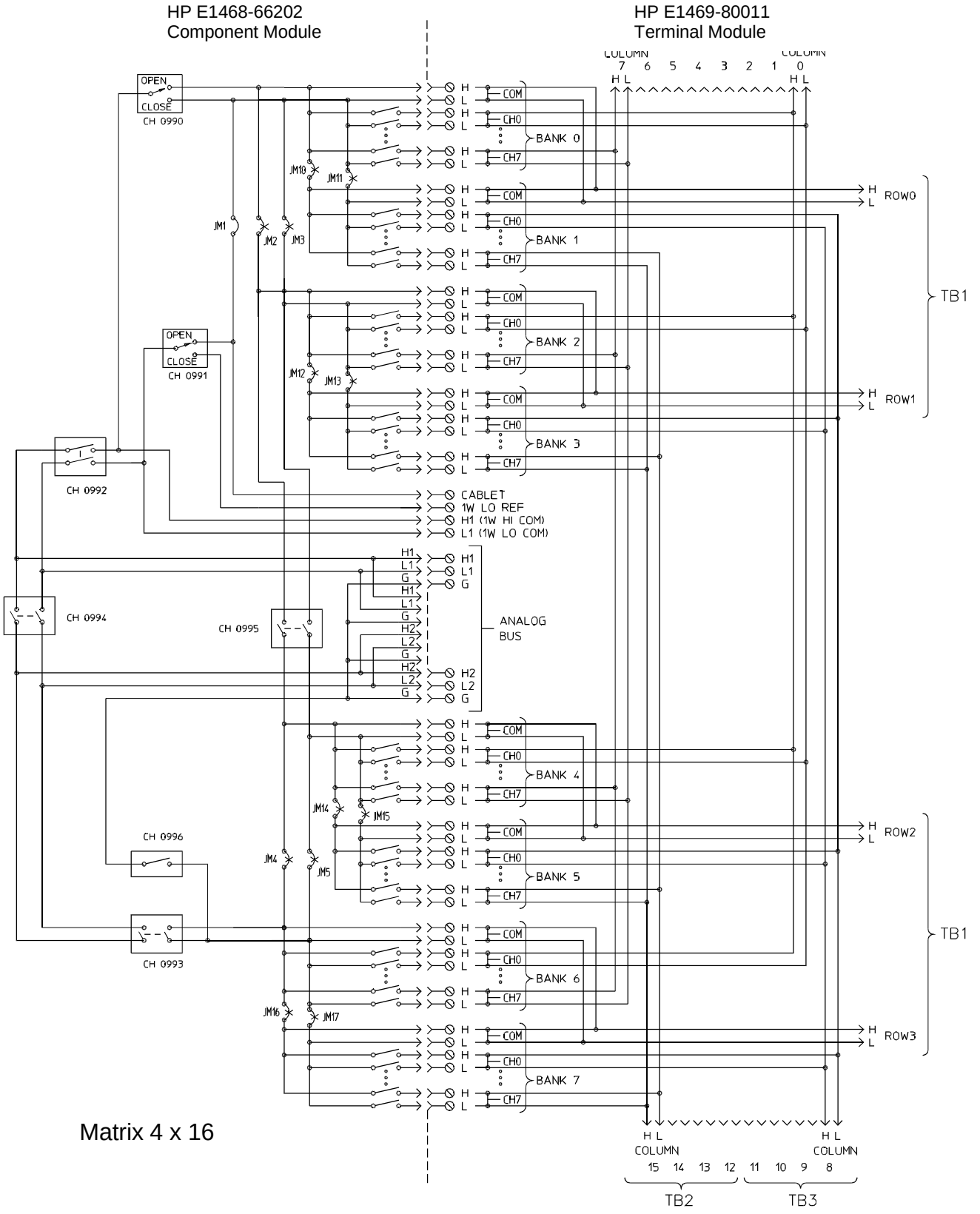


Figure 1-2. HP E1469A Simplified Schematic

Instrument Definition

Hewlett-Packard plug-in modules installed in an HP mainframe are treated as independent instruments having a unique secondary HP-IB address. Each instrument is also assigned a dedicated error queue, input and output buffers, status registers, and, if applicable, dedicated mainframe memory space for readings or data. An instrument may be composed of a single plug-in module (such as a counter) or multiple plug-in modules (for a switchbox or scanning voltmeter instrument).

Programming the Relay Matrix Switch Module

To program the relay matrix switch module using SCPI (Standard Commands for Programmable Instruments), you must select the computer language, interface address, and SCPI commands to be used. Guidelines to select SCPI commands for the relay matrix switch module follow.

Note This discussion applies only to SCPI programming. See Appendix B for information on relay matrix registers.

Specifying SCPI Commands

To address specific channels (relays) within a relay matrix, you must specify the SCPI command and matrix channel address. Use CLOSe <*channel_list*> to close specified relay(s); OPEN <*channel_list*> to open the relay(s); and SCAN <*channel_list*> to close the set of relays specified.

Note Channel numbers can be entered in the *channel_list* in any random order.

Matrix Card Numbers

The matrix card (module) number depends on the switchbox configuration (single module or multiple module) set for the matrices. (***Leading zeroes can be ignored for the card number***.) For a single-module switchbox, the card number is always 01.

For a multiple-module switchbox, the card numbers are 01, 02,...nn. The module with the lowest logical address is card number 01, the module with the next-lowest logical address is card number 02, and so on.

HP E1468A Relay Matrix Channel Addresses

For the HP E1468A Relay Matrix Switch Module, the channel address (*channel_list*) has the form (@ssrc) where ss = matrix card number (01-99), r = row number, and c = column number.

You can address single channels (@ssrc); multiple channels (@ssrc,ssrc,...); sequential channels (@ssrc:ssrc); groups of sequential channels (@ssrc:ssrc,ssrc:ssrc); or any combination.

HP E1468A 8x8 Relay Matrix Switch Module channel numbers are:

r: 0 to 7 (one digit)
c: 0 to 7 (one digit)

HP E1469A Relay Matrix Channel Addresses

For the HP E1469A Relay Matrix Switch Module, the channel address (*channel_list*) has the form (@ssrrcc) where ss = matrix card number (01-99), rr = row number, and cc = column number.

You can address single channels (@ssrrcc); multiple channels (@ssrrcc,ssrrcc,...); sequential channels (@ssrrcc:ssrrcc); groups of sequential channels (@ssrrcc:ssrrcc,ssrrcc:ssrrcc); or any combination.

HP E1469A 4x16 Relay Matrix Switch Module channel numbers are:

rr: 00 to 03 (two digits)
cc: 00 to 15 (two digits)

For example, CLOS (@10214) closes row 02, column 14 of card 01 of an HP E1469A Relay Matrix Switch Module.

Only valid channels can be accessed in a channel list or channel range. Also, the channel list or channel range must be from a lower channel number to a higher channel number. For example, CLOS (@100:233) is acceptable, but CLOS (@233:100) generates an error.

Initial Operation

An example program follows which uses HP BASIC and SCPI language to get you started using the relay matrix switch module. The example assumes an HP 9000 Series 200/300 controller and a Hewlett-Packard Interface Bus (HP-IB)[1].

The program closes row 03, column 12 of an HP E1469A 4x16 Relay Matrix Switch Module at logical address 112 (secondary address = 112/8 = 14) and queries the result. The result is returned to the controller and displayed (*1* = relay closed, *0* = relay open).

```
10 OUTPUT 70914; "*RST"                  ! Resets the module.  Sets all relays to open.
20 OUTPUT 70914; "CLOS  (@10312)"        ! Closes channel row 03, column 12 on the first module in the switchbox. Note that [ROUTe:] is implied and, therefore, optional.
30 OUTPUT 70914; "CLOS?  (@10312)"       ! Query channel.
40 ENTER 70914; Value                    ! Enter result into variable Value.
50 PRINT Value                           ! Print results (should print "1").
60 END
```

1 HP-IB is the Hewlett-Packard implementation on the IEEE 488.2-1987 standard.

Configuring, Wiring, and Installing the Relay Matrix Switch Modules

WARNING

SHOCK HAZARD. Only service-trained personnel who are aware of the hazards involved should install, remove, or configure the relay matrix switch modules. Before removing any installed module, disconnect AC power from the VXI mainframe and from any devices connected to the relay matrix switch modules.

CHANNEL WIRING INSULATION. All channels that have a common connection must be insulated so that the user is protected from electrical shock in the event that two or more channels are connected together.

CAUTION

Maximum Inputs. The maximum voltage that can be applied to any terminal is 220Vdc/250V_{rms}. The maximum current that can be applied to any terminal is 1A at 30Vdc/V_{rms}, or 0.3A at 220Vdc/250V_{rms}. The maximum power that can be applied to any terminal is 40VA.

Static Electricity. Static electricity is a major cause of component failure. To prevent damage to the electrical components in the relay matrix switch module, observe anti-static techniques when removing or installing the module or when working on the module.

Setting the Logical Address Switch

The logical address switch (LADDR) factory setting is 112. Valid addresses are from 1 to 255. If the matrix module is used with an HP E1406A Command Module in a C-size mainframe, refer to the *C-Size VXIbus Systems Configuration Guide* for addressing information. Refer to Figure 1-3 for switch information.

Note The address switch value must be a multiple of 8 if the module is the first module in a "switchbox" used with a VXIbus command module using SCPI commands.

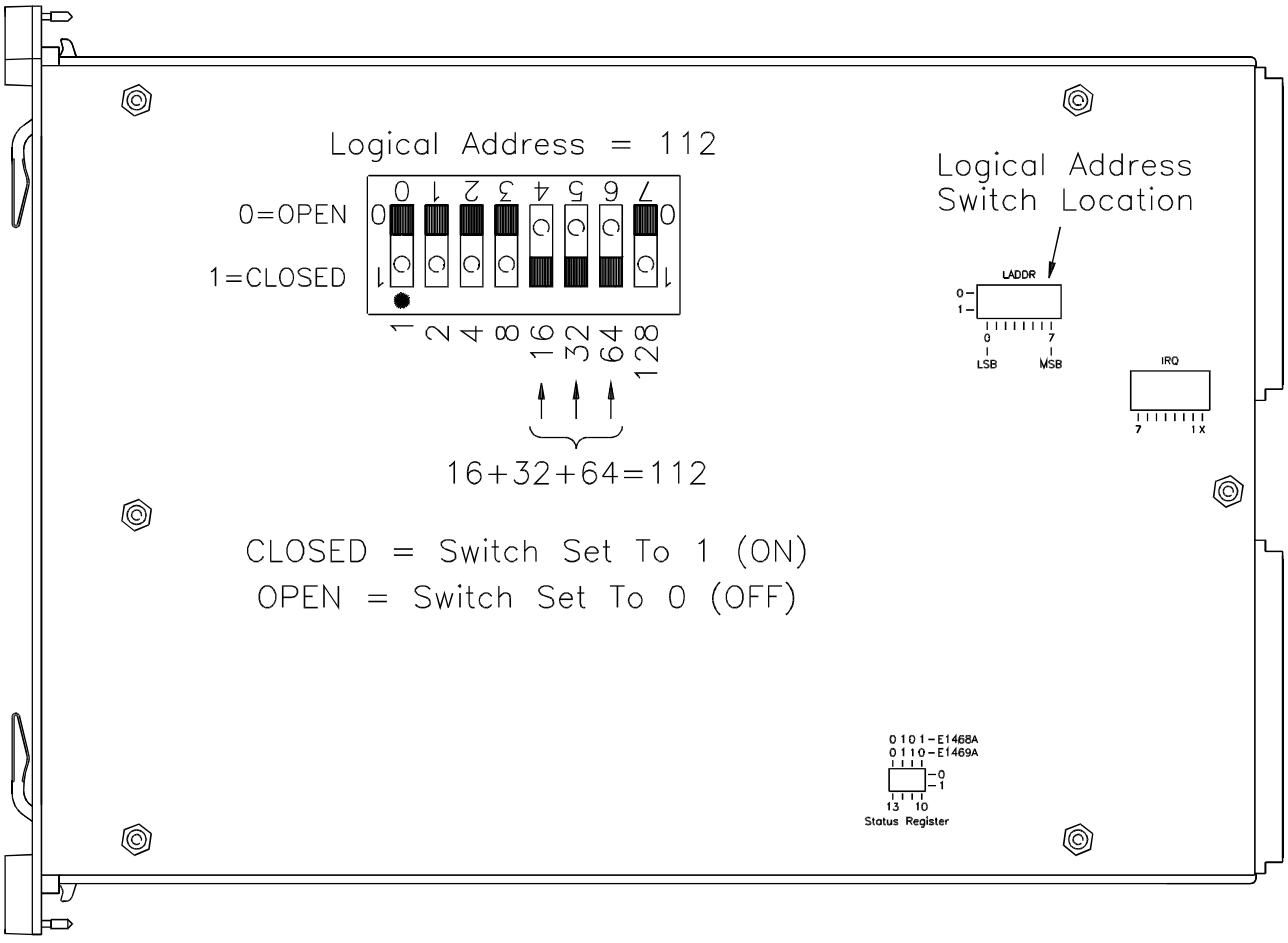


Figure 1-3. Setting the Logical Address Switch

Setting the Status Register Switch

Four bits of the status register switch (bits 10-13) define whether the relay matrix switch module is an HP E1468A or E1469A. These bits are set automatically when the terminal module is installed. To ensure proper operation, even without the terminal module, set the status register switch as shown in Figure 1-4. However, if the status register switch is set for the HP E1468A but the matrix terminal module is an HP E1469A (or vice versa), the interface won't be able to correctly identify and an error will occur.

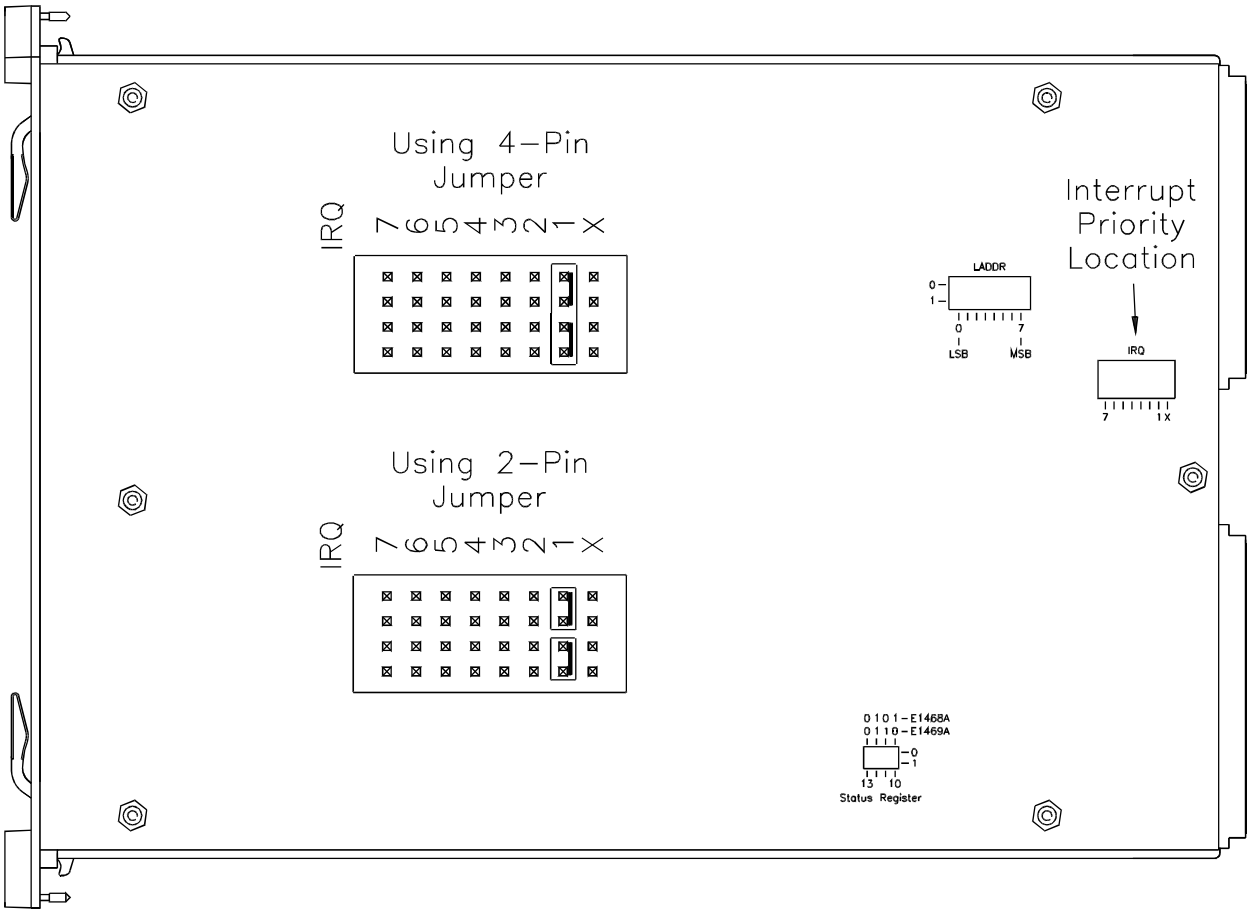


Figure 1-4. Setting the Status Register Switch

Selecting the Interrupt Priority

The HP E1468A/E1469A Relay Matrix Switch Modules generate an interrupt after a channel has been closed. These interrupts are sent to, and acknowledgments are received from, the command module (for example, an HP E1406A) through the VXIbus backplane interrupt lines.

For most applications where the relay matrix switch module is installed in an HP 75000 Series C mainframe, the interrupt priority jumper does not have to be moved. This is because the VXIbus interrupt lines have the same priority, and interrupt priority is established by installing the modules in slots numerically closest to the HP E1406A Command Module. Thus slot 1 has a higher priority than slot 2, slot 2 has a higher priority than slot 3, and so on.

Refer to Figure 1-5 to change the interrupt priority. You can select eight different interrupt priority levels. Level 1 is the lowest priority and level 7 is the highest priority. Level X disables the interrupt. The module's factory setting is level 1. To change, remove the four-pin jumper (HP part number 1258-0247) from the old priority location and reinstall in the new priority location (Figure 1-5 shows a priority change from 1 to 7). If the four-pin jumper is not used, the two jumper locations must have the same interrupt priority level selected.

Note

The interrupt priority jumper must be installed in position 1 when using the HP E1406A Command Module. Level X interrupt priority should not be used under normal operating conditions. Changing the interrupt priority level jumper is not recommended.

Locate and set the interrupt priority.

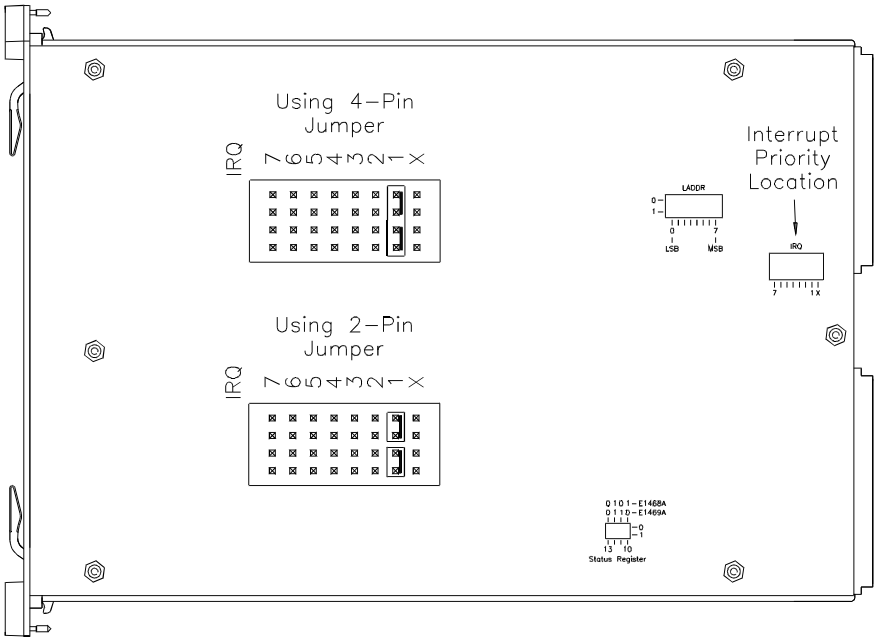


Figure 1-5. Interrupt Priority Selection

Installing the Relay Matrix Switch Module in a Mainframe

The HP E1468A/E1469A may be installed in any slot (except slot 0) in a C-size mainframe. Refer to Figure 1-6 to install the module in a mainframe.

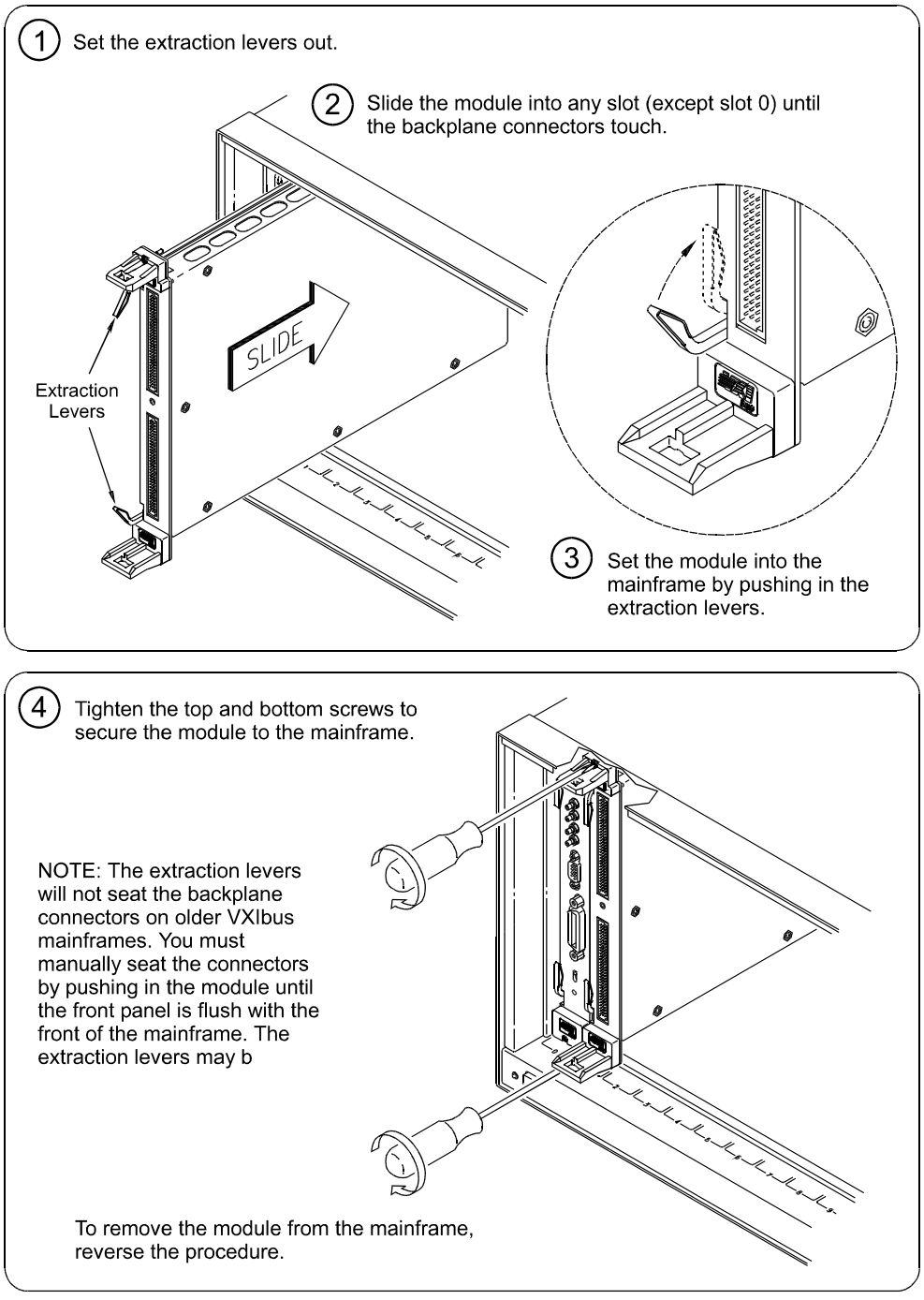


Figure 1-6. Installing the Switch Module in a VXIbus Mainframe

Wiring the Terminal Module

User wiring to the relay matrix switch modules are to the High (H) and Low (L) terminal connections. Figures 1-7 and 1-8 show the rudiments of the terminal module assembly. Expansion connectors allow for creating larger matrices (see later in this chapter).

Maximum terminal wire size is No. 16 AWG. Wire ends should be stripped 6mm (0.25 in.) and tinned. When wiring all channels, use a smaller gauge wire (No. 20 - 22 AWG).

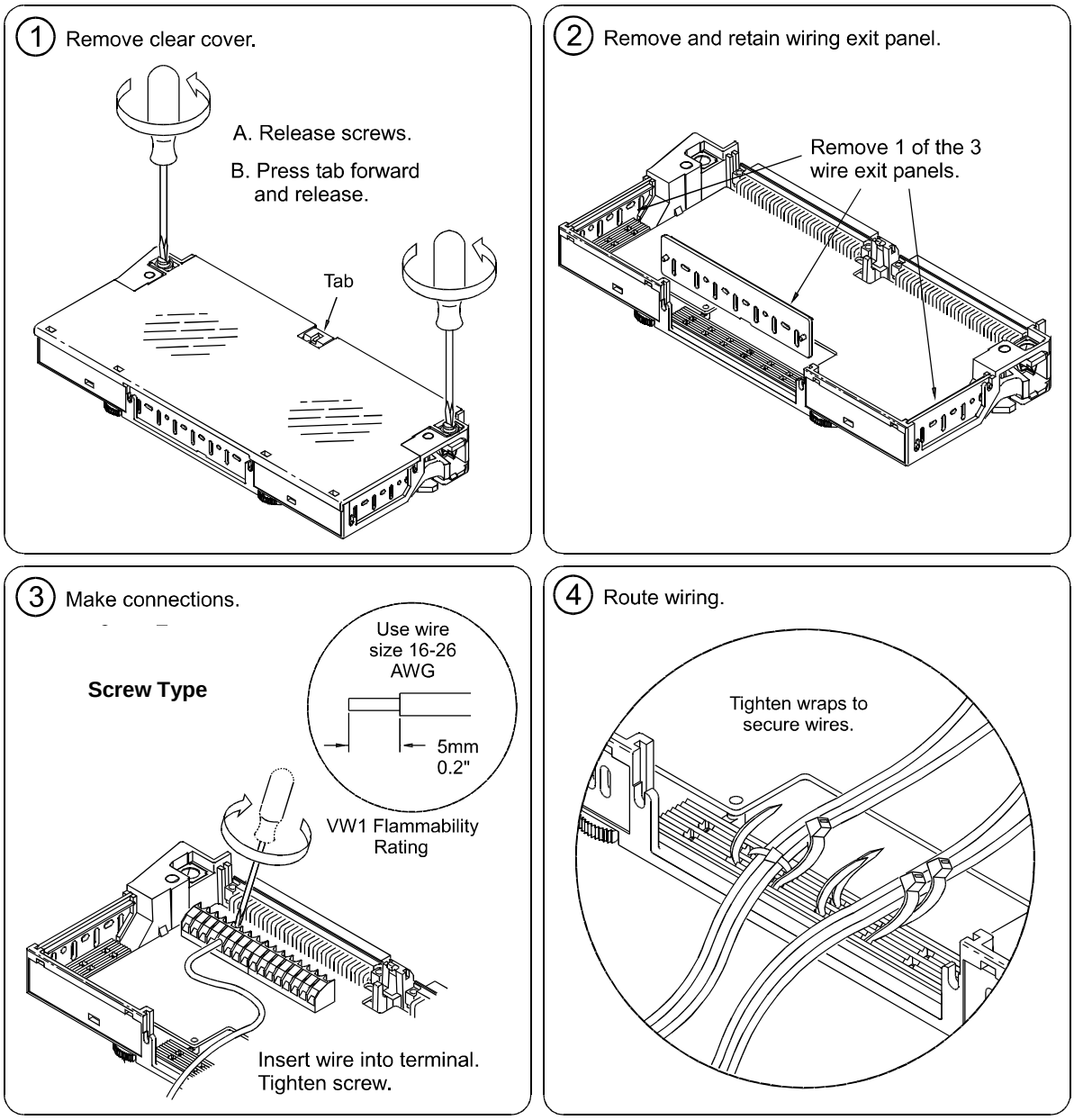


Figure 1-7. Wiring the Terminal Module

Continued on next page.

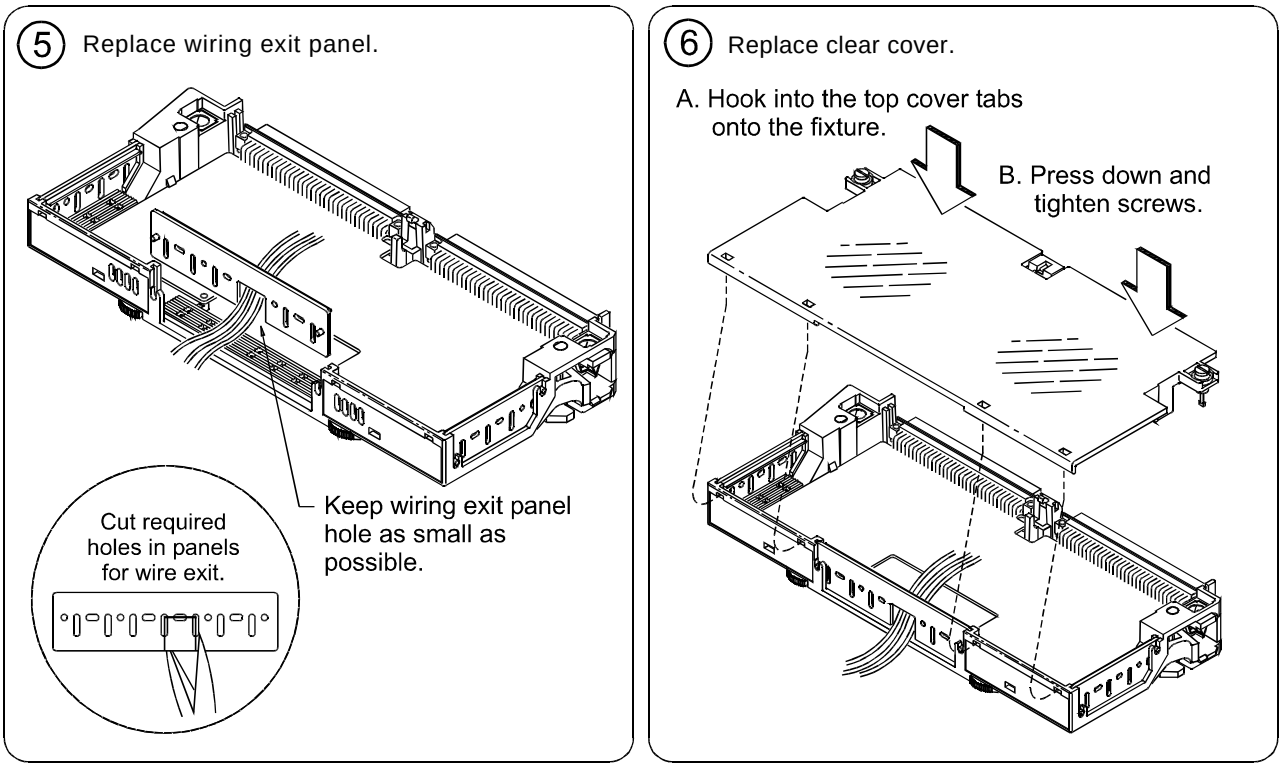


Figure 1-8. Wiring the Terminal Module

Continued from previous page.

Attaching the Terminal Module to the Relay Matrix Switch Module

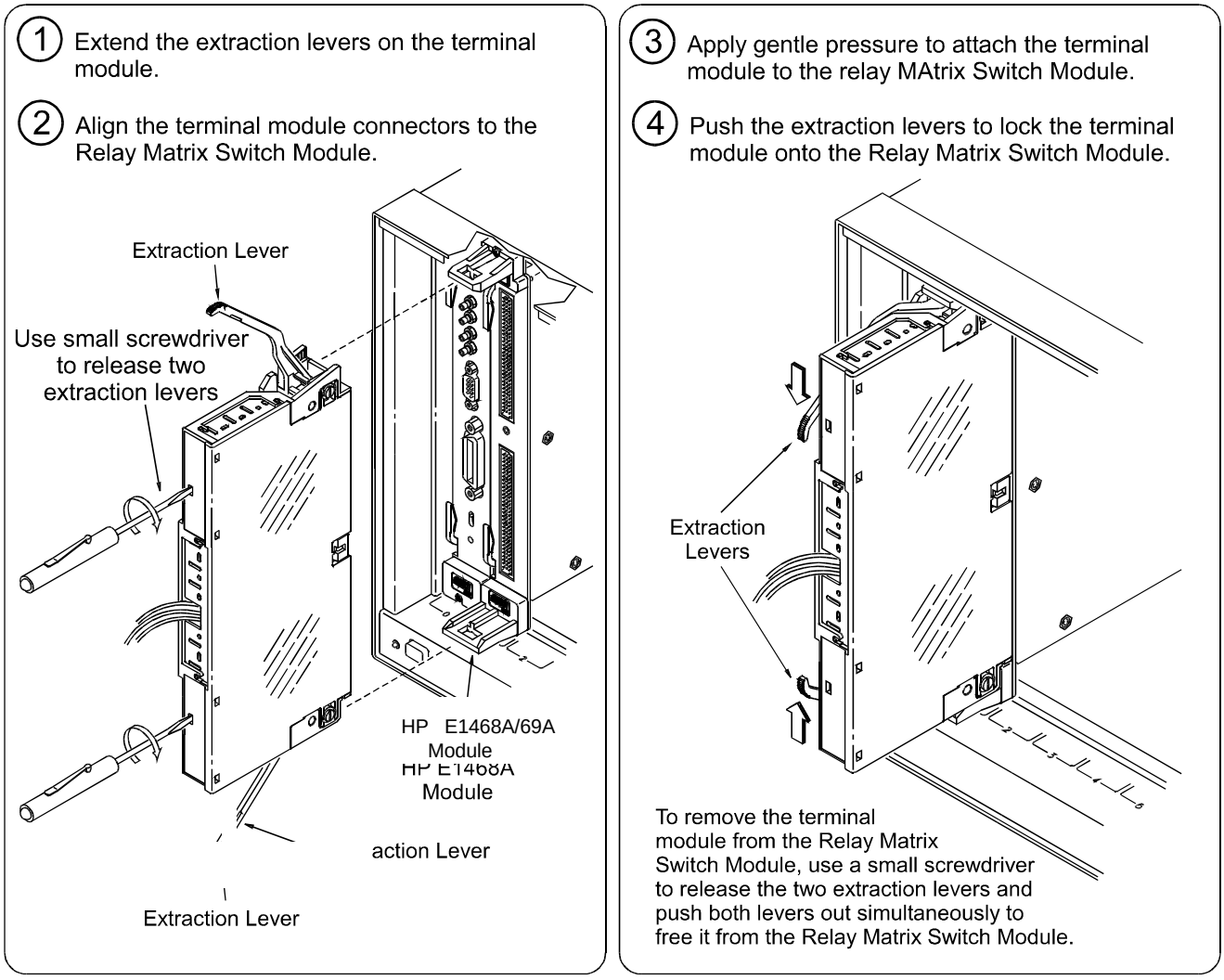


Figure 1-9. Attaching the Terminal Module to the Relay Matrix Switch Module

Connector Pin-Out

The relay matrix switch module consists of a component module (E1468-66202) and a terminal module (E1468-80011 or E1469-80011). Figure 1-10 illustrates the front panel of the E1468-66202. The relay matrix switch module's connector pin-out is illustrated below. The terminal module makes the row and column connection in order to make the matrix configuration (refer to Figures 1-1 and 1-2).

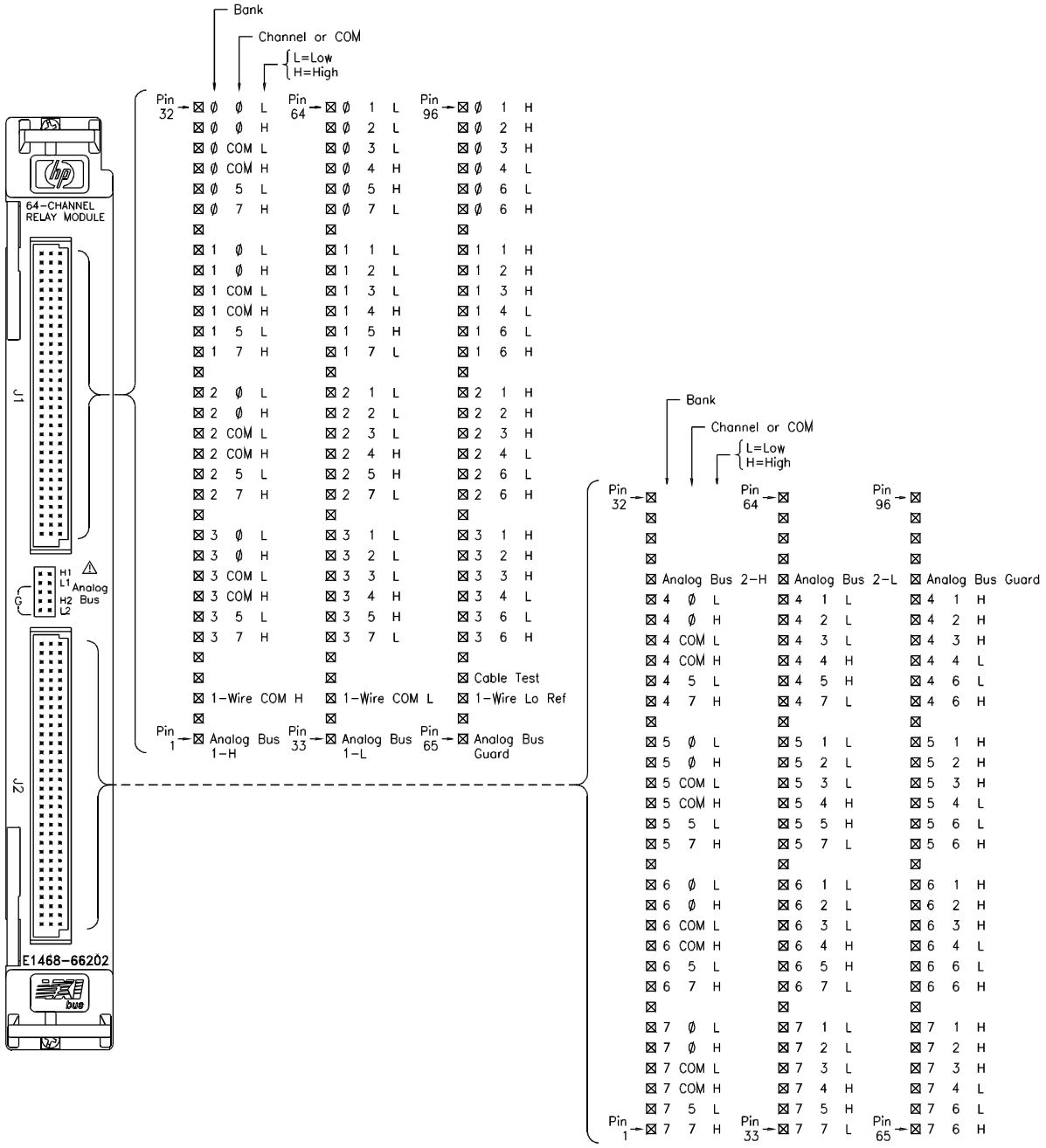


Figure 1-10. Relay Matrix Switch Module Connector Pin-out

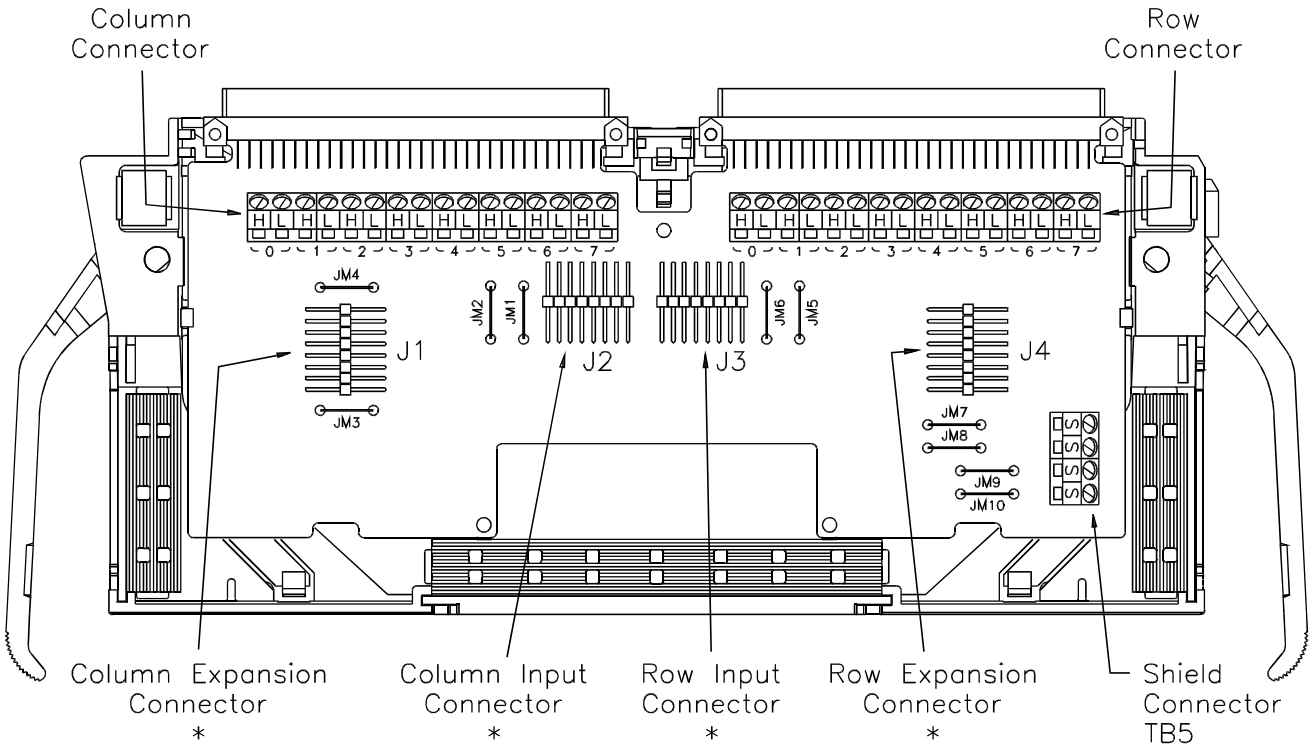
Configuring the HP E1468A Terminal Module

Figure 1-11 below shows the HP E1468A terminal module connectors and associated row/column designators. Shielding jumpers JM1 - JM10 are shown. Refer to Figure 1-13 and "Creating Larger Matrices with the HP E1468A/E1469A Modules" later in this chapter for information on using the expansion connectors J1 - J4, and for shield wiring details.

Reducing Common Mode Noise

Jumpers JM1 - JM10 on the terminal module connect row/column shields to earth ground through the VXIbus backplane. You may want to remove one or more of these jumpers to reduce common mode noise.

HP E1468A Terminal Module
(HP E1468-80011)



* In parallel with the screw terminals.
See Figure 1-13 for shield details.

Figure 1-11. HP E1468A 8 x 8 Matrix Switch Terminal Module

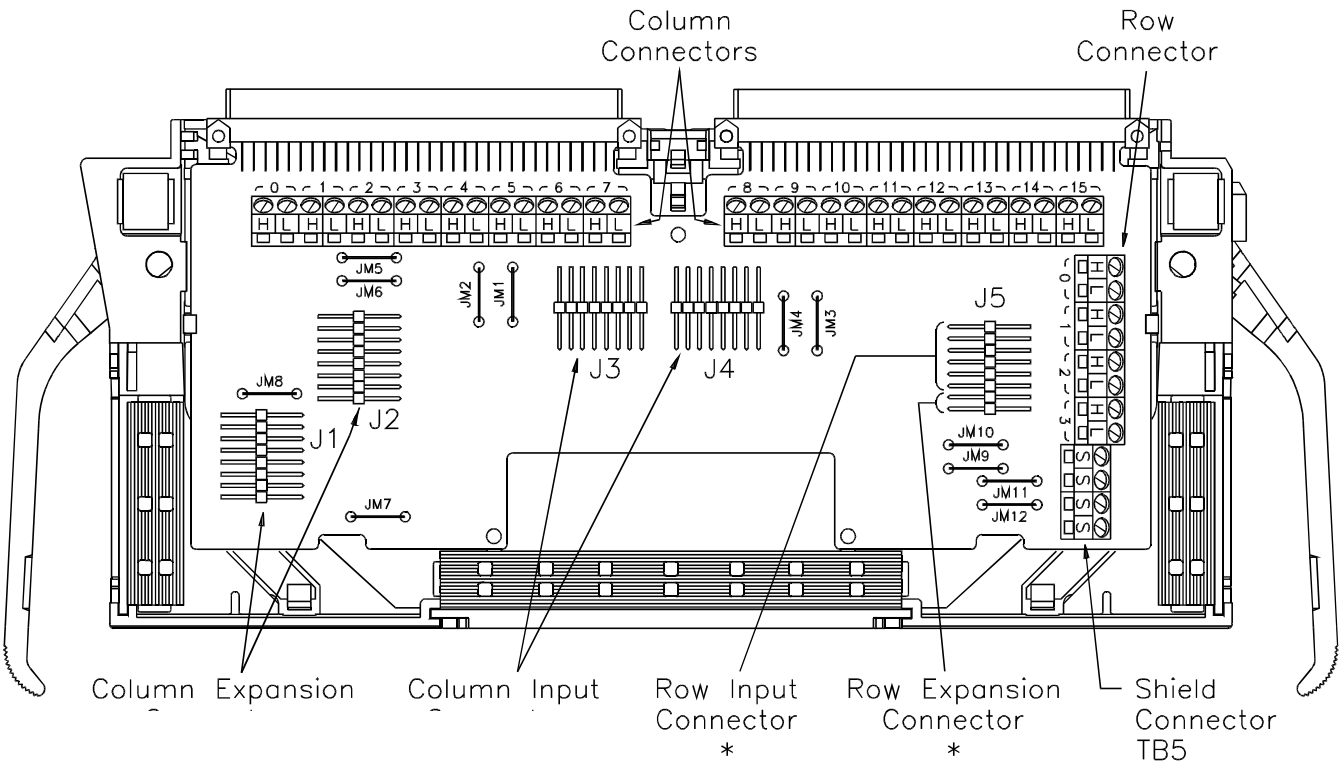
Configuring the HP E1469A Terminal Module

Figure 1-12 below shows the HP E1469A terminal module connectors and associated row/column designators. Shielding jumpers JM1 - JM12 are shown. Refer to Figure 1-13 and "Creating Larger Matrices with the HP E1468A/E1469A Modules" later in this chapter for information on using the expansion connectors J1 - J5, and for shield wiring details.

Reducing Common Mode Noise

Jumpers JM1 - JM12 on the terminal module connect row/column shields to earth ground through the VXIbus backplane. You may want to remove one or more of these jumpers to reduce common mode noise.

HP E1469A Terminal Module
(HP E1469-80011)



* In parallel with the screw terminals.
See Figure 1-13 for shield details.

Figure 1-12. HP E1469A 4 x 16 Matrix Switch Terminal Module

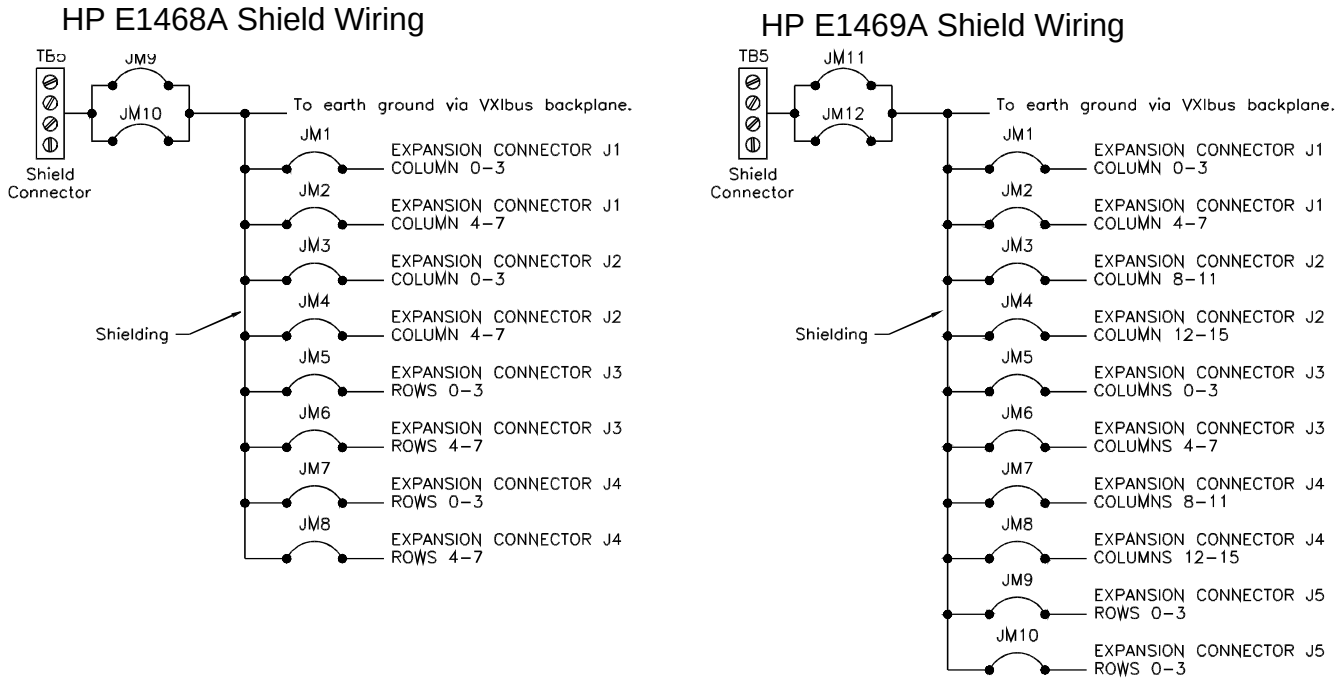


Figure 1-13. HP E1468A and E1469A Shield Wiring

Creating Larger Matrices with the HP E1468A/69A Modules

Use the expansion connectors on the terminal module to interconnect modules together to create a larger matrix. (See Figures 1-14 to 1-16 for expansion details.) Use HP part number E1468-80002 Daisy-Chain Cable (i.e., a 4-pair High and Low cable assembly) for easy expansion between modules. This cable provides a quick-disconnect allowing easy removal of modules.

8 x 24 Matrix

Figure 1-14 shows how to connect three HP E1468A Relay Matrix Switch Modules to create an 8-row by 24-column matrix. This configuration requires four HP E1468-80002 Daisy-Chain Cables.

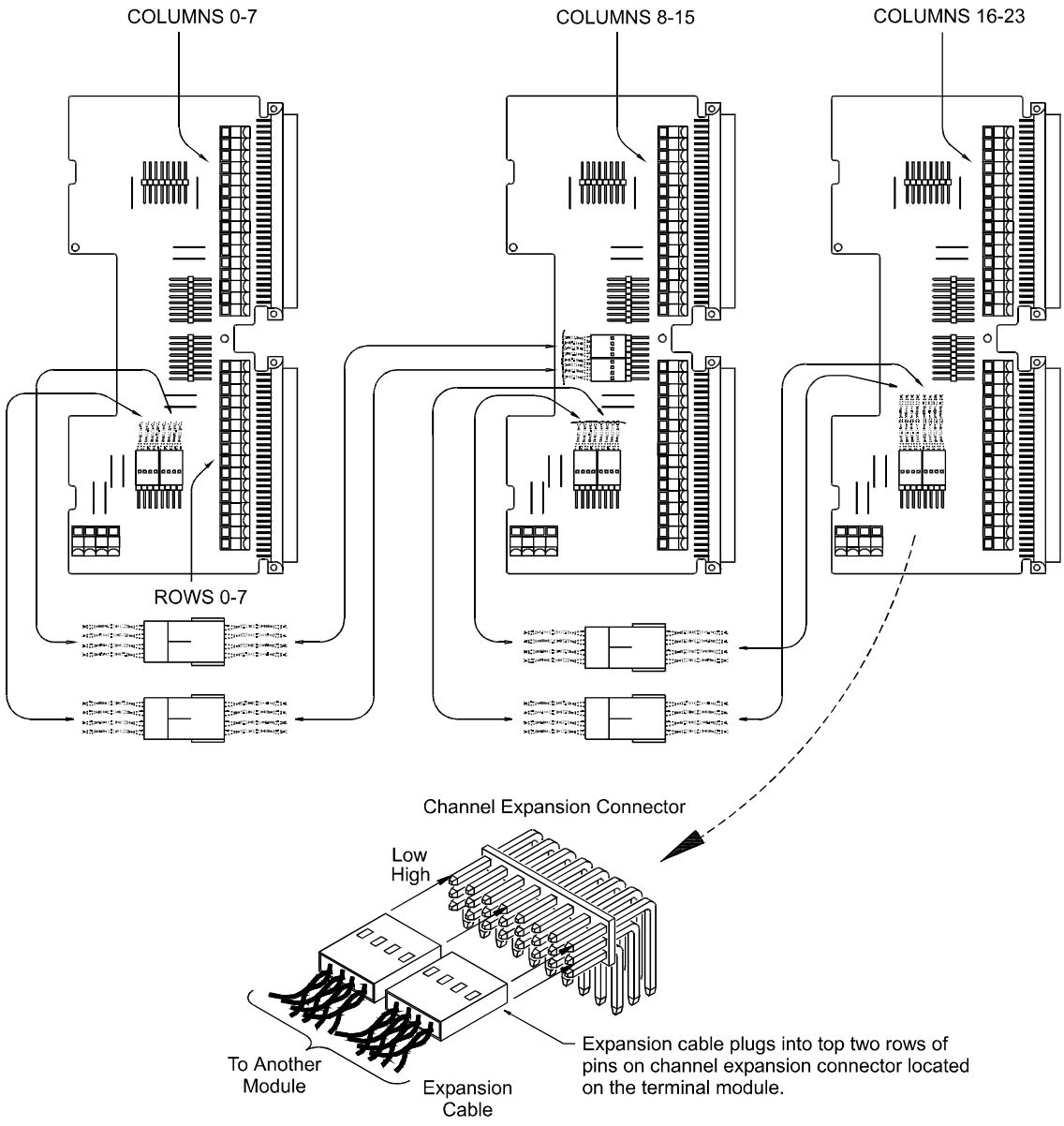


Figure 1-14. 8-Row x 24-Column Matrix Using HP E1468A

16 x 16 Matrix

Figure 1-15 shows how to connect four HP E1468A Relay Matrix Switch Modules to create a 16-row by 16-column matrix. This configuration requires eight HP E1468-80002 Daisy-Chain Cables.

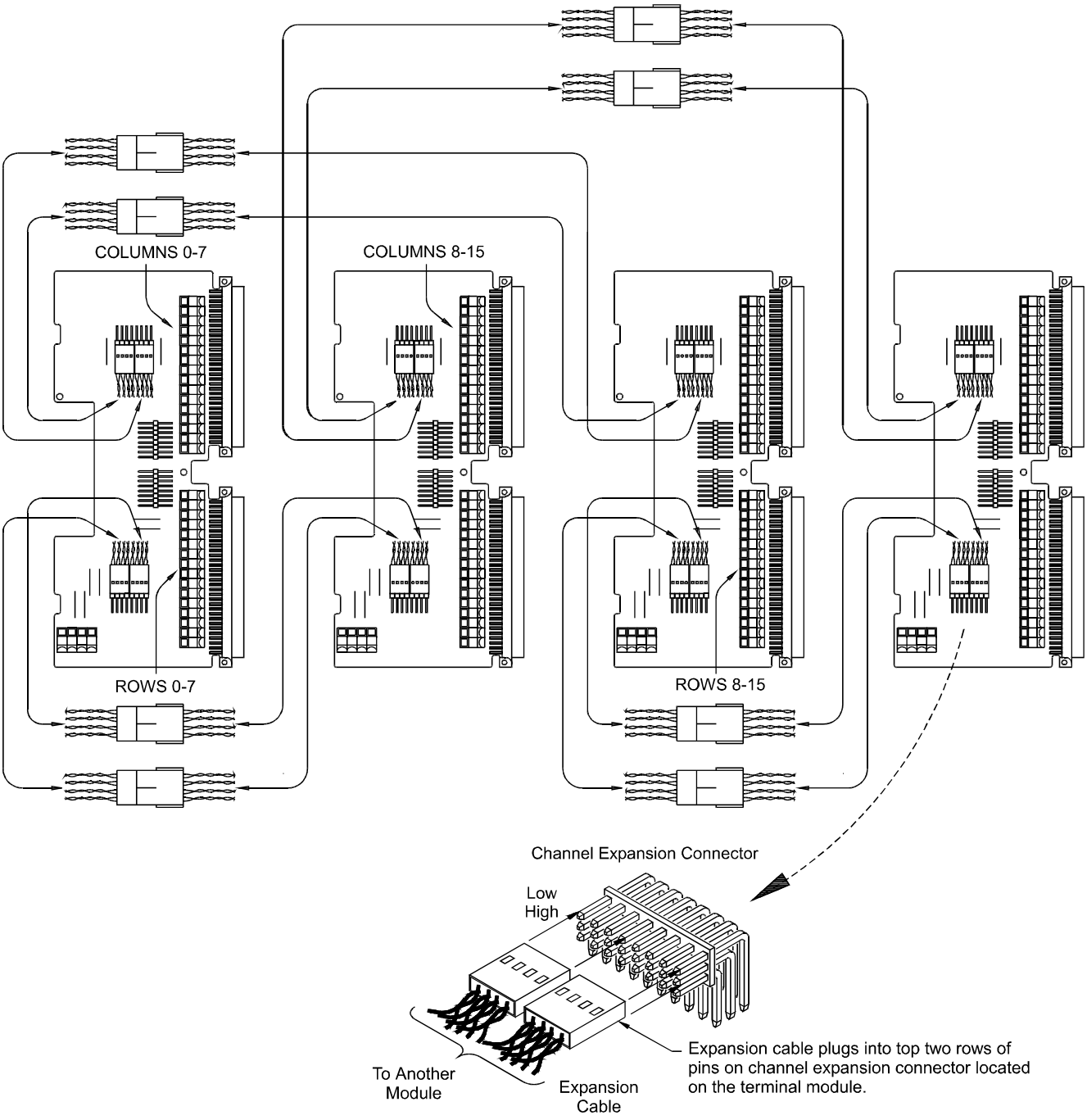


Figure 1-15. 16-Row x 16-Column Matrix Using HP E1468A

4 x 48 Matrix

Figure 1-16 shows how to connect three HP E1469A Relay Matrix Switch Modules to create a 4-row by 48-column matrix. This configuration requires two HP E1468-80002 Daisy-Chain Cables.

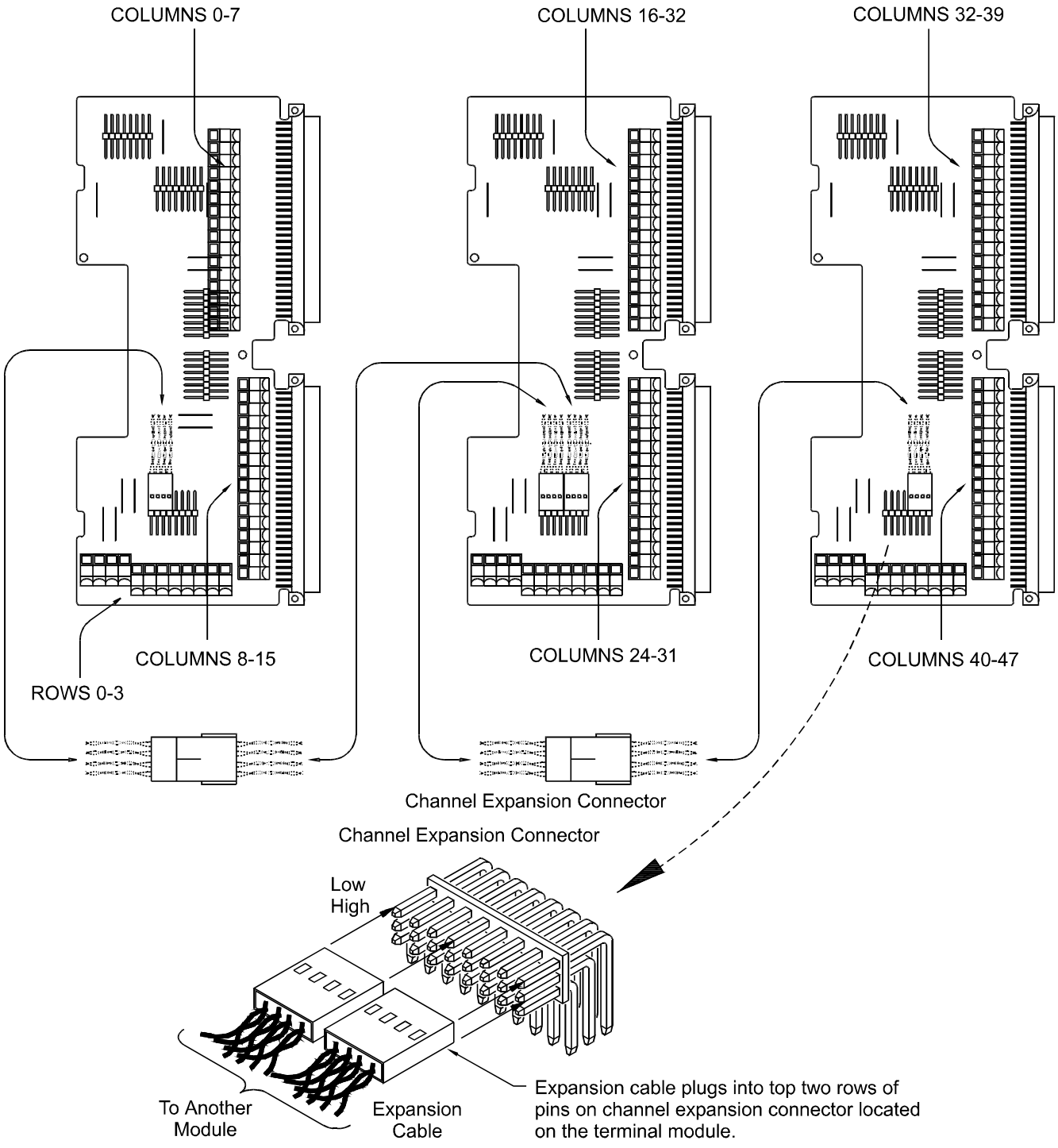


Figure 1-16. 4-Row x 48-Column Matrix Using HP E1469A

Additional Cables

To assist you in wiring your matrix modules into your test system, the following is a list of cables that are available from your local Hewlett-Packard Sales and Service office.

Description	Finished Length	End "A"	End "B"	HP Part Number
Module expansion connector with quick disconnect (twisted pair)	~30 cm.	4 x 2 connector for expansion connectors on terminal modules.	4 x 2 connector for expansion connectors on terminal modules.	E1468-80002
50 *W* Coax	2.0 m	2-pin TLA*	BNC (molded over)	E1065-61620
Dual banana instrument	2.0 m	3-pin TLA*	Dual banana	E1066-61620
SMB instrument	2.0 m	2-pin TLA*	SMB (molded over)	E1068-61620

* TLA is a family of connector/cable assemblies with good transmission line design made by a Hewlett-Packard supplier. The 2- and 3-pin TLA connectors are designed to fit on one channel of the terminal module expansion connectors.

Chapter 2
Using the Relay Matrix Switch Module

Using This Chapter

This chapter uses typical examples to show how to use the relay matrix switch modules. It contains the following sections:

- Relay Matrix Switch Module Commands . . . . . Page 31
- Power-on and Reset Conditions. . . . . Page 32
- Module Identification. . . . . Page 32
- Switching Channels . . . . . Page 33
- Recalling and Saving States. . . . . Page 35
- Detecting Error Conditions . . . . . Page 36
- Synchronizing the Relay Matrix Switch Module . . . . . Page 37
- Querying the Relay Matrix Switch Module. . . . . Page 37

All examples in this chapter use an HP-IB select code of 7, primary address of 09, and secondary address of 14 (LADDR = 112) for the matrix module.

Relay Matrix Switch Module Commands

Table 2-1 explains some of the commands used in this chapter. Refer to Chapter 3 for additional information.

Table 2-1. Relay Matrix Commands Used in Chapter 2

Command	Description
INITiate[:IMMediate]	Starts the scan sequence and closes the first channel in the *channel_list*.
[ROUTe:]CLOSe <*channel_list*>	Closes the channels in the *channel_list.*
[ROUTe:]CLOSe? <*channel_list*>	Queries the state of the channels in the *channel_ list*.
[ROUTe:]OPEN <*channel_list*>	Opens the channels in the *channel_list*.
[ROUTe:]OPEN? <*channel_list*>	Queries the state of channels in the *channel_ list*.
[ROUTe:]SCAN <*channel_list*>	Defines the *channel_list* to be scanned. Channels specified are closed one at a time.
TRIGger:SOURce <*source*> *source* = BUS | EXT | HOLD | IMM | TTLT | ECLT	Selects the trigger *source* to advance the scan.
*CLS	Clears switchbox status registers and error queue.
*RST	Resets the hardware to a known state.

Note that the commands in square brackets ([ ]) are implied and not sent with the command.

Power-on and Reset Conditions

When power is first applied to the relay matrix switch modules or *RST (reset) is executed, all relays are open. Table 2-2 lists the parameters and default values for the switchbox functions described in this chapter.

Table 2-2. *RST (Reset) Default Conditions and Values

Parameter	Default	Description
ARM:COUNt	1	Number of scanning cycles is 1.
TRIGger:SOURce	IMM	Will advance scanning cycles automatically.
INITiate:CONTinuous	OFF	Number of scanning cycles set by ARM:COUNt.
OUTPut[:EXTernal][:STATe]	OFF	Trigger output from EXTernal, TTLTrg, or ECLTrg sources is disabled.

Note that the commands in brackets ([ ]) are implied and are not sent with the command.

Module Identification

The following short program uses the *RST, *CLS, *IDN?, CTYP?, and CDES? commands to reset and identify the relay matrix switch modules.

```
10  DIM A$[50]; B$[50], C$[50]
20  OUTPUT 70914; "*RST; *CLS; *IDN?"
30  ENTER 70914; A$
40  OUTPUT 70914; "SYST:CDES? 1"
50  ENTER 70914; B$
60  OUTPUT 70914; "SYST:CTYP? 1"
70  ENTER 70914; C$
80  PRINT A$
90  PRINT B$
100 PRINT C$
110 END
```

A typical print will look like:

```
HEWLETT-PACKARD,SWITCHBOX,0,A.04.00
4x16 2-WIRE MATRIX
HEWLETT-PACKARD,E1469A,0,A.04.00
```

Switching Channels

Use CLOSe <*channel_list*> to close one or more relay matrix switch channels, and OPEN <*channel_list*> to open the channel(s). *channel_list* has the following forms:

- **HP E1468A only:** @ssrc where

  ss = card number (01-99)

  r = row number

  c = column number

- **HP E1469A only:** @ssrrcc where

  ss = card number (01-99)

  rr = row number

  cc = column number

Row and Column Definitions	
HP E1468A 8x8 Matrix r: 0 to 7 (one digit) c: 0 to 7 (one digit)	HP E1469A 4x16 Matrix rr: 00 to 03 (two digits) cc: 00 to 15 (two digits)

Refer to Chapter 3, [ROUTe:]OPEN and [ROUTe:]CLOSe, for additional information. To OPEN or CLOSe multiple channels, place a comma (**,**) between the channel numbers. For example, to close channels 10103 and 10201, execute CLOS 10103,10201. To OPEN or CLOSe a contiguous range of channels place a colon (**:**) between the first and last channel numbers.

The following program shows how to close and open row 2 (02), column 14 on an HP E1469A Relay Matrix Switch Module (card #1):

```
10  DISP "TEST E1469A MATRIX"
20  OUTPUT 70914; "ROUT:CLOS (@10214)"
30  OUTPUT 70914; "ROUT:OPEN (@10214)"
40  END
```

The following program sequences through each channel on an HP E1468A 8x8 Relay Matrix Switch Module.

```
10  DIM E$[128]
20  FOR I = 0 TO 7
30    FOR J = 0 TO 7
40      A = 100 + 10 * I + J
50        OUTPUT 70914; "ROUT:CLOS (@ ";A;")"
60        OUTPUT 70914; "ROUT:CLOS? (@100:177)"
70       ENTER 70914; E$
80     PRINT "CHANNEL CLOSED NOW"; E$
90       OUTPUT 70914; "ROUT:OPEN (@ ";A;")"
100    NEXT J
110 NEXT I
120 END
```

To use this program with the HP E1469A 4x16 Relay Matrix Switch Module, replace lines 20, 30, 40, and 60 with the following:

```
20  FOR I = 0 TO 3
30    FOR J = 0 TO 15
40      A = 10000 + 100 * I + J
60        OUTPUT 70914; "ROUT:CLOS? (@10000:10315)"
```

Query Open/Closed Channels

The CLOSe? <*channel_list*> and OPEN? <*channel_list*> commands return the current state of the specified channel. See Chapter 3, [ROUTe:]OPEN? and [ROUTe:]CLOSe? commands for additional information.

Recalling and Saving States

The *SAV *<numeric_state>* command stores the current state of the switchbox channels. Up to 10 states may be stored by specifying the *<numeric_state>* as an integer 0 through 9. The following states are stored:

- Channel relay states (open or closed)
- ARM:COUNt
- TRIGger:SOURce *<source>*
- OUTPut[:EXTernal][:STATe]
- INITiate:CONTinuous

The *RCL *<numeric_state>* command recalls the specified previously stored state. If the specified *<numeric_state>* does not exist, the relay matrix switch module configures to its power-on/reset states (refer to Table 2-2).

The following program shows how to save and recall relay matrix switch states:

```
10  !Close channels on the matrix module and save the state as number 5.
20  OUTPUT 70914; "CLOS (@10000:10015)"
30  OUTPUT 70914; "*SAV 5"
40  !Reset and clear the module, close other channels.
50  OUTPUT 70914; "*RST; *CLS"
60  OUTPUT 70914; "CLOS (@10113,10112,10200)"
70  !Recall the stored state.  Only the channels closed in the stored state.
80  !are closed.  All other channels in the switchbox are opened.
90  OUTPUT 70914; "*RCL 5"
100 END
```

Detecting Error Conditions

Use the SYST:ERR? command to poll the switchbox for errors. The following program attempts an illegal channel closure and polls for the error message:

```
10  DIM Err_num$[256]
20  OUTPUT 70914; "CLOS (@10500)"
30  OUTPUT 70914; "SYST:ERR?"
40  ENTER 70914; Err_num$
50  PRINT Err_num$
```

The following program uses an interrupt to signal the controller when an error occurs. Again, the SYST:ERR? command returns the error message.

```
10  !Call subprogram Errmsg if a matrix programming error occurs.
20  !Enable the computer to respond to the interrupt from the matrix.
30  ON INTR 7 CALL Errmsg
40  ENABLE INTR 7:2
50  !Unmask the Event Status bit in the matrix's Status Register (*SRE 32).
60  !Unmask the matrix error conditions in its Standard Event Status
70  !Register (*ESE 64).
80  OUTPUT 70914; "*SRE 32; *ESE 64"
90  OUTPUT 70914 ".  .  .
100 !Continue program execution.
ù
•
•
190 END
200 !When an error occurs, clear the matrix module to regain control.
210 !Execute a Serial Poll to clear the Service Request bit in the Status
220 !Register.  Read all error messages in the matrix error queue.
230 !Clear all bits in the matrix Standard Event Status Register.
240 SUB Errmsg
250 DIM Message$[256]
260 CLEAR 709
270 B = SPOLL (70914)
280 REPEAT
290 OUTPUT 70914; "SYST:ERR?"
300 ENTER 70914; Code, Message$
310 PRINT Code, Message$
320 UNTIL Code = 0
330 OUTPUT 70914; "*CLS"
340 STOP
350 SUBEND
```

Synchronizing the Relay Matrix Switch Module

The following example shows how to synchronize a relay matrix switch module with measurement instruments. In this example, the matrix module switches a signal to a multimeter. The program verifies that the channel is closed before the multimeter begins its measurement.

```
10  !Reset the module.
20  OUTPUT 70914; "*RST"
30  !Close a channel.
40  OUTPUT 70914; "CLOS (@10012)"
50  !Wait for operation complete.
60  OUTPUT 70914; "*OPC?"
70  ENTER 70914; Opc_value
80  !Test that the channel is closed.
90  OUTPUT 70914; "CLOS? (@10012)"
100 ENTER 70914; A
110 !When channel is closed, measure the voltage.
120 OUTPUT 70903; "MEAS:VOLT:DC?"
130 ENTER 70903; Meas_value
140 !Print the measured value.
150 PRINT Meas_value
160 END
```

Querying the Relay Matrix Switch Module

All query commands end with a "**?**". All data is sent to the output buffer where you can retrieve it into your computer. The following are valid query commands:

Number of Scanning Cycles:	ARM:COUN?
Channel Closed:	CLOS?
Scanning State:	INIT:CONT?
Channel Open:	OPEN?
ECL Trigger Output State:	OUTP:ECLT*n*?
External Trigger Output State:	OUTP:EXT?
TTL Trigger Output State:	OUTP:TTLT*n*?
Status Operation Enable:	STAT:OPER:ENAB?
Status Operation Event:	STAT:OPER[:EVEN]?
Module Description:	SYST:CDES? <*number*>
Module Type:	SYST:CTYP? <*number*>
System Error:	SYST:ERR?
Trigger Source:	TRIG:SOUR?

Chapter 3
Relay Matrix Switch Command Reference

About This Chapter

This chapter describes the Standard Commands for Programmable Instruments (SCPI) commands and the IEEE 488.2 Common commands for the HP E1468A and HP E1469A Relay Matrix Switch Modules.

See the *HP E1406 Command Module User's Manual* for additional information on SCPI and Common commands. This chapter contains the following sections:

- Command Types . . . . . Page 39
- SCPI Command Reference . . . . . Page 42
- IEEE 488.2 Common Command Reference . . . . . Page 67
- SCPI Commands Quick Reference . . . . . Page 68

Command Types

Commands are separated into two types: IEEE 488.2 Common commands and SCPI commands.

Common Command Format

The IEEE 488.2 standard defines the Common commands that perform functions like reset, self-test, status byte query, and so forth. Common commands are four or five characters in length, always begin with an asterisk (*****), and may include one or more parameters. The command keyword is separated from the first parameter by a space character. Some examples of Common commands are shown below:

*RST, *ESE <*mask*>, *STB?

SCPI Command Format

SCPI commands perform functions like closing switches, making measurements, and querying instrument states or retrieving data. A subsystem command structure is a hierarchical structure that usually consists of a top-level (or root) command, one or more lower-level commands, and their parameters.

The following example shows part of a typical subsystem:

```
[ROUTe:]
    CLOSe <channel_list>
    SCAN <channel_list>
        MODE?
```

[ROUTe:] is the root command, CLOSe and SCAN are second-level commands with parameters, and :MODE? is a third-level command. [ROUTe:] is also an implied command and is, therefore, optional.

Command Separator

A colon (**:**) always separates one command from the next lower-level command as shown below:

```
[ROUTe:]SCAN:MODE?
```

Colons separate the root command from the second-level command ([ROUTe:]SCAN) and the second level from the third level (SCAN:MODE?).

Abbreviated Commands

The command syntax shows most commands as a mixture of upper- and lowercase letters. The uppercase letters indicate the abbreviated spelling for the command. For shorter program lines, send the abbreviated form. For better program readability, you may send the entire command. The instrument will accept either the abbreviated form or the entire command.

For example, if the command syntax shows DIAGnostic, then DIAG and DIAGNOSTIC are both acceptable forms. Other forms of DIAGnostic, such as DIAGN or DIAGNOS will generate an error. You may use upper- or lowercase letters. Therefore, DIAGNOSTIC, diagnostic, and DiAgNoStIc are all acceptable.

Implied Commands

Implied commands appear in square brackets (**[ ]**) in the command syntax. ***The brackets are not part of the command and are not sent to the instrument.*** Suppose you send a second-level command but do not send the preceding implied command. In this case, the instrument assumes you intended to use the implied command and it responds as if you had sent it. Examine the [SOURce] subsystem shown below:

```
[SOURce:]
    PULSe
        :COUNt
        :COUNt?
        :PERiod
        :PERiod?
```

The root command [SOURce:] is an implied command. To set the instrument's pulse count to 25, you can send either of the following command statements:

```
SOUR:PULS:COUN 25      or      PULS:COUN 25
```

Variable Command Syntax

Some commands have what appears to be a variable syntax. For example:

OUTP:ECLT*n* and OUTP:TTLT*n*

In these commands, the *n* is replaced by a number. No space is left between the command and the number because the number is not a parameter. The number is part of the command syntax. In the case of OUTP:ECLT*n*, *n* can range from 0 to 1. In OUTP:TTLT*n*, *n* can range from 0 through 7.

Parameter Types

The following list contains explanations and examples of parameter types you will see later in this chapter.

- **Boolean parameters** represent a single binary condition that is either true or false (for example, ON, OFF, 1, 0). Any non-zero value is considered true.
- **Discrete Parameters** selects from a finite number of values. These parameters use mnemonics to represent each valid setting. An example is the TRIGger:SOURce <*source*> command where *source* can be BUS, EXTernal, HOLD, IMMediate, ECLTrg*n*, or TTLTrg*n*.
- **Numeric Parameters** are commonly used decimal representations of numbers including optional signs, decimal points, and scientific notation (for example, 123, 123E2, -123, -1.23E2, .123, 1.23E-2, 1.23000E- 01). Special cases include MIN, MAX, DEFault, and INFinity.
- **Optional Parameters** are shown within square brackets (**[ ]**). The brackets are not part of the command and are not sent to the instrument. If you do not specify a value for an optional parameter, the instrument chooses a default value. For example, consider the ARM:COUNt? [MIN|MAX] command. If you send the command without specifying a parameter, the present ARM:COUNt value is returned. If you send the MIN parameter, the command returns the minimum count available. If you send the MAX parameter, the command returns the maximum count available. Be sure to place a space between the command and the parameter.

Linking Commands

Linking IEEE 488.2 Common Commands with SCPI Commands. Use a semicolon (**;**) between the commands. For example:

*RST;OUTP ON or TRIG:SOUR HOLD;*TRG

Linking Multiple SCPI commands. Use both a semicolon (**;**)and a colon (**:**) between the commands. For example:

ARM:COUN 1;:TRIG:SOUR EXT

SCPI Command Reference

This section describes the Standard Commands for Programmable Instruments (SCPI) commands for the relay matrix switch modules. Commands are listed alphabetically by subsystem and within each subsystem.

- ABORt commands . . . . . Page 43
- ARM commands . . . . . Page 44
- INITiate commands. . . . . Page 46
- OUTPut commands . . . . . Page 48
- [ROUTe:] commands . . . . . Page 52
- STATus commands . . . . . Page 57
- SYSTem commands. . . . . Page 61
- TRIGger commands . . . . . Page 64

The ABORt command subsystem stops a scan in progress when the scan is enabled via the interface and the trigger source is TRIGger:SOURce BUS or TRIGger:SOURce HOLD.

Subsystem Syntax

```
ABORt
```

Comments

- **ABORt Actions:** ABORt stops the scan and invalidates the current *channel_list*.
- **Stopping Scan Enabled Via Interface:** When a scan is enabled via an interface, an interface CLEAR command (CLEAR 7) can be used to stop the scan. When the scan is enabled via the interface and TRIG:SOUR BUS or HOLD is set, you can use ABORt to stop the scan.
- **Restarting a Scan:** Use the INIT command to restart the scan.
- **Related Commands:** ARM, INITiate:CONTinuous, [ROUTe:]SCAN, TRIGger

Example

Stopping a Scan with ABORt.

This example stops a (continuous) scan in progress.

```
TRIG:SOUR BUS                 ! *TRG command is trigger source.
INIT:CONT ON                  ! Set continuous scanning.
SCAN (@10000:10003)           ! Scan channels 00-03.
INIT                          ! Start scan, close channel 00.
   .
   .
ABOR                          ! Abort scan in progress.
```

The ARM subsystem selects the number of scanning cycles (1 to 32,767) for each INITiate command.

Subsystem Syntax

```
ARM
    :COUNt <number> MIN | MAX
    :COUNt? [MIN | MAX]
```

:COUNt

ARM:COUNt <*number*> MIN | MAX allows scanning cycles to occur a multiple of times (1 to 32,767) with one INITiate command when INITiate:CONTinuous OFF | 0 is set. MIN sets 1 cycle and MAX sets 32,767 cycles.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*number*>	numeric	1-32,767 | MIN | MAX	1

Comments

- **Number of Scans:** Use only values between 1 and 32,767 for the number of scanning cycles.
- **Related Commands:** ABORt, INITiate[:IMMediate]
- ***RST Condition:** ARM:COUNt 1

Example

Setting Ten Scanning Cycles.

This example sets a relay matrix switch module for 10 scans of channels 00 through 03.

```
ARM:COUN 10                 ! Set 10 scans per INIT command.
SCAN (@10000:10003)         ! Scan channels 00-03.
INIT                        ! Start scan, close channel 00.
```

ARM:COUNt? [MIN | MAX] returns the current number of scanning cycles set by ARM:COUNt. The current number of scan cycles is returned when MIN or MAX is not supplied. With MIN or MAX as a parameter, MIN returns *1* and MAX returns *32767*.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
MIN | MAX	numeric	MIN = 1, MAX = 32,767	current cycles

Comments

- **Related Commands:** INITiate[:IMMediate]

Example

Query Number of Scans.

This example sets a switchbox for 10 scanning cycles and queries the number of scan cycles set. The ARM:COUN? command returns *10*.

```
ARM:COUN 10          ! Set 10 scans per INIT command.
ARM:COUN?            ! Query number of scans.
```

The INITiate command subsystem selects continuous scanning cycles and starts the scanning cycle.

Subsystem Syntax

```
INITiate
    :CONTinuous <mode>
    :CONTinuous?
    [:IMMediate]
```

:CONTinuous

INITiate:CONTinuous <*mode*> enables or disables continuous scanning cycles for the switchbox.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*mode*>	boolean	0 | 1 | OFF | ON	OFF | 0

Comments

- **Continuous Scanning Operation:** Continuous scanning is enabled with the INITiate:CONTinuous ON or INITiate:CONTinuous 1 command. Sending the INITiate[:IMMediate] command closes the first channel in the channel list. Each trigger from the source specified by the TRIGger:SOURce command advances the scan through the channel list. A trigger at the end of the channel list closes the first channel in the channel list and the scan cycle repeats.
- **Non-Continuous Scanning Operation:** Non-continuous scanning is enabled with the INITiate:CONTinuous OFF or INITiate:CONTinuous 0 command. Sending the INITiate[:IMMediate] command closes the first channel in the channel list. Each trigger from the source specified by the TRIGger:SOURce command advances the scan through the channel list. At the end of the scanning cycle, the last channel in the channel list is closed and the scanning cycle stops.
- **Stopping Continuous Scan:** See the ABORt command.
- **Related Commands:** ABORt, ARM:COUNt, TRIGger:SOURce
- ***RST Condition:** INITiate:CONTinuous OFF | 0

Example **Enabling Continuous Scanning.**

This example enables continuous scanning of channels 00 through 03 of a single-module switchbox. Since TRIGger:SOURce IMMediate (default) is set, use an interface clear command (such as CLEAR) to stop the scan.

```
INIT:CONT ON                    ! Enable continuous scanning.
SCAN (@10000:10003)             ! Scan channels 00-03.
INIT                            ! Start scan cycle, close
                                channel 00.
```

:CONTinuous?

INITiate:CONTinuous? queries the scanning state. With continuous scanning enabled, the command returns *1*. With continuous scanning disabled, the command returns *0*.

Example

Query Continuous Scanning State.

This example enables continuous scanning of a switchbox and queries the state. Since continuous scanning is enabled, INIT:CONT? returns *1*.

```
INIT:CONT ON                    ! Enable continuous scanning.
INIT:CONT?                      ! Query continuous scanning state.
```

[:IMMediate]

INITiate[:IMMediate] starts the scanning process and closes the first channel in the channel list. Successive triggers from the source selected by the TRIGger:SOURce command advance the scan through the channel list.

Comments

- **Starting the Scanning Cycle:** The INITiate[:IMMediate] command starts scanning by closing the first channel in the channel list. Each trigger received advances the scan to the next channel in the channel list. An invalid channel list definition causes an error (see [ROUTe:]SCAN).
- **Stopping Scanning Cycles:** See ABORt.

Example

Enabling a Single Scan.

This example enables a single scan of channels 00 through 03 of a single-module switchbox. The trigger source to advance the scan is immediate (internal) triggering set with (default) TRIGger:SOURce IMMediate.

```
SCAN (@10000:10003)             ! Scan channels 00-03.
INIT                            ! Begin scan, close channels 00 (use
                                immediate triggering).
```

The OUTPut command subsystem selects the source of the output trigger generated when a channel is closed during a scan. The selected output can be enabled, disabled, and queried. The three available outputs are the ECLTrg, TTLTrg trigger buses as well as the command module
(HP E1406A) front panel Trig Out port.

Subsystem Syntax

```
OUTPut
    :ECLTrgn      (:ECLTrg0 or :ECLTrg1)
        [:STATe] <mode>
        [:STATe]?
    [:EXTernal]
        [:STATe] <mode>
        [:STATe]?
    :TTLTrgn      (:TTLTrg0 through :TTLTrg7)
        [:STATe] <mode>
        [:STATe]?
```

:ECLTrg[:STATe]

OUTPut:ECLTrg*n*[:STATe] <*mode*> selects and enables which ECL Trigger bus line (0 or 1) will output a trigger when a channel is closed during a scan. This is also used to disable a selected ECL Trigger bus line. *n* specifies the ECL Trigger bus line (0 or 1) and *mode* enables (ON or 1) or disables (OFF or 0) the specified ECLTrg bus line.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
n	numeric	0 or 1	N/A
<*mode*>	boolean	0 | 1 | OFF | ON	OFF | 0

Comments

- **Enabling ECL Trigger Bus:** When enabled, a pulse is output from the selected ECL Trigger bus line (0 or 1) after each channel is closed during a scan. If disabled, a pulse is not output. The output is a negative going pulse.
- **ECL Trigger Bus Line Shared by Switchboxes:** Only one switchbox configuration can use the selected trigger at a time. When enabled, the selected ECL Trigger bus line (0 or 1) is pulsed by the switchbox each time a scanned channel is closed. To disable the output for a specific switchbox, send the OUTPut:ECLTrg*n* OFF or 0 command for that switchbox.

- **One Output Selected at a Time:** Only one output (ECLTrg 0 or 1; TTLTrg 0, 1, 2, 3, 4, 5, 6, or 7; or EXTernal) can be enabled at one time. Enabling a different output source will automatically disable the active output. For example, if TTLTrg1 is the active output, and TTLTrg4 is enabled, TTLTrg1 will become disabled and TTLTrg4 will become the active output.

- **Related Commands:** [ROUTe:]SCAN, TRIGger:SOURce, OUTPut:ECLTrg[:STATe]?

- ***RST Condition:** OUTPut:ECLTrg[:STATe] OFF (disabled).

Example **Enabling ECL Trigger Bus Line 0.**

```
OUTP:ECLT0:STAT 1          ! Enable ECL Trigger bus line 0 to output
                           pulse after each scanned channel is closed.
```

:ECLTrg[:STATe]?

OUTPut:ECLTrg[:STATe]? queries the present state of the specified ECL Trigger bus line. The command returns *1* if the specified ECLTrg bus line is enabled or *0* if disabled.

Example **Query ECL Trigger Bus Enable State.**

This example enables ECL Trigger bus line 0 and queries the enable state. The OUTPut:ECLTrg*n*? command returns *1* since the port is enabled.

```
OUTP:ECLT0:STAT  1         ! Enable ECL Trigger bus line 0.
OUTP:ECLT0?                ! Query bus enable state.
```

[:EXTernal][:STATe]

OUTPut[:EXTernal][:STATe] <*mode*> enables or disables the Trig Out port on the HP E1406A Command Module to output a trigger when a channel is closed during a scan. ON | 1 enables the port and OFF | 0 disables the port.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*mode*>	boolean	0 | 1 | OFF | ON	OFF | 0

Comments

- **Enabling Trig Out Port:** When enabled, a pulse is output from the Trig Out port after each scanned switchbox channel is closed. If disabled, a pulse is not output from the port after channel closures. The output is a negative going pulse.

- **Trig Out Port Shared by Switchboxes:** Only one switchbox configuration can use the selected trigger at a time. When enabled, the Trig Out port is pulsed by the switchbox each time a scanned channel is closed. To disable the output for a specific switchbox, send the OUTP OFF or 0 command for that switchbox.

- **One Output Selected at a Time:** Only one output (ECLTrg 0 or 1; TTLTrg 0, 1, 2, 3, 4, 5, 6, or 7; or EXTernal) can be enabled at one time. Enabling a different output source will automatically disable the active output. For example, if TTLTrg1 is the active output, and TTLTrg4 is enabled, TTLTrg1 will become disabled and TTLTrg4 will become the active output.

- **Related Commands:** [ROUTe:]SCAN, TRIGger:SOURce, OUTPut[:EXTernal][:STATe]?

- ***RST Condition:** OUTPut[:EXTernal][:STATe] OFF (disabled).

Example **Enabling Trig Out Port.**

```
OUTP:EXT 1                    ! Enable Trig Out port to output pulse after
                              each scanned channel is closed.
```

[:EXTernal][:STATe]?

OUTPut[:EXTernal][:STATe]? queries the present state of the Trig Out port. The command returns *1* if the port is enabled or *0* if disabled.

Example **Query Trig Out Port Enable State.**

This example enables the Trig Out port and queries the enable state. The OUTPut? command returns *1* since the port is enabled.

```
OUTP:EXT ON                   ! Enable Trig Out port.
OUTP:EXT?                     ! Query port enable state.
```

:TTLTrg[:STATe]

OUTPut:TTLTrg*n*[:STATe] <*mode*> selects and enables which TTL Trigger bus line (0 to 7) will output a trigger when a channel is closed during a scan. This is also used to disable a selected TTL Trigger bus line. *n* specifies the TTL Trigger bus line (0 to 7) and *mode* enables (ON or 1) or disables (OFF or 0) the specified TTL Trigger bus line.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
n	numeric	0 to 7	N/A
<*mode*>	boolean	0 | 1 | OFF | ON	OFF | 0

Comments

- **Enabling TTL Trigger Bus:** When enabled, a pulse is output from the selected TTL Trigger bus line (0 to 7) after each channel in the switchbox is closed during a scan. If disabled, a pulse is not output. The output is a negative going pulse.

- **TTL Trigger Bus Line Shared by Switchboxes:** Only one switchbox configuration can use the selected TTL Trigger at a time. When enabled, the selected TTL Trigger bus line (0 to 7) is pulsed by the switchbox each time a

scanned channel is closed. To disable the output for a specific switchbox, send the OUTPut:TTLTrg*n* OFF or 0 command for that switchbox.

- **One Output Selected at a Time:** Only one output (ECLTrg 0 or 1; TTLTrg 0, 1, 2, 3, 4, 5, 6, or 7; or EXTernal) can be enabled at one time. Enabling a different output source will automatically disable the active output. For example, if TTLTrg1 is the active output, and TTLTrg4 is enabled, TTLTrg1 will become disabled and TTLTrg4 will become the active output.
- **Related Commands:** [ROUTe:]SCAN, TRIGger:SOURce, OUTPut:TTLTrg[:STATe]?
- ***RST Condition:** OUTPut:TTLTrg[:STATe] OFF (disabled).

Example **Enabling TTL Trigger Bus Line 7.**

```
OUTP:TTLT7:STAT 1          ! Enable TTL Trigger bus line 7 to output
                           pulse after each scanned channel is closed.
```

:TTLTrg[:STATe]?

OUTPut:TTLTrg[:STATe]? queries the present state of the specified TTL Trigger bus line. The command returns *1* if the specified TTLTrg bus line is enabled or *0* if disabled.

Example **Query TTL Trigger Bus Enable State.**

This example enables TTL Trigger bus line 7 and queries the enable state. The OUTPut:TTLTrg*n*? command returns *1* since the port is enabled.

```
OUTP:TTLT7:STAT 1          ! Enable TTL Trigger bus line 7.
OUTP:TTLT7?                ! Query bus enable state.
```

The [ROUTe:] command subsystem controls switching and scanning operations for relay matrix switch modules in a switchbox.

Note This command opens all previously closed relays, therefore it should be the first relay configuration command.

Subsystem Syntax

```
[ROUTe:]
    CLOSe <channel_list>
    CLOSe? <channel_list>
    OPEN <channel_list>
    OPEN? <channel_list>
    SCAN <channel_list>
```

CLOSe

[ROUTe:]CLOSe <*channel_list*> closes the relay matrix channels specified by *channel_list*.

On the HP E1468A, *channel_list* has the form (@ssrc) where
ss = card number (01-99),
r = row number, and
c = column number.

On the HP E1469A, *channel_list* has the form (@ssrrcc) where
ss = card number (01-99),
rr = row number, and
cc = column number.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*channel_list*>	numeric	E1468A:E1469A: r = 0 to 7rr = 00 to 03 c = 0 to 7cc = 00 to 15	N/A

Comments

- **Closing Channels (HP E1468A Only):**
  To close a single channel use [ROUT:]CLOS (@ssrc) ;
  for multiple channels use [ROUT:]CLOS (@ssrc,ssrc,...) ;
  for sequential channels use [ROUT:]CLOS (@ssrc:ssrc) ;
  for groups of sequential channels use
  [ROUT:]CLOS (@ssrc:ssrc,ssrc:ssrc) ;
  or any combination.

Closure order for multiple channels with a single command is not guaranteed.

- **Closing Channels (HP E1469A Only):**
  To close a single channel use [ROUT:]CLOS (@ssrrcc) ;
  for multiple channels use [ROUT:]CLOS (@ssrrcc,ssrrcc,...) ;
  for sequential channels use [ROUT:]CLOS (@ssrrcc:ssrrcc) ;
  for groups of sequential channels use
  [ROUT:]CLOS (@ssrrcc:ssrrcc,ssrrcc:ssrrcc) ;
  or any combination.

Closure order for multiple channels with a single command is not guaranteed.

- **Related Commands:** [ROUTe:]OPEN, [ROUTe:]CLOSe?
- ***RST Condition:** All channels open.

Example **Closing Relay Matrix Switch Module Channels.**

This example closes channels 10100 and 20013 of a two-module switchbox (card numbers 01 and 02).

```
CLOS (@10100,20013)          ! Close channels 10100 and 20013.  10100
                             closes row 01, column 00 of card #1 and
                             20013 closes row 00 column 13 on card #2.
```

CLOSe?

[ROUTe:]CLOSe? <*channel_list*> returns the current state of the channel(s) queried. *channel_list* has the form (@ssrc) or (@ssrrcc) (see [ROUTe:]CLOSe for definition). The command returns *1* if channel(s) are closed or returns *0* if channel(s) are open.

Comments

- **Query is Software Readback:** The [ROUTe:]CLOSe? command returns the current software state of the channel(s) specified. It does not account for relay hardware failures. A maximum of 127 channels at a time can be queried for a multi-module switchbox.

Example **Query Channel Closure.**

This example closes channels 10100 and 20013 of a two-module switchbox and queries channel closure. Since the channels are programmed to be closed 1, 1 is returned as a string.

```
CLOS (@10100,20013)
CLOS? (@10100,20013)
```

OPEN

[ROUTe:]OPEN <*channel_list*> opens the relay matrix switch channels specified by *channel_list*.

On the HP E1468A, *channel_list* has the form (@ssrc) where
ss = card number (01-99),
r = row number, and
c = column number.
On the HP E1469A, *channel_list* has the form (@ssrrcc) where
ss = card number (01-99),
rr = row number, and
cc = column number.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*channel_list*>	numeric	E1468A:E1469A: r = 0 to 7rr = 00 to 03 c = 0 to 7cc = 00 to 15	N/A

Comments

- **Opening Channels (HP E1468A Only):**
  To open a single channel use [ROUT:]OPEN (@ssrc) ;
  for multiple channels use [ROUT:]OPEN (@ssrc,ssrc,...) ;
  for sequential channels use [ROUT:]OPEN (@ssrc:ssrc) ;
  for groups of sequential channels use
  [ROUT:]OPEN (@ssrc:ssrc,ssrc:ssrc) ;
  or any combination.

Opening order for multiple channels with a single command is not guaranteed.

- **Opening Channels (HP E1469A Only):**
  To open a single channel use [ROUT:]OPEN (@ssrrcc) ;
  for multiple channels use [ROUT:]OPEN (@ssrrcc,ssrrcc,...) ;
  for sequential channels use [ROUT:]OPEN (@ssrrcc:ssrrcc) ;
  for groups of sequential channels use
  [ROUT:]OPEN (@ssrrcc:ssrrcc,ssrrcc:ssrrcc) ;
  or any combination.

Opening order for multiple channels with a single command is not guaranteed.

- **Related Commands:** [ROUTe:]CLOSe, [ROUTe:]OPEN?
- ***RST Condition:** All channels open.

Example

Opening Channels.

This example opens channels 10100 and 20013 of a two-module switchbox (card numbers 01 and 02).

```
OPEN (@10100,20013)          ! Open channels 10100 and 20013.
```

OPEN?

[ROUTe:]OPEN? <*channel_list*> returns the current state of the channel(s) queried. *channel_list* has the form (@ssrc) or (@ssrrcc) (see [ROUTe:]OPEN for definition). The command returns *1* if channel(s) are open or returns *0* if channel(s) are closed.

Comments

- **Query is Software Readback:** The [ROUTe:]OPEN? command returns the current software state of the channels specified. It does not account for relay hardware failures. A maximum of 127 channels at a time can be queried for a multi-module switchbox.

Example

Query Channel Open State.

This example opens channels 10100 and 20013 of a two-module switchbox and queries channel 20013 state. Since channel 20013 is programmed to be open, *1* is returned.

```
OPEN (@10100,20013)
OPEN? (@20013)
```

SCAN

[ROUTe:]SCAN <*channel_list*> defines the channels to be scanned.

On the HP E1468A, *channel_list* has the form (@ssrc) where
ss = card number (01-99),
r = row number, and
c = column number.
On the HP E1469A, *channel_list* has the form (@ssrrcc) where
ss = card number (01-99),
rr = row number, and
cc = column number.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*channel_list*>	numeric	E1468A:E1469A: r = 0 to 7rr = 00 to 03 c = 0 to 7cc = 00 to 15	N/A

Comments

- **Defining Scan List:** When [ROUTe:]SCAN is executed, the *channel_list* is checked for valid card and channel numbers.
  An error is generated for an invalid *channel_list*.

- **Scanning Channels (HP E1468A Only):**
  To scan a single channel use [ROUT:]SCAN (@ssrc) ;
  for multiple channels use [ROUT:]SCAN (@ssrc,ssrc,...) ;
  for sequential channels use [ROUT:]SCAN (@ssrc:ssrc) ;
  for groups of sequential channels use

[ROUT:]SCAN (@ssrc:ssrc,ssrc:ssrc) ;
or any combination.

- **Scanning Channels (HP E1469A Only):**
  To scan a single channel use [ROUT:]SCAN (@ssrrcc) ;
  for multiple channels use [ROUT:]SCAN (@ssrrcc,ssrrcc,...) ;
  for sequential channels use [ROUT:]SCAN (@ssrrcc:ssrrcc) ;
  for groups of sequential channels use
  [ROUT:]SCAN (@ssrrcc:ssrrcc,ssrrcc:ssrrcc) ;
  or any combination.

- **Scanning Operation:** When a valid *channel_list* is defined, INITiate[:IMMediate] begins the scan and closes the first channel in the *channel_list*. Successive triggers from the source specified by TRIGger:SOURce advance the scan through the *channel_list*. At the end of the scan, the last trigger opens the last channel.

- **Stopping Scan:** See ABORt.

- **Related Commands:** TRIGger:SOURce

- ***RST Condition:** All channels open.

The STATus subsystem reports the bit values of the Operation Status Register (in the command module). It also allows you to unmask the bits you want reported from the Standard Event Register and to read the summary bits from the Status Byte register.

Subsystem Syntax

```
STATus
    :OPERation
        :CONDition?
        :ENABle <unmask>
        :ENABle?
        [:EVENt?]
    :PRESet
```

The STATus system contains four software registers (that is, they reside in a SCPI driver, not in the hardware), two of which are under IEEE 488.2 control; the Event Status Register (*ESE?) and the Status Byte Register (*STB?). The Operational Status bit (OPR), Service Request bit (RQS), Event Summary bit (ESB), Message Available bit (MAV) and Questionable Data bit (QUE) in the Status Byte Register (bits 7, 6, 5, 4 and 3 respectively) can be queried with the *STB? command. Use the *ESE? command to query the *unmask* value for the Event Status register (the bits you want logically OR'd into the summary bit). The registers are queried using decimal weighted bit values. The decimal equivalents for bits 0 through 15 are included in the figure which follows.

A numeric value of 256 executed in a STAT:OPER:ENABle <*unmask*> command allows only bit 8 to generate a summary bit. The decimal value for bit 8 is 256.

The decimal values are also used in the inverse manner to determine which bits are set from the total value returned by an EVENt or CONDition query. The switch module driver exploits only bit 8 of the Operation Status register. This bit is called the Scan Complete bit which is set whenever a scan operation completes. Since completion of a scan operation is an event in time, you will find that bit 8 will never appear set when STAT:OPER:COND? is queried. However, you can find bit 8 set with the STAT:OPER[:EVENt]? query command.

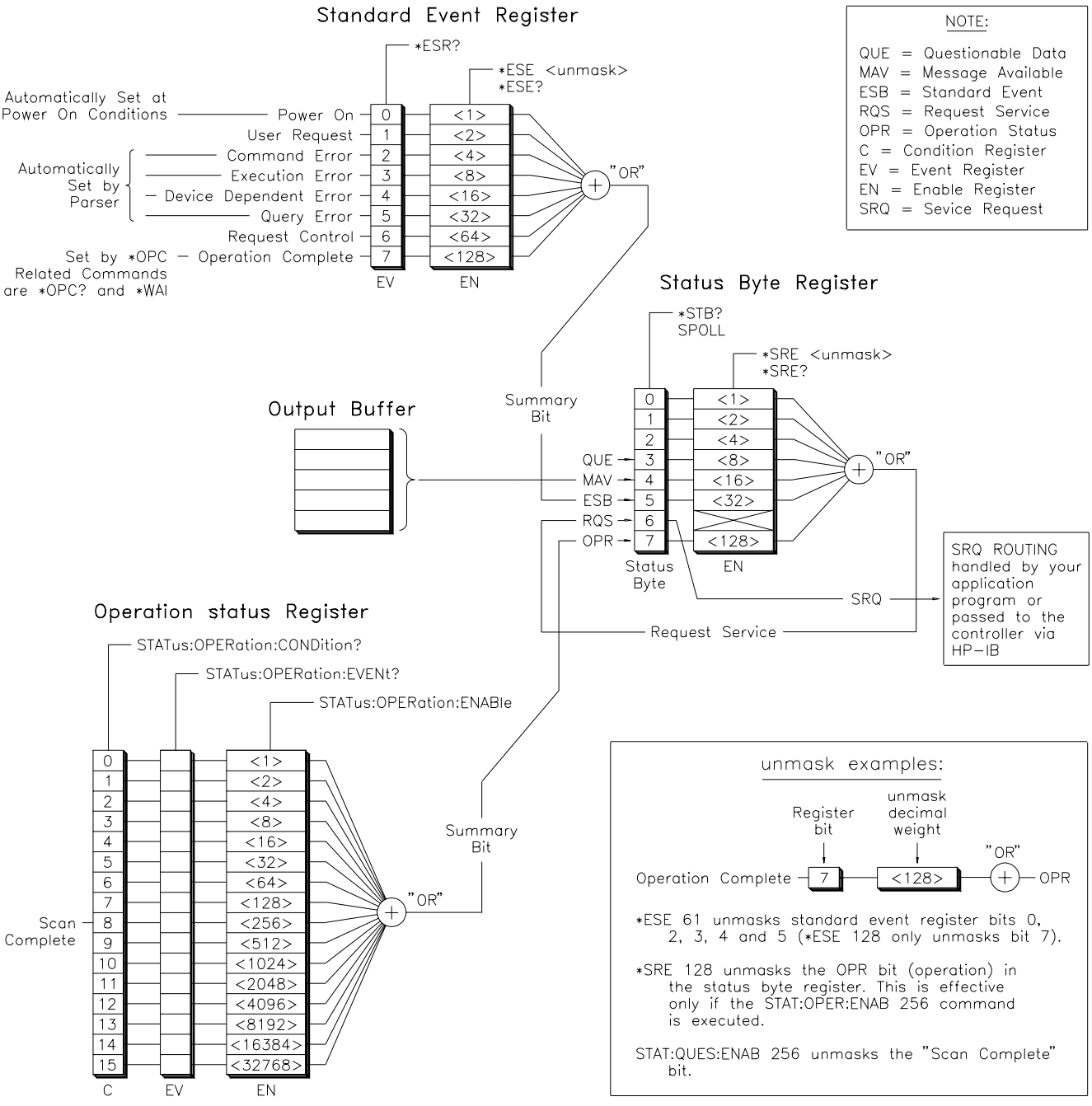


Figure 3-17. HP E1468A/E1469A Status System Register Diagram

:OPERation :CONDition?

STATus:OPERation:CONDition? returns the state of the Condition Register in the Operation Status Group. The state represents conditions which are part of the instrument's operation. The switch module driver does not set bit 8 in this register (see STATus:OPERation[:EVENt]?).

:OPERation:ENABle

STATus:OPERation:ENABle <*unmask*> sets an enable mask to allow events recorded in the Event Register to send a summary bit to the Status Byte Register (bit 7). For multiplexer modules, when bit 8 in the Operation Status Register is set to 1 and that bit is enabled by the STAT:OPER:ENABle command, bit 7 in the Status Register is set to 1.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*unmask*>	numeric	0 through 65,535	N/A

Comments

- **Setting Bit 7 of the Status Byte Register:** STATus:OPERation:ENABle 256 sets bit 7 of the Status Byte Register to 1 after bit 8 of the Operation Status Register is set to 1.
- **Related Commands:** [ROUTe:]SCAN

Example

Enabling Operation Status Register Bit 8.

```
STAT:OPER:ENAB 256          ! Enables bit 8 of the Operation Status
                            Enable Register to be reported to bit 7
                            (OPR) in the Status Register.
```

:OPERation:ENABle?

STATus:OPERation:ENABle? returns the bit value of the Operation Status Register.

Comments

- **Output Format:** Returns a decimal weighted value from 0 to 65,535 indicating which bits are set to true.
- **Maximum Value Returned:** The value returned is the value set by the STAT:OPER:ENAB <*unmask*> command. However, the maximum decimal weighted value used in this module is 256 (bit 8 set to true).

Example

Query the Operation Status Enable Register.

```
STAT:OPER:ENAB?             ! Queries the Operation Status Enable
                            Register.
```

:OPERation[:EVENt]?

STATus:OPERation[:EVENt]? returns which bits in the Event Register (Operation Status Group) are set. The Event Register indicates when there has been a time-related instrument event.

Comments

- **Setting Bit 8 of the Operation Status Register:** Bit 8 (Scan Complete) is set to 1 after a scanning cycle completes. Bit 8 returns to 0 (zero) after sending the STATus:OPERation[:EVENt]? command.
- **Returned Data After Sending the STATus:OPERation[:EVENt]? Command:** The command returns *+256* if bit 8 of the Operation Status Register is set to 1. The command returns *+0* if bit 8 of the Operation Status Register is set to 0.
- **Event Register Cleared:** Reading the Event Register with the STATus:OPERation[:EVENt]? command clears it.
- **ABORting a Scan:** Aborting a scan will leave bit 8 set to 0.
- **Related Commands:** [ROUTe:]SCAN

Example

Reading the Operation Status Register After a Scanning Cycle.

```
STAT:OPER?                  ! Returns the bit values of the Operation
                              Status Register.
read the register value     ! +256 shows bit 8 is set to 1;
                              +0 shows bit 8 is set to 0.
```

:PRESet

STATus:PRESet affects only the Enable Register by setting all Enable Register bits to 0. It does not affect either the "status byte" or the "standard event status". PRESet does not clear any of the Event Registers.

The SYSTem subsystem returns the error numbers and error messages in the error queue of a switchbox and returns the types and descriptions of modules (cards) in a switchbox.

Subsystem Syntax

```
SYSTem
    :CDEScription? <number>
    :CPON <number> | ALL
    :CTYPe? <number>
    :ERRor?
```

:CDEScription?

SYSTem:CDEScription? <*number*> returns the description of a selected module (card) in a switchbox.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*number*>	numeric	1 through 99	N/A

Comments

- **8x8 Relay Matrix Module Description:** The SYSTem:CDEScription? <*number*> command returns:

  ```
  8x8 Relay Matrix
  ```

- **4x16 Relay Matrix Module Description:** The SYST:CDEScription? <*number*> command returns:

  ```
  4x16 Relay Matrix
  ```

Example **Reading the Description of a Card #l Module.**

```
SYST:CDES? 1                    ! Returns the description.
```

:CPON

SYSTem:CPON <*number*> | ALL sets the selected module (card) in a switchbox to its power-on state.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*number*>	numeric	1 through 99	N/A

Comments

Matrix Module Power-On State: The power-on state is all channels (relays) open. Note that *RST opens all channels of all modules in a switchbox while SYSTem:CPON <*number*> opens the channels in only the module (card) specified in the command.

Example

Setting Card #1 Module to its Power-On State.

```
SYST:CPON 1                    ! Sets module #1 to power-on state.
```

:CTYPe?

SYSTem:CTYPe? <*number*> returns the module (card) type of a selected module in a switchbox.

Parameters

Parameter Name	Parameter Type	Range of Values	Default Value
<*number*>	numeric	1 through 99	N/A

Comments

- **8x8 Relay Matrix Module Model Number:** The SYSTem:CTYPe? <*number*> command returns:

  ```
  HEWLETT-PACKARD,El468A,0,A.02.00
  ```

  where the *0* after E1468A is the module serial number (always 0) and *A.02.00* is an example of the module revision code number.

- **4x16 Relay Matrix Switch Module Model Number:** The SYSTem:CTYPe? <*number*> command returns:

  ```
  HEWLETT-PACKARD,El469A,0,A.04.00
  ```

  where the *0* after E1469A is the module serial number (always 0) and *A.04.00* is an example of the module revision code number.

Example

Reading the Model Number of a Card #l Module.

```
SYST:CTYP? 1                   ! Return the model number.
```

:ERRor?

SYSTem:ERRor? returns the error numbers and corresponding error messages in the error queue of a switchbox. See Appendix C for a listing of some switchbox error numbers and messages.

Comments

- **Error Numbers/Messages in the Error Queue:** Each error generated by a switchbox stores an error number and corresponding error message in the error queue. The error message can be up to 255 characters long.
- **Clearing the Error Queue:** An error number/message is removed from the queue each time the SYSTem:ERRor? command is sent. The errors are cleared first-in, first-out. When the queue is empty, each following SYSTem:ERRor? command returns *0*, "No error". To clear all error numbers/messages in the queue, execute the *CLS command.
- **Maximum Error Numbers/Messages in the Error Queue:** The queue holds a maximum of 30 error numbers/messages for each switchbox. If the queue overflows, the last error number/message in the queue is replaced by *-350*, "Too many errors". The least recent error numbers/messages remain in the queue and the most recent are discarded.

Example

Reading the Error Queue.

```
SYST:ERR?                    ! Query the error queue.
```

The TRIGger command subsystem controls the triggering operation of relay matrix modules in a switchbox.

Subsystem Syntax

```
TRIGger
    [:IMMediate]
    :SOURce <source>
    :SOURce?
```

[:IMMediate]

TRIGger[:IMMediate] causes a trigger event to occur when the defined trigger *source* is TRIGger:SOURce BUS or TRIGger:SOURce HOLD.

Comments

- **Executing the TRIGger[:IMMediate] Command:** A channel list must be defined with [ROUTe:]SCAN <*channel_list*> and an INITiate[:IMMediate] command must be executed before TRIGger[:IMMediate] will execute.
- **BUS or HOLD Source Remains:** If selected, the TRIGger:SOURce BUS or TRIGger:SOURce HOLD commands remain in effect after triggering a switchbox with the TRIGger[:IMMediate] command.
- **Related Commands:** INITiate, [ROUTe:]SCAN

Example

Advancing Scan Using TRIGger Command.

This example scans a single-module switchbox from channel 00 through 03. Since TRIGger:SOURce HOLD is set, the scan is advanced one channel each time TRIGger is executed.

```
TRIG:SOUR HOLD          ! Sets trigger source to HOLD.
SCAN (@10000:10003)     ! Defines channel list.
INIT                    ! Begin scan, close channel 00.
loop statement          ! Start count loop.
TRIG                    ! Advance scan to next channel.
increment loop          ! Increment loop count.
```

TRIGger:SOURce <*source*> specifies the trigger *source* to advance the channel list during scanning.

Parameters

Parameter Name	Parameter Type	Parameter Description	Default Value
BUS	discrete	*TRG or GET command	IMM
ECLTrg*n*	numeric	ECL Trigger bus line <0 or 1>	IMM
EXTernal	discrete	Trig In port	IMM
HOLD	discrete	Hold Triggering	IMM
IMMediate	discrete	Immediate Triggering	IMM
TTLTrg*n*	numeric	TTL Trigger bus line <0 - 7>	IMM

Comments

- **Enabling the Trigger Source:** The TRIGger:SOURce command only selects the trigger source. The INITiate[:IMMediate] command enables the trigger source.
- **Using the TRIG Command:** You can use TRIGger[:IMMediate] to advance the scan when TRIGger:SOURce BUS or TRIGger:SOURce HOLD is selected.
- **Using External Trigger Inputs:** With TRIGger:SOURce EXTernal selected, only one switchbox at a time can use the external trigger input at the HP El406A Trig In port. The trigger input is assigned to the first switchbox that requested the external trigger source (with a TRIGger:SOURce EXTernal command).
- **Assigning External Trigger:** A switchbox assigned with TRIGger:SOURce EXTernal remains assigned to that *source* until the switchbox trigger *source* is changed to BUS, ECLT, HOLD, IMMediate, or TTLT. When the *source* is changed, the external trigger *source* is available to the next switchbox which requests it (with a TRIGger:SOURce EXTernal command). If a switchbox requests an external trigger input already assigned to another switchbox, an error is generated.
- **Using Bus Triggers:** To trigger the switchbox with TRIGger:SOURce BUS selected, use the IEEE 488.2 Common command *TRG or the HP-IB Group Execute Trigger (GET) command.
- **Trig Out Port Shared by Switchboxes:** See the OUTPut command.
- **Related Commands:** ABORt, [ROUTe:]SCAN, OUTPut
- ***RST Condition:** TRIGger:SOURce IMMediate

Example Scanning Using External Triggers.

This example uses external triggering (TRIG:SOUR EXT) to scan channels 00 through 03 of a single-module switchbox. The trigger *source* to advance the scan is the input to the Trig In port on the HP E1406A Command Module. When INIT is executed, the scan is started and channel 00 is closed. Then each trigger received at the Trig In port advances the scan to the next channel.

```
TRIG:SOUR EXT                  ! Select external triggering.
SCAN (@10000:10003)            ! Scan channels 00 through 03.
INIT                           ! Begin scan, close channel 00.
trigger externally             ! Advance scan to next channel.
```

Example Scanning Using Bus Triggers.

This example uses bus triggering (TRIG:SOUR BUS) to scan channels 00 through 03 of a single-module switchbox. The trigger source to advance the scan is the *TRG command (as set with TRIGger:SOURce BUS). When INIT is executed, the scan is started and channel 00 is closed. Then, each *TRG command advances the scan to the next channel.

```
TRIG:SOUR BUS                  ! Select interface (bus) triggering.
SCAN (@10000:10003)            ! Scan channels 00 through 03.
INIT                           ! Start scan, close channel 00.
loop statement                 ! Loop to scan all channels.
*TRG                           ! Advance scan using bus triggering.
increment loop                 ! Increment loop count.
```

:SOURce?

TRIGger:SOURce? returns the current trigger source for the switchbox. Command returns *BUS*, *ECLT*, *EXT*, *HOLD*, *IMM*, or *TTLT* for sources BUS, ECLTrg, EXTernal, HOLD, IMMediate, or TTLTrg, respectively.

Example Query Trigger Source.

This example sets external triggering and queries the trigger source. Since external triggering is set, TRIG:SOUR? returns *EXT*.

```
TRIG:SOUR EXT                  ! Set external trigger source.
TRIG:SOUR?                     ! Query trigger source.
```

The following table lists the IEEE 488.2 Common (*) commands that apply to the relay matrix switch modules. The operation of some of these commands is described in earlier in this manual. For more information on Common Commands, refer to the *HP E1406 Command Module User's Manual* or the *ANSI/IEEE Standard 488.2-1987*.

Command	Title	Description
*IDN?	Identification Query	Returns identification string of the switchbox.
*RST	Reset	Opens all channel relays, and sets matrix to known state.
*TST?	Self-Test Query (cc = card number with leading 0 deleted)	Returns *+0* if self test passes. Returns *+cc01* for firmware error. Returns *+cc02* for bus error. Returns *+cc03* for bad ID information. Returns *+cc10* if an interrupt was expected but not received. Returns *+cc11* if the busy bit was not held 9 to 17 msec.
*OPC	Operation Complete	Sets the request for OPC flag when all pending operations have completed. Also sets OPC bit in the Event Status Register.
*OPC?	Operation Complete Query	Returns a *1* to the output queue when all pending operations have completed. Used to ensure synchronization between multiple instruments.
*WAI	Wait to Continue	Halts execution of commands and queries until the No Operation Pending message is true.
*CLS	Clear Status Register	Clears all status registers, the request for OPC flag, and all queues (except output queue).
*ESE <*unmask*>	Event Status Enable	Used to set the bits in the Event Status Enable Register.
*ESE?	Event Status Enable Query	Queries the current contents in the Event Status Enable Register.
*ESR?	Event Status Register Query	Queries and clears current contents in the Event Status Register.
*SRE <*unmask*>	Service Request Enable	Used to set the Service Request Enable Register bits, and corresponding Serial Poll Status Register bits, to generate a service request.
*SRE?	Service Request Enable Query	Queries the current contents in the Service Request Enable Register.
*STB?	Status Byte Query	Queries the current contents in the Status Byte Register.
*TRG	Trigger	Triggers the switchbox to advance the scan when scan is enabled and trigger source is TRIGger:SOURce BUS.
*RCL <*n*>	Recall Saved State	Recalls previously stored matrix configuration. <*n*> (0 to 9) is the location in memory where the desired (previously stored) set-up is located.
*SAV <*n*>	Save Current State	Stores the current matrix configuration in memory. Stores current settings of the channel states. <*n*> (0 to 9) is the location in memory where the current set-up is to be stored.

The following table summarizes the SCPI commands for the relay matrix switch modules.

Command		Description
ABORt		Abort a scan in progress.
ARM	:COUNt <*number*> MIN | MAX :COUNT? [MIN | MAX]	Multiple scans per INIT command. Query number of scans.
INITiate	:CONTinuous ON | OFF | 1 | 0 :CONTinuous? [:IMMediate]	Enables/disables continuous scanning. Query continuous scan state. Starts a scanning cycle.
OUTPut	:ECLTrg*n*[:STATe] ON | OFF | 1 | 0 :ECLTrg*n*[:STATe]? [:EXTernal][:STATe] ON | OFF | 1 | 0 [:EXTernal][:STATe]? :TTLTrg*n*[:STATe] ON | OFF | 1 | 0 :TTLTrg*n*[:STATe]?	Enables/disables ECL Trigger bus line pulse. Query ECL Trigger bus line state. Enables/disables Trig Out pulse. Query port enable state. Enables/disables TTL Trigger bus line pulse. Query TTL Trigger bus line state.
[ROUTe:]	CLOSe <*channel_list*> CLOSe? <*channel_list*> OPEN <*channel_list*> OPEN? <*channel_list*> SCAN <*channel_list*>	Close channel(s). Query channel(s) closed. Open channel(s). Query channel(s) opened. Define channels for scanning.
STATus	:OPERation:CONDition? :OPERation:ENABle <*number*> :OPERation:ENABle? :OPERation[:EVENt]? :PRESet	Returns status of Operation Condition Register. Enables events in the Operation Event Register to be reported. Returns the mask value set by the :ENABle command. Returns the contents of the Operation Event Register. Sets Enable Register bits to 0.
SYSTem	:CDEScription? <*number*> :CPON <*number*> | *ALL* :CTYPe? <*number*> :ERRor?	Returns description of module in switchbox. Sets specified module to its power-on state. Returns the module type. Returns error number/message to error queue.
TRIGger	[:IMMediate] :SOURce BUS :SOURce ECLTrg*n* :SOURce EXTernal :SOURce HOLD :SOURce IMMediate :SOURce TTLTrg*n* :SOURce?	Causes a trigger to occur. Trigger source is *TRG. Trigger source is ECL Trigger bus line 0 or 1. Trigger source is Trig In port. Hold off triggering. Continuous (internal) triggering. Trigger source is TTL trigger bus line (0 - 7). Query scan trigger source.

Appendix A
HP E1468A/E1469A Specifications

Input Characteristics

Maximum Voltage Terminal to Terminal:
220 Vdc; 250 Vac_{rms}

Maximum Voltage Terminal to Chassis:
220 Vdc; 250 Vac_{rms}

Maximum Current per Channel (non-inductive):
1 Adc or ac_{rms} (Vmax <30 Vdc or V_{rms})
0.3 Adc or ac_{rms} (Vmax <220 Vdc or 250 V_{rms})

Maximum Power per Channel: 40VA

DC Performance

Thermal Offset per Channel: <7μV (differential H-L)

Closed Channel Resistance:
<1.5 Ω initially
<3.5 Ω at end of relay life

Insulation Resistance (between any two points):
$5x10^6$Ω (at 40°C, 95% RH)
$5x10^8$Ω (at 25°C, 40% RH)

AC Performance

Bandwidth (-3dB):
Z(load) = Z(source) = 50 Ω
2-Wire mode (4x16): >10 MHz
1-Wire mode (1x128): >3 MHz
**The -3 dB BW is typically >25 MHz*

Crosstalk Between Channels @ 10 kHz:
2-Wire mode (4x16): <-90 dB
1-Wire mode (1x128): <-60 dB

Open Channel Capacitance
(channel to channel, channel to common):
2-Wire mode (4x16): <30 pF
1-Wire mode (1x128): <380 pF

Closed Channel Capacitance (Hi-Lo, Lo-Chassis):
650/700pF

General

Module Size/ Device Type:
C-Size VXIbus, Register based, A16/D16, Interrupter (levels 1-7, jumper selectable)

Relay Life:
@ No Load: $5x10^6$ Operations
@ Full Load: 10^5 Operations

Note: *Relays are subject to normal wear-out based on the number of operations.*

Terminals:
Screw type, maximum wire size 16AWG

Power Requirements:

	+5 V	+24 V
Voltage:		
Peak Module Current (A):	0.10	0.13
Dynamic Module current (A):	0.10	0.02

Watts/slot: 5.0

Cooling/slot: 0.08 mm H_2O @ 0.42 liter/sec

Operating Temperature: 0° – 55°C

Operating Humidity: 65% RH, 0° – 40°C

Net Weight (kg): 1.6

Relay Life

Electromechanical relays are subject to normal wear-out. Relay life depends on several factors. The effects of loading and switching frequency are briefly discussed below:

Relay Load. In general, higher power switching reduces relay life. In addition, capacitive/inductive loads and high inrush currents (for example, turning on a lamp or starting a motor) reduces relay life. *Exceeding specified maximum inputs can cause catastrophic failure.*

Switching Frequency. Relay contacts heat up when switched. As the switching frequency increases, the contacts have less time to dissipate heat. The resulting increase in contact temperature also reduces relay life.

End-of-Life Detection

A preventive maintenance routine can prevent problems caused by unexpected relay failure. The end of the life of the relay can be determined by using one or more of the three methods described below. The best method (or combination of methods), as well as the failure criteria, depends on the application in which the relay is used.

Contact Resistance. As the relay begins to wear out, its contact resistance increases. When the resistance exceeds a predetermined value, the relay should be replaced.

Stability of Contact Resistance. The stability of the contact resistance decreases with age. Using this method, the contact resistance is measured several (5-10) times, and the variance of the measurements is determined. An increase in the variance indicates deteriorating performance.

Number of Operations. Relays can be replaced after a predetermined number of contact closures. However, this method requires knowledge of the applied load and life specifications for the applied load.

Replacement Strategy

The replacement strategy depends on the application. If some relays are used more often, or at a higher load, than the others, the relays can be individually replaced as needed. If all the relays see similar loads and switching frequencies, the entire circuit board can be replaced when the end of relay life approaches. The sensitivity of the application should be weighed against the cost of replacing relays with some useful life remaining.

Note Relays that wear out normally or fail due to misuse should not be considered defective and are not covered by the product's warranty.

Appendix B
Relay Matrix Registers

Register Definitions

The relay matrix switch module is a register-based device. See "Reading the Registers" or "Writing to the Registers" later in this appendix for more information.

Addressing the Registers

Register addresses for register-based devices are located in the upper 25% of VXI A16 address space. Every VXI device (up to 256 devices) is allocated a 32-word (64-byte) block of addresses.

Figure B-1 shows the register address location within A16 as it might be mapped by an embedded controller. Figure B-2 shows the location of A16 address space in the HP E1405/E1406 Command Module.

When you are reading or writing to a register, a hexadecimal or decimal register address is specified. This address consists of an A16 base address plus a register offset.

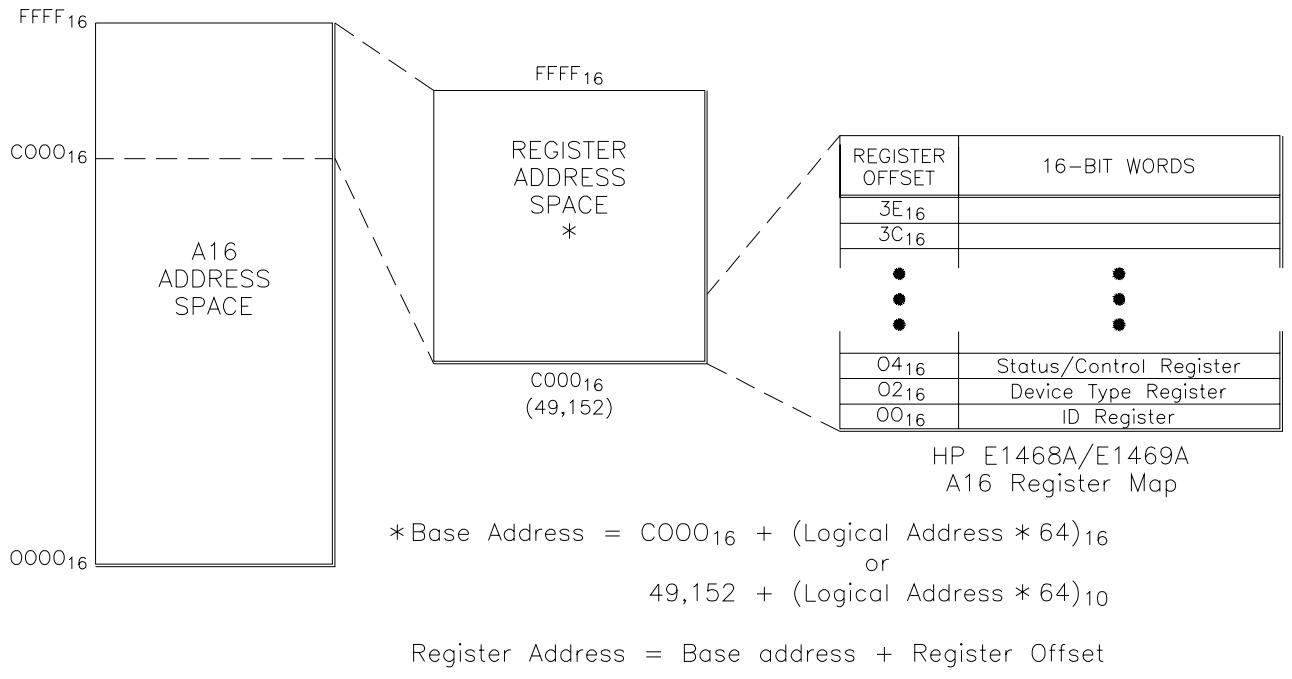


Figure B-1. Registers Within A16 Address Space

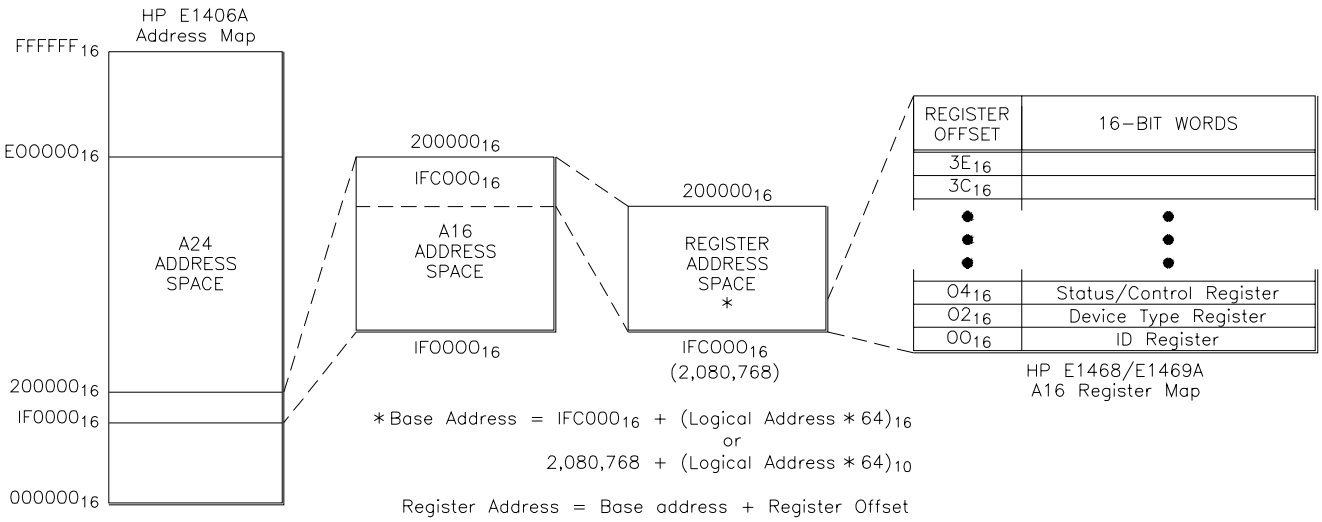


Figure B-2. Registers Within HP E1406A A16 Address Space

For example, the relay matrix switch module's Status/Control Register has an offset of 04_{16}. When you write to or read from this register, the offset is added to the base address to form the register address (using a logical address of 112):

register address = base address + register offset

a. $= 1FC000_{16} + (112 * 64)_{16} + 04_{16}$

b. $= 1FC000_{16} + 1C00_{16} + 04_{16} =$ **$1FDC04_{16}$**

or

a. $= 2,080,768 + (112 * 64) + 4$

b. $= 2,080,768 + 7168 + 8 =$ **2,087,940**

Reading the Registers

Figures 1-1 and 1-2 (see Chapter 1) show the channels grouped by banks. You can read the following relay matrix switch registers:

- Manufacturer ID Register (base + 00_{16})
- Device Type Register (base + 02_{16})
- Status/Control Register (base + 04_{16})
- Bank 0 Relay Control Register (base + 20_{16})
- Bank 1 Relay Control Register (base + 22_{16})
- Bank 2 Relay Control Register (base + 24_{16})
- Bank 3 Relay Control Register (base + 26_{16})
- Bank 4 Relay Control Register (base + 28_{16})
- Bank 5 Relay Control Register (base + $2A_{16}$)
- Bank 6 Relay Control Register (base + $2C_{16}$)
- Bank 7 Relay Control Register (base + $2E_{16}$)
- Channels 0990 - 0996 Relay Control Register (base + 30_{16})

ID and Device Type Registers

ID Register: Reading this register returns: $FFFF_{16}$. This shows Hewlett-Packard as the manufacturer and that the module is an A16 register- based device.

Device Type Register: Reading this register returns 0100_{16} if the device is an HP E1468A/E1469A Relay Matrix Switch Module.

Status/Control Register

The Status/Control Register informs the user about the module's status and configuration. Each relay requires about 12 msec execution time during which time the matrices are "busy". Bit 7 of this register is used to inform the user of a "busy" condition.

The interrupt generated after a channel has been closed can be disabled. Bit 6 of this register is used to inform the user of the interrupt status.

In addition, if a terminal module is connected to the switch module, the present configuration of the terminal module's status bit can be read. Bits 10, 11, 12, and 13 of this register are used to determine the configuration of the terminal module.

As an example, if the relay matrix switch module is not busy (bit 7), the interrupt is enabled (bit 6), then a read of the Status/Control Register (base + 04_{16}) returns DBBF.

Relay Control Registers

Reading these registers always returns $FFFF_{16}$.

Writing to the Registers

You can write the following relay matrix switch registers:

- Status/Control Register (base + 04_{16})
- Bank 0 Relay Control Register (base + 20_{16})
- Bank 1 Relay Control Register (base + 22_{16})
- Bank 2 Relay Control Register (base + 24_{16})
- Bank 3 Relay Control Register (base + 26_{16})
- Bank 4 Relay Control Register (base + 28_{16})
- Bank 5 Relay Control Register (base + $2A_{16}$)
- Bank 6 Relay Control Register (base + $2C_{16}$)
- Bank 7 Relay Control Register (base + $2E_{16}$)
- Channels 0990 - 0996 Relay Control Register (base + 30_{16})

Status/Control Register

Writes to the Status/Control Register (base + 04_{16}) enables you to disable/enable the interrupt generated when channels are closed.

Writing a 1 to bit 0 of the Status/Control Register (base + 04_{16}) does not change the state of the latching relays (individual channel relays). Writing a 1 to this bit has the same effect as removing power from the cardcage; because the relays are latching relays, they do not change state.

Note It is necessary to write a 0 to bit 0 after the reset has been performed before any other commands can be programmed and executed. SCPI commands take care of this automatically.

To disable the interrupt generated when channels are closed, write a 1 to bit 6 of the Status/Control Register (base + 04_{16}).

Note Typically, interrupts are only disabled to "peek-poke" a module. Refer to the operating manual of the command module before disabling the interrupt.

Relay Control Registers

Writes to the Relay Control Registers (base + 20_{16} to 30_{16}) enable you to switch desired channels. Figures 1-1 and 1-2 (see Chapter 1) show the schematics for the modules and the bank, row, and column information.

Any number of relays per bank can be closed at a time.

Manufacturer ID Register

base + 00_{16}	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write	Undefined															
Read*	Manufacturer ID															

*Returns $FFFF_{16}$ = Hewlett-Packard A16 only register based.

Device Type Register

base + 02_{16}	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write	Undefined															
Read	0100_{16}															

Status/Control Register

base + 04_{16}	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined									D	Undefined					R
Read**	Undefined		S4	S3	S2	S1	Undefined		B	D	Undefined					

*R = Latching relays stay in their current state.
*D = Disable interrupt by writing 1 in bit #6.
**B = Status "busy" is 0 in bit #7.
**D = Status "Interrupt disable" is 1 in bit #6.
**S4 -S1 = Status "Configuration Status bits" hardwired onto the terminal modules.

S4	S3	S2	S1		
0	1	1	0	=	E1469A 4x16 Matrix
0	1	0	1	=	E1468A 8x8 Matrix

Bank 0 Relay Control Register

base + 20_{16}	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined								CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
Read	Always Returns $FFFF_{16}$															

*Writes a 1 to close channel.

Bank 1 Relay Control Register

base + 22_{16}	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined								CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
Read	Always Returns $FFFF_{16}$															

*Writes a 1 to close channel.

Bank 2 Relay Control Register

base + 24_{16}	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined								CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
Read	Always Returns $FFFF_{16}$															

*Writes a 1 to close channel.

Bank 3 Relay Control Register

base + 26_{16}	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined								CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
Read	Always Returns $FFFF_{16}$															

*Writes a 1 to close channel.

Bank 4 Relay Control Register

base + 28_{16}	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined								CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
Read	Always Returns $FFFF_{16}$															

*Writes a 1 to close channel.

Bank 5 Relay Control Register

base + $2A_{16}$	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined								CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
Read	Always Returns $FFFF_{16}$															

*Writes a 1 to close channel.

Bank 6 Relay Control Register

base + $2C_{16}$	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined								CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
Read	Always Returns $FFFF_{16}$															

*Writes a 1 to close channel.

Bank 7 Relay Control Register

base + $2E_{16}$	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined								CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
Read	Always Returns $FFFF_{16}$															

*Writes a 1 to close channel.

Channels 0990 - 0996 Relay Control Register

base + 30_{16}	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Write*	Undefined									CH6	CH5	CH4	CH3	CH2	CH1	CH0
Read	Always Returns $FFFF_{16}$															

*Writes a 1 to close channel.

Appendix C
Relay Matrix Error Messages

The following table lists the error messages associated with the relay matrix switch modules programmed with SCPI commands. See the *HP E1406 Command Module User's Manual* for complete information on error messages.

Number	Title	Potential Causes
–211	Trigger Ignored	Trigger received when scan not enabled. Trigger received after scan complete. Trigger too fast.
–213	INIT Ignored	Attempting to execute an INIT command when a scan is already in progress.
–224	Illegal Parameter Value	Attempting to execute a command with a parameter not applicable to the command.
–350	Too Many Errors	The queue holds a maximum of 30 error numbers/messages for each switchbox. The queue has overflowed.
1500	External Trigger Source Already Allocated	Assigning an external trigger source to a switchbox when the trigger source has already been assigned to another switchbox.
2000	Invalid Card Number	Addressing an module (card) in a switchbox that is not part of the switchbox.
2001	Invalid Channel Number	Attempting to address a channel of a module in a switchbox that is not supported by the module (for example, channel 99 of Matrix module).
2006	Command Not Supported on this Card	Sending a command to a module (card) in a switchbox that is unsupported by the module.
2008	Scan List Not Initialized	Executing a Scan without the INIT command.
2009	Too Many Channels in Channel List	Attempting to address more channels than available in the switchbox.
2012	Invalid Channel Range	Invalid channel(s) specified in SCAN <*channel_list*> command. Attempting to begin scanning when no valid channel list is defined.
2600	Function Not Supported on this Card	Sending a command to a module (card) in a switchbox that is not supported by the module or switchbox.
2601	Channel List Required	Sending a command requiring a channel list without the channel list.

*

*CLS, 67
*ESE, 67
*ESE?, 57, 67
*ESR?, 67
*IDN?, 67
*OPC, 67
*OPC?, 67
*RCL, 35, 67
*RST, 32, 67
*SAV, 35, 67
*SRE, 67
*SRE?, 67
*STB?, 57, 67
*TRG, 65, 67
*TST?, 67
*WAI, 67

A

A16 Address Space, 71
Abbreviated SCPI Commands, 40
ABORt Subsystem, 43
Address
  logical address, 12
  matrix channel, 13
  registers, 71
  secondary HP-IB, 12
  setting logical address, 16
  specific channels, 12 - 13
ARM Subsystem
  ARM:COUNt, 32, 44
  ARM:COUNt?, 45
Attaching Terminal Module to Switch Module, 22

B

Bits
  Enable Register bit, 60
  Message Available bit, 57
  Operational Status bit, 57
  Questionable Data bit, 57
  Scan Complete bit, 57
  Service Request bit, 57
  Standard Event Summary bit, 57
  Status bit, 73
  Summary bit, 59
Boolean Command Parameters, 41

C

Cables
  50 ohm, 30
  daisy-chain cables, 27 - 30
  dual banana instrument, 30
  expansion cables, 27 - 30
  SMB instrument, 30
Card Number, 12
CAUTIONS, 15
CDEScription?, (SYST:CDES?), 61
Certification, 3
Changing
  interrupt priority, 18
  logical address switch setting, 16
  status register switch setting, 17
Channel
  address, 12 - 13
  expansion connectors, 27
  list, 13
  numbers, 13
  range, 13
Channels
  closing a contiguous range of, 33
  closing specific channels, 52
  opening a contiguous range of, 33
  opening specific channels, 54
  querying closure, 34, 53, 55
  recalling state of, 35
  saving state of, 35
  scanning, 55
  sequencing through each, 34
  switching, 33
CLOSe, ([ROUTe:]CLOSe), 12, 33, 52
CLOSe?, ([ROUTe:]CLOSe?), 34, 53
Closing
  contiguous range of channels, 33
  first channel in channel list, 47
  matrix module channels, 52
  specific channels, 52
Command Module
  ECL Trigger bus lines, 48
  Trig Out port, using, 49 - 50
Commands
  ABORt subsystem, 43
  ARM subsystem, 44 - 45
  IEEE 488.2 common, 67
  INITiate subsystem, 46 - 47
  linking other commands, 41

OUTPut subsystem, 48 - 51
[ROUTe:] subsystem, 52 - 55
SCPI, 12
STATus subsystem, 57, 59 - 60
SYSTem subsystem, 61 - 63
TRIGger subsystem, 64 - 66
types of, 39
Comment Sheet, reader, 7
Common (*) Commands
*CLS, 67
*ESE, 67
*ESE?, 57, 67
*ESR?, 67
*IDN?, 67
*OPC, 67
*OPC?, 67
*RCL, 35, 67
*RST, 32, 67
*SAV, 35, 67
*SRE, 67
*SRE?, 67
*STB?, 57, 67
*TRG, 65, 67
*TST?, 67
*WAI, 67
format, 39
linking with SCPI commands, 41
list of, 67
Condition Register, 59
CONDition?, (STAT:OPER:COND?), 59
Configuring
E1468A terminal module, 24
E1469A terminal module, 25
Conformity, declaration of, 5
Connectors
channel expansion connectors, 27
E1468A terminal module, 24
E1469A terminal module, 25
pin-out diagram, 23
Contact Resistance, relays, 70
Continuous Scanning, 46
CONTinuous, (INIT:CONT), 32, 46
CONTinuous?, (INIT:CONT?), 47
COUNt, (ARM:COUNt), 32, 44
COUNt?, (ARM:COUNt?), 45
CPON, (SYST:CPON), 62
Creating Larger Matrices, 27
CTYPe?, (SYST:CTYP?), 62

D

Daisy-Chain Cable, 27 - 30
Declaration of Conformity, 5
Definition, instrument, 12
Description
command parameters (SCPI), 41
commands, types of, 39
Common (*) commands, 39
expansion cables, 27
HP-IB, 14
instrument, 12
interrupt priority, 18
logical address switch, 16
relay matrix module, 9
relay matrix registers, 71
SCPI command format, 39
SCPI commands, 42
status register switch, 17
Detecting Error Conditions, 36, 79 - 80
Device ID Register, 73
Device Type Register, 73, 75
Disabling Continuous Scanning Cycles, 46
Discrete Command Parameters, 41
Documentation History, 4
Dual Banana Instrument Cables, 30

E

ECL Trigger, enabling and setting, 48
ECLTrg
[:STATe], (OUTP:ECLT[:STATe]), 48
[:STATe]?, (OUTP:ECLT[:STATe]?), 49
Electromechanical Relays, 70
Enable Register Bits, 60
ENABle, (STAT:OPER:ENAB), 59
ENABle?, (STAT:OPER:ENAB?), 59
Enabling Continuous Scanning Cycles, 46
Enabling Trig Out port, 50
ERRor?, (SYST:ERR?), 36, 63
Errors
detecting error conditions, 36
messages and numbers, 79 - 80
polling switchbox for, 36
Event Register, 59 - 60
[:EVENt]?, (STAT:OPER[:EVENt]?), 60
Example Program
closing channels, 33
getting started, 14
identify the module, 32
opening channels, 33
polling for error message, 36
reset the module, 32
save/recall matrix switch state, 35
sequencing through each channel, 34
switching a signal to multimeter, 37
synchronizing with measurement instruments, 37
[:EXTernal]
[:STATe], (OUTP[:EXT][:STATe]), 32
[:STATe], (OUTP[:EXT][:STATe]), 49
[:STATe]?, (OUTP[:EXT][:STATe]?), 50
External Trigger Inputs, 65
Extraction Levers, 19, 22

F

Fifty-ohm Coax Cables, 30
Format
  Common (*) commands, 39
  SCPI commands, 39

G

Generating an Interrupt, 18
Group Execute Trigger (GET), 65

H

HP-IB
  description, 14
  Group Execute Trigger (GET), 65
  secondary address, 12

I

Identification of Module, 32
[:IMMediate], (INIT[:IMM]), 47
[:IMMediate], (TRIG[:IMM]), 64
Implied SCPI Commands, 40
Initial Operation, 14
INITiate Subsystem
  INIT:CONT, 32, 46
  INIT:CONT?, 47
  INIT[:IMM], 47
Installing Switch Module in Mainframe, 19
Instrument Definition, 12
Interrupt
  generating, 18
  priority, changing, 18
  priority, jumper, 18
  priority, selecting, 18

J

Jumper, interrupt priority, 18
Jumpers, terminal module, 24 - 25

L

Linking Commands, 41
Logical Address
  lowest, 12
  setting, 16
Logical Address Switch, 16

M

Manufacturer ID Register, 75
Matrix Module
  16 x 16, 28
  4 x 48, 29
  8 x 24, 27
  attaching terminal module to, 22
  card numbers, 12
  channel address, 13
  channel addresses, 13
  channel numbers, 13
  closing a range of channels, 33
  connector pin-out, 23
  daisy-chain cable, 27, 29
  description of, 9
  E1468A simplified schematic, 10
  E1469A simplified schematic, 11
  error messages, 79 - 80
  error numbers, 79 - 80
  expansion cable, 28
  identification, 32
  initial operation, 14
  installing in a mainframe, 19
  instrument definition, 12
  opening a range of channels, 33
  query commands, 37
  recalling state of, 35
  register definitions, 71
  relays, 9
  reset/default conditions, 32
  saving state of, 35
  scanning channels, 55
  SCPI commands, 42, 68
  specifications, 69 - 70
  switching channels, 33
  synchronizing with measurement instruments, 37
  terminal module wiring, 20, 27
  wiring together, 9
  writing to registers, 74
Message Available Bit, 57
Multimeter
  switching a signal to, 37
Multiple-module
  card number, 12
  switchbox configuration, 12

N

Non-continuous Scanning, 46
Number of Relay Operations, 70
Numeric Command Parameters, 41

O

OPEN, ([ROUTe:]OPEN), 12, 33, 54
OPEN?, ([ROUTe:]OPEN?), 34, 55
Opening
  contiguous range of channels, 33
  specific channels, 54

OPERation
CONDition?, (STAT:OPER:COND), 59
ENABle, (STAT:OPER:ENAB), 59
ENABle?, (STAT:OPER:ENAB?), 59
[:EVENt]?, (STAT:OPER[:EVENt]?), 60
Operation Status Register, 57, 59 - 60
Operation, initial, 14
Operational Status Bit, 57
Optional Command Parameters, 41
OUTPut Subsystem
OUTP:ECLT[:STAT], 48
OUTP:ECLT[:STAT]?, 49
OUTP:TTLT[:STAT], 50
OUTP:TTLT[:STAT]?, 51
OUTP[:EXT][:STAT], 32, 49
OUTP[:EXT][:STAT]?, 50

P

Parameters
boolean, 41
discrete, 41
numeric, 41
optional, 41
types of (SCPI), 41
Pin-out, connector, 23
Power-on Conditions, 32
PRESet, (STAT:PRES?), 60
Programming using SCPI, 12

Q

Query
commands, 37
continuous scanning state, 47
ECLTrg bus line state, 49
number of scanning cycles, 45
open/closed channels, 34, 53, 55
Trig Out port, state of, 50
trigger source, 66
TTLTrg bus line state, 51
Questionable Data Bit, 57
Quick Reference SCPI Commands, 68

R

Reader Comment Sheet, 7
Reading Registers, 71, 73
Recalling State of Channels, 35
Registers
addressing, 71
base address, 71
Condition register, 59
definition of, 71
Device ID register, 73
Device Type register, 73, 75
Event register, 59 - 60
Manufacturer ID register, 75
Operation Status, 57, 59 - 60
reading registers, 71, 73
Relay Control register, 73 - 77
Standard Event, 57
Standard Event Status, 57
Status Byte, 57, 59
Status/Control register, 72 - 75
writing to registers, 71, 73 - 74
Relay Control Register, 73 - 74
bank 0, 75
bank 1, 75
bank 2, 76
bank 3, 76
bank 4, 76
bank 5, 76
bank 6, 76
bank 7, 77
channels 0990-0996, 77
Relays
addressing specific relays, 12
contact resistance, 70
end-of-life detection, 70
life expectancy, 69 - 70
number of operations, 70
relay matrix module, 9
replacement strategy, 70
stability of contact resistance, 70
switching frequency, 70
Replacement Strategy, relays, 70
Reset Conditions, 32
Resistance, relay contact, 70
Restarting a Scan, 43
[ROUTe:] subsystem
[ROUTe:]CLOSe, 12, 33, 52
[ROUTe:]CLOSe?, 34, 53
[ROUTe:]OPEN, 12, 33, 54
[ROUTe:]OPEN?, 34, 55
[ROUTe:]SCAN, 12, 55

S

Safety Warnings, 4
Saving State of Channels, 35
Scan
restarting, 43
stopping, 43
Scan Complete Bit, 57
Scan Cycles
enabling and disabling, 46
querying, 45
selecting and starting, 46 - 47
setting, 44
SCAN, ([ROUTe:]SCAN), 12, 55
Scanning Channels

defining channels to be scanned, 55
using bus triggering, 66
using external triggering, 66
SCPI Commands
abbreviated, 40
ABORt, 43
ARM:COUNt, 32, 44
ARM:COUNt?, 45
boolean parameters, 41
discrete parameters, 41
format, 39
implied, 40
INITiate:CONTinuous, 32, 46
INITiate:CONTinuous?, 47
INITiate[:IMMediate], 47
linking, 41
numeric parameters, 41
optional parameters, 41
OUTPut:ECLTrg[:STATe], 48
OUTPut:ECLTrg[:STATe]?, 49
OUTPut:TTLTrg[:STATe], 50
OUTPut:TTLTrg[:STATe]?, 51
OUTPut[:EXTernal][:STATe], 32, 49
OUTPut[:EXTernal][:STATe]?, 50
parameter types, 41
programming with, 12
quick reference, 68
reference, 42
[ROUTe:]CLOSe, 12, 33, 52
[ROUTe:]CLOSe?, 34, 53
[ROUTe:]OPEN, 12, 33, 54
[ROUTe:]OPEN?, 34, 55
[ROUTe:]SCAN, 12, 55
separator, 40
specifying, 12
STATus:OPERation:CONDition?, 59
STATus:OPERation:ENABle, 59
STATus:OPERation:ENABle?, 59
STATus:OPERation[:EVENt]?, 60
STATus:PRESet, 60
subsystem, example of, 39
SYSTem:CDEScription?, 61
SYSTem:CPON, 62
SYSTem:CTYPe?, 62
SYSTem:ERRor?, 36, 63
TRIGger:SOURce, 32, 65
TRIGger:SOURce?, 66
TRIGger[:IMMediate], 64
variable command syntax, 41
Secondary HP-IB Address, 12
Selecting
interrupt priority, 18
trigger source, 65
Separator, SCPI commands, 40
Service Request Bit, 57
Setting
logical address switch, 16
number of scanning cycles, 44
status register switch, 17
Shock Hazard, 15
Single-module
card number, 12
switchbox configuration, 12
switchbox, scanning channels, 66
SMB Instrument Cables, 30
SOURce, (TRIG:SOUR), 32, 65
SOURce?, (TRIG:SOUR?), 66
Specifications, Matrix Module, 69 - 70
Specifying
channel addresses, 12 - 13
channels to be scanned, 55
SCPI Commands, 12
Standard Event Register, 57
Standard Event Status Register, 57
Standard Event Summary Bit, 57
Starting
scan, 43
scanning cycles, 46 - 47
scanning process, 47
[:STATe]
(OUTP:ECLT[:STATe]), 48
(OUTP:TTLT[:STATe]), 50
(OUTP[:EXT][:STATe]), 32, 49
[:STATe]?
(OUTP:ECLT[:STATe]?), 49
(OUTP:TTLT[:STATe]?), 51
(OUTP[:EXT][:STATe]?), 50
States, recalling and saving, 35
Static Electricity, using caution with, 15
Status Bit, 73
Status Byte Register, 57, 59
Status Register Switch, 17
STATus Subsystem
STAT:OPER:COND?, 59
STAT:OPER:ENAB, 59
STAT:OPER:ENAB?, 59
STAT:OPER[:EVENt]?, 60
STAT:PRES, 60
Status/Control Register, 72 - 75
Stopping a Scan, 43
Subsystem, SCPI commands, 39
Subsystems (SCPI Commands)
ABORt, 43
ARM, 44 - 45
INITiate, 46 - 47
OUTPut, 48 - 51
[ROUTe:], 52 - 55
STATus, 57, 59 - 60
SYSTem, 61 - 63
TRIGger, 64 - 66
Summary Bit, 59

Switch Driver, 57
Switchbox
  card numbers, 12
  multiple-module, 12
  polling for errors, 36
  saving state of channels, 35
  single-module, 12
  single-module scanning channels, 66
  using Trig Out port, 49
Switches
  logical address switch, 16
  status register switch, 17
Switching Channels, 33
Switching Frequency, relay, 70
Synchronizing with measurement instruments, 37
SYSTem Subsystem
  SYST:CDES?, 61
  SYST:CPON, 62
  SYST:CTYP?, 62
  SYST:ERR?, 36, 63

T

Terminal Block
  See Terminal Module
Terminal Module
  attaching to relay matrix switch module, 22
  configuring E1468A, 24
  configuring E1469A, 25
  creating larger matrices with, 27
  expansion connectors, 27
  jumpers, 24 - 25
  shield wiring, 27
  wiring, 20, 27
Trigger
  inputs, external, 65
  source, 65
TRIGger Subsystem
  TRIG:SOUR, 32, 65
  TRIG:SOUR?, 66
  TRIG[:IMM], 64
TTL Trigger, enabling and setting, 50
TTLTrg
  [:STATe], (OUTP:TTLT[:STATe]), 50
  [:STATe]?, (OUTP:TTLT[:STATe]?), 51

V

Variable SCPI Command Syntax, 41
Voltage, maximum, 15
VXIbus Interrupt Lines, 18

W

WARNINGS, 4, 15
Warranty, 3
Wiring
  multiple modules together, 9
  terminal module, 20, 27
Writing to registers, 71, 73 - 74