HP 3235 Switch/Test Unit

HP 3235 Programming and System Information Manual

Manual Updates are available for the HP 3235 user manuals.
Order HP Part Number 03235-90091.

HEWLETT PACKARD

Manual Part Number: 03235-90001
Microfiche Part Number: 03235-99001

Printed: AUGUST 1986 Edition 1
Printed in U.S.A. U0288
Update 2. FEBRUARY 1988

Printing History

The Printing History shown below lists the printing dates of all Editions and Updates created for this manual. The Edition number changes as the manual undergoes subsequent revisions. Editions are numbered sequentially starting with Edition 1. Updates, which are issued between Editions, contain individual replacement pages which the customer uses to update the current Edition of the manual. Updates are numbered sequentially starting with Update 1. When a new Edition is created, all Updates associated with the previous Edition are merged into the manual. Each new Edition or Update also includes a revised copy of this printing history page.

Many product updates and revisions do not require manual changes and, conversely, manual corrections may be done without accompanying product changes. Therefore, do not expect a one-to-one correspondence between product updates and manual updates.

Edition 1 (Part Number 03235-90001) . . . . AUGUST 1986
Update 1 (Part Number 03235-90090) . . . . MARCH 1987
Update 2 (Part Number 03235-90091) . . . . FEBRUARY 1988

HEWLETT PACKARD

Herstellerbescheinigung

Hiermit wird bescheinigt, daß das Gerät/System HP 3235
in Übereinstimmung mit den Bestimmungen von Postverfügung 1046/84 funkentstört ist.

Der Deutschen Bundespost wurde das Inverkehrbringen dieses Gerätes/Systems angezeigt und die Berechtigung zur Überprüfung der Serie auf Einhaltung der Bestimmungen eingeräumt.

Zusatzinformation fur Meß- und Testgeräte

Werden Meß- und Testgeräte mit ungeschirmten Kabeln und/oder in offenen Meßaufbauten verwendet, so ist vom Betreiber sicherzustellen, daß die Funk-Entstörbestimmungen unter Betriebsbedingungen an seiner Grundstücksgrenze eingehalten werden.

Manufacturer's declaration

This is to certify that the equipment HP 3235
is in accordance with the Radio Interference Requirements of Directive FTZ 1046/84. The German Bundespost was notified that this equipment was put into circulation, the right to check the series for compliance with the requirements was granted.

Additional Information for Test- and Measurement Equipment

If Test- and Measurement Equipment is operated with unscreened cables and/or used for measurements on open set-ups, the user has to assure that under operating conditions the Radio Interference Limits are still met at the border of his premises.

HEWLETT PACKARD

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 Bureau of Standards, to the extent allowed by the Bureau's calibration facility, and to the calibration facilities of other International Standards Organization members.

WARRANTY

This Hewlett-Packard instrument product is warranted against defects in materials and workmanship for a period of one year from date of shipment. During the warranty period, Hewlett-Packard Company will, at its option, either repair or replace products which prove to be defective.

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

Duration and conditions of warranty for this instrument may be superceded when the instrument is integrated into (becomes a part of) other -hp- instrument products.

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

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

EXCLUSIVE REMEDIES

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

ASSISTANCE

Product maintenance agreements and other customer assistance agreements are available for Hewlett-Packard products.

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

F

HEWLETT PACKARD

SAFETY SUMMARY

The following general safety precautions must be observed during all phases of operation, service, and repair of this instrument. 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 instrument. Hewlett-Packard Company assumes no liability for the customer's failure to comply with these requirements. This is a Safety Class 1 instrument.

GROUND THE INSTRUMENT

To minimize shock hazard, the instrument chassis and cabinet must be connected to an electrical ground. The instrument is equipped with a three-conductor ac power cable. The power cable must either be plugged into an approved three-contact electrical outlet or used with a three-contact to two-contact adapter with the grounding wire (green) firmly connected to an electrical ground (safety ground) at the power outlet. The power jack and mating plug of the power cable meet International Electrotechnical Commission (IEC) safety standards.

DO NOT OPERATE IN AN EXPLOSIVE ATMOSPHERE

Do not operate the instrument in the presence of flammable gases or fumes. Operation of any electrical instrument in such an environment constitutes a definite safety hazard.

KEEP AWAY FROM LIVE CIRCUITS

Operating personnel must not remove instrument covers. Component replacement and internal adjustments must be made by qualified maintenance personnel. Do not replace components with power cable connected. Under certain conditions, dangerous voltages may exist even with the power cable removed. To avoid injuries, always disconnect power and discharge circuits before touching them.

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 INSTRUMENT

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

DO NOT OPERATE A DAMAGED INSTRUMENT

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

DANGEROUS PROCEDURE WARNINGS

Warnings, such as the example below, precede potentially dangerous procedures throughout this manual. Instructions contained in the warnings must be followed.

WARNING

Dangerous voltages, capable of causing death, are present in this instrument. Use extreme caution when handling, testing, and adjusting.

A

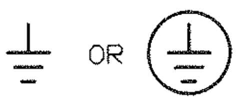
Operating and Safety Symbols

Symbols Used On Products And In Manuals

~ LINE — AC line voltage input receptacle.

Instruction manual symbol affixed to product. Cautions the user to refer to respective instruction manual procedures to avoid possible damage to the product.

Indicates dangerous voltage – terminals connected to interior voltage exceeding 1000 volts.



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

Clean ground (low-noise). Indicates terminal that must be connected to earth ground before operating equipment – for single common connections and protection against electrical shock in case of fault.

Frame or chassis ground. Indicates equipment chassis ground terminal – normally connects to equipment frame and all metal parts.

Affixed to product containing static sensitive devices – use anti-static handling procedures to prevent electrostatic discharge damage to components.

NOTE

NOTE

Calls attention to a procedure, practice, or condition that requires special attention by the reader.

CAUTION

CAUTION

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

WARNING

WARNING

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

WARNING and CAUTION Symbols

The HP 3235 Mainframe, the HP 3235E Extenders, and the plug-in modules contain subscripted international warning symbols (e.g., ⚠$_{1}$). These warning symbols refer the user to the operating manuals for further information. The table below shows the warning symbols used on the HP 3235 and its plug-in modules.

HP 3235 Switch/Test Unit Warning Symbols

Symbol	Meaning	Where Used
⚠$_{1}$	Hazardous voltages may be present when the Quick Interconnect Fixture is open.	The Quick Interconnect Fixture. See Chapter 2 in the Programming and System Information Manual for more information.
⚠$_{2}$	A shock hazard may exist. Turn off all power to the HP 3235 mainframe and extenders before installing or removing plug-in modules. Do not exceed 42V maximum.	HP 34504, 34505, 34506, 34508, 34510, 34511, 34521, and 34524 Plug-In Modules.
⚠$_{3}$	A shock hazard may exist. Turn off all power to the HP 3235 mainframe and extenders before installing or removing plug-in modules. Do not exceed 42V maximum.	HP 34501, 34502, 34503, 34507, 34509, 34511, 34520, and 34522 Plug-In Modules.
⚠$_{4}$	The sheet metal screws on the HP 34520 Multimeter are at GUARD potential.	HP 34520 Plug-In Module. See Chapter 2 in the Programming and System Information Manual for more information.
⚠$_{5}$	Do not carry the HP 3235 mainframe or extender frame by the Quick Interconnect Mechanism handles.	Quick Interconnect Mechanism handles.

Chapter 6
Scanning Applications

Chapter 7
Variables and Math

Chapter 8
Subroutines

Chapter 9
Trigger Buses and System Timing

Chapter 10
Interrupts

Chapter 11
HP 3235 Advanced Programming

Chapter 12
Signal Integrity

Chapter 13
Self Test

Chapter 14
CIIL Commands

List of Figures

List of Tables

Chapter 1

Introduction to the HP 3235

Introduction

Whether you work with large production test systems or smaller R&D test systems, the HP 3235 Switch/Test Unit (STU) simplifies the task of signal switching. With its various plug-in modules, the HP 3235 is a flexible and easy to use instrument. The switch modules route signals from external instruments to your device-under-test. Use the optional multimeter or an external instrument to measure test signals routed by the HP 3235's switch modules.

This chapter introduces the HP 3235 and shows how it functions in general test system applications. It also highlights some of the important features of the HP 3235 through a tour of the plug-in module cage, rear panel, and control panel. At the end of this chapter is a table of the HP 3235's available options and accessories.

Overview of the HP 3235

Your HP 3235 provides high integrity signal switching with some measurement and signal source capabilities in a versatile, rack mountable instrument. It serves as the signal switching interface between your test system instrumentation and the device-under-test.

Key features of the HP 3235 include:

- Each frame (mainframe and extenders) has a ten slot plug-in module cage. Install any plug-in module in any slot.
- Up to 7 extender frames (HP 3235E) may connect to each mainframe providing a total of 80 slots. (See the Installation Manual.)
- The HP 3235 is HP-IB* compatible making it easy to program from a variety of computers.
- Both switching and instrument plug-in modules are available. (See the Plug-in Module Manual.)
- Four backplane analog buses allow isolated signal routing between plug-in modules and between frames. (See Chapter 2 in this manual.)
- Two trigger buses pass digital trigger signals between plug-in modules and frames. (See Chapter 9 in this manual)
- The Quick Interconnect Fixture permits the test system operator to quickly change cabling connections to the HP 3235.

---

*HP-IB(Hewlett-Packard Interface Bus) is Hewlett-Packard's implementation of IEEE 488-1978 and ANSI MC1.1.

- The optional HP 34550A diagnostic control panel makes your test system easy to debug. (See Chapter 3 in this manual.)
- The HP 3235 has modular construction for easy servicing.
- The optional HP 345xxD diagnostic test fixtures help verify proper operation of your HP 3235 Plug-in Modules.

Using the HP 3235 in a Test System

Figure 1-1 shows the HP 3235 STU in a simplified test system. As shown, your test system probably consists of additional instruments such as programmable power supplies, frequency synthesizers, and signal analyzers. The HP 3235 switches most test signals between the system instrumentation and the device-under-test (DUT).

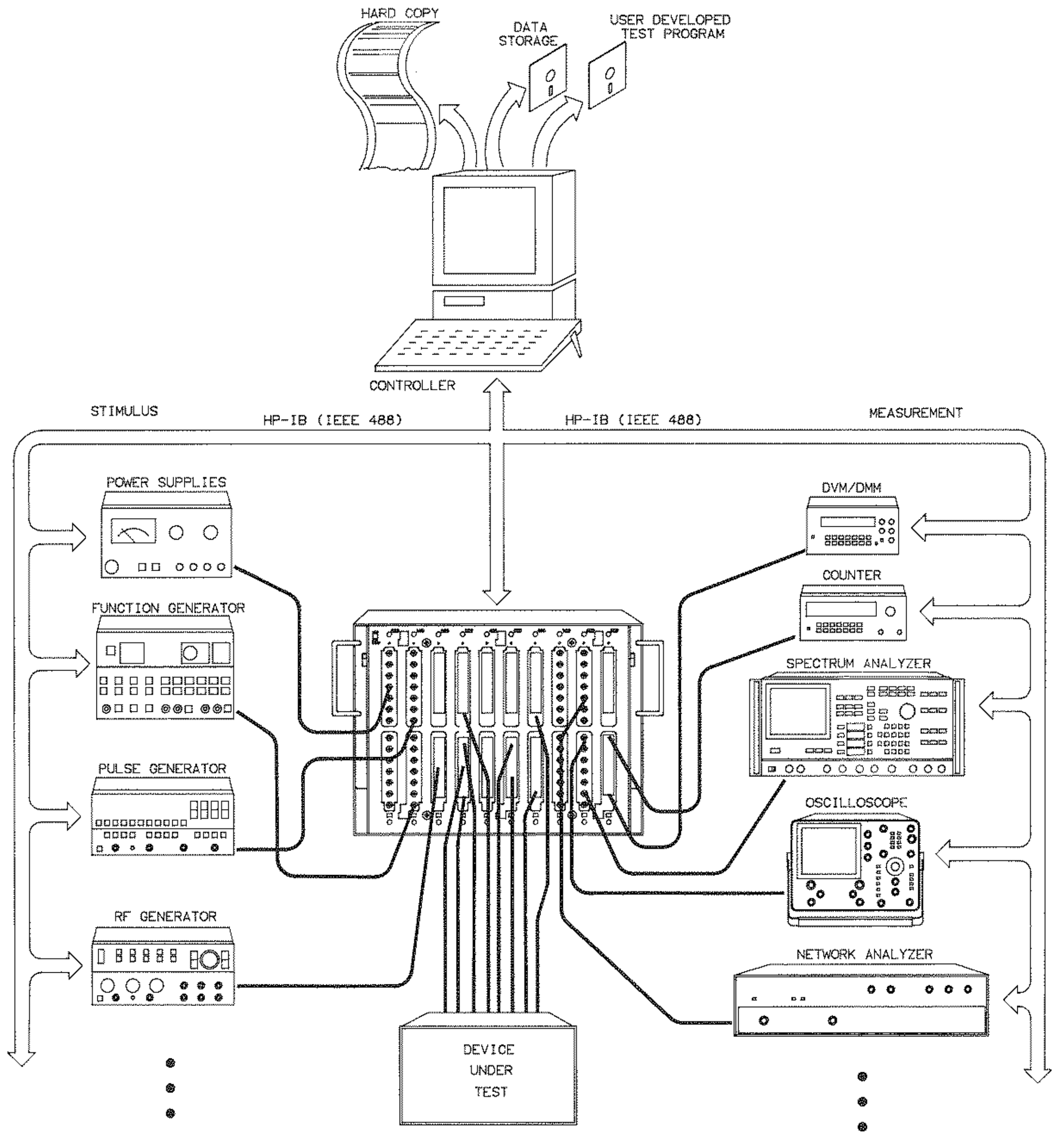


Figure 1-1. Simplified Test System

In addition to signal switching, many test systems also include: signal stimulus, measurement, communication, and data reduction activities.

Switching

The HP 3235 can switch a wide variety of test signals. You can create many different switch topologies by wiring the plug-in modules together. Refer to Figure 1-2. In addition, the four analog backplane buses (described in Chapter 2 of this manual) allow up to four isolated 2-wire connection paths between the plug-in modules. The switching specifications for the HP 3235 are in Appendix A of this manual.

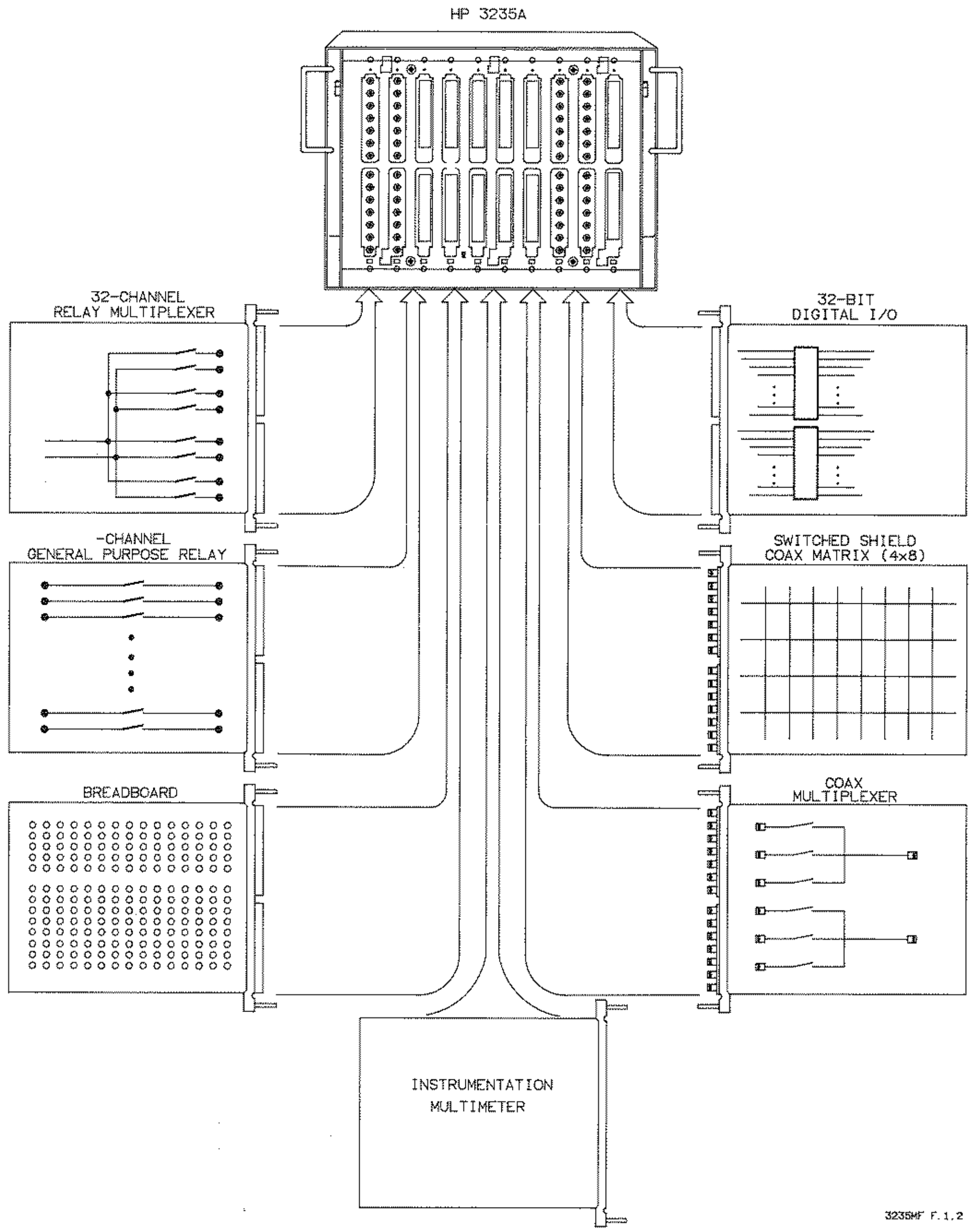


Figure 1-2. The Capabilities of the HP 3235

Stimulus

The HP 3235 has some capability to stimulate the DUT (with the HP 34522 Digital I/O Module). However, most stimulus signals will come from other sources switched through the HP 3235. Stimulus signals can include AC or DC power supplies, AC and DC test signals, precision pulse trains, and digital words.

Measurement

After applying a stimulus, you may need to measure the results. The HP 3235 has an optional 6 1/2 digit multimeter with a built-in frequency counter to handle many measurement tasks. Of course, your HP 3235 can switch the response signals to other external measurement devices such as power meters, distortion analyzers, and digital oscilloscopes.

Some applications require external system status reports (switch open/closed, object present/absent, on/off, etc.). The HP 3235 can monitor these status reports and, if necessary, interrupt the system controller. The system controller, in turn, can alert the operator or command the HP 3235 to perform some appropriate task.

Data Communication

The HP 3235 communicates with the controller via the HP-IB. Communication involves the system controller sending commands to the HP 3235 for execution, and the HP 3235 returning data to the controller. Appropriate programming allows the HP 3235 to interrupt the controller.

With the HP-IB interface, the system controller can communicate with up to 14 different devices or instruments. For example, you can connect a system printer and disc drive plus other instruments (such as a digital oscilloscope and frequency source) on the same interface bus. Refer to the HP 3235 Installation Manual for restrictions on using the HP-IB interface.

Data Reduction and Storage

Although the HP 3235 has the ability to manipulate data, your system controller will do most of the data reduction. Your HP 3235 could compare measured values with specified limits and alert the operator or system controller if the tested data falls outside the specified limits. For more information, refer to Chapter 7 in this manual. A controller also provides data display and hard copy print-out, data storage, and interaction with the system operator.

HP 3235 Description

Now that we have seen where the HP 3235 fits in a test system, let's take a closer look at the instrument itself. In later chapters we'll look at each feature in more detail.

Plug-in Module Cage

Figure 1-3 shows the HP 3235's plug-in module cage features.

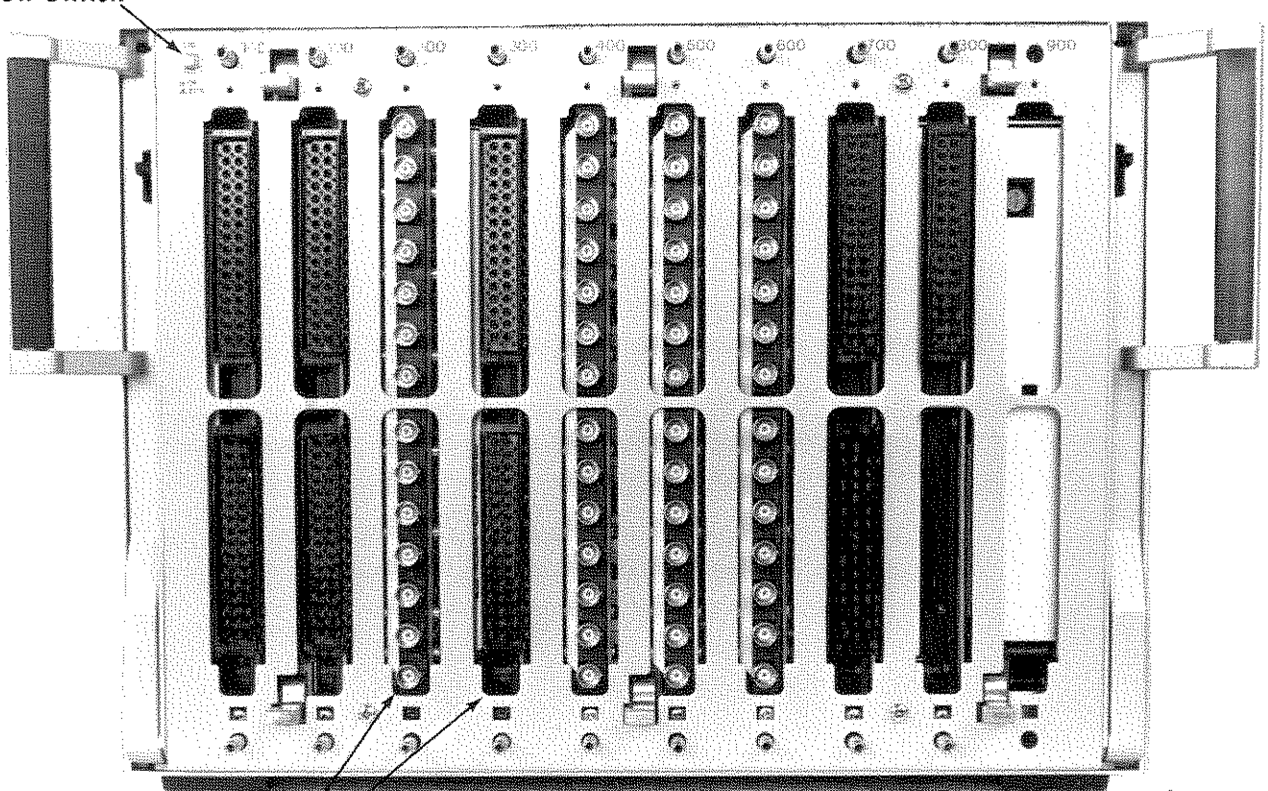


Figure 1-3. HP 3235 Plug-in Module Cage

Rear Panel

Figure 1-4 shows the HP 3235's rear panel features.

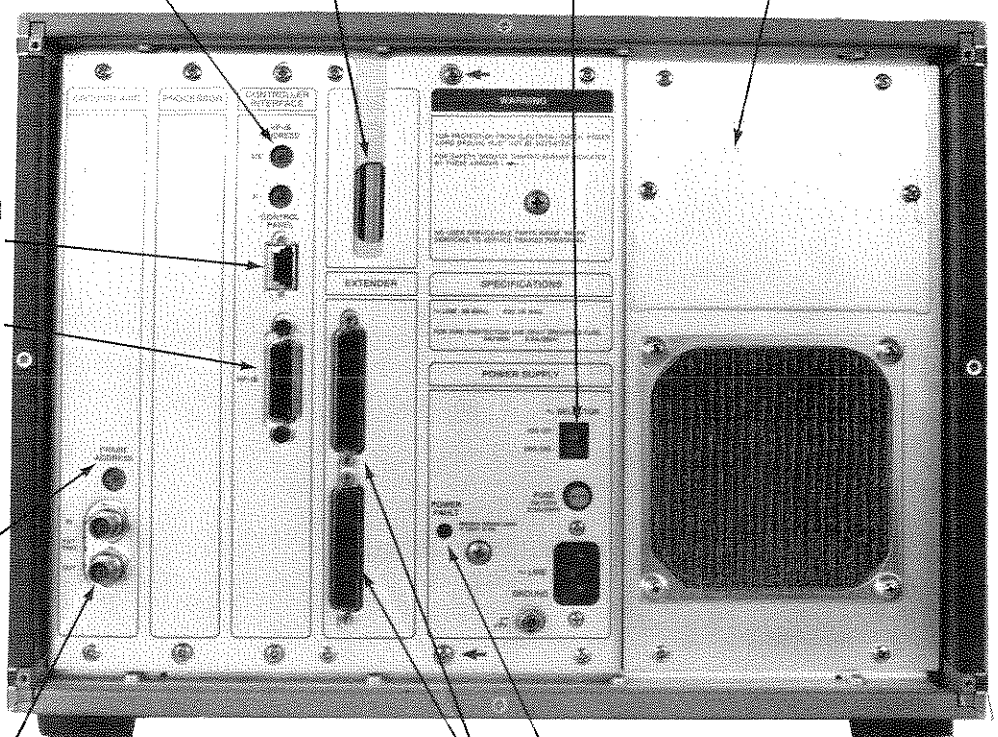


Figure 1-4. HP 3235 Rear Panel Features

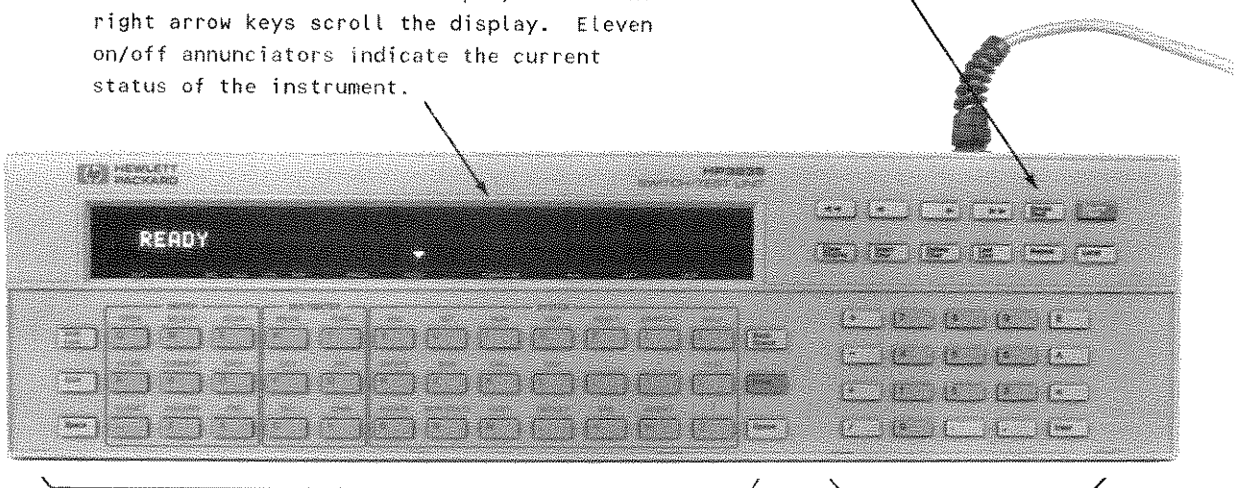
Control Panel

The optional HP 34550A Control Panel (keyboard with display) provides a sophisticated tool for debugging your test system. You can execute any HP 3235 command from the keyboard with the results shown in the display. An operator may also use the control panel to respond to queries from the system controller. Refer to Chapter 3 in this manual for detailed information on using the control panel. Figure 1-5 shows the control panel and summarizes its features.

Display

The control panel display is a 40 character window in a 256 character display. Left and right arrow keys scroll the display. Eleven on/off annunciators indicate the current status of the instrument.

Immediate Execute and Edit Keys



Alphabetic, Symbol, and Command Header Keys

The keyboard can execute any of the HP 3235 commands. Twenty-six command headers have been predefined in functional groups.

Numeric Keypad and User Defined Keys

The 10 number keys have a **USER** function for defining long command strings.

Figure 1-5. Control Panel

The keyboard has a QWERTY style layout with a separate numeric keypad. Dedicated keys provide editing and immediate execute functions. The alphabet keys double as predefined command headers. You can even customize the keyboard by redefining most of the keys.

Options and Accessories

Table 1-1 lists the available options and accessories for your HP 3235.

Table 1-1. Options and Accessories

Description of Option/Accessory	Use This Number When Ordering With HP 3235	Use This Number When Ordering Separately
32-Channel Armature Relay Module	HP 34501A (solder-eye multiplexer) HP 34501B (screw term. multiplexer) HP 34501M (screw term. 4x8 matrix) HP 34501T (screw term. thermocouple)	HP 34501A (solder-eye multiplexer) HP 34501B (screw term. multiplexer) HP 34501M (screw term. 4x8 matrix) HP 34501T (screw term. thermocouple)
32-Channel Reed Relay Module	HP 34502A (solder-eye multiplexer) HP 34502B (screw term. multiplexer) HP 34502M (screw term. 4x8 matrix) HP 34502T (screw term. thermocouple)	HP 34502A (solder-eye multiplexer) HP 34502B (screw term. multiplexer) HP 34502M (screw term. 4x8 matrix) HP 34502T (screw term. thermocouple)
General Purpose Relay Module	HP 34503A (solder-eye) HP 34503B (screw term.)	HP 34503A (solder-eye) HP 34503B (screw term.)
Switched-Shield Coaxial Multiplexer Module	HP 34504A (BNC cable) HP 34504B (BNC bulkhead) HP 34504C (no term. blk)	HP 34504A (BNC cable) HP 34504B (BNC bulkhead) HP 34504C (no term. blk)
50 Ohm RF Multiplexer Module	HP 34505A (BNC cable) HP 34505B (BNC bulkhead) HP 34505C (no term. blk)	HP 34505A (BNC cable) HP 34505B (BNC bulkhead) HP 34505C (no term. blk)
Switched-Shield Coaxial Matrix Module	HP 34506A (BNC cable) HP 34506B (BNC bulkhead) HP 34506C (no term. blk)	HP 34506A (BNC cable) HP 34506B (BNC bulkhead) HP 34506C (no term. blk)
32-Channel Mercury-Wetted Relay Module	HP 34507A (solder-eye multiplexer) HP 34507B (screw term. multiplexer) HP 34507M (screw term. 4x8 matrix) HP 34507T (screw term. thermocouple)	HP 34507A (solder-eye multiplexer) HP 34507B (screw term. multiplexer) HP 34507M (screw term. 4x8 matrix) HP 34507T (screw term. thermocouple)

Table 1-1. Options and Accessories (cont'd)

Description of Option/Accessory	Use This Number When Ordering With HP 3235	Use This Number When Ordering Separately
75 Ohm RF Multiplexer Module	HP 34508A (BNC cable) HP 34508B (BNC bulkhead) HP 34508C (no term. blk)	HP 34508A (BNC cable) HP 34508B (BNC bulkhead) HP 34508C (no term. blk)
32-Channel Relay Driver Module	HP 34509A (connector and ribbon cable) HP 34509B (screw term.) HP 34509C (ribbon cable, no term. blk)	HP 34509A (connector and ribbon cable) HP 34509B (screw term.) HP 34509C (ribbon cable, no term. blk)
8-Channel Power Actuator Module	HP 34510B (screw term.)	HP 34510B (screw term.)
64-Channel Armature Relay Module	HP 34511B (screw term.) HP 34511M (matrix screw term.)	HP 34511B (screw term.) HP 34511M (matrix screw term.)
6-1/2 Digit Multimeter Module	HP 34520A (solder-eye) HP 34520B (screw term.)	HP 34520A (solder-eye) HP 34520B (screw term.)
AC/DC Source Module	HP 34521A (BNC cable) HP 34521B (BNC bulkhead)	HP 34521A (BNC cable) HP 34521B (BNC bulkhead)
32-Bit Digital I/O Module	HP 34522A (solder-eye) HP 34522B (screw term.)	HP 34522A (solder-eye) HP 34522B (screw term.)
Breadboard Module	HP 34523A (solder-eye) HP 34523B (screw term.)	HP 34523A (solder-eye) HP 34523B (screw term.)
4-Channel D/A Converter Module	HP 34524A (solder-eye) HP 34524B (screw term.)	HP 34524A (solder-eye) HP 34524B (screw term.)
Microwave Switch (for use with HP 34509 and HP 34530T)	HP 34530A (18 GHz) HP 34530B (26.5 GHz)	HP 34530A (18 GHz) HP 34530B (26.5 GHz)
1x6 Microwave Switch (for use with HP 34509 and HP 34530R/34530T)	HP 34531A (w/out 50 ohm termination) HP 34531B (with 50 ohm termination)	HP 34531A (w/out 50 ohm termination) HP 34531B (with 50 ohm termination)

Table 1-1. Options and Accessories (cont'd)

Description of Option/Accessory	Use This Number When Ordering With HP 3235	Use This Number When Ordering Separately
Control Panel (detachable)	HP 34550A	HP 34550A
System Expansion Card	Option 560 to HP 3235A	HP 34560-66201
Extra Length Extender Cables (3 meters)	Option 570 to HP 3235E	HP 34570A
HP-IB Controller	Option 580 to HP 3235A	HP 03235-80580
Quick Interconnect Mechanism and Fixture	Option 590 to Either HP 3235A or 3235E	Not Applicable
Additional Quick Interconnect Fixture	HP 34591A	HP 34591A
Wire Distribution Frame	HP 34592A	HP 34592A
Quick Interconnect Fixture Cover	HP 03235-04111	HP 03235-04111
BNC Feedthrough Panel	HP 34593A (BNC cable) HP 34593B (BNC bulkhead) HP 34593C (no term. blk)	HP 34593A (BNC cable) HP 34593B (BNC bulkhead) HP 34593C (no term. blk)
Low Frequency Feedthrough Panel	HP 34594A (solder-eye) HP 34594B (screw term.)	HP 34594A (solder-eye) HP 34594B (screw term.)
Microwave Switch Rack Panel Kit for HP 34530A/B and HP 3331XX switches	HP 34530R	HP 34530R
Microwave Switch Terminal Block Kit for HP 34530A/B	HP 34530T	HP 34530T
1x6 Microwave Switch Rack Panel Kit for HP 34531A/B	HP 34531R	HP 34531R
Microwave Attenuator Rack Panel Kit for HP 33320 attenuators	HP 34532R	HP 34532R

Table 1-1. Options and Accessories (cont'd)

Description of Option/Accessory	Use This Number When Ordering With HP 3235	Use This Number When Ordering Separately
Rack Mounting Kit for HP 34550A Control Panel (Includes Vertical Mount Kit and Angled Mount Kit)	HP 34551A	HP 34551A
Rack Mounting Kit for HP 3235A or HP 3235E (Includes Forward and Reverse Rack Mount Kits)	Option 908 to Either HP 3235A or 3235E	HP 03235-80908
Cabling Kit to Increase Number of Columns With Multiple HP 34506 Coaxial Matrix Modules	HP 34581A	HP 34581A
Cabling Kit to Increase Number of Rows With Multiple HP 34506 Coaxial Matrix Modules	HP 34582A	HP 34582A
Set of Four 50 Ohm Termination Resistors for HP 34505 RF Multiplexer	HP 34585A	HP 34585A
Set of Four 75 Ohm Termination Resistors for HP 34508 RF Multiplexer	HP 34586A	HP 34586A
Extra Set of HP 3235 User Documentation	HP 03235-90000	HP 03235-90000
HP-IB Cables	HP 10833A (1 meter) HP 10833B (2 meters) HP 10833C (4 meters) HP 10833D (0.5 meter)	HP 10833A (1 meter) HP 10833B (2 meters) HP 10833C (4 meters) HP 10833D (0.5 meter)
23 cm (9 in.) BNC-BNC Cable	HP 10502A	HP 10502A
122 cm (48 in.) BNC-BNC Cable	HP 10503A	HP 10503A
BNC Tee (m-f-f)	HP 1250-0781	HP 1250-0781

Additional Terminal Blocks

Table 1-1. Options and Accessories (cont'd)

Description of Terminal Block	Use This Number When Ordering
Solder-Eye Terminal Block for HP 34501, 34502, and 34507 32-Channel Relay Modules	HP 34501AT
Screw Terminal Block for HP 34501, 34502, and 34507 32-Channel Relay Modules	HP 34501BT
4x8 Matrix Screw Terminal Block for HP 34501, 34502, and 34507 32-Channel Relay Modules	HP 34501MT
Thermocouple Compensated Screw Terminal Block for HP 34501, 34502, and 34507 32-Channel Relay Modules	HP 34501TT
Solder-Eye Terminal Block for HP 34503 General Purpose Relay Module	HP 34503AT
Screw Terminal Block for HP 34503 General Purpose Relay Module	HP 34503BT
BNC Cable Terminal Block for HP 34504 Switched-Shield Coaxial Multiplexer Module	HP 34504AT
BNC Bulkhead Terminal Block for HP 34504 Switched-Shield Coaxial Multiplexer Module	HP 34504BT
BNC Cable Terminal Block for HP 34505 RF Multiplexer Module	HP 34505AT
BNC Bulkhead Terminal Block for HP 34505 RF Multiplexer Module	HP 34505BT
BNC Cable Terminal Block for HP 34506 Switched-Shield Coaxial Matrix Module	HP 34506AT
BNC Bulkhead Terminal Block for HP 34506 Switched-Shield Coaxial Matrix Module	HP 34506BT

Additional Terminal Blocks (cont'd)

Table 1-1. Options and Accessories (cont'd)

Description of Terminal Block	Use This Number When Ordering
BNC Cable Terminal Block for HP 34508 RF Multiplexer Module	HP 34508AT
BNC Bulkhead Terminal Block for HP 34508 RF Multiplexer Module	HP 34508BT
Ribbon Cable and Connector for HP 34509 Relay Driver Module	HP 34509AT
Screw Terminal Block for HP 34509 Relay Driver Module	HP 34509BT
Screw Terminal Block for HP 34510 Power Actuator Module	HP 34510BT
Screw Terminal Block for HP 34511 64-Channel Relay Module	HP 34511BT
Matrix Screw Terminal Block for HP 34511 64-Channel Relay Module	HP 34511MT
Solder-Eye Terminal Block for HP 34520 Multimeter Module	HP 34520AT
Screw Terminal Block for HP 34520 Multimeter Module	HP 34520BT
BNC Cable Terminal Block for HP 34521 Source Module	HP 34521AT
BNC Bulkhead Terminal Block for HP 34521 Source Module	HP 34521BT
Solder-Eye Terminal Block for HP 34522 Digital I/O Module	HP 34522AT
Screw Terminal Block for HP 34522 Digital I/O Module	HP 34522BT

Additional Terminal Blocks (cont'd)

Table 1-1. Options and Accessories (cont'd)

Description of Terminal Block	Use This Number When Ordering
Solder-Eye Terminal Block for HP 34524 D/A Converter Module	HP 34524AT
Screw Terminal Block for HP 34524 D/A Converter Module	HP 34524BT
Solder-Eye Terminal Block for HP 34523 Breadboard Module or HP 34594 Low Frequency Feedthrough Panel	HP 34523AT
Screw Terminal Block for HP 34523 Breadboard Module or HP 34594 Low Frequency Feedthrough Panel	HP 34523BT
BNC Cable Terminal Block for HP 34593 BNC Feedthrough Panel	HP 34593AT
BNC Bulkhead Terminal Block HP 34593 BNC Feedthrough Panel	HP 34593BT

Diagnostic Test Fixtures

Table 1-1. Options and Accessories (cont'd)

Description of Self-Test Fixture	Use This Number When Ordering
Fixture for HP 34501, 34502, and 34507 32-Channel Relay Modules	HP 34501D
Fixture for HP 34503 General Purpose Relay Module	HP 34503D
Fixture for HP 34504 Switched-Shield Coaxial Multiplexer Module	HP 34504D
Fixture for HP 34505 and 34508 RF Multiplexer Modules	HP 34505D
Fixture for HP 34506 Switched-Shield Coaxial Matrix Module	HP 34506D
Fixture for HP 34509 Relay Driver Module	HP 34509D
Fixture for HP 34510 Power Actuator Module	HP 34510D
Fixture for HP 34511 64-Channel Relay Module	HP 34511D
Fixture for HP 34520 DMM Module. Includes HP 34562-61601 and HP 34562-61602 Test Cables (see below)	HP 34520D
Fixture for HP 34521 Source Module	HP 34521D
Fixture for HP 34522 Digital I/O Module	HP 34522D
Fixture for HP 34524 D/A Converter Module	HP 34524D
Test Fixture Chaining Cable	HP 34562-61601
Test Module Extender Cable	HP 34562-61602

Chapter 2
HP 3235 Hardware Description

Introduction

This chapter introduces you to some of the hardware of the HP 3235. It starts by describing the power-on sequence and then discusses the four analog backplane buses and the two internal trigger buses. It also provides system information on the quick interconnect fixture, floating measurements and guarding, and the power fault condition.

Later chapters provide more detailed information on using the analog and trigger buses. Chapter 6, for example, describes scanning using the four analog buses and Chapter 9 gives more information and examples for the two trigger buses. Refer to the HP 3235 Installation Manual for wiring and configuration information.

The Power-On Sequence

The power-on sequence takes about 10 seconds to complete. It consists of running the power-on self-test, polling the plug-in modules, and going to an idle "READY" state.

NOTE

The HP 3235 will not start its power-on self test until every extender connected to the mainframe is turned on. If any extender loses power, the HP 3235 generates a power failure error and does not execute any additional commands until power is restored to the entire system.

Power-On Self Test

The power-on self test verifies that the HP 3235 is operating but does not ensure proper operation (i.e., accuracy of multimeter readings, etc.). The power-on self-test performs a test on the program ROMs and the read/write RAM memory. Also included are timer chip and trigger bus tests. For more information, refer to Chapter 13 in this manual.

NOTE

The error register stores the first four errors that occur during or after the power-on self-test. Errors are also noted on the optional control panel display. The section in Chapter 4 entitled "Error Handling" explains how to read the error register.

Plug-In Module Polling

Next, the HP 3235 polls the plug-in modules installed in the mainframe and extenders. The control panel (if installed) displays the number of plug-in modules found.

READY

Last, the HP 3235 goes to its *READY* state. This means that the HP 3235 has completed its power-on sequence. The mainframe processor is ready to receive and execute commands. The optional control panel beeps and shows the *READY* message.

Power-On State

When your HP 3235 completes its power-on sequence, all plug-in modules are set to their power-on state. In most cases, this means that all relays are open. You can program the HP 34501 Armature Relay Module to open or close specified relays when a power failure occurs. Similarly, specified relays on the HP 34504 Coaxial Multiplexer, the HP 34506 Coaxial Matrix, and HP 34511 64-Channel Armature Module can be programmed to open when a power failure occurs. The specified relays remain in the programmed state (open or closed) when power is restored. Refer to the **PFCLOSE**, **PFOPEN**, and **PFSAME** commands in Chapter 5 for more information.

Table 2-1 shows the power-on state of the HP 3235 plug-in modules.

Table 2-1. HP 3235 Plug-In Module Power-On States

Plug-In Module	Power-On State
HP 34501 Armature Relay Module	All Relays Open.*
HP 34502 Reed Relay Module	All Relays Open.
HP 34503 General Purpose Relay Module	All Relays Open.
HP 34504 Coaxial Multiplexer Module	All Relays Open.*
HP 34505 50 Ohm RF Multiplexer Module	All Relays Open.
HP 34506 Coaxial Matrix Module	All Relays Open.*
HP 34507 Mercury-Wetted Relay Module	All Relays Open.
HP 34508 75 Ohm RF Multiplexer Module	All Relays Open.
HP 34509 Relay Driver Module	All Relays Open, Non-Latching Mode Selected.
HP 34510 Power Actuator Module	All Relays in Normally Closed Position.
HP 34511 64-Channel Relay Module	All Relays Open.*
HP 34520 Multimeter Module	DC Volts, Automatic Trigger, Autorange, 6-1/2 digit Measurements.
HP 34521 AC/DC Source Module	DC Volts, 0 VDC Output, High-Resolution Mode, 0 Ohm Output Impedance, Autorange.
HP 34522 Digital I/O Module	Four 8-Bit Input Ports.
HP 34523 Breadboard Module	User-Defined.
HP 34524 D/A Converter Module	Voltage Mode, 0 VDC Output
Extender Bus Relays (discussed later in this chapter)	All Extender Bus Relays Closed.

*These modules have programmable power-failure states. The relays do not change state when power is restored.

If Power-on SRQ has been enabled (**PONSRQ ON** Command, see Chapter 10), then when power is cycled the HP 3235 asserts the HP-IB SRQ and attempts to interrupt the system controller. Your system controller must be programmed to respond to the SRQ interrupt.

NOTE

In most applications, the HP 3235 mainframe power-on state provides adequate performance for most tasks. Chapter 11 in this manual describes several advanced programming concepts to give you more flexibility in programming and using your HP 3235.

Backplane Buses

Each frame (mainframe and extender) backplane has four analog-signal line pairs (high and low) to carry signals between the plug-in modules. These four line pairs are the Analog Buses (AB0 through AB3). In addition, two trigger bus lines can carry synchronizing, pulse, or trigger signals. Refer to Figure 2-1. You have access to all of these backplane buses through different plug-in modules and can control what occurs on each bus through various HP 3235 commands. Appendix A contains the analog and trigger bus specifications.

Analog Buses

The analog bus system has four separate (isolated) 2-wire buses. These buses carry analog signals between the plug-in modules, to the extenders, or to external instruments. For example, the HP 3235 can switch a signal from one plug-in module through an analog bus to another module. This second module may be the multimeter module or another switch module; it may be in the same or another frame. The analog buses are rated at 250 volts peak at 1MHz.

Use the **OPEN**, **CLOSE**, **CONNECT**, and **DISCONN** commands to route signals from the plug-in modules through the analog buses. These commands specify relays on the various switch modules allowing you to route signals in the most efficient manner for your system. Refer to Chapter 5 in this manual for more information.

WARNING

In case of component failure or programming error, a voltage input to a module may be present on any or all terminals of that module. Also, a voltage input to any module may be present on the terminals of any modules interconnected by an analog bus, analog bus extender cables, or user wiring.

If the analog cable is NOT connected between the mainframe and the extenders, each frame is considered a separate circuit.

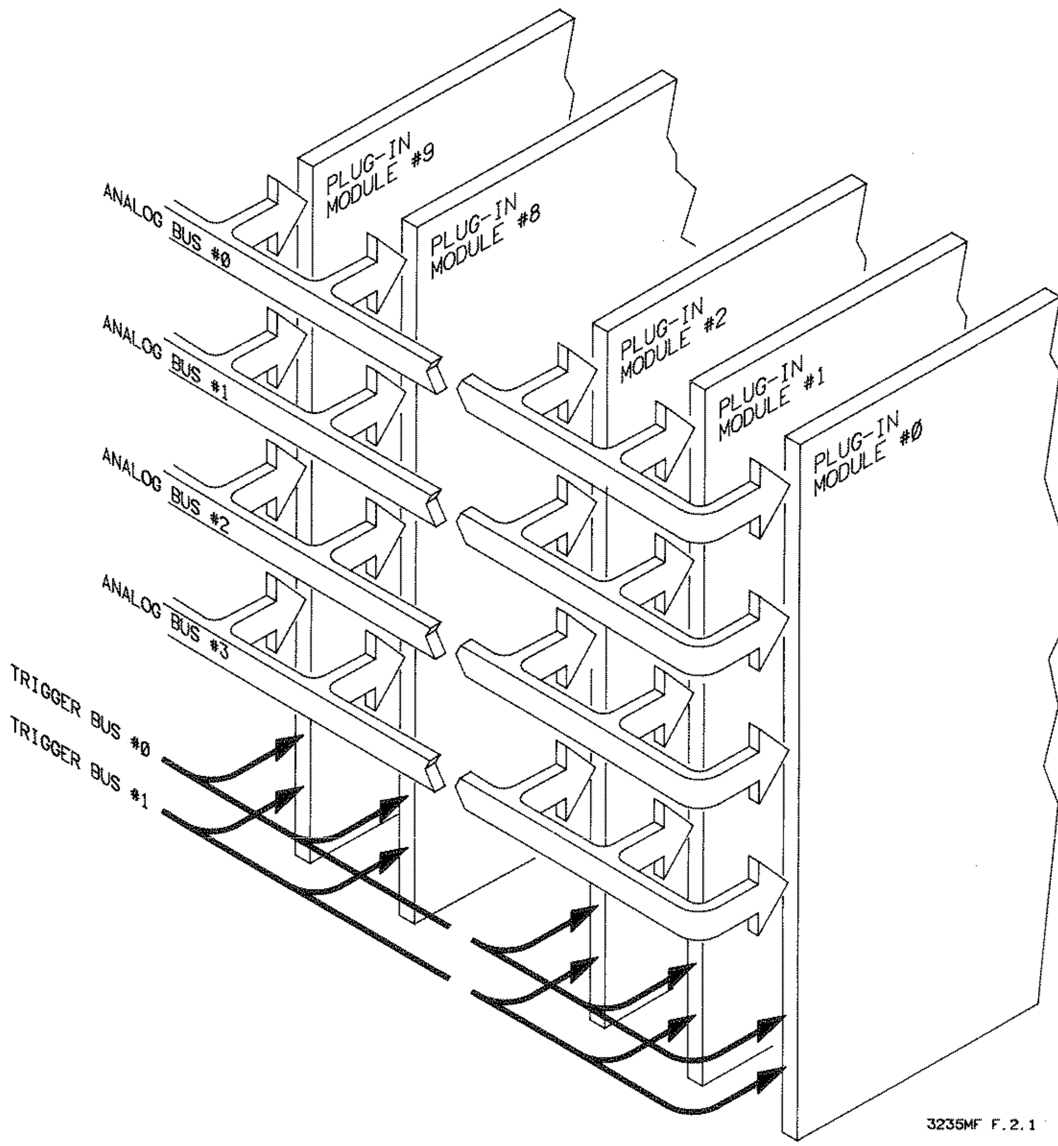


Figure 2-1. Backplane Signal Buses

Trigger Buses

The two trigger buses shown in Figure 2-1 pass digital signals between plug-in modules or across frames (mainframe and extenders). For example, you can pass trigger signals from the External Trigger In and External Trigger Out BNC connectors for scanning with external instruments (see Chapter 6). Other commands allow you to send pulse trains, square waves, or other trigger signals through the trigger buses. The HP 34504 and 34506 coaxial modules also provide access to the trigger buses.

The trigger bus system consists of two bidirectional digital lines within each cardcage, a pair of lines connecting the mainframe to each extender, and switching to provide for internal or external sources for the trigger signals. Refer to Chapter 9 for more information. Common uses for the Trigger Bus lines include carrying:

- Trigger signals input/output through a coaxial module
- Bit serial digital signals
- Measurement Complete signal from the HP 34520 Multimeter
- Sync signals from a signal source
- Scan advance signals
- Pulse and square wave signals
- Event Sense from the HP 34522 Digital I/O module

External Bus Connections

You can connect external instruments to the analog buses via a connector inside the Backplane Access Module (HP 3235 rear panel). The Backplane Access Module replaces the HP 34560 System Expansion Module in mainframes that do not connect to extenders. It provides a direct connection to the analog buses; anything present on the analog buses is also present on the external bus connections. Refer to the HP 3235 Installation Manual for specific information on connecting to the analog bus.

WARNING

SHOCK HAZARD. Only qualified, service trained personnel who are aware of the hazards involved should install, remove, or configure any module. Before touch ing any installed module (card cage or rear panel), turn off all power to the mainframe, extenders, and to all external devices connected to the mainframe, extenders, and modules.

HP 3235E Extender Frames

This section briefly describes the HP 3235E Extender frame and the connections necessary to connect it to an HP 3235A mainframe. It also describes the extender frame addressing.

Each HP 3235E Extender Frame provides an additional 10-slot card cage. You can install any plug-in module in any slot in the extenders. Connecting up to seven HP 3235E Extenders to the mainframe (daisy-chain fashion) allows the mainframe to directly control up to 80 plug-in modules.

Extender Connections

To use the extenders you must have the HP 34560 System Expansion Card (Option 560) installed in the mainframe. The extenders connect to the mainframe by a digital and (optionally) an analog cable. Refer to the HP 3235 Installation Manual for information on connecting the extenders.

Digital Extender Cables

The extenders connect to the mainframe by digital extender cables. The digital extender cables carry data and programming information between the extenders and the mainframe. Each HP 3235E Extender frame includes one of these cables. Maximum (cumulative) digital extender cable length is 10 meters.

Analog Extender Cables

The analog extender cable links the four analog buses of the mainframe to the extenders. With the cable installed, the mainframe and extenders form one continuous electrical circuit. This cable is optional and allows you to connect multiplexer modules or a multimeter module installed in separate frames.

WARNING

The HP 3235 and 3235E and certain plug-in modules allow up to 250V peak to be applied to the analog buses. Voltages present on the buses are also present on the terminals and screws of the analog extender connector. This connector is located inside the HP 34560 Extender Module and must be accessed by qualified, service trained personnel only.

Extender Bus Relays

Each HP 34560 System Expansion Module has four "extender bus relays", one relay for each analog bus. These relays allow you to isolate any one analog bus within a frame from the analog extender cable. At power-on or reset, the HP 3235 closes all four extender bus relays on each frame. Figure 2-2 shows these extender bus relays.

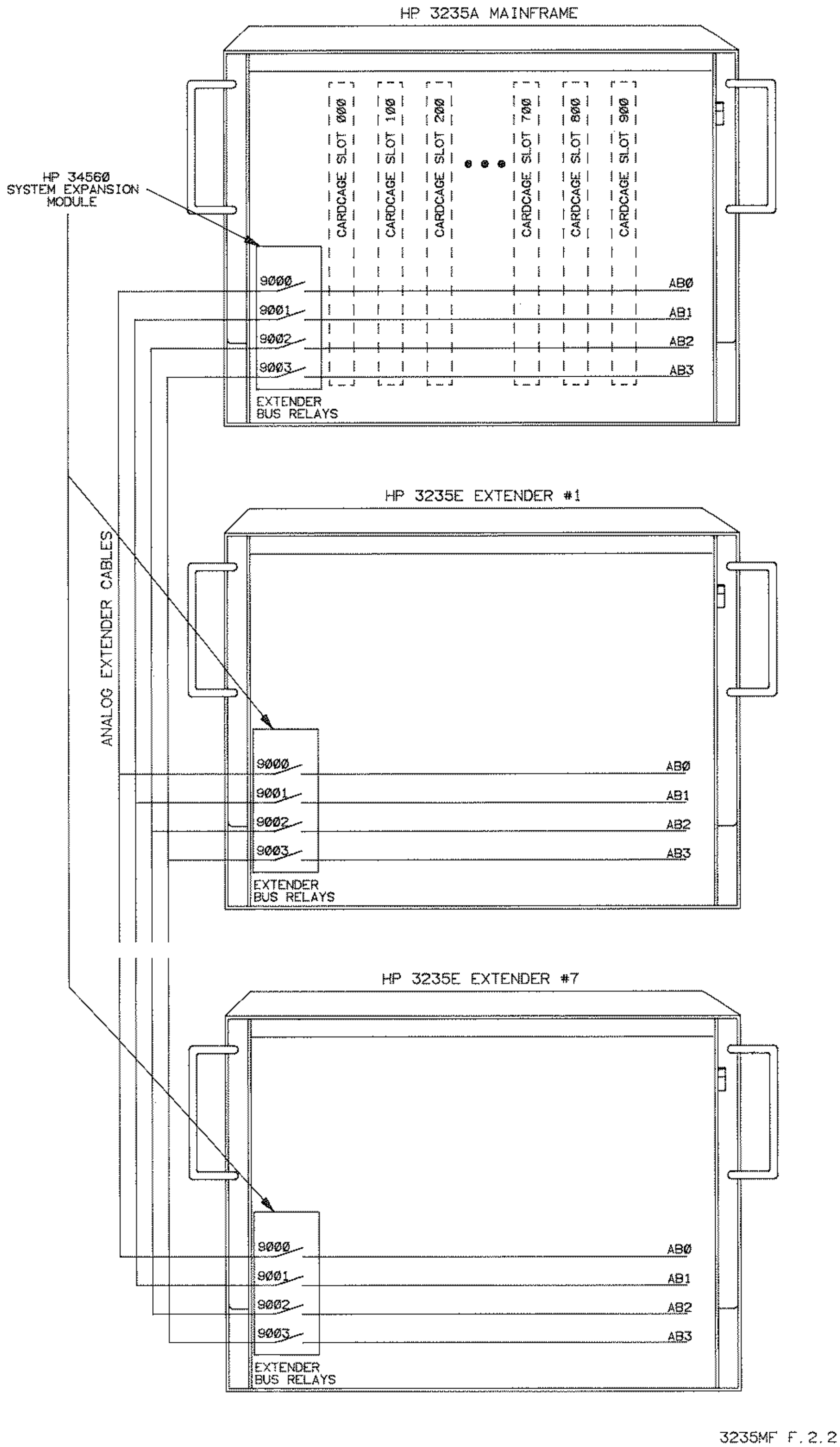


Figure 2-2. Analog Extender Cables and Relays

The **OPEN** and**CLOSE** commands control these relays. The address for these relays is of the form: *9E0N*, where *E* is the frame number (0 for the mainframe, 1-7 for extenders) and *N* is the analog bus number (0-3).

Extender Addresses

Each extender has a unique address set by a switch on its rear panel. The first extender, connected directly to the mainframe, is typically set to address 1, the second extender to address 2, etc. Addressing for plug-in modules installed in an extender is discussed in Chapter 4.

NOTE

The mainframe must have the frame address switch set to 0 (zero).

The **EXTEND?** command returns seven integers (separated by commas) indicating the addresses of attached extenders. If an extender does not exist for a specific address, the integer returned is a 0 (zero). For example, if you have extenders at addresses 1, 2 and 4, the integer string returned would be:

```
1, 2, 0, 4, 0, 0, 0
```

The following program shows how to read the integer string.

```
10 DIM A$[50]
20 OUTPUT 709; "EXTEND?"
30 ENTER 709; A$
40 PRINT A$
50 END
```

The Quick Interconnect Fixture

Figure 2-3 shows the HP 34591A Quick Interconnect Fixture. The fixture fits inside the interconnect mechanism and accommodates both DIN type terminal blocks (for the relay multiplexers, digital I/O module, and the multimeter module) as well as BNC terminal blocks (for the coaxial matrix and multiplexer modules). Refer to the HP 3235 Installation Manual for instructions on assembling plug-in module terminal cards into the fixture.

Optional feedthrough panels (HP 34593A/B/C and 34594A/B) allow external instruments, wired through the rear panel access chamber, to connect through the fixture to the DUT. Refer to the Installation manual for information on using the feedthrough panels.

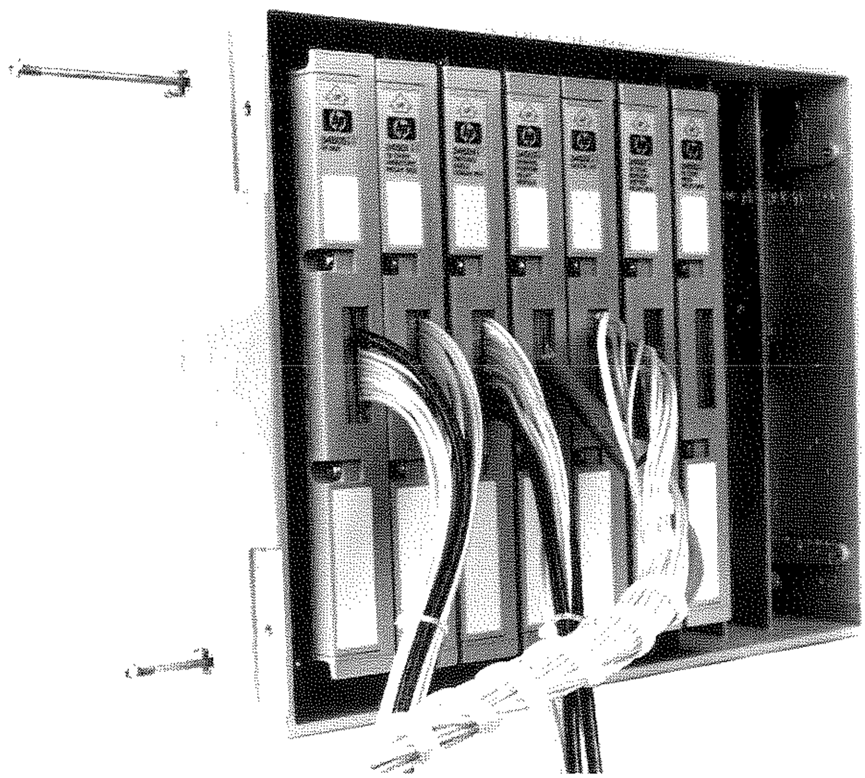


Figure 2-3. Quick Interconnect Fixture

CAUTION

Do not pick up or attempt to carry the HP 3235 by the quick interconnect handles.

Operating the Mechanism

Refer to Figure 2-4. To open the mechanism, grasp both mechanism handles and press the latches down with your thumbs. Pull the handles toward you. Release the lower Latch Bar first, then release the upper Latch Bar. Now you can remove the fixture and install a new one on the mounting pins. Engage the upper Latch Bar first, then the lower Latch Bar. Lock the fixture onto the mechanism before proceeding. Close the mechanism by slowly pushing up on both mechanism handles simultaneously. Make certain that both latches engage the HP 3235 case.

NOTE

The HP 3235 Installation Manual provides a complete description of installing and removing Quick Interconnect Fixtures from the mechanism.

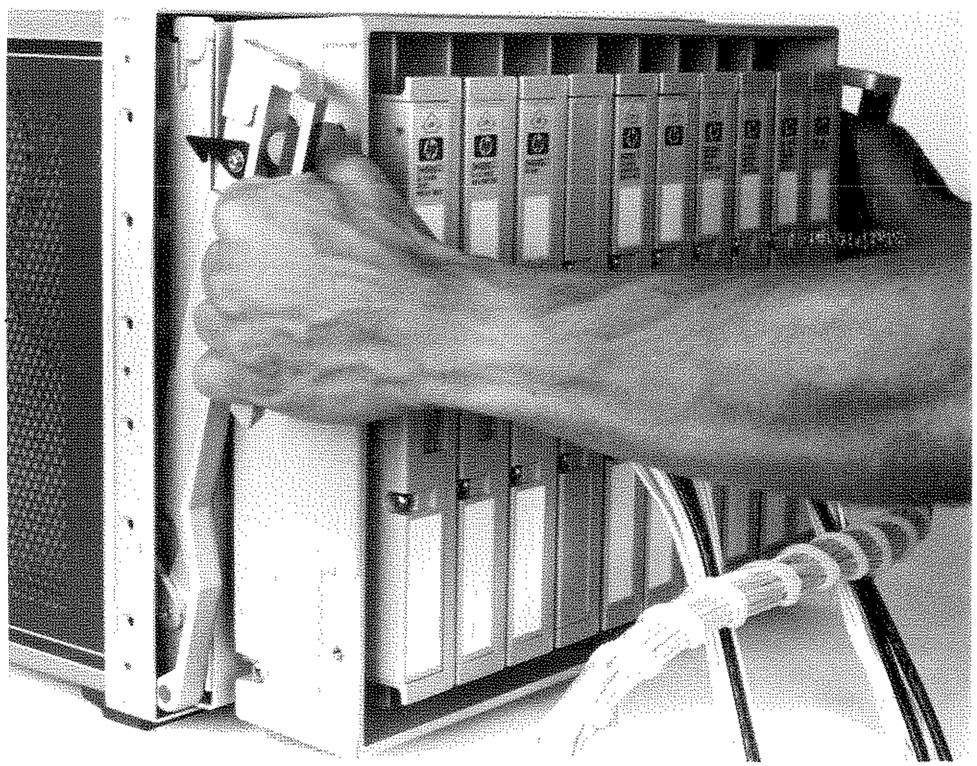


Figure 2-4. Opening the Quick Interconnect Mechanism

An interlock switch monitors whether the quick interconnect mechanism is open or closed. This interlock switch has two sets of contacts: one set interrupts the internal processor when the mainframe mechanism opens the second set is available for user connections. You can also program the HP 3235 to interrupt the system controller when the mainframe quick interconnect mechanism opens (see Chapter 10). If the mainframe mechanism opens, the HP 3235 aborts any commands in progress but does not change the state of any relays. The HP 3235 mainframe does not sense the state of any Quick Interconnect mechanisms mounted on the HP 3235E Extender frames.

WARNING

A lethal voltage may be present on the plug-in module's terminal connections when the fixture is open. This condition may exist if a high voltage is applied through the analog bus connections on the rear panel through one of the relay multiplexer modules or the Feedthrough Panels.

Do not handle either the fixture or any of the plug-in modules until verifying that all power sources have been removed.

The HP 3235 takes no action when the fixture closes. The operator should resume program execution.

Querying the Fixture

You can determine if the quick interconnect mechanism is open or closed with the **FIXTURE?** query command. The **FIXTURE?** command returns a "0" if the mainframe mechanism is open or if no fixture is installed. It returns a "1" if the mainframe mechanism is closed. The **FIXTURE?** command cannot determine the status of any HP 3235E extender mechanisms. Refer to the **FIXTURE?** command in the HP 3235 Language Reference Manual for more information.

Connecting to the Interlock Switch

The second set of interlock switch contacts is available in the rear panel access chamber. These contacts are open when the mechanism is closed. They can provide added safety by sensing when the mechanism opens and signaling the system to remove power from any high voltage source connected to an HP 3235 plug-in module or backplane. For information on connecting to these interlock switch contacts, refer to the HP 3235 Installation Manual.

For example, you might connect these contacts through a power supply to the coil on a normally-open relay. The relay contacts could connect to some high voltage source. When the mechanism opens, the relay opens thus disconnecting the source from the HP 3235 module or backplane.

Although the **FIXTURE?** command works only with the HP 3235 mainframe, these interlock switch contacts are available on both the mainframe and extenders.

Floating Measurements and Guarding

This section discusses how and why to use the guard connection when making floating measurements. For more specific information on guarding, contact your local HP Sales and Service office for a copy of HP Application Note 123, Floating Measurements and Guarding.

Measurements where a voltage difference exists between the signal source ground and the HP 3235 ground are floating measurements. Thus, measurements are said to float by the amount of common mode voltage. Figure 2-5 is a simple conceptual diagram of a floating instrument showing possible common mode noise currents.

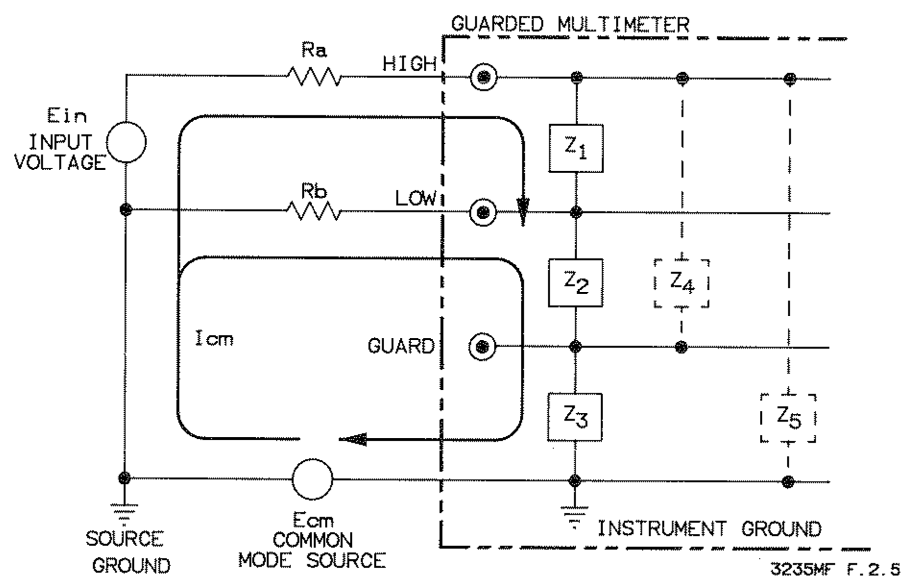


Figure 2-5. Simple Floating Multimeter

Non-Guarded Measurements

As shipped from the factory, the guard connections inside the HP 3235A mainframe and 3235E extender frames connect to the frame chassis (earth ground) through a shorting bar on the Guard Isolator Board. The Guard Isolator Board is located inside the rear panel access chamber. Refer to the Installation Manual for the exact location of the shorting bar.

This shorting bar must be in place if (1) you have either the HP 34504 Coaxial Multiplexer Module, the HP 34505 RF Multiplexer, the HP 34506 Coaxial Matrix Module, or the HP 34508 RF Multiplexer Module installed, and (2) any other module in the same frame or circuit is switching more than 42V peak. The outer BNC conductor on these modules may be connected to the guard wiring (see the respective Plug-in Module Manuals). In the event of insulation breakdown, the shorting bar in place prevents high voltage from being present on the guard wiring.

Figure 2-6 shows a conceptual picture of a guarded instrument with the guard connected to chassis. Although this is different than shown in the HP Application Note 123, this is the recommended guard connection.

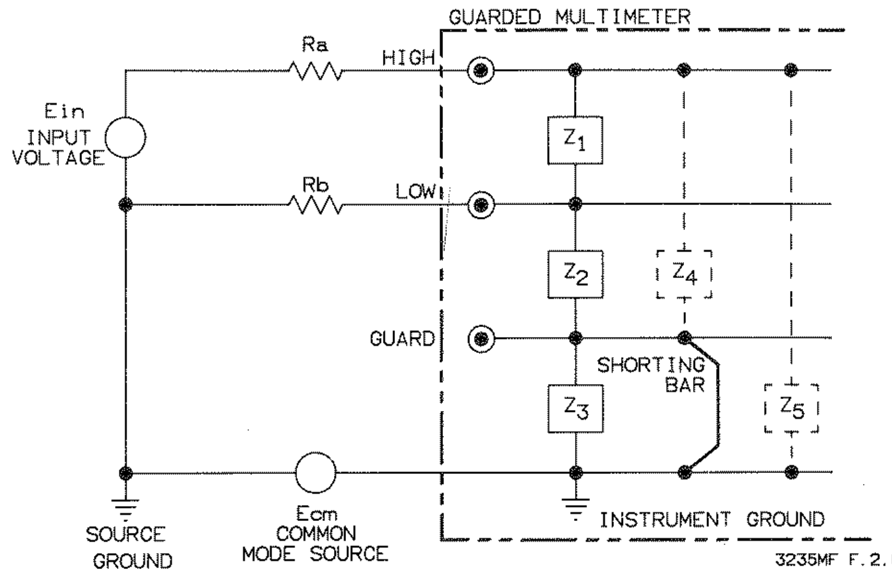


Figure 2-6. Chassis Guarded Instrument

If the analog extender cable is not connected between frames then each frame is considered a separate circuit. Thus, you can have one frame with the shorting bar removed and another frame with the shorting bar in place.

WARNING

Do not defeat the third wire on the AC power cord. The third wire connects the HP 3235 chassis to earth ground. The power cord must either be plugged into an approved 3-contact electrical outlet or used with a 3-contact to 2-contact adapter with the grounding wire (green) firmly connected to an electrical ground (safety ground).

WARNING

When guard is isolated (shorting bar removed) from earth ground (floating) in an HP 3235 with a module containing BNC connectors (HP 34504, 34505, 34506, 34508, or 34521), you must limit all external inputs to less than 42V peak (referenced to earth ground). If you exceed this limit and a fault occurs, hazardous voltages could be applied to exposed metal on the BNC connectors and cables. This condition could cause injury if anyone touches the exposed metal on the BNC connectors or cables.

Guarded Measurements

If you need to isolate (float) the DUT ground from the test system ground, then open the shorting bar. This allows the system to make guarded measurements and reduce common mode voltage problems. In the HP 3235, do not allow the guard wiring to float more than 42V above chassis. This section describes how to use the guard connection in the HP 3235.

Connecting the Guard

Each relay multiplexer module has 24 Guard connections. All guard connections tie together on the plug-in module. Refer to Figure 2-7. If you are going to connect the guard, then for best common mode noise rejection all signal sources must share the same ground. With the shorting bar open the guard connections tie to the guard frame inside the instrument frame but not to chassis ground.

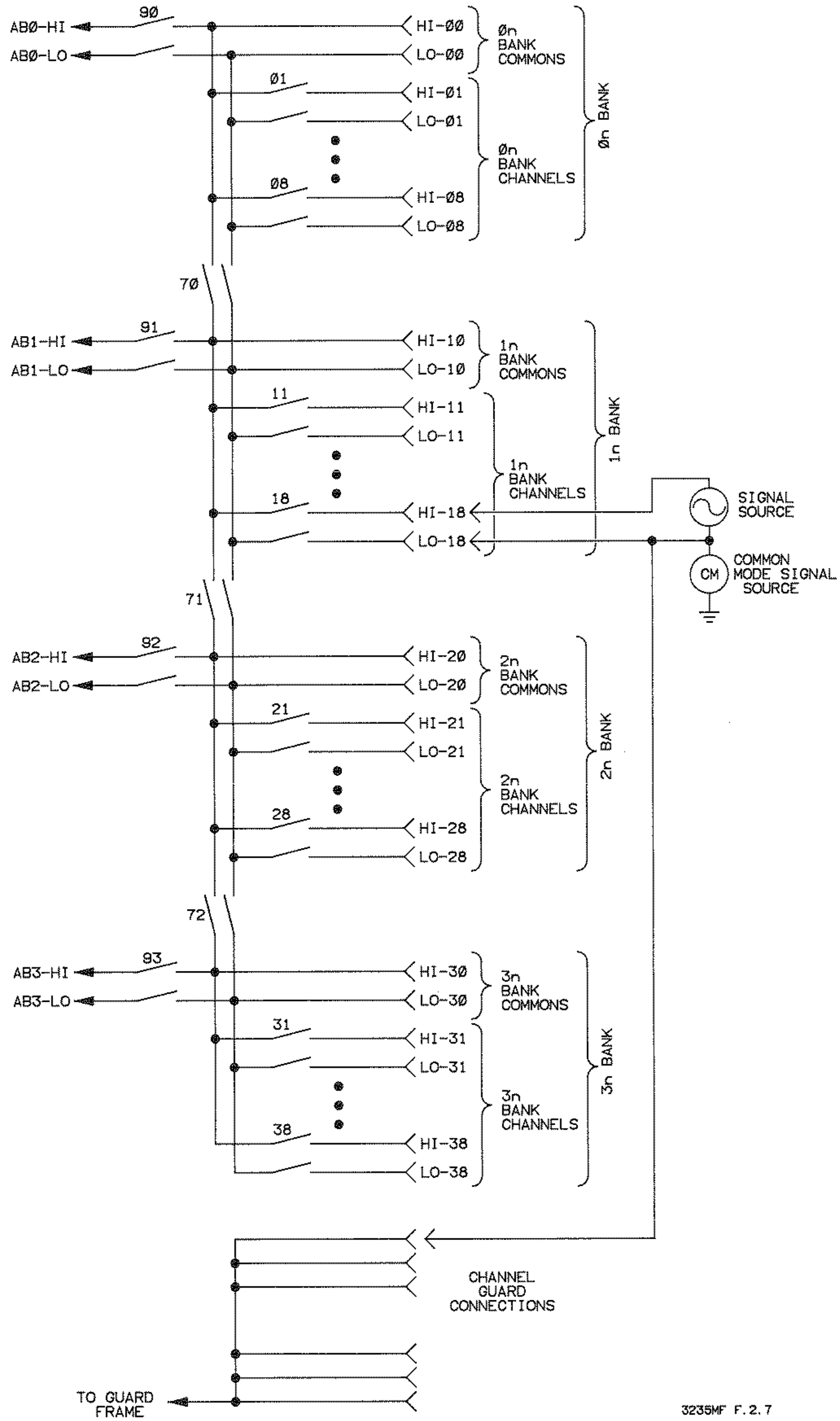


Figure 2-7. HP Relay Multiplexer Simplified Schematic

Connect the guard terminal to the "low" connection of your source signal as shown in Figure 2-8. The guard connection shunts common mode currents away from the test lead resistances. Do not connect the guard to the "LO" on the multiplexer terminal block as this could increase common mode noise.

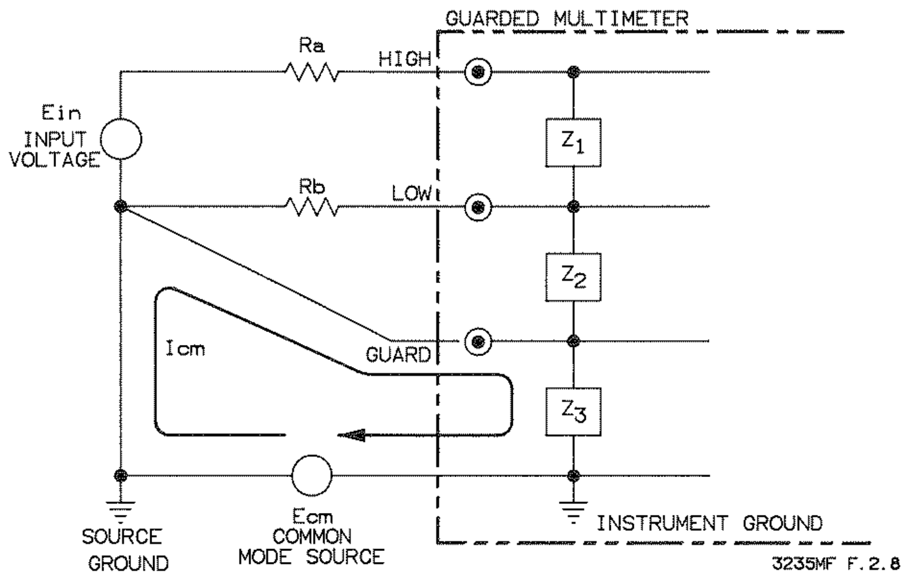


Figure 2-8. Proper Guard Connection

Switching the Guard

For guarded measurements where the source grounds are at different potentials, you should use a paired scanning technique. Refer to Figure 2-9. Select one bank of a multiplexer module as the signal source bank. Connect the source grounds to the HI's on the second bank. Connect the HI common of the second bank to one of the guard terminals. Now to make a guarded measurement, you must close two channels; one in bank 1 for the actual signal, and the corresponding channel in the second bank for the respective guard.

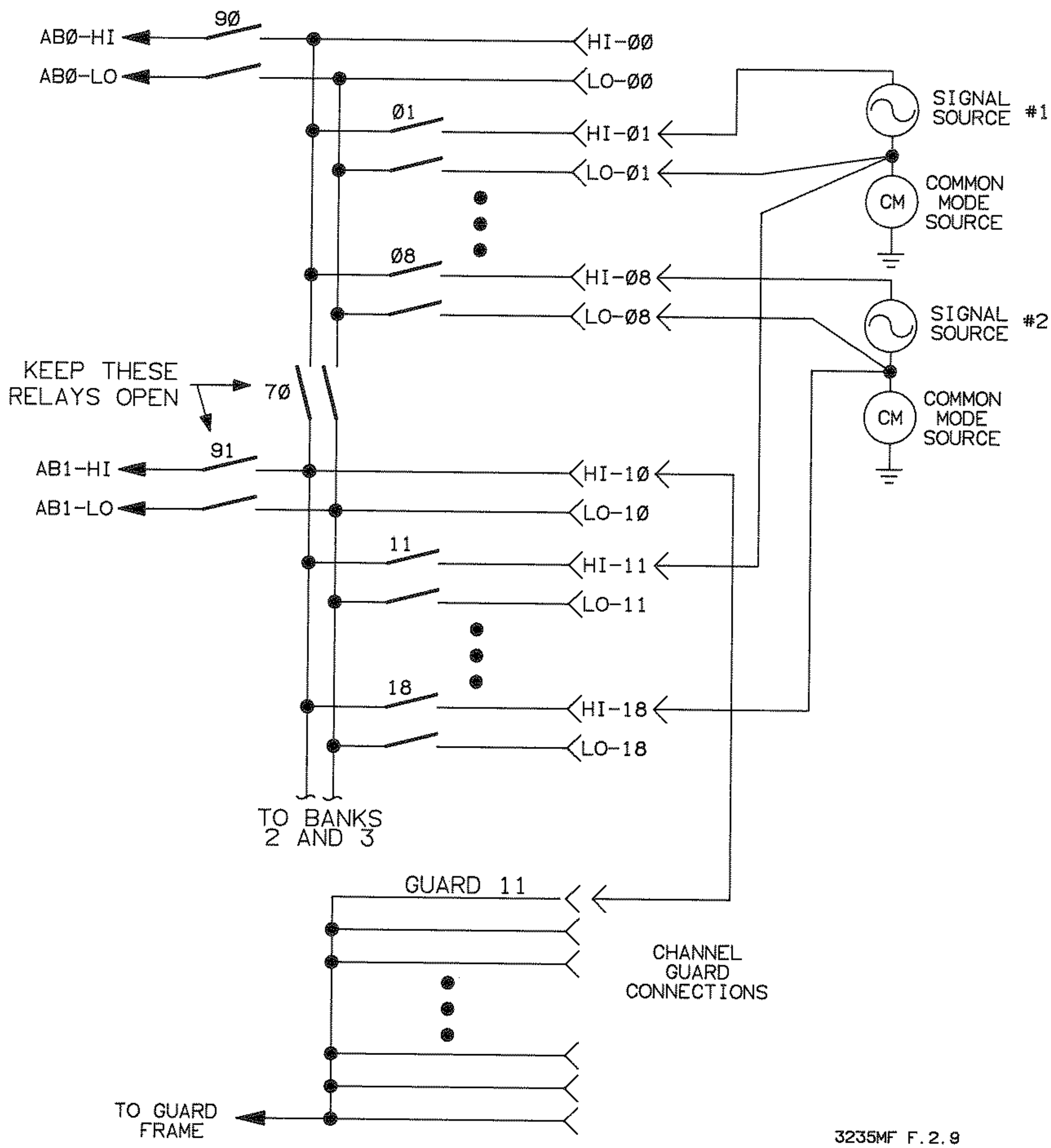


Figure 2-9. Guard Switching Diagram

Power Fault

A power fault condition occurs when either (1) the HP 3235 overheats or (2) an internal power supply experiences an over-voltage condition. When a power fault occurs, the internal power supplies shut down and the power fault indicator light on the instrument rear panel turns on.

The HP 3235 may overheat if the temperature of the operating environment exceeds the environmental specifications. For example, the HP 3235 may overheat if it is rack mounted with several other instruments and no ventilation is provided in the rack.

Restoring Operation

If a power fault condition occurs, you can restore operation by either cycling the power switch or by removing the AC power cord. If you cycle the power switch to the off position, the power fault light will remain on (indefinitely) until you cycle the switch to its on position or remove the AC power cord. When you turn the power switch on, the power supply resets and turns the power fault light off.

If you remove the AC power cord, you must wait for the power fault light to extinguish before replacing the power cord. Depending on what caused the power fault condition, this may take several minutes.

NOTE

If the fault condition is not corrected, the HP 3235 returns to the power fault condition. Check to ensure that the HP 3235's internal cooling fan is working and that the instrument is adequately ventilated. If the power fault condition persists, refer to the HP 3235 Service Manual.

Chapter 3
HP 34550 Control Panel

Introduction

This chapter discusses the optional HP 34550 Control Panel. If you do not have the Control Panel, skip this chapter. Figure 3-1 shows the control panel for your HP 3235. This control panel can help you debug your test system and operators can use it to enter data or respond to system queries. This chapter discusses the following features:

- The Display
- The Keyboard
- Redefining the Keyboard
- Locking out the Keyboard
- The Monitor Mode
- Other Command Execution

This chapter covers information unique to the control panel. Chapter 4 in this manual covers remote programming information. You can execute all HP 3235 commands from either the control panel or remotely from the system controller.

Display

The control panel display is a 40 character window in a 256 character display. Left and right arrow keys scroll the display. Eleven on/off annunciators indicate the current status of the instrument. Pressing the **Clear Display** key erases the current display.

Edit Keys

Edit keys provide typing aids when executing commands from the keyboard. Use the right and left arrow keys to shift long displays (e.g. error messages).

Immediate Execute Keys

These immediate execution keys clear any partial keyboard entry and execute the specific command. These keys can be active even if the HP 3235 is in remote or executing a subroutine.

Alphabetic, Symbol, and Command Header Keys

The keyboard can execute any of the HP 3235 commands. Twenty-six command headers have been predefined in functional groups. At power-on the keyboard is in the letter mode. Use the **Shift** or **Shift Lock** keys for command headers. To execute a command, manually type it in or use the pre-defined keys. Press **Enter** to execute the command.

Numeric Keypad and User Defined Keys

The 10 number keys have a **User** function for defining long command strings.

Figure 3-1. HP 34550 Control Panel

The Display

The control panel display is a 40 character window into a 256 character display buffer. Eleven annunciators provide on/off status indications for HP-IB activity and mainframe status. The eleven annunciators, left to right, and their functions are:

- **SHIFT** When this annunciator is on it means that the keyboard is in its shifted mode.
- **SRQ** When the HP 3235 asserts the HP-IB SRQ interrupt line, this annunciator turns on. Refer to Chapter 10 for information on interrupts.
- **TALK** This annunciator turns on when your system controller addresses the HP 3235 to talk (i.e. send data).
- **LSTN** This annunciator turns on when your system controller addresses the HP 3235 to listen (i.e. receive commands or data).
- **REM** When the system controller places the HP 3235 in its remote state, this annunciator is on.
- **ERROR** When an error occurs, this annunciator is turned on. See Chapter 4 for information on handling errors.
- **READY** This annunciator indicates that the HP 3235 is ready to accept new commands. See the READY state defined in Chapter 2.
- **SUBROUTINE** This annunciator turns on when your system controller is downloading a subroutine. Refer to Chapter 8 in this manual for information on subroutines.
- **RUN** This annunciator turns on when the HP 3235 is **RUN** executing a subroutine. See Chapter 8 in this manual for informations on subroutines.
- **KEY** The **KEYS ON** mode turns this annunciator on. See the last section in this chapter.
- **USER** This annunciator turns on when a user has pressed the **USER** key. See User Defined Keys later in this chapter.

Display Messages

The display can hold four classes of messages. The four classes are (highest priority first): keyboard echo, error messages and single displays, monitor, and idle display. The **Clear Display**, **Device Clear**, or **Enter** keys clear the current message and display the next priority message.

Keyboard Echo

In this mode, the display echoes keyboard entries. If the entry is longer than 40 characters, the display scrolls to show the cursor position. The HP 3235 processor does not allow any other display messages while you are entering a keyboard command. Therefore, you must either press the **Enter** key (which enters the displayed command string) or clear the display to release the display for other uses. When you execute a command from the keyboard, the command stays in the display until (1) it finishes execution or (2) the command generates another display.

Error Messages and Single Displays

Error messages contain an error number and a description of the error. They remain in the display until (1) another error message occurs, (2) you write a message to the display (see the **DISP** command), or (3) you use the keyboard (keyboard echo mode). Refer to the Chapter 4 for more information on recalling and clearing errors.

Single displays include responses to queries, **DISP** command statements, digital I/O read statements, or single measurement results from the multimeter. Error messages overwrite the single display.

Commands such as **CAT** (subroutine catalog) and **LIST** *name* (subroutine list) return several lines of information. When you execute these commands from the keyboard,the display shows the message:

```
PRESS Single Step TO CONTINUE DISPLAY
```

Pressing the **Single Step** key shows the results of the command, one line at a time. You can abort the command by pressing the **Clear Display** key.

Monitor Mode

You can use the **MON** command to display multimeter readings, module states, or echo commands from the system controller. For more information, refer to the section on Monitor Modes later in this chapter or the **MON** command in the Language Reference Manual.

Idle Display

If no other display messages are active, the display shows READY. This is also the power-on display unless an error occurs.

Writing to The Display

You can write messages to the control panel display from your system controller, the keyboard, or from subroutines with the **DISP** (or **DSP**) command. For example:

```
OUTPUT 709; "DISP 'DISPLAY THIS MESSAGE'"
```

would cause the control panel display to show the message:

```
DISPLAY THIS MESSAGE.
```

NOTE

Clear the display by sending the **DISP** *followed by two single quote marks. For example:*

```
OUTPUT 709; "DISP ''"
```

Turning the Display Off

You can slightly increase the overall HP 3235 system speed by turning the control panel display off with the **DISP OFF** command:

```
OUTPUT 709; "DISP OFF"
```

This sets the display to a series of hyphens across the display; all annunciators are off. Use the **DISP ON** command or press any front panel key to return the display to normal operation.

Returning the Contents of the Display

The **DISP?** command returns the current contents of the display to the system controller. For example, to send the contents of a **MON STATE** display to the system controller, execute:

```
10 DIM A$[256]
20 OUTPUT 709; "DISP ' '"        ! CLEAR THE DISPLAY
30 OUTPUT 709; "MON STATE 800"   ! MONITOR STATE OF SLOT 800
40 WAIT .5                       ! WAIT FOR THE COMMAND TO DISPLAY
50 OUTPUT 709; "DISP?"           ! RETURN DISPLAY CONTENTS
60 ENTER 709; A$                 ! ENTER DISPLAY CONTENTS
70 PRINT A$                      ! PRINT DISPLAY CONTENTS
80 END
```

The Keyboard

Refer to Figure 3-1. The control panel has a QWERTY style keyboard with a separate numeric keypad. Just to the right of the display are four immediate execute control keys and eight edit keys. The shifted alphabet and punctuation keys provide command headers for easy command execution. You may also redefine any of the shifted alphabet and symbol keys and assign long command strings to the user (numeric) keys (see the **DEFKEY** command later in this chapter).

You can execute any HP 3235 command from either the keyboard or the system controller. The display echoes all keyboard entries until you press the **Enter** key. The maximum length for a keyboard command string is 128 characters and the display scrolls so that the cursor is always visible.

Alphabet, Number, and Symbol keys

At power-on, the keyboard is in the letter mode.

For example:

press:

RSTATE [B] SCAN [E] SCAN [E] RESET [P]

[Enter]

Your HP 3235 beeps once. This audible beep sounds when an error occurs or you can program it to alert an operator.

Use the **Shift** and **Shift Lock** keys to display the predefined command headers. These two keys operate similar to a standard typewriter's shift and shift lock keys. That is, the **Shift** key shifts only the next key pressed. The **Shift Lock** key stays in effect until you turn it off. When the keyboard is in the shift lock mode, the **Shift** or **Shift Lock** keys return the keyboard to the unshifted mode. Note that when the SHIFT annunciator at the left end of the display is on, the next key pressed is shifted.

press:

[Shift] ADDR? [>] [Enter]

This displays the HP-IB address of the HP 3235. The factory set address is 09.

Immediate Execute Keys

The four immediate execute keys clear any partial keyboard entry and then execute. These keys are:

- **Device Clear**: Acts the same as the HP-IB CLEAR command (see Appendix D). It clears the HP-IB input and output buffers and the error register. It also aborts any running subroutine and turns on the display (if the display was off). It does not, however, change the state of any plug-in module.
- **Local**: Takes the HP 3235 out of the HP-IB Remote state and puts it in the local mode (REM annunciator off). The local mode allows full use of the keyboard. In the remote mode, commands executed from the keyboard which alter the state of the instrument (e.g. **CLOSE** and **OPEN**) result in an error; keyboard commands which query the instrument (e.g. **CLOSE?**) execute without error. The HP-IB LOCAL LOCKOUT command or the HP 3235 **RMT** command disable the **Local** key.
- **Single Step**: Has two functions: (1) execute one line of a subroutine at a time or (2) advances to the next line with the **CAT** and **LIST** commands. Refer to Chapter 8 for more information. Abort the single step feature, for multiline displays with the **CAT** and **LIST** command, by pressing the **Clear Display** key.
- **Repeat**: Re-executes the last keyboard command string executed.

Edit Keys

Edit keys allow you to alter unexecuted command entries in the display. These keys are:

- **<, <<, >** and **>>**: Move (scroll) the display window for reading command entries and instrument messages. The double arrow keys move the display 20 characters. The arrows show the direction the cursor moves, even if the cursor is not visible (as with error messages).
- **Clear Display**: Clears the current display and displays the next level message. The lowest level message is the *READY* message.
- **Insert Char** and **Delete Char**: Modifies the displayed command string at the cursor position.
- **Last Line**: Recalls the last command string executed from the keyboard to the display for editing.

Redefining the Keys

The **DEFKEY** command allows you to redefine any of the shifted alphabet, or punctuation keys and assign long command strings to the special **User** keys. You cannot redefine the edit keys, **Shift** or **Shift Lock** keys, math operator keys (**+**, **-**, *****, **/**, **E**, **.**, and ^), number keys, **Backspace**, **Space**, **User**, or **Enter** keys. These redefined key assignments are stored in continuous (battery backed-up) memory and remain assigned even if the mainframe loses power.

Redefining the Alphabet and Punctuation Keys

You can redefine the alphabet and punctuation keys with up to ten (10) characters. These may include any command header. For example:

---

press:

	DEFKEY		"	SELECT	"	DEFKEY
Shift	,	Space	"	W	"	,

"	FUNC	RUN	CONT	SCAN	"
"	T	I	M	E	"

[Enter]

Or execute from your system controller:

```
OUTPUT 709; "DEFKEY 'W','TIME'"
```

This redefines the shifted 'W' key to the **TIME** command header. **TIME** returns the current time of day in seconds past midnight (see Chapter 9 in this manual for additional information). To execute the new definition, press:

	SELECT	
Shift	W	Enter

Redefining the User Keys

You can redefine the ten **User** keys with up to 80 characters each. The User keys are associated with the ten number keys. The User keys can include any command header(s) and associated parameters. They may also include scan lists (see Chapter 6). For example:

---

press:

	DEFKEY			DEFKEY
Shift	,	Space	5	,

"		OPEN				;
"	Shift	Q	1	Ø	1	;

	CLOSE				;
Shift	A	1	Ø	2	;

	MON STATE				"
Shift	N	1	Ø	Ø	"

[Enter]

This sequence defines user key 5 to open channel 101, close channel 102, and monitor the Plug-in Module in slot 100. To execute the redefined number key press:

[User] [5] [Enter]

Storing and Loading the Redefined Keys

The **DEFKEY?** *key* command returns the redefined command string for the specified key. You can store and load the key assignments from your system controller; however, you cannot execute the redefined key assignments from the system controller.

The following program example shows how to read a key definition and re-instate it from your system controller:

```
10 DIM A$[10]
20 OUTPUT 709;"DEFKEY? 'G'"! READ THE CURRENT ASSIGNMENT
30 ENTER 709; A$! ENTER INTO CONTROLLER
40 PRINT A$ !  PRINT THE CURRENT ASSIGNMENT
50 OUTPUT 709;"DEFKEY 'F',";A$ !  ASSIGN IT TO ANOTHER KEY
60 END
```

Returning to the Original Definitions

The **DEFKEY DEFAULT** command returns all redefined alphabetic and punctuation keys to their original definition. The ten **User** keys are unchanged.

Locking Out the Keyboard

You can lockout the entire keyboard by executing the **LOCK ON** command. All the keys, including **Local** and **Device Clear**, become unusable. If you execute **LOCK ON**, you can use **LOCK OFF** to regain use of the keyboard. For example:

```
OUTPUT 709; "LOCK ON"      ! LOCKS OUT KEYBOARD

OUTPUT 709; "LOCK OFF"     ! UNLOCKS KEYBOARD
```

Normally, when the HP 3235 is in the remote state, the **Local** key returns the mainframe to its local mode. That is, it executes all commands from the keyboard. The **RMT** command prevents the control panel **Local** key from returning the mainframe to its local mode. To re-enable the keyboard **Local**key, execute the **LOCAL** command.

HELP with Commands

If you forget which parameters are valid for a particular command, simply type the command, follow it with a colon (:) and press **Enter**. The HP 3235 displays all the parameters that are valid for that command. For example; press:

FUNC [T] MEAS [R] RUN [I] ACV [G] TEST [:] [Enter]

The display shows all the valid parameters for the **TRIG** command.

Press the **Last Line** key to return the command strig to the display. Press the **Backspace** key to delete the colon. Now type in the appropriate parameter(s) and press the **Enter** key to execute the command.

NOTE

In strings with multiple commands, the HP 3235 executes all complete and correct commands prior to the one with the colon. For example, if you typed:

CLOSE 101; OPEN:

the HP 3235 closes channel 101 then displays the valid parameters for the **OPEN** *command.*

Monitor Modes

The control panel has six different monitor modes. These six modes are:

- **MON HPIB**: Display command headers with parameters as the HP 3235 receives them from the system controller and executes them.
- **MON** *ch#*: Display every multimeter measurement from one specified channel during a **MEAS** or **VERIFY** scanning sequence (see Chapter 6 for more information on scanning).
- **MON ALL**: Display every multimeter measurement from every channel during a **MEAS** or **VERIFY** scanning sequence (see Chapter 6 for more information on scanning).
- **MON STATE** *slot#*: Display the state of the plug-in module in a specified slot.
- **MON STATE ALL**: Display the state of any plug-in module after a command modifies the state of that module.
- **MON OFF**: Turn the monitor modes off.

NOTE

Monitor displays have a low priority. Any error or display messages or the keyboard echo overwrites the monitor mode displays. Clear the display before executing the monitor mode commands. If you execute the **MEAS** *command from the control panel, the multimeter measurements overwrite the* **MON** *ch# or* **MON ALL** *displays.*

The following sections describe the five monitor displays.

Monitor HP-IB

In this mode the control panel display echoes all command headers and parameters sent to the HP 3235 from the system controller. The following example shows how this is done:

	MON	SSTATE	RESET	RUN	RSTATE	
[Shift]	[J]	[H]	[P]	[I]	[B]	[Enter]

The control panel display shows:

```
HP-IB:  Waiting ....
```

Now, execute a command from your system controller. For example:

```
OUTPUT 709; "CLOSE 101"
```

The control panel display shows:

```
HP-IB:   CLOSE 101
```

If you execute several commands joined into one string, the control panel individually displays each command header with its associated parameter but they may execute so quickly that you don't see them. The last command remains in the display.

Monitor One Channel

The **MEAS** command scans through and measures one or more channels. The **MON** *ch#* command monitors and displays the multimeter measurements from one specified channel in the **MEAS** channel list. The control panel display is updated only when that channel is measured during the **MEAS** or **VERIFY** scanning process. Refer to Chapter 6 for more information on scanning.

For example, if you want to monitor channel 613, execute:

[Shift] [J] (MON) [6] [1] [3] [Enter]

The display shows:

```
Chan ????:  Waiting..."
```

Then, after you execute the **MEAS** command that includes channel 613, the display shows the channel number and the multimeter reading. For example:

```
Chan   613:-2.4050900E-01
```

The actual multimeter display updates each time the channel is measured.

Monitor All Channels

As the **MEAS** command scans through and measures each channel in a channel list, the **MON ALL** command displays each multimeter reading for each channel in that list. You need to either add a small delay time to the scan or use a slow, externally paced scan in order to watch each channel's measurement.

The display format is identical to that described under Monitor One Channel except that the channel number is also updated. Refer to Chapter 6 for more information on scanning.

Monitor a Plug-in Module State

The **MON STATE** *slot#* command displays the state of the plug-in module in the specified slot. If the module is a relay module, the display shows which relays are closed. If the module is the HP 34520 Multimeter, the display shows every reading the multimeter makes, the multimeter function and range for each reading, the channel number, and more. If the module is the HP 34522 Digital I/O module, the display shows the status of the 32 individual bits, status of the eight interrupt event bits, and more. Refer to the appropriate Plug-In Module Manual for the format used for specific plug-in modules.

Monitor All Plug-in Modules

The **MON STATE ALL** command displays the state of the latest plug-in module modified by a command. The format for each module's display is described in the Programming and Configuration manual for that plug-in module.

Turning Monitor Off

To turn the monitor mode off either press the **Device Clear** key or execute:

[Shift] [J] (MON) [O] (CAT) [F] (DCV) [F] (DCV) [Enter]

Using the Control Panel for Data Entry

Your system controller may send the **KEYS ON** command which puts the control panel in a data entry mode. This mode allows the test system operator to respond to system queries. In this mode, the control panel sends keyboard entries to the HP-IB output buffer instead of interpreting them as commands. The **KEYS OFF** command returns the keyboard to its normal operation. For example:

```
10 OUTPUT 709; "DSP 'ENTER SERIAL NUMBER: '"
20 OUTPUT 709; "KEYS ON"
30 ENTER 709; A$
40 PRINT A$
50 OUTPUT 709; "KEYS OFF"
60 END
```

The HP-IB **CLEAR** command, or the control panel **Local** and **Device Clear** keys also return the control panel to normal operation.

Chapter 4

HP 3235 Programming Overview

Introduction

This chapter shows you how to send commands to the HP 3235, and describes several of the mainframe commands. It also provides instructions for addressing modules, handling errors, and using the **HELP** function.

Install your HP 3235 according to the instructions given in the Installation Manual before proceeding through this chapter. Connect the HP-IB cable from the HP 3235 to your system controller. You should perform the examples in this chapter to acquaint yourself with the HP 3235's operation.

Command Execution

The HP 3235 executes commands in the same sequence it receives them. It finishes executing one command before it begins executing the next.

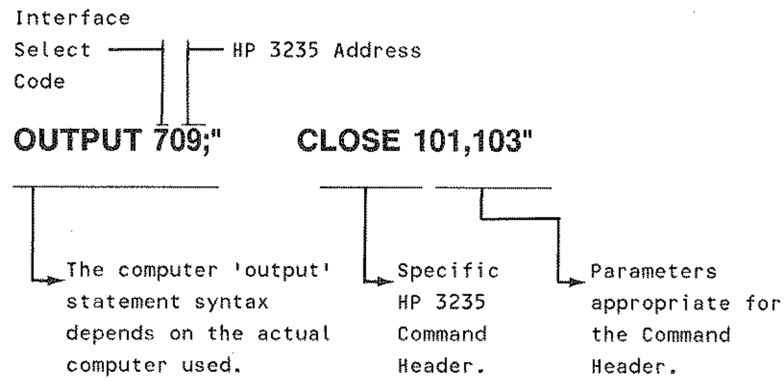
Results of commands executed from the control panel keyboard are sent to the control panel display. Results of commands executed from your system controller are sent to the HP 3235's HP-IB output buffer. In the power-on mode, the output buffer holds the result(s) of the last command executed that produced output data. Any new output data overwrites existing data. See Chapter 11 for more information on the HP-IB output buffer.

Note on Remote HP-IB Programming

Your HP 3235 communicates with a system controller over the HP-IB interface bus. The following information on remote programming is important regardless of the type of system controller used.

Each instrument connected to the HP-IB interface bus has a unique address. The address allows the system controller to select individual instruments when sending commands or receiving data. Refer to the Installation Manual for information on setting the HP-IB address.

All the examples in this manual use the following format for sending commands to your HP 3235:



The word **OUTPUT** is a statement specific to certain HP computers for sending commands or data out of the computer. The number 709 refers to the computer's interface select code (ISC), which is 7, and the HP 3235's HP-IB address which is factory preset to 09. Only one instrument connected to a particular HP-IB interface may have the address 09. If you need to change the address of your HP 3235, refer to the Installation Manual. All examples in this manual have the HP 3235 set at address 09.

All program examples in this manual use an enhanced BASIC programming language such as the HP Series 80, Series 200, or Series 300 desktop computers use. If you aren't using an HP series computer, read through this chapter anyway. Be sure to read the I/O programming manual for your system controller to find out how to send commands and receive data.

Command Termination

Valid command terminators are semicolons (;), carriage return (*cr*), and linefeed (*lf*), or EOI asserted concurrent with the last character sent. The terminator initiates the execution of the command. The next character in the command string is held off until the command finishes execution. Since most controllers send *crlf* after each output string, sequential execution of the command string occurs. Refer to Chapter 11 for additional information on command execution.

Format Conventions

Either a comma or a space must separate command headers from the first parameter. Likewise, either a comma or a space must separate subsequent parameters.

Internal HP 3235 variables (see Chapter 7) can replace numeric parameters in all commands. For example:

```
OUTPUT 709; "LET Z=103; CLOSE Z" ! CLOSES CHANNEL 103
```

All data formats are compatible with IEEE 728 standards. Data formats are:

Alphanumerics: Lower case is equivalent to upper case. User defined HP 3235 variable names are 1 to 9 characters long. The first character must be a letter. Remaining characters may be either letters, numbers, the underline character (_), or a question mark (?).

Numbers: Numbers must be either integer, real, or scientific notation (±ddd.dddE±ddd). See Chapter 7.

Quoted String: Quoted Strings include any characters surrounded by either " or '. Either "" or '' represent quotes within a command string. See the **DISP** command for examples.

Math: The symbols *, /, <, >, =, (,), +, -, and ^ are legal only in math expressions or quoted strings.

Block Data: This binary data format is in the form:

#A,length byte 1, length byte 2, data bytes

(See Chapter 11 on data formats).

All other symbols are illegal and result in an error.

System Commands

This section of the chapter describes several commands which control mainframe operation. With one exception (the **RESET** command), these commands do not alter the state of any plug-in module.

Reset Commands

Two commands return your HP 3235 and extenders to known states. Carefully read the descriptions before using the commands as they perform different functions.

CLR

The **CLR** command returns the HP 3235 to its *READY* (or idle) state and is the same as the HP-IB CLEAR command (see Appendix D). The *READY* states is defined in Chapter 2 of this manual. The **CLR** does not reset the hardware (mainframe, extenders, and plug-in modules) configuration.

To execute the **CLR** command:

```
OUTPUT 709; "CLR"
```

RESET

The command **RESET** (optionally **RST**) resets the mainframe to its power-on state except that it does not perform the self-test or delete variables and subroutines. It also takes the following actions:

- Clears the HP-IB Input and Output Buffers.
- Clears the Error Register.
- Clears all status register bits.
- Turns off the **KEYS** and **MEM** modes.
- Aborts any running subroutines.
- Turns on the control panel display (if it was off) and shows the *READY* message.
- Resets the mainframe and extenders to their turn-on state.
- Resets each plug-in module to its turn-on state (see Chapter 2).

To execute the **RESET** command:

```
OUTPUT 709; "RESET"
```

You can also reset an individual plug-in module or a group of modules with the **RESET** command. Simply specify the slot address of the modules (slot addressing is discussed later in this chapter). The following example resets only the mainframe modules in slots 100 and 300 through 600, and the module in extender number 2, slot 200.

```
OUTPUT 709; "RESET 100,300-600,2200"
```

Programming Hint

A good programming practice is to begin your programs with an HP-IB Selected Device Clear (CLEAR 709; see Appendix D) followed by the **RESET** *command. These commands put your mainframe and extenders in a known state with input and output buffers cleared. This eliminates errors caused by a previous program leaving the instrument in an unknown state.*

Resetting Modules

You can use the **CRESET** command (or the **RESET** command with a slot parameter) to reset one or more plug-in modules. These commands reset each module listed without affecting other modules or the mainframe/extender. For example, to reset only the module in mainframe slot 700, send:

```
OUTPUT 709; "CRESET 700"
```

or

```
OUTPUT 709; "RESET 700"
```

NOTE

If you do not specify any slots, the **RESET** *command resets the entire HP 3235 system (mainframe, extenders, and all plug-in modules).*

You can abbreviate the **RESET** command with **RST**. For example, the following statement also resets the module in mainframe slot 700:

```
OUTPUT 709; "RST 700"
```

To reset the modules in mainframe slots 100 through 400, send:

```
OUTPUT 709; "CRESET 100-400"
```

or

```
OUTPUT 709; "RESET 100-400"
```

or

```
OUTPUT 709; "RST 100-400"
```

Table 4-1 shows the reset states for all plug-in modules.

Table 4-1. Plug-In Module Reset States

Plug-In Module	Reset State
HP 34501 Armature Relay Module	All Relays Open.
HP 34502 Reed Relay Module	All Relays Open.
HP 34503 General Purpose Relay Module	All Relays Open.
HP 34504 Coaxial Multiplexer Module	All Relays Open.
HP 34505 50 Ohm RF Multiplexer Module	All Relays Open.
HP 34506 Coaxial Matrix Module	All Relays Open.
HP 34507 Mercury-Wetted Relay Module	All Relays Open.
HP 34508 75 Ohm RF Multiplexer Module	All Relays Open.
HP 34509 Relay Driver Module	All Relays Open, Non-Latching Mode Selected.
HP 34510 Power Actuator Module	All Relays in Normally Closed Position.
HP 34511 64-Channel Relay Module	All Relays Open.
HP 34520 Multimeter Module	DC Volts, Automatic Trigger, Autorange, 6-1/2 digit Measurements.
HP 34521 AC/DC Source Module	DC Volts, 0 VDC Output, High-Resolution Mode, 0 Ohm Output Impedance, Autorange.
HP 34522 Digital I/O Module	Four 8-Bit Input Ports.
HP 34523 Breadboard Module	User-Defined.
HP 34524 D/A Converter Module	Voltage Mode, 0 VDC Output
Extender Bus Relays (discussed later in this chapter)	All Extender Bus Relays Closed.

Identification

Three commands provide identification information from your HP 3235 and help verify HP-IB operation. These commands are:

- **ID?** -- With no parameter, **ID?** returns the model number of your instrument (i.e., *"HP3235"*). With a slot number parameter, it returns the model number and name of the plug-in module installed in that slot. If you do not have a module in the specified slot, the HP 3235 returns the string: *00000 Empty Slot.* The following program demonstrates the **ID?** command.

```
10  OUTPUT 709; "ID?"          ! SENDS ID QUERY
20  ENTER 709; A$              ! ENTER THE RETURNED STRING
30  PRINT A$                   ! PRINT THE STRING: HP3235
40  END
```

Table 4-2 shows the strings returned by the **ID?** *slot#* command.

Table 4-2. ID? Strings

Module	String Returned
HP 34501	34501 Armature Relay Multiplexer
HP 34502	34502 Reed Relay Multiplexer
HP 34503	34503 General Purpose Relay
HP 34504	34504 Switched-Shield Coaxial Multiplexer
HP 34505	34505 RF Multiplexer
HP 34506	34506 Switched-Shield Coaxial Matrix
HP 34507	34507 Mercury-Wetted Multiplexer
HP 34508	34508 RF 75 Ohm Multiplexer
HP 34509	34509 Relay Driver
HP 34510	34510 10 Amp Switch
HP 34511	34511 64 Channel Multiplexer
HP 34520	34520 Multimeter
HP 34521	34521 Source
HP 34522	34522 Digital I/O
HP 34523	34523 Breadboard*
HP 34524	34524 Quad DAC

*This string is returned by the HP 34523 Breadboard Module regardless of the settings of the user jumpers on that module.

The following program uses **ID?** *slot#* to query each of the ten mainframe slots and return the plug-in module model number for each slot.

```
10  DIM A$[255]
20  FOR I=0 TO 900 STEP 100         ! LOOP COUNTER
30  OUTPUT 709; "ID? ";I            ! QUERY EACH SLOT
40  ENTER 709; A$                   ! ENTER THE MODULE NUMBER
50  PRINT "SLOT # ";I;" HAS PLUG-IN MODULE TYPE: ";A$
60  NEXT I                          ! INCREMENT LOOP COUNTER
70  END
```

- IDN? -- **IDN?** returns four ASCII strings. The first string returned contains the words *HEWLETT PACKARD*. The second string is the model number *3235*. The third string is a zero (*0*). The fourth string is a four-digit firmware revision code number (e.g., *2506*). The following program demonstrates the **IDN?** command.

```
10  OUTPUT 709; "IDN?"          ! EXECUTE IDN?
20  FOR I=1 TO 4                ! SET LOOP COUNTER
30  ENTER 709; A$               ! ENTER ONE STRING AT A TIME
40  PRINT A$                    ! PRINT ONE STRING
50  NEXT I                      ! INCREMENT COUNTER
60  END
```

- CTYPE? -- **CTYPE?** (or **CTYPE**) returns the last digit(s) of the model number of the plug-in module in the specified slot. If the slot does not contain a module, the HP 3235 returns the number *0*.

The following program returns and displays the last digit(s) of the module's model number in mainframe slot 300.

```
10  OUTPUT 709; "CTYPE? 300"
20  ENTER 709;A
30  PRINT A
40  END
```

Table 4-3 lists the **CTYPE?** responses and the corresponding module types.

Table 4-3. CTYPE? Responses

CTYPE? Response	Description
0	No module installed in specified slot
1	HP 34501 32-Channel Armature Relay Module
2	HP 34502 32-Channel Reed Relay Module
3	HP 34503 GP Relay Multiplexer Module
4	HP 34504 Coaxial Multiplexer Module
5	HP 34505 50 Ohm RF Multiplexer Module
6	HP 34506 Coaxial Matrix Module
7	HP 34507 32-Channel Mercury-Wetted Relay Module
8	HP 34508 75 Ohm RF Multiplexer Module
9	HP 34509 Relay Driver Module
10	HP 34510 Power Actuator Module
11	HP 34511 64-Channel Armature Relay Module
20	HP 34520 Multimeter Module
21	HP 34521 AC/DC Source Module
22	HP 34522 Digital I/O Module
24	HP 34524 D/A Converter Module
*48-63	HP 34523 Breadboard Module

*The HP 34523 Breadboard Module returns an integer which is determined by the settings of the user jumpers on that module.

Echo a String

The **ECHO** command causes the HP 3235 to "echo" an ASCII string back to your system controller. Use this command to help verify HP-IB operation. For example:

```
10  OUTPUT 709; "ECHO 'THIS IS A TEST'"     ! ECHO THE STRING
20  ENTER 709; A$                           ! ENTER THE ECHOED STRING
30  PRINT A$                                ! PRINT THE ECHOED STRING
40  END
```

The controller prints the string: "THIS IS A TEST".

Module Addressing

Before sending a plug-in module command, you must specify an address to designate which module will receive the command and, if applicable, which relay or channel will respond to the command. The address is is of the form *esnn*, where *e* is the mainframe or extender number (0 = mainframe, 1 - 7 = extenders), *s* is the slot number (0 - 9), and *nn* is the relay or channel number.

The way you specify an address depends on the type of command you are using. Plug-in module commands consist of three types: (1) commands that address a slot, (2) commands that address a relay or channel, and (3) commands that require a predefined address.

Commands That Address a Slot

Commands such as **CTYPE** (module type) and **CRESET** (module reset) apply to a module as a whole and do not reference a particular relay or channel. Commands of this type contain a *slot#* or *slot_list* parameter in their syntax statement. When using this type of command, you specify only the *e* and *s* digits and set each of the *nn* digits to 0. For commands having a *slot#* parameter you can specify only one slot per command. For example, to direct the **CTYPE** command to the module in mainframe slot 100, send:

```
OUTPUT 709; "CTYPE 0100"
```

For convenience, when using a slot in the mainframe, you can omit the *e* digit. For example, the following statement also directs the **CTYPE** command to the module in mainframe slot 100:

```
OUTPUT 709; "CTYPE 100"
```

NOTE

Most examples in this manual assume the plug-in modules are installed in the mainframe, therefore the leading zero of each address is omitted.

For commands using a *slot_list* parameter you can specify a single slot (*es00*), multiple slots (*es00,es00, ...*), sequential slots (*es00-es00*), groups of sequential slots (*es00-es00,es00-es00*), or any combination of slots and groups. For example, to direct the **CRESET** command to the plug-in modules in slots 100, 300, 400, 500, and 700 of the mainframe and slot 800 of extender 1, send:

```
OUTPUT 709; "CRESET 100,300-500,700,1800"
```

Commands That Address a Relay or Channel

Commands such as **CLOSE** and **OPEN** apply to individual relays or channels on a plug-in module. Commands of this type contain a *relay#*, *relay_list*, *ch#*, *ch_list*, or *connector#* parameter in their syntax statement. When using one of these commands, you specify the complete four digit address (you can omit the leading zeros when addressing a module in the mainframe). Refer to the appropriate plug-in module manual for a particular module's relay or channel numbers.

For commands having a *relay#* or *ch#* parameter you can specify only one relay or channel per command. For example, to query the state of channel 12 on the module in mainframe slot 200, send:

```
OUTPUT 709; "CLOSE? 212"
```

For commands using a *relay_list* or *ch_list* parameter, you can specify singles (*esnn*), multiples (*esnn,esnn, ...*), sequentials (*esnn-esnn*), groups of sequentials (*esnn-esnn,esnn-esnn*), or any combination of these. For example, to close relays 2 through 5 on the module in mainframe slot 100 and relays 1 and 3 on the module in mainframe slot 200, send:

```
OUTPUT 709; "CLOSE 102-105,201,203"
```

Commands Requiring a Predefined Address

Commands that are specifically for the HP 34520 Multimeter Module, HP 34521 AC/DC Source Module, HP 34522 Digital I/O Module, or HP 34524 D/A Converter Module do not specify an address in their command statements. These commands do not have any kind of *slot*, *relay*, *channel*, or *connector* parameter in their command syntax. Before sending one of these commands, send the **USE** command to designate which module (*use device*) will receive commands. After designating a *use device*, all subsequent multimeter, source, digital I/O, or D/A converter commands are sent to that slot. The *use device* is remembered by the mainframe and remains active until another **USE** command is executed or power is cycled.

Whenever power is applied to the mainframe, it searches its slots and the extender slots for installed HP 34520 Multimeter Modules. If only one multimeter module is found, the mainframe designates that module as the *use device*. If more than one multimeter module is found, the mainframe designates the module in the lowest numbered slot as the *use device*. If no multimeter module is found at power-on, the mainframe does not designate a *use device*. The power-on *use device* remains in effect until you change it by executing another **USE** command.

Designating a Multimeter or Source Module

When designating the HP 34520 Multimeter or the HP 34521 Source Module as the *use device*, you specify the *e* and *s* digits and set each of the *nn* digits to 0. For example, to designate the multimeter or source module in slot 800 of the mainframe, send:

```
OUTPUT 709; "USE 800"
```

NOTE

The HP 34520 Multimeter Module occupies two slots and is always addressed using the lowest of the two slot numbers.

Designating a Digital I/O Port

When designating the HP 34522 Digital I/O Module as the *use device*, you specify the *e* and *s* digits and specify one of four ports with the *nn* digits (*nn* = 00, 10, 20, or 30). For example, to designate port 10 of the digital I/O module in slot 700 of the mainframe, send:

```
OUTPUT 709; "USE 710"
```

Designating a D/A Converter Channel

When designating the HP 34524 D/A Converter Module as the *use device*, you specify the *e* and *s* digits and specify one of four channels with the *nn* digits (*nn* = 00, 01, 02, or 03). For example, to designate channel 02 of the D/A converter module in slot 300 of the mainframe, send:

```
OUTPUT 709; "USE 302"
```

NOTE

*You can use the optional control panel to specify a different **use device** than that specified from the controller (over the HP-IB). If you do this, commands sent from the control panel go to the device that was specified from the control panel and commands sent from the controller go to the device specified by the controller. This is useful when an operator is interactively controlling one of the **use devices** from the front panel.*

Verifying Destinations

The **USE?** query command returns the mainframe or extender slot number presently designated by the **USE** command. For example, the following program reads and displays the designated slot number.

```
10  OUTPUT 709; "USE?"
20  ENTER 709; A
30  PRINT "Use Device Is "; A
40  END
```

If an HP 34520 Multimeter Module is not installed when power is applied to HP 3235, the **USE?** command returns "-1".

Error Handling

Errors may occur when the mainframe processor interprets and executes commands. Typical causes for errors include: command syntax errors, execution errors, math errors, signal overload errors, and hardware errors. The optional control panel display shows all errors. Refer to Appendix C in this manual for a list of error numbers and descriptions.

Error Detection

When an error occurs, the control panel display shows an error number, a brief description, and, if the error was a command syntax error, the offending string. The control panel shows the last error that occurred.

The HP 3235 processor stores errors in a first-in-first-out list which holds the first four errors that occur. You can read the list, one error message at a time, using either the **ERR?** or **ERRSTR?** commands. Reading the list empties it. You can clear the entire list with the **CLR**, **RESET** or HP-IB **CLEAR** commands.

The **ERR?** command returns the numeric code of the first error to occur. The **ERRSTR?** command returns the error number, a comma, and the error message string enclosed in quotes. You must re-execute either **ERR?** or **ERRSTR?** to get the next error code. When no errors are left in the list, the **ERR?** command returns a value of zero. The **ERRSTR?** command returns: *0, "NO ERROR"*.

Any error sets the error bit (bit 5) in the status register (see Chapters 10 and 11). The processor clears the error bit at power-on or after you read the entire error list. An HP-IB Serial Poll does not clear the error bit. Refer to Chapter 10 "Interrupts" for information and examples on SRQ interrupts when errors occur.

The following paragraphs describe how the HP 3235 detects and reports errors.

Syntax Errors

Syntax errors occur when the HP 3235 receives an illegal command. These commands may be from either the system controller or the control panel keyboard. Typical errors are: illegal characters in the command, misspelled command headers, illegal number formats, and missing or extra parameters. When a syntax error occurs, the HP 3235 processor skips characters in the command string until it reaches a command terminator (semicolon, *cr*, *lf*, or EOI). The processor then resumes execution of following commands. If the error occurs in a subroutine statement (while the controller is downloading a subroutine statement to the HP 3235), the processor ignores the command, logs the error, but stores the rest of the subroutine. If a mismatch error occurs in a subroutine (e.g., a **FOR...NEXT** loop mismatch), then the subroutine is not stored.

The HP 3235 returns the offending command string up to and including the offending character(s). It also attempts to give an explanation of why the error occurred (extra parameter, parameter out of range, etc.). In most cases error messages also give indications for valid parameters.

Execution Errors

Execution errors occur when you attempt to execute a syntactically correct command which includes a parameter range error, a prohibited relay, invalid configurations, etc. When the HP 3235 processor detects an execution error, it aborts the command and continues with the next command. If the error occurs during a subroutine execution, the HP 3235 logs the error and aborts the subroutine.

Math Errors

Math errors include dividing by zero, trig functions out of range, etc. When the HP 3235 processor detects a math error, it aborts the command and continues with the next command. Math errors in subroutines are logged and abort the subroutine.

Hardware Errors

The control panel display indicates any hardware errors that occur. If the mainframe does not power-up correctly, refer to the HP 3235 Service Manual.

Reading the Error Register

The following example shows how to read the error register from the system controller.

```
10  DIM Message$[255]
20  REPEAT
30    OUTPUT 709; "ERRSTR?"
40    ENTER 709; Code,Message$
50    PRINT Code,Message$
60  UNTIL Code = 0
70  END
```

If no errors exist, the **ERR?** command returns *0* and the **ERRSTR?** command returns *0, "NO ERROR"*.

Using the HELP Function

After becoming familiar with the HP 3235 and its command set, you will use the manuals less frequently. However, if you need information on a particular command's structure or some term used in a command, the **HELP** function is available to assist you.

The **HELP** function is similar to a built-in quick reference guide. It provides a complete syntax statement and definition for all of the HP 3235 commands. It also provides explanations for topics other than commands, such as definitions of terms used in commands and glossary items.

To use the **HELP** function, simply execute the **HELP** command followed by a one-word topic. You can execute the **HELP** commands from either the optional control panel or the system controller. For a list of general topics, execute the command **HELP HELP** (or **HELP TOPIC**). The following program demonstrates how to use the **HELP** function from your system controller.

```
10  DIM A$[255]
20  OUTPUT 709; "HELP HELP"
30  ENTER 709; A$
40  PRINT A$
50  END
```

The HP 3235 returns the string:

```
HELP: HELP <topic> ; HELP is available for the following topics:
SYSTEM, HP-IB, PANEL, TIMERS, TRIGBUS, SWITCHING, METER, SOURCE,
DIGIO, MODULES, SCANNING, SUBS, MATH, HPIB_CMDS, CIIL_CMDS, TESTING,
and INTERRUPTS.
```

The **HELP** function provides more information on the listed topics. Let's consider an example. You want to connect channel 213 to analog bus 0 but you can't remember the exact command to close the appropriate relays. Execute the command **HELP SWITCHING** (either from the optional control panel or from your system controller) to get information. The HP 3235 returns the message:

```
SWITCHING: HELP is available for: OPEN, CLOSE, CLOSE?, SELECT,
CONNECT, DISCONN, PROHIBIT, ALLOW, PROHIBIT?, PFOPEN, PFSAME, PFCLOSE,
and PFCLOSE?
```

The **CONNECT** command automatically connects the channels but you need to know its syntax. Execute the **HELP CONNECT** command. The HP 3235 returns the string:

```
CONNECT: CONNECT [ONLY] {<connector> <connector> | <chan> ABn} ;
Closes the relays necessary to form a path between two connectors or
between a channel and an analog bus.  Ex: CONNECT 401,413.
```

This shows the **CONNECT** command syntax and a short definition of the command. Note that the parameter *connector* is enclosed in brackets (< >). The brackets mean that additional information is available to define *connector*. Execute the command **HELP CONNECTOR**. The HP 3235 returns the string:

```
CONNECTOR: A number of the form ESRC; E = extender, S = slot, R = row,
C = column.  Ex: 524 = mainframe, slot 500, row 20, column 4.
```

This string defines the *connector#* parameter.

NOTE

*Some command keywords (**SET CLKSRC**, **SET OUTPUT**, etc.) and some command parameters (**ch_list**, **slot_list**, etc.) contain two words. When using the help function with these commands or parameters, use the underscore character ("_") between the two words. For example, execute:*

```
OUTPUT 709; "HELP SET_CLKSRC"
```

Chapter 5
Switching

Introduction

Three operations apply to the switching plug-in modules:

- Operating Switches
- Prohibiting Closures
- Setting Power Failure States

This chapter describes how to perform each of these three operations. It also discusses sequential operation for switching commands. For more information about the individual commands referenced in this chapter, refer to the Language Reference Manual.

Chapter 4 in this manual describes general relay addressing. For the specific addressing scheme used in a particular module, refer to the individual plug-in module Programming and Configuration manuals.

Channel Definitions

The term *channel* refers to any relay or group of relays on a plug-in module that can switch a signal from one user connector to another user connector. For example, Figure 5-1 shows a simplified schematic for the 32-channel relay multiplexer modules. Note that a single relay closes the circuit between a channel connector and the common connector.

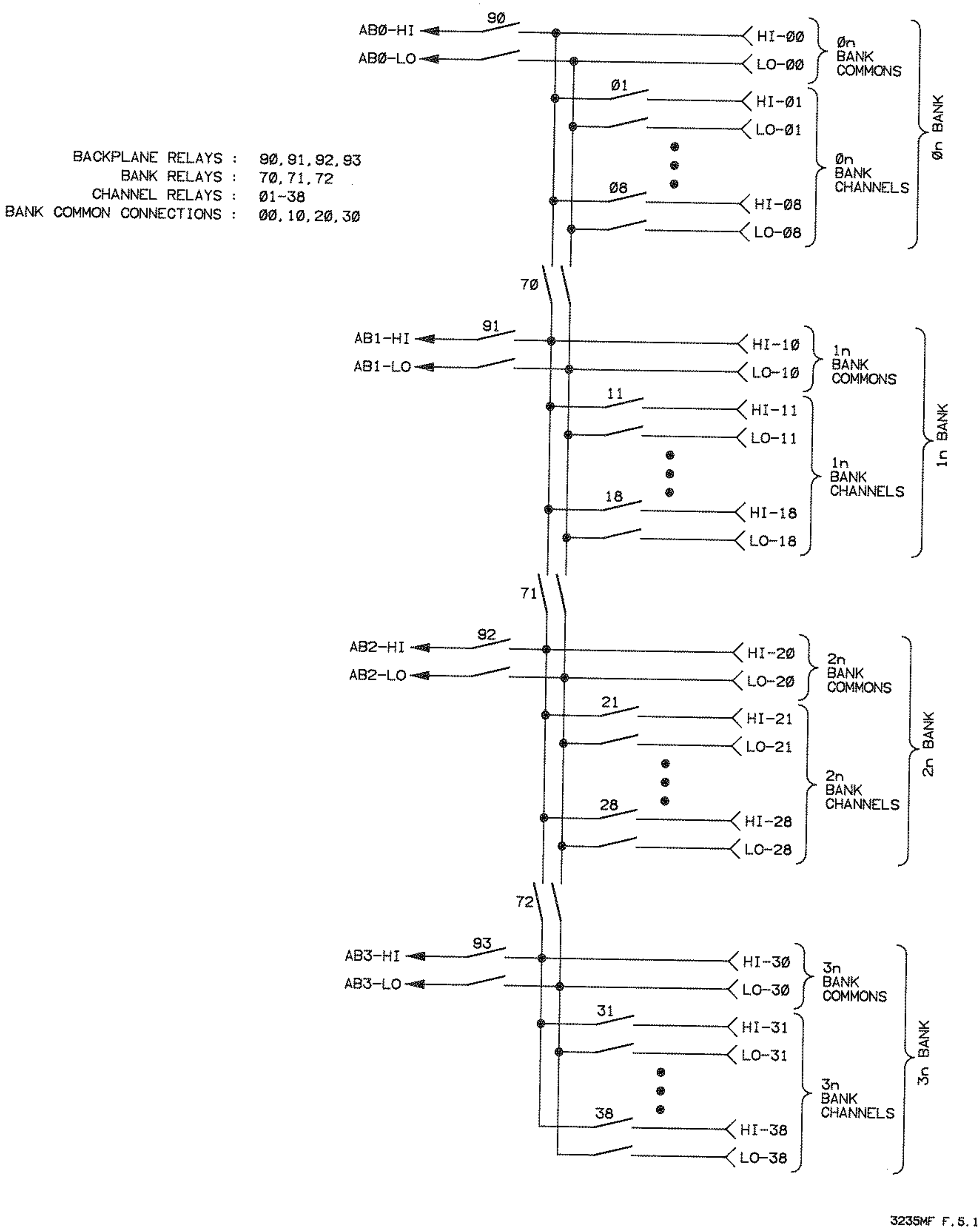


Figure 5-1. 32-Channel Multiplexer Simplified Schematic

On other modules, each channel can consist of several relays. The HP 34505 RF Multiplexer Module, for example, closes three relays to connect a user channel to the common. Refer to Figure 5-2.

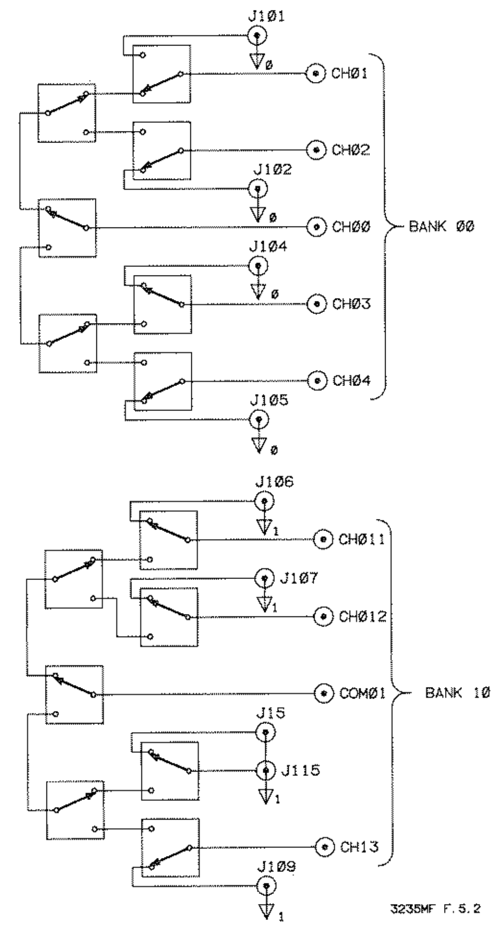


Figure 5-2. HP 34505 RF Multiplexer Simplified Schematic

Relay Definition

In addition to channel relays, some relay plug-in modules contain *backplane relays*, *bank relays*, and *trigger buffer relays*. Refer to the individual plug-in module Programming and Configuration manuals for more information on a particular module's relays.

Bank Definition

The term *bank* refers to any group of channels that are able to switch signals to a common point. For example, in Figure 5-2, bank 00 consists of channels 01 through 04 since each of these channels can switch a signal to a common point (common 00).

Operating Relays and Channels

For non-scanning applications, switches are controlled using the **CLOSE**, **OPEN**, **SELECT**, **CONNECT**, and **DISCONN** commands.

NOTE

When more than one channel number is specified in a command (i.e., **CLOSE 102,104,205,607***) the channels may not open or close in the sequence specified. This is because some relay types are faster than others. The HP 34502 reed relays and 34507 mercury-wetted relays are significantly faster than the HP 34501 armature relays.*

Closing Relays or Channels

You can use the **CLOSE** command to close one or more relays or channels (assuming the relays or channels are not prohibited; refer to "Prohibiting Closures" later in this chapter). For example, to close channels 03 and 12 on the switch module in mainframe slot 200, send:

```
OUTPUT 709;"CLOSE 203,212"
```

Opening Relays and Channels

The **OPEN** command allows you to open one or more relays or channels. For example, to open the previously closed channels, send:

```
OUTPUT 709;"OPEN 203,212"
```

Selecting Channels

The **SELECT** command ensures break-before-make channel operation. **SELECT** applies only to channels, it cannot operate a non-channel relay. The **SELECT** command first opens all channels in the same bank(s) as the specified channel(s), then (after the channel(s) has settled) it closes the specified channel(s). For example, assume the module in slot 200 consists of 2 banks containing 16 channels each (bank 0 = channels 01-16, bank 1 = channels 17-32). To open all channels in bank 0 and then close channel 03, send:

```
OUTPUT 709;"SELECT 203"
```

With the **SELECT** command, those channels in the same bank as the specified channel open before the channel closes. For example, the following command statement opens all channels in both banks, and then closes channels 02 and 20:

```
OUTPUT 709;"SELECT 202,220"
```

Refer to the individual plug-in module Programming and Configuration manuals for specific information on how to use the **CLOSE**, **OPEN**, and **SELECT** commands with a particular module.

Verifying Relay or Channel States

You can use the **CLOSE?** command to verify the state of a particular relay or channel. The **CLOSE?** command returns a "0" if the relay or channel is open or a "1" if it is closed. For example, the following program verifies the state of channel 02 on the module in mainframe slot 200:

```
10  OUTPUT 709; "CLOSE? 202"
20  ENTER 709; A
30  PRINT "Switch State (0=Open/1=Closed) = ";A
40  END
```

CAUTION

The HP 3235 "remembers" the state of each relay in its memory. When the HP 3235 receives a **CLOSE?** *command it searches its memory to determine if it received a* **CLOSE** *or* **OPEN** *command for that relay or channel. Therefore, it is possible that the command would show a relay or channel closed but because of relay or other hardware failure, the relay or channel is still open.*

Connecting Terminals

The **CONNECT** command works only with the HP 34501, 34502, and 34507 32-Channel Relay Multiplexer modules or the HP 34506 Coaxial Matrix Module. However, its parameters are different for the two module types.

The **CONNECT** command verifies that a viable path exists between the two connector numbers and then closes the appropriate relay(s). If a connection path does not exist, an error is generated.

When used with the 32-channel relay multiplexers, the **CONNECT** command has the form:

CONNECT [ONLY,] ***ch#*****, AB*****n***

The channel (*ch#*) parameter and the analog bus parameter (*ABn*) may be specified in either order. The *ABn* parameter may also be a backplane relay number (i.e., es9n; for example, **CONNECT 101,190**). The **CONNECT** command closes all relays necessary to connect the specified connector to the specified analog bus. Thus **CONNECT** closes bank relays as well as backplane relays. You cannot use **CONNECT** to directly connect a connector on one plug-in module to a connector on another module.

The following example connects channel 11 in mainframe slot 100 to analog bus 0:

```
OUTPUT 709; "CONNECT 111,AB0"
```

When using the **CONNECT** command with the coaxial matrix module, you specify the connector numbers instead of the channel and analog bus. **CONNECT** has the form:

CONNECT [ONLY,] ***connector#, connector#***

The connector numbers specify the row and column to connect in the matrix.

You can specify the row and column in either order (i.e., *row, column* or *column, row*).

The following two examples connect column 01 to row 20 on the coaxial matrix module installed in slot 200 of the mainframe.

```
OUTPUT 709; "CONNECT 201,220"
```

```
OUTPUT 709; "CONNECT 220,201"
```

The ONLY Parameter

The **ONLY** parameter to the **CONNECT** command ensures that all relays on all banks on the module specified are open before closing the appropriate relays. The following example opens all relays on the module in slot 100 before connecting channel 11 to analog bus 0:

```
OUTPUT 709; "CONNECT ONLY,111,AB0"
```

Opening the Connection

The connections made with the **CONNECT** command may be opened using the **OPEN** command to open individual relays, the **RESET** command to open all relays, or the **DISCONN** (disconnect) command to open the specified path. The **DISCONN** command opens the path between the specified connector and the specified analog bus (for the 32-channel relay multiplexers) or the matrix connection terminals (for the coaxial matrix module). **DISCONN** has the same format as the **CONNECT** command. Thus, to disconnect row 01 from column 20 on the coaxial matrix module in slot 200, execute:

```
OUTPUT 709; "DISCONN 201,220"
```

Prohibiting and Allowing Closures

The **PROHIBIT** command allows you to prohibit closures of specified plug-in module switches. This can prevent accidental closures that could cause test system failures or equipment damage. The **ALLOW** command cancels the prohibited state for one or more relays.

To ensure the proper operation of **PROHIBIT**, prohibited channels should be specified before any channels are closed or before storing the state of any switch modules. **PROHIBIT** can be overridden by recalling a state that contains a prohibited closure. See Chapter 11 in this manual for information on storing and recalling module states. Execution of a **PROHIBIT** command does not open any relay or channel already closed.

Prohibiting Closures

Prohibited closures are stored in continuous memory (remembered by the HP 3235 when power is removed). In the event of power failure, any prohibited closures will once again be prohibited when power is re-applied. If you try to close a prohibited relay or channel, it remains open and the *PROHIBITED SWITCH* error occurs.

WARNING

Do not use the **PROHIBIT** *command to prevent operating personnel from contacting dangerous voltages.* **PROHIBIT** *is not fail-safe and can be overridden if a stored state is recalled (see* **RSTATE** *command in Chapter 11).*

When prohibiting closures you have the choice of prohibiting any closures, two closures, or all closures.

Prohibiting Any Closures

The **ANYOF** parameter prevents closures on *any* relays or channels in the specified list. For example, to open channels 03 and 06 on the module in mainframe slot 100, and switch 12 on the module in slot 200, and then prohibit them from closing, send:

```
10 OUTPUT 709;"OPEN 103,106,212"
20 OUTPUT 709;"PROHIBIT ANYOF,103,106,212"
30 END
```

Prohibiting Two Closures

The **TWOOF** parameter prevents closures on any *two* relays or channels in the specified list. However, closing a single relay or channel in the list is not prohibited. For example, the following program opens channels 01, 10, 11, and 13 on the module in mainframe slot 100, and then prevents any two of them from closing:

```
10 OUTPUT 709;"OPEN 101,110,111,113"
20 OUTPUT 709;"PROHIBIT TWOOF,101,110,111,113"
30 END
```

You can still close one of these channels, for example:

```
OUTPUT 709;"CLOSE 101"
```

Now, as long as one of the channels is closed, you cannot close any of the other three channels.

Prohibiting All Closures

The **ALLOF** parameter prevents closures of *all* the relays or channels specified in the list. However, you can still close up to all but one relay or channel in the list. For example, the following program opens all channels on the module in mainframe slot 100, and then prevents channel 101 from connecting to analog bus 1:

```
10 OUTPUT 709;"CRESET 100"
20 OUTPUT 709; "PROHIBIT ALLOF,101,170,191"
30 END
```

You can close any two of the channels, but not all three.

Multiple Prohibit Modes

You can have more than one prohibit mode in effect at one time. For example, after running the following program, ALLOF channels 111-116, and TWOOF channels 114-124 are prohibited (notice that channels 114, 115, and 116 are in both lists):

```
10 OUTPUT 709; "CRESET 100"
20 OUTPUT 709; "PROHIBIT ALLOF,111-116"
30 OUTPUT 709; "PROHIBIT TWOOF,114-124"
40 END
```

You cannot, however, have two prohibit modes in effect that specify the same relay or channel list in exactly the same format. For example, the following statement specifies a different mode but uses the same list as line 30 of the above program:

```
OUTPUT 709; "PROHIBIT ALLOF,114-124"
```

In this case, the latter command (**PROHIBIT ALLOF**) overwrites the previous command which means the prohibit mode specified by line 30 is no longer in effect.

Verifying Prohibited Closures

The **PROHIBIT?** command returns information on prohibited closures. **PROHIBIT?** returns the prohibit type (**ANYOF**, **TWOOF**, or **ALLOF**) followed by up to ten relay or channel numbers per line. The numbers are returned in the order they were prohibited. The last line returned is the word *DONE*. For example, the following program prohibits **ALLOF** the channels 101, 106, 304, 506 and **TWOOF** the channels 202, 203, and 204. It then sends the **PROHIBIT?** command and cycles through the enter/print loop until all prohibited type(s) and channels are printed:

```
10 DIM A$[60]
20 OUTPUT 709;"PROHIBIT ALLOF 101,106,304,506"
30 OUTPUT 709;"PROHIBIT TWOOF 202,203,204"
40 OUTPUT 709;"PROHIBIT?"
50 REPEAT
60 ENTER 709;A$
```

```
70 PRINT A$
80 UNTIL A$="DONE"
90 END
```

The system controller displays the following:

```
ALLOF     101,  106,  304,  506
TWOOF     202,  203,  204
DONE
```

NOTE

If you are using the optional control panel, use the **Single Step** key to get the next line of display.

Allowing Closures

The **ALLOW** command permits you to close channel or channel combinations that were previously prohibited. As with the **PROHIBIT** command, the **ALLOW** command gives you the choice of any closures, two closures, or all closures.

NOTE

The **ALLOW** *command must use the same syntax (both combination and relay list parameters) as the* **PROHIBIT** *command it cancels. For example, if you execute:*

```
OUTPUT 709; "PROHIBIT ALLOF, 313,314,315,317"
```

you must send the following command to cancel all of the prohibited relays or channels.

```
OUTPUT 709; "ALLOW ALLOF, 313,314,315,317"
```

The command:

```
OUTPUT 709; "ALLOW ALLOF 313-317"
```

results in an error and the four relays remain prohibited.

Allowing Any Closures

The **ALLOW ANYOF** command enables closure of relays or channels previously prohibited with the **PROHIBIT ANYOF** command. You must use the same combination parameter (in this case, **ANYOF**) and the same channel numbers. For example:

```
10 OUTPUT 709; "PROHIBIT ANYOF, 103,106,212"
  .
  .
  .
100 OUTPUT 709;"ALLOW ANYOF,103,106,212"
```

Allowing Two Closures

Use the **ALLOW TWOOF** command to enable closure of relays or channels previously disabled with the **PROHIBIT TWOOF** command. The **ALLOW TWOOF** command must use the same parameters previously used in the **PROHIBIT TWOOF** command:

```
OUTPUT 709; "PROHIBIT TWOOF,101,110,111,113"
  .
  .
  .
100 OUTPUT 709;"ALLOW TWOOF,101,110,111,113"
```

Allowing All Closures

Use the **ALLOW ALLOF** command to enable closure of relays or channels previously disabled with the **PROHIBIT ALLOF** command. The **ALLOW ALLOF** command must use the same parameters previously used in the **PROHIBIT ALLOF** command:

```
10 OUTPUT 709; "PROHIBIT ALLOF,101-138"
  .
  .
  .
100 OUTPUT 709;"ALLOW ALLOF,101-138"
```

Canceling Prohibited Channels

If you want to cancel all the prohibited relays or channels on all modules, execute the **ALLOW** command with the **ALL** parameter. For example, the following statement cancels all previously prohibited closures:

```
OUTPUT 709;"ALLOW ALL"
```

You now have normal control over all relays and channels.

Setting Power Failure States

Normally, when the HP 3235 loses power, all relays or channels on the plug-in modules open. However, three commands allow you to specify the relay states that some modules assume when power is removed. The three commands are: **PFCLOSE**, **PFOPEN**, and **PFSAME**. After you specify power failure states for the relays or channels on a particular plug-in module, the HP 3235 sets these relays or channels to the specified states whenever it detects power failure. You can specify power failure states for the following plug-in modules only: HP 34501 Armature Relay Module, HP 34504 Coaxial Multiplexer Module, HP 34506 RF Multiplexer Module, and HP 34511 64-Channel Armature Relay Module. The power failure states are stored in HP 3235 non-volatile memory.

On the HP 34501, you can specify which relays will be opened, which will be closed, and which will retain the same state when power is removed. On the HP 34504 and 34506, you can only specify that all channels on the module be opened or left in their present state when power is removed. On these modules, you cannot specify individual channel states; the power failure commands apply to all channels on the entire module. On the HP 34511, you can specify which individual relays will be opened or left in their present state when power is removed.

Power Fail Closures

The **PFCLOSE** command specifies which relays or channels on the HP 34501 close when power fails. For example, if you want relays 201, 304, 215, and 332 to close when a power failure occurs, send:

```
OUTPUT 709; "PFCLOSE 201,304,215,332"
```

Power Fail Openings

The **PFOPEN** command specifies which relays or channels open when a power failure occurs. On the HP 34501 and 34511, you can specify individual relays. Therefore, on the HP 34501, the **PFOPEN** can be used to cancel a **PFCLOSE** command for one or more relays. On the HP 34504 and 34506, you specify only one channel to open all channels on that module when power fails. For example, to open relay 304 on the HP 34501 in mainframe slot 300 and all the channels on the HP 34504 in mainframe slot 600, execute:

```
OUTPUT 709; "PFOPEN 304,601"
```

The following program statement resets all power failure states so that all relays in the mainframe and extenders open when power is removed.

```
OUTPUT 709; "PFOPEN 0-9999"
```

No State Change at Power Fail

The **PFSAME** command specifies relays or channels which are to remain in the same state when power fails. That is, relays or channels specified in the **PFSAME** command do not specifically open or close when power fails; they remain in the same state they were in immediately prior to the power failure. For the HP 34501 and 34511, you can specify individual relays. For the HP 34504 and 34506, you just have to specify one channel and all channels on the module remain in the same state. To program channels 304 and 405 to remain in the same state when power fails, execute:

```
OUTPUT 709; "PFSAME 304,405"
```

Scanning Applications

Introduction

This chapter explains two primary methods of scanning. The two methods are (1) scanning with the HP 34520 Multimeter and (2) scanning with an external instrument (voltmeter, counter, etc.).

In addition, we'll show you how to:

- Scan and verify the results of the scan.
- Optimize for speed.
- Monitor one or more channels in the scan.
- Scan with the HP 34504, 34505, and 34508 Coaxial Modules.
- Abort a scan in progress.

The **MEAS** and **VERIFY** commands automatically scan through a list of channels using the HP 34520 Multimeter Module. The **SCAN** and **PSCAN** commands scan with external measuring instruments (voltmeters, counters, etc.) in the scan.

Why Scanning in a Test System?

In Chapter 5 you learned to control individual switches. But the HP 3235 provides a wider range of test system possibilities with channel scanning. Scanning automatically sequences through a list of channels. Normally this includes making a measurement on each channel. You could, however, sequence through a list of channels applying a test signal to each channel. Allowing the HP 3235 to control the scanning operation enhances your overall system performance by relieving the system controller of that burden.

Initial Setup

Scanning involves closing and opening relays to connect your DUT to some measuring instrument. Generally this will involve at least one of the four analog backplane buses.

CAUTION

A hazardous condition may exist if, prior to scanning, you have any user channels closed and connected to the backplane bus that will be used for scanning. Therefore, good programming practice is to open all relays on all plug-in modules prior to scanning. You can do this by executing the **RST** *or* **CRESET** *commands or by executing the command* **OPEN 0-999** *(***OPEN 0-7999** *for extenders).*

In some cases you may need to configure bank or backplane relays, and/or user channels to perform some task other than the scanning sequence. For example, you may need to apply power to the DUT or test fixture. It is good programming practice to open all relays on all plug-in modules and then configure the HP 3235 as needed. Do not close any channels that may interfere with the scan or connect to the same backplane bus that will be used for the scanning operation.

Measurement Instrument Setup

In general, you should set up the measurement instrument before beginning the scan. With the HP 34520 Multimeter for example, you might want to set the integration time (NPLC) for the desired accuracy. If all the measurements can be made from the same voltmeter range, you could manually set the range. This would improve the scanning speed. The **MEAS** command sets the measurement function (DCV, ACV, etc.) and the multimeter terminals. If you are using an external instrument, you would need to set the measurement function, terminal source, etc.

NOTE

If you are scanning with the HP 34520 Multimeter, specify the multimeter's location with the **USE** *command prior to executing* **MEAS**. *For example:*

```
OUTPUT 709; "USE 800"    ! SPECIFIES MULTIMETER IN MAINFRAME
                         ! SLOT 800
```

Scanning and Prohibited Channels

If your scan list (list of channels to be scanned) contains any prohibited channels (see the **PROHIBIT** command, Chapter 5), the HP 3235 does not close the prohibited channel(s) and generates a prohibited channel error. However, it does trigger the measuring instrument. Therefore, a false reading appears for any prohibited channel in the scan list.

Scanning With the HP 34520 Multimeter

The **MEAS** command allows you to automatically measure one or more channels. **MEAS** automatically closes appropriate bank and backplane relays for the measurement.

MEAS has the following format:

MEAS [*function*,] [**AB***n*,] [**FAST**,]*ch_list*
[**FRONT**,]

The optional *function* parameter specifies the measurement type; i.e. DCV, ACV, OHM, etc. The default is DC voltage measurements. The HP 34520 Multimeter Manual explains the various measurement functions in detail. Table 6-1 lists the possible measurement functions.

Table 6-1. Measurement Functions

Function Parameter	Definition
DCV	DC Voltage Measurement
ACV	AC Voltage Measurement
ACDCV	AC+DC Voltage Measurement
DCI	DC Current Measurement
ACI	AC Current Measurement
ACDCI	AC+DC Current Measurement
OHM	2-Wire Resistance Measurement
OHMF	4-Wire Resistance Measurement*
FREQ	Frequency Measurement
PER	Period Measurement

* See 4-wire Measurements later in this chapter.

NOTE

Current measurements (i.e. DCI, ACI, and ACDCI) can only be made through the HP 34520 Multimeter's FRONT terminals. Do not specify one of the analog backplane buses for current measurements.

The optional **AB***n* parameter specifies which analog bus is used in the scan. If you do not specify an analog bus, the **MEAS** command defaults to analog bus 0. You can also specify **FRONT** to use the multimeter's front terminals rather than the analog backplanes. To do this you must externally connect the common terminals from the multiplexer module(s) being used to the multimeter's front terminals.

The **FAST** parameter (discussed in detail later in this chapter) is used only with the HP 34502 and 34507 Relay Modules.

The *ch_list* parameter is the list of channels you want to scan. Each channel number must be in the form *esnn* where *e* represents the extender number (0 or blank for the mainframe), *s* is the slot number (0 through 9), and *nn* is the channel number. You can specify a single channel (*esnn*), multiple channels separated by commas (*esnn,esnn,esnn, ... esnn*), contiguous channels (*esnn-esnn*), or groups of contiguous channels (*esnn-esnn,esnn-esnn*). You can also specify an array (see Chapter 7 in this manual) containing the channel list.

When you specify a contiguous series of channels, the HP 3235 ignores any undefined channel numbers. For example, the HP 34501, 34502, or 34507 relay modules do not use channel numbers 09, 10, 19, 20, 29, and 30. Therefore, if you specify channels 1-38, the unused channels are ignored.

A typical **MEAS** command looks like:

```
OUTPUT 709; "MEAS DCV,201-232"
```

This command specifies a DC voltage measurement on all channels between 201 and 232 (inclusive).

NOTE

The **MEAS** *command sets the HP 34520 Multimeter Number of Readings Per Trigger (***NRDGS***) to 1 and Trigger to auto. Other multimeter parameters, such as function, range, and terminal, are either defaulted or as you set them in the* **MEAS** *or other commands prior to executing the* **MEAS** *command. If you want to specify one range for the multimeter (rather than have it autorange each measurement) use the* **FUNC** *command with a* **RANGE** *parameter. If the* **MEAS** *command changes the multimeter's measurement function, it sets the multimeter to autorange.*

How MEAS Works

The **MEAS** command performs 13 steps. We'll list these steps and then discuss them in detail.

1. Sets up the HP 34520 multimeter to the specified function. The multimeter uses all defaults for other measurement parameters (autorange on, autozero on, etc.) unless specified prior to executing the **MEAS** command.

2. If the **FAST** parameter was specified, the HP 3235 closes all the backplane relays necessary for the scan. See the discussion on **FAST** later in this chapter.

3. Waits for the specified scan trigger (**STRIG**) signal. If you did not previously specify an **STRIG** signal, **MEAS** skips this step.

4. Looks at the first channel in the channel list. Opens all relays connected to the same bank and then closes the appropriate bank and backplane relays and the first channel in the list to connect the signal to the multimeter. (Break-Before-Make action)

5. Pulses the **CHCLOSED** destination when the relays have settled.

6. Measures the signal.

7. Sends the measurement result to either the HP-IB Output buffer or the control panel display (wherever the **MEAS** command originated) or to a specified memory location (see the **MEM** command).

8. Waits for the specified **SADV** signal. If you did not previously specify an **SADV** source, **MEAS** skips this step.

9. Opens the channel, closes the appropriate relays for the next channel in the channel list (Break-Before-Make action) and pulses the **CHCLOSED** destination.

10. Repeats steps 5 through 9 for the remaining channels in the channel list.

11. After the last channel has been measured, waits for one more **SADV** pulse (if specified).

12. Opens all the backplane and channel relays associated with the last channel. Any bank relays closed during the scanning process remain closed.

13. If the **FAST** parameter was specified, opens all the backplane relays associated with the scan.

Now let's look at these steps in more detail:

Step 1. The **MEAS** command sets the multimeter module specified by the **USE** command to the specified measurement function.

You should do all other multimeter configuration (setting integration time, setting fixed measurement range, etc.) prior to executing the **MEAS** command. Otherwise, the multimeter uses its default values. If the **MEAS** command changes the multimeter function it also automatically sets autorange on regardless or any prior range command.

Step 2. If the **FAST** parameter was specified in the **MEAS** command, the HP 3235 searches through the scan list and closes all the backplane relays necessary to complete the scan. This decreases the scan time since the backplane relays are armature relays and are relatively slow to close and settle. compared to the reed relays on the HP 34502 Reed Relay Module and the HP 34507 Mercury-Wetted Relay Module. Refer to the discussion on the **FAST** parameter later in this chapter.

Step 3. The HP 3235 then waits for a Scan Trigger signal. The Scan Trigger signal initiates the scanning sequence. The power-on default source for the Scan Trigger is an automatic trigger. Thus, if you do not specify a Scan Trigger signal source, the HP 3235 uses its default source and triggers automatically. Use the **STRIG** command to specify alternate sources for the Scan Trigger signal.

Table 6-2 shows the possible the Scan Trigger sources.

Table 6-2. STRIG (Scan Trigger) Sources

Parameter	Definition
AUTO	Start on receipt of MEAS Command (Default)
EXTIN	Starts on External Trigger In (BNC) pulse
TB0	Starts when Trigger Bus 0 is pulsed
TB1	Starts when Trigger Bus 1 is pulsed
HOLD	Wait for STRIG SGL each time to start
SGL	Software Scan Trigger

The **STRIG AUTO** parameter provides automatic triggering. In general, this also provides the fastest and easiest triggering.

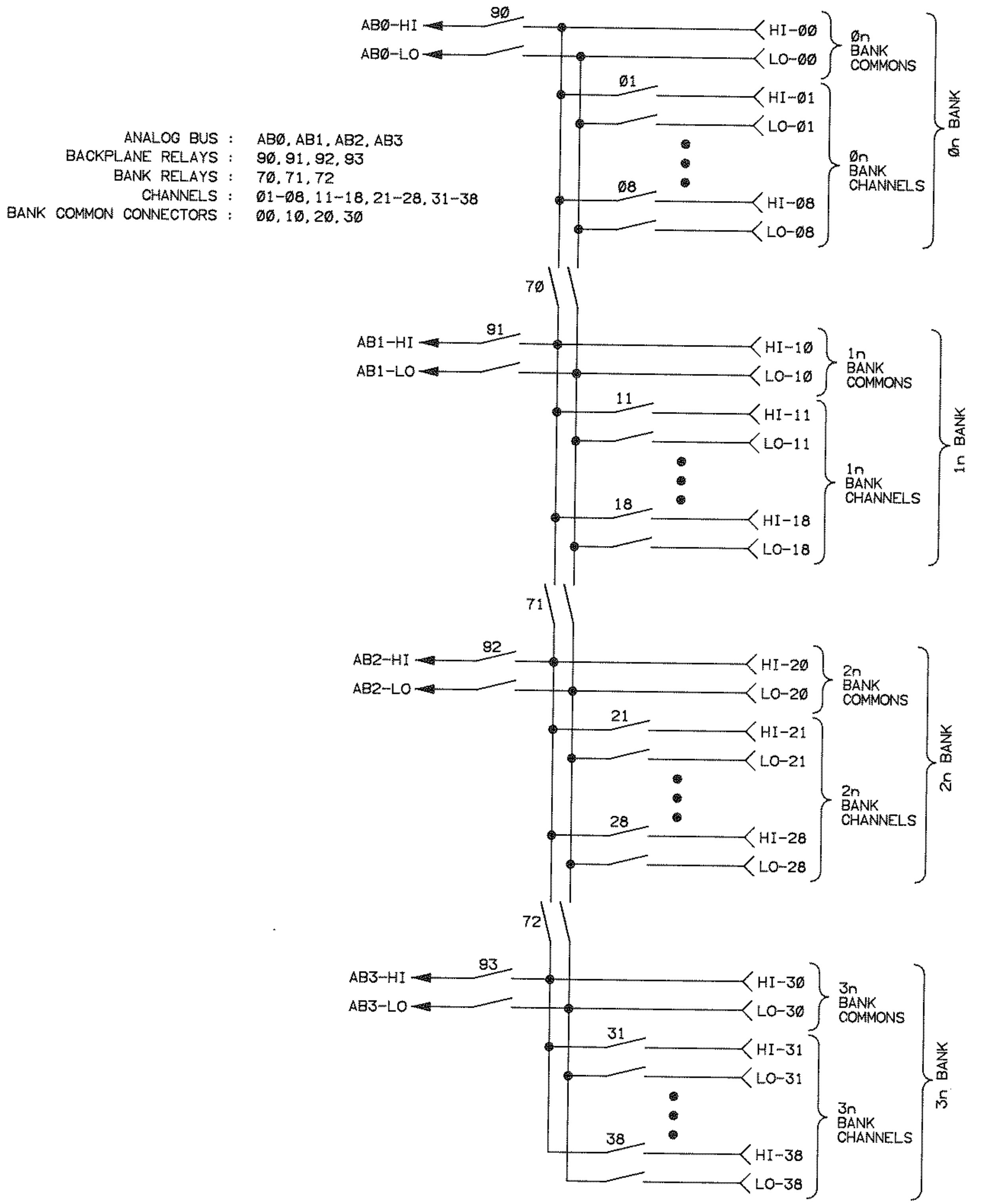
STRIG EXTIN accepts a TTL level trigger pulse from the External Trigger In BNC connector on the back panel of the HP 3235.

The two backplane Trigger Buses, TB0 and TB1, can also act as Scan Trigger sources. You can use the **DRIVETBn** command to specify the source for driving the trigger buses or you can drive the trigger buses through either the HP 34504 Coaxial Multiplexer module, the HP 34506 Coaxial Matrix module, or the HP 34522 Digital I/O module. See the **TRIGBUF** command for more information.

While **EXTIN**, **TB0**, and **TB1** provide hardware triggers, you can specify a software trigger with the **HOLD** and **SGL** parameters. Specify **STRIG HOLD** before executing the **MEAS** command. This will cause the HP 3235 to continue with the next command received and not start the scan until it receives the **STRIG SGL** command. The **STRIG SGL** command can come from either the control panel keyboard or the system controller via HP-IB. In this mode, you can set up other plug-in modules prior to executing the **STRIG SGL** command.

Step 4. Refer to Figure 6-1. In this step, the HP 3235 opens any relays already closed in the Plug-in Module Bank associated with the first channel in the channel list. For example, if you specify channel 13 in the channel list, the HP 3235 opens channels 11 through 18. The HP 3235 then closes the channel specified (in this example, channel 13) and any backplane and bank relays necessary to connect the source (channel 13) to the multimeter.

The HP 3235 looks to see what bank relays need to close to connect the channel to the backplane. All channels that are in the same bank are opened before closing the bank relays. In the example, to connect channel 13 to analog bus 0, the HP 3235 will need to close bank relay 70. This makes all channels between 1 and 18 on the same bank.



3235MF F.6.1

Figure 6-1. A Typical General Purpose Scanner Module

If you did not specify an analog bus with the **MEAS** command, the HP 3235 will default to Analog Bus 0. Thus to connect channel 13 , the HP 3235 closes channel 13, bank relay 70, and backplane relay 90. Of course, the HP 3235 also closes the appropriate terminal relay on the multimeter to connect it to the analog bus.

If, as another example, you specified Analog Bus 1 and channel 33, the HP 3235 would close channel 33, bank relays 71 and 72, and backplane relay 91.

CAUTION

The **MEAS** *command opens all the channel relays in the same bank as the current channel in the channel list. It does not open any bank relays (relay numbers 70 through 72). Therefore, to prevent possible damage to the HP 3235, the Plug-in Module, and/or any external circuitry, you should ensure that all bank relays and backplane relays not essential to your configuration, are opened prior to initiating the* **MEAS** *command. Execute the* **OPEN 0-999** *command.*

Another solution is to close bank relays 70, 71, and 72 prior to initiating the scan. With this done, the HP 3235 sees all 32 channels as one bank. When the **MEAS** *command opens all channels in the bank, all 32 channels will open. You can, of course, close any combination of bank relays on a Plug-in Module to redefine the bank size.*

Step 5. Each time a channel in the scan list closes, the HP 3235 outputs a channel closed pulse. The **CHCLOSED** command specifies the destination for this pulse. Table 6-3 lists the possible destinations for the **CHCLOSED** command:

Table 6-3. CHCLOSED Destinations

Parameter	Description
OFF	No Channel Closed Output Pulse
EXT	Pulse the External Trigger Out BNC
TB0	Pulse Trigger Bus 0 (see Chapter 9)
TB1	Pulse Trigger Bus 1 (see Chapter 9)

Step 6. Next, the multimeter makes the actual measurement specified.

Step 7. The measurement from the channel is sent wherever specified. If you executed the **MEAS** command from the control panel, the reading is displayed. If you executed **MEAS** from the system controller via HP-IB, the measurement goes to the HP-IB output buffer. If you specified a memory location (**MEM** command), such as a reading array, the measurement is stored there.

Step 8. Now, the HP 3235 waits for the Scan Advance Signal specified with the **SADV** command. The power-on default for **SADV** is an automatic scan advance. Thus, if you did not specify another **SADV** source, the HP 3235 uses the default source and automatically advances.

Table 6-4 shows the possible sources for the Scan Advance Trigger (SADV) signal.

Table 6-4. SADV (Scan Advance) Trigger Sources

Parameter	Definition
AUTO	Automatic Advance (Default)
EXTIN	Advances on External Trigger In (BNC) pulse
TB0	Advances when Trigger Bus 0 is pulsed
TB1	Advances when Trigger Bus 1 is pulsed
HOLD	Wait for SADV SGL each time to advance
SGL	Software Scan Advance Single

The **SADV AUTO** parameter provides automatic channel advance. In general, this also provides the fastest and easiest scan advance method.

SADV EXTIN accepts a TTL level trigger pulse from the External Input BNC connector on the back panel of the HP 3235.

The two backplane Trigger Buses TB0 and TB1 can also act as Scan Advance sources. You can use the **DRIVETB*n*** command to specify the source for driving the trigger buses. You can also drive the trigger buses through either the HP 34504 Coaxial Multiplexer Module or the HP 34506 Coaxial Matrix Module or from the HP 34522 Digital I/O Module. See the **TRIGBUF** command for more information.

While **EXTIN**, **TB0**, and **TB1** provide hardware triggers, you can specify a software trigger with the **HOLD** and **SGL** parameters. Specify **SADV HOLD** before executing the **MEAS** command. This causes the HP 3235 to stop the scanning sequence and accept other commands. It continues with the scanning when it receives the **SADV SGL** command. The **SADV SGL** command can come from either the control panel keyboard or the system controller via HP-IB. In this mode, you can set up other plug-in modules prior to executing the **SADV SGL** command.

Step 9. When the HP 3235 receives the specified **SADV** signal, it opens the previous channel closed (but not the bank or backplane relays) and closes the next channel in the channel list and any necessary bank and backplane relays. If the next channel in the channel list is on a different Plug-In module, the backplane relay on the first module opens.

The HP 3235 also outputs a pulse to the destination designated by the **CHCLOSED** command. This pulse indicates that the channel relays have closed and settled. For additional information, see the **CHCLOSED** command in the HP 3235 Language Reference Manual.

Step 10. Repeat Steps 5 through 9 for the remaining channels in the channel list.

Step 11. After the multimeter measures the last channel in the channel list, the

HP 3235 waits for one more **SADV** signal (if specified).

Step 12. Finally, the HP 3235 opens any channel and backplane relays associated with the scan. All bank relays closed prior to the scan or within the scan remain closed.

Step 13. If the **FAST** parameter was specified in the **MEAS** command, the HP 3235 opens all the backplane relays relays used in the scan. See the **FAST** parameter description later in this chapter.

Reading Storage

You can store the measurements in simple variables or arrays inside the HP 3235. Refer to Chapter 7 for more information on using variables and arrays. Use the **MEM** *<var>* command to send the measurement(s) to a variable or array. Dimension all arrays prior to executing the **MEAS** command. All arrays are base 0. Use the **VREAD** command to transfer the measurements out of the array.

For example:

```
10 DIM Readings(7)
20 OUTPUT 709; "DIM RDGS (7)"  ! DIMENSION FOR EIGHT ELEMENTS
30 OUTPUT 709; "MEM RDGS"
40 OUTPUT 709; "MEAS DCV 1-8"
50 OUTPUT 709; "MEM OFF"
60 OUTPUT 709; "VREAD RDGS"
70 ENTER 709; Readings(*)
80 PRINT Readings(*)
90 END
```

Scanning Examples

Let's look at a few simple scanning examples. As with other examples in this manual, the HP 3235 Multimeter Module is installed in mainframe slot 800. Relay multiplexer modules (HP 34501, 34502, or 34507) are installed in slots 000 through 300.

Scanning Example 1. In this first example, we only measure one channel. All the default triggers are used.

```
10 OUTPUT 709; "RST; USE 800; MEAS DCV 122"  ! MEASURE DCV CHAN 122
20 ENTER 709; A
30 PRINT A
40 END
```

In this example, the HP 34520 multimeter makes a DC voltage measurement on channel 122. Because we did not specify an Analog Bus, Analog Bus 0 is used. Bank relays 170 and 171 remain closed after the measurement.

Scanning Example 2. For this second example, we use a channel list of contiguous channels. Again, all the default triggers are used.

```
10 DIM Readings (31) ! 32 ELEMENT REAL ARRAY IN CONTROLLER
20 OUTPUT 709; "RST; USE 800" ! SPECIFY MULTIMETER LOCATION
30 OUTPUT 709; "MEAS DCV 201-238"
40 ENTER 709; Readings(*) ! TRANSFER READINGS TO CONTROLLER
50 PRINT Readings(*)
```

```
60 END
```

Again, we did not specify an Analog Bus, therefore, **MEAS** defaults to Analog Bus 0. At the end of this scan, bank relays 270, 271, and 272 remain closed.

Scanning Example 3. In this third example, we use an external **STRIG** signal and automatic scan advance. You would use this type of trigger arrangement when you want to initiate the scan on some external event. We also specify Analog Bus 2 for the measurements. The HP 3235 stores the readings in an array called CHREADS.

```
10 DIM Readings (31)  ! 32 ELEMENT REAL ARRAY IN CONTROLLER
20 OUTPUT 709; "RST; USE 800" ! SPECIFY MULTIMETER LOCATION
30 OUTPUT 709; "DIM CHREADS (31)" ! 32 ELEMENT ARRAY IN HP 3235
40 OUTPUT 709; "STRIG EXTIN"  ! EXTERNAL TRIGGER INPUT BNC
50 OUTPUT 709; "MEM CHREADS; MEAS DCV AB2 201-238"
55 ! HP 3235 waits for STRIG signal here
56 ! HP-IB communication is held off
60 OUTPUT 709; "MEM OFF; VREAD CHREADS"
70 ENTER 709; Readings(*) ! TRANSFER READINGS TO CONTROLLER
80 PRINT Readings(*)
90 END
```

After the HP 3235 receives the **MEAS** command it waits to receive a pulse from the External Trigger In BNC connector. When it receives the pulse, it executes the **MEAS** scanning sequence. At the end of this scan, bank relays 270, 271, and 272 remain closed.

Scanning Example 4. In this example we use the **SADV HOLD** and **SADV SGL** commands to execute the scanning.

```
10 DIM Readings (7) ! EIGHT ELEMENT REAL ARRAY IN CONTROLLER
20 OUTPUT 709; "RST; USE 800" ! SPECIFY MULTIMETER
30 OUTPUT 709; "DIM CHREADS (7)" ! EIGHT ELEMENT ARRAY IN HP 3235
40 OUTPUT 709; "SADV HOLD" ! SCAN ADVANCE HOLD
50 OUTPUT 709; "MEM CHREADS; MEAS DCV 101-108"
60 FOR I=1 TO 8
70 ! OPTIONAL MODULE CONFIGURATION DURING THIS TIME
80 OUTPUT 709; "SADV SGL" ! SINGLE STEP THE SCAN ADVANCE
90 NEXT I
100 OUTPUT 709; "MEM OFF; VREAD CHREADS"
110 ENTER 709; Readings (*) ! TRANSFER READINGS TO CONTROLLER
120 PRINT Readings(*)
130 END
```

Note that one **SADV** signal is required after each channel measurement including the final one which opens the final channel and completes the scan.

Scanning Example 5. In this example we use the External Trigger In BNC to drive the **STRIG** signal. Scan advance is automatic.

```
10 DIM Readings(5)
20 OUTPUT 709; "RST; USE 800"
30 OUTPUT 709; "SCRATCH; DIM TEMP(5)"
```

```
40 OUTPUT 709; "STRIG EXTIN" ! EXTERNAL SCAN TRIGGER
50 OUTPUT 709; "MEM TEMP"
60 OUTPUT 709;"MEAS DCV,301-306"
61 ! HP 3235 WAITS FOR THE STRIG SIGNAL
70 OUTPUT 709; "MEM OFF; VREAD TEMP"
80 ENTER 709; Readings(*) ! TRANSFER READINGS TO CONTROLLER
90 PRINT Readings(*)
100 END
```

To trigger the scan from either the HP 34504 Coaxial Multiplexer Module or the HP 34506 Coaxial Matrix module via Trigger Bus 1, replace line 40 with the following:

```
40 OUTPUT 709; "STRIG TB1; TRIGBUF 501 IN"
```

The coaxial module is in slot 500 of the mainframe for this example.

The FAST Parameter

The **FAST** parameter for the **MEAS** command is used only when multiple HP 34502 and 34507 Relay Modules are used for scanning. **FAST** provides the fastest scanning speed. It considers the channel list and closes all the backplane relays necessary for the scan. This decreases the scanning time since the backplane relays (armature relays) are 10 times slower than the channel relays (dry reed or mercury-wetted relays). Thus, the HP 3235 does not have to wait for each backplane relay to settle when scanning across multiplexer modules. This does, however, add a small amount of circuit capacitance due to the analog backplane relays being closed during the scan. At the end of the scan, all the backplane relays open automatically.

For example, if you execute the command:

```
OUTPUT 709; "MEAS DCV AB2 FAST 101-238"
```

then the backplane relays 192 and 292 close and remain closed during the scan. At the end of the scan the two relays open.

For fastest scanning, manually close the bank relays to configure the multiplexer banks prior to scanning so that only the channel relays move during the scan.

CAUTION

The **FAST** *parameter does not open all the user channels that might be closed (from previous commands) and connected to the Analog Bus used for the scanning. To avoid possible damage to the HP 3235, the Plug-in modules, or any external circuitry, ensure that no channel connects to the Analog Bus used for the scan. You can do this with the command:* **OPEN 1-999** *(***OPEN 1-7999** *for the extenders) if no relays are prohibited.*

4-Wire Measurements

If you need to scan with a 4-wire resistance measurement use **OHMF** as the *function* parameter with the **MEAS** command. The scanning action is the same as described earlier for **MEAS** but with the following exceptions:

1. Your channel list must specify channels only in banks 0 or 1 of the relay

multiplexer module(s). These are the multimeter current source connections. The HP 3235 automatically closes the corresponding channel (nn+20) in banks 2 or 3 for the multimeter sense connections.

2. You can specify either AB0 or AB1 for the Analog Bus. This bus is for the multimeter current source. The HP 3235 automatically selects ABn+2 for the sense bus.

For example, If you execute:

```
OUTPUT 709; "MEAS OHMF AB1 101-108"
```

channels 101 through 108 (inclusive) are used for the multimeter current source connections. Channels 121 through 128 (inclusive) are used for the multimeter sense connections. Channel pairs (nn and nn+20) close simultaneously. The appropriate bank and backplane relays (i.e. to connect channel 101 and channel 121 to the respective analog buses) also close. Analog Bus 1 carries the multimeter current source and Analog Bus 3 carries the sense signal.

Figure 6-2 shows the wiring connections for 4-wire scanning.

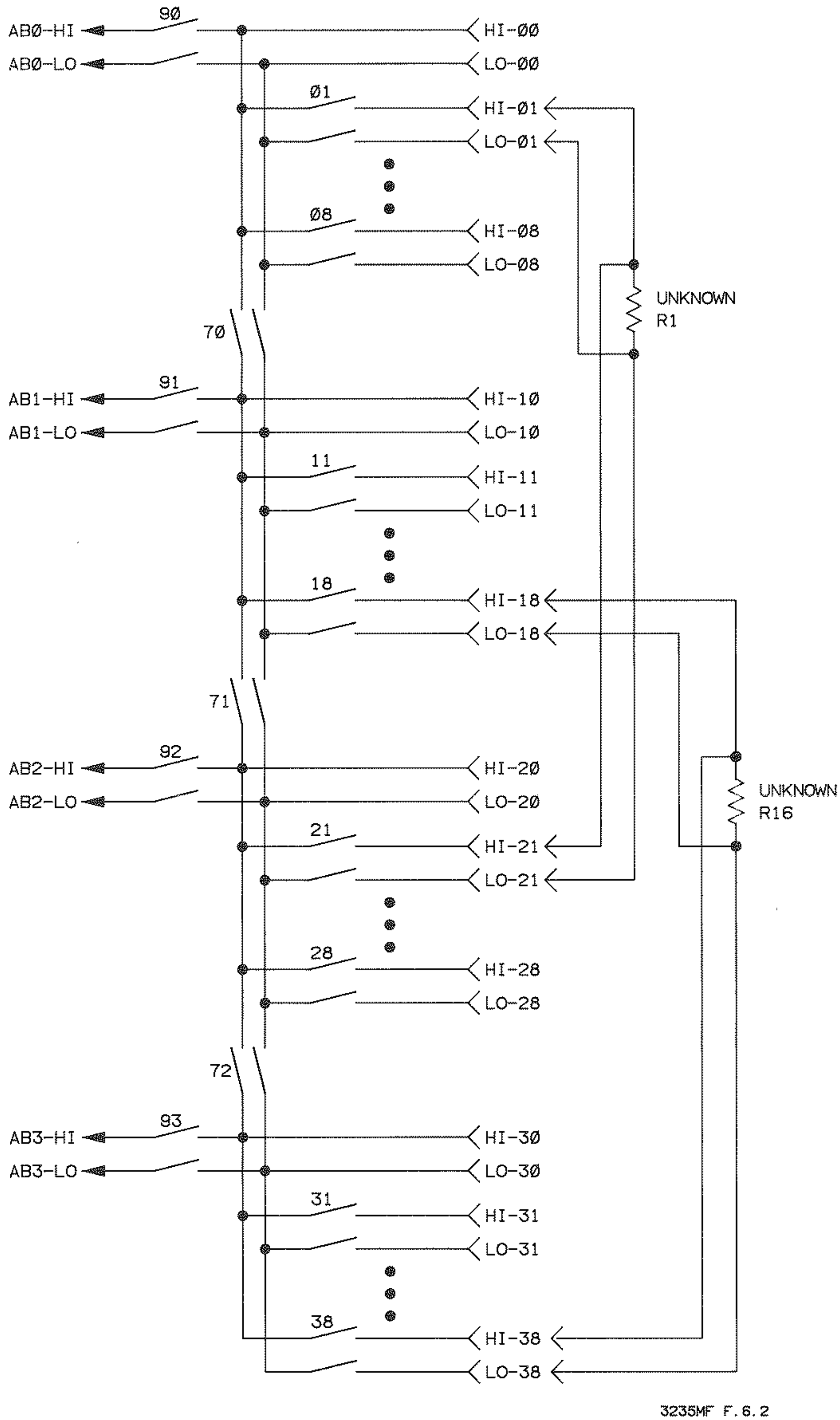


Figure 6-2. 4-Wire Scanning

3. The source and sense banks must not be shorted together.

The following example shows a simple 4-wire scanning sequence.

```
10  DIM Readings(7)
20  OUTPUT 709; "RST;USE 800"
30  OUTPUT 709; "SCRATCH;DIM RDGS(7)"
40  OUTPUT 709; "MEM RDGS"
50  OUTPUT 709; "MEAS OHMF AB1,101-108"
60  OUTPUT 709; "MEM OFF;VREAD RDGS"
70  ENTER 709; Readings(*)
80  PRINT Readings(*)
90  END
```

Scanning With Verification

If you do not need the actual measurement results from a scan but only a pass or fail limit comparison, you can use the **VERIFY** command. The **VERIFY** command reduces the data formatting and transfer time required by your system controller and the controller time required to make the comparisons. The **VERIFY** command uses the HP 34520 Multimeter Module for the measurements.

The **VERIFY** command is similar to the **MEAS** command except that **VERIFY** does not store the measurement results or send them to the control panel or HP-IB output buffer. Instead, the HP 3235 compares each measurement to a lower and upper limit which you specify. If, at the end of the scan, all measurements are within the specified limits, the **VERIFY** command returns "0". If any one measurement is outside the specified limits, the **VERIFY** command returns "1".

The **VERIFY** command does not return the measurement values or the channel number(s) which failed, if any. To do this, you can write an HP 3235 subroutine using the **MEAS** command to store readings in an internal array. Then you can use the **LIMIT** command to limit test the measurements. An example program follows:

```
10  OUTPUT 709; "REAL LOWER(7),UPPER(7),RDGS(7)"
20  OUTPUT 709; "FILL UPPER 5.1,5.1,12.2,-7.05,12.2,18.5,12.2,5.1"
30  OUTPUT 709; "FILL LOWER 4.9,4.9,11.8,-6.95,11.8,17.8,11.8,4.9"
40  OUTPUT 709; "SUB MYVERIFY"
50  OUTPUT 709; "MEM RDGS"
60  OUTPUT 709; "MEAS DCV 101-108"
70  OUTPUT 709; "MEM OFF"
80  OUTPUT 709; "LIMIT RDGS,LOWER,UPPER"
90  OUTPUT 709; "SUBEND"
100 !
110 DIM Dat(7)
120 OUTPUT 709; "CALL MYVERIFY"
130 ENTER 709; Passfail
140 IF Passfail = 0 THEN
150     PRINT "TEST PASSED"
160 ELSE
170     PRINT "TEST FAILED"
180     OUTPUT 709; "VREAD RDGS"
190     ENTER 709; Dat(*)
200 END IF
210 ! Process the Readings as Necessary
```

The **VERIFY** command has the following syntax:

VERIFY [*function*,] [**AB***n*,] [**FRONT**,] [**FAST**,] *low_array*, *high_array*, *ch_list*

The upper and lower limit arrays are internal HP 3235 arrays. See Chapter 7 for additional information on arrays. You need to fill the upper and lower limit arrays prior to executing the **VERIFY** command. Both arrays must be dimensioned REAL and must be at least the same size as the channel list. **VERIFY** compares the measurement from the first channel in the scan list to the first elements in the lower and upper limit arrays. Then it compares the measurement from the second channel to the second elements in the upper and lower arrays, etc. Remember, all arrays are option base 0 (the first element is number 0). You can also use an array to store the channel list. The following example demonstrates how to use **VERIFY**.

```
10  OUTPUT 709; "RST;USE 800"
20  OUTPUT 709; "DIM UPPER(4),LOWER(4)"
30  OUTPUT 709; "FILL UPPER 5.07,5.07,12.5,-12.7,5.05"
40  OUTPUT 709; "FILL LOWER 4.95,4.95,11.8,-11.6,4.95"
50  OUTPUT 709; "VERIFY DCV LOWER,UPPER,101,104,108,102,106"
60  ENTER 709; A
70  IF A=0 THEN PRINT "TEST PASSED"
80  IF A=1 THEN PRINT "TEST FAILED"
90  END
```

If the **VERIFY** test fails, the HP 3235 sets bit 10 in the status word. This allows you to interrupt the system controller (SRQ Interrupt) if the test fails. The **VERIFY** command does not clear the status register bit if the test passes. Therefore, you should always clear the register before beginning the scan. Refer to Chapter 10 in this manual for information and examples on using interrupts.

To use **VERIFY** in HP 3235 subroutines, use the **MEM** command to send the result ("0" or "1") to an internal variable. You can test the variable with an **IF...THEN** statement as shown in the following program.

```
10  OUTPUT 709; "RST;USE 800"
20  OUTPUT 709; "DIM UPPER(4),LOWER(4)"
30  OUTPUT 709; "FILL UPPER 5.07,5.07,12.5,-12.7,5.05"
40  OUTPUT 709; "FILL LOWER 4.95,4.95,11.8,-11.6,4.95"
50  OUTPUT 709; "SUB TESTER"
60  OUTPUT 709; "MEM A"
70  OUTPUT 709; "VERIFY DCV LOWER,UPPER 101,104,108,102,106"
80  OUTPUT 709; "MEM OFF"
90  OUTPUT 709; "IF A=1 THEN; DISP 'TEST FAILED'"
100 OUTPUT 709; "ELSE; DISP 'TEST PASSED'"
110 OUTPUT 709; "END IF"
120 OUTPUT 709; "SUBEND"
130 END
```

Scanning with External Instruments

One important feature of the HP 3235 is its ability to work with other instruments while scanning. You can connect high speed voltmeters, counters, or other measuring or stimulus instruments to the analog bus and use the synchronizing signals in the HP 3235 to coordinate the scan.

Two commands, **SCAN** and **PSCAN**, govern the scanning operation. Other commands, such as **CHCLOSED** and **SADV**, coordinate the triggering between the HP 3235 and your external measuring instrument. The following information explains the physical connections and software setup necessary for scanning with an external instrument.

Connections Required

Figure 6-3 shows the physical connections required between the HP 3235 and your external measuring instrument. For this discussion, we assume the external instrument is a voltmeter. The concepts are similar regardless of the type of instrument used.

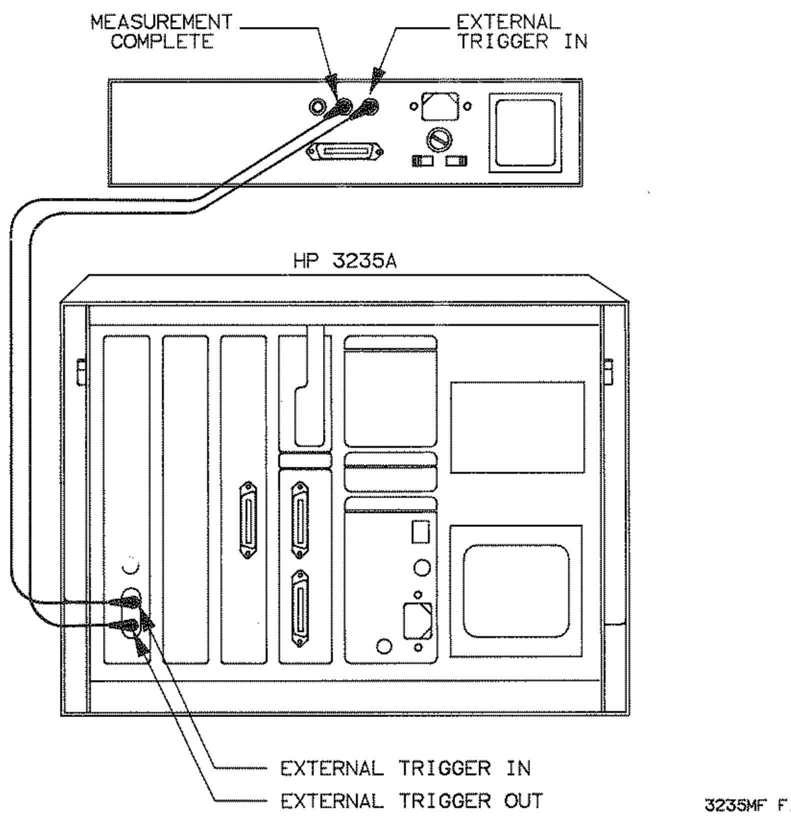


Figure 6-3. Connections for External Instrument Scanning

The analog bus connections can be from either the HP 3235 Backplane Access Module (see Chapter 2 and the Installation Manual) or the bank common connections from a multiplexer plug-in module.

The **CHCLOSED** command (discussed later in this chapter) specifies the destination for a negative going TTL pulse after each channel in the channel list closes and has settled. This pulse triggers the external voltmeter. When the external voltmeter completes the measurement, it outputs a pulse (VOLTMETER COMPLETE or MEASUREMENT COMPLETE) to the HP 3235. This pulse advances the scanner channel (see the **SADV** command discussed later). When the channel closes and settles, the HP 3235 outputs a new trigger signal and the process repeats.

External Instrument Setup

Any external instrument setup including measurement function, range, measurement complete source, external trigger source, accuracy, etc., should be done before beginning the scan. In this way, the instrument is prepared for the scan command.

HP 3235 Setup

Aside from the physical connections necessary (explained earlier), you must program the HP 3235 to output the necessary trigger pulse and receive the necessary scan advance signal. The next few paragraphs describe how to set up the HP 3235 for external instrument scanning.

Scan Trigger

The **STRIG** command specifies the signal that initiates a channel scan. When the HP 3235 receives the scan trigger pulse, it closes the next channel in the scan list and pulses (see **CHCLOSED**) the external instrument. The power-on default for this command is an automatic trigger. Thus, when you execute either the **SCAN** or **PSCAN** commands, the scanning starts immediately. Possible sources for the **STRIG** signal are listed in Table 6-5.

Table 6-5. STRIG (Scan Trigger) Sources

Source Parameter	Definition
AUTO	Automatic triggering with receipt of SCAN command (power-on default)
EXTIN	Initiate Scan on External Trigger In
TB0	Initiate Scan on Trigger Bus 0 pulse
TB1	Initiate Scan on Trigger Bus 1 pulse
HOLD	Wait for STRIG SGL command
SGL	Software Scan Trigger single

Channel Closed Signal

The **CHCLOSED** command specifies the destination for the channel closed signal. This signal is a negative going TTL pulse that occurs when the scan list channel and the bank and backplane relays have closed and settled. Table 6-6 shows the possible **CHCLOSED** sources.

Table 6-6. CHCLOSED (Channel Closed) Sources

Source Parameter	Definition
EXT	Pulse the External Trigger Out BNC
TB0	Pulse Trigger Bus 0
TB1	Pulse Trigger Bus 1
OFF	No Channel Closed Pulse

The purpose for the **CHCLOSED** signal is to inform external instruments that the relays have closed and settled. For external scanning set it for EXT and connect the HP 3235 External Trigger Out BNC to your measuring instrument's External Trigger In. You could also use one of the trigger buses (TB0 or TB1) and either the HP 34505 or 34506 coaxial modules.

Scanner Advance

After the measuring instrument completes its measurement, it must inform the HP 3235 to advance to the next channel in the channel list. This is known as scanner advance and is normally done with the measurement complete or voltmeter complete signal.

The **SADV** command in the HP 3235 specifies the source for receiving the Scanner Advance signal. For external instrument scanning, set **SADV** for **EXTIN** and connect the measuring instrument's Measurement Complete output to the HP 3235's External Trigger In BNC connector. The **SADV EXTIN** must be a negative going TTL level pulse. Table 6-7 shows other possible **SADV** sources.

Table 6-7. SADV (Scan Advance) Trigger Sources

Source Parameter	Definition
AUTO	Automatic Scan Advance
EXTIN	Advances on External Trigger In BNC
TB0	Advances on Trigger Bus 0 pulse
TB1	Advances on Trigger Bus 1 pulse
HOLD	Wait for SADV SGL before advancing
SGL	Software Channel advance

Scanning

The **SCAN** command allows you to automatically measure one or more channels using an external instrument. **SCAN** automatically closes appropriate bank and backplane relays for the measurement.

SCAN has the following format:

SCAN [AB*n*,][FAST,]*ch_list*

The optional **AB*n*** parameter specifies the analog bus to use in the scan. If you do not specify an analog bus, the **SCAN** command does not connect to any analog backplane bus.

The **FAST** parameter (discussed in detail later) is used only with multiple HP 34502 or 34507 Relay Modules.

The *ch_list* parameter is the list of channels you want to scan. Each channel number must be in the form *esnn* where *e* represents the extender number (0 or blank for the mainframe), *s* is the slot number (0 through 9), and *nn* is the channel number. You can specify a single channel (*esnn*), multiple channels separated by commas (*esnn,esnn,esnn, ... esnn*), contiguous channels (*esnn-esnn*), or groups of contiguous channels (*esnn-esnn,esnn-esnn*). You can also store the channel list in an array (see Chapter 7).

When you specify a contiguous series of channels, the HP 3235 ignores any unused channel numbers. For example, the HP 34501, 34502, or 34507 relay modules do not use channel numbers 09, 10, 19, 20, 29, and 30. Therefore, if you specify channels 1-38 the unused channels are ignored.

A typical **SCAN** command looks like:

```
OUTPUT 709; "SCAN 201-232"
```

This command scans channels between 201 and 232 (inclusive). It defaults to analog bus 0.

How SCAN Works

The **SCAN** command performs nine steps. We'll list these steps and then discuss them in detail.

1. Closes the backplane relays associated with the scan if the **FAST** parameter was specified.
2. Waits for the specified **STRIG** signal. If you did not previously specify an **STRIG** signal, **SCAN** skips this step.
3. Looks at the first channel in the channel list. Opens all relays connected to the same bank and then closes the appropriate bank and backplane relays and the specified channel relay to connect the signal to the specified analog bus.
4. Outputs the **CHCLOSED** signal through the specified port.
5. Waits for the specified **SADV** signal from the specified source.
6. Opens the channel and closes the appropriate relays for the next channel in the channel list.
7. Repeats steps 3 through 6 for the remaining channels in the channel list.
8. Waits for one more **SADV** pulse (if specified) after the last channel has been closed and the **CHCLOSED** signal sent.
9. Opens all the backplane and channel relays associated with the last channel in the channel list. Bank relays remain closed.

Let's look at these steps in more detail.

Step 1. If you specified the **FAST** parameter in the **SCAN** command, the HP 3235 looks at the channel list and closes the necessary backplane relays on the multiplexer modules involved in the scan. This improves the scanning speed. Refer to the **FAST** parameter discussion later in this chapter for more information.

Step 2. The HP 3235 waits for a Scan Trigger signal. Use the **STRIG** command to specify the source of the Scan Trigger signal. The power-on default source for the Scan Trigger is an automatic trigger called **SCAN**. Thus, if you do not specify a Scan Trigger signal source, the HP 3235 uses the default source and automatically triggers.

Table 6-8 shows the possible Scan Trigger sources:

Table 6-8. Scan Trigger (STRIG) Sources

Parameter	Definition
AUTO	Automatic Triggering on receipt of STRIG (Default)
EXTIN	Starts on External Trigger In (BNC) pulse
TB0	Starts when Trigger Bus 0 is pulsed
TB1	Starts when Trigger Bus 1 is pulsed
HOLD	Wait for STRIG SGL each time to start
SGL	Software Scan Trigger

The **STRIG AUTO** parameter provides automatic triggering. In general, this also provides the easiest triggering.

STRIG EXTIN accepts a negative going TTL level trigger pulse from the External Trigger In BNC connector on the back panel of the HP 3235.

The two Trigger Buses TB0 and TB1 can also act as Scan Trigger sources. You can use the **DRIVETBn** command to specify the source for driving the trigger buses or you can drive the trigger buses through either the HP 34504 Coaxial Multiplexer Module, HP 34506 Coaxial Matrix Module, or the HP 34522 Digital I/O module. See the **TRIGBUF** command for more information.

While **EXTIN**, **TB0**, and **TB1** provide hardware triggers, you can specify a software trigger with the **HOLD** and **SGL** parameters. When you specify **STRIG HOLD**, the HP 3235 continues with the next command received and holds up the scan until it receives the **STRIG SGL** command. The command can come from either the control panel keyboard or the system controller via HP-IB. In this mode, you can set up other plug-in modules prior to executing the **STRIG SGL** command.

Step 3. Refer to Figure 6-4. In this step, the HP 3235 opens any relays already closed in the Plug-in Module Bank associated with the channel in the channel list. For example, if you specify channel 13 in the scan list, the HP 3235 opens channels 11 through 18. The HP 3235 then closes the channel specified (in our example, channel 13) and any backplane and bank relays necessary to connect the source (channel 13) to the specified analog bus.

If you did not specify an analog bus with the **SCAN** command, the HP 3235 defaults to Analog Bus 0. Thus to connect channel 13, the HP 3235 closes channel 13, bank relay 70, and backplane relay 90.

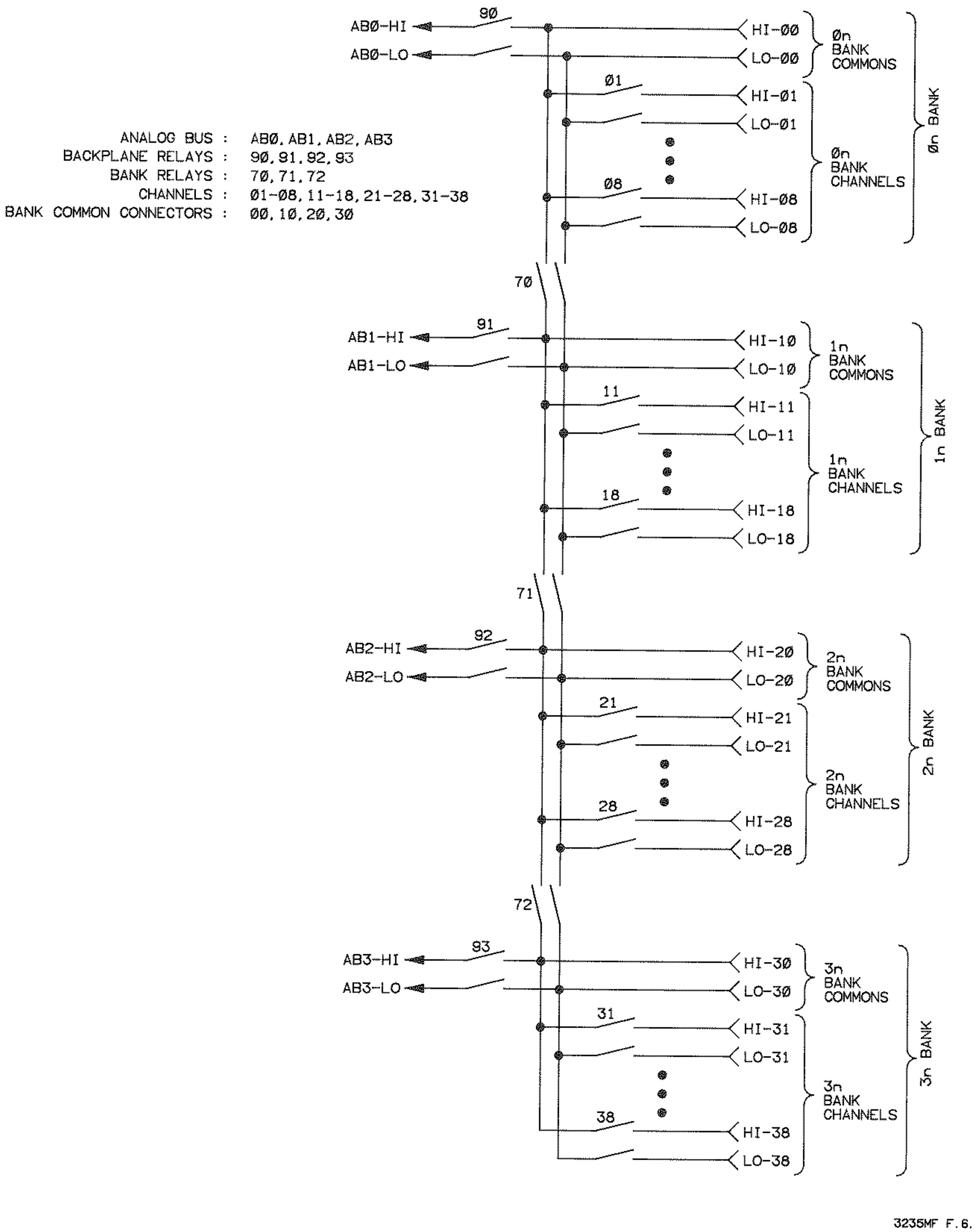


Figure 6-4. A Typical General Purpose Scanner

If, as another example, you specified Analog Bus 1 and channel 33, the HP 3235 would close channel 33, bank relays 71 and 72, and backplane relay 91.

CAUTION

The SCAN command opens all the channel relays in the same bank as the channel in the channel list. It does not open any bank relays (relay numbers 70 through 72). Therefore, to prevent possible damage to the HP 3235, the Plug-in Module, and/or any external circuitry, you should ensure that all bank relays and backplane relays not essential to your configuration, are opened prior to initiating the **SCAN** *command. Execute the* **OPEN 0-999** *command.*

One solution is to close bank relays 70, 71, and 72 prior to initiating the scan. With this done, the HP 3235 sees all 32 channels as one bank. When the **SCAN** *command opens all channels in the bank, all 32 channels open. You can, of course, close any combination of bank relays on a Plug-in Module to redefine the bank size.*

Step 4. The HP 3235 outputs a pulse to the destination designated by **CHCLOSED**. This pulse triggers the external measuring instrument. For scanning with an external measuring instrument, use the **CHCLOSED EXT** command. Table 6-9 shows the possible **CHCLOSED** parameters.

Table 6-9. CHCLOSED (Channel Closed) Trigger Parameters

Parameter	Definition
EXT	Pulse External Trigger Out BNC
TB0	Pulse Trigger Bus 0
TB1	Pulse Trigger Bus 1
OFF	No Channel Closed Pulse

Step 5. The HP 3235 waits for the specified Scan Advance Signal specified with the **SADV** command. This indicates that the measuring instrument has completed the measurement. For scanning with an external measuring instrument, use **SADV EXTIN.**

The power-on default for **SADV** is an automatic trigger called **AUTO**. Thus, if you did not specify another **SADV** source, the HP 3235 will use the default source and automatically advance. Table 6-10 shows the possible Scan Advance Trigger (**SADV**) sources.

Table 6-10. SADV (Scan Advance) Trigger Sources

Parameter	Definition
AUTO	Automatic Channel Advance (Default)
EXTIN	Advances on External Trigger In (BNC) pulse
TB0	Advances when Trigger Bus 0 is pulsed
TB1	Advances when Trigger Bus 1 is pulsed
HOLD	Wait for SADV SGL each time to advance
SGL	Software Scan Advance Single

The **SADV AUTO** parameter provides automatic channel advance. In general, this also provides the fastest channel advance method.

SADV EXTIN accepts a TTL level trigger pulse from the External In BNC connector on the back panel of the HP 3235.

The two Trigger Buses TB0 and TB1 can also act as Scan Advance sources. You can use the **DRIVETB***n* command to specify the source driving the trigger bus or you can drive the trigger buses from the HP 34506 Coaxial Matrix Module or the HP 34504 Coaxial Multiplexer Module. See the **TRIGBUF** command for more information.

While **EXTIN**, **TB0**, and **TB1** provide hardware triggers, you can specify a software trigger with the **HOLD** and **SGL** parameters. When you specify **SADV HOLD**, the HP 3235 stops the scan and accepts other commands until it receives the **SADV SGL** command. The command can come from either the control panel keyboard or the system controller via HP-IB. In this mode, you can set up other plug-in modules prior to executing the **SADV SGL** command.

Step 6. When the HP 3235 receives the specified **SADV** signal, it opens the last channel closed (but not the bank relays) and closes the next channel in the channel list and any necessary bank relays. The backplane relays change only if the next channel in the channel list is on a different Plug-in Module.

Step 7. Repeats Steps 3 through 6 for the remaining channels in the scan list. If the next channel in the list is on a different plug-in module, the backplane relays on the first module open.

NOTE

Remember, the HP 3235 does not open any bank relays during the scanning process. Therefore, configure the bank and backplane relays ahead of time so that you know what relays are closed throughout the scan.

Step 8. After the last channel in the channel list is measured, the HP 3235 waits for one more **SADV** signal. If you did not specify an **SADV** signal, this happens automatically.

Step 9. Finally, the HP 3235 opens any channel and backplane relays associated with the scan. All bank relays closed prior to the scan or within the scan remain closed.

Scanning Example

The following example shows the general procedure for scanning with an voltmeter. The voltmeter complete signal connects to the HP 3235's External Trigger In BNC and the voltmeter's external trigger input connects to the HP 3235's External Trigger Out BNC. Connect the voltmeter's input terminals to Analog Bus 0 through the Backplane Access Module.

```
10 ! PROGRAM VOLTMETER AS NECESSARY HERE E.G. TRIG EXT, FUNC DCV
20 OUTPUT 709; "STRIG EXTIN; CHCLOSED EXT"
30 OUTPUT 709; "SCAN 301-306"
40 ! ENTER READINGS FROM VOLTMETER
50 END
```

The next example uses an HP 3478 multimeter as the measuring instrument. Line 10 sets the multimeter to: 2-wire ohms (F3), 300kΩ range (R5), 4 1/2 digits of display (N4), External Trigger (T2), and Autozero on (Z1).

```
10 OUTPUT 723; "F3R5N4T2Z1"  ! PROGRAM HP 3478 MULTIMETER
20 OUTPUT 709; "STRIG EXTIN; CHCLOSED EXT"
30 OUTPUT 709; "SADV HOLD"
40 OUTPUT 709; "SCAN AB0 310-306"
50 FOR I=1 TO 6
60 ENTER 723; A(I)
70 OUTPUT 709; "SADV"
80 PRINT A(I)
90 NEXT I
100 END
```

The FAST Parameter

Use the **FAST** parameter for the **SCAN** command only when multiple HP 34502 Reed Relay Modules or HP 34507 Mercury-Wetted Relay Modules are used in the scan. It provides the fastest scanning speed. **FAST** considers the channel list and closes all the backplane relays necessary for the scan. This decreases the scanning time since the backplane relays (armature relays) are 10 times slower than the channel relays (reed relays). Thus, the HP 3235 does not have to wait for each backplane relay to settle. This does, however, add a small amount of circuit capacitance. At the end of the scan, the backplane relays open automatically.

For example, if you execute the command:

```
OUTPUT 709; "SCAN AB2 FAST 101-238"
```

then the backplane relays 192 and 292 close and remain closed during the scan. At the end of the scan the two relays open.

CAUTION

The **FAST** *parameter does not open all the user channels already closed (from previous commands) and connected to the Analog Bus used for the scanning. To avoid possible damage to the HP 3235, the Plug-in modules, or any external circuitry, ensure that no channel is connected to the Analog Bus used for the scan. You can do this with the* **OPEN** *0-999 command if no relays are prohibited.*

How PSCAN Works

If you need to scan with a 4-wire measurement (for example, 4-wire resistance measurements) use the **PSCAN** command. The scanning action is the same as **SCAN** but with two exceptions:

1. You may specify up to two analog buses. How you use these buses is left up to you. The default is that no analog bus is used.

2. You must specify two parallel channel lists. Respective channels in each list close simultaneously. For example, you could specify 101-108,131-138. This has the same number of channels in each list. You can also use arrays to specify channel lists. Each pair of channels to be closed must be in different banks.

If you only specify one list of channel numbers, the HP 3235 divides the list in half to create two lists. For example, if you specify 101,102,105,207,203,208 then the HP 3235 creates two lists; the first list is 101, 102, 105, the second list is 207, 203, and 208. Channel 101 is paired with 207, channel 102 is paired with 203, and channel 105 is paired with 208.

If you execute:

```
OUTPUT 709; "PSCAN AB1,AB3,101-108,201-208"
```

then channels 101 through 108 (inclusive) connect to Analog Bus 1. Channels 201 through 208 (inclusive) connect to Analog Bus 3. Channel pairs (101 and 210, 102 and 202, etc.) close together along with the appropriate bank and backplane relays.

Figure 6-5 shows the wiring connections for 4-wire scanning.

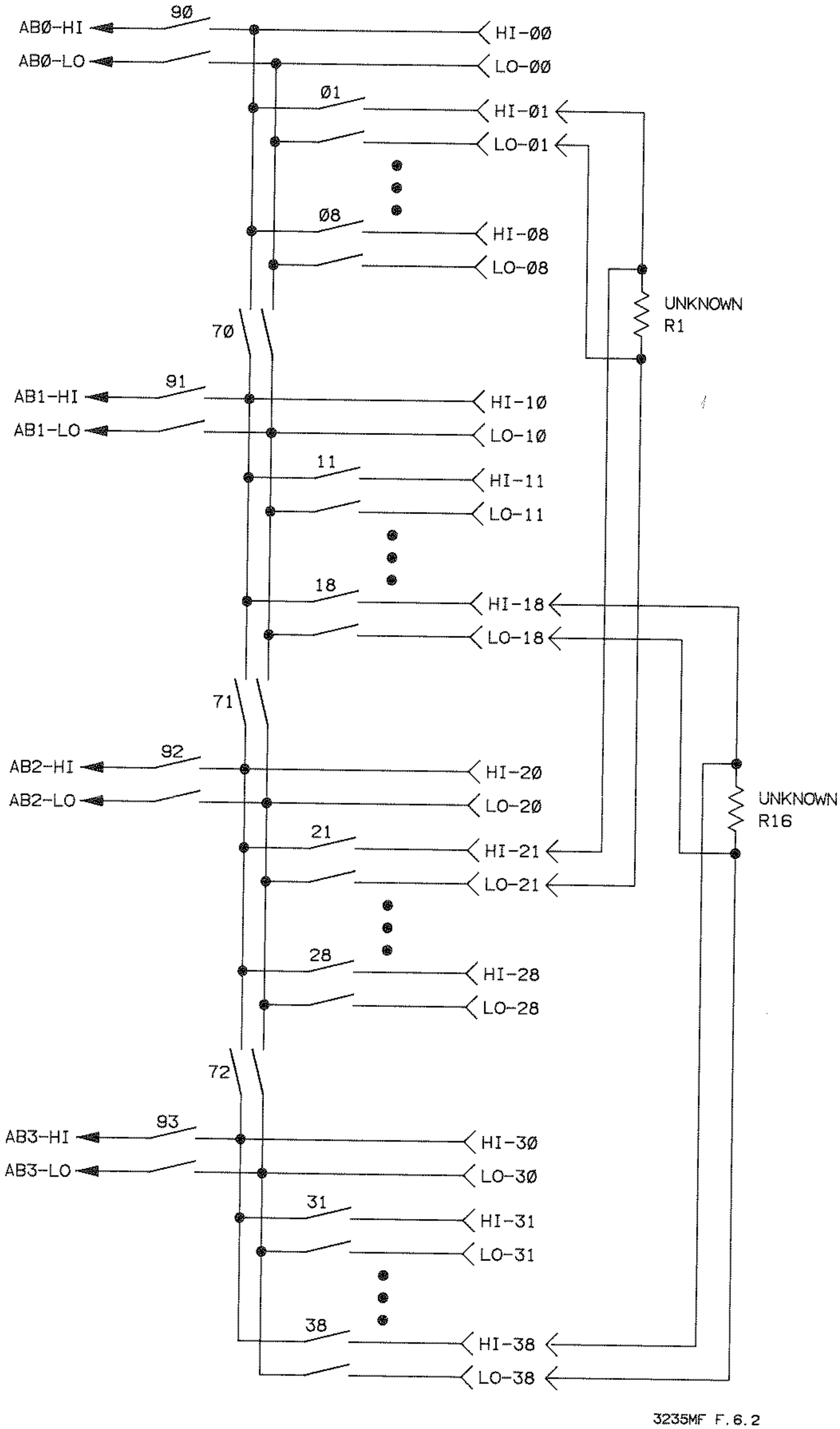


Figure 6-5. 4-Wire Scanning

Optimizing for Speed

Whether you are using the HP 34520 Multimeter or some external instrument, several things can increase the overall scanning speed. However, tradeoffs are inherent with everything you do to increase speed.

Some operations of the HP 3235 or the HP 34520 Multimeter require extra overhead time to complete. Many of these operations are not important when speed is desired. Therefore, if maximum measurement speed is your goal, consider each of the following items carefully.

The following discussions focus on the HP 34520 Multimeter. However, many of the considerations are applicable to other measuring instruments. Read them carefully and use those that apply to your test system.

The Multimeter

Several factors can affect the speed of measurements. These factors include ranging the multimeter, the multimeter autozero, delay time, and integration time. Each of these have tradeoffs that might affect measurement accuracy.

Ranging

Autoranging each measurement can require a large amount of time. If possible, set the multimeter to a fixed range with the **RANGE** command. You must determine the maximum signal to be measured and set the multimeter to an appropriate range in advance. The disadvantage of this is that significantly smaller signals may not be measured with the highest resolution. For the fixed range to stay fixed you should also specify the measurement function prior to executing the **MEAS** command. If you specify a different measurement function as a parameter of the **MEAS** command, it turns autorange on.

If you have a wide variety of signal levels to measure, consider organizing the channel list and/or your channel input wiring to group signal levels by value. That is, group all the high values together and the low values together. You can use two or more **MEAS** scan lists, setting the range for each on, to measure the signals.

Autozero

Turning Autozero off (**AZERO OFF**) increases the reading rate but may reduce long term accuracy. The multimeter's autozero corrects for any drift in the internal circuitry. With autozero on, the multimeter performs an autozero for each measurement. With autozero off, the multimeter takes one zero reading and corrects each subsequent reading with the one zero reading for each range. The multimeter is not likely to drift during short scans. For more information on autozero, refer to the HP 34520 Multimeter Manual.

Integration Time

With the HP 34520 Multimeter, your choice of integration time determines:

- Measurement Accuracy
- Resolution
- Normal Mode Rejection
- Speed

Integration time is one of the primary contributors to fast DC and resistance measurements. Refer to Table 6-11 for a list of the integration tradeoffs for DC voltage measurements. This table shows that, for example, if you select NPLC 100 (NPLC is the Number of Power Line Cycles per integration period), the multimeter requires 1.67 seconds (plus overhead) per reading. While this results in the highest accuracy and resolution, the maximum number of DC voltage readings is only 0.48 readings per second.

Table 6-11. HP 34520 Integration Time Relationships (DCV, 60Hz power line)

NPLC	Power Line Cycles (60 Hz)	Normal Mode Rejection (AC)	Maximum Reading Rate
0.0005	0.0005 (10uS)	0dB	1350
0.005	0.005 (100uS)	0dB	1250
0.1	0.1 (1.67mS)	0dB	360
1.0	1.0 (16.7mS)	60dB	53
10.0	10.0 (166.6mS)	80dB	4.8
100.0	100.0 (1.67 S)	90dB	0.48

For AC measurements, the integration time is less important because of the settling time delay. The AC fast or slow setting (use the **ACBAND** command) has a larger influence for measurements made at .0005, .005, .1 and 10 PLCs. For the fastest AC measurements, use the **ACBAND** command with a frequency greater than 400 Hz to select the AC fast mode. Refer to the HP 34520 Multimeter Manual for additional information.

Plug-in Modules

Use the HP 34502 Reed Relay Modules or HP 34507 Mercury-Wetted Relay Modules for the fastest speed. The reed relays move faster than other types of relays (e.g. armature relays).

If you have more than one HP 34502 Reed Relay Module and/or HP 34507 Mercury-Wetted Relay module involved in the scan list, use the **FAST** parameter with **MEAS**,**VERIFY**, **SCAN**, or **PSCAN**. The **FAST** parameter closes all the backplane relays on all the multiplexer modules involved in the scan list thus eliminating the settling time for those relays.

Memory Usage

Use the **MEM** command to store multimeter measurements in an HP 3235 array rather than send them individually across the HP-IB interface. This saves not only the data transfer time, but the HP 3235 stores the measurements in its internal format and doesn't have to convert them for ASCII transfer.

Data Output

The fastest method to send data over the HP-IB interface is to set **OFORMAT** to Binary. Refer to Chapter 11 in this manual for more information on binary data transfer.

Other Scanning Considerations

Other considerations for scanning include aborting the scan, scanning across extender frames, and HP-IB interaction.

Measuring a Single Channel

In some situations, the **MEAS** command may not provide you with the flexibility you need. To manually scan individual channels, you must close the appropriate relays (channel, bank, and backplane) to connect the signal to the analog bus. Then select the multimeter terminal with the **TERM** command. Execute a **TRIG SGL** command to trigger the multimeter. **TRIG SGL** sends the measurement results either to the HP-IB output buffer or the control panel display (wherever the command originated) or to a specified memory location (see the **MEM** command). The following program shows how to measure one channel.

```
10 OUTPUT 709; "RST; USE 800"
20 OUTPUT 709; "OPEN 1-999" ! OPEN UNNECESSARY CHANNELS
30 OUTPUT 709; "CLOSE 121,171,170,190"  ! CONNECT CH 121
                                        ! TO ABO
40 OUTPUT 709; "TERM ABO"   ! SET MULTIMETER TERMINAL
50 OUTPUT 709; "TRIG SGL"   ! SINGLE TRIGGER THE MULTIMETER
60 ENTER 709; A
70 PRINT A
80 END
```

Aborting the Scan

If, at any time, you need to abort a scan in progress, use the HP-IB CLEAR command (Refer to Appendix D):

```
CLEAR 709
```

Scanning Across Extender Frames

Throughout this chapter we have only discussed scanning within one frame, specifically the mainframe. In cases where you have a large number of points to measure, you may need to have a channel list cover channels in one or more extenders.

The Extender Bus Relays (see Chapter 2) connect the analog buses between extender frames. These relays close automatically at power-on or reset. Thus, to scan channels in different extenders, simply provide the complete channel address in the channel list. The **MEAS**, **VERIFY**, **SCAN**, and **PSCAN** commands close the appropriate backplane and bank relays to connect any channel in any extender to the designated multimeter (which may be in any extender frame).

NOTE

If you want to use an analog backplane bus in one extender for scanning and the same bus in another extender for another purpose, you must open the extender bus relays. Use the **OPEN 9e0n** *command where* e *is the extender number you want to separate (1 - 7) and* n *is the extender bus relay for the particular analog bus you want to separate (0 - 3). See Chapter 2. If you do this you must close the appropriate extender bus relay(s) before scanning across extender frames.*

HP-IB Interaction

At power-on the HP 3235 has its input buffer mode off. This insures that the system controller remains synchronized with the HP 3235 as long as the controller sends double terminators with each output string (see Chapter 11). The HP 3235 completes execution of one command before it accepts the next command from the interface bus. With the **MEAS** or **VERIFY** commands this is necessary. It prevents accidental reconfiguration of the mainframe or extenders. With the **SADV HOLD** and **SADV SGL** commands you can reconfigure the frames as necessary between channels.

However, with the **SCAN** and **PSCAN** commands you may want to reconfigure some external instrument after you set up the channel list. To do this you must terminate the scan command to release the HP-IB interface. Use a semicolon to terminate the scan command and then suppress the carriage return/linefeed (*crlf*) in your system controller. The following two program statements show how to do this with HP series computers. If you have another type of computer refer to its remote programming information to determine how to suppress the *crlf*.

```
OUTPUT 709; "SCAN 101-138;";    ! HP SERIES 200/300; FINAL
                                ! SEMICOLON SUPPRESSES CRLF

OUTPUT 709 USING "#,K"; "SCAN 101-138;"  ! HP SERIES 80 SUPPRESS
                                         ! CRLF
```

If you cannot suppress the *crlf* you could turn the HP 3235's input buffer on. This releases the interface but you may lose command execution synchronization with the system controller. Refer to Chapter 11 for more information on using the input buffer.

Scanning With the Coaxial Modules

Your HP 3235 can scan using either the HP 34504 Switched-Shield Coaxial Multiplexer, the HP 34505 RF Multiplexer Module, or the HP 34508 RF Multiplexer Module. These modules do not have connections to the analog backplane buses. Therefore, you need to connect the bank common to your measuring instrument. You can use either the HP 34520 Multimeter (frequency counter or voltage measurements) or an external measuring instrument.

To use the HP 34520 Multimeter, specify the **FRONT** terminals with the **MEAS** command. Otherwise, the **MEAS** command works the same as described earlier.

Figure 6-6 shows the connections necessary to connect the HP 34505 RF Multiplexer module to the HP 34520 Multimeter.

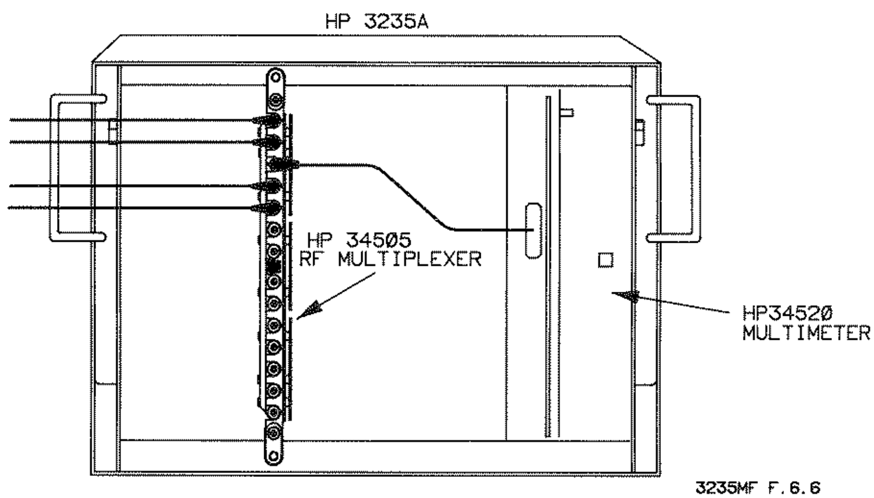


Figure 6-6. Scanning with the HP 34505 and 34520

The following program scans channels 201 through 204 making a frequency measurement on each channel using the HP 34520 Multimeter Module.

```
10 DIM Reads(3)
20 OUTPUT 709; "RST; USE 800"
30 OUTPUT 709; "MEAS FREQ FRONT 201-204"
40 ENTER 709; Reads(*)
50 PRINT Reads(*)
60 END
```

To use an external instrument, make the appropriate connections described under "Scanning With External Instruments" earlier in this chapter.

Scanning With the Digital I/O Module

The **MEAS**, **PSCAN**, **SCAN**, and **VERIFY** commands can be used with the HP 34522 Digital I/O Module to scan through external relays. The scan list for the scanning commands specifies bits on the digital module which drive external relays. The relays are scanned break-before-make in the order listed in the scanning command. Use one of the available handshake modes to ensure proper settling delays. The scanning commands ensure that all bits on the same digital I/O port(s) as the specified bit(s) are set before scanning begins. The size of the port is determined by the **XFERWIDTH** command. For additional information, refer to the section on "Controlling External Relays" in the HP 34522 Digital I/O Module Manual.

Monitoring

The **MON** command allows you to monitor the measurements from one or all channels in a channel list. Refer to Chapter 3 for specific information on using the **MON** command. Use the **MON** *ch#* command to monitor one channel in the channel list during the **MEAS** or **VERIFY** scanning process only. The control panel display is updated with a new reading each time that channel is measured during the scan process. To monitor every channel in the channel list, use the **MON ALL** command.

The following program example shows how to monitor channel 121 in the **MEAS** channel list.

```
10 DIM Readings(31)
20 OUTPUT 709; "RST; USE 800"
30 OUTPUT 709; "MON 121; MEAS 101-138"
40 ENTER 709; Readings(*)
50 PRINT Readings(*)
60 END
```

As another example, replace line 30 in the above program with:

```
30 OUTPUT 709; "MON STATE 800;MEAS 101-138"
```

This displays the state of the multimeter module (slot 800) and shows the measurement functions, trigger sources, etc.

Chapter 7

Variables and Math

Introduction

The HP 3235 provides simple variables, one-dimension arrays, and arithmetic operators and functions. This chapter shows you how to use numeric variables and arrays. It also describes the various math functions available in the HP 3235 command language.

Variables

Your HP 3235 employs two forms of numeric variables: simple variables (also called scalars) and subscripted arrays. Variable usage in the HP 3235 is very similar to variable usage in an enhanced BASIC language. The HP 3235 does not provide string variables.

All variables are global among control panel, HP-IB, and subroutine operations. This means that you can dynamically change variable values. For example, if you have a simple FOR ... NEXT loop in a subroutine, you can change the value of the loop counter variable from the keyboard or system controller while the subroutine is running.

Type Declarations

The HP 3235 uses two data types for its variables: INTEGER and REAL. All variables are REAL unless you declare them INTEGER. The valid range for REAL numbers is:

$-1.797\ 693\ 134\ 862\ 315 \times 10^{308}$ through $1.797\ 693\ 134\ 862\ 315 \times 10^{308}$

The smallest non-zero REAL value allowed is:

$\pm\ 2.225\ 073\ 858\ 507\ 202 \times 10^{-308}$

A REAL number can have a value of zero.

An INTEGER can have any whole-number value from:

-32,768 through +32,767

Declare arrays as either REAL or INTEGER. The **DIM** command declares array elements as REAL. For example:

REAL A, B, ARRAY(10)
DIM A(5)
INTEGER X,Y, Z(20)

The HP 3235 creates variables automatically when a variable name appears in an assignment statement or memory (**MEM**) statement:

LET A=*expression*
A=*expression*

```
MEM A
```

Some commands naturally expect certain types of variables. A **CLOSE** or **OPEN** command expects an integer number. The **WAIT** command expects a real number. Similarly, commands return certain types of variables. The **CLOSE?** command returns an integer number. Voltmeter measurements are always real (see **MEAS** command). All variables are REAL unless otherwise specified.

Programming Hint

Once an array type is declared, you cannot redeclare it as a different type without scratching memory first. See the **SCRATCH** command. If you execute a real number with a command that expects an integer, the HP 3235 rounds it to an integer. Likewise, if you send an integer with a command that expects a real number, the HP 3235 converts it. Therefore, you can minimize system overhead time by allocating variables according to their use. For example;

```
OUTPUT 709; "INTEGER HTRVALV; HTRVALV=104"
```

Type Conversions

The HP 3235 automatically converts between REAL and INTEGER values whenever necessary. When REAL numbers are converted to INTEGER representations, information may be lost. Two potential problem areas exist in this conversion, rounding errors and range errors.

When a REAL number is converted to an INTEGER, the REAL is rounded to the closest INTEGER value. All information to the right of the decimal point is lost.

Range errors exist when converting REAL numbers to integers. While REAL values range from approximately -10^{308} to $+10^{308}$, the INTEGER range is only from -32 768 to +32767 (approximately -10^{4} to $+10^{4}$). Obviously not all REAL numbers can be rounded to an equivalent INTEGER value. This problem can generate an Integer overflow error.

Using Variables

Simple variable and array names may be up to ten characters long. The first character must be a letter (A-Z) and the remaining nine may be either letters, numbers, the underscore character (_), or the question mark character (?). Upper case is the same as lower case. Variable names must not be the same as any HP 3235 command headers or parameters (i.e. **CLOSE**, **TIMER**, **KEYS**, **FAST**, etc.).

You can assign any numeric variable with the optional **LET** command. For example, the following three statements are equivalent:

```
LET HTRVALV=12
HTRVALV=12
HTRVALV = 3*4
```

Variables can replace numeric parameters in any HP 3235 command that uses numeric parameters.

Three example uses are: (1) channel addresses, (2) numeric data storage, and (3) in numeric calculations. This chapter discusses all three of these uses.

NOTE

Expressions used in parameter lists should be enclosed in parenthesis. For example:

```
CLOSE (SLOT*100+BANK)
```

Variables as Channel Addresses

For easy program debugging, or if you frequently select a specific channel, you can assign a channel address to a variable. Any command that uses a channel address also accepts a variable name. For example:

```
OUTPUT 709; "LET SOURCE1=104; CLOSE SOURCE1"
```

This has several advantages. For example, if you frequently move plug-in modules, you can assign the slot number to a variable, and calculate the channel numbers:

```
10 OUTPUT 709; "LET VALVES=200"
20 OUTPUT 709; "LET HTRVALV=VALVES+4"
30 OUTPUT 709; "LET CNTRLVLV=VALVES+7"
     .
     .
120 OUTPUT 709; "OPEN CNTRLVLV; CLOSE HTRVALV"
```

Thus, if you move the plug-in module, you only need to change one line in your program.

Another advantage of using internal variables for channel addresses is that you reduce system controller and HP 3235 overhead time. If you send the channel number with the command (i.e. **CLOSE 104**) or if you store channel assignments in your system controller (i.e. **OUTPUT 709; "CLOSE ";HTRVALV**) the controller uses a small amount of overhead time to encode the strings. The HP 3235 must also decode the strings. However, using internal variables reduces the overhead because the HP 3235 has already decoded the channel address.

Variables for Data Storage

Sometimes you want to store data in the local HP 3235 mainframe memory rather than send it to the HP-IB output buffer or display it on the control panel. The **MEM** command instructs the HP 3235 to send the next piece of data to the specified variable or array.

NOTE

If you specify a simple variable (scalar), **MEM** *turns off after it stores one result. Subsequent data is sent to either the HP-IB output buffer or the control panel display (wherever the command originated). If you specify an array, you must turn the* **MEM** *mode off (***MEM OFF***) when you finish storing data in the array. Otherwise, the HP 3235 will attempt to store any future data in the array.*

The following program shows how to store a voltmeter reading into a simple numeric variable:

```
10 OUTPUT 709; "REAL READING"        ! REAL VARIABLE READING
20 OUTPUT 709; "MEM READING"
30 OUTPUT 709; "MEAS 201"            !MEASURES CHANNEL 201
```

Numeric Calculations

Any variable, whether simple or array, can be used in numeric calculations. Several math functions are available in the HP 3235 command set that allow you to manipulate data. Math functions are described later in this chapter.

Reading Variable Values

The **FETCH** or **VREAD** commands return the value of a specified variable. For example, to read the value of PWRSUPP1, execute:

```
10 OUTPUT 709; "FETCH PWRSUPP1"
20 ENTER 709; A
30 PRINT "PWRSUPP1 is ";A
40 END
```

Arrays

You can allocate memory space for one-dimensional arrays (base 0) in the HP 3235. For real arrays, use either the **DIM** *name (size)* or **REAL** *name(size)* command to create the array. For integer arrays, use the **INTEGER** *name (size)* command. All arrays have a lower bound of 0 (zero). Arrays do not have a default size. To create a ten element array specify a size of 9. For example:

```
DIM TESTER (9)
```

Array names are subject to the same rules as numeric variable names. To specify an particular array element you must be followed by a subscript enclosed in parentheses. The range of subscripts is an integer from 0 through 32767, but the maximum array size is determined by available memory. A non-integer subscript is rounded to the nearest integer.

Arrays may be resized by redeclaring them. This initializes each element in the array to a value of zero (0). You cannot, however, redefine the type of array (**REAL** or **INTEGER**) without scratching memory first (see the **SCRATCH** command).

Array elements may be used in the same ways simple variables are used. You can use arrays to store channel lists as well as data.

Programming Hint

Real arrays use a large amount of memory. If you are storing channel lists or any other integer data, specify an integer array. Refer to the **MEMAVAIL** command in Chapter 11 for more information on efficient memory usage.

Filling Arrays

Array elements are initialized to 0 (zero) when they are declared (**DIM**, **REAL**, or **INTEGER**) or resized. Once you have dimensioned an array, use the **FILL** command to load your values into it. The **FILL** command has the following form:

FILL *name number, number, ... number*

For example, to fill an array with channel numbers:

```
10 OUTPUT 709; "INTEGER CHLIST(9)"
20 OUTPUT 709; "FILL CHLIST 103,231,115,116,117,201,101 205,112,110"
```

You can use a negative number to indicate a contiguous series of channel numbers:

```
10 OUTPUT 709; "INTEGER CHLIST(1)"
20 OUTPUT 709; "FILL CHLIST 103,-122" ! INCLUDES CHANNELS 103 THROUGH 122
```

Thus, in this two element array, we have specified all the channels between 103 and 122.

If you specify more data elements than you dimensioned the array for, a subscript out of bounds error occurs.

To fill an array with measurement data, such as from the multimeter, use the **MEM** command. The HP 3235 stores subsequent data in the specified array. You must specify **MEM OFF** as soon as you finish adding data to the array.

```
10 OUTPUT 709; "INTEGER CHLIST(9); REAL MEASDATA(9)"
20 OUTPUT 709; "FILL CHLIST 101,211,103,106,113,120,205,121,100,213"
30 OUTPUT 709; "MEM MEASDATA"
40 OUTPUT 709; "MEAS CHLIST"
50 OUTPUT 709; "MEM OFF"
```

NOTE

The optional start_index parameter to the **MEM** command allows you to specify the starting location for filling the array.

Array Size

The **SIZE?** query command returns the number of elements in the specified array. This is one more than the dimensioned (**DIM**, **REAL**, or **INTEGER**) maximum index in the array because of the base 0 indexing. Thus, if you dimension a ten element array, (**DIM CHLIST (9)**), **SIZE?** would return the number ten (10).

Reading Array Values

The **FETCH** and **VREAD** commands return the value of individual array elements. You must specify the subscript of the array element. Remember, arrays begin with element 0. For example, to read the value of the third element in CHLIST, execute:

```
10 OUTPUT 709; "VREAD CHLIST(2)"
```

```
20 ENTER 709; A
30 PRINT "Third element in CHLIST: ";A
```

Use the **VREAD** command to read the entire array. For example:

```
10 DIM Channel(9)
20 OUTPUT 709; "MEM OFF; VREAD CHLIST"
30 ENTER 709; Channel(*)
40 PRINT Channel(*)
50 END
```

Before using **VREAD** make sure the memory mode is turned off (**MEM OFF**). The array elements are separated by a carriage return/line feed (*crlf*).

Purging Arrays and Variables

All variables are stored in volatile memory. If the mainframe loses power, all variables are lost. The **SCRATCH** command also purges all variables, subroutines, and some stored states (explained in Chapter 11).

General Purpose Math

You can use general math expressions, following standard BASIC language conventions, from either the control panel keyboard, the system controller, or within subroutines. Available are the standard math operators, general math functions, trigonometric functions, and binary functions. The HP 3235 also has a simple calculator mode.

Math Operators

In addition to the usual math operators, +, -, *, /, and ^, two more arithmetic operators exist. These operators are **DIV** (integer division) and **MOD** (modulo). In general, they work just like other math operators. Unary minus operations should be written as:

```
A=0-B
```

Integer division (**DIV**) returns the integer portion of the quotient. In other words, normal division takes place but all digits to the right of the decimal point are truncated (not rounded).

The **MOD** (Modulo) operator returns the remainder resulting from a division. Like **DIV**, a normal division takes place, but **MOD** returns only the remainder. For instance, when you divide 7 by 3, the division result is 2 with a remainder of 1. **MOD** returns the 1 as a result of the division, **DIV** returns the 2.

```
10 OUTPUT 709; "LET A = 7 DIV 3; FETCH A"
20 ENTER 709; A                    ! RETURNS 2
30 PRINT A
40 OUTPUT 709; "LET A = 7 MOD 3; FETCH A"
50 ENTER 709;A                      ! RETURNS 1
60 PRINT A
70 END
```

Relational math operators (<, >, <=, >=, =, <>) and logical operators (**AND** and **OR**) are allowed in any expression.

General Math Functions

Table 7-1 explains the general math functions available in your HP 3235. The arguments, denoted by X and Y, may be numbers, numeric variables, functions, array elements, or numeric expressions in parenthesis.

Table 7-1. General Math Functions

Function and Argument	Meaning
ABS (X)	Absolute value of X.
SQR (X)	Positive square root of X.

Logarithmic Functions

The HP 3235 can compute both natural and common logarithms. The logarithmic functions are shown in Table 7-2.

Table 7-2. Logarithmic Functions

Function/ Argument	Meaning
LOG (X)	$Log_e(X)$; natural logarithm of a positive X to the base e.
EXP (X)	e^X; Natural antilogarithm. Raises e to the power of X.
LGT (X)	Log_{10}; common logarithm of a positive X to the base 10.

Trigonometric Functions

Three programmable trigonometric functions are provided in the HP 3235. They are shown in Table 7-3.

Table 7-3. Trigonometric Functions

Function and Argument	Meaning (X in radians)
SIN (X)	Sine of X.
COS (X)	Cosine of X.
ATN (X)	Arctangent of X.

Logical Functions

The HP 3235 has four logical functions: **AND** (inclusive-AND), **OR** (inclusive -OR), **EXOR** (exclusive-OR), and **NOT** (invert). The first three functions compare two arguments and return either a '1' or a '0' based upon the respective truth table. Any non-zero value (positive or negative) in an argument is considered a logic 1. Only zero is treated as logic 0. The commands have the form:

argument A **AND** *argument B*

argument A **OR** *argument B*

argument A **EXOR** *argument B*

NOT *argument*

The truth tables for the four functions are:

AND

A	B	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	A OR B
0	0	0
0	1	1
1	0	1
1	1	1

EXOR

A	B	A EXOR B
0	0	0
0	1	1
1	0	1
1	1	0

NOT

A	NOT A
0	1
1	0

Binary Functions

The HP 3235 provides seven binary functions. These can help in digital pattern generation. When using the binary functions, argument values (X and Y) of real variables are rounded to integers in the range of -32768 to +32767. The binary functions are shown in Table 7-4.

Table 7-4. Binary Functions

Function/Argument	Meaning
BINAND(X,Y)	Bit-by-bit logical AND of the arguments.
BINCMP(X)	Bit-by-bit binary complement of the argument.
BINEOR(X,Y)	Bit-by-bit logical Exclusive-OR of the arguments.
BINIOR(X,Y)	Bit-by-bit logical Inclusive-OR of the arguments.
BIT(X,position)	Returns "0" or "1" representing the logic value of the specified bit of the argument. The bit position is in the range 0 (lsb) to 15 (msb).
ROTATE(X,displacement)	Returns an integer obtained by rotating the argument a specified number of positions with bit wraparound.*
SHIFT(X,displacement)	Returns an integer obtained by shifting the argument a specified number of positions without bit wraparound.*

* If the displacement is positive, rotating or shifting is toward the least significant bit. If the displacement is negative, rotating or shifting is toward the most significant bit.

Math Hierarchy

The HP 3235 evaluates parenthetical expressions before evaluating any math functions outside of parentheses. If two or more operations of the same priority are in the expression, the hierarchy is from left to right.

Table 7-5. Math Hierarchy

Highest Priority	Parentheses
	Functions: SIN, COS, etc.
	Exponentiation
	*, /, MOD, DIV,
	+, -
Lowest Priority	Relational operators: <, >, <=, >=, etc. and logical operators: AND, OR, etc.

Math Errors

When evaluating a math expression, the following errors may occur. The HP 3235 treats math errors just like any other execution errors. Refer to Chapter 4 for information on handling errors.

Table 7-6. Math Errors

Error Description
Division By Zero
Real Overflow
Real Underflow
Integer Overflow
Square Root of a Negative Number
Log of a Non-Positive Number
Illegal Real Number
Trig Argument Out of Range
BCD Exponent Too Big
HEX, Octal, or Decimal Argument Error

Making Comparisons Work

If you are comparing integer numbers, no special precautions are necessary. However, if you are comparing REAL values, especially those which are the results of calculations, it is possible to run into problems due to rounding and other limitations inherent in the system. For example, consider the use of comparison operators in **IF...THEN** statements to check for equality in any situation resembling the following:

```
10  OUTPUT 709; "SUB TESTER"
20  OUTPUT 709; "A=25.3765477"
30  OUTPUT 709; "IF SIN(A)^2 + COS(A)^2 = 1 THEN"
40  OUTPUT 709; "DISP 'EQUAL'"
50  OUTPUT 709; "ELSE"
60  OUTPUT 709; "DISP 'NOT EQUAL'"
70  OUTPUT 709; "END IF"
80  OUTPUT 709; "SUBEND"
90  OUTPUT 709; "CALL TESTER"
100 END
```

You may find that the equality test fails due to rounding errors or other errors caused by the inherent limitations of finite machines. A repeating decimal or irrational number cannot be represented exactly in any finite machine like the HP 3235.

A good example of equality error occurs when multiplying or dividing data values. A product of two non-integer values nearly always results in more digits to the right of the decimal point than exists in either of the two numbers being multiplied. When you are comparing two numbers, as with the **LIMIT** or **VERIFY** commands, always leave a sufficient error margin for the test limits.

Limit Testing

One additional math function, the **LIMIT** command, compares a set of test values (multimeter readings, for instance) to lower limit and upper limit variables or arrays. This allows fast pass/fail testing within the HP 3235. The test values must be stored in an array. Each test value is compared to corresponding elements in the upper and lower limit arrays. You must fill the upper and lower limit arrays prior to executing **LIMIT**. The **LIMIT** command returns a "0" if all the comparisons pass or a "1" if any comparisons fail.

It also sets the Limit Fail bit (bit 10) in the status register if the test fails. The **LIMIT** command does not clear the status register bit if the test passes. Therefore, you should always clear the status register (see Chapter 10) before beginning the **LIMIT** test.

The following program shows how to use **LIMIT**:

```
10  OUTPUT 709; "CLR"      ! CLEAR THE STATUS REGISTER
20  OUTPUT 709; "REAL UPLIMIT(4), LOWLIMIT(4), MEASDATA(4)"
30  OUTPUT 709; "INTEGER CHLIST(4)"
40  OUTPUT 709; "FILL UPLIMIT 5.1,5.1,12.2,-7.05,12.2"
50  OUTPUT 709; "FILL LOWLIMIT 4.9,4.9,11.9,-6.95,11.9"
60  OUTPUT 709; "FILL CHLIST 122,113,105,101,107"
70  OUTPUT 709; "MEM MEASDATA"
80  OUTPUT 709; "MEAS CHLIST"
90  OUTPUT 709; "MEM OFF"
100 OUTPUT 709; "LIMIT MEASDATA,LOWLIMIT,UPLIMIT"
110 ENTER 709;A
120 IF A=0 THEN PRINT "TEST PASSED"
130 IF A=1 THEN PRINT "TEST FAILED"
140 END
```

If the test failed, you can determine which readings failed by testing them individually. Replace lines 100 through 140 of the previous program with the following lines to test the readings individually.

```
100 FOR I=0 TO 4
110   OUTPUT 709; "LIMIT MEASDATA(";I;"),LOWLIMIT(";I";"),UPLIMIT(";I;")"
120   ENTER 709; A
130   IF A=0 THEN
140     PRINT "READING ";I;" PASSED"
150   ELSE
160     PRINT "READING ";I;" FAILED"
170   END IF
180 NEXT I
190 END
```

This could also be done in a subroutine. Refer to Chapter 8 for information.

Chapter 8

Subroutines

Introduction

The HP 3235 can store and execute (on command) BASIC language subroutines. These subroutines can either be down-loaded into the HP 3235 mainframe memory from a remote controller (such as one of the HP Series 200/300 computers), or you can enter from the control panel keyboard. This chapter acquaints you with the structure and usage of subroutines. It also discusses specific commands that are used only with subroutines.

Subroutines, down-loaded into the HP 3235 prior to making measurements and then called later with a single command from the system controller, allow the controller to perform other tasks while the HP 3235 is busy with some other activity such as scanning. This provides a multi-tasking capability to your remote controller because the HP 3235 controller module is acting like a separate computer running a task all by itself. Also, commands in an HP 3235 subroutine execute faster than those same commands received over the HP-IB because of the way the HP 3235 stores the subroutine commands internally.

Subroutine Definition

A subroutine is a series of HP 3235 commands beginning with a **SUB** *name* command and ending with a **SUBEND** command. The **SUB** *name* command assigns a name to the subroutine which you can execute later. Subroutines are stored in the HP 3235's volatile memory.

What commands are allowed inside a Subroutine?

Any HP 3235 command may be stored and executed inside a subroutine. Three conditional and looping statements are provided for use only within subroutines.

How many different Subroutines can be stored?

The exact number of subroutines that can be stored depends on the individual sizes of the subroutines. A typical short subroutine of 10 commands (including the **SUB** *name* and **SUBEND** commands) might average about 600 bytes. Refer to Chapter 11 for information on memory useage.

Can I nest Subroutines?

Yes! Nesting subroutines is the ability to have one subroutine call (execute) another subroutine. You can nest up to 10 subroutines.

Writing And Loading Subroutines

The subroutine program examples in this chapter illustrate simple operations of the HP 3235 that you can copy and use in more complex mainline programs of your own design.

Programming Hint

You should execute the **SCRATCH** command and down-load the subroutines and channel lists from your system controller at the beginning of your test system program. This helps memory management for the HP 3235 and ensures the subroutines are down-loaded and ready when they are needed.

Creating the Subroutine

Executing the **SUB** *name* command instructs the HP 3235 to store all subsequent commands, until the **SUBEND** command, in a subroutine.

Subroutine names may be from 1 to 10 characters long. The first character must be a letter (A-Z) and the remaining nine may be either letters, numbers, the underscore character (_), or the question mark (?). You can use either upper or lower case but the HP 3235 converts all alpha characters to upper case. Subroutine names must not be the same as any HP 3235 command header (i.e. **CLOSE**, **TIMER**, **RUN**, etc.), parameter (i.e. **FAST**) or variable name.

NOTE

Names are created by the **MEM**, **DIM**, **REAL**, and **INTEGER** commands, subroutines, stored states, and variable assignments (i.e. **LET a=301**). Once a name is defined, it cannot be redefined to a different function until the memory is scratched (**SCRATCH** command).

The following example shows how to create a simple subroutine:

```
10 OUTPUT 709; "SUB CLSCHAN"
20 OUTPUT 709; "OPEN 103;CLOSE 113;MEAS 213"
30 OUTPUT 709; "SUBEND"
```

The two statements, **SUB** *CLSCHAN* and **SUBEND**, along with the three commands on line 20 form the subroutine named *CLSCHAN*.

Although you can enter subroutines from the control panel, we do not recommend it because you cannot edit them from the control panel.

When a subroutine statement is entered, the HP 3235 checks it for syntax errors, just like normal commands. If the syntax is not correct, an error occurs and that command is not stored. You should edit your subroutine in the system controller and down-load it again. The HP 3235 stores the subroutine in volatile memory.

You can then execute the subroutine from either the control panel keyboard or from the system controller. The subroutine will not be stored if a subroutine nesting error exists when the **SUBEND** command is executed (e.g. a called subroutine does not exist in the HP 3235's memory).

If you create or down-load a subroutine using a name that already exits for a subroutine resident in memory, the new subroutine overwrites the old subroutine.

Editing Subroutines

The control panel keyboard has no facility to edit subroutines. Therefore, you should create your subroutines on your system controller and store them off-line. In this way they are always available for down-loading should you accidentally purge the subroutine. You can also edit the subroutine at the system controller for easy updating and storage.

Purging Subroutines

Subroutines, variables and arrays are destroyed from memory when power is removed or when certain commands are executed. Therefore, we recommend that you write all of your non-trivial subroutines using your system controller and then store them on disc or tape for later use and for easy revision. Please note that the **RST** and **CLR** commands do not purge subroutines.

The **DELSUB** command purges individual subroutines although the subroutine name still appears in the **CAT** (catalog) list.

Two conditions will destroy all existing subroutines, variables, and arrays from mainframe memory:

- Removing power from the HP 3235 mainframe.
- Executing a **SCRATCH** command.

Command Types

Subroutine related commands are used *only* with subroutines. Definition/deletion commands deal with the storage, viewing, and deletion of subroutines from mainframe memory. Execution commands control execution of subroutines from inside or outside a subroutine.

Definition/ Deletion Commands

Subroutine definition/deletion commands identify the beginning and end of subroutines, store and delete subroutines from memory, and list the subroutines presently stored in mainframe memory.

The syntax formats for the subroutine definition/deletion commands are as follows:

SUB *name*

SUBEND

DELSUB *name*

SCRATCH

CAT

LIST *name*

COMPRESS *name*

SUB/SUBEND

Every HP 3235 subroutine must contain the two commands: **SUB** *name* and **SUBEND**.

The **SUB** *name* command must be the first line of all HP 3235 subroutines. It identifies where the subroutine begins and assigns the name to the subroutine. It also causes the HP 3235 to begin storing the subroutine in its memory.

The **SUBEND** command must be the last line of all subroutines. It identifies where the subroutine ends and also terminates the entry of the subroutine. Commands listed between the **SUB** *name* and the **SUBEND** command are executed, in order, every time the subroutine is executed.

Only one **SUB** *name* and one **SUBEND** command is allowed in any one subroutine. Additional **SUB** *name* or **SUBEND** commands generate errors.

DELSUB

The **DELSUB** *name* command deletes *only* the specified subroutine from mainframe memory. It does *not* delete the subroutine name itself from the catalog of subroutines.

SCRATCH The **SCRATCH** command deletes (scratches) *all* HP 3235 subroutines, variables, and arrays from mainframe memory. It also deletes all name definitions from the catalog. **SCRATCH** gives an error but does not purge the memory if it is executed while a subroutine is executing.

CAT The **CAT** command (catalog) lists the names of all HP 3235 subroutines, simple variables, stored states, and arrays that are presently stored in mainframe memory. **CAT** also displays the number of bytes required to store each subroutine, and the type and number of elements in each array. If a subroutine has been defined, and then deleted using **DELSUB**, it is listed with a size of zero. If there are no more arrays or subroutines to be listed, the **CAT** command returns the word "DONE". Refer to Chapter 11 for information on stored states. The format for the catalog is:

For Subroutines:	SUB *sub_name* SIZE *dddd* TEXT SIZE *dddd*	
For Integer Arrays:	IARRAY *array_name* SIZE *dddd*	
For Real Arrays:	RARRAY *array_name* SIZE *dddd*	
For Stored States:	CONT *state_name*	(nonvolatile memory)
	STATE *state_name*	(volatile memory)
For Simple Variables:	INT *variable_name*	
	REAL *variable_name*	

The **CAT** command returns two "size" values for subroutines. The first value is the size, in bytes, of the subroutine's source code. This is the HP 3235 machine code version of the subroutine. The second value is the size in bytes, of the subroutine text code which you can list and single-step. When you compress a subroutine (using the **COMPRESS** command), the TEXT SIZE goes to "0" but the machine code SIZE remains the same. The size value returned for arrays is the actual number of elements in the array. The following program shows how to use the **CAT** command.

```
10  DIM A$[80]
20  OUTPUT 709; "CAT"
30  REPEAT
40    ENTER 709; A$
50    PRINT A$
60  UNTIL A$ = "DONE"
70  END
```

LIST The **LIST** *name* command allows you to list the specified subroutine. You can list the subroutine either to the control panel display (execute **LIST** *name* followed by the STEP key) or to your system controller. Remember, you cannot edit the subroutine from the control panel; you must edit it at your system controller. The following program shows how to list the subroutine *CLSCHAN* to your system controller.

```
10  DIM A$[100]
20  OUTPUT 709; "LIST CLSCHAN"
30  REPEAT
40    ENTER 709; A$
50    PRINT A$
60  UNTIL A$ = "SUBEND ;"
70  END
```

COMPRESS

The **COMPRESS** *name* command leaves the subroutine in its machine code form. This saves internal memory but eliminates the ability to list or single-step the subroutine.

Execution Commands

Subroutine execution commands control the execution of a subroutine. The HP 3235 has six execution commands:

CALL *name*

RUN *name*

PAUSE

CONT

STEP [*name*]

ABORT

Subroutine CALL

The **CALL** *name* command executes the named subroutine and waits for it to finish. This means that no subsequent commands are accepted (either from the HP-IB interface or the control panel keyboard) until the subroutine finishes. The READY Bit (Bit 4 in the Status Register) is clear (0) while the subroutine is running. When the subroutine finishes execution, the READY Bit is set and the mainframe is ready to receive additional commands.

The **CALL** *name* command may also be used inside a subroutine to call another subroutine. This provides the expanded capability of "nested" subroutines. You may nest up to 10 subroutines.

If you execute the **USE** or **MEM** command inside of a **CALL** executed subroutine, the HP 3235 retains that assignment after the subroutine finishes execution.

Subroutine RUN

The **RUN** *name* command executes the named subroutine in parallel with other commands. This means that the mainframe executes the subroutine as it finds time between executing other commands. Commands executed from the keyboard or from the system controller temporarily interrupt execution of the subroutine. When you **RUN** a subroutine, the HP 3235 sets the READY Bit (bit 4 in the Status Register) indicating that the mainframe is ready to accept and execute more commands. Do not use the **RUN** command to nest subroutines.

If you execute the **USE** command inside of a **RUN** subroutine, the HP 3235 retains that assignment only during the execution of the subroutine. After the subroutine finishes execution, the assignment reverts back to its value before the subroutine was run.

Knowing When a Subroutine is Running

The **RUNNING?** query command returns a "1" if the subroutine is currently running or paused (see "Subroutine PAUSE") or a "0" if the subroutine is not running. The following program shows how to use the **RUNNING?** command.

```
10  OUTPUT 709; "RUN CLSCHAN"
20  OUTPUT 709; "RUNNING?"
30  ENTER 709; A
40  IF A=1 THEN PRINT "SUBROUTINE IS RUNNING"
50  IF A=0 THEN PRINT "SUBROUTINE IS NOT RUNNING"
60  END
```

Subroutine PAUSE

You can cause a **RUN** subroutine to pause execution with the **PAUSE** command. Once a subroutine has been paused you must execute a **CONT** command to resume execution. **CONT** lets the subroutine continue running to completion, starting from the next command after the **PAUSE** command.

NOTE

A **PAUSE** *command embedded within a* **CALL** *executed subroutine pauses a* **RUN** *executed subroutine (if one is running). You cannot pause a* **CALL** *executed subroutine.*

After pausing a subroutine, you can also STEP through it (using the **STEP** command or the control panel **Single Step** key) line by line. Executing the **CONT** command resumes subroutine execution.

The HP 3235 generates an error if you try to execute either **CONT** or **STEP** when a **RUN** executed subroutine is not paused or doesn't exist.

Knowing When a Subroutine is Paused

The **PAUSED?** query command returns a "1" if the subroutine is currently paused or a "0" if the subroutine is paused. The following program shows how to use the **PAUSED?** command.

```
10  OUTPUT 709; "RUN CLSCHAN;PAUSE"
20  OUTPUT 709; "PAUSED?"
30  ENTER 709; A
40  IF A=1 THEN PRINT "SUBROUTINE IS PAUSED"
50  IF A=0 THEN PRINT "SUBROUTINE IS NOT PAUSED"
60  END
```

Stepping Through a Subroutine

The **STEP** *name* command allows you to step through your subroutine, line by line, to verify its operation. After you execute **STEP** *name*, use the **STEP** command (or press the control panel **Single Step** key) to sequence through the subroutine. Each time you execute **STEP**, the next subroutine line is displayed and executed. If the line generates output data, such as a multimeter measurement, the data is displayed. You can execute the **CONT** command to cease stepping the subroutine and cause it to execute normally.

Aborting a Subroutine

The HP-IB CLEAR command (see Appendix D) aborts execution of a **CALL** executed subroutine and returns control to the HP-IB command input buffer or the control panel.

ABORT halts execution of any **RUN** executed subroutine and returns control to the HP-IB command input buffer or keyboard. **ABORT** can only stop an executing subroutine at the end of a subroutine statement. If the subroutine statement takes a long time to complete (e.g., **WAIT 100**), the **ABORT** command may time-out and generate an error message. Either press the Control Panel Device Clear key or execute an HP-IB Device Clear to regain control. **ABORT** cancels any nested subroutines. **ABORT** does nothing if no subroutine is running.

Exiting a Subroutine

A subroutine executes until it reaches the **SUBEND** command. Control then reverts back to either the subroutine that called it (nested subroutines) or to the HP-IB input buffer or control panel keyboard (whichever executed the subroutine). The **RETURN** command can also cause the subroutine to end. For example, if you want to have a conditional termination of the subroutine, place **RETURN** inside an **IF...THEN** construct (discussed later in this chapter). The **RETURN** statement returns control to the caller without executing the **SUBEND** statement.

Nesting Subroutines

One subroutine may call a second (nested) subroutine for execution before the first subroutine finishes execution. When the second subroutine executes the **SUBEND** statement, the first subroutine continues with the next command following the embedded **CALL** *name* command. Do not use the **RUN** command to nest subroutines.

Every subroutine must have the **SUBEND** statement. However, you can use the **RETURN** command as an early way of exiting a nested subroutine. For example:

```
10  OUTPUT 709; "SUB CHLMEAS"
20  OUTPUT 709; "MEM A"
30  OUTPUT 709; "MEAS 113"
40  OUTPUT 709; "MEM OFF"
50  OUTPUT 709; "IF A < 5.05 THEN; RETURN"
60  OUTPUT 709; "ELSE"
70  OUTPUT 709; "CLOSE 201"
80  OUTPUT 709; "END IF"
90  OUTPUT 709; "SUBEND"
100 END
```

The HP 3235 has two requirements for nesting subroutines. First, the subroutine called from within another subroutine must be stored in mainframe memory before the subroutine doing the calling is stored. This is because the HP 3235 checks the syntax of each command as it stores the subroutine. When it encounters an embedded **CALL** *name* command, it checks to see if a subroutine by that name exists in memory. If not, it generates an error. Second, subroutines may not be nested more than 10 levels deep. You cannot put one subroutine inside of another subroutine. For example, the following program will result in an error.

```
10  OUTPUT 709; "SUB CLSCHAN"
20  OUTPUT 709; "OPEN 321;CLOSE 123"
30  OUTPUT 709; "SUB OPCHAN"          ! THIS RESULTS IN AN ERROR
40  OUTPUT 709; "OPEN 232;CLOSE 212"
50  OUTPUT 709; "SUBEND"
60  END
```

Conditional and Looping Statements in Subroutines

Your HP 3235 provides three BASIC language constructs for conditional branching and looping. Use these constructs only within the context of a subroutine. Conditional branching and looping constructs provide for repetitive tests, initializing arrays, etc.

The three constructs are: **FOR...NEXT** and **WHILE...END WHILE** loops, and **IF...THEN...ELSE** conditional branches. These constructs are similar to those used in an enhanced BASIC language. The only exception is that HP 3235 subroutines do not have line numbers or GOTO statements for branching. Looping and conditional branch structures may be nested seven deep.

Details on using the conditional branching and loops are in the HP 3235 Language Reference Manual.

FOR...NEXT Loops

If you need to repeatedly execute a series of statements, use a **FOR...NEXT** loop. The complete syntax for the **FOR...NEXT** loop is:

FOR *loop counter = initial value* **TO** *final value* **[STEP** *size***]**

Subroutine Segment

NEXT *loop counter*

The loop counter can be any simple variable. The *initial* and *final value* may be either numbers, numeric variables or mathematical expressions. The optional *step size* may also be any number (positive or negative), numeric variable, or expression. If, when the **FOR** statement is executed, the loop counter has a value greater than the *final value*, the loop is skipped.

The *Subroutine Segment* repeatedly executes until the loop counter exceeds the final value. The following example shows a simple **FOR...NEXT** loop:

```
10 OUTPUT 709; "REAL READINGS (7)"
20 OUTPUT 709; "SUB CLSCAN"
30 OUTPUT 709; "MEM READINGS"
40 OUTPUT 709; "FOR A = 1 TO 8"
50 OUTPUT 709; "MEAS (100+A)"
60 OUTPUT 709; "NEXT A"
70 OUTPUT 709; "MEM OFF"
80 OUTPUT 709; "SUBEND"
90 END
```

WHILE... END WHILE Loops

WHILE...END WHILE loops also provide repeated execution of a subroutine segment while the expression is true. The syntax for **WHILE...END WHILE** loops is:

WHILE *expression*

Subroutine Segment

END WHILE

As long as the expression evaluates to true (not zero) then the *Subroutine Segment* executes repeatedly. When the expression evaluates to false (zero), then subroutine execution continues with the statement after the **END WHILE** statement.

The following program demonstrates a simple **WHILE...END WHILE** loop:

```
10 OUTPUT 709; "REAL READINGS (7)"
20 OUTPUT 709; "SUB CLSCAN"
30 OUTPUT 709; "A=1"
40 OUTPUT 709; "MEM READINGS"
50 OUTPUT 709; "WHILE A<=8"
60 OUTPUT 709; "MEAS (100+A)"
70 OUTPUT 709; "A=A+1"
80 OUTPUT 709; "END WHILE"
90 OUTPUT 709; "MEM OFF"
100 OUTPUT 709; "SUBEND"
110 END
```

IF...THEN Branching

The **IF...THEN** construct provides conditional branching within subroutines. The complete syntax for **IF...THEN** constructs is:

IF *expression* **THEN**

Subroutine Segment

[ELSE]

[Subroutine Segment]

END IF

If the original expression (in the **IF...THEN** statement) evaluates to true (zero), then the first *subroutine segment* is executed. If the expression evaluates to false (zero) then subroutine execution continues with the subroutine segment after the **ELSE** statement or, if you haven't used the **ELSE** statement, then subroutine execution continues after the **END IF** statement.

The following example shows a simple **IF...THEN** construct:

```
10 OUTPUT 709; "REAL READINGS (7)
20 OUTPUT 709; "SUB CLSCAN"
30 OUTPUT 709; "A=1"
40 OUTPUT 709; "MEM READINGS"
50 OUTPUT 709; "IF A<8 THEN"
60 OUTPUT 709; "MEAS (100+A)"
70 OUTPUT 709; "A=A+1"
80 OUTPUT 709; "END IF"
90 OUTPUT 709; "MEM OFF"
100 OUTPUT 709; "SUBEND"
110 END
```

Chapter 9

Trigger Buses and System Timing

Introduction

This chapter describes the trigger bus system and how to use it. The trigger bus system has many uses including synchronizing one or more HP 34520 Multimeters to some event (e.g. external, timer, or software), indicating an event interrupt output from the HP 34522 Digital I/O module, and passing digital pulses through either the HP 34504 Coaxial Multiplexer or the HP 34506 Coaxial Matrix Modules.

This chapter also describes the two timers in the HP 3235. These timers can generate square waves or pulses and count-down from preset values and interrupt the system controller when the counter underflows.

The Trigger Bus System

The trigger bus system consists of three circuits: (1) two bidirectional, TTL level trigger buses inside each frame's card cage (called Backplane Trigger Buses), (2) a pair of trigger lines connecting the mainframe to every extender (called Extender Trigger Buses), and (3) a network for isolating/switching different sources and destinations for the trigger bus signals (each frame). within the plug-in module cage, the trigger bus lines are referenced to Guard.

Figure 9-1 is a simplified schematic of the trigger bus system within the mainframe. Extender frames are identical but do not have the timer channels, External In and Out BNCs, and the CPU. The isolation circuitry allows only one signal from the switching network to drive one of the backplane trigger buses and only one backplane trigger bus to drive a signal into the switching network.

In general you need to specify one source and at least one destination for the trigger bus signals. For example, you might specify the External Trigger In BNC (source) to drive Trigger Bus 0 and the HP 34520 Multimeter (destination) to receive its measurement trigger from Trigger Bus 0.

Anytime you specify a source for driving a trigger bus, the HP 3235 automatically cancels any previous source designation for that same trigger bus. You can have any number of devices receive the trigger bus signals.

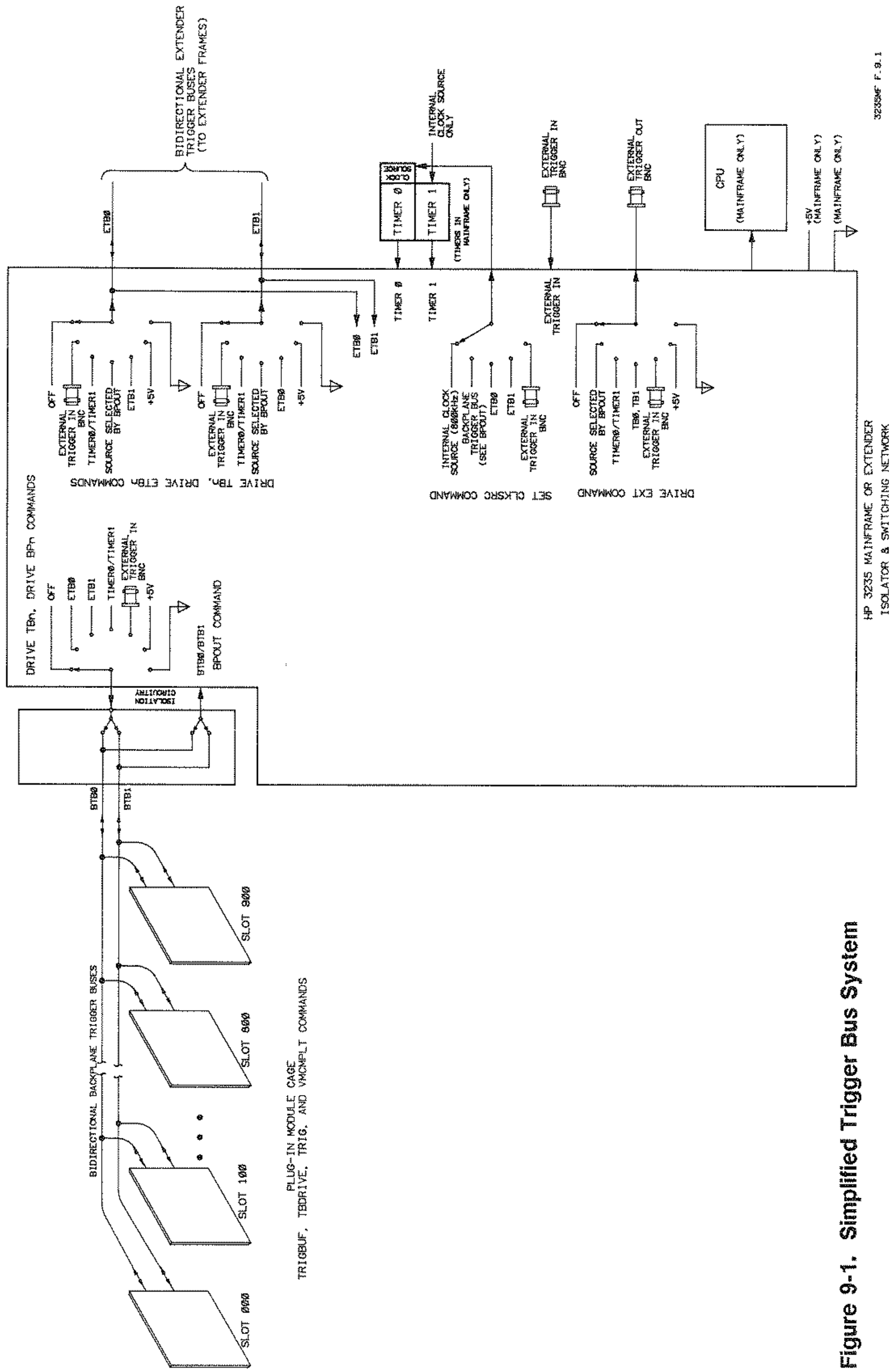


Figure 9-1. Simplified Trigger Bus System

NOTE

At power-on, both trigger buses are configured to be continuous throughout the HP 3235 system. A selected source will drive the trigger buses in all frames connected via the extender system. Refer to the **AUTOTB** *command later in this chapter to isolate trigger buses between frames.*

You can use the trigger bus system in three distinct ways:

- Using the backplane trigger buses to pass signals to/from plug-in modules.
- Driving the trigger buses from other sources (external instruments, timers, HP-IB, etc.).
- Isolating the trigger buses between extender frames.

Passing Trigger Signals from one Plug-in Module to Another

Within each frame's card cage the trigger buses can pass trigger signals from one plug-in module to another plug-in module. Five plug-in modules can both drive the trigger buses and receive signals from the trigger buses. The five modules are:

- HP 34504 Switched Shield Coaxial Multiplexer Module
- HP 34506 Switched Shield Coaxial Matrix Module
- HP 34520 Multimeter Module
- HP 34522 Digital I/O Module
- HP 34523 Breadboard Module

Using the Coaxial Modules

Refer to Figure 9-2. Both the HP 34504 Coaxial Multiplexer and 34506 Coaxial Matrix Modules can input external signals to the trigger buses or output signals from the trigger buses to external circuitry. The HP 34506 Coaxial Matrix Module has two bidirectional buffers (one for each trigger bus) independent of the matrix switches. The HP 34504 Coaxial Multiplexer also has two bidirectional buffers (one for each trigger bus) but the trigger signals must route through the multiplexer switches.

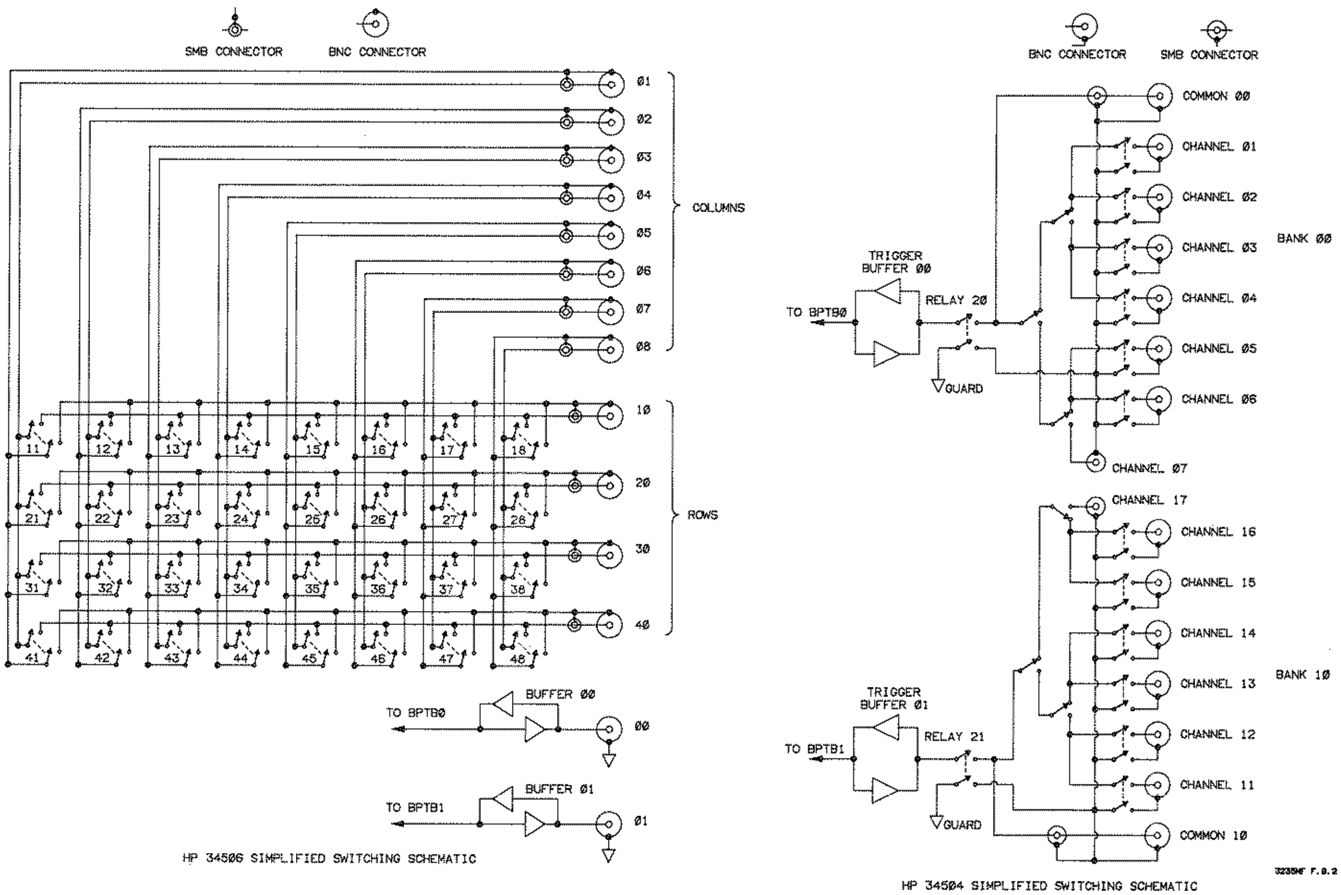


Figure 9-2. HP 34504 and 34506 Simplified Schematics

The **TRIGBUF** command specifies the buffer and the mode (input, output, or off) on the specified module. **TRIGBUF** has the form:

TRIGBUF *buffer#,mode*

The buffer number has the form *esnn* where *es* represents the frame (mainframe or extender) and slot number respectively. The *nn* represents the actual buffer number on the module where "00" represents Trigger Bus 0, and "01" represents Trigger Bus 1.

Table 9-1 lists the three choices for the *mode* parameter.

Table 9-1. TRIGBUF *mode* Choices

mode Parameter	Definition
OFF	Turn Off the Specified Trigger Buffer.
IN	Trigger Buffer Specified Inputs Signal from External Circuitry to Trigger Bus.
OUT	Trigger Buffer Specified Outputs Signal from Trigger Bus to External Circuitry.

The following program example shows how to input a trigger signal from channel 13 of an HP 34504 Coaxial Multiplexer Module installed in slot 700, pass it through Trigger Bus 1, and output the signal through an HP 34506 Coaxial Matrix Module in slot 200.

```
10 OUTPUT 709; "RST"                ! RESET THE MAINFRAME
20 OUTPUT 709; "CLOSE 713"          ! CLOSE CHANNEL 713
30 OUTPUT 709; "TRIGBUF 0701,IN"    ! SLOT 700, BUFFER/TRIGGER BUS 01
35                                  ! INPUT SIGNAL
40 OUTPUT 709; "TRIGBUF 0201,OUT"   ! SLOT 200, BUFFER/TRIGGER BUS 01
45                                  ! OUTPUT SIGNAL
50 END
```

NOTE

*Transients, caused by opening or closing relays or setting the HP 34504 or 34506 Coaxial Module trigger bus buffers, can cause false trigger signals on the trigger bus lines. Therefore, a good programming practice is to open/close any relays on the HP 34504 Switched-Shield Coaxial Multiplexer Module and enable the trigger buffer (***TRIGBUF*** command) for either coaxial module prior to designating the trigger destination. For example, if you are using a trigger signal to trigger the HP 34520 Multimeter, set-up the entire trigger bus system before arming the multimeter triggering.*

Refer to the individual plug-in module Programming and Configuration manuals for additional information on using these modules with the trigger buses.

Using the Multimeter Module

The HP 34520 Multimeter can receive trigger signals from the trigger buses or drive the trigger buses with a voltmeter complete signal. Refer to the HP 34520 Programming and Configuration Manual for additional information. The multimeter must be specified as the current *use device* with the **USE** command. For example, if the multimeter is in slot 800, execute:

```
OUTPUT 709; "USE 800"
```

Triggering the Multimeter

The HP 34520 Multimeter can receive its measurement trigger or its sample pacing signal (see the **TRIG** and **NRDGS** commands in the HP 34520 Multimeter Manual) from either trigger bus. For example, to specify Trigger Bus 1 as the source for the measurement trigger, execute:

```
OUTPUT 709; "TRIG TB1"
```

Voltmeter Complete

Voltmeter complete signals the end of one multimeter reading. The **VMCMPLT** command specifies the destination for the voltmeter complete signal and has the following syntax:

VMCMPLT *destination*

You can specify either **NONE** (or **OFF**), **TB0**, or **TB1** as the destination. For example, to send the voltmeter complete signal through BPTB0 to a trigger bus output buffer on an HP 34506 Coaxial Matrix module in slot 400, execute the following:

```
OUTPUT 709; "USE 800; VMCMPLT TB0; TRIGBUF 400,OUT"
```

Using the Digital I/O Module

The HP 34522 Digital I/O Module can either drive the trigger buses or receive a second rank trigger signal (buffered transfer only) from the buses. Refer to the HP 34522 Programming and Configuration Manual for additional information. The Digital I/O Module must be specified as the current *use device* with the **USE** command. For example, if the module is in slot 300, execute:

```
OUTPUT 709; "USE 300"
```

Driving the Trigger Bus

The **TBDRIVE** command selects the source on the Digital I/O Module which is to drive the specified trigger bus and has the following syntax:

TBDRIVE TB*n*, *source*

The **TB*n*** parameter selects which trigger bus (either TB0 or TB1) the module drives. Table 9-2 lists the eight choices for the ***source*** parameter.

Table 9-2. TBDRIVE *source* Choices

source Parameter	Description
OFF	Turn Off the Trigger Bus Driver.
SRTRIG	Drive from the Second Rank Clock Signal.
SRACK	Drive from the Second Rank Acknowledge Signal.
EXT	Drive from the Digital I/O External Trigger.
EXTBAR	Drive from the Digital I/O Inverted External Trigger.
EVENT	Drive from the Event Sense Interrupt.
LOW	Force the Trigger Bus to its LOW State (0 Volts).
HIGH	Force the Trigger Bus to its HIGH State (+5 Volts).

The following example shows how to drive Backplane Trigger Bus 0 from the Digital I/O module's (slot 300) external trigger input.

```
OUTPUT 709; "USE 300; TBDRIVE TB0,EXT"
```

Driving the Digital I/O Module

The **SRTRIG** command specifies the trigger source (second rank trigger) for buffered transfers from the Digital I/O Module. Two of the eight possible sources for this trigger are the Backplane Trigger Buses. Refer to the HP 34522 Digital I/O Manual for additional information.

Using the Breadboard Module

The HP 34523 Breadboard Module has the capability to have two bidirectional trigger bus buffers (one for each trigger bus) for receiving or sending trigger bus signals. The **TRIGBUF** command controls these two buffers. Refer to the HP 34523 Breadboard Manual for information.

Other Trigger Bus Sources and Destinations

Figure 9-1 showed several potential sources/destinations for the trigger bus system. These sources/destinations include the External Trigger In and Out BNC connectors, the timers, and the HP 3235 CPU. This section of the chapter describes how to use the trigger buses with other sources and destinations.

NOTE

Sources other than the Plug-in modules can drive only one of the backplane trigger buses at a time. This is because the isolation circuitry (see Figure 9-1) provides only one signal path in each direction. For example, if the external Trigger In BNC is driving Trigger Bus 0, then Trigger Bus 1 must either be driven by one plug-in module or not used.

Driving the Trigger Buses

The **DRIVETB*n*** command specifies the source (other than the plug-in modules) to drive the backplane trigger buses in all frames from the mainframe. It has the form:

DRIVETB*n* *source*

Table 9-3 shows the possible sources for driving the trigger buses.

Table 9-3. DRIVETB*n* sources

Source Parameter	Description
OFF	Disable the trigger bus drivers
EXTIN	Drive the trigger bus from the External Trigger In BNC
TIMER0	Drive the trigger bus from Timer 0
TIMER1	Drive the trigger bus from Timer 1
LOW	Force the trigger bus LOW (0 volts)
HIGH	Force the trigger bus HIGH (+5 volts)
TRG	Pulse the trigger bus (LOW then HIGH) when the HP 3235 receives either the TRG command or an HP-IB Group Execute Trigger

The following paragraphs describe these sources.

External Trigger In BNC

Signals input through the External Trigger In BNC can drive either backplane trigger bus. Trigger signals should be TTL levels. To source a trigger signal from the External Trigger In BNC connector through Trigger Bus 1 and trigger the HP 34520 Multimeter in slot 800, execute:

```
10 OUTPUT 709; "DRIVETB1 EXTIN"     ! DRIVE TRIGGER BUS EXTERNALLY
20 OUTPUT 709; "USE 800; TRIG TB1"  ! MULTIMETER TRIGGER SOURCE
30 END
```

TIMER0/ TIMER1

The HP 3235 has two timers which can drive either trigger bus. The timers can either generate single pulses or square waves. Refer to the section on "Using the Timers" later in this chapter for more information. The following example shows how to pulse trigger bus 0 for 1 mS using Timer 1.

```
10 OUTPUT 709; "DRIVETB0 TIMER1"    ! TRIGGER BUS SOURCE IS TIMER 1
20 OUTPUT 709; "SET OUTPUT TIMER1,ON;PULSE TIMER1,.001"
30 END
```

LOW/HIGH/TRG

The HP 3235's CPU (internal microprocessor) can force either trigger bus to a logic HIGH state (+5 volts) or a LOW state (0 volts). For example:

```
OUTPUT 709; "DRIVETB0 LOW"  ! FORCE TRIGGER BUS 0 LOW
```

The CPU can also pulse the trigger bus when the HP 3235 receives either the **TRG** command or the HP-IB Group Execute Trigger (GET). For additional information on the HP-IB GET refer to Appendix D in this manual. The following example shows how to pulse Trigger Bus 1 with both the **TRG** command and the HP-IB GET. In this mode, the steady state is high; the **TRG** command pulses the line low then high again.

```
10 OUTPUT 709; "DRIVETB1 TRG"       ! SOFTWARE TRIGGER SOURCE
20 OUTPUT 709; "TRG"                ! PULSE TRIGGER BUS 1
30 TRIGGER 709                      ! PULSE TRIGGER BUS 1
40 END
```

Reading the State of the Trigger Buses

The **TB*n*?** command returns the logic level of the specified trigger bus. It returns a '0' if the bus is at a logic low (0 volts) or a '1' for a logic high (+5 volts). The command has the form:

TB*n*? *[frame]*

To determine the state of Trigger Bus 0 in extender frame 3, execute the following:

```
10 OUTPUT 709; "TB0? 3"
20 ENTER 709; A
30 PRINT A
40 END
```

Driving the External Trigger Out BNC

The **DRIVEEXT** command selects the source to drive the External Trigger Out BNC on the mainframe. The command has the form:

DRIVEEXT *source*

The *source* parameter has eight possible values as shown in Table 9-4.

Table 9-4. DRIVEEXT *source* Parameters

Source Parameters	Description
OFF	Disable the External Trigger Out BNC
TB0	Drive from Trigger Bus 0
TB1	Drive from Trigger Bus 1
EXTIN	Drive from the External Trigger In BNC
TIMER0	Drive from Timer 0
TIMER1	Drive from Timer 1
LOW	Force the External Trigger Out BNC LOW (0 volts)
HIGH	Force the External Trigger Out BNC HIGH (+5 volts)

The following example shows how to connect the multimeter module's Voltmeter Complete pulse to the External Trigger Out BNC connector.

```
10 OUTPUT 709; "USE 800"        ! SELECT MULTIMETER IN SLOT 800
20 OUTPUT 709; "VMCMPLT TB0"    ! VOLTMETER COMPLETE TO TRIGGER BUS 0
30 OUTPUT 709; "DRIVEEXT TB0"     ! DRIVE EXT. TRIG. OUT BNC FROM TB0
40 END
```

NOTE

In addition to the limitation that sources other than plug-in modules can drive only one backplane trigger bus at a time, the internal switching will not allow:

DRIVETB*n* TRG; DRIVEEXT TB*n*

Using the Special Trigger Bus Configurations

Until now we have discussed using the trigger buses between plug-in modules within a frame or using sources at the mainframe to drive the entire trigger bus system. Four additional commands provide added control when using the trigger buses with extenders and different trigger signals in different extender frames.

NOTE

Only the mainframe External Trigger In and External Trigger Out BNC connectors may be used. These BNC connectors on the extender frames are not functional.

Using Frames Individually

The **AUTOTB** command connects or disconnects the extender trigger buses between each frame. With **AUTOTB ON** the extender and mainframe trigger buses are always connected together. Thus, any signal driving a backplane trigger bus in one frame appears on the extender trigger bus. This default setting is adequate for most applications. If you must set up different backplane trigger signals in your extenders, you must specify **AUTOTB OFF** and use the **DRIVEBP*n***, **DRIVEETB*n***, and **BPOUT** commands.

AUTOTB has the form:

AUTOTB ***mode***

where *mode* is either *on* or *off*.

Selecting Sources within a Frame

With **AUTOTB OFF**, you can have separate sources driving the backplane trigger bus lines in separate frames. The signal driving a backplane trigger bus is not placed on the extender trigger bus lines. The **DRIVEBP*n*** command specifies the source for driving the backplane trigger bus lines. It has the form:

DRIVEBP*n* ***source [,frame#]***

The *n* parameter can be either 0 or 1 for backplane trigger bus 0 or backplane trigger bus 1, respectively.

The *frame* parameter specifies the frame (0 for mainframe, 1-7 for extenders).

Table 9-5 lists the possible sources for driving the backplane trigger bus.

Table 9-5. DRIVEBP*n* *source* Parameters

Source Parameters	Description
OFF	Disable the specified backplane trigger bus
ETB0	Drive the specified backplane trigger bus from Extender Trigger Bus 0
ETB1	Drive the specified backplane trigger bus from Extender Trigger Bus 1
EXTIN*	Drive the specified backplane trigger bus from External Trigger In BNC
TIMER0*	Drive the specified backplane trigger bus from Timer 0
TIMER1*	Drive the specified backplane trigger bus from Timer 1
LOW	Force the specified trigger bus LOW (0 volts)
HIGH	Force the specified trigger bus HIGH (+5 volts)

* Valid only when frame = 0 (mainframe)

For example, to drive backplane trigger bus 0 in extender frame 3 from extender trigger bus 1, execute the following program.

```
10 OUTPUT 709; "AUTOTB OFF" ! DISCONNECT THE TRIGGER BUSES BETWEEN FRAMES
20 OUTPUT 709; "DRIVEBP0 ETB1,3"  ! DRIVE TB0 FROM EXTENDER BUS 1,FRAME 3
30 END
```

Specifying the Trigger Bus in an Extender

The **BPOUT** command specifies which backplane trigger bus in a specified frame sources a signal into the switching network for that frame. Only one backplane trigger bus can source a signal into the network at any one time. **AUTOTB** must be off. **BPOUT** has the form:

BPOUT *bus# [,frame#]*

The *bus#* is either 0 or 1 for Backplane Trigger Bus 0 or Backplane Trigger Bus 1, respectively. The optional *frame* specifies which frame sources the backplane signal. Default is the mainframe. For example, if you want to source a signal through Trigger Bus 0 from an HP 34506 Coaxial Matrix Module installed in slot 400 of extender 3, execute the following program:

```
10 OUTPUT 709; "AUTOTB OFF" ! DISCONNECT THE TRIGGER BUSES BETWEEN FRAMES
20 OUTPUT 709; "TRIGBUF 3400,IN"  ! TRIGGER SOURCE
30 OUTPUT 709; "BPOUT 0,3"        ! DRIVE TB0 IN EXTENDER FRAME 3
40 END
```

Driving the Extender Trigger Bus

The **DRIVEETB*n*** command specifies the frame and the source in that frame to drive the extender trigger bus between frames. The syntax for the command is:

DRIVEETB*n* *source [,frame#]*

The *n* parameter can be either 0 or 1 for extender trigger bus 0 or extender trigger bus 1, respectively. The *frame* parameter specifies which frame (0 for the mainframe or 1-7 for extenders) will drive the specified extender trigger bus.

The *source* parameter has nine choices listed in Table 9-6.

Table 9-6. DRIVEETB*n* *source* Parameters

Source Parameter	Description
OFF	Disable the specified extender trigger bus
BP	Drive the specified extender trigger bus from the source specified by the BPOUT command
ETB0*	Drive the specified extender trigger bus from Extender Trigger Bus 0
ETB1**	Drive the specified extender trigger bus from Extender Trigger Bus 1
EXTIN***	Drive the specified extender trigger bus from External Trigger In BNC
TIMER0***	Drive the specified extender trigger bus from Timer 0
TIMER1***	Drive the specified extender trigger bus from Timer 1
LOW	Force the specified extender trigger bus LOW (0 volts)
HIGH	Force the specified extender trigger bus HIGH (+5 volts)

* Valid only when n=1
** Valid only when n=0
*** Valid only when frame = 0 (mainframe)

To use the *BP* parameter, you must specify the **BPOUT** command with the same frame number as the **DRIVEETB*n*** command. For example, to drive extender trigger bus 0 from backplane trigger bus 1 in frame 3, execute the following program.

```
10 OUTPUT 709; "BPOUT 1,3"  ! USE TB1, FRAME 3 AS SOURCE
20 OUTPUT 709; "DRIVEETB0 BP,3"  ! DRIVE EXTENDER TRIGGER BUS 0 FROM
25                               ! TB1, FRAME 3
30 END
```

Examples

The following examples show how to use the trigger bus commands to create a path between frames for trigger signals. Read the discussion for each example to understand how it works.

Example 1. Signal from Module in Frame 1 to Module in Frame 0

The following program shows how to input a trigger signal through an HP 34506 Coaxial Matrix module in slot 600, frame 1, send it through the extender trigger bus 0, and output the signal through an HP 34506 Coaxial Matrix module in slot 4, frame 0.

```
10 OUTPUT 709; "AUTOTB OFF"   ! DISCONNECT TRIGGER BUSES BETWEEN FRAMES
20 OUTPUT 709; "TRIGBUF 1600,IN"    ! INPUT SIGNAL SLOT 600, FRAME 1
25                            ! TO BACKPLANE TRIGGER BUS 0 IN FRAME 1
30 OUTPUT 709; "BPOUT 0,1"  ! TBO SIGNAL INTO SWITCHING NETWORK, FRAME 1
40 OUTPUT 709; "DRIVEETBO BP,1" ! DRIVE EXTENDER TRIG. BUS 0 FROM FRAME 1
50 OUTPUT 709; "DRIVEBPO ETBO,0"  ! DRIVE BACKPLANE TRIG. BUS 0, FRAME 0
55                                ! FROM EXTENDER TRIG. BUS 0
60 OUTPUT 709; "TRIGBUF 400,OUT" ! OUTPUT TRIG. SIGNAL SLOT 400, FRAME 0
70 END
```

Example 2. Multimeter Complete from Frame 3 to External Out BNC Frame 0

This example shows how to pass the multimeter complete signal from the multimeter in slot 800 of extender frame 3 to the External Trigger Out BNC connector on the mainframe (frame 0).

```
10 OUTPUT 709; "AUTOTB OFF"   ! DISCONNECT TRIGGER BUSES BETWEEN FRAMES
20 OUTPUT 709; "USE 3800"    ! SELECT MULTIMETER IN SLOT 800 FRAME 3
30 OUTPUT 709; "VMCMPLT TBO" ! MULTIMETER COMPLETE TO BPTBO, FRAME 3
40 OUTPUT 709; "BPOUT 0,3"   ! TBO SIGNAL INTO SWITCHING NETWORK, FRAME 3
50 OUTPUT 709; "DRIVEETB1 BP,3" ! DRIVE EXTENDER TRIG. BUS 1 FROM FRAME 3
60 OUTPUT 709; "DRIVEBPO ETB1,0" ! DRIVE BACKPLANE TRIG. BUS 0, FRAME 0
65                               !   FROM EXTENDER TRIG. BUS 1
70 OUTPUT 709; "DRIVEEXT TBO"    ! DRIVE EXT. TRIG. OUT BNC FROM TBO
80 END
```

Using the Timer Channels

Your HP 3235 has two timer channels which can drive the trigger buses. Each timer channel can generate single pulses or square waves, or count down from a specified value, have the current value read by the system controller, and interrupt the system controller when the counter crosses zero. The mainframe also has a system clock with time and wait functions.

Enabling the Timers

The **SET OUTPUT** command enables or disables the timer channels output. **SET OUTPUT** has the form:

SET OUTPUT TIMER*n* *mode*

You must specify the timer channel, either Timer0 or Timer1, and the state. Possible modes are: *ON* and *OFF*. At power-on, the timer channels are disabled (*OFF*, 0 volts). You must set the timer channel output to *ON* to generate pulses or square waves. For example, the following program shows how to enable timer1 to generate a 1 kHz square wave. Squarewaves are discussed later in this chapter.

```
OUTPUT 709; "SET OUTPUT TIMER1 ON; SQWAVE TIMER1,1E-3"
```

Setting the Clock Source

The clock source determines the rate at which the timer channels operate. The two timer channels have an 800 kHz internal clock source. Alternately, you can specify an external clock source for Timer0 with the **SET CLKSRC** command. Timer1 must use the internal 800 kHz clock source. **SET CLKSRC** has the form:

SET CLKSRC TIMER0,*source*

NOTE

An external clock source may be specified for timer1 without an error; however no source will be connected.

Table 9-7 shows the possible sources for the **SET CLKSRC** command for Timer0

Table 9-7. SET CLKSRC Sources

Source Parameter	Description
INT	Use internal (800 kHz) source.
BP	Use the mainframe backplane trigger bus source specified by the BPOUT command or the most most recent TRIGBUF, TBDRIVE, VMCMPLT, etc. command
ETB0	Use Extender Trigger Bus 0 as the source.
ETB1	Use Extender Trigger Bus 1 as the source.
EXTIN	Use the External Trigger In BNC as the source

For example, to set the Timer 0 clock source to the External Trigger In BNC, execute:

```
OUTPUT 709; "SET CLKSRC TIMER0,EXTIN"
```

Pulse Output

You can use the timer channels to generate single pulses with the **PULSE** command. The pulses are Low-High-Low TTL level pulses. The **PULSE** command has the form:

PULSE TIMER*n*, ***width***

The **TIMER*n*** parameter specifies which timer channels generates the pulse: either timer0 or timer1. The *width* parameter specifies the pulse width. If *width* is less than 1, the units are in seconds (i.e. 0.015 means 0.015 seconds). If *width* is greater than or equal to 1, the units are in integral input clock pulses. For the internal clock source (800 kHz), one clock pulse equals 1.25 µS. Valid ranges for the *width* parameter are 0.625E-6 to 0.081918 seconds or 1 to 65535 clock pulses. Minimum pulse width is 1.25 µS.

Execute the **SET OUTPUT TIMER*n* ON** command prior to executing the **PULSE** command to enable the timer output. For accurately timed pulses, you should set the clock source (**SET CLKSRC**) to internal. For example, to output a single 1 millisecond pulse through the channel 00 trigger buffer on an HP 34506 Coaxial Matrix module in slot 600, execute the following program:

```
10 OUTPUT 709; "RST"
20 OUTPUT 709; "DRIVETBO TIMER0"
30 OUTPUT 709; "TRIGBUF 600,OUT"
40 OUTPUT 709; "SET CLKSRC TIMER0 INT; SET OUTPUT TIMER0 ON"
50 OUTPUT 709; "PULSE TIMER0 .001"
60 END
```

NOTE

When changing from generating square waves to pulses, output a single pulse before setting the final destination of the pulse. This resets the timer channels and ensures that they are set to the pulse mode.

Square Wave Output

The timer channels can also generate 50% duty cycle TTL level square waves with the **SQWAVE** command. This command has the form:

SQWAVE TIMER*n*, *period*

The **TIMER***n* parameter specifies which timer channel generates the square wave: either Timer 0 or Timer 1. The *period* parameter specifies the period of the square wave. If *period* is less than 1, the units are in seconds (i.e. 0.1 means a period of 0.1 seconds). If *period* is greater than 1, the units are in clock pulses (i.e. 25 means a period of 25 clock pulses). Using the internal clock source (800 kHz), one clock pulse equals 2.5 µS. The valid ranges for *period* are 2.5µS to 0.16384124 seconds and 2 to 131070 clock pulses (the internal clock pulse is 1.25µ*S).*

Execute the **SET OUTPUT TIMERn ON** command prior to executing the **SQWAVE** command to enable the timer output. For accurately timed pulses, you should set the clock source (**CLKSRC**) to internal. For example, to generate a 1 kHz square wave using Timer 1, and output it through the External Trigger Out BNC, execute the following:

```
10 OUTPUT 709; "RST"
20 OUTPUT 709; "SET CLKSRC TIMER1 INT; SET OUTPUT TIMER1 ON"
30 OUTPUT 709; "SQWAVE TIMER1 .001"
40 OUTPUT 709; "DRIVEEXT TIMER1
50 END
```

Counter Functions

You can also use both timers as down-counters. That is, you load a timer with a specified value and incoming pulses decrement the counter until it reaches zero (0). When the counter crosses zero, it will automatically reload with the value you specified. You can also interrupt the system controller when the counter crosses zero.

Setting the Counter

The **SET COUNT** command loads the specified counter with a specified value. The counter counts down at the rate determined by the **SET CLKSRC** (set clock source) command. Each pulse from the clock source decrements the counter value. The next clock pulse after the counter reaches zero reloads the counter with your specified value.

SET COUNT has the form:

SET COUNT TIMER*n*, *value*

The **TIMER***n* parameter specifies the timer to use, either Timer 0 or Timer1. the *value* parameter sets the counter value to any number between 0 and 65535.

Two ways to use the counters are to (1) count trigger bus events and (2) interrupt the system controller when the counter reaches zero. To use the counters to count trigger bus events, set the **SET CLKSRC** to the appropriate source (trigger bus or external). Set the counter to a value of 65535 and run the test. Use the **READ COUNT** command to read the current value of the counter. Subtract the value returned by **READ COUNT** from the initial value (65535) to get the number of events that have occurred. The following program shows how to do this.

```
10 OUTPUT 709; "RST"
20 OUTPUT 709; "SET COUNT TIMER0 65535" ! SET COUNTER VALUE
30 OUTPUT 709; "SET CLKSRC TIMER0 EXTIN" ! SET CLOCK SOURCE EXTIN BNC
40 WAIT 5 ! PULSE THE EXTERNAL TRIG IN BNC DURING THIS PERIOD
50 OUTPUT 709; "READ COUNT TIMER0"
60 ENTER 709; A
70 PRINT 65535-A;" COUNTS HAVE OCCURRED"
80 END
```

Reading the Counter

The **READ COUNT** command returns the current counter value from the specified timer. The command has the form:

READ COUNT TIMER*n*

The result goes either to the HP-IB output buffer (if the **READ COUNT** command originated from the system controller), the optional control panel (if the **READ COUNT** command originated from the control panel keyboard), or to an internal HP 3235 variable or array. To send the results to a variable or an array use the **MEM** command (see Chapter 7 of this manual) as shown in the following example.

```
10 OUTPUT 709;"RST"
20 OUTPUT 709;"SET COUNT TIMER0 65535"! SET COUNTER VALUE
30 OUTPUT 709;"SET CLKSRC TIMER0 EXTIN"! SET CLOCK SOURCE EXTIN BNC
40 WAIT 5 !  PULSE THE EXTERNAL TRIG IN BNC DURING THIS PERIOD
50 OUTPUT 709;"MEM RDCNT; READ COUNT TIMER0" ! STORE READING IN RDCNT
60 OUTPUT 709; "VREAD RDCNT"
70 ENTER 709; A
80 PRINT 65535-A;" COUNTS HAVE OCCURRED"
90 END
```

Enabling Timer Interrupts

To have a timer/counter interrupt the system controller when the counter reaches zero, you must enable the interrupt with the **ENABLE INTR TIMERn** command. You must also set the appropriate RQS Mask bit for the Status Register. Refer to Chapter 10 in this manual for additional information on interrupts. The interrupt enable command has the form:

ENABLE INTR TIMER*n*

To use the timer/counters to interrupt the system controller you first load a value into a timer/counter. For example, if you want the interrupt after 100 events, first set the clock source (see **SET CLKSRC** command) for Timer 0 to an external source. Remember, only timer 0 can use external clock sources. Next, load the value (see the **SET COUNT** command) 99 into timer 0. Set bit 12 (Timer0 Interrupt) in the RQS mask with the **RQS** command. Finally, enable the interrupt for Timer 0. When an interrupt occurs, the HP 3235 can either initiate a local subroutine or interrupt the system controller. The following program is a timer interrupt program for an HP series 200 or 300 computer.

```
10 OUTPUT 709; "RST"
20 OUTPUT 709; "SET CLKSRC TIMER0 EXTIN"
30 OUTPUT 709; "SET COUNT TIMER0 100"
40 OUTPUT 709; "RQS 4096"
50 OUTPUT 709; "ENABLE INTR TIMER0"
60 ON INTR 7 GOTO 90
70 ENABLE INTR 7;2
80 GOTO 80                ! WAIT FOR INTERRUPT
90 A=SPOLL(709)
100 IF BINAND(A,4096) THEN PRINT "INTERRUPT OCCURRED FROM TIMER 0"
110 ELSE
120 PRINT "INTERRUPT OCCURRED FROM OTHER SOURCE"
130 END
```

Disabling Timer Interrupts

The **DISABLE INTR** command disables the timer interrupts. The form for the command is:

DISABLE INTR TIMER*n*

NOTE

Both **ENABLE INTR** *and* **DISABLE INTR** *have uses other than timer interrupts. Refer to Chapter 10 in this manual and the HP 3235 Language Reference Manual for additional information.*

Clock Functions

The HP 3235 also has clock functions for timing purposes. The clock timer is not battery backed-up so you should not rely on it as your system timer. It does provide time-of-day information. It also provides a timer for pausing the HP 3235 a specified amount of time.

Setting the Time

The **SET TIME** command sets the time of day in the HP 3235's internal clock. It has the form:

SET TIME *seconds*

The *seconds* parameter represents the number of seconds past midnight. It has a valid range of 0 through 86399.99 with a resolution of 0.01 seconds. For example, to set the time to 2:30:15 PM (52215 seconds past midnight), execute the following:

```
OUTPUT 709; "SET TIME 52215"
```

Reading the Time of Day

The **TIME** command returns the current time of day as the number of seconds past midnight (or since power-on). The value returned is a real number. The value is sent to the HP-IB output buffer if the **TIME** command originated from the system controller; the optional control panel display if the command originated from the keyboard; or to an internal HP 3235 variable or array location if the **MEM** command is executed. The following example shows how to read the time information into your system controller:

```
10 OUTPUT 709; "TIME"
20 ENTER 709; A
30 PRINT A;" is the number of seconds past midnight"
40 END
```

Waiting a Specified Time

The **WAIT** command causes the HP 3235 to suspend operations for a specified amount of time. The READY annunciator on the control panel display goes off during the wait time. No other commands are executed while the HP 3235 is executing the **WAIT** command. The command has the form:

WAIT *seconds*

The *seconds* parameter in specified in units of seconds. The resolution for *seconds* is 1 mS for values less than 30 mS. The resolution is 10 mS for values greater than 30 mS. The valid range for the *seconds* parameter is 0 through 21,474,836 seconds.

For example, to suspend operations for five (5) seconds, execute:

```
OUTPUT 709; "WAIT 5"
```

Use the HP-IB Clear command (see Appendix D) or the **Device Clear** key on the optional control panel to abort the wait period.

Waiting for a Trigger

The HP 3235 can also wait for a falling edge to occur or a timeout, whichever comes first. The **WAITFOR** command causes the HP 3235 to suspend operations until either a falling edge occurs from a specified source or a specified time (optional) elapses. The command has the form:

WAITFOR *source [,timeout_value]*

The possible sources are shown in Table 9-8.

Table 9-8. WAITFOR Source Choices

Source Parameter	Description
EXTIN	Wait for a falling edge on External Trigger In BNC
TBO	Wait for a falling edge on Trigger Bus 0
TB1	Wait for a falling edge on Trigger Bus 1

The *timeout_value* is the amount of time, in seconds, the HP 3235 will wait for the trigger to occur. If no falling edge occurs on the specified source within the allotted time, the HP 3235 generates an error, the **WAITFOR** command completes, and the HP 3235 executes the next command received. The valid range for the *timeout_value* is 0 to 32767 seconds. If you specify 0 as the *timeout_value* (no timeout limit), the HP 3235 waits indefinitely. For example, to have the HP 3235 wait five (5) seconds for a falling edge on Trigger Bus 1, execute:

```
OUTPUT 709; "WAITFOR TB1,5"
```

Chapter 10

Interrupts

Introduction

This chapter provides interrupt programming information for the HP 3235. Interrupt structuring allows you to enable or disable individual interrupts for a variety of events. Interrupt events are: Data Available, Control Panel SRQ key, **SRQ** command, power-on or reset, READY state, errors, backplane interrupts from the HP 34522 Digital I/O moduel or the HP 34523 Breadboard module, Limit Test failures for the **VERIFY** and **LIMIT** commands, timer channel underflows, and Quick Interconnect Fixture open.

Interrupts can be serviced by your system controller. In addition, interrupts from plug-in modules and the timer channels can be serviced by calling HP 3235 subroutines. Figure 10-1 illustrates the entire interrupt system.

Plug-in Module Interrupts

This portion of the chapter explains how to program the HP 34522 Digital I/O Module and the HP 34523 Breadboard Module to generate an event interrupt and how to call a subroutine when an interrupt occurs. Later in this chapter we will show how to cause an event interrupt to interrupt the system controller.

Before sending a command to either the HP 34522 Digital I/O module or the HP 34523 Breadboard module, you must execute the **USE** command to designate which module will receive the commands.

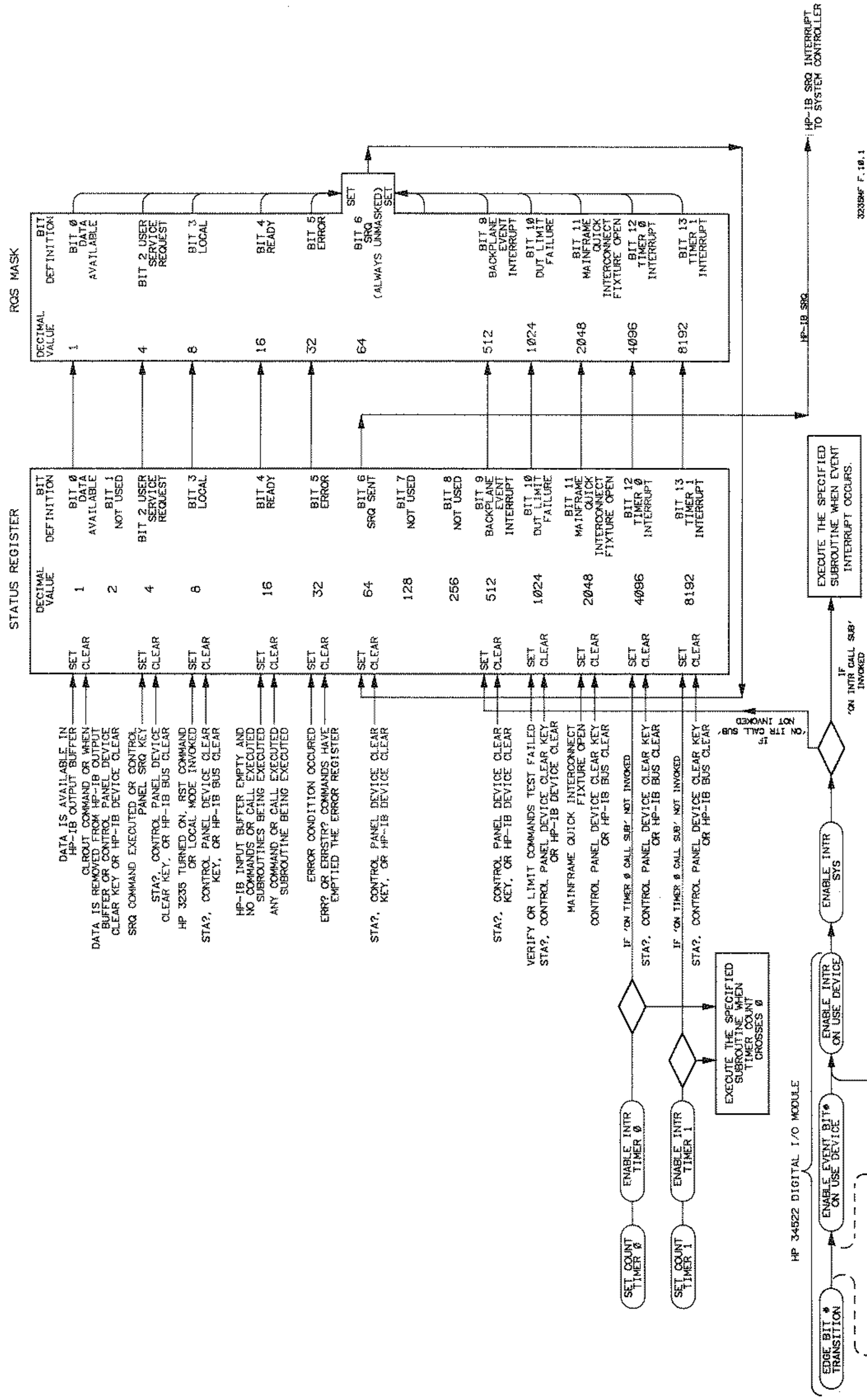


Figure 10-1. Interrupt System

Enabling Plug-In Module Interrupts

The **ENABLE INTR SYS** command enables the mainframe to acknowledge interrupts from the plug-in modules. The following example shows how to enable interrupts from the plug-in modules.

```
OUTPUT 709; "ENABLE INTR SYS"
```

The **DISABLE INTR SYS** command disables interrupts from the plug-in modules.

```
OUTPUT 709; "DISABLE INTR SYS"
```

Event Interrupts

The HP 34522 Digital I/O Module contains eight event interrupt bits which can be enabled to interrupt the mainframe when a programmed edge (rising or falling) is detected. This is called a backplane event interrupt or simply an event interrupt. The HP 34522 can be enabled to detect event interrupts on each of the eight bits. Event interrupts can either initiate subroutine executions or the HP 3235 mainframe can be programmed to interrupt the system controller. For additional information on programming the Digital I/O Module, refer to the HP 34522 Digital I/O Programming and Configuration Manual.

The HP 34523 Breadboard Module has one interrupt line to the mainframe processor. With the appropriate circuitry installed on the module, this line can be multiplexed to provide one-of-several event interrupt signals. Refer to the HP 34523 Breadboard Programming and Configuration Manual for additional information.

The following sections show how to enable event interrupts, disable event interrupts, and how to call HP 3235 subroutines when an event interrupt occurs.

Enabling Event Interrupts

The **ENABLE INTR** (enable interrupt) command enables the HP 34522 or 34523 to interrupt the mainframe when the designated edge is detected on an event bit. The **ENABLE INTR** command enables the interrupt from the slot specified by the last **USE** command.

NOTE

When programming the HP 34522 Digital I/O Module for interrupts you must:

1. *Specify the event bit transition (either low-to-high or high-to-low) for the event bit. Use the* **EDGE** *command.*

2. *Optionally clear the event bit to eliminate possible previous interrupts on that bit. Use the* **CLEAR EVENT** *command.*

3. *Enable the event bit to detect events with the* **ENABLE EVENT** *command.*

4. *Enable the interrupt with the* **ENABLE INTR** *command.*

In addition to the **ENABLE INTR** *command, the HP 34523 Breadboard Module requires whatever programming your circuit configuration demands.*

The following program example enables the HP 34522 in mainframe slot 200 to interrupt when a low-to-high edge is detected on event bit 3.

```
10  OUTPUT 709; "USE 200"          ! USE THE HP 34522 IN SLOT 200
20  OUTPUT 709; "EDGE 3,LH"        ! RISING EDGE IS ACTIVE FOR BIT 3
30  OUTPUT 709; "CLEAR EVENT 3"    ! CLEAR EVENT BIT 3
40  OUTPUT 709; "ENABLE EVENT 3"   ! ENABLE INTR FOR BIT 3
50  OUTPUT 709; "ENABLE INTR"      ! ENABLE INTR FROM SLOT 200
60  OUTPUT 709; "ENABLE INTR SYS"  ! ENABLE PLUG-IN MODULE INTR
70  GOTO 70                        ! WAIT FOR INTR
80  END
```

When a event interrupt occurs, the HP 3235 automatically disables future interrupts from that plug-in module. This means that any future events occurring on that plug-in module will not generate another interrupt unless the corresponding module is re-enabled with another **ENABLE INTR** command.

Disabling Event Interrupts

An event interrupt which has been enabled can be disabled with the **DISABLE INTR** command. The **ENABLE INTR** command must be executed to re-enable event interrupts from the module specified by the **USE** command. For example, to disable the HP 34522 in mainframe slot 200, execute:

```
OUTPUT 709; "USE 200;DISABLE INTR"
```

When the HP 3235 services an interrupting plug-in module and there is no **ON INTR CALL SUB** invoked for the interrupt, it disables the interrupt system (**DISABLE INTR SYS**). This prevents a second plug-in module from interrupting while the HP 3235 is servicing the first module. Therefore, you must re-enable the interrupt system (**ENABLE INTR SYS**) as part of the interrupt service routine if you expect additional interrupts.

Calling Subroutines on Event Interrupts

Backplane event interrupts can be serviced by either interrupting the system controller or by calling HP 3235 subroutines. This section describes how to call an HP 3235 subroutine on an event interrupt. Controller servicing of interrupts is discussed later in this chapter.

The **ON INTR CALL** *sub_name* command executes the named subroutine when an enabled backplane event interrupt occurs. When an interrupt occurs, the HP 3235 prohibits other interrupts from calling subroutines.

Commands (including **CALL** executed subroutines, see Chapter 8) executed from the system controller (via HP-IB) or from the optional control panel have a higher priority than subroutines called from an interrupt. This means that commands from the controller or keyboard are executed immediately even if a subroutine is running that was called from an interrupt. A subroutine called from an interrupt executes as the mainframe microprocessor finds time between commands from the system controller and control panel commands.

RUN executed subroutines (see Chapter 8) carry a lower priority than subroutines called from an interrupt. This means that a subroutine called from an interrupt executes normally and the **RUN** executed subroutine executes only as the microprocessor finds time.

The subroutine specified in the **ON INTR CALL** *sub_name* command must be defined in the HP 3235's memory prior to executing the command. If the subroutine does not exist, it generates an error.

The following example demonstrates how an interrupt-called subroutine operates.

```
10   OUTPUT 709; "RST"                        ! RESET THE HP 3235
20   OUTPUT 709; "SUB INTERR1"                ! CREATE THE SUBROUTINE
30   OUTPUT 709; "USE 200"                    ! USE DEVICE IS DIG I/O
40   OUTPUT 709; "DISP 'Interrupt Occurred'"    ! DISPLAY MESSAGE
50   OUTPUT 709; "WRITE 125"                  ! WRITE WORD TO PORT
60   OUTPUT 709; "SUBEND"                     ! END SUBROUTINE
70   OUTPUT 709; "USE 200"                    ! USE DEVICE IS DIG I/O
80   OUTPUT 709; "EDGE 3,LH"                  ! RISING EDGE FOR BIT 3
90   OUTPUT 709; "ENABLE EVENT 3"             ! ENABLE INTR FOR BIT 3
100  OUTPUT 709; "ENABLE INTR"                ! ENABLE INTR FROM SLOT 200
110  OUTPUT 709; "ENABLE INTR SYS"            ! ENABLE PLUG-IN MODULE INTR
120  OUTPUT 709; "ON INTR CALL INTERR1"       ! CALL SUB WHEN INTR OCCURS
130  GOTO 130                                 ! WAIT FOR INTR
140  END
```

Interrupt Query

The **INTR?** command returns the slot address of the most recent module to interrupt and be serviced by the mainframe processor. The slot address is a number in the form *es00* where *e* represents the frame number and *s* represents the slot number. If no interrupt has occurred since power-on, reset, or the last **INTR?** command, the command returns a value of -1.

Interrupt servicing occurs when the HP 3235 completes the command or subroutine it was executing at the time of the interrupt. If an interrupt is cleared or disabled after it occurs but before it is serviced, **INTR?** will not show that the interrupt has occurred and an unexpected interrupt error may occur.

The following program shows how to use the **INTR?** command.

```
10   OUTPUT 709; "SUB SERVICE"              ! CREATE THE SUBROUTINE
20   OUTPUT 709; "INTEGER INTRSLOT"         ! DEFINE VARIABLE
30   OUTPUT 709; "MEM INTRSLOT"             ! ENABLE MEMORY MODE
40   OUTPUT 709; "INTR?"                    ! INTERRUPT QUERY
50   OUTPUT 709; "MEM OFF"                  ! DISABLE MEMORY MODE
60   OUTPUT 709; "DISP 'Slot Number',INTRSLOT,' Interrupted'"
70   OUTPUT 709; "WAIT 2"                   ! DISPLAY MESSAGE FOR 2 SEC
80   OUTPUT 709; "USE 200"                  ! USE DEVICE IS DIG I/O
90   OUTPUT 709; "DISP 'Ready For Next Interrupt'"
100  OUTPUT 709; "CLEAR EVENT 3"            ! CLEAR EVENT BIT
110  OUTPUT 709; "ENABLE INTR"              ! RE-ENABLE INTR FROM SLOT 200
120  OUTPUT 709; "SUBEND"                   ! END SUBROUTINE
130  OUTPUT 709; "USE 200"                  ! USE DEVICE IS DIG I/O
140  OUTPUT 709; "EDGE 3,LH"                ! RISING EDGE FOR BIT 3
150  OUTPUT 709; "ENABLE EVENT 3"           ! ENABLE INTR FROM BIT 3
160  OUTPUT 709; "ON INTR CALL SERVICE"     ! CALL SUB WHEN INTR OCCURS
170  OUTPUT 709; "ENABLE INTR"              ! ENABLE INTR FROM SLOT 200
180  OUTPUT 709; "ENABLE INTR SYS"          ! ENABLE PLUG-IN MODULE INTR
190  GOTO 190                               ! WAIT FOR INTR
200  END
```

Timer Interrupts

The two timer/counters (see Chapter 9) can also interrupt when the counters underflow past zero. This occurs on the clock pulse after the counters reach zero. They automatically reset to the last value written to them when this occurs (see the **SET COUNT** command). For example, **SET COUNT 5** requires six (6) clock pulses to interrupt. This section of the chapter shows how to use the timers to generate an interrupt.

Enabling Timer Interrupts

Timer interrupts can be serviced by either interrupting the system controller or by calling subroutines. This section describes how to call a subroutine on a timer interrupt. Controller servicing of interrupts is discussed later in this chapter.

The **ON TIMERn CALL** *sub_name* command executes the named subroutine when an enabled timer interrupt occurs. When an interrupt occurs, the HP 3235 prohibits other interrupts from calling subroutines.

Calling Subroutines on Timer Interrupts

Commands (including **CALL** executed subroutines, see Chapter 8) executed from the system controller (via HP-IB) or from the optional control panel have a higher priority than subroutines called from an interrupt. This means that commands from the controller or keyboard are executed immediately even if a subroutine is running that was called from an interrupt. A **CALL** executed subroutine executes normally and the subroutine called from an interrupt executes as the mainframe microprocessor finds time. If an **ON TIMERn CALL** *sub_name* command is part of a **CALL** executed subroutine and the interrupt occurs while the **CALL** executed subroutine is executing, the **CALL** executed subroutine will continue to completion before executing the subroutine called from the interrupt.

RUN executed subroutines (see Chapter 8) carry a lower priority than subroutines called from an interrupt. This means that a subroutine called from an interrupt executes normally and the **RUN** executed subroutine executes only as the microprocessor finds time.

The subroutine specified in the **ON TIMERn CALL** *sub_name* command must reside in the HP 3235's memory prior to executing the command. If the subroutine does not exist, it generates an error.

The following program is a simple timer interrupt program to call a subroutine.

```
10  OUTPUT 709; "RST"                          ! RESET THE HP 3235
20  OUTPUT 709; "SUB TIMERINTR"                ! CREATE SUBROUTINE
30  OUTPUT 709; "DISP 'INTERRUPT OCCURRED'"    ! DISPLAY MESSAGE
40  OUTPUT 709; "SUBEND"                       ! END SUBROUTINE
50  OUTPUT 709; "SET CLKSRC TIMER0 EXTIN"      ! SET CLOCK SOURCE
60  OUTPUT 709; "SET COUNT TIMER0 99"          ! SET TIMER COUNT
70  OUTPUT 709; "ENABLE INTR TIMER0"           ! ENABLE INTR FROM TIMER
80  OUTPUT 709; "ON TIMER0 CALL TIMERINTR"     ! CALL SUB WHEN TIMER
85                                             ! UNDERFLOWS PAST ZERO
90  GOTO 90                                    ! WAIT FOR INTR
100 END
```

Disabling Timer Interrupts

The **DISABLE INTR** command disables the timer interrupts. The form for the command is:

DISABLE INTR TIMER*n*

Instrument Status and Interrupting the Controller

One important feature of your HP 3235 is its ability to interrupt the system controller when certain specified conditions are met. Of course, the system controller must be appropriately programmed to respond to the interrupt. The Service Request (SRQ, a function of the HP-IB Interface) implements this feature and is independent of all other HP-IB activity. The following list gives the possible sources of an SRQ interrupt.

- **Data Available.** The HP 3235 can interrupt the system controller when data is available such as from a scan sequence.
- **User Service Request.** The mainframe can interrupt when it executes the **SRQ** command or the control panel SRQ key is pressed.
- **LOCAL.** The mainframe can interrupt at power-on, when it is reset, or when the mainframe enters the Local mode.
- **READY.** The mainframe can interrupt when it is in the READY state. That is, the HP-IB input buffer is empty, the mainframe is not executing a command or a **CALL** executed subroutine.
- **ERROR.** The mainframe can interrupt when an error condition occurs.
- **Backplane Event Interrupt.** The mainframe can interrupt when an event interrupt occurs from a plug-in module.
- **DUT Limit Failure.** The mainframe can interrupt when either the **VERIFY** or **LIMIT** tests fails
- **Quick Interconnect Fixture Open.** The mainframe only (not the extender frames) can interrupt when its Quick Interconnect Fixture is open.
- **Timer Interrupt.** The two timer channels can interrupt when the counters underflow past zero.

The remainder of this chapter shows how to program the HP 3235 to interrupt your system controller. We also show how to program the HP Series 200/300 technical computers to respond to the interrupt.

The Status Register

The Status Register is a 16-bit register that monitors the current status of the HP 3235. Table 10-1 shows the bit numbers, bit values, and bit definitions for the Status Register.

Table 10-1. Status Register

Bit Number	Decimal Value	Description
0	1	DATA AVAILABLE. Set true (1) when data is available in the HP-IB Output buffer. Cleared (0) by the CLROUT command or when data is otherwise removed from the buffer.
1	2	Always zero (not used).
2	4	USER SERVICE REQUEST. Set true (1) when the SRQ command is executed over HP-IB or inside a subroutine or the Control Panel SRQ key is pressed. Cleared (0) with the STA? or CLR commands or an HP-IB Bus "Clear".
3	8	LOCAL. Set true (1) when the HP 3235 mainframe is turned on, reset, or when the mainframe enters the Local Mode. Cleared (0) with the STA? or CLR commands or the HP-IB Bus "Clear".
4	16	READY. Set true (1) when the HP-IB Input buffer is empty and the HP 3235 is not executing a command or a CALL executed subroutine. Cleared (0) when the HP 3235 is executing a command or a CALL executed subroutine.
5	32	ERROR. Set true (1) when an error condition occurs. Cleared (0) when ERR? or ERRSTR? command clears the error register, with the CLR command or HP-IB Bus "Clear"
6	64	SRQ SENT. Set true (SRQ sent to system controller) when any of the other unmasked bits in the Status Register are set true. Cleared by STB? or CLR commands, the control panel Device Clear key, or an HP-IB Bus "Clear".
7-8	128, 256	Always zero (not used)
9	512	BACKPLANE EVENT INTERRUPT. Set true (1) when a plug-in module event interrupt occurs. Cleared (0) by the STA? or CLR commands, the control panel Device Clear key, or the HP-IB "Clear".
10	1024	DEVICE-UNDER-TEST LIMIT FAILURE. Set true (1) when either the LIMIT or VERIFY commands test fails. Cleared with the STA? or CLR commands, the control panel Device Clear key, or the HP-IB "Clear".

Table 10-1 (cont'd)

Bit Number	Decimal Value	
11	2048	FIXTURE OPEN. Set true (1) when the mainframe quick interconnect fixture is open. Cleared (0) with STA? STA? or CLR commands, the control panel Device Clear key, or the HP-IB Bus "Clear".
12	4096	TIMER0 INTERRUPT. Set true (1) when Timer0 has been enabled to interrupt and an ON TIMER0 CALL sub_name is not invoked. Cleared (0) with STA? CLR commands, the control panel Device Clear key, or the HP-IB Bus Clear.
13	8192	TIMER1 INTERRUPT. Set true (1) when Timer1 has been enabled to interrupt and an ON TIMER1 CALL sub_name is not invoked. Cleared (0) with STA? or Clear commands, the control panel Device Clear key, the HP-IB Bus Clear.
14-15	-----	Not Used.

Service Request Conditions

Bit 6 (SRQ) is set when any event corresponding to one of the other status bits occurs AND when the corresponding RQS Mask bit is unmasked. The RQS Mask is described later in this chapter. The SRQ bit is cleared when the unmasked bits are cleared or after a system controller executes an HP-IB Serial Poll (see Appendix D), the STB? command is executed, or the event setting a Status Register bit is cleared (for example, reading the error register).

The SRQ annunciator on the optional control panel is turned on when Bit 6 (SRQ) is set. This indicates that the HP 3235 is attempting to interrupt the system controller.

Clearing the Status Register

The **CLR** command, the control panel Device Clear key, or the HP-IB Device Clear (see Appendix D) clear certain bits in the Status Register. See Table 10-1 for specific information. This also resets the HP-IB SRQ line to the system controller.

Reading the Status Register

The **STA?** command returns a decimal value indicating the status of the HP 3235. This decimal value is equal to the sum of the decimal values of the bits in the Status Register that are set (true). See Table 10-1 for a list of decimal values. For example, if an error condition has occurred, **STA?** returns a value of 32. If an error condition has occurred and the mainframe is requesting service from the system controller (Bit 6 set), **STA?** returns a value of 96 (32 for the error plus 64 for the SRQ).

Executing **STA?** clears the bits that are set by an event (i.e. fixture open, timer interrupt, backplane event interrupt, Limit Test interrupt, power-on/local interrupt, and user SRQ). Bits which reflect the state of something are not cleared (i.e. error, data available, and ready). Bit 6 (and the HP-IB SRQ line) is cleared only if no unmasked bits (see the **RQS** command later in this chapter) remain set.

The following program shows how to read and decode the Status Register.

```
10 OUTPUT 709; "STA?"      ! POLL STATUS REGISTER
20 ENTER 709;A             ! ENTER DECIMAL VALUE
30 PRINT A
40 IF A=0 THEN PRINT "NO BITS SET"
50 IF BINAND(A,1) THEN PRINT "DATA AVAILABLE"          ! BIT 0
60 IF BINAND(A,4) THEN PRINT "USER SERVICE REQUEST" ! BIT 2
70 IF BINAND(A,8) THEN PRINT "LOCAL"                   ! BIT 3
80 IF BINAND(A,32) THEN PRINT "ERROR"                  ! BIT 5
90 IF BINAND(A,64) THEN PRINT "SRQ SENT"               ! BIT 6
100 IF BINAND(A,512) THEN PRINT "BACKPLANE EVENT INTERRUPT" ! BIT 9
110 IF BINAND(A,1024) THEN PRINT "DUT LIMIT FAILURE" ! BIT 10
120 IF BINAND(A,2048) THEN PRINT "FIXTURE OPEN" ! BIT 11
130 IF BINAND(A,4096) THEN PRINT "TIMER0 INTERRUPT" ! TIMER0
140 IF BINAND(A,8192) THEN PRINT "TIMER1 INTERRUPT" ! TIMER1
150 END
```

NOTE

Bit 4, Ready, will always show cleared because the HP 3235 is busy executing the **STA?** *or* **STB?** *command.*

The **STB?** command returns a decimal value equal to the sum of the decimal values of the lower eight bits (bits 0 through 7) of the Status Register. **STB?** clears Bit 6 (SRQ Sent) in the Status Register. The following program uses **STB?** to read the lower eight bits and display the decimal value.

```
10 OUTPUT 709; "STB?"
20 ENTER 709; A
30 PRINT "THE STATUS BYTE VALUE IS: ";A
40 END
```

The Status Register at Power-on

At power-on, both the **STA?** and the **STB?** commands return the value '8' if the power-on SRQ was disabled (**PONSRQ OFF**, discussed later in this chapter). If power-on SRQ is enabled (**PONSRQ ON**), they return a value of '72' and the HP 3235 asserts the HP-IB SRQ line.

Data Destination

If the **MEM** command is used (see Chapter 7) the value returned by either **STA?** or **STB?** is stored in the named variable or array. Otherwise, the result goes to the control panel display if the command originated from the control panel or the HP-IB output buffer if the command originated from the system controller.

The RQS Mask

The RQS mask determines which condition(s) in the status register sets bit 6 (SRQ) in the Status Register and generates an SRQ interrupt to the system controller. Refer to Figure 10-1. If a particular event monitored by the Status register occurs, and the corresponding bit in the RQS mask is "unmasked" (see the **RQS** command) then the HP 3235 sets Bit 6 (SRQ) in the Status Register. When Bit 6 of the Status Register is true, the HP 3235 asserts the HP-IB SRQ and the control panel SRQ annunciator turns on. Your system controller must be programmed to respond to the SRQ interrupt.

Setting the RQS Mask

The **RQS** command sets or "unmasks" bits in the RQS Mask. Bit Definition in the RQS Mask is the same as the Status Register. See Table 10-1. Only bits 0, 2, 3, 4, 5, 9, 10, 11, 12, and 13 can be unmasked. Other bits are ignored. **RQS** has the form:

RQS *unmask_value*

The *unmask_value* is a decimal value equal to the sum of the bit values you want to unmask. For example, to unmask bits 5 (Error bit, decimal value 32) and9 (Backplane Event Interrupt, decimal value 512), execute:

```
OUTPUT 709; "RQS (512+32)"
```

or

```
OUTPUT 709; "RQS 544"
```

Now, whenever either an error occurs or a plug-in module generates a backplane event interrupt, the HP 3235 attempts to interrupt the system controller.

Executing **RQS 0** clears the RQS Mask and thereby clears bit 6 (SRQ) in the Status Register.

Reading the RQS Mask Value

The **RQS?** command returns a decimal value equal to the sum of the bit values that are unmasked. Bit values are the same as the Status Register. See Table 10-1. The following program uses **RQS?** to return the current value of the RQS mask.

```
10 OUTPUT 709; "RQS?"
20 ENTER 709;A
30 PRINT "RQS VALUE = ";A
40 PRINT "UNMASKED BITS ARE: "
50 IF BINAND(A,1) THEN PRINT "DATA AVAILABLE"      ! BIT 0
60 IF BINAND(A,4) THEN PRINT "USER SERVICE REQUEST" ! BIT 2
70 IF BINAND(A,8) THEN PRINT "LOCAL"               ! BIT 3
80 IF BINAND(A,16) THEN PRINT "READY"              ! BIT 4
90 IF BINAND(A,32) THEN PRINT "ERROR"              ! BIT 5
100 IF BINAND(A,512) THEN PRINT "BACKPLANE INTERRUPT EVENT" ! BIT 9
110 IF BINAND(A,1024) THEN PRINT "DUT LIMIT FAILED"  ! BIT 10
120 IF BINAND(A,2048) THEN PRINT "FIXTURE OPEN" ! BIT 11
130 IF BINAND(A,4096) THEN PRINT "TIMER0 INTERRUPT" ! BIT 12
140 IF BINAND(A,8192) THEN PRINT "TIMER1 INTERRUPT" ! BIT 13
150 END
```

To have a timer/counter interrupt the system controller when the counter underflows past zero and reloads the initial value, you must enable the interrupt with the **ENABLE INTR** command. You must also set the appropriate RQS Mask bit for the Status Register. The timer interrupt enable command has the form:

ENABLE INTR TIMER*n*

To use the timer/counters to interrupt the system controllers you first load a value into a timer/counter. For example, if you want Timer 0 to interrupt after 100 events, first set the clock source (see **SET CLKSRC** command) for Timer 0 to an external source. Remember, only Timer 0 can use external clock sources. Next, load the value 99 into Timer 0 (see the **SET COUNT** command). In this way, event number 100 will underflow the counter and generate the interrupt. Set bit 12 (Timer 0 Interrupt) in the RQS mask with the **RQS** command. Finally, enable the interrupt for Timer 0. You must also program your system controller to respond to the SRQ interrupt.

Controller Response to a Service Request

When the SRQ message is sent, the system controller must first determine which instrument (if more than one is enabled to assert SRQ) is interrupting. This is done by conducting an HP-IB Serial Poll (see Appendix D) of each instrument enabled to generate an SRQ interrupt. When an instrument is polled it responds by sending its (8-bit) Status Register value. By looking at bit 6, the system controller can determine if that instrument asserted SRQ. If bit 6 is true then the system controller can look at the other bits to determine the nature of the request. If the Status Byte for the instrument polled indicates that the instrument is not asserting SRQ (Bit 6 = 0) then the controller continues by polling the next instrument on the interface. This assumes that the system has been programmed to respond to SRQ interrupts.

Examples

The following examples demonstrate how to use the SRQ interrupt with an HP Series 200/300 Technical Computer.

Example 1. Control Panel SRQ. This program example interrupts the system controller when the SRQ command is executed from the control panel. It is a simple program that demonstrates interrupt programming.

```
10  OUTPUT 709; "RST"          ! RESET THE HP 3235
20  OUTPUT 709; "RQS 4"        ! USER SERVICE REQUEST MASK
30  ON INTR 7 GOTO 60          ! SPECIFY CONTROLLER INTERRUPT ROUTINE
40  ENABLE INTR 7;2            ! ENABLE CONTROLLER INTERRUPT ON HPIB SRQ
50  GOTO 50                    ! WAIT FOR SRQ COMMAND
60  OUTPUT 709; "STA?"         ! GET STATUS REGISTER VALUE
70  ENTER 709; A
80  IF BINAND(A,4) THEN
90  PRINT "SRQ ENTERED"        ! TEST FOR USER SVC. REQUEST
100 ELSE
110 PRINT "SOMETHING ELSE INTERRUPTED"
120 END IF
130 END
```

Line 30 instructs the controller what action to take when an interrupt occurs. Line 40 enables SRQ interrupts (mask value 2) on HP-IB interface select code 7. Refer to the interfacing manual for your system controller for interrupt servicing information.

Example 2. Data Ready at End of Scan Sequence. Scanning (see Chapter 6) is a feature of the HP 3235 that allows it to sequence through a list of channels making a measurement on each channel. This example creates a subroutine to scan through a list of 32 channels and store the readings in an array. At the end of the scan, the measurements are sent to the HP-IB output buffer and the HP 3235 interrupts the system controller.

```
10   DIM A(31)                             ! DIM ARRAY FOR 32 READINGS
20   OUTPUT 709; "RST; USE 800"            ! RESET HP 3235; USE MULTIMETER
30   OUTPUT 709; "RQS 1"                   ! RQS MASK FOR DATA READY
40   OUTPUT 709; "SUB INTERRUPT"           ! CREATE SUBROUTINE
50   OUTPUT 709; "DIM RDGS(31)"            ! DIM ARRAY FOR 32 READINGS
60   OUTPUT 709; "MEM RDGS"                ! TURN ON MEMORY MODE
70   OUTPUT 709; "MEAS DCV AB0,101-138"    ! SCAN THE CHANNELS
80   OUTPUT 709; "MEM OFF; VREAD RDGS"     ! PUT READINGS IN OUTPUT BUFFER
90   OUTPUT 709; "SUBEND"                  ! END SUBROUTINE
100  ON INTR 7 GOTO 170                    ! SETUP CONTROLLER INTERRUPT
110  ENABLE INTR 7;2                       ! ENABLE CONTROLLER INTERRUPT
120  OUTPUT 709; "RUN INTERRUPT"           ! CALL SUBROUTINE
130  !
140  ! THE SYSTEM CONTROLLER CAN DO OTHER ACTIVITIES DURING THE SCAN
150  !
160  GOTO 160
170  ENTER 709; A(*)                       ! ENTER READINGS
180  PRINT A(*)                            ! PRINT READINGS
190  END
```

Example 3. Timer Interrupt. In this example we load Timer 0 with a value of 10. Each incoming pulse on the HP 3235's External Trigger IN BNC decrements the counter. When the counter underflows (11 counts) the HP 3235 interrupts the system controller.

```
10   OUTPUT 709; "RST"                         ! RESET THE HP 3235
20   OUTPUT 709; "SET CLKSRC TIMER0 EXTIN"     ! SET CLOCK SRC FOR TIMER 0
30   OUTPUT 709; "SET COUNT TIMER0 10"         ! SET COUNT TO 10
40   OUTPUT 709; "RQS 4096"                    ! UNMASK TIMER 0 INTERRUPT
50   OUTPUT 709; "ENABLE INTR TIMER0"          ! ENABLE INTERRUPT
60   ON INTR 7 GOTO 90                         ! SETUP CONTROLLER SRQ INTR
70   ENABLE INTR 7;2                           ! ENABLE CONTROLLER INTR
80   GOTO 80                                   ! WAIT FOR INTERRUPT
90   OUTPUT 709; "STA?"                        ! READ STATUS REGISTER
100  ENTER 709; A                              ! READ VALUE
110  IF BIT(A,12) THEN                         ! TEST FOR TIMER 0 INTR
120  PRINT "Timer Interrupt Occurred"          ! PRINT MESSAGE
130  OUTPUT 709; "ENABLE INTR TIMER0"          ! RE-ENABLE TIMER INTR
140  ELSE
150  PRINT "Something Else Caused Interrupt"
160  END IF
170  END
```

Example 4. Digital I/O Interrupt. This example waits for an event interrupt from event bit 3 on the HP 34522 Digital I/O Module in slot 200. This example is similar to the example shown earlier in this chapter but this one demonstrates how to interrupt the system controller.

```
10   OUTPUT 709; "RST 200; USE 200"        ! RESET THE HP 3235
20   OUTPUT 709; "EDGE 3,LH"               ! RISING EDGE FOR BIT 3
30   OUTPUT 709; "CLEAR EVENT 3"           ! CLEAR BIT 3
40   OUTPUT 709; "ENABLE EVENT 3"          ! ENABLE INTR FROM BIT 3
50   OUTPUT 709; "ENABLE INTR"             ! ENABLE INTR FROM SLOT 200
60   OUTPUT 709; "ENABLE INTR SYS"         ! ENABLE PLUG-IN MODULE INTR
70   OUTPUT 709; "RQS 512"                 ! UNMASK BACKPLANE INTR BIT
80   ON INTR 7 GOTO 110                    ! SETUP CONTROLLER SRQ INTR
90   ENABLE INTR 7;2                       ! ENABLE CONTROLLER INTR
100  GOTO 100                              ! WAIT FOR INTR
110  OUTPUT 709; "STA?"                    ! READ STATUS REGISTER
120  ENTER 709; A                          ! READ VALUE
130  OUTPUT 709; "INTR?"                   ! INTERRUPT QUERY
140  ENTER 709; Slot                       ! READ SLOT NUMBER
150  IF BINAND(A,512) THEN                 ! TEST FOR BACKPLANE INTR
160  PRINT "Digital Interrupt From Slot"; Slot  ! PRINT MESSAGE
170  OUTPUT 709; "ENABLE INTR"             ! RE-ENABLE BACKPLANE INTR
180  ELSE
190  PRINT "Something Else Caused Interrupt"
200  END IF
210  END
```

Example 5. Fixture Open. This example shows how to program the HP 3235 to interrupt the system controller when the mainframe Quick Interconnect Fixture is open. Note that only the HP 3235 mainframe can interrupt when its fixture is open; the HP 3235E Extender frames do not have this feature.

```
10   OUTPUT 709; "RST"          ! RESET THE HP 3235
20   OUTPUT 709; "RQS 2048"     ! SET MASK FOR FIXTURE OPEN
30   ON INTR 7 GOTO 60          ! SET CONTROLLER INTERRUPT ROUTINE
40   ENABLE INTR 7;2            ! ENABLE CONTROLLER INTERRUPT
50   GOTO 50                    ! WAIT FOR FIXTURE OPEN
60   OUTPUT 709; "STA?"         ! READ STATUS REGISTER VALUE
70   ENTER 709;A                ! READ VALUE
80   IF BIT (A,11) THEN         ! TEST FOR FIXTURE OPEN INTR
90   PRINT "Mainframe Interconnect Fixture Open"
100  ELSE
110  PRINT "Something Else Caused Interrupt"
120  END IF
130  END
```

Power-On SRQ Interrupt

You can program your HP 3235 to interrupt the system controller when power is applied to the mainframe. The **PONSRQ** command enables or disables the HP 3235 to assert the HP-IB SRQ line and attempt to interrupt the system controller. If power is removed from one of the extender frames, a power failure error occurs. Thus, **PONSRQ** applies to the extender frames as well as the mainframe.

When power-on SRQ is enabled (**PONSRQ ON**), the RQS mask is set to value "8" at power on thus enabling the SRQ interrupt. If the power-on SRQ is disabled (**PONSRQ OFF**) the RQS mask is set to value "0" at power-on. The **PONSRQ** state (**ON** or **OFF**) is stored in nonvolatile memory so it is not lost when power is removed from the mainframe.

The following example shows how to program the HP 3235 and the system controller for power-on interrupt.

```
10   OUTPUT 709; "RST"        ! RESET THE HP 3235 AND CLEAR STATUS REG
20   OUTPUT 709; "STA?"       ! CLEAR POSSIBLE PENDING SRQ
30   ENTER 709; Temp          ! READ VALUE
40   OUTPUT 709; "PONSRQ ON"  ! SET PONSRQ ON
50   ON INTR 7 GOTO 90        ! SET CONTROLLER INTERRUPT ROUTINE
60   ENABLE INTR 7;2          ! ENABLE CONTROLLER INTERRUPT
70   PRINT "Cycle Power On HP 3235 Mainframe"
80   GOTO 80                  ! WAIT FOR HP 3235 POWER-ON
90   OUTPUT 709; "STA?"       ! GET STATUS REGISTERS
100  ENTER 709; A             ! READ VALUE
110  IF BIT(A,3) THEN         ! TEST FOR POWER-ON SRQ
120  PRINT "Power-On SRQ"
130  ELSE
140  PRINT "Something Else Caused SRQ"
150  END IF
160  END
```

Chapter 11

HP 3235 Advanced Programming

Introduction

This chapter shows you how to use some of the advanced programming capabilities of your HP 3235. In particular it covers:

- HP-IB Local and Remote considerations
- I/O Modes, Input and Output Buffering, and Output Data Formats
- Storing and Recalling Instrument States
- Internal Memory Usage
- HP 3235 Internal Task Priority

HP-IB Local/Remote Considerations

The HP 3235 executes commands from two different sources: the control panel keyboard and the system controller (via HP-IB). Commands from these two sources are treated differently depending on whether the HP 3235 is in the local or remote mode.

Local Mode

In the HP-IB LOCAL mode (REMOTE annunciator is off in the control panel display), the HP 3235 executes commands from both sources.

The instrument is in LOCAL mode at power-on and will stay in that mode until you execute a remote command (see next heading). The HP 3235A is in LOCAL if the REMOTE annunciator is off. Execute either the **LCL** or **LOCAL** commands or press the control panel LOCAL key to return the mainframe to LOCAL Mode. You may also execute the HP-IB LOCAL command. Refer to Appendix D in this manual for more information on the HP-IB bus commands. For example:

```
OUTPUT 709; "LOCAL"
```

Remote Mode

In the HP-IB REMOTE mode the REMOTE annunciator is on in the optional HP 34550 Control Panel. Your HP 3235 executes commands from the system controller. The HP 3235 can also execute commands from the control panel keyboard which do not change any hardware state. For example, in the remote mode, you can close a relay channel from the system controller but not from the control panel keyboard. You can, however, determine the status of the multimeter, read an error message, etc. from the control panel because these do not change the state of the hardware.

Because most controllers set the HP-IB REN line true at power-on or reset, executing any command (except **LOCAL** and **LCL**) will also put your HP 3235A in its REMOTE mode.

Sequential Relay Command Execution

The HP 3235 mainframe executes commands in the order received. For example, after you send the command to close a relay, the processor will wait until the relay settles before interpreting and executing the next command. There may be times, however, when you do not want the mainframe processor to wait. For example, you may want to process prior measurement results while waiting for the next channel relay to settle.

The SERIAL Command

The **SERIAL** command enables or disables sequential command execution when opening or closing relays. **SERIAL** has two values: either **ON** or **OFF**. The **ON** value (power-on, default value) causes the HP 3235 to wait until relays have settled from one command before executing the next command. This mode is discussed first.

SERIAL ON

When the HP 3235 receives a command to close a relay(s), it starts the specified relay(s) moving (opening or closing) and then waits for it (them) to settle before accepting a new command. This is true for commands executed from the system controller and the optional control panel. All relays in a given HP 3235 command start moving at the same time. With the serial mode enabled, the following command sequence operates in break-before-make fashion (relay 101 opens before 202 closes):

```
10 OUTPUT 709;"SERIAL ON"
20 OUTPUT 709;"OPEN 101; CLOSE 202"
30 END
```

As another example, if you run the following program:

```
10 OUTPUT 709; "RST; USE 800"          ! RESET AND ASSIGN MULTIMETER
20 OUTPUT 709; "SERIAL ON"             ! TURN SERIAL MODE ON
30 OUTPUT 709; "OPEN 270,370; CLOSE 405,470,491,515,571,592"
40 OUTPUT 709; "TRIG"                  ! TRIGGER THE MULTIMETER
50 ENTER 709;Rdg                       ! ENTER THE READING
60 PRINT Rdg                           ! PRINT THE READING
70 END
```

the HP 3235 opens channels 270 and 370 and then waits for them to settle before closing the specified channels. Once they have settled, the HP 3235 accepts the trigger command in line 40.

SERIAL OFF

When the serial mode is disabled (**SERIAL OFF** command) switching commands are defined as being done as soon as the HP 3235 writes the open/close information to the plug-in module. In other words, the HP 3235 does not wait until the relays have settled before beginning to execute the next command. This allows you to open or close one or more relays and immediately begin executing another command while the relays are settling. For example:

```
10 OUTPUT 709; "SERIAL OFF"
20 OUTPUT 709; "OPEN 101; CLOSE 202"
30 END
```

The HP 3235 begins opening channel 101 and then begins closing channel 202. This approximately halves the settling time since both relays move and settle together.

The multimeter trigger program above with **SERIAL** off may cause the multimeter to trigger before the relays have settled:

```
10 OUTPUT 709; "RST; USE 800"          ! RESET AND ASSIGN MULTIMETER
20 OUTPUT 709; "SERIAL OFF"            ! TURN SERIAL MODE OFF
30 OUTPUT 709; "OPEN 270,370; CLOSE 405,470,491,515,571,592"
40 OUTPUT 709; "TRIG"                  ! TRIGGER THE MULTIMETER BEFORE
45                                     !   RELAYS SETTLE
50 ENTER 709;Rdg                       ! ENTER THE READING
60 PRINT Rdg                           ! PRINT THE READING
70 END
```

The HP 3235 starts the relays specified in Line 30 moving and then, without waiting for the relays to settle, triggers the multimeter.

The SETTLE Command

The **SETTLE** command is useful only when the serial mode is disabled (**SERIAL OFF**). It ensures that all the relays that were started moving by a previous HP 3235 command have settled before the HP 3235 executes another command. In the previous example, you may want to ensure that the relays in line 30 are settled before triggering the multimeter (line 50).

```
10 OUTPUT 709; "RST; USE 800"          ! RESET AND ASSIGN MULTIMETER
20 OUTPUT 709; "SERIAL OFF"            ! TURN SERIAL MODE OFF
30 OUTPUT 709; "OPEN 270,370; CLOSE 405,470,491,515,571,592"
40  OUTPUT 709; "SETTLE"
50 OUTPUT 709; "TRIG"                  !  TRIGGER THE MULTIMETER
60 ENTER 709;Rdg                       ! ENTER THE READING
70 PRINT Rdg                           ! PRINT THE READING
```

Buffering

The HP 3235 accepts commands and data from your system controller in either a buffered or unbuffered mode. Similarly, it can send data to the controller in a buffered or unbuffered mode. At power-on, both input buffering and output buffering are off. The following two sections describe input and output buffering.

Input Buffering

When input buffering is off (power-on state, see the **INBUF** command) and your system controller addresses the HP 3235 to listen, the HP 3235 accepts commands from the controller one character at a time as it is ready. The HP 3235 holds the HP-IB handshake while it is processing a character. After processing the character, the HP 3235 completes the handshake and accepts the next character. When the mainframe processor receives a valid terminator it executes the command. Valid command terminators are semicolons (;), carriage returns (*cr*), line feeds (*lf*), and EOI sent concurrent with the last character sent. Most controllers send *cr* and *lf* after each OUTPUT statement. The first terminator (*cr*) initiates command execution. The final terminator (*lf*) is held off until command execution is complete. Then the HP 3235 accepts the *lf* and frees the HP-IB interface for communication with other instruments. Thus, sending *cr lf* can hold up the system controller until the command finishes execution. The HP 3235 executes HP-IB bus commands, such as Group Execute Trigger (GET) or Device Clear (DCL), in sequence with other characters coming over the HP-IB interface. This means that it is normally not necessary to add programmed delays in the system controller to allow more time for commands to execute.

If input buffering is on and the controller addresses the HP 3235 to listen, it stores incoming characters in a first-in-first-out (FIFO) circular buffer. This technique minimizes bus transfer time and allows your system controller to communicate with other instruments while the HP 3235 is processing commands. When the input buffer becomes full, the HP 3235 holds off bus communication until it can process commands and buffer space becomes available. The HP 3235 input buffer is 256 characters long. To turn the input buffer on, execute:

```
OUTPUT 709; "INBUF ON"
```

In HP-IB systems with multiple instruments, you may need to have the HP 3235 complete all of its commands before sending commands to other instruments (e.g. close a relay and trigger an external voltmeter). If you have the input buffer on, three methods exist for your system controller program to determine when the HP 3235 has completed its command execution.

1. The program can Serial Poll the HP 3235 until the Ready bit (bit 4) is set. Refer to Appendix D for more information.

2. The system controller sends the READY command and reads a number from the HP 3235 mainframe. When the mainframe returns a 1, command string execution is complete. For example:

```
10 OUTPUT 709; "CLOSE 100-132;OPEN 100-132;CLOSE 100-132;OPEN 100-132"
20 OUTPUT 709; "READY"
30 ENTER 709;A
```

```
40 PRINT "READY"
50 END
```

3. Set the RQS mask to enable SRQ with the Ready bit (bit number 4). The system controller must monitor the SRQ line. When the SRQ line becomes true, command execution is complete. Refer to Chapter 7, Interrupt Handling for more information.

With input buffering on, HP-IB bus commands, such as GET or DCL execute immediately.

In either mode, the HP 3235 processor parses the incoming command string into valid commands and parameters. If an error occurs, the processor sets the error status bit (bit 5) in the status register, sets the SRQ bit (if not masked), records an error message, and displays the offending string on the control panel display. The processor resumes parsing after the next command terminator in the input buffer. Refer to the section on Error Handling in Chapter 4 of this manual.

Output Buffering

The HP 3235 sends output data to the same source that originated the command. That is, if the command came from the system controller the data generated by the command is sent to the HP-IB output buffer. When the system controller addresses the HP 3235 to talk, it sends the data in the output buffer. When the output buffer is off (power-on condition), new data overwrites any data left in the output buffer. **OUTBUF OFF** is the power-on state for the output buffer.

The command **OUTBUF ON** changes the output buffer to a FIFO circular buffer. In this mode the HP 3235 sends data in groups and does not overwrite data in the buffer. For example:

```
10 OUTPUT 709; "OUTBUF ON"
20 OUTPUT 709; "CTYPE 000; CTYPE 100; CTYPE 200; CTYPE 300"
30 ENTER 709; A,B,C,D
40 PRINT A,B,C,D
50 END
```

The four plug-in module types are sent in a group. NOTE: depending on what type of controller you are using, you may need to specify an image statement to ignore the terminators between data strings when entering the data. This is because a *cr lf* is sent after each data string.

The command **OUTBUF OFF** returns the output buffer to its overwrite mode. You can erase any data in the output buffer by executing **CLROUT**, **CLR**, or the HP-IB CLEAR commands. For example:

```
OUTPUT 709; "CLROUT"
```

or

```
CLEAR 709
```

In either mode, the output buffer holds up to 2048 bytes. One multimeter reading requires 16 bytes. Therefore, the output buffer can hold up to 128 multimeter readings.

If your system controller addresses the HP 3235 to talk (send data) and the output buffer is empty, the HP 3235 will not respond until output data becomes available.

If the system controller asks for more data than the output buffer can hold (or is currently in the output buffer), the HP 3235 will hold off HP-IB bus communications until the output data becomes available and is sent to the controller. The HP 3235 will not accept any new commands during this time. If the buffer is full when the HP 3235 generates new output data, the HP 3235 processor releases the input buffer (if it was held) and waits for the system controller to read the output buffer. You must read the output buffer or you must execute an HP-IB Device Clear (see Appendix D) before the HP 3235 will process new commands.

Status bit 0 in the Status Byte is the data available bit. The processor sets this bit when output data is available and clears the bit when the output buffer is empty. Refer to Chapter 7 for information on SRQ interrupts using this bit.

EOI Termination

At power-on your HP 3235 uses *cr lf* as the output data terminator. The command **END ON** causes the HP 3235 to send EOI concurrently with the *lf*. (At power-on, **END** is off so EOI is never sent.) Note that in **OUTBUF ON** mode, multiple responses may be in the output buffer with EOI set. This requires multiple reads by an HP Series 200 or 300 system controller (unless an image statement is used to suppress EOI termination) to empty the output buffer. For example:

```
10 OUTPUT 709;"END ON; OUTBUF ON"
20 OUTPUT 709;"CTYPE 000; CTYPE 100; CTYPE 200"
30 FOR I= 1 TO 3
40 ENTER 709; A
50 PRINT A
60 NEXT I
70 END
```

Output Data

Many HP 3235 commands generate output data. This output data goes to the control panel display if the command came from the control panel keyboard or a control panel keyboard initiated subroutine. If the command came from the system controller or from a system controller initiated subroutine, the output data goes to the HP-IB output buffer. You can also store results in the mainframe's internal memory using the **MEM** command. The HP 3235 returns numeric results in either ASCII or binary formats. The default (power-on) format is ASCII (6 digit signed integers for integer results and 8 digit scientific notation for real results). These formats conform to IEEE-728* NR1 and NR3 formats, respectively. The available binary formats are 16-bit, 2'scomplement for integers or IEEE-754* format for 64-bit real numbers.

You select either the ASCII or BINARY output format with the **OFORMAT** command. For example:

```
OUTPUT 709; "OFORMAT BINARY"     ! SELECTS BINARY AS OUTPUT FORMAT
```

Output messages consist of one or more elements. Each element is either a number, a string, or a #A binary (IEEE 728 binary format). In ASCII messages with multiple elements (such as multiple readings from the multimeter), a *cr lf* follows each element.

ASCII Format

At power-on, the output format is set to ASCII. This is the format you will use most often. All HP-IB controllers can communicate in ASCII format.

Binary Format

If your system controller uses 16-Bit, 2's complement integers and/or IEEE 754 Binary representation for internally storing data, you may use the Binary output format. Its primary advantages come if you are transferring a lot of data from the HP 34522 Digital I/O module or readings from the HP 34520 Multimeter, or array data values.

NOTE

Refer to your system controller I/O manual for information on binary data transfers.

Binary data outputs are preceded with an IEEE-728 Block A header and are followed by a *cr lf*. The Block A header consists of four bytes: the pound sign (#), the letter 'A', and two bytes which indicate the number of data bytes to follow. These two bytes represent one 16-bit number; the first byte represents bits 15 through 8, the second byte represents bits 7 through 0.

*IEEE-728 is the IEEE standard for communications protocols.
*IEEE-754 is the IEEE standard for binary floating point representation.

NOTE

This binary format is only useful if your system controller uses the same binary format internally. The HP Series 200 and 300 computers use the same format. Refer to your system controller's operating manual for information on using the binary formats.

The **BLOCKOUT OFF** command disables the Block A header information and *cr lf*. Only the binary data is sent. EOI is sent concurrent with the last character sent if **END ON** is enabled. Refer to the HP 3235 Language Reference Manual for more information. This is the most useful binary mode since few controllers automatically strip off the IEEE-728 #A header and terminator. An HP Series 200/300 computer example of this is:

```
10 DIM A(1000)
20 OUTPUT 709; "DIM A (1000)"
30 ASSIGN @Bin TO 709; FORMAT OFF
40 OUTPUT 709; "OFORMAT BINARY; BLOCKOUT OFF" ! SET BIN MODE, AND
45                                            ! BLOCK HEADER OFF
50 OUTPUT 709; "VREAD A"
60 ENTER @Bin; A(*)                           ! BINARY READ
70 END
```

Stored States

Often you may find it convenient to have a particular, repeatable channel configuration that encompasses several plug-in modules. Opening and closing each individual relay every time you want that configuration becomes cumbersome and time consuming. The **SSTATE** (Store State) and **RSTATE** (Recall State) commands make this task easy by storing states in internal HP 3235 memory. You can also store and recall states from your system controller with the **SET?** and **SET** commands.

Local Storage/ Recall

The hardware state of plug-in modules and the backplane analog and trigger bus switching may be stored in internal HP 3235 memory for later recall. States may also be stored in continuous (nonvolatile) memory if specified. Recalling a stored state is generally faster than sending the separate commands because the state is stored in an internal format and the commands are not decoded each time. States may be stored and subsequently recalled by name for one plug-in moduel, a group of plug-in modules, or the entire HP 3235 system. Only the hardware state of the instrument is stored. I/O modes (such as buffering), memory contents (arrays, variables, and other stored states), and data formats are not stored.

NOTE

When storing/recalling states, the plug-in module type for each slot represented in the stored state must not change when that state is recalled.

Instrument states may be stored until memory is full. The number of states that may be stored depends on the complexity of the module configurations and the number of subroutines or arrays already stored in memory. Purging unneeded states frees memory for other uses (see the **PURGE** command later in this section). You can determine the amount of memory available with the **MEMAVAIL** command described later in this chapter.

If power fails while a state is being stored to continuous memory, the entire continuous memory contents become invalid. At power-on, the continuous memory is erased.

Local State Storage

The **SSTATE** command allows you to store the hardware state of one or more plug-in modules or the entire instrument (mainframe plus extenders) state in internal memory. The command has the form:

SSTATE *name [,slot_list][,***CONT***]*

Stored state names may be up to ten characters long. The first character must be a letter (A-Z) but the remaining nine characters may be either letters, numbers (0-9), the underscore character (_), or the question mark (?). Names must not be the same as reserved HP 3235 commands (e.g. **CLOSE**, **TIME**, **WAIT**), existing subroutine names, or defined variable or array names.

The *slot_list* parameter allows you to store the hardware state of one or more plug-in modules. If you do not specify a *slot_list*, the entire instrument state is stored. Each element in the *slot_list* must be in the form *es00* where *e* is the mainframe or extender number (0=mainframe, 1-7=extender number), and *s* is the slot number (0-9). You can specify a single slot (*es00*), multiple slots (*es00,es00, ...*), sequential slots (*es00-es00*), or groups of sequential slots (*es00-es00,es00-es00*), or any combination.

Normally, the state is stored in volatile memory and is subsequently lost if power is removed from the mainframe. However, if you specify **CONT** with the **SSTATE** command, the state is stored in continuous (nonvolatile) memory and is not lost when power is removed.

Recalling Stored States

The **RSTATE** command recalls stored states from internal HP 3235 memory. The command has the form:

RSTATE *name [,slot_list]*

The *name* parameter must be the same as some stored state. The *slot_list* must have the same number of elements as the specified stored state and the modules must match one-for-one. However, it is not necessary to specify the same slot numbers used in the **SSTATE** command.

The exact module state is recalled. On a given module, for example, if you close a relay that was open when the module state was stored and subsequently recall that module's state, that relay will open. A closed relay that is also closed in the stored state remains closed (it does not open then reclose). Slots not specified in the **RSTATE** command remain unchanged.

Examples

The following examples demonstrate the usefulness of storing and recalling instrument and module states.

Example 1. Storing/Recalling Instrument State. The following program stores the hardware state of the entire HP 3235 system (mainframe plus extenders) in continuous (nonvolatile) memory. Cycle power on the HP 3235 mainframe when instructed. Continue the system controller program to recall the instrument hardware state from continuous memory. You can monitor the state of the modules (with the **MONSTATE** command) to verify that the previous state is recalled.

```
10 OUTPUT 709; "RST"
20 OUTPUT 709; "CLOSE 101,105,202,301,412,506"  ! CLOSE SOME RELAYS
30 OUTPUT 709; "SSTATE STATE1,CONT" ! STORE STATE IN CONTINUOUS MEMORY
40 PRINT "CYCLE POWER ON THE HP 3235"
50 BEEP
60 PAUSE
70 OUTPUT 709; "RSTATE STATE1"  ! RECALL STORED STATE
80 END
```

Example 2. Storing/Recalling a Plug-in Module State. The following program stores the hardware state of one plug-in module in volatile memory. The program then resets the module, closes other relays, and recalls the stored state. Notice that relays closed after the state was stored are open after the state is recalled. Watch the **MON STATE** (if you have the optional control panel) to see the relay activity. The relays closed in line 90 open when the stored state is recalled because none of those relays were closed when the state was stored.

```
10   OUTPUT 709; "RST 200"                ! RESET SLOT 200
20   OUTPUT 709; "MON STATE 200"          ! ENABLE MONITOR MODE
30   OUTPUT 709; "CLOSE 201,204,207,214"  ! CLOSE RELAYS
40   WAIT 2                               ! WATCH MONITOR DISPLAY
50   OUTPUT 709; "SSTATE STATE2,200"      ! STORE STATE OF SLOT 200 MODULE
60   OUTPUT 709; "RST 200"                ! RESET SLOT 200 TO OPEN RELAYS
70   OUTPUT 709; "MON STATE 200"          ! RE-ENABLE MONITOR MODE
80   WAIT 2                               ! WATCH MONITOR DISPLAY
90   OUTPUT 709; "CLOSE 203,206,210"      ! CLOSE OTHER RELAYS
100  WAIT 2                               ! WATCH MONITOR DISPLAY
110  OUTPUT 709; "RSTATE STATE2,200"      ! RECALL STATE AND WATCH MONITOR
120  END
```

Remote Storage and Recall

Two commands enable you to store and recall the hardware instrument state for your HP 3235 system (mainframe plus extenders) in your system controller. The **SET?** command sends the entire instrument state to the system controller. The **SET** command returns the stored state from the system controller and restores the HP 3235 to that state.

The **SET?** command sends the state information to the system controller in IEEE-728 Block A format. This includes the header and the data block followed by carriage return and line feed (*cr lf*). Block A format consists of four bytes: the "#" sign, the letter "A", and two bytes which indicate the number of bytes of binary data to follow (not including the *cr lf*).

The following program loads the hardware state of the mainframe, extenders, and all installed plug-in modules into the system controller (**SET?** command). The state is then loaded from the system controller back to the HP 3235 using the **SET** command.

```
10   DIM State$[1000]                     ! DIMENSION ARRAY
20   OUTPUT 709; "RST"                    ! RESET THE HP 3235
30   OUTPUT 709; "CLOSE 201,204,207,214"  ! CLOSE RELAYS
40   OUTPUT 709; "MON STATE 200"          ! ENABLE MONITOR MODE
50   OUTPUT 709; "SET?"                   ! DUMP STATE TO CONTROLLER
60   ENTER 709; State$                    ! ENTER STATE
70   PRINT "1. REMOVE POWER FROM HP 3235"
80   PRINT "2. RESTORE POWER TO HP 3235"
90   PRINT "3. PRESS THE 'CONTINUE' KEY ON SYSTEM CONTROLLER"
100  PAUSE
110  OUTPUT 709; "RST"                    ! RESET THE HP 3235
120  OUTPUT 709; "MON STATE 200"          ! RE-ENABLE MONITOR MODE
130  WAIT 2                               ! WATCH MONITOR DISPLAY
140  OUTPUT 709; "SET"; State$            ! LOAD STATE FROM CONTROLLER
150  END
```

Memory Management

Your HP 3235 has approximately 480K bytes of volatile memory and approximately 6K bytes of continuous (battery-backed) memory. An additional 2K bytes of continuous memory is reserved for redefined control panel keys and the ten User keys. The actual amount of available memory depends on the number of plug-in modules installed in the HP 3235 system and other factors. This section shows you how to efficiently use the local memory in your HP 3235.

Memory Usage

Volatile memory constitutes the largest block of internal user memory. This memory is not battery-backed and hence any information stored in this memory is lost if the HP 3235 loses power. The volatile memory stores the following information:

1. Variables and Arrays
2. Subroutines
3. Stored States
4. Plug-in Module State records (i.e., which relays are closed, etc.)
5. Information for the **ON INTR** blocks
6. Expanded Prohibited Channel lists
7. Temporarily stores the expanded channel lists during a scan

The continuous memory stores the following information:

1. Prohibited channel lists as you defined them (e.g., 101-113).
2. Stored States if you specify **CONT** (see "Stored States" earlier in this chapter).

NOTE

It is possible to lose the contents of the continuous memory if power fails while the HP 3235 is executing a command which stores data in continuous memory. Therefore, all prohibited channel lists and stored states should also be stored in the system controller as a back-up.

The **MEMAVAIL?** command returns the size (in bytes) of the largest available block of volatile memory and the available size of continuous memory. **MEMAVAIL?** returns two REAL numbers separated by commas. The first number is the largest continuous block of volatile memory. The second number is the size of the available continuous memory.

The following program example shows how to use the **MEMAVAIL?** command. The first value printed is volatile memory result (in bytes) and the second value is continuous memory result (in bytes).

```
10  DIM A$[60]
20  OUTPUT 709; "MEMAVAIL?"
30  ENTER 709; A$
40  PRINT A$
50  END
```

Using Memory Efficiently

When you purge an item from volatile memory a 'hole' remains where that data was stored. The HP 3235 does not pack the remaining data. However, two adjacent holes are combined into one large one. You can fragment the volatile memory by leaving several 'holes' in the memory. The **MEMAVAIL?** command returns the size of the largest hole in the volatile memory. The following suggestions can help you avoid fragmenting the memory and thus ensuring the maximum amount of available memory. The HP 3235 will not allocate space for an item (such as an array) over multiple holes; it must have one contiguous block.

1. Define the items that are more permanent (i.e. prohibited channels, stored states, variable names, subroutines, etc.) first. Define the items that are subject to change (i.e. arrays that will be redimensioned and nonpermanent subroutines) last.

2. Purge items in the opposite order that you created them. Two adjacent holes are combined into one large one.

3. Download and compress (see Chapter 8 in this manual) each debugged subroutine individually rather than downloading several subroutines and then compressing them. If a subroutine consists of more than 100 lines, consider breaking it down into 2 or 3 smaller subroutines.

4. Use INTEGER arrays where possible. They use less memory than REAL arrays.

Other Memory Related Commands

Chapter 7 in this manual shows how to create and use variables and arrays. The **FILL** command loads specified values into arrays. The **MEM** command stores numeric data generated by other commands into variables or arrays. Several other HP 3235 commands directly affect memory usage or identify the user defined contents of memory. These commands are described in the following paragraphs.

CAT

The **CAT** command (catalog) lists all the names of the subroutines, simple variables, stored states, and arrays that are presently stored in mainframe memory. **CAT** also displays the number of bytes required to store each subroutine, and the type and number of elements in each array. If a subroutine has been defined, and then deleted using **DELSUB**, it is listed with a size of zero. If there are no more arrays or subroutines to be listed, the HP 3235 returns the word "DONE".

The format for the catalog is:

for subroutines:	*SUB*	*sub_name*	*SIZE dddd TEXT SIZE dddd*	
for integer arrays:	*IARRAY*	*array_name*	*size*	
for real arrays:	*RARRAY*	*array_name*	*size*	
for stored states:	*CONT*	*state_name*		(nonvolatile memory)
	STATE	*state_name*		(volatile memory)
simple variables:	*INT*	*variable_name*		
	REAL	*variable_name*		

Subroutines return two sizes. The first **SIZE** is the size, in bytes, for the compiled code. This is the HP 3235 machine code version of the subroutine. The second

TEXT SIZE is the size, in bytes, for the subroutine source code which you can list and single-step. When you compress a subroutine (see **COMPRESS** later in this section), the **TEXT SIZE** goes to 0 but the machine code **SIZE** remains the same. The size returned for arrays is the actual number of elements in the array. The following program shows how to use the **CAT** command:

```
10 DIM A$[80]
20 OUTPUT 709; "CAT"
30 ENTER 709; A$
40 PRINT A$
50 IF A$<>"DONE" THEN 30
60 END
```

COMPRESS

The **COMPRESS** *name* command leaves the subroutine in its machine code form. This saves internal memory but eliminates the ability to list or single step the subroutine.

DELSUB

The **DELSUB** *name* command deletes *only* the specified subroutine from mainframe memory. It does *not* delete the subroutine name itself from the catalog of subroutines.

PURGE

PURGE deletes a specified stored state from HP 3235 memory (either continuous or volatile). However, the state name is shown in the **CAT** listing as PURGED *name* and cannot be redefined.

SCRATCH

The **SCRATCH** command deletes (scratches) *all* subroutines, variables and arrays, and stored states from mainframe memory. It also deletes all name definitions from the catalog. **SCRATCH** gives an error but does not purge the memory if it is executed while a subroutine is executing. The optional **CONT** parameter deletes only those states stored in continuous memory.

HP 3235 Internal Task Priority

Four independent tasks affect command execution in the HP 3235. In the order of highest priority first, these tasks are:

1. Control Panel keyboard executed commands.

2. System Controller (via HP-IB) executed commands.

3. The **ON INTR CALL** *sub_name* and **ON TIMER***n* **CALL** *sub_name* executed subroutines (see Chapter 10).

4. **RUN** executed subroutines.

Four rules govern how the HP 3235 executes commands (or subroutines) from these four command sources. These rules are:

1. Any task can only execute one command or subroutine at a time.

2. The HP 3235 always executes the highest priority task that has something to do.

3. Any task that executes a wait command of >30mS allows the next lower priority task to execute some or all of its command(s).

4. If a task requires a resource (such as an analog backplane or trigger bus) that another task is using, the second task waits until the first task finishes with the resource. This is true even if the second task is a higher priority task.

These simple rules of priority and shared resources will help you predict what the HP 3235 will do in most circumstances. For example, they help explain why only one subroutine can be executing at any time (see Chapter 8) and why you can enter commands from the control panel keyboard and the HP-IB at the same time (see Chapter 4).

Chapter 12

Signal Integrity

Introduction

This chapter presents guidelines for maintaining signal integrity in your test system. Your test system configuration is a function of what you are testing, and how often you change the test system or the device(s) being tested. It may also depend on the required system throughput. For example, if you are testing DC power supplies, short cable lengths are not vital to the test. However, if you are testing VHF receivers, short lead lengths between the receiver and the test instruments are often crucial. If you are testing several different types of receivers, you may want to use the Quick Interconnect Fixture and short test lead lengths. Consequently, you may need to compromise the test system design to maximize signal integrity while maintaining test throughput and ease of use. This chapter shows you how to maintain signal integrity.

This chapter is not intended to provide complete and in-depth information on signal integrity. It does provide guidelines to help you in designing your test system. For additional information, refer to the excellent books and articles listed in the bibliography at the end of this chapter.

General Information

The following information provides general guidelines for using the HP 3235 in your test system. Refer to the specifications for a particular plug-in module for information on relay contact resistance, thermal offset, environmental conditions, etc.

Multiplexer Configurations

The multiplexer configuration is most commonly used for signal switching into instrumentation. Generally in this signal switching configuration, only one signal is connected at a time to the measuring device and the switching is break-before-make; i.e., the input is disconnected before a new input is connected.

Multiplexing configurations are used primarily to connect multiple signals to a single output such as a number of thermocouples to a multimeter or a number of test points to an oscilloscope. the main advantage of this configuration is economy.

Two wire multiplexers are useful for floating measurements. Inductive coupling can cause the generation of ground loops in the low lead. To break these loops it is necessary to switch the low and high leads. This is particularly useful for capacitance and inductance measurements at frequencies below 1 MHz. Also, devices such as thermocouples and transducers can be connected to voltmeters with good common mode noise rejection.

Guarded measurements shunt common mode noise current away from the measurement input.

Single ended multiplexer configurations are useful for high level signals where measurement accuracy is not critical and where all the measuring instruments share a common ground with all the signal sources to be measured. See the HP 34501, 34502, 34507 Programming and Configuration Manual for an example of single ended measurements.

Heating of Adjacent Modules

Any plug-in module dissipating more than 10 watts may affect the thermal offsets on adjacent modules. For example, assume you have an HP 34523 Breadboard module with voltage regulator circuits next to an HP 34502 multiplexer module switching low level signals. Heat from the regulator circuits cause thermal offset voltages on the adjacent multiplexer module. These offset voltages may adversely affect the low level signals. Therefore, keep modules switching low level signals away from modules with high power dissipation.

Signal Levels

Once a relay is used for power switching (high voltage and high current), its contact resistance may increase to a few ohms. This may make the relays unsuitable for switching some low level signals. This includes the bank and backplane relays as well as the front end multiplexer relays. We recommend that you do not mix high level and low level signal switching on the same relay module. Instead, consider designating some relay modules for switching high voltages and current and other modules for low level signals.

Coaxial Cables

For the best high frequency performance, user cables for the BNC-type plug-in modules (HP 34504, 34505, and 34506) should have at least two braided shields or one braid and a foil wrap. To maintain low DC offset voltages, cables should have copper center conductors, not copper-clad steel. RG-223/U cable is recommended. Table 12-1 lists the 50Ω cables (BNCs on both ends) available from Hewlett-Packard.

Table 12-1. HP Coaxial Cables

Length	Part Number
30 cm (12 in)	HP 8120-1838
61 cm (24 in)	HP 8120-1839
122 cm (48 in)	HP 8120-1840

Sources of Electronic Noise

The measurement of low level signals in a test system environment can best be accomplished with careful attention to the details of grounding and shielding. The following are major offenders of noise coupling into these systems:

1. Conductively coupled noise
2. Coupling through a common impedance
3. Electric and magnetic fields

Besides these sources of noise, some systems are sensitive to noise from galvanic action, thermocouple noise, electrolytic action, triboelectric effect, and conductor motion. In electronic test systems designed for the testing of electronic modules, the important noise sources are generally:

- **Conductively Coupled Noise.** The easiest way to couple noise into the circuit is on the conductor leading into the circuit. A wire running through a noisy environment has an excellent chance of picking up unwanted noise and transferring it to the circuit. Major offenders are often the power supply leads connected to the circuit.
- **Coupling Through a Common Impedance.** Common impedance coupling occurs when currents from two different circuits flow through a common impedance (e.g. a ground loop). The ground voltage of each is affected by the other. As far as each circuit is concerned, the ground potential is modulated by the ground current flowing from the other circuit in the common ground impedance.
- **Electric and Magnetic Fields.** Radiated magnetic and electric fields occur whenever an electric charge is moved or a potential difference exists. In a circuit, high frequency interference may be rectified and appear as DC errors. Particularly, radio and TV broadcasts in close proximity to the circuit of interest will tend to cause this kind of problem. Extremely strong fields can cause non-linear behavior of active circuitry without being suspected. Hence, it is imperative to shield sensitive circuitry from these fields.

Sources of Noise in a Switch System

Of particular importance to the test engineer is knowing the source of the unwanted noise in his system. A simple list of rules may be helpful in reducing unwanted noise, but it is seldom enough for an understanding of where the noise is coming from.

In many cases the noise culprit is the adjacent channel in the switch system. A look at the simplified equivalent circuit in Figure 12-1 shows that most of the capacitance in the switch system resides across the switch contacts and between the adjacent conductive paths. Since noise coupling is a function of area and proximity, a simple way to reduce the coupling is to separate the switches and the conductors from each other. Another method, as shown in Figure 12-2, is to short every other conductor to ground. This is not practical in many cases. Increasing the switch density is a desirable objective; it gives you more capability in a smaller package. Further, the systems being tested today tend to be much more complex and have larger point counts than ever before. Hence, you are faced with a dilemma of having to increase the distance between channels and component density at the same time.

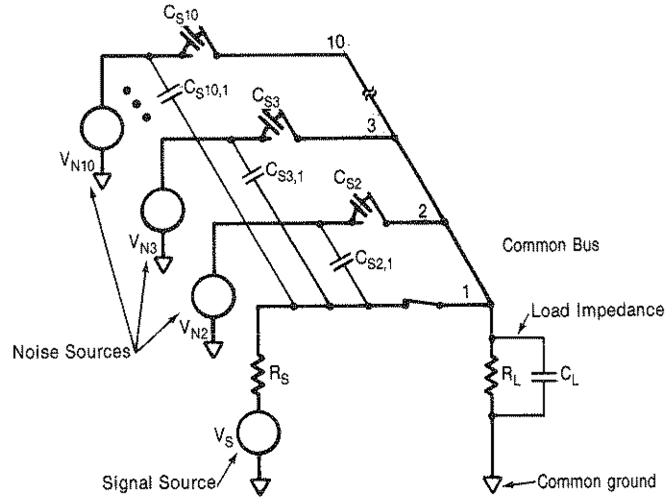


Figure 12-1. Noise Coupling

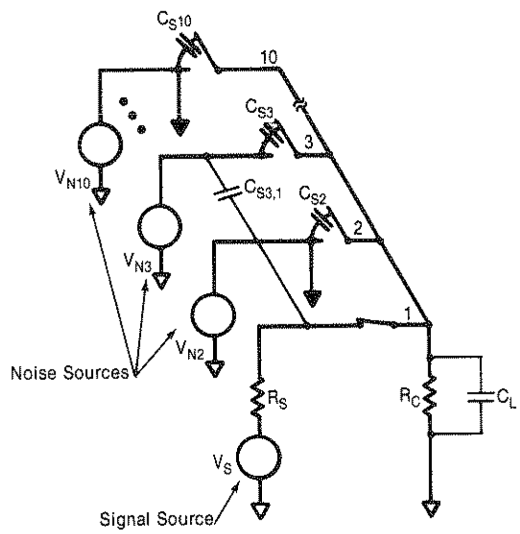


Figure 12-2. Shorting Every Other Conductor

One solution to this problem is to connect the large amplitude signals as far away as possible from the low level signal leads. In addition, ground leads should be placed between the signal leads to provide a convenient path for the interfering noise.

Noise Reduction

Noise reduction techniques apply equally to single point systems and multipoint systems. Let's look at the following topics:

- Shielding
- Grounding
- Isolation

Shielding

Shielding test systems against noise involves shielding against both capacitive (electric) and inductive (magnetic) coupling in the system.

Shielding Against Capacitive Coupling

Capacitive noise coupling between channels is both the easiest to understand and shield against in a test system. As Figure 12-3 shows, the coupling of noise from one conductor to the other is a result of the capacitance between the conductors. In most cases, the load impedance R is much less than the impedance of the stray capacitance C_{12} plus C_{2G}. Therefore, when

$$R \ll \frac{1}{j\omega(C_{12} + C_{2G})}$$

the noise voltage can be determined by

$$V_N = j\omega RC_{12}V_1$$

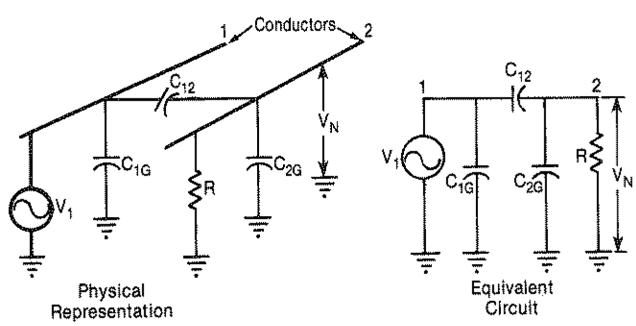


Figure 12-3. Capacitive Coupling Between Conductors†

† Ott, H., "Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

If you could reduce the capacitance, C_{12}, then the noise voltage coupled into the conductor would be reduced. Physical separation will accomplish up to about 8dB of noise reduction for distance of 40 times the diameter of the conductor. More separation between the conductors has little effect. Figure 12-4 shows the frequency response of capacitively coupled noise voltage.

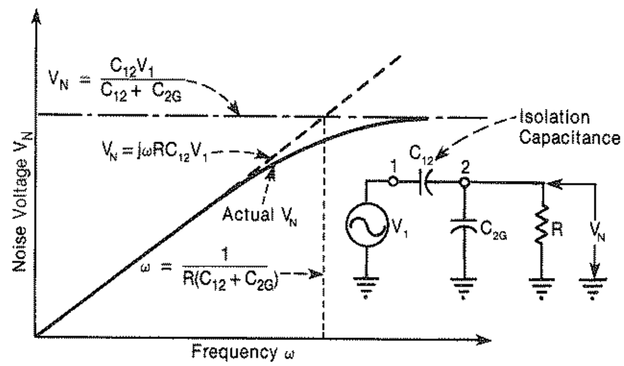


Figure 12-4. Capacitive Coupling Frequency Response†

Another alternative is to operate at lower resistance. Loading or mismatching may make this unfeasible. Surrounding the signal conductor with a grounded shield with uniform electric field (the geometry of the shield is smooth) reduces the noise voltage on the signal conductor to zero provided the conductor does not extend beyond the shield at high frequency. If the shield is not grounded, then the noise voltage picked up by the center conductor is equal to the noise voltage on conductor 1. The shield merely couples the noise voltage as shown in Figure 12-5.

† Ott, H., Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

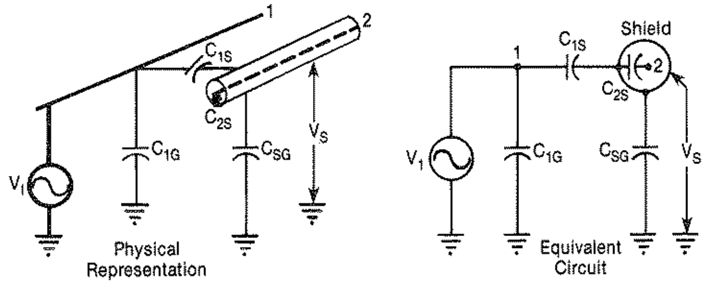


Figure 12-5. Unguarded Shield Capacitive Coupling†

If the center conductor is exposed beyond the confines of the ungrounded shield, and it has a finite resistance to ground, then the circuit is as shown in Figure 12-6. Again the noise voltage reduces to

$$V_n = j\omega RC_{12}V_1$$

Now, however, C_{12} is greatly reduced because of the shield. If the shield is braided or not uniform then the effect of the holes in the shield must be included in C_{12}.

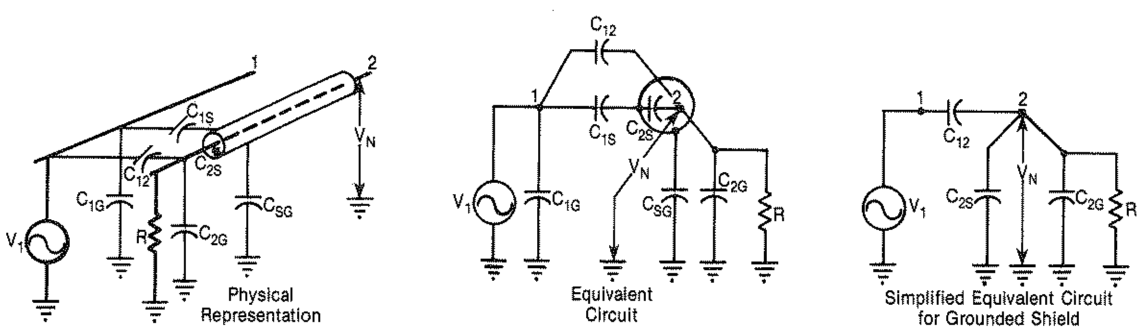


Figure 12-6. Capacitive Coupling†

† Ott, H.,"Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976.

Shielding Against Inductive Coupling

Inductive coupling is much more difficult to eliminate than capacitive coupling. In fact, the above illustration for excellent electric field shielding has no effect on inductively coupled noise.

An important derivation for inductive coupling is that the mutual inductance between shield and the center conductor equals the shield inductance.

$$M = L_s$$

In this case, the two conductors, the shield and the center conductor, do not even have to be coaxial. From the circuit diagram in Figure 12-7 and the fact that $L_s = M$, we can derive the following relationship:

$$V_N = \frac{j\omega V_s}{j\omega + \frac{R_s}{L_s}}$$

The 3dB point on the graph of V_N versus frequency is defined as the cutoff frequency of the shield, ω_c. It can be shown that this frequency is $\omega_c = R_s/L_s$.

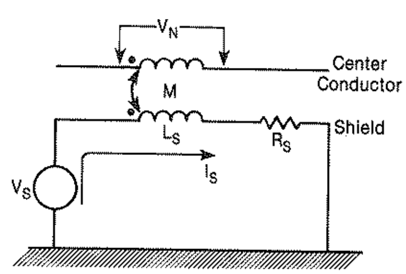


Figure 12-7. Equivalent Circuit of Shielded Conductor†

† Ott, H.,"Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976.

If current is allowed to flow in the shield by connecting both ends of the shield to ground, at 5 times above the cutoff frequency, ω_c, most of the noise current flows in the shield and cancels the noise current flowing in the center conductor. Below this frequency the ground plane carries most of the noise. Hence, for audio and higher frequency applications the shield should be grounded at both ends. The noise reduction is a result of the field generated by the current in the shield that cancels the conductor's field and has little to do with the magnetic properties of the shield material.

Shielding Against Magnetic Radiation

Since inductive noise coupling is proportional to the loop area cutting the magnetic flux of the noise, receiver circuits can be best protected against magnetic fields by decreasing the area of the receiver loop. See Figure 12-8A.

Referring to Figure 12-8C, a shield placed around the conductor does not change the loop area and, therefore, provides no magnetic shielding. In Figure 12-8B, the circuit provides excellent magnetic field protection above the cutoff frequency, ω_c; however, below cutoff frequency most of the current in it will produce an IR drop in the shield and appear in the circuit as a noise voltage. If there is a difference in ground potential between the two points in the ground plane, this too will produce noise voltage in the circuit. Whenever a shield is used in a circuit and is grounded at both ends of the circuit, only limited low frequency magnetic field protection is possible because of large noise currents induced in the ground loop.

For maximum protection against induced noise at low frequencies, the shield should not be one of the signal conductors and one end of the circuit must be isolated from ground.

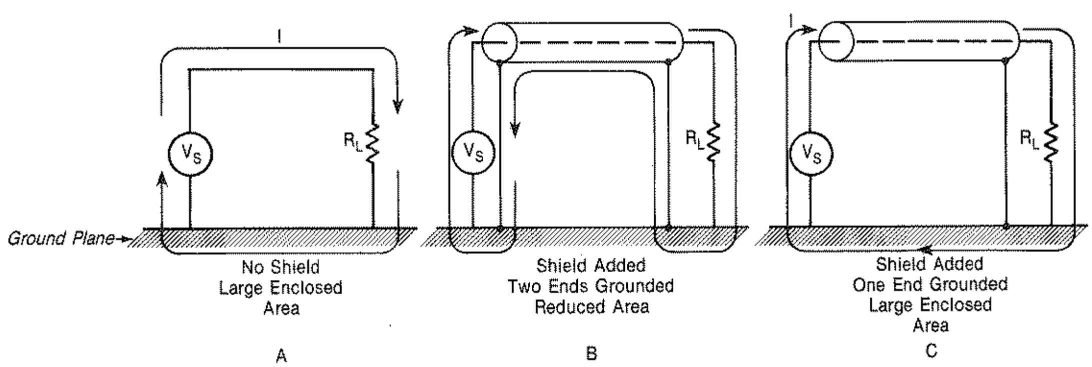


Figure 12-8. Effect of Shield on Receiver Loop Area†

† Ott, H.,"Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

Coaxial Cable Versus Twisted Pair

Twisted pairs, where the wires are twisted together to reduce the loop area, and shielded twisted pairs should be used for applications up to a few hundred kHz. Above this frequency these cables are prone to signal loss. By contrast, the impedance of coax cable is relatively uniform from DC to VHF (30 MHz to 300 MHz) frequencies.

A coaxial cable with its shield grounded at one point provides a substantial amount of protection from capacitive pickup. A double shielded or triaxial cable with insulation between the two shields provides the maximum protection against noise coupling. Since the noise current flows through the outer shield and the signal return current flows through the inner shield, the two currents do not flow through a common impedance for noise coupling. Fortunately (since triaxial cables are expensive and awkward to use), above 1 MHz skin effect on the shield of coaxial cable tends to simulate triaxial cable. The noise current will flow on the exterior of the shield and the signal return current will flow on the interior of the shield.

Up to a few hundred kilohertz the shielded twisted pair cable has performance comparable to triaxial cable without the expense and awkwardness. An unshielded twisted pair, unless it is used in a balanced circuit, offers little protection against capacitive pick-up.

Grounding

Signal ground has two definitions:

1. A signal ground is an equipotential circuit reference point for a circuit or a system.

2. A signal ground is a low impedance path for current to return to the source.

The first definition is the classical interpretation of the idealized ground. The second emphasizes the realities of the IR drops that can occur in the ground plane and couple noise into the signal conductor.

The design objectives of a grounding system are:

1. To minimize the noise voltage generated by currents from two or more circuits flowing through a common ground impedance.

2. To avoid creating ground loops that are susceptible to magnetic fields and differences in ground potential.

NOTE

Improper Grounding can be a Primary Noise Source

Grounds can be divided into two parts; safety and signal grounds. Safety grounds are provided so that a breakdown in impedance between the equipment chassis and the high voltage line of the power line will result in a low impedance current path to ground.

Figure 12-9 shows that the instrument chassis should be grounded for safety. Otherwise, it may reach a dangerous voltage level through stray impedances or insulation breakdown. In the HP 3235, if you remove the isolator shorting bar (see Chapter 2), the chassis must not float more than 42V above earth ground.

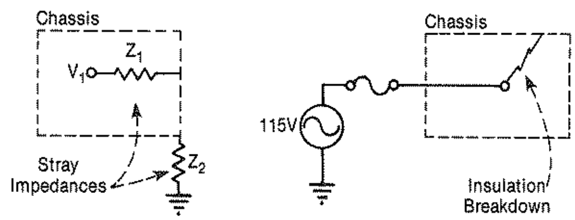


Figure 12-9. Instrument Chassis Ground†

Figure 12-10 shows a standard 3-wire 115V AC distribution circuit. Do not defeat the third wire safety ground.

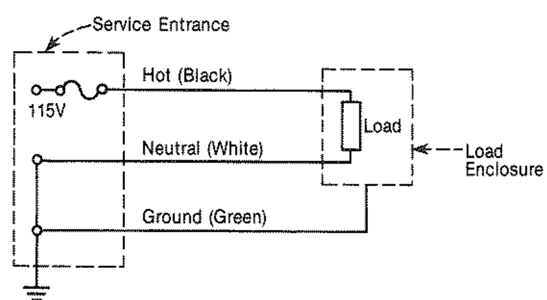


Figure 12-10. Standard 115 AC Power Distribution†

† Ott, H.,"Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

Single Point Ground Systems

Signal grounds are either single point or multipoint grounds. Single point grounds can be connected in series or parallel in systems. For noise coupling the least desirable grounding scheme is the series ground, but it is also the least expensive and the easiest to wire. It is therefore the most widely used. For non-critical applications the series ground may work satisfactorily. The circuits most sensitive to noise should be placed nearest to primary ground point.

Figure 12-11 shows that a common series ground connection. It is undesirable from a noise standpoint but has the advantage of simple wiring.

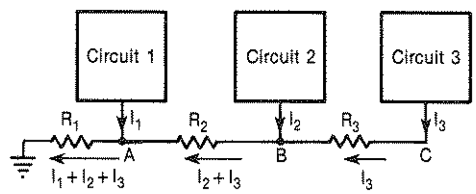


Figure 12-11. Common Ground System†

Figure 12-12 shows a separate, parallel ground system. It provides good low-frequency grounding, but is mechanically cumbersome. This parallel ground system is the most desirable configuration for frequencies in the audio range. This configuration eliminates cross coupling between ground currents through different circuits. The scheme, which is not as simple as the series connections, is awkward to wire and use.

† Ott, H., "Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

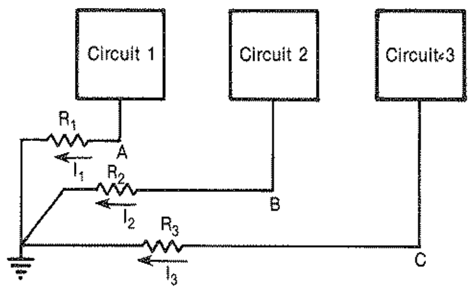


Figure 12-12 Separate Ground System†

At high frequencies the parallel single point connection is very limited. It is here that the inductance of the ground conductors increases the ground impedance and also produces capacitive coupling between the ground leads. This situation becomes worse as the frequency increases to the extent that the ground leads begin to act like antennas and radiate noise. As a rule, ground leads should always be less than 1/20 of a wavelength to prevent radiation and maintain low impedance.

Multipoint Ground Systems

For high frequency applications the multipoint grounding system should be used. Typically, the circuits are connected to the nearest available low impedance ground plane, normally the system chassis. however, just as parallel single point configurations are not effective for high frequencies, multipoint ground systems should be avoided at low frequencies. In multipoint ground systems all the ground currents from all the circuits flow through a common ground impedance - the ground plane, as shown in Figure 12-13. This ground system is typically a good choice at frequencies above 10 MHz. Impedances R_1 - R_3 and L_1 - L_3 should be minimized.

† Ott, H., "Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

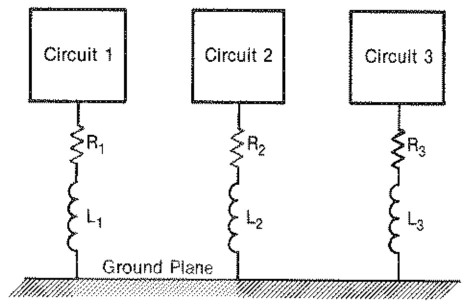


Figure 12-13. Multipoint Ground System†

Grounding Summary

Normally, at frequencies below 1 MHz a single point ground system is preferred; above 10 MHz the multipoint ground is best. From 1 to 10 MHz, a single point ground system can be used as long as the length of the longest ground return is less than 1/20 of a wavelength. If this not possible, then a multipoint should be used.

For practical reasons, most systems require three separate ground returns. Signal grounds should be separate from hardware grounds and noisy grounds like relay and motor grounds. In sensitive systems, for example, separating the signal grounds into low level and digital grounds prevents the higher level, much noisier digital signals from coupling into the low level leads. If AC power is distributed throughout the system, the power ground should be connected to the chassis or hardware ground.

Figure 12-14 shows three classes of grounding connections that should be kept separate to reduce noise coupling.

† Ott, H., "Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

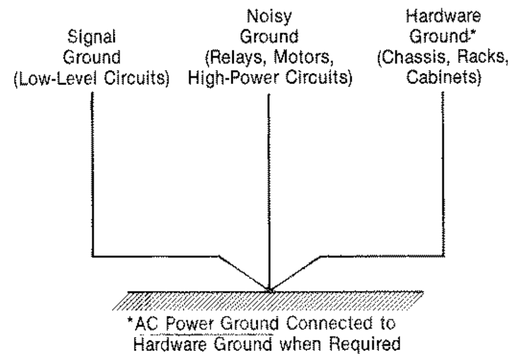


Figure 12-14. Three Classes of Grounding†

A single ground reference system should always be used for low level work. If it is not used, any differences in ground potential will show up as noise on the signal path. As shown in Figure 12-15, a measurement system with a return path through two grounds will produce a noise voltage across the measurement terminals of the instrument. Disconnecting this path on the instrument side and allowing the instrument to float its low terminal above ground effectively opens the low frequency connection and greatly reduces the common mode voltage coupling into the circuit.

Figure 12-15 shows that inside the ideal floating voltmeter the impedance between low and ground is extremely high at DC. The insertion of Z_2 in the ground loops creates a voltage divider on the noise presented to the measurement terminals. The common mode noise is attenuated by a factor of R_b/Z_2.

† Ott, H., "Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

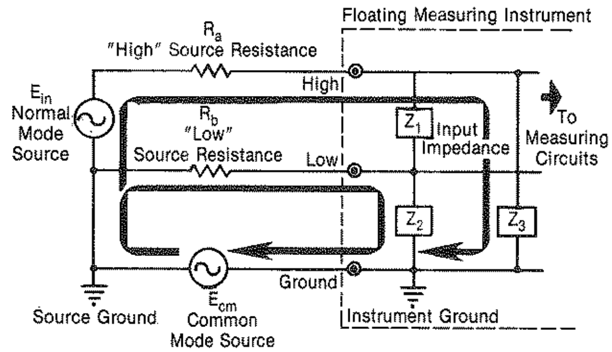


Figure 12-15. Floating Multimeter

Grounding Shielded Cable

Shields on cables used at frequencies less than 1 MHz should only be connected to a ground at one point. In the the case of shielded twisted pair, to connect ground at more than one point may inductively couple unequal voltages into the signal cable and be a source of noise. The IR drop in the shield of a coaxial cable will couple into the signal conductor, and is a problem if the shield is connected to ground at more than one point.

Since we must connect ground somewhere, where should that be?

1. For a grounded load and an ungrounded source, the shield should always be connected to the load common terminal even if this point is not earth ground.

2. For an ungrounded load and grounded source, the shield should be connected to the source common terminal, even if it is not at earth ground.

3. If both source and load are grounded, we have no choice but to ground the shield at both ends. This is the least desirable of all circuit configurations for noise reduction.

Figure 12-16 shows the proper connections for shielded twisted pair and coaxial cables at low frequency. The grounds may have large potential differences which can cause ground loops and couple noise into the circuits.

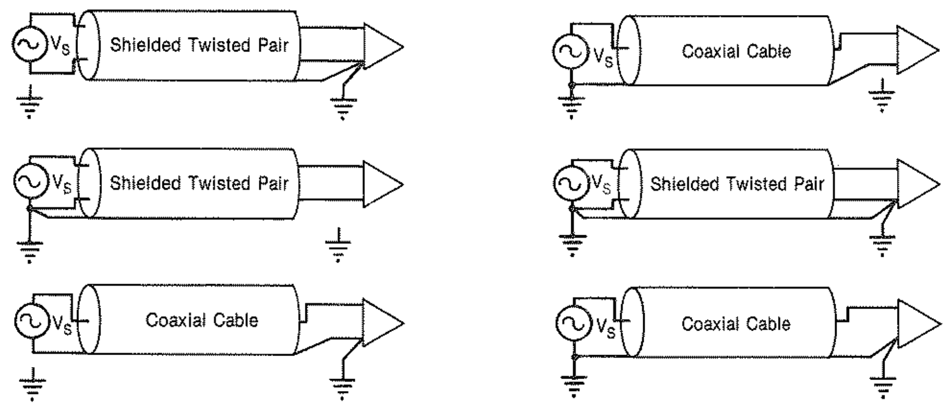


Figure 12-16. Low Frequency Grounding Schemes†

Up to this point we have discussed shielding only at low frequencies. Shield grounding at high frequencies, above 1 MHz, requires more than one connection to ground to guarantee that the shield remains at ground potential. Besides, capacitive coupling between the shield and ground plane tends to complete the circuit at high frequency and makes the shield isolation difficult to impossible. At high frequency, it is common practice to ground the shield at every 1/10 wavelength to assure good grounding.

If there is any question about the ground in a system, it is better to isolate the questionable ground and find some other way of making a good ground connection.

A hybrid ground formed by inserting small capacitors every 1/10 wavelength provides a single ground at DC and multiple grounds at high frequency. Further, its performance can be characterized reliably.

† Ott, H.,"Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

Isolation

Ground loops in very noisy environments can be broken by the use of isolation transformers as shown in Figure 12-17. In cases where a DC path must be maintained a longitudinal choke can be used to suppress higher frequency loops.

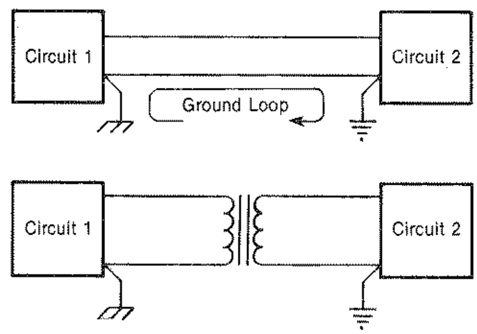


Figure 12-17. Ground Loops†

Other Methods of Reducing Ground Coupled Noise

Other methods of reducing the effects of ground loop noise apply directly to the measuring receiver load. The receiver can be configured to minimize the effect of ground loops by using a balanced input. Even more noise rejection can be obtained at low frequency with a guard shield around the load.

Noise Reduction Techniques in Switching Systems

The interface between your DUT and switching system provides an excellent opportunity for inadvertently introducing noise into your system. Follow these simple rules for the cable harness:

1. High and low level leads should not share the same cable harness.
2. Shield integrity should not be broken through the system. For low frequencies, the shield must be insulated to avoid grounding at some other point in the system.

For Connectors:

1. High and low level leads should be separated on the connector.
2. Ground leads should be placed between the signal leads where possible.
3. Any unused connector pins should remain between the signal leads and be grounded.

† Ott, H., "Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

4. Cable shields should each have their own connector pin through the connector.

The sources of noise in the switching system are those generated internally by the circuitry driving the switches, by thermal unbalance across the switches, by noise coupling from other conductors in the system, and by noise generated outside the system. Thermal unbalance can be minimized by a mechanical design that assures that all the contacts in the relay are subject to the same temperature gradient across the leads and by using latching relays. In this case the major source of thermal generation, the coil energy, is removed so that the relay contacts are always in thermal equilibrium.

Noise coupling from adjacent channels into the measurement channel(s) presents a great danger to signal integrity. Proper shielding and grounding techniques can remove many of these problems in hard-wired systems, but when the signal must be selectively switched into an oscilloscope, counter, or some other measuring instrument, the problem can become acute.

Tree Switching

Tree switching is used to reduce the stray open switch capacitance seen in large systems as a result of connecting the unused relays in parallel in the system. This capacitance is reduced by introducing a relay in series with the input relays. For a 20-channel multiplexer this series switch effectively reduces the stray capacitance seen by the measuring circuit to 1/20 the value without the tree switch. The result is less cross talk and faster measurement settling time for the system. This is shown in Figure 12-18.

NOTE

Figure 12-18 shows a general purpose 32 channel multiplexer. The HP 34501, 34502, and 34507 Relay Multiplexers are similar in design. The backplane relays act as "Tree Switches", and the bank relays isolate banks of relays. e bank relays.

Other ways of reducing open switch capacitance, and therefore coupled noise, are to either make the switch and the switch contact gap large or to make the contact area extremely small. HP products which have these characteristics are the HP 33311-13 microwave switches and the HP 59307A VHF switch. The HP33311-13 employ a very long throw in its switching action and is housed in a precisely milled enclosure to ensure signal integrity to 18 GHz and beyond. The 59307A uses a very small contact area that reduces the capacitance across the switch. This is an economical approach but it does limit the amount of current that you can pass through the switch. Insertion loss, a function of the resistance of the switch, tends to be high.

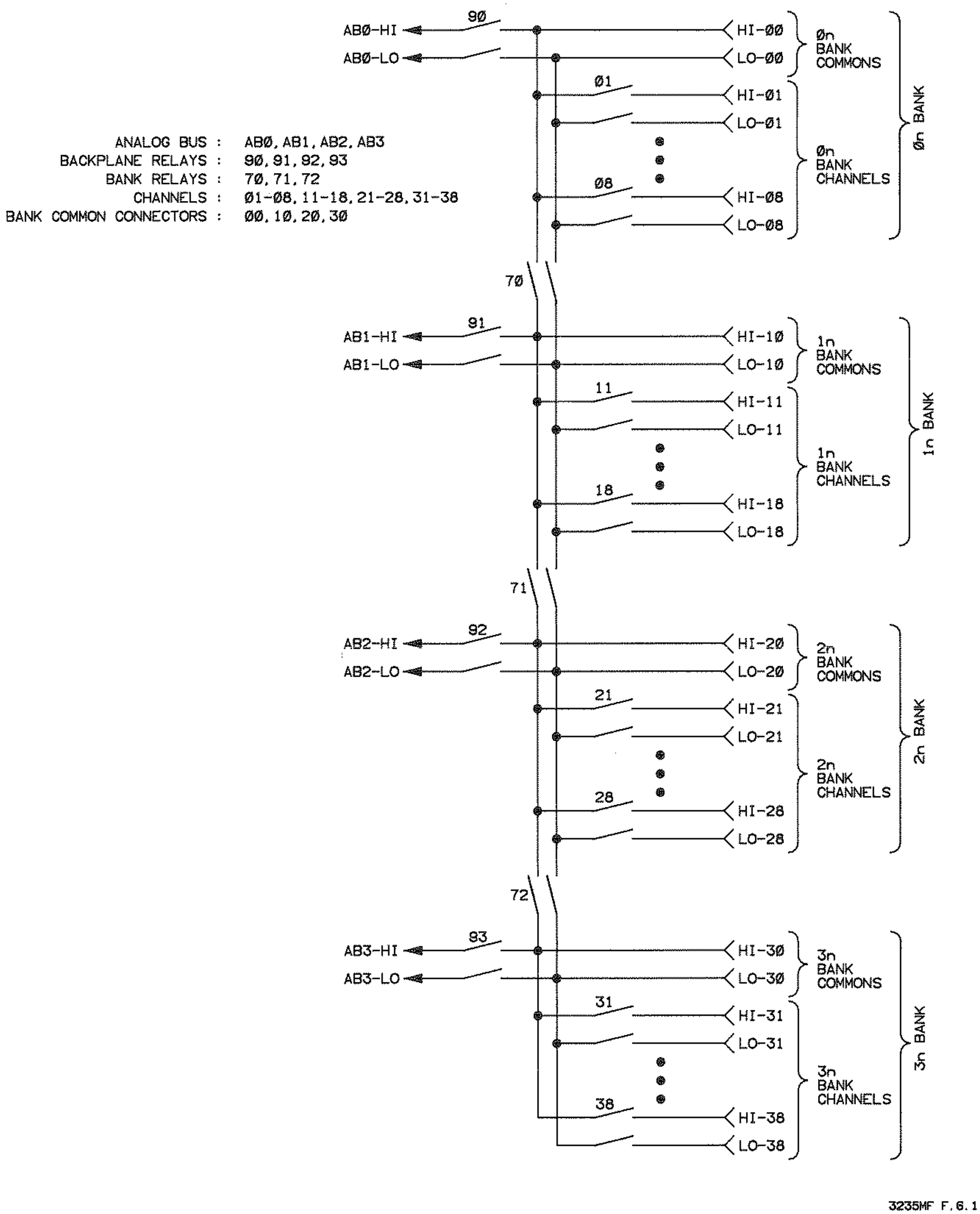


Figure 12-18 Tree Switching

Impedance Mismatching

Impedance mismatching in switching systems can cause many unpredictable measurements results if the signal is either a high frequency cw waveform or is a digital signal with fast transition times. The reflected wave at the point of the impedance mismatch adds vectorially with the incident waveform creating a standing wave on the transmission line. The ratio of the sum of the peaks of these waves to the difference (the Voltage Standing Wave Ratio; VSWR) can be calculated from the differences in the characteristic impedance of the system and the switch.

$$\text{For } Z_O > Z_L \text{ then VSWR} = \frac{Z_L}{Z_O}$$

and

$$\text{For } Z_O < Z_L \text{ then VSWR} = \frac{Z_O}{Z_L}$$

For systems that do not experience loss, Z_O is the characteristic impedance of the transmission line. Z_L is the load impedance.

As long as the wavelength of the input waveform is large compared to the cable length, these effects are minimal and can largely be ignored. For example, for coaxial airline, signals less than 10 MHz will generally cause no problems with reflections. To understand this, a simple calculation of the wavelength versus frequency can be made from the relationship:

$$\tau \times f = C,$$

where: τ = wavelength
f = frequency
c = speed of light in the media

Frequency	Wavelength
1 MHz	300 m.
10 MHz	30 m.
100 MHz	3 m.

Many times high frequency cw signals are not common in systems, but there are many encounters with pulsed signals and pulsed signal routing. Fast transition time digital pulses can cause double triggering, double counting, and race conditions in the test system if there is a mismatch through the system. Hence, it is important to know what signals are going to cause a problem and how to estimate the size of the problem.

The relative amplitude of the reflected waveform can be calculated from the VSWR specification for the switch from the relationships:

$$BW \times t_r = .35$$

and

$$E_r = E_i \times \frac{V-1}{V+1}$$

Where: BW = Bandwidth (maximum frequency)
t_r = Pulse Risetime
E_r = Amplitude of the reflected wave
E_i = Amplitude of the incident wave
V = The specified VSWR

For example, let's look at the reflections of a 5 volt signal (with 100 MHz components) being routed through a switch with a VSRW of 2.0 at 100 MHz. First, let's calculate the risetime of the signal:

$$t_r = \frac{0.35}{100 \times 10^6}$$

Therefore,

$$t_r = 3.5 \text{ nS}$$

We can calculate the amplitude of the reflected wave from

$$E_r = 5 \times \frac{(2-1)}{(2+1)} = 1.67V$$

which for the sample case is equal to 33% of the amplitude of the incident waveform. In the case of a 5 volt signal, this amplitude could cause even TTL with risetimes of 3.5 ns to indicate a false state.

A Noise Reduction Check List†

When designing your system the following check list will be helpful in reducing noise.

A. Suppressing Noise At Source:

- ☐ Enclose noise sources in a shielded enclosure.
- ☐ Filter all leads leaving a noisy environment.
- ☐ Limit pulse risetimes.
- ☐ Relay coils should be provided with some form of surge damping.
- ☐ Twist noisy leads together.*
- ☐ Shield and Twist Noisy Leads.
- ☐ Ground both ends of shields used to suppress radiated interference (shield does not need to be insulated).*

B. Eliminating Noise Coupling:

- ☐ Twist low level signal leads.*
- ☐ Place low level leads near chassis (especially if the circuit impedance is high).
- ☐ Twist and shield signal leads (coaxial cable may be used at high frequencies).
- ☐ Shielded cables used to protect low level signal leads should be grounded at one end only (coaxial cable may be used at high frequencies with shield grounded at both ends).*
- ☐ Insulate shield on signal leads.
- ☐ When low level signal leads and noisy leads are in the same connector, separate them and place the ground leads between them.*
- ☐ Carry shield on signal leads through connectors on a separate pin.
- ☐ Avoid common ground leads between high and low level equipment.*
- ☐ Keep hardware grounds separate from circuit grounds.*
- ☐ Keep ground leads as short as possible.*
- ☐ Use conductive coatings in place of nonconductive coatings for protection of metallic surfaces.
- ☐ Separate noisy and quiet leads.*
- ☐ Ground circuits at one point only (except at high frequencies).*
- ☐ Avoid questionable or accidental grounds.
- ☐ For sensitive applications, operate source and load balanced to ground.
- ☐ Place sensitive equipment in shielded enclosures.
- ☐ Filter or decouple any leads entering enclosures containing sensitive equipment.
- ☐ Keep the length of sensitive leads as short as possible.*
- ☐ Keep the length of leads extending beyond cable shields as short as possible.*

† Ott, H., "Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976

* Low cost noise reduction techniques

☐ Use low impedance power distribution lines.
☐ Avoid ground loops.*
☐ Consider using the following devices for breaking ground loops:

- Isolation transformers
- Optical couplers
- Differential amplifiers
- Guarded amplifiers
- Balanced circuits

C. Reducing Noise at Receiver:

☐ Use only necessary bandwidth.
☐ Use frequency selective filters when applicable.
☐ Provide proper power supply decoupling.
☐ Bypass electrolytic capacitors with small high frequency capacitors.
☐ Separate signal, noisy, and hardware grounds.*
☐ Use shielded enclosures.
☐ With tubular capacitors, connect outside foil end to ground.*

---

* Low cost noise reduction techniques

Bibliography

1. Ott, H.W., "Noise Reduction Techniques in Electronic Systems", John Wiley and Sons, 1976.

2. Ott, H.W. , "Ground - A Path for Current Flow", EMC Technology and Interference Control news", January - March 1983, p. 44-48.

3. Vance, E.V., "Cable Grounding for the Control of EMI", EMC Technology and Interference Control News", January - March 1983, p. 54-58.

4. Morrison, R., "Grounding in Instrumentation Systems", EMC Technology and Interference Control News, January - March 1983, p. 50-52.

5. Wedlock, B. and Roberge, J., "Electronic Components and Measurements". Prentice- Hall,Inc., 1969.

6. Whitmer, R., "Electromagnetics", Prentice Hall, Inc., 1963.

7. Adam, S., "Microwave Theory and Applications", Prentice Hall Inc., 1969.

8. Scott, R., "Linear Circuits", Addison - Wesley Publishing Co., Inc., 1960.

9. Hewlett-Packard Application Note #94, "Connector Design Employing TDR Techniques", January 1968.

10 . Hewlett-Packard Application Note #191, "Precision Time Interval Measurements Using and Electronic Counter", 1976.

11. Hewlett-Packard Application Note #150, "Spectrum Analysis... Spectrum Analysis Basics", 1976.

12. Hewlett-Packard Application Note #67, "Cable Testing with Time Domain Reflectometry", 1968.

13. Hewlett-Packard Application Note #290, 'Practical Temperature Measurements", 1980.

14. Hewlett-Packard Application Note #322, "Microwave Switching From SPDT To Full Access Matrix".

Self Test

Introduction

Your HP 3235 has five internal test routines. These routines allow you to verify proper operation of the HP 3235, test specific plug-in modules, and characterize the switching relays on the multiplexer, matrix, and general purpose relay modules. This chapter describes the test routines and shows how to perform them. For detailed information on each test routine, refer to the HP 3235 Service Manual.

The first test is executed automatically when power is applied to the HP 3235. This test is referred to as the *Power-On Self Test* and it checks the system modules and a portion of the HP 34520 Multimeter Module (if installed).

The second test is the *Confidence Self Test* and is performed when the **TEST** (or **TST**) command is executed. This command tests only the user plug-in modules and the HP 34550 Control Panel, and is designed to give every possible assurance that the modules are working properly without affecting the hardware or software state. No test equipment is required for **TEST** and it can be executed with all signal inputs and outputs in place.

The third test is the *System Self Test* and is performed when the **FTEST SYS** (or **FTST SYS**) command is executed. This command tests the HP 3235's trigger bus system.

The fourth test is the *Fixtured Self Test* and is performed when the **FTEST** (or **FTST**) command is executed. This command provides 90% confidence that the HP 3235 and its plug-in modules are operating properly within the specified performance limits. Self test fixtures must be installed on each plug-in module to be tested (the exception to this is the HP 34520 Multimeter which can be tested without a fixture). An HP 34520 Multimeter Module and an HP 34520D self test fixture are required for testing relay modules.

The fifth test is the *Data Self Test* and is performed when the **DTEST** or (**DTST**) command is executed. This test is a subset of the fixtured self test routine for the relay modules. The **DTEST** command returns the measured path resistance (relay contact resistance + current limiting resistor + circuit board trace resistance).

If any of the tests described in this chapter fail, the first four errors are logged in the HP 3235's error register. The following example shows how to read the contents of the error register. For additional information on the error register, refer to Chapter 4 in this manual.

```
10  DIM Message$[256]               ! DIMENSION HP 3235 ARRAY
20  REPEAT                          ! BEGIN LOOP
30    OUTPUT 709; "ERRSTR?"         ! READ ERROR REGISTER
40    ENTER 709; Code,Message$      ! ENTER VALUES
50    PRINT Code,Message$           ! PRINT VALUES
60  UNTIL Code=0                    ! END LOOP
70  END
```

Power-On Self Test

The Power-On Self Test executes automatically when power is applied to the HP 3235 mainframe and all connected extenders. All system side modules in the mainframe and all connected extenders are tested. No user side plug-in modules are tested. The HP 3235 polls each slot in the system and internally stores the module IDs. The number of modules found at power-on is displayed on the control panel. If no self test errors are found, *READY* is displayed on the control panel.

If an error occurs during the self test, the HP 3235 error register logs the first four errors and the last error remains displayed on the control panel. Self test error messages are in the form: *STF n <message>* where *n* is the number of the self test that failed and *<message>* describes the failure.

At power-on, tests are performed on the following functional areas:

- ROM and RAM handshake and checksum
- Timers
- I/O timeout circuitry
- System Data Bus, Address Bus, and Decoding Circuitry
- Crossguard (isolation) data
- HP-IB
- Control panel transmit and receive
- Extender frame control logic
- Extender frame system and crossguard data bus
- Internal hardware and backplane interrupts

Confidence Self Test

The **TEST** (or **TST**) command performs the Confidence Self Test. This test does not change the hardware or software state of the HP 3235 mainframe or extenders. If you specify slot numbers with the **TEST** command, the HP 3235 tests only those slots (e.g., **TEST 200,300** tests the modules in mainframe slots 200 and 300). If you do not specify a slot number, the **TEST** command tests all mainframe and extender slots plus the HP 34550 Control Panel.

To test all slots in the system plus the control panel, send:

```
OUTPUT 709; "TEST"
```

To test the modules in slots 100 and 500 of the mainframe and slot 300 of extender 2, send:

```
OUTPUT 709; "TEST 100,500,2300"
```

The message "PASS" or "FAIL" is returned to the control panel display at the completion of **TEST**. Any failures that occur are noted on the control panel display and the first four errors are stored in the error register. Use the **ERR?** or **ERRSTR?** commands to read the error register (see Chapter 4 in this manual).

NOTE

If is possible for a failure on one user module to be reported as a failure on all plug-in modules. If failures are reported from all installed plug-in modules, remove the modules from the cardcage and repeat **TEST**. *If "PASS" is returned with no modules installed, replace one module at a time, repeating the* **TEST** ***slot*** *command for each module to isolate the defective module. If "FAIL" is returned with no modules installed, repeat the Power-On Self Test and System Self Test (***FTEST SYS*** *command) to determine the system failure.*

The **TEST** command performs the tests described below. For additional information, refer to the HP 3235 Service Manual.

1. If **TEST** is executed without specifying any slot numbers, the control panel is tested first. During the test, all control panel display segments and all annunciators (except SHIFT on the far left) are turned on. The control panel test is not performed if any slot number are specified with **TEST**.

2. If **TEST** is executed with a slot number or a list of slot number, various tests are performed on the modules in the specified slots. The specific tests performed vary from module to module, but in general, the following occurs:

- Each internal register on the plug-in module is addressed. If a register does not respond, the test fails.

- The HP 3235 reads the identification (ID) register on each module. Each slot must have the same module type as was installed at power-on or the test fails.

- The "busy" time for each switching module is measured and compared to limits stored in the HP 3235 mainframe memory. If the time is not within the limits, the tests fails.

3. If the **TEST** command is sent to the HP 34520 Multimeter Module, the HP 3235 opens the module's backplane relays and then reads and stores the present state of the multimeter (i.e., measurement function, range, etc.). At the completion of the test, the HP 3235 restores the multimeter to its previous state. The specific tests performed on the multimeter follow:

   - Multimeter power-on self test
   - Integrator convergence test
   - Front end zero test
   - Current source test
   - AC/DC offset autocal circuitry test
   - AC flatness autocal circuitry test
   - Nonvolatile RAM/ROM checksum test
   - Cal protection circuitry test

System Self Test

The **FTEST SYS** (or **FTST SYS**) command performs the System Self Test which tests the HP 3235's trigger bus system. Executing **FTEST SYS 0** performs the default tests on the trigger buses. Executing **FTEST SYS 1** performs additional tests using the HP 3235's External Trigger In and External Trigger Out BNCs connected on all frames. The **FTEST SYS** command does not change the hardware or software state of the HP 3235 mainframe or extenders.

The **FTEST SYS** command returns "PASS" if all tests pass or "FAIL" if one or more tests fail. Any failures that occur are noted on the control panel display and the first four errors are stored in the error register. Use the **ERR?** or **ERRSTR?** command to read the error register (refer to Chapter 4 for more information on the error register).

NOTE

The trigger bus lines are active during the **FTEST SYS** *command. Therefore, any devices connected to the trigger bus system may be activated when the test is performed.*

To perform the default trigger bus test, send:

```
OUTPUT 709; "FTEST SYS"
```

or

```
OUTPUT 709; "FTEST SYS,0"
```

The **FTEST SYS** command performs the following default trigger bus tests:

- **Mainframe Trigger Bus Tests.** These tests verify the various multiplexer paths contained in the mainframe trigger bus switching network. The test includes 12 DC voltage tests and four signal tests. During the DC voltage tests, various trigger paths are selected and verified using high and low voltages. For the signal tests, the output of timer channel 1 is looped back to the clock input of timer channel 0 using different signal paths. Timer 1 outputs pulses for each test and the count in Timer 0 must match the number of pulses sent by Timer 1.

- **Mainframe Expander Module Trigger Bus Tests.** This group of 18 tests verify the Extender Trigger bus system in the HP 3235 mainframe Expander Module.

- **Trigger Interrupt Tests.** These four tests verify the routing of backplane trigger bus signals and their ability to interrupt the mainframe.

- **Expander Frame Trigger Bus Tests.** These tests verify all of the trigger bus paths in each HP 3235E Extender frame. Some of these tests are duplications of the mainframe tests. They test all extender frame crossguard and expander modules.

In addition to the default tests just described, additional tests on the trigger bus system may be performed in each of the four groups by connecting the External Trigger In and Out BNC connectors using a short cable (this must be done on each frame). To perform the default tests plus the additional tests, send:

```
OUTPUT 709; "FTEST SYS,1"
```

Fixtured Self Test

The **FTEST** (or **FTST**) command performs a pass/fail Fixtured Self Test on the HP 3235 plug-in modules and tests the system module trigger bus circuitry. Diagnostic Self Test fixtures must be installed on each plug-in module to be tested. If relay modules are being tested, an HP 34520 Multimeter Module and HP 34520D self test fixture must be used. Refer to the HP 3235 Service Manual for a detailed explanation of the **FTEST** command.

NOTE

The **FTEST** *command resets the mainframe, all extenders connected to the mainframe, and any installed plug-in modules.*

If you specify slot numbers with the **FTEST** command, the HP 3235 tests only those slots (e.g., **FTEST 200,300** tests the modules in mainframe slots 200 and 300). If you do not specify a slot number, the HP 3235 tests all modules in the system (up to a maximum of 10 slots). However, each module must have a test fixture attached and relay modules must be connected, through the test fixture, to an HP 34520 Multimeter Module. (Refer to "Test Fixtures Required" later in this chapter for a list of the self test fixtures.) The control panel is also tested if you do not specify a slot number with the **FTEST** command.

If more than one slot is to be tested using a single **FTEST** command, the slot ID jumper on each test fixture must be set to correspond to the slot of the module being tested. If only one test fixture at a time is connected to the multimeter, the slot ID jumper may be left in the ALL position. If two or more test fixtures with the slot ID jumper in the ALL position are connected to the multimeter, faults on one module could be falsely reported as faults on other modules.

Relay Module Test

For the relay modules, **FTEST** measures the resistance of signal paths through all the relays using the HP 34520 Multimeter Module. The measurements are made through the HP 34520D self test fixture and chaining cable which attaches to the relay module test fixtures. The test routine then compares the measured resistance values to limits defined for each module type. If the measured path resistances are within the specified limits, the test passes and "PASS" is displayed on the control panel. If any path resistance is outside the limits, the test fails and "FAIL" is displayed on the control panel.

To test all plug-in modules in the system plus the control panel, send:

```
OUTPUT 709; "FTEST"
```

To test the plug-in modules in slots 100 and 500 of the mainframe and slot 300 of extender 2, send:

```
OUTPUT 709; "FTEST 100,500,2300"
```

The backplane relays on the HP 34501, 34502, and 34507 relay multiplexer modules can be tested using the **FTEST** *es01* command. For example, to test the backplane relays on the relay multiplexer module in mainframe slot 200, send:

```
OUTPUT 709; "FTEST 201"
```

Multimeter Module Test

The **FTEST** routine for the HP 34520 Multimeter first opens the module's backplane relays and then performs the Power-On and Confidence (**TEST**) tests. The internal trigger buses are also tested. The **FTEST** command does not test the backplane relays (however, the backplane relays are tested as part of the relay modules backplane relay test **FTEST** *es01*).

Source Module Test

The **FTEST** routine for the HP 34521 AC/DC Source Module tests the output functions (DC volts, DC current, AC volts), the module's functional circuit blocks, and isolates faults to the component-level. The tests are performed using the HP 34520 Multimeter's self test fixture (HP 34520D) and chaining cable which attaches to the Source Module's self test fixture (HP 34521D).

Quick, accurate electronic auto-calibration of the HP 34521 can be performed using the **FTEST** *es42* command. A more detailed explanation of the **FTEST** command is provided in the HP 3235 Service Manual.

Digital I/O Module Test

The **FTEST** routine for the HP 34522 Digital I/O Module tests the I/O lines, the handshake lines and handshake modes, the high current output lines, and the interrupt lines. The HP 34520 Multimeter is not used to perform the fixtured self test on the HP 34522 (the HP 34522D self test fixture does not connect to the multimeter).

D/A Converter Module Test

The **FTEST** routine for the HP 34524 D/A Converter Module tests the output functions (DC volts, DC current) and the module's functional circuit blocks. The tests are performed using the HP 34520 Multimeter's self test fixture (HP 34520D) and chaining cable which attaches to the D/A Converter Module's self test fixture (HP 34524D).

Quick, accurate electronic auto-calibration of the HP 34524 can be performed using the **FTEST** *es42* command. A more detailed explanation of the **FTEST** command is provided in the HP 3235 Service Manual.

Test Fixtures Required

Table 13-1 lists the self test fixtures required for **FTEST**.

Table 13-1. Diagnostic Test Fixtures

Description of Self-Test Fixture	Use This Number When Ordering
Fixture for HP 34501, 34502, and 34507 32-Channel Relay Modules	HP 34501D
Fixture for HP 34503 General Purpose Relay Module	HP 34503D
Fixture for HP 34504 Switched-Shield Coaxial Multiplexer Module	HP 34504D
Fixture for HP 34505 and 34508 RF Multiplexer Modules	HP 34505D
Fixture for HP 34506 Switched-Shield Coaxial Matrix Module	HP 34506D
Fixture for HP 34509 Relay Driver Module	HP 34509D
Fixture for HP 34510 Power Actuator Module	HP 34510D
Fixture for HP 34511 64-Channel Relay Module	HP 34511D
Fixture for HP 34520 DMM Module. Includes HP 34562-61601 and HP 34562-61602 Test Cables (see below)	HP 34520D
Fixture for HP 34521 Source Module	HP 34521D
Fixture for HP 34522 Digital I/O Module	HP 34522D
Fixture for HP 34524 D/A Converter Module	HP 34524D
Test Fixture Chaining Cable	HP 34562-61601
Test Module Extender Cable	HP 34562-61602

Running the Fixtured Test

The following paragraphs describe how to perform a fixtured test using the **FTEST** command.

CAUTION

TO PREVENT CONNECTOR BREAKAGE, do not attempt to remove a terminal block or self test fixture from a plug-in module without using the extractors or quick interconnect mechanism (when terminal blocks or self test fixtures are installed in a quick interconnect fixture). Install plug-in modules into the card cage before installing terminal blocks or self test fixtures. Secure the module in the card cage with mounting screws at the top and bottom of the plug-in module connector.

Modules secured in the card cage provide connector support when terminal blocks or self test fixtures are installed and removed. When removing a terminal block or self test fixture not in a quick interconnect fixture, ALWAYS use extractors to partially disengage the terminal block or self test fixture from the module. Failure to do this can excessively stress the connector when one side of the terminal block or self test fixture is disengaged and the other side is still fully engaged. Attempting to pull the terminal block off creates a lever arm which pivots on the engaged connector end, thus breaking the connector.

---

1. Remove the terminal blocks from each module to be tested.

2. Remove the cover from each test fixture. If the test fixture is for a relay module, set the slot address jumper to correspond to the slot number where that module is installed. The slot address jumpers are labeled "ALL" and 0 through 9. The jumper may be left in the "ALL" position if you are testing only one module. The numbers 0 through 9 correspond to slots 000 through 900, respectively. Replace the cover on the test fixture. The test fixtures for the HP 34520 Multimeter and HP 34522 Digital I/O modules do not have slot address jumpers. The slot address jumpers are shown in Figure 13-1.

NOTE

If the slot address jumper is correctly set, the red LED on each test fixture turns on when that module is being tested.

---

3. Install the test fixtures on the plug-in modules. The test fixtures connect to the plug-in modules in the same manner as the terminal blocks.

4. If the modules to be tested and the multimeter used in the test are installed in the same frame, connect the HP 34562-61601 Test Fixture Chaining Cable (one is included with the HP 34520D self test fixture) to each test fixture and the multimeter's test fixture. Figure 13-2 shows the connections.

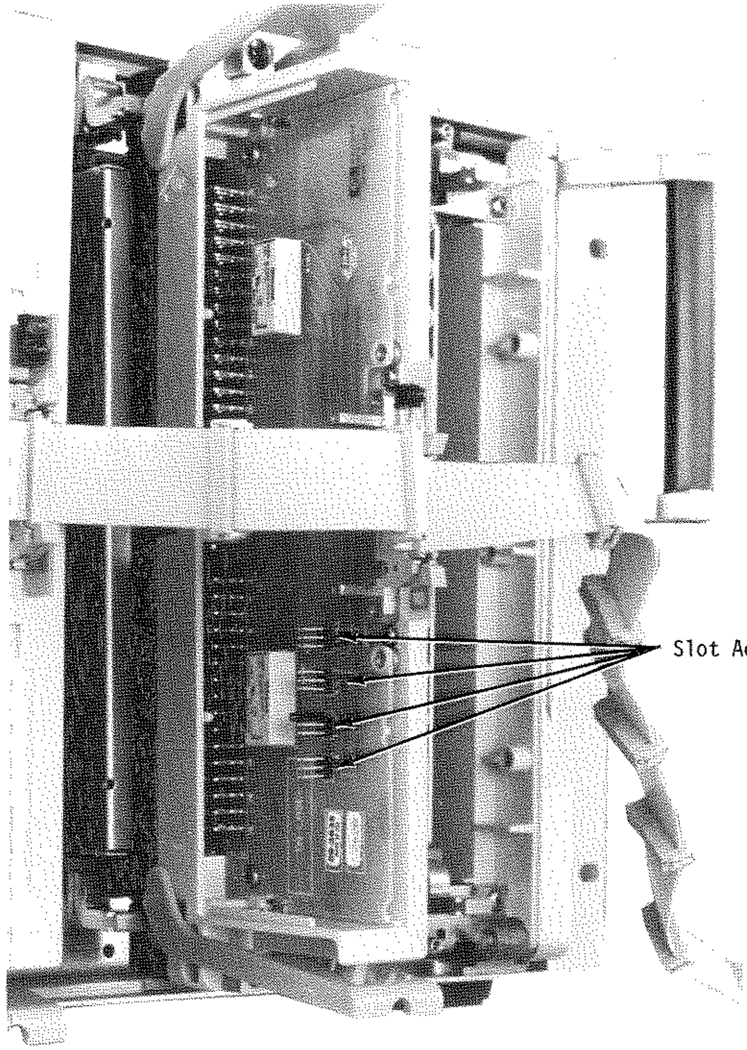


Figure 13-1. Slot Address Jumpers on Installed Module

If the modules to be tested are installed in a different frame than the multimeter, connect the modules using the HP 34562-61601 Test Fixture Chaining Cable. Then connect the Test Fixture Chaining Cable to the multimeter using the HP 34562-61602 Test Module Extender Cable.

NOTE

You can install the test fixtures and chaining cable in a Quick Interconnect Fixture. This allows you to easily and consistently fixture each module for the test.

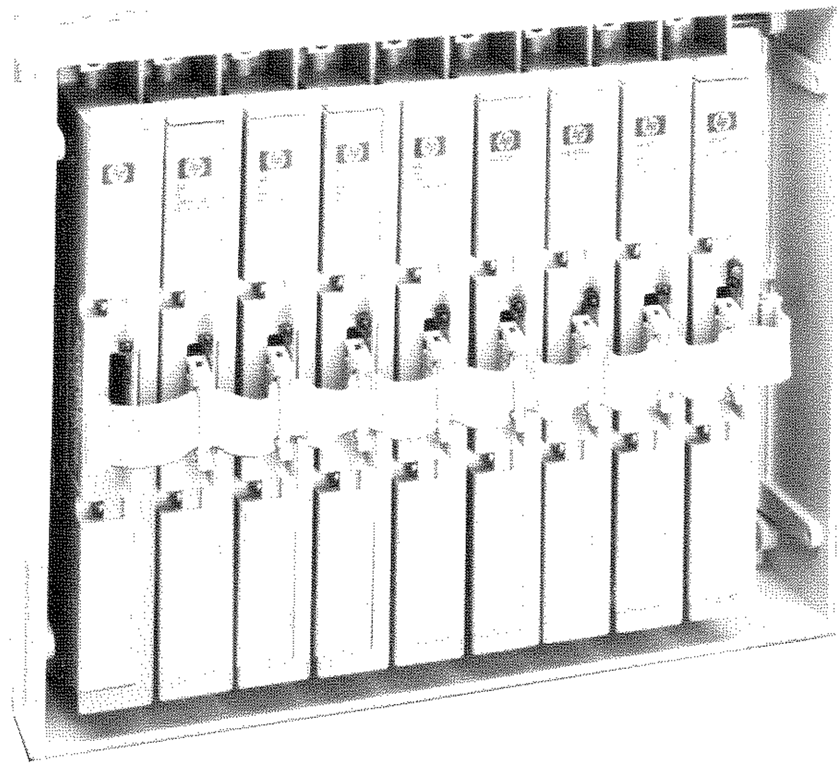


Figure 13-2. Connecting the Chaining Cable

5. Specify an HP 34520 Multimeter as the testing multimeter by sending the **USE** command. For example, if the multimeter is installed in mainframe slot 800, execute:

```
OUTPUT 709; "USE 800"
```

NOTE

If the module selected with the **USE** *command is not an HP 34520 Multimeter Module but one is installed in the mainframe or an extender, the* **FTEST** *command resets the HP 3235 and selects the HP 34520 in the lowest numbered slot.*

---

6. Execute the **FTEST** command with the appropriate slot list. The **FTEST** command returns "PASS" if all tests pass or "FAIL" if any test fails. Any failures that occur are noted on the control panel display and the first four errors are stored in the error register. Use the **ERR?** or **ERRSTR?** commands to read the error register (see Chapter 4 in this manual).

The following program shows how to run **FTEST** on a specific group of modules. If all tests pass, the message "ALL TESTS PASSED" is displayed on the system controller. If any tests fail, the message "TEST FAILED" is displayed and the errors are printed.

```
10   OUTPUT 709; "RST"              ! RESET THE HP 3235
20   OUTPUT 709; "USE 800"          ! SPECIFY MULTIMETER
30   PRINT "INSTALL TEST FIXTURES AND PRESS CONTROLLER 'CONTINUE' KEY"
40   PAUSE
50   OUTPUT 709; "FTEST 100,300,400,500,700"  ! EXECUTE TEST
60   ENTER 709; A$                  ! ENTER RESULT
70   IF A$ = "FAIL" THEN            ! TEST FAILED
80     PRINT "TEST FAILED"
90     DIM Message$[250]
100    REPEAT                       ! PRINT ERROR MESSAGES
110      OUTPUT 709; "ERRSTR?"
120      ENTER 709; Code,Message$
130      PRINT Code,Message$
140    UNTIL Code = 0
150  ELSE
160    PRINT "ALL TESTS PASSED"     ! TESTS PASSED
170  END IF
180  END
```

Fixtured Self Test With Data Report

The **DTEST** (or **DTST**) command is valid only for the relay plug-in modules and performs a fixtured self test with data report. The **DTEST** command executes the same tests as **FTEST** for the relay modules but returns the actual measured path resistance (relay contact resistance + current limiting resistor + circuit board trace resistance). The values returned for the path resistances are in ohms. Refer to the HP 3235 Service Manual for a description of the paths measured.

NOTE

The **DTEST** *command resets the mainframe, all extenders connected to the mainframe, and any installed plug-in modules.*

If you specify slot numbers with the **DTEST** command, the HP 3235 tests only those slots (e.g., **DTEST 200,300** tests the modules in mainframe slots 200 and 300). If you do not specify a slot number, the HP 3235 tests all modules in the system (up to a maximum of 10 slots). If you execute **DTEST** without any slots specified, the HP 3235 tests the entire HP 3235 system connected by the chaining cable.

The following example performs a fixtured test with data report on one module and stores the measurement results in an HP 3235 array.

```
10  DIM Test1(500)
20  OUTPUT 709; "DIM RESULTS(500)"
30  OUTPUT 709; "USE 800"
40  OUTPUT 709; "MEM RESULTS"
50  OUTPUT 709; "DTEST 200"
60  OUTPUT 709; "MEM OFF;VREAD RESULTS"
70  ENTER 709; Test1(*)
80  PRINT Test1(*)
90  END
```

CIIL Commands

Introduction

Your HP 3235 provides the standard five Switching Module CIIL (Control Intermediate Interface Language) op-codes. These op-codes are similar to other HP 3235 commands but provide compatibility with the proposed MATE (Modular Automated Test Equipment) standard 2806763 Rev B. The CIIL op-codes are for the switching modules only; other module types (e.g. Sensor or Stimulus) are not included.

NOTE

The CIIL commands are valid at all times. You do not need to invoke a "CIIL Mode".

CIIL Switching Op-Codes

The five Switching Module CIIL op-codes are **CON**, **DIS**, **CNF**, **IST**, and **STA**. Each of these commands are described below. For additional information refer to the HP 3235 Language Reference Manual.

- **CON** - connect op-code

  Formal Syntax:

  CON *path-data*

  This op-code instructs a switching module to close the specified relays or channels. It is the CIIL version of the **CLOSE** command (see Chapter 5). When used with the switch modules, **CON** closes the specified relays or channels. When used with the HP 34522 Digital I/O Module, **CON** 'clears' the specified bits. The *path-data* specifies the channels or digital I/O bits to close. Each channel or bit entry in the *path-data* has the form *esnn* where *e* is the mainframe or extender number (0=mainframe, 1-7 = extender), *s* is the slot number (0-9), and *nn* is the relay, channel, or bit number. Multiple entries must be separated by either a comma or a space. For example, to close channels 114, 170, and 191 (connects channel 114 to analog bus 1), send the string:

  ```
  CON 114,170,191
  ```

- **DIS** - disconnect op-code

  Formal Syntax:

  DIS *path-data*

This op-code instructs a switching module to open the specified relays. It is the CIIL version of the **OPEN** command (see Chapter 5). When used with the switching modules, **DIS** opens the specified relays or channels. When used with the HP 34522 Digital I/O Module, **DIS** 'sets' the specified bits. The *path-data* specifies the channels or digital I/O bits to open. Each channel or bit entry in the *path-data* has the form *esnn* where *e* is the mainframe or extender number (0=mainframe, 1-7= extender), *s* is the slot number (0-9), and *nn* is the relay, channel, or bit number. Multiple entries must be separated by a comma or a space. For example, to open channels 114, 170, and 191, execute the string:

```
DIS 114,170,191
```

- **CNF** - internal confidence test op-code

  Formal Syntax:

  CNF

  No confidence test is performed. The results are reported in response to the CIIL STATUS op-code (**STA**). No operands are associated with this op-code.

- **IST** - internal self-test op-code

  Formal Syntax:

  IST

  No internal self-test is performed. The results are reported in response to the CIIL STATUS op-code (**STA**). No operands are associated with this op-code.

- **STA** - status op-code

  Formal Syntax:

  STA

  This op-code causes the HP 3235 to return the most recent error code and error string. It is the CIIL version of the **ERRSTR?** command (see Chapter 4). It returns the error code and string of the most recent error, deletes that error from the error register, and clears the error bit in the status register when all errors have been read. If no error is present, **STA** returns a blank space. The normal (no error) response message is:

```
<sp><cr><lf>
```

  Abnormal (error) messages are of the form:

```
F07GPS00<sp>(MOD):<error number><error string><cr><lf>
```

  Typical error messages are:

```
F07GPS00 (MOD):    61OUT OF RANGE -- 8.00000000E+02
F07GPS00 (MOD):     1INCOMPLETE COMMAND -- A; EXPECTED COMMAND HEADER
F07GPS00 (MOD):    62WRONG CARD TYPE -- USE MULTIMETER CARD
```

Additional Op-Codes

Two additional op-codes provide device clear capabilities. These two commands are:

- **CIIL** - invokes CIIL definition for device clear action

  Formal Syntax:

  CIIL

  This op-code (not a standard CIIL op-code) puts the HP 3235 in its complete CIIL mode. Specifically, it causes device clear commands to act according to CIIL rules. The CIIL mode Device Clear causes the same action as the native mode **RESET** (or **RST**) command. That is, the HP 3235 is reset to its power-on state. All plug-in modules are reset, the HP-IB input and output buffers are cleared, etc. Refer to the **RESET** command in Chapter 4 of this manual for complete information. Either **CIIL** or **GAL** (op-code described next) is in effect at any time. The state of **CIIL** versus **GAL** is stored in continuous (nonvolatile) memory and thus is not lost when power is removed from the HP 3235.

- **GAL** - use native language definition for device clear op-code

  Formal Syntax:

  GAL

  This op-code causes a device clear to function equivalent to the native mode **CLR** command (see Chapter 4). A device clear does not change the hardware state of the HP 3235. Either **CIIL** or **GAL** is in effect at any time. The state of **GAL** versus **CIIL** is stored in continuous (nonvolatile) memory.

Specifications

Introduction

The specifications for the HP 3235 are the performance characteristics of the instrument which are certified. These specifications are listed in this appendix and are the performance standards or limits against which the instrument is tested. Included in the table are some supplemental characteristics of the HP 3235. These should be considered as additional and general information for you, the user. Because of the many operating capabilities of the instrument, exercise care when testing the instrument specifications.

Specifications for the plug-in modules are provided in the HP 3235 Plug-In Module Manual.

Any changes in specifications due to manufacturing changes, design, or traceability to the National Bureau of Standards will be covered in a manual change supplement.

WARNING

If any of the following symptoms exist or are suspected, remove the HP 3235 from service. Do not use the instrument until safe operation is verified by service trained personnel.

1. *Visible damage.*
2. *Severe transport stress.*
3. *Prolonged storage under adverse conditions.*
4. *Failure to perform intended measurements or functions.*

If necessary, return the instrument to a Hewlett-Packard service office for service and repair to ensure that safety features are maintained.

Environmental

OPERATING TEMPERATURE: 0-55°C (32-130°F)
STORAGE TEMPERATURE: −40°C to 75°C (−40°C to 165°F)
HUMIDITY RANGE: 95% R.H., 0°C TO 40°C

Power

LINE VOLTAGE: 90-132V (115V) OR 192-264 (230V) switch selectable, 47-66 Hz. Fused at 5A (115V) or 2.5A (230V).

Size

HP 3235 CARDCAGE: 310mm H (without feet)x 426mm W x 594mm D (12.2"x16.8"x23.4")
HEIGHT WITH FEET: 325mm (12.8")
DEPTH WITH TERMINAL BLOCKS: 693mm (27.3")
Allow 90mm (3.5") additional depth for wiring to terminal blocks.
Allow 102mm (4") additional depth for analog and digital expansion cables.

Weight

	Net		Shipping	
	kg	lbs	kg	lbs
HP 3235 CARDCAGE without option 590	17	37.4	27.6	52
with option 590	21	46	28	61
HP 34520 DMM MODULE	5.5	12	6.6	14.5
EACH MODULE (maximum) excludes HP 34520	3	6.5	3.6	8

SYSTEM MEASUREMENT ACCURACY

These specifications give total system accuracy for a signal switched through the analog backplane buses from one of the 32 channel multiplexer modules to a HP 34520 DMM Module in the same frame. Specifications reflect inaccuracies contributed by the multiplexer modules, the backplane buses, and the multimeter module.

Only specifications for measurement accuracy are shown here. Refer to HP 34520 Multimeter specifications for input characteristics. No system accuracies are specified for DC and AC current measurements because current cannot be switched through the backplane analog buses.

Measurement accuracy for signals switched through the HP 34501 32 Channel Armature Relay Multiplexer

DC VOLTAGE

TABLE 10-1. DC VOLTS Measurement Accuracy: ±(% of reading + volts) Auto-zero on. Tcal is the temperature of calibration from 18°C to 28°C.

24 HOUR: Tcal ±1°C. After one hour warm-up. Accuracy relative to calibration standard.

Range	% of Reading	+	Volts NPLC: 100	10	1	.1	.005	.0005
30mV	.0017%	+	4.9μV	5.01μV	5.7μV	7.2μV	19μV	70μV
300mV	.001%	+	5.1μV	5.1μV	5.7μV	9μV	50μV	400μV
3V	.00085%	+	7μV	8μV	8μV	50μV	400μV	4mV
30V	.0012%	+	110μV	120μV	220μV	700μV	4mV	40mV
250V	.0031%	+	600μV	700μV	700μV	5mV	40mV	400mV

90 DAY: Tcal ±5°C. After one hour warm-up.

Range	% of Reading	+	Volts NPLC: 100	10	1	.1	.005	.0005
30mV	.0053%	+	6.12μV	6.26μV	7.15μV	8.9μV	21μV	70μV
300mV	.0038%	+	6.3μV	6.4μV	7.2μV	11μV	50μV	400μV
3V	.003%	+	8μV	9μV	9μV	50μV	40μV	4mV
30V	.0048%	+	220μV	230μV	330μV	800μV	4mV	40mV
250V	.0063%	+	700μV	800μV	800μV	5mV	40mV	400mV

1 YEAR: Tcal ± 5°C. After one hour warm-up.
Add .001% of reading to 90 day specifications.

Temperature Coefficient: For operating temperatures outside specified range, refer to the temperature coefficients for the HP 34520 DMM module.

Auto-Zero Off: Add additional accuracy error as specified for the HP 34520 DMM module.

Range-to-Range Response Error: For default settling times, add .0005% of input voltage step to the first reading following a range change.

RESISTANCE

TABLE 10-2. RESISTANCE Measurement Accuracy: ±(% of reading + ohms) Four-wire ohms, auto-zero on, offset compensation on or off. Tcal is the temperature of calibration from 18°C to 28°C.

24 HOUR: Tcal ±1°C. After one hour warm-up, relative to calibration standard.

Range	% of Reading[1]	+	Ohms NPLC: 100	10	1	.1	.005	.0005
30Ω	.0035%	+	5.01mΩ	5.12mΩ	5.84mΩ	7.4mΩ	19mΩ	70mΩ
300Ω	.002%	+	5.2mΩ	5.2mΩ	5.9mΩ	9mΩ	50mΩ	400mΩ
3kΩ	.0018%	+	7mΩ	8mΩ	8mΩ	50mΩ	400mΩ	4Ω
30kΩ	.0045%	+	70mΩ	80mΩ	80mΩ	500mΩ	4mΩ	40Ω
300kΩ	.032%	+	800mΩ	900mΩ	1Ω	5Ω	40Ω	400Ω
3MΩ	.304%	+	14Ω	16Ω	18Ω	80Ω	500Ω	5kΩ
30MΩ	2.95%	+	820Ω	850Ω	950Ω	1.5kΩ	6kΩ	40kΩ
300MΩ[2]	25%	+	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ
3GΩ[2]	85%	+	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ

[1] Specification applies for ≤(40°C, 95% RH). For typical performance, ≤(25°C, 40% RH), add the following % of reading to the ohms offset listed above:

Range	30Ω	300Ω	3kΩ,30kΩ	300kΩ	3MΩ	30MΩ	300MΩ	3GΩ
% of rdg. (Typ.)*	.0035%	.002%	.0016%	.0021%	.007%	.053%	1.27%	12.6%

[2] Specification applies for two-wire ohms only, NPLC ≥1, inputs ≥10% of full scale, and within 24 hours of ACAL.

90 DAY: Tcal ±1°C. After one hour warm-up.

Range	% of Reading[1]	+	Ohms NPLC: 100	10	1	.1	.005	.0005
30Ω	.0078%	+	6.12mΩ	6.26mΩ	7.15mΩ	8.9mΩ	20mΩ	70mΩ
300Ω	.0058%	+	6.3mΩ	6.4mΩ	7.2mΩ	11mΩ	50mΩ	400mΩ
3kΩ	.0051%	+	9mΩ	10mΩ	10mΩ	50mΩ	400mΩ	4Ω
30kΩ	.0078%	+	90mΩ	100mΩ	100mΩ	500mΩ	4Ω	40Ω
300kΩ	.0036%	+	1Ω	1.1Ω	1.2Ω	5Ω	40Ω	400Ω
3MΩ	.308%	+	15Ω	17Ω	19Ω	80Ω	500Ω	5kΩ
30MΩ	2.95%	+	830Ω	860Ω	960Ω	1.5kΩ	6kΩ	50kΩ
300MΩ[2,3]	26%	+	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ
3GΩ[2,3]	100%	+	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ

[1] Specification applies for ≤(40°C, 95% RH). For typical performance, ≤(25°C, 40% RH), add the following % of reading to the ohms offset listed above:

Range	30Ω	300Ω	3kΩ,30kΩ	300kΩ	3MΩ	30MΩ	300MΩ	3GΩ
% of rdg. (Typ.)*	.0078%	.0058%	.0048%	.0063%	.011%	.062%	2.80%	28%

[2] For 90 day and 1 year specification under stable condition (±1°C), following ACAL, use 24 hour specifications instead.

[3] Specification applies for two-wire ohms only, NPLC ≥1, inputs ≥10% of full scale.

1 YEAR: Tcal ±5°C. After one hour warm-up.
For all ranges except 30MΩ, add 0.0015% of reading to the 90 day specifications.
For 30MΩ range, add 0.0125% of reading to the 90 day specifications.

Temperature Coefficient: For operating temperatures outside specified range, use the temperature coefficients for the HP 34520 DMM module.

Two-wire Ohms Accuracy: Add ±5.7Ω to all four-wire specifications.

Auto-Zero Off: Add additional accuracy error as specified for the HP 34520 DMM module.

Response Time: First reading meets accuracy specification with preprogrammed settling times and no external circuit capacitance. An additional delay of 0.1 sec is necessary after a range or function change.

TRUE RMS AC VOLTAGE

AC voltage accuracy specified for **sine wave inputs >10% of range.** DC component <10% of AC component. AC slow filter on (ACBAND 20). After one-hour warm-up. Autozero on.

AC volts measurement accuracy for signals switched through the HP 34501 armature multiplexer module is the same as for a signal input directly to the HP 34520 multimeter.

DC VOLTAGE

Maximum measured input for signals switched through the reed relay module is limited to **125V DC or AC Peak.**

TABLE 10-3. DC VOLTS Measurement Accuracy: ±(% of reading + volts) Auto-zero on. Tcal is the temperature of calibration from 18°C to 28°C.

24 HOUR: Tcal ±1°C. After one hour warm-up. Accuracy relative to calibration standard.

Range	% of Reading +		Volts					
		NPLC	100	10	1	.1	.005	.0005
30mV	.0017%	+	4.9μV	5.01μV	5.7μV	7.2μV	19μV	70μV
300mV	.001%	+	5.1μV	5.1μV	5.7μV	9μV	50μV	400μV
3V	.00085%	+	7μV	8μV	8μV	50μV	400μV	4mV
30V	.0034%	+	110μV	120μV	220μV	700μV	4mV	40mV
250V	.0052%	+	600μV	700μV	700μV	5mV	40mV	400mV

90 DAY: Tcal ±5°C. After one hour warm-up.

Range	% of Reading +		Volts					
		NPLC	100	10	1	.1	.005	.0005
30mV	.0053%	+	6.12μV	6.26μV	7.15μV	8.9μV	21μV	70μV
300mV	.0038%	+	6.3μV	6.4μV	7.2μV	11μV	50μV	400μV
3V	.0030%	+	8μV	9μV	9μV	50μV	400μV	4mV
30V	.0070%	+	220μV	230μV	330μV	800μV	4mV	40mV
250V	.0085%	+	700μV	800μV	800μV	5mV	40mV	400mV

1 YEAR: Tcal ± 5°C. After one hour warm-up.
Add .001% of reading to 90 day specifications.

Temperature Coefficient: For operating temperatures outside specified range, refer to the temperature coefficients for the HP 34520 DMM module.

Auto-Zero Off: Add additional accuracy error as specified for the HP 34520 DMM module.

Range-to-Range Response Error: For default settling times, add .0005% of input voltage step to the first reading following a range change.

RESISTANCE

TABLE 10.4. RESISTANCE Measurement Accuracy: ±(% of reading + ohms) Four-wire ohms, auto-zero on, offset compensation on or off. Tcal is the temperature of calibration from 18°C to 28°C.

24 HOUR: Tcal ±1°C. After one hour warm-up, relative to calibration standard.

Range	% of Reading[1] +		Ohms					
		NPLC	100	10	1	.1	.005	.0005
30Ω			NOT SPECIFIED					
300Ω			NOT SPECIFIED					
3kΩ	.0018%	+	7mΩ	8mΩ	8mΩ	50mΩ	400mΩ	4Ω
30kΩ	.0045%	+	70mΩ	80mΩ	80mΩ	500mΩ	4Ω	40Ω
300kΩ	.032%	+	800mΩ	900mΩ	1Ω	5Ω	40Ω	400Ω
3MΩ	.304%	+	14Ω	16Ω	18Ω	80Ω	500Ω	5kΩ
30MΩ	2.94%	+	820Ω	850Ω	950Ω	1.5kΩ	6kΩ	50kΩ
300MΩ[2]	25%	+	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ
3GΩ[2]	85%	+	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ

[1] *Specification applies for ≤(40°C, 95% RH). For typical performance, ≤(25°C, 40% RH), add the following % of reading to the ohms offset listed above:*

	Range							
	30Ω	*300Ω*	*3kΩ,30kΩ*	*300kΩ*	*3MΩ*	*30MΩ*	*300MΩ*	*3GΩ*
% of rdg. (Typ.)•	*NOT SPECIFIED*		*.0016%*	*.0021%*	*.0070%*	*.053%*	*1.27%*	*12.6%*

[2] *Specification applies for two-wire ohms only, NPLC ≥1, inputs ≥10% of full scale, and within 24 hours of ACAL.*

90 DAY: Tcal ±1°C. After one hour warm-up.

Range	% of Reading[1] +		Ohms					
		NPLC	100	10	1	.1	.005	.0005
30Ω			NOT SPECIFIED					
300Ω			NOT SPECIFIED					
3kΩ	.0051%	+	9mΩ	10mΩ	10mΩ	50mΩ	400mΩ	4Ω
30kΩ	.0078%	+	90mΩ	100mΩ	100mΩ	500mΩ	4Ω	40Ω
300kΩ	.0360%	+	1Ω	1.1Ω	1.2Ω	5Ω	40Ω	400Ω
3MΩ	.307%	+	15Ω	17Ω	19Ω	80Ω	500Ω	5kΩ
30MΩ	2.95%	+	830Ω	860Ω	960Ω	1.5kΩ	6kΩ	50kΩ
300MΩ[2,3]	26%	+	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ
3GΩ[2,3]	100%	+	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ

[1] *Specification applies for ≤(40°C, 95% RH). For typical performance, ≤(25°C, 40% RH), add the following % of reading to the ohms offset listed above:*

	Range							
	30Ω	*300Ω*	*3kΩ,30kΩ*	*300kΩ*	*3MΩ*	*30MΩ*	*300MΩ*	*3GΩ*
% of rdg. (Typ.)•	*NOT SPECIFIED*		*.0048%*	*.0063%*	*.011%*	*.062%*	*2.8%*	*28%*

[2] *For 90 day and 1 year specification under stable condition (±1°C), following ACAL, use 24 hour specifications instead.*

[3] *Specification applies for two-wire ohms only, NPLC ≥1, inputs ≥10% of full scale.*

1 YEAR: Tcal ±5°C. After one hour warm-up.
For all ranges except 30MΩ, add 0.0015% of reading to the 90 day specifications.
For 30MΩ range, add 0.0125% of reading to the 90 day specifications.

Temperature Coefficient: For operating temperatures outside specified range, use the temperature coefficients for the HP 34520 DMM module.

Two-wire Ohms Accuracy: Add ±220Ω to all four-wire specifications.

Auto-Zero Off: Add additional accuracy error as specified for the HP 34520 DMM module.

Response Time: First reading meets accuracy specification with preprogrammed settling times and no external circuit capacitance. An additional delay of 0.1 sec is necessary after a range or function change.

TRUE RMS AC VOLTAGE

AC voltage accuracy specified **for sine wave inputs >10% of range. Maximum input** limited to **<125V Peak.** DC component <10% of AC component. AC slow filter on (ACBAND 20). After one hour warm-up. Auto-zero on.

For AC voltage measurement accuracy of signals switched through the HP 34502 reed relay module, add the error below (% of reading) to the AC voltage specifications for the HP 34520 DMM.

Frequency (Hz)	20-20k	20k-100k	100k-300k	300k-1M
% of Reading	±.03%	±.3%	±3%	±30%

03235-90001 Update 1 (MARCH 1987)

SYSTEM MEASUREMENT ACCURACY (continued)

Measurement accuracy for signals switched through the HP 34507 32 Channel Mercury Wetted Reed Relay Multiplexer

DC VOLTAGE

TABLE 10-5. DC VOLTS Measurement Accuracy: ±(% of reading + volts) Auto-zero on. Tcal is the temperature of calibration from 18°C to 28°C.

24 HOUR: Tcal ±1°C. After one hour warm-up. Accuracy relative to calibration standard.

Range	% of Reading	+	Volts, NPLC 100	10	1	.1	.005	.0005
30mV	.0017%	+	20.4μV	20.4μV	20.6μV	21.1μV	27μV	70μV
300mV	.001%	+	20.5μV	20.5μV	20.6μV	22μV	50μV	400μV
3V	.00085%	+	21μV	22μV	22μV	50μV	400μV	4mV
30V	.0014%	+	110μV	120μV	220μV	700μV	4mV	40mV
250V	.0033%	+	600μV	700μV	700μV	5mV	40mV	400mV

90 DAY: Tcal ±5°C. After one hour warm-up.

Range	% of Reading	+	Volts, NPLC 100	10	1	.1	.005	.0005
30mV	.0053%	+	20.7μV	20.8μV	21.1μV	21.7μV	29μV	70μV
300mV	.0038%	+	20.8μV	20.8μV	21.1μV	23μV	50μV	400μV
3V	.0030%	+	22μV	22μV	22μV	50μV	400μV	4mV
30V	.0050%	+	220μV	230μV	330μV	800μV	4mV	40mV
250V	.0065%	+	700μV	800μV	800μV	5mV	40mV	400mV

1 YEAR: Tcal ± 5°C. After one hour warm-up.
Add .001% of reading to 90 day specifications.

Temperature Coefficient: For operating temperatures outside specified range, refer to the temperature coefficients for the HP 34520 DMM module.

Auto-Zero Off: Add additional accuracy error as specified for the HP 34520 DMM module.

Range-to-Range Response Error: For default settling times, add 0.0005% of input voltage step to the first reading following a range change.

RESISTANCE

TABLE 10-6. RESISTANCE Measurement Accuracy: ±(% of reading + ohms) Four-wire ohms, auto-zero on, offset compensation on or off. Tcal is the temperature of calibration from 18°C to 28°C.

24 HOUR: Tcal ±1°C. After one hour warm-up, relative to calibration standard.

Range	% of Reading[1]	+	Ohms, NPLC 100	10	1	.1	.005	.0005
30Ω	NOT SPECIFIED							
300Ω	.002%	+	20.5mΩ	20.5mΩ	20.7mΩ	22mΩ	50mΩ	400mΩ
3kΩ	.0018%	+	21mΩ	22mΩ	22mΩ	50mΩ	400mΩ	4Ω
30kΩ	.0045%	+	210mΩ	220mΩ	220mΩ	500mΩ	4Ω	40Ω
300kΩ	.032%	+	2.2Ω	2.2Ω	2.2Ω	5Ω	40Ω	400Ω
3MΩ	.304%	+	24Ω	25Ω	27Ω	80Ω	500Ω	5kΩ
30MΩ	2.94%	+	830Ω	860Ω	960Ω	1.5kΩ	6kΩ	40kΩ
300MΩ[2]	25%	+	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ
3GΩ[2]	86%	+	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ

[1] Specification applies for ≤(40°C, 95% RH). For typical performance, ≤(25°C, 40% RH), add the following % of reading to the ohms offset listed above:

Range	30Ω	300Ω	3kΩ,30kΩ	300kΩ	3MΩ	30MΩ	300MΩ	3GΩ
% of rdg. (Typ.)*		.002%	.0016%	.0021%	.007%	.053%	1.33%	13.2%

[2] Specification applies for two-wire ohms only, NPLC ≥1, inputs ≥10% of full scale, and within 24 hours of ACAL.

90 DAY: Tcal ±1°C. After one hour warm-up.

Range	% of Reading[1]	+	Ohms, NPLC 100	10	1	.1	.005	.0005
30Ω	NOT SPECIFIED							
300Ω	.0058%	+	20.8mΩ	20.8mΩ	21.1mΩ	23mΩ	50mΩ	400mΩ
3kΩ	.0051%	+	22mΩ	22mΩ	22mΩ	50mΩ	400mΩ	4Ω
30kΩ	.0078%	+	220mΩ	220mΩ	220mΩ	500mΩ	4Ω	40Ω
300kΩ	.036%	+	2.2Ω	2.3Ω	2.3Ω	5Ω	40Ω	400Ω
3MΩ	.308%	+	25Ω	26Ω	27Ω	80Ω	500Ω	5kΩ
30MΩ	2.95%	+	840Ω	870Ω	970Ω	1.5kΩ	6kΩ	50kΩ
300MΩ[2,3]	26%	+	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ	100kΩ
3GΩ[2,3]	100%	+	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ	1MΩ

[1] Specification applies for ≤(40°C, 95% RH). For typical performance, ≤(25°C, 40% RH), add the following % of reading to the ohms offset listed above:

Range	30Ω	300Ω	3kΩ,30kΩ	300kΩ	3MΩ	30MΩ	300MΩ	3GΩ
% of rdg. (Typ.)*		.0058%	.0048%	.0063%	.011%	.062%	2.85%	28.5%

[2] For 90 day and 1 year specification under stable condition (±1°C), following ACAL, use 24 hour specifications instead.

[3] Specification applies for two-wire ohms only, NPLC ≥1, inputs ≥10% of full scale.

1 YEAR: Tcal ±5°C. After one hour warm-up.
For all ranges except 30MΩ, add 0.0015% of reading to the 90 day specifications.
For 30MΩ range, add 0.0125% of reading to the 90 day specifications.

Temperature Coefficient: For operating temperatures outside specified range, use the temperature coefficients for the HP 34520 DMM module.

Two-wire Ohms Accuracy: Add ±26Ω to all four-wire specifications.

Auto-Zero Off: Add additional accuracy error as specified for the HP 34520 DMM module.

Response Time: First reading meets accuracy specification with preprogrammed settling times and no external circuit capacitance. An additional delay of 0.1 sec is necessary after a range or function change.

TRUE RMS AC VOLTAGE

AC voltage accuracy specified **for sine wave inputs >10% of range.** DC component <10% of AC component. AC slow filter on (ACBAND 20). After one hour warm-up. Auto-zero on.

For AC voltage measurement accuracy of signals switched through the HP 34507 mercury-wetted reed relay module, add the error below (% of reading) to the AC voltage specifications for the HP 34520 DMM.

Frequency (Hz)	20-20k	20k-100k	100k-300k	300k-1M
% of Reading	±0%	±0%	±.03%	±.3%

HP 34520 DMM Noise Rejection with Switched Inputs

The following tables illustrate the noise rejection capability of the HP 34520 DMM module when measurement signals are switched from any of the 32 channel multiplexers (HP 34501, HP 34502, or HP 34507) via the backplane signal buses. Note that use of the guard in the HP 3235 improves the noise rejection of the DMM.

TABLE 10-7. Noise Rejection for DC VOLTS FUNCTION (dB)

		No. of Channels			
		≤ 78 (one frame)		≤ 2528 (8 connected frames)	
	NPLC	Guarded	Non-Guarded	Guarded	Non-Guarded
AC NMR (50/60 Hz ± .09%)	≤1	0	0	0	0
	1	60	60	60	60
	10	80	80	80	80
	100	90	90	90	90
AC ECMR (50/60 Hz ± .09%) 1kΩ imbalance in low lead	≤1	86	66	80	60
	1	146	126	140	120
	10	160	146	160	140
	100	160	156	160	150
DC ECMR (50/60 Hz ± .09%) 1 kΩ imbalance in low lead		>140	>120	>140	>120

TABLE 10-8. Noise Rejection for AC VOLTS FUNCTION (dB)

		No. of Channels			
		≤ 78 (one frame)		≤ 2528 (8 connected frames)	
	NPLC	Guarded	Non-Guarded	Guarded	Non-Guarded
AC ECMR (50/60 Hz ± .09%) 1 kΩ imbalance in low lead		86	66	80	60

SWITCHING AND OPERATING RATES

*The **supplemental characteristics** listed in this section are intended to provide information useful in applying the instrument by giving **typical**, but non-warranted performance information.*

TABLE 11-1. Switching and Operating Speeds (msec)
This table compares switching speeds for different HP 3235 switch modules. It also compares the execution rate from an external controller to the execution rate from an internal HP 3235 subroutine.

Times listed for the Series 9000 and the HP 1000 are for executing the command OUTPUT 709; "Operation/Command" via HP-IB. Included is the HP-IB transfer time and the time for the HP 3235 to execute the command. (Default conditions of INBUF OFF, SERIAL ON, DISP ON apply.)

Local execution rates reflect the time required to execute a command from a subroutine previously downloaded into HP 3235 memory. HP-IB transfer times for the command string are not included. (SERIAL ON, DISP OFF apply.)

Operation/ Command	Accessory (HP Model Number)	Series 9000 Model 200/300 (Basic)	HP 1000 (Fortran)	Local Execution
CLOSE 101 or OPEN 101	34501,34503,	19	21	15
	34506,34505	24	26	20
	34502	5.0	6.7	1.5
	34507	7.7	9.3	4.0
	34522	3.8	5.7	.62
CLOSE 101,102,...112 or OPEN 101,102,...112 (10 channels on same module)	34501,34503	33	34	20
	34502	18.4	20	5.5
	34506	194	197	178
	34507	20.9	22.6	8
	34522	19.0	20.7	6.2
SELECT 101	34501	34	35.6	30
	34503	32	35	28
	34502	6.5	8.2	3
	34505	42	45	38
	34506	44	45.6	40
	34507	11.7	13.4	8
	34522	4.8	6.5	1.2
CRESET	34501,34502,34503, 34507	20	21.5	16
	34505	25	26.5	21
	34506	83	84.2	78
	34522,34523	5.5	7.0	2
SSTATE name, 100 (single module)	34501,34502,34503, 34505,34506,34507, 34522,34523	6.7	8.3	2
RSTATE name, 100 (single module)	34501,34502,34503, 34507	21	22.5	16
	34505	26	28.5	21
	34506	162	163	151
	34522,34523	6.9	8.1	2.0
CONNECT 101, AB0 or DISCONNECT 101,AB0	34501,34502,34507	21	23	16
CONNECT 101,110 or DISCONNECT 101,110	34506	26	28	21
CONNECT ONLY, 101, AB0	34501,34502,34507,	35	37	30
CONNECT ONLY 101,110	34506	45	47	40
BBREAD 100, 1	34523	5	9.3	.24
or BBWRITE 800,1,511		4.9	6.8	.17
READ	34522	3.5	7.9	.47
or WRITE 511		3.8	5.4	.38
WRITEBLK (16 bit words, 1000 words, strobe handshake)	34522	25	25	23
READBLK (16 bit words, 1000 words, strobe handshake)	34522	25	26	24

OTHER SYSTEM OPERATING RATES

TABLE 11-2. Data Transfer Times from Internal Memory (msec)

Array Size		Output Format: Ascii Integer	Ascii Real	Binary Integer	Binary Real
HP 9000 Series 200/300 (Basic)	10	14.0	20.5	4.2	5.8
	100	108.8	174.6	9.0	24.6
HP 1000 (Fortran)	10	11.5	13.3	7.5	9.4
	100	202	263	17.5	39.5

TABLE 11-3. Interrupt Timing

Interrupt Latency: ≥ 1 msec
The time to begin execution of a subroutine in internal memory after interrupt from a HP 34522 Digital I/O module or HP 34523 Breadboard module when the mainframe is idle.

Interrupt Rate: 400 Hz
The maximum rate interrupts can be serviced with a subroutine in internal memory (service routine ≤ 1000 μsec). Interrupt rate decreases as execution time of service routine is increased.

CHARACTERISTIC HP 34520 DMM MEASUREMENT SPEEDS WITH SWITCHED INPUTS

The following tables are useful for estimating the measurement (and compare) time for a random scan of channels on a single plug-in module. If the measurement channels are in different banks, the bank relays are closed prior to initiating the MEAS or VERIFY command. Rates listed reflect NPLC 0, DISP OFF, AZERO OFF, fixed range and function. For operating rates at other integration times, refer to Table 11-7.

Table 11-4. Operating Speeds for MEAS or VERIFY from an External Controller
This table lists the measurement time per channel for a random scan of 10 channels.* The measurement sequence is initiated in the controller and readings are returned to the controller via HP-IB. Times apply for either ascii or binary output mode, except as noted.

Controller	Function	Module Type: HP 34501	HP 34502	HP 34507
HP 9000 Series 200/300 (Basic)	MEAS DCV,DCI,Ω (DELAY 0)[1]	308msec	76msec (ascii) 67msec (binary)	118msec
	MEAS ACV,ACI (preprogrammed settling time, ACBAND 400)[2]	1.31sec	1.08sec	1.13sec
	VERIFY DCV,DCI,Ω (DELAY 0)[1]	310msec	69msec	122msec
	VERIFY ACV,ACI (preprogrammed settling time, ACBAND 400)[2]	1.32sec	1.07sec	1.13sec
HP 1000 (Fortran)	MEAS DCV,DCI,Ω (DELAY 0)[1]	320msec	85msec 75msec	134msec
	MEAS ACV,ACI (preprogrammed settling time, ACBAND 400)[2]	1.32sec	1.09sec	1.14sec
	VERIFY DCV,DCI,Ω (DELAY 0)[1]	318msec	77msec	130msec
	VERIFY ACV,ACI (preprogrammed settling time, ACBAND 400)[2]	1.32sec	1.08sec	1.13sec

[1]Ω measurement rates apply for <300 kΩ ranges.
[2]Add 900 msec per channel for ACBAND <400 Hz.

**Times measured using a 10 element chan. list (e.g. MEAS DCV 101,115,134...127). For command execution from a controller, use of a prestored array instead of a channel list (e.g. MEAS DCV array name) will improve measurement rate. For the 10 channel case above, speed improves by 0.6 μsec per channel.*

TABLE 11-5. Operating Speeds for MEAS or VERIFY using Internal Memory
This table lists the operating time per channel for a random scan of 10 channels* as in Table 11-4, however the measurement sequence is executed from and readings stored to the internal HP 3235 memory.

	Function	HP 34501	HP 34502	HP 34507
MEAS or VERIFY	DCV,DCI,Ω (DELAY 0)[1]	294msec	54msec	107msec
	ACV,ACI (preprogrammed settling times, ACBAND 400)[2]	1.3sec	1.06sec	1.11sec

[1] *Ω measurement rates apply for < 300kΩ ranges.*
[2] *Add 900 msec per channel for ACBAND <400 Hz.*

Times measured using a 10 element channel list (e.g. MEAS DCV 101,115,134...127). This represents the fastest scan rate. For command execution from local memory, speed decreases by 0.2 μsec per channel if a prestored array is used instead of a channel list (e.g. MEAS DCV array name).

TABLE 11-6. General Operating Speeds for MEAS and VERIFY
To determine the measurement time per channel for a random scan of N channels, use the formulas provided in this table. Operating conditions from Table 11-5 apply. For integration cycles other than 10 μs (NPLC 0) use the per channel adders in Table 11-7.

		Fixed Time + Time/Channel (msec)		
	Function	HP 34501	HP 34502	HP 34507
MEAS or VERIFY	DCV,DCI,Ω (DELAY 0)[1]	8 + 29	31 + 2.4	25 + 8.2
	ACV,ACI (preprogrammed settling times, ACBAND 400)[2]	8 + 129	31 + 103	25 + 109

[1] *Ω measurement rates apply for < 300kΩ ranges.*
[2] *Add 900 msec per channel for ACBAND <400 Hz.*

TABLE 11-7. Effect of Integration Time on MEAS and VERIFY
To determine measurement (and compare) speeds with the HP 34520 DMM for NPLC other than 0, add the appropriate time listed below to the measurement time from either Table 11-4, 11-5, or 11-6.

	Increase in Measurement Time (msec/channel)					
NPLC	0.0005	0.005	0.1	1	10	100
60 Hz	0	0.1	2.1	18.2	202	2020
50 Hz	0	0.1	2.1	21.5	249	2420

Introduction

The following section summarizes the HP 3235 commands contained in this manual. It is divided into two groups: "Mainframe Commands" and "Plug-In Module Commands".

Mainframe Commands

HP Common Instrument Capabilities

CLR	Device Clear.
DISP or DSP	Enable/Disable Control Panel Display.
ERR?	Read Error Code.
ERRSTR?	Read Error Code and String.
ID?	Identify Mainframe or Plug-In Modules.
IDN?	Read Identity of System.
LCL or LOCAL	Return to Local Mode.
LOCK	Local Lockout.
RESET or RST	Read System or Slot.
REV?	Return Firmware Revision Code.
RMT	Execute Local Lockout.
RQS	Service Request Enable.
RQS?	Read Service Request Mask.
SER?	Read Serial Number.
SET	Load Instrument State From Controller.
SET?	Dump Instrument State to Controller.
STA?	Read Status Word.
STB?	Read Status Byte.
TEST	Perform Self-Test.
TRG	Pulse Backplane Trigger Bus.

Data Formats/Buffering

BLOCKOUT	Set Block Output Mode for Data to HP-IB.
CLROUT	Clear HP-IB Output Buffer.
END	Enable/Disable HP-IB EOI Function.
INBUF	Enable/Disable Command Input Buffer.
KEYS	Send Messages from Control Panel to Controller.
OFORMAT	Select Output Format.
OUTBUF	Enable/Disable Data Output Buffer.

Error Reporting

ERR?	Read Error Code.
ERRSTR?	Read Error Code and String.

Control Panel Operation

ADDR?	Read HP-IB Address of HP 3235.
BEEP	Enable/Disable Beeper Mode.
DISP or DSP	Enable/Disable Control Panel Display.

DISP? or DSP?	Enter Contents of Control Panel Display to Controller.
ECHO	Test Communication Between HP 3235 and Controller.
KEYS	Send Messages from Control Panel to Controller.
MON	Monitor Various HP 3235 Conditions.
PONSRQ	Enable/Disable Power-On Service Request.
READY?	Return "1" When HP 3235 is Not Busy.
SRQ	Programmed Service Request.

Monitoring

MON	Monitor Various HP 3235 Conditions.

Variable Storage and Output

DISP or DSP	Display Variable or String.
FETCH	Read Variable, Expression, or String.
MEM	Enable/Disable Memory Output Mode.
VREAD	Read Variable or Array.

User-Defined Key Commands

DEFKEY	Redefine Control Panel Key.
DEFKEY?	Read User-Defined Control Panel Key.

Memory Usage

CAT	Catalog of Arrays, Variables, Subroutines, States.
COMPRESS	Compress Subroutine.
DELSUB	Delete Subroutine from Memory.
MEM	Enable/Disable Memory Output Mode.
MEMAVAIL?	Return Size of Volatile and Continuous Memory.
PURGE	Purge Stored State from Memory.
SCRATCH	Delete Arrays, Variables, Subroutines, States.

General Purpose Math

+	Real or Integer Addition.
-	Real or Integer Subtraction.
*	Real or Integer Multiplication.
/	Real Division.
^	Exponentiation.
ABS	Absolute Value.
AND	Logical AND.
ATN	Arctangent.
BINAND	Binary AND.
BINCMP	Binary Complement.
BINEOR	Binary Exclusive-OR.
BINIOR	Binary Inclusive-OR.
BIT	Read Bit Value.
COS	Cosine.
DIM	Dimension Real Array.
DIV	Return Integer Portion of Division.
EXOR	Logical Exclusive-OR.
EXP	Exponent.

FILL	Fill Array.
FILLBIN	Fill Array with Binary Values.
INTEGER	Dimension Integer Array or Variable.
LET	LET Assignment.
LGT	Logarithm.
LOG	Natural Logarithm.
MOD	Return Remainder of Division.
NOT	Logical Complement.
OR	Logical OR.
REAL	Dimension Real Array or Variable.
ROTATE	Rotate Bits with Wraparound.
SHIFT	Shift Bits without Wraparound.
SIN	Sine.
SIZE?	Read Number of Elements in Array.
SQR	Square Root.

Limit Testing

LIMIT	Test Value Against Lower and Upper Limits.

Help System

HELP	Help System.

Subroutine Commands

ABORT	Abort Subroutine Execution.
CALL	Call Subroutine.
COMPRESS	Compress Subroutine.
CONT	Continue Execution of Stepped/Paused Subroutine.
DELSUB	Delete Subroutine from Memory.
END IF	End of **IF...THEN** Loop.
END WHILE	End of **WHILE** Loop.
FOR...NEXT	Repeat Loop Until Counter Passes Preset Value.
IF...THEN	Conditional Branching Within Subroutine.
LIST	List Subroutine.
PAUSE	Pause Execution of Subroutine.
PAUSED?	Returns "1" if Subroutine is Paused.
RETURN	Return from Subroutine to Calling Program.
RUN	Run Subroutine.
RUNNING?	Returns "1" if Subroutine is Running or Paused.
STEP	Single-Step Subroutine.
SUB	Begin Subroutine.
SUBEND	End Subroutine.
WHILE	Execute Loop Until Expression is True.

State Storage Commands

PURGE	Purge Stored State from Memory.
RSTATE	Recall Stored State.
SET	Load Instrument State from Controller.
SET?	Dump Instrument State to Controller.
SSTATE	Store State.

Self-Test Commands

DTEST or DTST	Perform Data Test.
FTEST or FST	Perform Fixtured Self-Test on Modules.
FTEST SYS or FTST SYS	Perform Fixtured Self-Test on System.
TEST or TST	Perform Confidence Self-Test.

Addressing Commands

USE	Select Device for Subsequent Commands.
USE?	Return the Current Use Device Number.

Backplane Trigger Bus Commands

AUTOTB	Enable/Disable Trigger Bus Connection.
BPOUT	Designate Trigger Bus Path into Switching Network.
DRIVEBPn	Drive Backplane Trigger Bus in Specific Frame.
DRIVEETBn	Drive Extender Trigger Bus Between Frame.
DRIVEEXT	Drive External Trigger BNC Connectors.
DRIVETBn	Drive Trigger Bus in all Frames.
TBDRIVE	Drive Trigger Bus from HP 34522 Digital I/O.
TBn?	Read Logic State of Trigger Bus.
TRIGBUF	Enable/Disable Trigger Buffer.

Timing/Synchronization Commands

SERIAL	Enable/Disable Serial Command Execution.
SET TIME	Set HP 3235 Internal Clock.
SETTLE	Wait for Commands to Settle.
TIME	Read HP 3235 Internal Clock.
WAIT	Wait Specific Time.
WAITFOR	Wait for Specific Trigger Event.

Timer Channel Commands

DISABLE INTR	Disable Interrupts on Timer Channel.
ENABLE INTR	Enable Interrupts on Timer Channel.
PULSE	Pulse Timer Channel.
READ COUNT	Read Timer Channel Count.
SET CLKSRC	Set Timer Channel Clock Source.
SET COUNT	Set Timer Channel Count.
SET OUTPUT	Enable/Disable Timer Channel Output.
SQWAVE	Output Square Wave from Timer Channel.

CRESET	Card (Module) Reset.
CTYPE? or CTYPE	Read Card (Module) Type.
EXTEND?	Read Extender Numbers.
ID?	Read Module or System Identity.
RESET or RST	Reset System or Slot.

Interrupt and Error Handling

DISABLE ERROR	Disable Module Error Generation.
DISABLE INTR	Disable Interrupt.
DISABLE INTR SYS	Disable Interrupts from Plug-In Module System.
ENABLE ERROR	Enable Module Error Generation.
ENABLE INTR	Enable Interrupt.
ENABLE INTR SYS	Enable Interrupts from Plug-In Module System.
INTR?	Read Which Slot Interrupted.
OFF INTR	Disable Interrupt-Initiated Action.
ON INTR	Enable Interrupt-Initiated Action.

Option 580 HP-IB Controller Commands

ASSIGN	Assign I/O Path.
AUTOST IS	Enter File Name as Source of Auto-Start Commands.
CAT	Return a List of Files on MSUS.
CLEAR	Device Clear.
CREATE ASCII	Create ASCII File.
CREATE BDAT	Create BDAT File
DEVICE IS	Assign External Device.
ENTER	Input Data from Device or File.
INITIAL	Initialize Mass Storage Device.
MEAS	Scan Using DEVICE IS DMM or Counter.
MSI	Set Default Mass Storage Device.
OUTPUT	Output Data to Device or File.
PRINT	Output Data to PRINTER IS Device.
PRINTER IS	Assigns Device as Recipient of PRINT Command.
PURGE	Remove File from Mass Storage Device.
SPOLL	Serial Poll.
TRIGGER	Send HP-IB Group Execute Trigger.
VERIFY	Scan Using DEVICE IS DMM or Counter.

Switching Plug-In Module Commands

ALLOW	Allow Prohibited Relay Closures.
CLOSE	Close Relay.
CLOSE?	Read State of Relay.
CONFIG	Select Relay Configuration.
CONNECT	Matrix Row/Column or Backplane Connection.
DISCONN	Matrix Row/Column or Backplane Disconnection.
JUMPER	Specify Jumper Settings.
MON	Monitor Various HP 3235 Conditions.
OPEN	Open Relay.
PROHIBIT	Prohibit Relay Closure.
PROHIBIT?	Read Prohibited Relay State.

SELECT	Open All Channels Then Close Specific Channels.
TRIGBUF	Control the Bidirectional Trigger Buffers.

Power Fail/Fixture Commands

FIXTURE?	Read Status of Quick Interconnect Fixture.
PFCLOSE	Close Relays When Power Fails.
PFCLOSE?	Read Power Fail State of Relay.
PFOPEN	Open Relays When Power Fails.
PFSAME	State of Relays Remains Unchanged When Power Fails.

HP 34520 Multimeter Commands

ACAL	Perform Auto-calibration Routines.
ACBAND	Select Slow or Fast AC Measurement Mode.
ACBAND?	Read Present AC Bandwidth.
ARANGE	Enable/Disable Autorange Function.
ARANGE?	Read ON/OFF State of Autorange Function.
AUXERR?	Read Auxiliary Register.
AZERO	Enable/Disable Autozero Function.
AZERO?	Read Autozero Setting.
CALNUM?	Read Calibration Number.
DELAY	Select Input Settling Delay.
DELAY?	Read Input Settling Delay.
FIXEDZ	Enable/Disable Fixed Resistance Function.
FIXEDZ?	Read Fixed Resistance Setting.
FSOURCE	Select Frequency Source.
FSOURCE?	Read Present Frequency Source.
FUNC	Select Measurement Function.
FUNC?	Read Measurement Function.
LFREQ	Set Line Frequency Reference.
LFREQ?	Read Line Frequency Reference.
LINE?	Read Line Frequency Switch.
NPLC	Set Integration Time.
NPLC?	Read Integration Time in PLCs.
NRDGS	Set Number of Readings Per Trigger.
NRDGS?	Read Number of Readings Per Trigger.
OCOMP	Enable/Disable Offset Compensated Ohms Function.
OCOMP?	Read State of Offset Compensated Ohms Function.
RANGE	Set Measurement Range.
RANGE?	Read Present Measurement Range.
READ	Read Measurement Results.
TBUFF	Enable/Disable Trigger Buffering.
TBUFF?	Read ON/OFF State of Trigger Buffering Function.
TERM	Select Input Terminals.
TERM?	Read Present Input Terminals.
TIMER	Select Timer Interval.
TIMER?	Read Present Timer Interval.
TRIG	Select Measurement Trigger Event.
TRIG?	Read Measurement Trigger Event.
USE	Select Multimeter to Receive Subsequent Commands.
USE?	Read Use Device.
VMCMPLT	Select Destination for Voltmeter Complete Signal.
VMCMPLT?	Read Voltmeter Complete Destination.

HP 34521 AC/DC Source Module Command

ADJUST ACV	Generate Remote-Sensed Sinewave Voltage.
ADJUST DCV	Generate Remote-Sensed DC Voltage.
APPLY ACV	Generate AC Sine Waveform.
APPLY DCI	Generate DC Current.
APPLY DCMEMI	Generate DC Current Triggered-Sequence.
APPLY DCMEMV	Generate DC Voltage Triggered-Sequence.
APPLY DCV	Generate DC Voltage.
APPLY RPV	Generate AC Ramp Waveform.
APPLY SQV	Generate AC Square Waveform.
APPLY WFV	Generate AC Arbitrary Waveform.
ARANGE	Enable/Disable Autorange Function.
ARANGE?	Read ON/OFF State of Autorange Function.
DCOFF	Select DC Offset Voltage.
DCRES	Select DC Resolution Mode.
DCRES?	Read DC Resolution Setting.
DELAY	Set Delay Time.
DELAY?	Read Present Delay Time.
DUTY	Set Duty Cycle.
FILLAC	Precompute Sine Waveform.
FILLRP	Precompute Ramp Waveform.
FILLWF	Precompute Arbitrary Waveform.
FREQ	Set Output Frequency.
IMP	Set Output Impedance.
OUTPUT?	Read Present Output Level.
PANG	Set Phase Angle.
RANGE	Set Output Range.
RANGE?	Read Present Output Range.
REFIN	Select Reference Frequency Input Source.
REFIN?	Read Reference Frequency Input Source.
REFOUT	Select Reference Frequency Output Destination.
REFOUT?	Read Reference Frequency Output Destination.
SYNCOUT	Set SYNC Signal Destination.
SYNCOUT?	Read SYNC Signal Destination.
TERM	Select Output Terminal.
TERM?	Read Output Terminal.
TRIGIN	Select Trigger Event.
TRIGIN?	Read Trigger Event.
TRIGMODE	Select AC Triggering Mode.
TRIGOUT	Enable/Disable TRIGGER Outputs.
TRIGOUT?	Read TRIGGER Output State.
USE	Select Source Module to Receive Subsequent Commands.
USE?	Read Use Device.

HP 34522 Digital I/O Commands

CLEAR EVENT	Clear Event Bit.
CLEAR EVENTERR	Clear Event Error Bit.
CLOSE	Clear Digital I/O Bits.
CLOSE?	Read Logic State of Digital I/O Bit.
DISABLE ERROR	Disable Error Generation.
DISABLE EVENT	Disable Event Bit.
DISABLE EVENTERR	Disable Event Error Bit.

DISABLE INTR	Disable Interrupt.
EDGE	Set Active Edge for Event Bit.
ENABLE ERROR	Enable Error Generation.
ENABLE EVENT	Enable Event Bit.
ENABLE EVENTERR	Enable Event Error Bit.
ENABLE INTR	Enable Interrupt.
HSDONE?	Read Status of Output Handshake.
HSTIME	Set Handshake Delay Time.
HSTYPE	Select Input or Output Handshake Type.
LSENSE	Select Logic Sense for Data and Handshake Lines.
OPEN	Set Digital I/O Bits.
PULLUP	Configure Output Ports as Totem-Pole or Open-Collector.
RBIT	Read Logic Value of Single Bit.
READ	Read 8-, 16-, or 32-Bit Word from Port.
READBLK	Read Series of Words from Port.
RFLG	Read Logic State of FLG Handshake Line.
RLEVENT	Read Live State of Event Bits.
RSEVENT	Read Stored State of Event Bits.
RSEVERR	Read Stored Event Error State of Event Bits.
SELECT	Set All Bits on Port Then Clear Specified Bits.
SRTRIG	Select Second Rank Trigger for Buffered Transfers.
TBDRIVE	Drive Trigger Bus from HP 34522 Digital I/O.
USE	Select Digital I/O Port for Subsequent Commands.
USE?	Read Use Device.
WBIT	Write "0" or "1" to Single Bit.
WCTL	Write "0" or "1" to CTL Handshake Line.
WRITE	Write 8-, 16-, or 32-Bit Word to Port.
WRITEBLK	Write Series of Words to Port.
XFERMODE	Select Live or Buffered Output Mode.
XFERWIDTH	Define Port Size.

HP 34523 Breadboard Commands

ALLOW	Allow Prohibited Relay Closures.
BBREAD	Read Value from Breadboard Register.
BBWRITE	Write Value to Breadboard Register.
CLOSE	Close Relay.
CLOSE?	Read State of Relay.
OPEN	Open Relay.
PROHIBIT	Prohibit Relay Closure.
PROHIBIT?	Read Prohibited Relay State.
TRIGBUF	Enable/Disable Trigger Buffer.

HP 34524 D/A Converter Commands

ADJUST DCV	Generate Remote-Sensed DC Voltage.
APPLY DCI	Generate DC Current.
APPLY DCV	Generate DC Voltage.
OUTPUT?	Read Present Output Level.
USE	Select D/A Channel to Receive Subsequent Commands.
USE?	Read Use Device Selected.

Scanning Commands

CHCLOSED	Select Port For Scan Channel Closed Pulse.
MEAS	Scan using Multimeter Module.
PSCAN	Scan Channel Pairs using External Instrument.
SADV	Select Scan Advance Source.
SCAN	Scan Channel List using External Instrument.
STRIG	Select Scan Trigger Source.
VERIFY	Scan Channel List and Perform Limit Checks.

CIIL Commands

CIIL	Enable CIIL Device Clear Function.
CNF	CIIL Internal Confidence Test Command.
CON	CIIL Version of **CLOSE** Command.
DIS	CIIL Version of **OPEN** Command.
GAL	CIIL Version of **CLR** Command.
IST	CIIL Internal Self-Test Command.
STA	CIIL Version of **ERRSTR?** Command.

Error Messages

Syntax

ERROR 01 "INCOMPLETE COMMAND"
ERROR 02 "SYNTAX"
ERROR 03 "CANNOT RE-TYPE A VARIABLE"
ERROR 04 "ERROR IN #A BLOCK"
ERROR 05 "ARRAY SIZE OR TYPE MISMATCH"
ERROR 06 "COMMAND TOO LONG"

Subroutines

ERROR 21 "NOT ALLOWED IN SUB"
ERROR 22 "ALLOWED ONLY IN SUB"
ERROR 23 "NEXT WITHOUT FOR"
ERROR 24 "NEXT VARIABLE NOT SAME AS FOR VARIABLE"
ERROR 25 "EXPECTED NEXT"
ERROR 26 "ELSE OR END IF WITHOUT IF"
ERROR 27 "EXPECTED END IF"
ERROR 28 "END WHILE WITHOUT WHILE"
ERROR 29 "EXPECTED END WHILE"
ERROR 30 "CONTROL VARIABLE IN USE"

Memory

ERROR 41 "OUT OF MEMORY"
ERROR 42 "TOO MANY NESTED CALLS"
ERROR 43 "TOO MANY NESTED STRUCTURES"

Subroutine Execution/Stored States

ERROR 51 "SUB WAS DELETED"
ERROR 52 "NO ACTIVE SUB"
ERROR 53 "SUB NOT PAUSED"
ERROR 54 "SUB IS RUNNING"
ERROR 55 "CANNOT DELSUB AN ACTIVE SUB"
ERROR 56 "STATE WAS PURGED"
ERROR 57 "STATE DOES NOT MATCH"
ERROR 58 "SUB WAS COMPRESSED"

Parameters

ERROR 61 "OUT OF RANGE"
ERROR 62 "EMPTY SLOT"
ERROR 63 "NO SUCH EXTENDER"
ERROR 64 "WRONG CARD TYPE"
ERROR 65 "COMMAND INCOMPATIBLE WITH SETUP"
ERROR 66 "SUBSCRIPT OUT OF BOUNDS"
ERROR 67 "ARRAY TOO SMALL"
ERROR 68 "ON BLOCK NOT FOUND"
ERROR 69 "MUST BE IN LOCAL"
ERROR 70 "SETTINGS CONFLICT"

ERROR 71 "INVALID FUNCTION"
ERROR 72 "NO HP-IB DEVICE"

General

ERROR 81 "COMMAND TIMEOUT"
ERROR 82 "BUSY TOO LONG"
ERROR 83 "BP ERROR FROM SLOT"
ERROR 84 "UNEXPECTED INTR"
ERROR 85 "SYSTEM TRIG TOO FAST"
ERROR 86 "PROHIBITED SWITCH"

System

ERROR 91 "SYSTEM ERROR"
ERROR 92 "BUS ERROR"
ERROR 93 "SYSTEM TRAP"
ERROR 94 "MATH ERROR"
ERROR 95 "CPU EXCEPTION"

Multimeter

Triggering

ERROR 101 "ILLEGAL TRIGGERING SETUP"
ERROR 102 "MM TRIG TOO FAST"

Cal Setup

ERROR 111 "PERIOD CAL ILLEGAL"
ERROR 112 "EXT OHMS CAL ILLEGAL"
ERROR 113 "AC OFFSET CAL ILLEGAL"
ERROR 114 "CAL SWITCH NOT ENABLED"
ERROR 115 "CAL SWITCH ENABLED"
ERROR 116 "CAL INPUT OUT OF RANGE"

Cal Hardware Failure

ERROR 121 "AC OFFSET ACAL FAIL"
ERROR 122 "AC FLATNESS ACAL FAIL"
ERROR 123 "OHMS PRECHARGE ACAL FAIL"
ERROR 124 "EXT OHMS ACAL FAIL"

Multimeter Hardware Failure

ERROR 131 "CAL MEMORY LOST"
ERROR 132 "AUTO CAL MEMORY LOST -- ACALINIT required"
ERROR 133 "BAD CHECKSUM ON"
ERROR 134 "MM Hardware Error"
ERROR 135 "MM timeout"

Digital I/O

ERROR 151 "INTERRUPT OVERRUN"

Source and D/A Converter

ERROR 161 "MIN/MAX LIMIT ERROR"
ERROR 162 "EXCESSIVE TERMINAL VOLTAGE, RELAYS OPEN"
ERROR 163 "ENABLE CAL RAM"
ERROR 164 "CAL RAM FAIL"
ERROR 165 "SPURIOUS HW ERROR, RELAYS OPEN"
ERROR 166 "ERROR STUCK, CARD"
ERROR 167 "ADJUST FAILED"

Power Actuator

ERROR 169 "FIXTURE LOW CURRENT"

Mass Storage

ERROR 171 "IMPROPER FILE NAME"
ERROR 172 "IMPROPER DEVICE TYPE"
ERROR 173 "IMPROPER MSUS"
ERROR 174 "UNSUPPORTED DRIVE TYPE"
ERROR 175 "UNSUPPORTED SECTOR SIZE"
ERROR 176 "DRIVE NOT FOUND OR BAD ADDRESS"
ERROR 177 "INVALID UNIT/VOLUME NUMBER"
ERROR 178 "INVALID MASS STORAGE PARAMETER"
ERROR 179 "MEDIA CHANGED OR NOT IN DRIVE"
ERROR 180 "MEDIA IS WRITE PROTECTED"
ERROR 181 "DIRECTORY FULL"
ERROR 182 "NO ROOM ON DISK"
ERROR 183 "FILE NOT FOUND"
ERROR 184 "DUPLICATE FILE NAME"
ERROR 185 "IMPROPER FILE TYPE"
ERROR 186 "PATH NAME NOT ASSIGNED"
ERROR 187 "FILE OPEN"
ERROR 188 "END OF FILE FOUND"
ERROR 189 "INITIALIZATION FAILED"
ERROR 190 "MASS STORAGE SYSTEM ERROR"
ERROR 191 "BAD SELECT CODE"
ERROR 192 "I/O OPERATION NOT ALLOWED"
ERROR 193 "UNINITIALIZED MEDIA"

Self-Test

ERROR 201 "PON TEST"
ERROR 202 "DTACK FAIL"
ERROR 203 "CARD ID FAIL"
ERROR 204 "CARD BUSY (D7) STUCK HIGH"
ERROR 205 "CARD BUSY (D7) STUCK LOW"
ERROR 206 "BUSY TOO SHORT"
ERROR 207 "BUSY TOO LONG"
ERROR 208 "RE-TRIG BUSY TIME FAIL"

ERROR 209 "BP /BUSY STUCK LOW"
ERROR 210 "RELAY READBACK FAIL"
ERROR 211 "TRIGGER BUS FAIL"
ERROR 212 "HP-IB CTLR TEST"

ERROR 221 "MM SELF TEST"
ERROR 222 "DIG SELF TEST"
ERROR 223 "RELAY OPEN"
ERROR 224 "RELAY SHORTED"
ERROR 225 "TRIGBUF FAIL"
ERROR 226 "SOURCE SELF TEST"
ERROR 227 "RELAY DRIVER SELF TEST"
ERROR 228 "DAC SELF TEST"

HP 3235 HP-IB Capabilities

Introduction

The following topics, arranged in alphabetic order, deal with specific HP-IB command messages. These commands are functions of the HP-IB interface and all HP-IB compatible instruments respond, in some manner, to these commands. Refer to the remote HP-IB programming manual for each instrument connected to the HP-IB interface of your test system to see how they respond to these commands.

Refer also to the I/O programming manual for your system controller for specific syntax and interface activity for each message. The IEEE-488 terminology is shown in parenthesis following each command title. The examples given here apply to the HP Series 80, Series 200 and Series 300 desktop computers.

Many of the commands in this section have counterparts that are HP 3235 command headers. This makes the commands more valuable and easy to understand. Use these commands if your system controller allows them, otherwise use the respective HP 3235 command header.

The HP 3235 ignores the Parallel Poll (PPOLL) and Pass Control commands.

ABORT (IFC)

The ABORT command aborts all HP-IB communication. The HP 3235A exits the remote state and returns to local mode. It does not reset, self-test, clear buffers, etc.

Example

```
ABORT 7        ! Abort all communication in isc 7
```

CLEAR (DCL or SDC)

The HP 3235 executes a CLEAR command immediately upon receipt (if INBUF is ON). The CLEAR command is the same as the HP 3235 CLR command and does the following:

- Clears HP-IB input and output buffers.
- Clears the error list and bit 5 in the Status Register.
- Aborts any running subroutines.
- If a subroutine is being downloaded from the system controller when you execute CLEAR, the subroutine is aborted and not stored.
- Clears the control panel display
- Does not reset any of the Plug-In Modules.
- If a scan is in progress it is aborted, the relays stay in whatever state they were in when CLEAR was executed.
- The multimeter does not change states (modes).

Example

```
CLEAR 7 ! Clears all instruments (DCL) on interface 7

CLEAR 709  ! Clears only the instrument (SDC) at address 09.
```

LOCAL (GTL)

The LOCAL command removes the HP 3235 from the remote state. The HP 3235 continues to execute any commands received while it was in the remote state and are still in the input buffer. LOCAL allows unlimited use of the control panel keyboard. See the sections on LOCAL and REMOTE modes earlier in this chapter.

Example

```
LOCAL 7  ! All devices on interface 7 go to local state.

LOCAL 709  ! Only device at address 09 goes to local state.
```

LOCAL LOCKOUT (LLO)

LOCAL LOCKOUT disables the Local key on the control panel. LOCAL LOCKOUT is the same as RMT. To regain full control of the keyboard, execute: LOCAL 709. With the Local key disabled, you cannot change the state of the HP 3235. You can, however, query the system and monitor results.

Example

```
LOCAL LOCKOUT 7  ! sends LOCAL LOCKOUT to all instruments
                       on the interface.
```

REMOTE

The REMOTE 709 command puts the HP 3235 in its remote state. The REMOTE 7 command, does not, in itself, put the HP 3235 in its remote state. After sending REMOTE 7, the HP 3235 will go to its remote state only after the system controller addresses it.

You will seldom find it necessary to execute the REMOTE command. The REMOTE command is independent of any other HP-IB activity and is sent on a single interface bus line called REN. Most controllers set the REN line true at power-on or when reset which has the same effect as executing REMOTE 7. Therefore, when your controller addresses the HP 3235, such as OUTPUT 709; "CLOSE 101", the mainframe will go to its remote state.

Example

```
REMOTE 7  ! Sets the REN line true, does not set HP 3235 to remote.

REMOTE 709  ! sets HP 3235 to remote state.
```

SPOLL (Serial Poll)

The SERIAL POLL command, like the STB? command (HP 3235 command set), returns a number representing the set bits in the status register (status byte). The returned number is the sum of the decimal values of the bits that are set.

Note that SPOLL does not clear any of the bits in the status register except bit 6, SRQ. The main difference between SPOLL and STB? is that SPOLL does not interrupt the processor to get the register information. With the STB? command, the mainframe processor must process the command and will always show busy (bit 4). Therefore, use SPOLL to determine the readiness of the mainframe for more commands.

Status Register Bits

The Status Register bits and their corresponding weights are:

Bit Number	Decimal Weight	Description (Bit is set when:)
0	1	Data is available and will be sent when the HP 3235 is addressed to talk.
1	2	Always zero.
2	4	User SRQ from keyboard or subroutine.
3	8	Power-on occurred, reset, or control panel LOCAL key pressed or LCL executed.
4	16	READY -- HP-IB input buffer is empty and no command or CALLed subroutine is in progress.
5	32	Error Occurred.
6	64	SRQ from this instrument (Service Request).
7	128	Always zero.
8	256	Always zero

For more information on interrupts, refer to Chapter 7 in this manual.

Example

```
10 OUTPUT 709; "RST"
20 A=SPOLL (709)
30 PRINT A
40 END
```

The following program returns the same results but will show bit 4 as true:

```
10 OUTPUT 709; "RST"
20 OUTPUT 709";"STB?"
30 ENTER 709;A
40 PRINT A
50 END
```

TRIGGER (GET)

If the DRIVETB command is set for TRG (see Chapter 5), then the HP-IB TRIGGER command pulses the specified trigger bus line. If the Input buffer is on (INBUF ON), then GET executes immediately even if other commands are still in the input buffer waiting for processing.

Example

```
TRIGGER 7   ! Triggers the interface bus

TRIGGER 709   ! Triggers only the instrument at address 09
```

Glossary

Analog Bus The analog bus system has four separate (isolated) 2-wire buses. These buses carry analog signals between plug-in modules, to the extenders, or to external instruments (see also AB*n*).

AB*n* Analog bus parameter. Specifies which of the four analog buses is used. There are four choices for *n*: 0, 1, 2, or 3 (see also Analog Bus).

array_name Mainframe array name. Array names may contain up to 10 characters. The first character must be a letter (A-Z) but the remaining nine characters can be letters, numbers (0-9), the underscore character ("_"), or the question mark ("?"). Array names must not be the same as HP 3235 commands or parameters (see Appendix A for a list of reserved keywords), or stored state names.

Backplane Relay Backplane relays connect the relay multiplexer banks to the analog backplane. Backplane relays addresses have the form *es9n*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders), *s* is the slot number (0 through 9), and *9n* is the backplane relay number (90 through 93).

Bank Relay Bank relays connect the relay multiplexer banks together. Bank relays addresses have the form *es7n*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders), *s* is the slot number (0 through 9), and *7n* is the bank relay number (70 through 72).

ch# Channel number parameter. Has the form *esnn*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders), *s* is the slot number (0 through 9), and *nn* is the channel number.

ch_list Channel list parameter. Has the form *esnn*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders), *s* is the slot number (0 through 9), and *nn* is the channel number.

The *ch_list* parameter can specify a single channel (*esnn*), multiple channels (*esnn, esnn, ...*), sequential channels (*esnn-esnn*), groups of sequential channels (*esnn-esnn, esnn-esnn*), or any combination of channels and groups.

Channel The term *channel* refers to any relay or group of relays on a plug-in module that can switch a signal from one user connector to another user connector (see also *ch#* and *ch_list*).

connector# HP 34506 Coaxial Matrix Module row and column numbers. Has the form *esnn*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders), *s* is the slot number (0 through 9), and *nn* is the connector number (row and column).

Controller Any computer that is capable of controlling the HP 3235 over HP-IB.

device# The *device#* parameter is used by the **USE** command to specify the device (such as the HP 34520 Multimeter Module or an HP 34522 Digital I/O port) for subsequent commands.

Has the form *esnn*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders), *s* is the slot number (0 through 9), and *nn* is the device number. To specify an entire module or slot, the device number (*nn*) is specified as 00 (*es00*). To specify a port on the Digital I/O module, the device number is one of the following ports: 00, 10, 20, or 30 (*es00, es10, es20,* or *es30*).

Extender Relay Each HP 34560 System Expansion Module has four "extender bus relays", one relay for each of the four (AB0 through AB3) analog buses. These relays allow you to isolate any one analog bus within a frame (i.e., mainframe or extender) from the same bus within another frame.

Extender relay addresses have the form *9e0n*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders) and *n* is the extender relay number (0 through 3).

frame# Frame (mainframe or extender) number parameter. Specifies either the mainframe (0) or an extender (1 through 7).

relay# Relay number parameter. Has the form *esnn*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders), *s* is the slot number (0 through 9), and *nn* is the relay number.

relay_ list Relay list parameter. Has the form *esnn*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders), *s* is the slot number (0 through 9), and *nn* is the relay number.

The *relay_list* parameter can specify a single relay (*esnn*), multiple relays (*esnn, esnn, ...*), sequential relays (*esnn-esnn*), groups of sequential relays (*esnn-esnn, esnn-esnn*), or any combination of relays and groups.

Relay In addition to channel relays, some plug-in modules contain *backplane relays, bank relays,* and *trigger buffer relays.* These relays are referred to simply as *relays* (see also *relay#* and *relay_list*).

slot# Slot number parameter. Has the form *es00*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders) and *s* is the slot number (0 through 9).

slot_list Slot list parameter. Has the form *es00*, where *e* is the mainframe or extender number (0 = mainframe, 1-7 = extenders) and *s* is the slot number (0 through 9).

The *slot_list* parameter can specify a single slot (*es00*), multiple slots (*es00, es00,* ...), sequential slots (*es00-es00*), groups of sequential slots (*es00-es00, es00-es00*) or any combination of slots and groups.

Slot Each HP 3235 mainframe and extender contains 10 *slots* to install plug-in modules. The slots are numbered from 000 through 900 for each frame (see also *slot#* and *slot_list*).

sub_name Mainframe subroutine name. Subroutine names may contain up to 10 characters. The first character must be a letter (A-Z) but the remaining nine characters can be letters, numbers (0-9), the underscore character ("_"), or the question mark ("?"). Subroutine names must not be the same as HP 3235 commands or parameters (see Appendix A for a list of reserved keywords), previously defined array or variable names, or stored state names.

System Controller Any computer that is capable of controlling one or more instruments over HP-IB.

variable Mainframe variable name. Variable names may contain up to 10 characters. The first character must be a letter (A-Z) but the remaining nine characters can be letters, numbers (0-9), the underscore character ("_"), or the question mark ("?"). Variable names must not be the same as HP 3235 commands or parameters (see Appendix A for a list of reserved keywords), or stored state names.

- D -

- E -

- F -

- G -

- H -

- I -

- K -

- L -

- M -

- N -

- O -

- P -

- Q -

- R -

- S -

- T -

- U -

- V -

- W -