tek.com/keithley

Model 2750 Multimeter/Switch System

Reference Manual

2750-901-01 Rev. A August 2023

2750-901-01A

KEITHLEY
A Tektronix Company

Model 2750

Multimeter/Switch System

Reference Manual

© 2023, Keithley Instruments, LLC

Cleveland, Ohio, U.S.A.

All rights reserved.

Any unauthorized reproduction, photocopy, or use of the information herein, in whole or in part, without the prior written approval of Keithley Instruments, LLC, is strictly prohibited.

These are the original instructions in English.

All Keithley Instruments product names are trademarks or registered trademarks of Keithley Instruments, LLC. Other brand names are trademarks or registered trademarks of their respective holders.

Microsoft, Visual C++, Excel, and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Document number: 2750-901-01 Rev. A August 2023

KEITHLEY
A Tektronix Company

Safety precautions

The following safety precautions should be observed before using this product and any associated instrumentation. Although some instruments and accessories would normally be used with nonhazardous voltages, there are situations where hazardous conditions may be present.

This product is intended for use by personnel who recognize shock hazards and are familiar with the safety precautions required to avoid possible injury. Read and follow all installation, operation, and maintenance information carefully before using the product. Refer to the user documentation for complete product specifications.

If the product is used in a manner not specified, the protection provided by the product warranty may be impaired.

The types of product users are:

Responsible body is the individual or group responsible for the use and maintenance of equipment, for ensuring that the equipment is operated within its specifications and operating limits, and for ensuring that operators are adequately trained.

Operators use the product for its intended function. They must be trained in electrical safety procedures and proper use of the instrument. They must be protected from electric shock and contact with hazardous live circuits.

Maintenance personnel perform routine procedures on the product to keep it operating properly, for example, setting the line voltage or replacing consumable materials. Maintenance procedures are described in the user documentation. The procedures explicitly state if the operator may perform them. Otherwise, they should be performed only by service personnel.

Service personnel are trained to work on live circuits, perform safe installations, and repair products. Only properly trained service personnel may perform installation and service procedures.

Keithley products are designed for use with electrical signals that are measurement, control, and data I/O connections, with low transient overvoltages, and must not be directly connected to mains voltage or to voltage sources with high transient overvoltages. Measurement Category II (as referenced in IEC 60664) connections require protection for high transient overvoltages often associated with local AC mains connections. Certain Keithley measuring instruments may be connected to mains. These instruments will be marked as category II or higher.

Unless explicitly allowed in the specifications, operating manual, and instrument labels, do not connect any instrument to mains.

Exercise extreme caution when a shock hazard is present. Lethal voltage may be present on cable connector jacks or test fixtures. The American National Standards Institute (ANSI) states that a shock hazard exists when voltage levels greater than 30 V RMS, 42.4 V peak, or 60 VDC are present. A good safety practice is to expect that hazardous voltage is present in any unknown circuit before measuring.

Operators of this product must be protected from electric shock at all times. The responsible body must ensure that operators are prevented access and/or insulated from every connection point. In some cases, connections must be exposed to potential human contact. Product operators in these circumstances must be trained to protect themselves from the risk of electric shock. If the circuit is capable of operating at or above 1000 V, no conductive part of the circuit may be exposed.

Do not connect switching cards directly to unlimited power circuits. They are intended to be used with impedance-limited sources. NEVER connect switching cards directly to AC mains. When connecting sources to switching cards, install protective devices to limit fault current and voltage to the card.

Before operating an instrument, ensure that the line cord is connected to a properly-grounded power receptacle. Inspect the connecting cables, test leads, and jumpers for possible wear, cracks, or breaks before each use.

When installing equipment where access to the main power cord is restricted, such as rack mounting, a separate main input power disconnect device must be provided in close proximity to the equipment and within easy reach of the operator.

For maximum safety, do not touch the product, test cables, or any other instruments while power is applied to the circuit under test. ALWAYS remove power from the entire test system and discharge any capacitors before connecting or disconnecting cables or jumpers, installing or removing switching cards, or making internal changes, such as installing or removing jumpers.

Do not touch any object that could provide a current path to the common side of the circuit under test or power line (earth) ground. Always make measurements with dry hands while standing on a dry, insulated surface capable of withstanding the voltage being measured.

For safety, instruments and accessories must be used in accordance with the operating instructions. If the instruments or accessories are used in a manner not specified in the operating instructions, the protection provided by the equipment may be impaired.

Do not exceed the maximum signal levels of the instruments and accessories. Maximum signal levels are defined in the specifications and operating information and shown on the instrument panels, test fixture panels, and switching cards.

When fuses are used in a product, replace with the same type and rating for continued protection against fire hazard.

Chassis connections must only be used as shield connections for measuring circuits, NOT as protective earth (safety ground) connections.

If you are using a test fixture, keep the lid closed while power is applied to the device under test. Safe operation requires the use of a lid interlock.

If a ⏚ screw is present, connect it to protective earth (safety ground) using the wire recommended in the user documentation.

The ⚠ symbol on an instrument means caution, risk of hazard. The user must refer to the operating instructions located in the user documentation in all cases where the symbol is marked on the instrument.

The ⚡ symbol on an instrument means warning, risk of electric shock. Use standard safety precautions to avoid personal contact with these voltages.

The ♨ symbol on an instrument shows that the surface may be hot. Avoid personal contact to prevent burns.

The ⏚ symbol indicates a connection terminal to the equipment frame.

If this Hg symbol is on a product, it indicates that mercury is present in the display lamp. Please note that the lamp must be properly disposed of according to federal, state, and local laws.

The **WARNING** heading in the user documentation explains hazards that might result in personal injury or death. Always read the associated information very carefully before performing the indicated procedure.

The **CAUTION** heading in the user documentation explains hazards that could damage the instrument. Such damage may invalidate the warranty.

The **CAUTION** heading with the ⚠ symbol in the user documentation explains hazards that could result in moderate or minor injury or damage the instrument. Always read the associated information very carefully before performing the indicated procedure. Damage to the instrument may invalidate the warranty.

Instrumentation and accessories shall not be connected to humans.

Before performing any maintenance, disconnect the line cord and all test cables.

To maintain protection from electric shock and fire, replacement components in mains circuits — including the power transformer, test leads, and input jacks — must be purchased from Keithley. Standard fuses with applicable national safety approvals may be used if the rating and type are the same. The detachable mains power cord provided with the instrument may only be replaced with a similarly rated power cord. Other components that are not safety-related may be purchased from other suppliers as long as they are equivalent to the original component (note that selected parts should be purchased only through Keithley to maintain accuracy and functionality of the product). If you are unsure about the applicability of a replacement component, call a Keithley office for information.

Unless otherwise noted in product-specific literature, Keithley instruments are designed to operate indoors only, in the following environment: Altitude at or below 2,000 m (6,562 ft); temperature 0 °C to 50 °C (32 °F to 122 °F); and pollution degree 1 or 2.

To clean an instrument, use a cloth dampened with deionized water or mild, water-based cleaner. Clean the exterior of the instrument only. Do not apply cleaner directly to the instrument or allow liquids to enter or spill on the instrument. Products that consist of a circuit board with no case or chassis (e.g., a data acquisition board for installation into a computer) should never require cleaning if handled according to instructions. If the board becomes contaminated and operation is affected, the board should be returned to the factory for proper cleaning/servicing.

Safety precaution revision as of June 2018.

Table of contents

Introduction 1-1

Welcome 1-1

Extended warranty 1-1

Contact information 1-1

Customer documentation 1-2

Product software and drivers 1-2

Remote operations 2-1

Introduction 2-1

Operation enhancements 2-1
- Pseudocards 2-1
- Autozero 2-2
- dB calculation 2-2
- Separate function setups 2-2
- DCV input divider 2-2
- Multiple channel operation 2-3

GPIB communications interface 2-3
- GPIB standards 2-3
- GPIB connections 2-4
- Select GPIB and set the primary address 2-5
- IEEE-488 front-panel operation 2-6
- General bus commands 2-7

Programming syntax 2-10
- Command words 2-10
- Query commands 2-12
- Program messages 2-12
- Response messages 2-15
- Multiple response messages 2-15
- Message exchange protocol 2-15

RS-232 serial interface 2-15
- RS-232 connections 2-16
- Select and configure the RS-232 interface 2-17
- Sending and receiving data 2-18
- Baud rate 2-18
- Signal handshaking (flow control) 2-18
- Terminator 2-19
- Error messages 2-19

Switching and scanning 3-1

Switching module operation 3-1

Plug-in switching modules 3-2

Pseudocards 3-3

Channel assignments 3-4

Identify installed modules 3-5
- CARD menu 3-5

Switching module queries (remote operation) ... 3-7

Closing and opening channels using system channel operation ... 3-9
Close and open operation ... 3-11
Exercise 2 — Closing and opening channels (system channel operation) ... 3-12

Remote programming — system channel control commands ... 3-12

Non-amp and non-measure switching modules ... 3-13
Non-amps switching modules ... 3-13
Making amps measurements ... 3-14
Non-measure switching modules ... 3-14

Multiple channel operation ... 3-15
Controlling multiple channels ... 3-16
Multiple channel operation anomalies ... 3-19
Dual independent multiplexers ... 3-20

Scanning fundamentals ... 3-25
Sequential and non-sequential scans ... 3-26
Scan process ... 3-26
Trigger models when using the front panel ... 3-27
Trigger model using remote operations ... 3-33

Scan configuration ... 3-34
Channel setup considerations ... 3-35
Effects of function changes on the scan list ... 3-35
Saving the configured scan ... 3-36
Scan reset ... 3-37
Simple scan ... 3-37
Advanced scan ... 3-39
Monitor channel ... 3-43
Autochannel configuration ... 3-44
Saving setup ... 3-45
Autoscan ... 3-45

Scan operation ... 3-46
Basic scan ... 3-46
Manual/external trigger scan ... 3-47
Monitor scan (analog trigger) ... 3-48

Scanning using remote programming ... 3-51
Channel setup ... 3-52
Buffer ... 3-52

Scanning examples ... 3-52
External trigger scan ... 3-53
Monitor scan ... 3-54

Making measurements ... 4-1

Performance considerations ... 4-1
Autozero ... 4-1
Line cycle synchronization ... 4-2
Channel list parameter (<clist>) ... 4-3
DCV input divider ... 4-3
Crest factor ... 4-4
Low-level considerations ... 4-4

Measurement range ... 4-7
Manual ranging ... 4-8
Autoranging ... 4-8
Ranges when scanning ... 4-8

Digital filters 4-9
Filter characteristics 4-10
Filter control and configuration 4-11
Digital filter example 4-12

Relative offset 4-12
Setting the relative offset for non-scan measurements 4-12
Setting the relative offset for scanning 4-13

Math 4-14
mx+b 4-15
Percent 4-16
Reciprocal (1/X) 4-17

Ratio and channel average 4-18
Basic ratio or channel average operation 4-19
Ratio and channel average when scanning 4-20
Advanced scan configuration notes 4-21

dB 4-21
dB and scanning 4-22

Low-level voltage measurement considerations 4-22
Magnetic fields 4-22
Radio frequency interference 4-23

Remote DMM measurements using the front-panel inputs 4-23
Exercise: Basic DMM measurements using remote commands 4-24

Signal processing sequence 4-25
Basic signal processing 4-26
Signal processing using instrument features 4-27
Signal processing using Ratio or Ch Avg 4-28
Data flow (remote operation) 4-29

Measurement considerations 5-1

Introduction 5-1

Thermoelectric potentials 5-1
Thermoelectric coefficients 5-2

Thermoelectric generation 5-2
Minimizing thermal EMFs 5-3
Canceling residual thermal offsets 5-4

Source resistance noise 5-4
Johnson noise equation 5-4
Minimizing source resistance noise 5-5

Magnetic fields 5-5

Radio frequency interference 5-6

Ground loops 5-6

Shielding 5-8

Meter loading 5-9

Limits 6-1

Introduction 6-1

Limits 6-1
Tips to use Limit 2 test 6-3
Overflow readings 6-3
Scanning and limits 6-3

Basic limits operation 6-4
Setting limits 6-4
Beeper settings 6-4
Enabling and disabling limits 6-4

Digital I/O 6-5
Digital input (trigger link input) 6-5

Digital outputs 6-5
Logic sense 6-7
Pulse option 6-7
Master limit latch 6-7
Controlling external devices using sink mode 6-7
Using logic control as the source mode 6-9
Setting digital output 6-9
Scanning and the digital output 6-11

Application: Sorting resistors 6-11
Limits 6-11
Digital outputs 6-13

Triggering 7-1

Triggering 7-1
Idle 7-2
Control source and event detection 7-2
Delay (auto or manual) 7-2
Device action 7-4
Output trigger 7-4
External triggering 7-4
Triggering using remote programming 7-10

Programming examples for triggering 7-14
Example: Trigger and return a single reading 7-14
Example: Trigger and return multiple readings 7-15
Example: Return a single reading (continuous triggering) 7-16

Relay closure count 7-17

Reading buffers 8-1

Buffer overview 8-1

Autoclear 8-1
Enabling and disabling buffer autoclear 8-2

Timestamps 8-2
Setting time and date 8-3
Selecting timestamp 8-4

Storing readings 8-4

Recalling readings 8-5

Buffer statistics 8-6

Remote programming and the buffer 8-7

Command reference 9-1

Using the SCPI command reference 9-1
Command name and summary table 9-3
Command usage 9-4
Command details 9-4
Example section 9-4
Related commands list 9-5
Channel assignments 9-5

SCPI reference tables 9-6
CALCulate command summary 9-7
DISPlay command summary 9-9
FORMat command summary 9-10
ROUTe command summary 9-10
SENSe command summary 9-11
STATus command summary 9-24
SYSTem command summary 9-25
TRACe command summary 9-27
TRIGger command summary 9-28
UNIT command summary 9-29
:CONFigure? 9-30
:FETCh? 9-32
:INITiate:CONTinuous 9-33
:INITiate[:IMMediate] 9-34
:MEASure? 9-35
:READ? 9-37

CALCulate subsystem 9-39
:CALCulate[1]:DATA:FRESh? 9-39
:CALCulate[1]:DATA[:LATest]? 9-39
:CALCulate[1]:FORMat 9-40
:CALCulate[1]:KMATh:MBFactor 9-41
:CALCulate[1]:KMATh:MMFactor 9-43
:CALCulate[1]:KMATh:MUNits 9-44
:CALCulate[1]:KMATh:PERCent 9-45
:CALCulate[1]:KMATh:PERCent:ACQuire 9-45
:CALCulate[1]:STATe 9-46

CALCulate2 subsystem 9-47
:CALCulate2:DATA? 9-47
:CALCulate2:FORMat 9-48
:CALCulate2:IMMediate 9-49
:CALCulate2:STATe 9-50

CALCulate3 subsystem 9-50
:CALCulate3:LIMit1:CLEar:AUTO 9-50
:CALCulate3:LIMit1:CLEar[:IMMediate] 9-51
:CALCulate3:LIMit1:FAIL? 9-51
:CALCulate3:LIMit1:LOWer[:DATA] 9-52
:CALCulate3:LIMit1:STATe 9-53
:CALCulate3:LIMit1:UPPer[:DATA] 9-54
:CALCulate3:LIMit2:CLEar:AUTO 9-54
:CALCulate3:LIMit2:CLEar[:IMMediate] 9-55
:CALCulate3:LIMit2:FAIL? 9-55
:CALCulate3:LIMit2:LOWer[:DATA] 9-56
:CALCulate3:LIMit2:STATe 9-57
:CALCulate3:LIMit2:UPPer[:DATA] 9-58
:CALCulate3:MLIMit:LATChed 9-58
:CALCulate3:OUTPut:LSENse 9-59
:CALCulate3:OUTPut:PULSe:TIMe 9-59

:CALCulate3:OUTPut:PULSe[:STATe] 9-60
:CALCulate3:OUTPut[:STATe] 9-60

DISPlay subsystem 9-61
:DISPlay[:WINDow[1]:TEXT:DATA 9-61
:DISPlay[:WINDow[1]]:TEXT:STATe 9-62
:DISPlay:ENABle 9-62

FORMat subsystem 9-63
:FORMat:BORDer 9-63
:FORMat:ELEMents 9-64
:FORMat[:DATA] 9-66

ROUTe subsystem 9-69
:ROUTe:CLOSe 9-69
:ROUTe:CLOSe:ACONfigure 9-69
:ROUTe:CLOSe:COUNt:INTerval 9-70
:ROUTe:CLOSe:COUNT? 9-71
:ROUTe:CLOSe:STATe? 9-71
:ROUTe:MONitor 9-72
:ROUTe:MONitor:DATA? 9-72
:ROUTe:MONitor:POINts 9-73
:ROUTe:MONitor:STATe 9-73
:ROUTe:MULTiple:CLOSe 9-74
:ROUTe:MULTiple:CLOSe:STATe? 9-75
:ROUTe:MULTiple:OPEN 9-75
:ROUTe:OPEN:ALL 9-76
:ROUTe:SCAN:LSELect 9-76
:ROUTe:SCAN:NVOLatile 9-77
:ROUTe:SCAN:TSOurce 9-77
:ROUTe:SCAN[:INTernal] 9-78

SENSe[1] subsystem 9-79
[:SENSe[1]]:CAVerage:DELay 9-79
[:SENSe[1]]:CAVerage[:STATe] 9-80
[:SENSe[1]]:CONTinuity:THReshold 9-80
[:SENSe[1]]:CURRent:AC:APERture 9-81
[:SENSe[1]]:CURRent:AC:AVERage:COUNt 9-81
[:SENSe[1]]:CURRent:AC:AVERage:TCONtrol 9-82
[:SENSe[1]]:CURRent:AC:AVERage:WINDow 9-82
[:SENSe[1]]:CURRent:AC:AVERage[:STATe] 9-83
[:SENSe[1]]:CURRent:AC:DETector:BANDwidth 9-84
[:SENSe[1]]:CURRent:AC:DIGits 9-85
[:SENSe[1]]:CURRent:AC:NPLCycles 9-86
[:SENSe[1]]:CURRent:AC:RANGe:AUTO 9-86
[:SENSe[1]]:CURRent:AC:RANGe[:UPPer] 9-87
[:SENSe[1]]:CURRent:AC:REFerence 9-88
[:SENSe[1]]:CURRent:AC:REFerence:STATe 9-89
[:SENSe[1]]:CURRent:AC:REFerence:ACQuire 9-89
[:SENSe[1]]:CURRent[:DC]:APERture 9-90
[:SENSe[1]]:CURRent[:DC]:AVERage:COUNt 9-91
[:SENSe[1]]:CURRent[:DC]:AVERage:TCONtrol 9-91
[:SENSe[1]]:CURRent[:DC]:AVERage:WINDow 9-92
[:SENSe[1]]:CURRent[:DC]:AVERage[:STATe] 9-93
[:SENSe[1]]:CURRent[:DC]:DIGits 9-94
[:SENSe[1]]:CURRent[:DC]:NPLCycles 9-95
[:SENSe[1]]:CURRent[:DC]:RANGe:AUTO 9-95
[:SENSe[1]]:CURRent[:DC]:RANGe[:UPPer] 9-96
[:SENSe[1]]:CURRent[:DC]:REFerence 9-97
[:SENSe[1]]:CURRent[:DC]:REFerence:ACQuire 9-98
[:SENSe[1]]:CURRent[:DC]:REFerence:STATe 9-98
[:SENSe[1]]:DATA:FRESh? 9-99

[:SENSe[1]]:DATA[:LATest]?.... 9-100
[:SENSe[1]]:FREQuency:APERture 9-101
[:SENSe[1]]:FREQuency:DIGits 9-102
[:SENSe[1]]:FREQuency:REFerence 9-103
[:SENSe[1]]:FREQuency:REFerence:ACQuire 9-104
[:SENSe[1]]:FREQuency:REFerence:STATe 9-104
[:SENSe[1]]:FREQuency:THReshold:VOLTage:RANGe 9-105
[:SENSe[1]]:FRESistance:APERture 9-106
[:SENSe[1]]:FRESistance:AVERage:COUNt 9-106
[:SENSe[1]]:FRESistance:AVERage:TCONtrol 9-107
[:SENSe[1]]:FRESistance:AVERage:WINDow 9-107
[:SENSe[1]]:FRESistance:AVERage[:STATe] 9-108
[:SENSe[1]]:FRESistance:DCIRcuit 9-108
[:SENSe[1]]:FRESistance:DIGits 9-109
[:SENSe[1]]:FRESistance:NPLCycles 9-110
[:SENSe[1]]:FRESistance:OCOMpensated 9-110
[:SENSe[1]]:FRESistance:RANGe:AUTO 9-111
[:SENSe[1]]:FRESistance:RANGe[:UPPer] 9-112
[:SENSe[1]]:FRESistance:REFerence 9-113
[:SENSe[1]]:FRESistance:REFerence:ACQuire 9-113
[:SENSe[1]]:FRESistance:REFerence:STATe 9-114
[:SENSe[1]]:FUNCtion 9-115
[:SENSe[1]]:PERiod:APERture 9-116
[:SENSe[1]]:PERiod:DIGits 9-117
[:SENSe[1]]:PERiod:REFerence 9-118
[:SENSe[1]]:PERiod:REFerence:ACQuire 9-119
[:SENSe[1]]:PERiod:REFerence:STATe 9-120
[:SENSe[1]]:PERiod:THReshold:VOLTage:RANGe 9-120
[:SENSe[1]]:RATio:DELay 9-121
[:SENSe[1]]:RATio[:STATe] 9-122
[:SENSe[1]]:RESistance:APERture 9-123
[:SENSe[1]]:RESistance:AVERage:COUNt 9-123
[:SENSe[1]]:RESistance:AVERage:TCONtrol 9-124
[:SENSe[1]]:RESistance:AVERage:WINDow 9-124
[:SENSe[1]]:RESistance:AVERage[:STATe] 9-125
[:SENSe[1]]:RESistance:DIGits 9-126
[:SENSe[1]]:RESistance:NPLCycles 9-127
[:SENSe[1]]:RESistance:RANGe:AUTO 9-127
[:SENSe[1]]:RESistance:RANGe[:UPPer] 9-128
[:SENSe[1]]:RESistance:REFerence 9-129
[:SENSe[1]]:RESistance:REFerence:ACQuire 9-130
[:SENSe[1]]:RESistance:REFerence:STATe 9-130
[:SENSe[1]]:TEMPerature:APERture 9-131
[:SENSe[1]]:TEMPerature:AVERage:COUNt 9-131
[:SENSe[1]]:TEMPerature:AVERage:TCONtrol 9-132
[:SENSe[1]]:TEMPerature:AVERage:WINDow 9-132
[:SENSe[1]]:TEMPerature:AVERage[:STATe] 9-133
[:SENSe[1]]:TEMPerature:DIGits 9-134
[:SENSe[1]]:TEMPerature:FRTD:ALPHa 9-135
[:SENSe[1]]:TEMPerature:FRTD:BETA 9-136
[:SENSe[1]]:TEMPerature:FRTD:DELTa 9-137
[:SENSe[1]]:TEMPerature:FRTD:RZERo 9-138
[:SENSe[1]]:TEMPerature:FRTD:TYPE 9-139
[:SENSe[1]]:TEMPerature:NPLCycles 9-140
[:SENSe[1]]:TEMPerature:REFerence 9-140
[:SENSe[1]]:TEMPerature:REFerence:ACQuire 9-141
[:SENSe[1]]:TEMPerature:REFerence:STATe 9-142
[:SENSe[1]]:TEMPeratureTCouple:ODETect 9-142
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:RSELect 9-143
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:SIMulated 9-144
[:SENSe[1]]:TEMPerature:TCouple:TYPE 9-145

[:SENSe[1]]:TEMPerature:THERmistor[:TYPE] 9-146
[:SENSe[1]]:TEMPerature:TRANsducer 9-147
[:SENSe[1]]:VOLTage:AC:APERture 9-147
[:SENSe[1]]:VOLTage:AC:AVERage:COUNt 9-148
[:SENSe[1]]:VOLTage:AC:AVERage:TCONtrol 9-148
[:SENSe[1]]:VOLTage:AC:AVERage:WINDow 9-149
[:SENSe[1]]:VOLTage:AC:AVERage[:STATe] 9-150
[:SENSe[1]]:VOLTage:AC:DETector:BANDwidth 9-150
[:SENSe[1]]:VOLTage:AC:DIGits 9-151
[:SENSe[1]]:VOLTage:AC:NPLCycles 9-152
[:SENSe[1]]:VOLTage:AC:RANGe:AUTO 9-152
[:SENSe[1]]:VOLTage:AC:RANGe[:UPPer] 9-153
[:SENSe[1]]:VOLTage:AC:REFerence 9-154
[:SENSe[1]]:VOLTage:AC:REFerence:ACQuire 9-154
[:SENSe[1]]:VOLTage:AC:REFerence:STATe 9-155
[:SENSe[1]]:VOLTage[:DC]:APERture 9-156
[:SENSe[1]]:VOLTage[:DC]:AVERage:COUNt 9-156
[:SENSe[1]]:VOLTage[:DC]:AVERage:TCONtrol 9-157
[:SENSe[1]]:VOLTage[:DC]:AVERage:WINDow 9-157
[:SENSe[1]]:VOLTage[:DC]:AVERage[:STATe] 9-158
[:SENSe[1]]:VOLTage[:DC]:DIGits 9-159
[:SENSe[1]]:VOLTage[:DC]:IDIVider 9-160
[:SENSe[1]]:VOLTage[:DC]:NPLCycles 9-160
[:SENSe[1]]:VOLTage[:DC]:RANGe:AUTO 9-161
[:SENSe[1]]:VOLTage[:DC]:RANGe[:UPPer] 9-162
[:SENSe[1]]:VOLTage[:DC]:REFerence 9-163
[:SENSe[1]]:VOLTage[:DC]:REFerence:ACQuire 9-164
[:SENSe[1]]:VOLTage[:DC]:REFerence:STATe 9-165

STATus subsystem 9-166
:STATus:MEASurement:CONDition? 9-166
:STATus:MEASurement:ENABle 9-166
:STATus:MEASurement[:EVENt]? 9-167
:STATus:OPERation:CONDition? 9-167
:STATus:OPERation:ENABle 9-168
:STATus:OPERation[:EVENt]? 9-168
:STATus:PRESet 9-169
:STATus:QUEStionable:CONDition? 9-169
:STATus:QUEStionable:ENABle 9-170
:STATus:QUEStionable[:EVENt]? 9-170
:STATus:QUEue:CLEar 9-170
:STATus:QUEue:DISable 9-171
:STATus:QUEue:ENABle 9-172
:STATus:QUEue[:NEXT]? 9-173

SYSTem subsystem 9-174
:SYSTem:AZERo:STATe 9-174
:SYSTem:BEEPer[:STATe] 9-175
:SYSTem:CARD:ACHannel:END? 9-175
:SYSTem:CARD:ACHannel[:STARt]? 9-176
:SYSTem:CARD:AOUTput:END? 9-176
:SYSTem:CARD:AOUTput[:STARt]? 9-177
:SYSTem:CARD:BANKs? 9-178
:SYSTem:CARD:CSOhms? 9-178
:SYSTem:CARD:DINPut:END? 9-179
:SYSTem:CARD:DINPut[:STARt]? 9-179
:SYSTem:CARD:DOUTput:END? 9-180
:SYSTem:CARD:DOUTput[:STARt]? 9-180
:SYSTem:CARD:ISOLated? 9-181
:SYSTem:CARD:MUX? 9-181
:SYSTem:CARD:SNOpen? 9-182
:SYSTem:CARD:SNUMber? 9-182

:SYSTem:CARD:SWOpen? 9-183
:SYSTem:CARD:SWRevision? 9-183
:SYSTem:CARD:TCHannel? 9-184
:SYSTem:CARD:TCOMpensated? 9-184
:SYSTem:CARD:VCHannel:END? 9-185
:SYSTem:CARD:VCHannel[:STARt]? 9-186
:SYSTem:CARD:VMAX? 9-186
:SYSTem:CLEar 9-187
:SYSTem:DATE 9-187
:SYSTem:ERRor? 9-188
:SYSTem:FRSWitch? 9-188
:SYSTem:KCLick 9-189
:SYSTem:KEY 9-190
:SYSTem:LFRequency? 9-191
:SYSTem:LOCal 9-192
:SYSTem:LSYNc:STATe 9-192
:SYSTem:PCARd 9-193
:SYSTem:POSetup 9-194
:SYSTem:PRESet 9-195
:SYSTem:REMote 9-196
:SYSTem:RNUMber:RESet 9-196
:SYSTem:RWLock 9-197
:SYSTem:TIME 9-197
:SYSTem:TSTamp:RELative:RESet 9-198
:SYSTem:TSTamp:TYPE 9-198
:SYSTem:VERSion? 9-199
SYSTem:CARD:VCHannel[:STARt]? 9-199

TRACe subsystem 9-200
:TRACe:CLEar:AUTO 9-200
:TRACe:CLEar[:IMMediate] 9-201
:TRACe:DATA:SELected? 9-201
:TRACe:DATA? 9-202
:TRACe:FEED 9-203
:TRACe:FEED:CONTrol 9-203
:TRACe:FREE? 9-204
:TRACe:NEXT? 9-204
:TRACe:NOTify 9-205
:TRACe:POINts 9-206
:TRACe:TSTamp:FORMat 9-207
:TRACe:TSTamp:TYPE? 9-207

TRIGger subsystem 9-208
:ABORt 9-208
:TRIGger[:SEQuence[1]]:COUNt 9-208
:TRIGger[:SEQuence[1]]:DELay 9-209
:TRIGger[:SEQuence[1]]:DELay:AUTO 9-210
:TRIGger[:SEQuence[1]]:SAMPle:COUNt 9-210
:TRIGger[:SEQuence[1]]:SIGNal 9-211
:TRIGger[:SEQuence[1]]:SOURce 9-211
:TRIGger[:SEQuence[1]]:TIMer 9-212

UNIT subsystem 9-213
:UNIT:TEMPerature 9-213
:UNIT:VOLTage[:DC] 9-214
:UNIT:VOLTage[:DC]:DB:REFerence 9-214
:UNIT:VOLTage:AC 9-215
:UNIT:VOLTage:AC:DB:REFerence 9-216

Common commands 10-1

Common command summary 10-1

Common command descriptions 10-2
*CLS 10-3
*ESE 10-4
*ESR? 10-5
*IDN? 10-6
*OPC 10-6
*OPT? 10-8
*RCL 10-9
*RST 10-10
*SAV 10-10
*SRE 10-11
*STB? 10-12
*TRG 10-14
*TST? 10-15
*WAI 10-15

Status model 11-1

Status structure 11-1
Status byte and SRQ 11-3
Status register sets 11-3
Queues 11-3

Clearing registers and queues 11-4

Programming enable registers 11-4

Reading registers 11-5

Status byte and service request (SRQ) 11-6
Status byte register 11-7
Service request enable register 11-8
Serial poll and SRQ 11-8
Status byte and service request commands 11-9
Serial poll programming example 11-9

Status register sets 11-11
Standard event status register 11-11
Operation event register 11-13
Measurement event register 11-14
Questionable event register 11-16
Condition registers 11-17
Event registers 11-17
Event enable registers 11-18

Queues 11-19

Temperature equations 12-1

Temperature equations 12-1

Thermocouple equation 12-1
Type B inverse function polynomial 12-2
Type E inverse function polynomial 12-2
Type J inverse function polynomial 12-3
Type K inverse function polynomial 12-3
Type N inverse function polynomial 12-4

Type R inverse function polynomial 12-4
Type S inverse function polynomial 12-5
Type T inverse function polynomial 12-5

Thermistor equation 12-6
Selecting a thermistor 12-6
Converting Kelvin to °C 12-7
Example 12-7

RTD equation 12-7

IEEE-488 bus overview 13-1

Introduction 13-1

Bus description 13-2

Bus lines 13-4
Data lines 13-4
Bus management lines 13-4
Handshake lines 13-5

Bus commands 13-6
Uniline commands 13-7
Universal multiline commands 13-7
Addressed multiline commands 13-8
Addressed commands 13-8
Unaddressed commands 13-8
Common commands 13-9
SCPI commands 13-9
Command codes 13-9
Typical command sequences 13-11
IEEE command groups 13-12

Interface function codes 13-13

Section 1

Introduction

In this section:

Welcome 1-1
Extended warranty 1-1
Contact information 1-1
Customer documentation 1-2
Product software and drivers..... 1-2

Welcome

Thank you for choosing a Keithley Instruments product. The Model 2750 Multimeter/Switch System combines precision measurement, switching, and control in a tightly integrated enclosure for either rack-mount or benchtop applications. The 2750 offers extended low ohms measurement capability and supports up to five 7700 switch cards for a maximum of 200, 2-pole multiplexed channels. It also contains a built-in 20 mV clamp that helps protect sensitive devices from damage and prevents self-heating errors during dry circuit testing.

This manual describes advanced operations and programming options for the 2750.

Extended warranty

Additional years of warranty coverage are available on many products. These valuable contracts protect you from unbudgeted service expenses and provide additional years of protection at a fraction of the price of a repair. Extended warranties are available on new and existing products. Contact your local Keithley Instruments office, sales partner, or distributor for details.

Contact information

If you have any questions after you review the information in this documentation, please contact your local Keithley Instruments office, sales partner, or distributor. You can also call the Tektronix corporate headquarters (toll-free inside the U.S. and Canada only) at 1-800-833-9200. For worldwide contact numbers, visit tek.com/contact-tek.

Customer documentation

The documentation for the 2750 includes a User's Manual and Reference Manual. You can access them from tek.com/keithley.

- **User's Manual:** Includes installation, instrument description, operation, and maintenance information.
- **Reference Manual:** Includes advanced operation topics and maintenance information. Programmers looking for a command reference and users looking for an in-depth description of how the instrument works should refer to the Reference Manual.

Product software and drivers

Go to the Product Support and Downloads web page to download drivers and software for your instrument.

Available drivers and software include:

- **KickStart Software:** Enables quick test setup and data visualization when using one or more instruments.
- **IVI-COM Driver:** An IVI instrument driver you can use to create your own test applications in C/C++, VB.NET, or C# programming languages. It can also be called from other languages that support calling a DLL or ActiveX (COM) object. Refer to IVI Foundation (ivifoundation.org) for additional information.
- **LabVIEW™ Software drivers:** Drivers to communicate with NI™ LabVIEW™ Software.
- **Keithley I/O layer:** Manages the communications between Keithley instrument drivers and software applications and the instrument itself. The I/O Layer handles differences in communications required to support GPIB, serial, ethernet, and other communications buses so that drivers and software applications do not need to handle the differences themselves.

Section 2

Remote operations

In this section:

Introduction 2-1
Operation enhancements 2-1
GPIB communications interface 2-3
Programming syntax 2-10
RS-232 serial interface 2-15

Introduction

You can use GPIB (IEEE-488) or RS-232 remote communications with the 2750. This section describes 2750 features that are only available through remote communications, how to set up remote communications, and provides basic programming syntax.

Operation enhancements

The following topics describe some operations you can do over the IEEE-488 bus and RS-232 interface that you cannot do from the front panel.

Pseudocards

Using remote operation, you can assign a pseudocard to an empty switching module slot. With a pseudocard installed, the 2750 operates as if the switching module is installed in the 2750. This feature allows you to configure your system without having the actual switching module installed. There is a pseudocard for every Keithley Model 7700 series switching module.

For more information, refer to Pseudocards (on page 3-3) and :SYSTem:PCARd (on page 9-193).

Autozero

Autozero is part of the normal measurement process to assure stable, accurate measurements. Autozero can be disabled to increase measurement speed. However, disabling autozero can cause readings to become inaccurate over time and temperature changes. Autozero can only be disabled using remote programming. It cannot be disabled from the front panel. For more information, refer to Autozero (on page 4-1).

dB calculation

Using remote programming, you can select the dB calculation for DC or AC voltage. The dB calculation makes it possible to compress a large range of measurements into a much smaller scope. Refer to dB (on page 4-21) to select and configure the dB calculation.

You cannot select dB from the front panel. However, once it is selected you can take the 2750 out of remote and use the front panel. When the instrument is reset to default conditions (or turned off), the dB setting is lost.

Separate function setups

Some of the front-panel settings are global, which mean that when you define the setting on one function, it also applies to the other functions. For example, if you set DCV for 3½ digits, all the other functions are also be set to 3½ digits.

Using remote programming, each function can have its own setup. For example, DCV can be set to 3½ digits, ACI can be set to 4½ digits, and DCI can be set to 5½ digits.

Global settings from the front panel that can be set separately using remote programming include digits, rate, and filter configuration (except on/off, which can be set separately).

NOTE

This does not include functions set in a scan channel setup. For scan channels, separate settings for digits, rate, and filter configuration can be set from either the front panel or remote programming. Refer to Switching and scanning (on page 3-1) for details on scan channel setup.

DCV input divider

Using remote programming, you can enable the DCV input divider for the 100 mV, 1 V, and 10 V ranges. When enabled, the input resistance for these DCV ranges are reduced to 10 MΩ. Refer to [:SENSe[1]]:VOLTage[:DC]:IDIVider (on page 9-160).

Multiple channel operation

For normal system channel operation, when one measurement channel is closed, the previous measurement channel opens. With the use of the ROUTe:MULTiple commands, you gain independent control of all switching module channels, including the relays that connect the input signal to the DMM. Multiple channel operation (on page 3-15) and ROUTe subsystem (on page 9-69) for additional information.

GPIB communications interface

The 2750 has a built-in GPIB (IEEE-488) interface. Over this interface, you can send data to the instrument and receive data from the instrument.

The following information describes how to make the GPIB connections to the instrument and how to set the GPIB address. It also describes how to set the data elements that are included with each reading and how to operate the 2750 from the front panel when GPIB is connected. General bus commands are also described.

GPIB standards

The GPIB is the IEEE-488 instrumentation data bus with hardware and programming standards originally adopted by the IEEE (Institute of Electrical and Electronic Engineers) in 1975. The 2750 conforms to these standards:

- IEEE-488.1-1987
- IEEE-488.2-1992

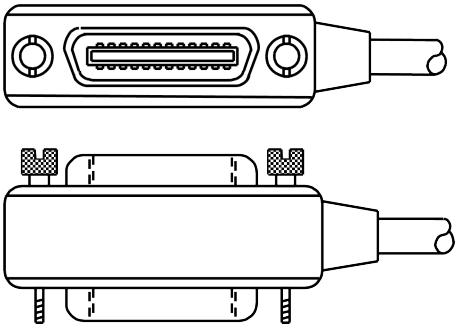
The standards define a syntax for sending data to and from instruments, how an instrument interprets this data, what registers should exist to record the state of the instrument, and a group of common commands.

The 2750 also conforms to the standard SCPI 1996.0 (Standard Commands for Programmable Instruments). This standard defines a command language protocol. It goes one step farther than IEEE-488.2-1992 and defines a standard set of commands to control every programmable aspect of an instrument.

GPIB connections

The 2750 can be connected to the IEEE-488 bus through a cable equipped with standard IEEE-488 connectors. An example is shown in the following figure. The connector can be stacked to allow a number of parallel connections to one instrument. There are two screws on each connector to ensure that connections remain secure. Present standards call for metric threads, which are identified with dark-colored screws. Earlier versions have different screws, which are silver-colored. Do not use these types of connectors on the 2750. It is designed for metric threads.

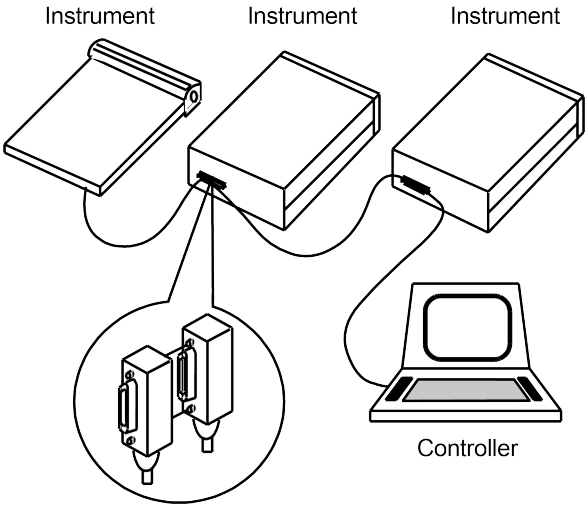
Figure 1: IEEE-488 connector



A typical connection scheme for a multi-instrument test system is shown in the following figure. It is recommended that you stack no more than three connectors on any one instrument to avoid possible mechanical damage.

To minimize interference caused by electromagnetic radiation, use only shielded IEEE-488 cables.

Figure 2: Multi-instrument connections



You can have up to 15 devices connected to a GPIB interface, including the controller. The maximum cable length is the lesser of either:

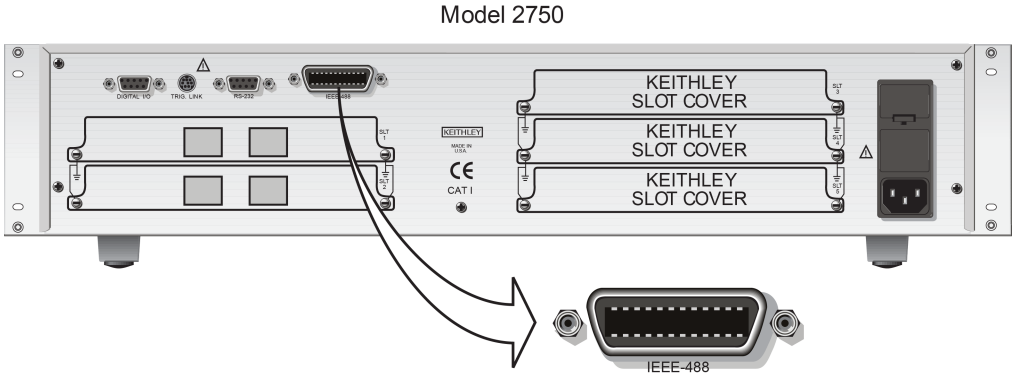
- The number of devices multiplied by 2 m (6.5 ft)
- 20 m (65.6 ft)

You may see erratic bus operation if you ignore these limits.

To connect the 2750 to the IEEE-488 bus:

1. Line up the cable connector with the connector on the rear panel. The connector is designed so that it fits only one way. The following figure shows the location of the IEEE-488 connector on the instrument.

Figure 3: IEEE-488 connector location on the 2750 rear panel



2. Tighten the screws securely, but do not overtighten them.
3. Add additional connectors from other instruments as required.
4. Make certain that the other end of the cable is properly connected to the controller. Most controllers are equipped with an IEEE-488 style connector, but a few may require a different type of connecting cable. Consult the instruction manual for your controller for the proper connecting method.

Select GPIB and set the primary address

The default GPIB address is 26. You can set the address from 1 to 30 if it is unique in the system. This address cannot conflict with an address that is assigned to another instrument or to the GPIB controller.

NOTE

GPIB controllers are usually set to 0 or 21. To be safe, do not configure any instrument to have an address of 21.

The instrument saves the address in nonvolatile memory. It does not change when you send a reset command or when you turn the power off and on again.

NOTE

Selecting GPIB disables the RS-232 interface. Disabling GPIB enables the RS-232.

To select the GPIB and set the primary address:

1. Press the SHIFT key and then the GPIB key. The GPIB ON or GPIB OFF message is displayed.
2. If the GPIB is already ON, press ENTER and proceed to step 3. Otherwise, press the ► key to place the cursor on OFF, press the ▲ or ▼ key to display the ON state, and then press ENTER.
3. Keep or change the presently displayed address (ADDR) value:
   a. **To keep the address:** Press ENTER.
   b. **To change the address:** Press the ► key to place the cursor on the address value, use the ◄, ►, ▲, and ▼ keys to display the new address value, and then press ENTER.

IEEE-488 front-panel operation

The following paragraphs discuss aspects of the front panel that are part of IEEE-488 (GPIB) operation, including messages, status indicators, and the LOCAL key.

IEEE-488 status indicators

The REM (remote), TALK (talk), LSTN (listen), and SRQ (service request) annunciators show the GPIB bus status.

The REM indicator is on when the instrument is in the remote state. The instrument must be addressed to listen with `REM` true before the REM indicator turns on. When the instrument is in the remote state, all front-panel keys except for the LOCAL key are locked out. When REM is turned off, the instrument is in the local state, and front-panel operation is restored.

The TALK indicator is on when the instrument is in the talker active state. To place the instrument in the talk state, address it to talk with the correct `MTA` (My Talk Address) command. TALK is off when the instrument is in the talker idle state. To place the instrument in the talker idle state, send an `UNT` (Untalk) command, address it to listen, or send the `IFC` (Interface Clear) command.

The LSTN indicator is on when the 2750 is in the listener active state, which is activated by addressing the instrument to listen with the correct `MLA` (My Listen Address) command. LSTN is off when the instrument is in the listener idle state. To place the instrument in the listener idle state, send `UNL` (Unlisten), address it to talk, or send the `IFC` (Interface Clear) command over the bus.

When the SRQ indicator is on, a service request was generated. You can program the instrument to generate a service request (SRQ) when one or more errors or conditions occur. This indicator stays on until the serial poll byte is read or all the conditions that caused the SRQ have ended.

If the REM indicator is on and LLO is in effect, LOCAL is locked out. If `TRIGger:SOURce` is set to manual, the TRIG key is active in remote.

LOCAL key

The LOCAL key cancels the remote state and restores local operation of the instrument.

Pressing LOCAL also turns off the REM indicator and returns the display to normal if a user-defined message was displayed.

The LOCAL key is inoperative if the LLO (Local Lockout) command is in effect.

General bus commands

General commands are commands that have the same general meaning, regardless of the instrument. The following table lists the general bus commands.

Command	Effect on 2750
`DCL`	Device clear. Returns the 2750 and all devices on the GPIB to known conditions. See DCL (on page 2-8) for details.
`GET`	Group execute trigger. Initiates a trigger. See GET for details.
`GTL`	Go to local. Cancel remote; restore 2750 front-panel operation. See GTL (on page 2-8) for details.
`IFC`	Interface clear. Goes into talker and listener idle states. See IFC for details.
`LLO`	Local lockout. LOCAL key locked out. See LLO (on page 2-9) for details.
`REN`	Remote enable. Goes into remote operation when next addressed to listen. See REN for details.
`SDC`	Selective device clear. Returns the 2750 to known conditions. See SDC for details.
`SPE, SPD`	Serial polling. Serial polls the 2750. See SPE, SPD (on page 2-9) for details.

DCL

Use the device clear (DCL) command to clear the GPIB interface and return it to a known state. The DCL command is not an addressed command, so all instruments equipped to implement DCL are returned to a known state simultaneously.

When the 2750 receives a DCL command, it:

- Clears the input buffer, output queue, and command queue
- Cancels deferred commands
- Clears any command that prevents the processing of any other device command

The DCL command does not affect instrument settings and stored data.

GET

GET is a GPIB trigger that is used as a trigger event to control operation. The 2750 reacts to this trigger if it is the programmed trigger control source. The following command selects the GPIB trigger control source:

```
TRIGger:SOURce  BUS
```

When a GPIB trigger is sent to the 2750, operation continues in the trigger model.

GTL

Use the go to local (GTL) command to put an instrument that is in remote mode instrument into local mode. Leaving the remote state also restores operation of all front-panel controls.

IFC

The controller sends the interface clear (IFC) command to place the 2750 in the talker idle state and the listener idle state. The instrument responds to the IFC command by canceling illumination of the front-panel TALK or LSTN lights if the instrument was previously placed in one of these states.

Transfer of command messages to the instrument and transfer of response messages from the instrument are not interrupted by the IFC command. If transfer of a response message from the instrument was suspended by IFC, transfer of the message resumes when the instrument is addressed to talk. If transfer of a command message to the instrument was suspended by the IFC command, the rest of the message can be sent when the instrument is addressed to listen.

This command does not affect the status of the instrument. Settings, data, and event registers are not changed.

To send the IFC command, the controller needs to set the IFC line true for a minimum of 100 µs.

With auto output off enabled (`SOURce1:CLEar:AUTO ON`), the output remains on if operation is terminated before the output has a chance to automatically turn off.

LLO

The LLO command prevents local operation of the instrument. After the instrument receives LLO, all its front-panel controls except the POWER switch are inoperative. In this state, pressing LOCAL does not restore control to the front panel. You must use the GTL command to restore control to the front panel. Cycling power will also cancel local lockout.

REN

The remote enable (REN) command is sent to the 2750 by the controller to set up the instrument for remote operation. Generally, place the instrument in the remote mode before you attempt to program it over the bus. Setting REN to true does not place the instrument in the remote state. You must address the instrument to listen after setting REN to true before it goes into remote operation.

The 2750 must be in remote to use the following commands to trigger and acquire readings:

- `INITiate` and then `FETCh?`
- `READ?`
- `MEASure?`

SDC

The selective device clear (SDC) command is an addressed command that performs essentially the same function as the device clear (DCL) command. However, because each device must be individually addressed, the SDC command provides a method to clear only selected instruments, instead of clearing all instruments simultaneously with the DCL command.

SPE, SPD

When the instrument detects the serial polling enable (SPE) and serial polling disable (SPD) events, it sends the status byte of the instrument. This contains the serial poll byte of the instrument.

The serial poll byte contains information about internal functions. Generally, the serial polling sequence is used by the controller to determine which of several instruments has requested service with the SRQ line.

The serial polling sequence may be performed at any time to obtain the status byte from the 2750. Refer to the Status model (on page 11-1) for details.

Programming syntax

The Standard Commands for Programmable Instruments (SCPI) standard is a syntax and set of commands that are used to control test and measurement devices. For more information, see the IEEE-488.2 and SCPI standards.

The following information describes some basic SCPI command information and how SCPI is used with the 2750 and presented in the 2750 documentation.

Commands that are listed as SCPI compliant have commands and parameters that are SCPI confirmed. Other commands are SCPI commands, but do not conform to the SCPI standard set of commands. It is not a recognized command by the SCPI consortium. SCPI confirmed commands that use one or more non-SCPI parameters are explained in the “Details” section of the command description.

Command words

Program messages are made up of one or more command words.

Some command words are enclosed in brackets ([ ]). These brackets denote an optional command word that does not need to be included in the program message. For example: `:INITiate[:IMMediate]`

These brackets indicate that `:IMMediate` is an optional command word and does not have to be used. When using optional command words in your program, do not include the brackets. For example, the above command can be sent in either of these ways:

```
:INITiate
```

```
:INITiate:IMMediate
```

Commands and command parameters

Common commands and SCPI commands may or may not use a parameter. Parameters are shown in angle brackets (< >). The following are some examples:

`*SAV <NRf>`	Parameter (`NRf`) required.
`*RST`	No parameter used.
`:INITiate:CONTinuous <b>`	Parameter <b> required.
`:SYSTem:PRESet`	No parameter used.

Put at least one space between the command word and the parameter. Do not include the angle brackets when sending the command.

Parameter types

The following table lists the common parameter types.

`<b>`	Boolean. Enables or disables an instrument operation. `0` or `OFF` disables the operation, and `1` or `ON` enables the operation. For example, the following command enables continuous initiation: `:INITiate:CONTinuous ON`
`<name>`	Name parameter. Select a parameter name from a listed group. For example, to select the percentage calculation: `:CALCulate:FORMat PERCent`
`<NRf>`	Numeric representation format. A number that can be expressed as an integer (such as 8), a real number (such as 23.6), or an exponent (2.3e6). For example, to set the buffer size to 20: `:TRACe:POINts 20`

`<n>`	Numeric value. Can consist of an NRf number or one of the following name parameters: `DEFault`, `MINimum`, or `MAXimum`. When the `DEFault` parameter is used, the instrument is programmed to the `*RST` default value. When the `MINimum` parameter is used, the instrument is programmed to the lowest allowable value. When the `MAXimum` parameter is used, the instrument is programmed to the largest allowable value.
	For example, to set the timer to 100 ms:
	`:TRIGger:TIMer 0.1`
	To set the timer to the default of 0.1 s:
	`:TRIGger:TIMer DEFault`
	To set the timer to the minimum of 1 ms:
	`:TRIGger:TIMer MINimum`
	To set the timer to the maximum of 999999.999 s:
	`:TRIGger:TIMer MAXimum`

`<list>`	Specify one or more switching channels. For example, to specify a scan list of channels 1 to 5, send: `:ROUTe:SCAN (@1:5)`

Long-form and short-form versions

This documentation shows SCPI commands with both uppercase and lowercase letters. The uppercase letters are the required elements of a command. The lowercase letters are optional. If you choose to include the letters that are shown in lowercase letters, you must include all of them.

When you send a command to the instrument, letter case is not important. You can mix uppercase and lowercase letters in program messages.

For example, you can send the command `:SYSTem:PRESet` in any of the following formats:

```
:SYSTem:PRESet
:SYST:PRES
:SYSTem:PRES
:syst:pres
```

NOTE

For fastest response to commands, always use short forms.

Case sensitivity

Common commands and SCPI commands are not case sensitive. You can use upper or lower case and any case combination. For example:

```
*RST = *rst
DATA? = data?
SYSTem:PRESet = system:preset
```

NOTE

Using all upper case results in slightly faster command response times.

Query commands

SCPI queries have a question mark (?) after the command. You can use the query to determine the present value of the parameters of the command or to get information from the instrument.

For example, to determine what the present setting for the timer is, you can send:

```
:TRIGger:TIMer?
```

This query returns the present setting.

If the command has `MINimum`, `MAXimum`, and `DEFault` options, you can use the query command to determine what the minimum, maximum, and default values are. In these queries, the ? is placed before the `MINimum`, `MAXimum`, or `DEFault` parameter. For example, to determine the default value for the timer, you can send:

```
:TRIGger:TIMer? DEFault
```

If you send two query commands without reading the response from the first, and then attempt to read the second response, you may receive some data from the first response followed by the complete second response. To avoid this, do not send a query command without reading the response. When you cannot avoid this situation, send a device clear before sending the second query command.

Program messages

A program message consists of one or more command words sent by the computer to the instrument.

Each common command is a three-letter acronym preceded by an asterisk (`*`). Common commands are described in Common commands (on page 10-1).

SCPI commands consist of several levels. The following discussion uses the `:STATus` subsystem to explain how command words are structured to create program messages.

Command structure

`:STATus`	Path (root)
`:OPERation`	Path
`:ENABle <NRf>`	Command and parameter
`:ENABle?`	Query command
`:PRESet`	Command

Single command messages

The `:STATus` command structure has three levels. The first level is made up of the root command (`:STATus`) and serves as a path. The second level is made up of another path (`:OPERation`) and a command (`:PRESet`). The third level is made up of one command for the `:OPERation` path. You can execute these commands by sending the following program messages:

```
:STAT:OPER:ENAB  1
:STAT:OPER:ENAB?
:STAT:PRES
```

In each of these program messages, the path pointer starts at the root command (`:STAT`) and moves down the command levels until the command is executed.

Multiple command messages

You can send multiple command messages in the same program message if they are separated by semicolons (;). The following is an example showing two commands in one program message:

```
:STAT:OPER; :STAT:OPER:ENAB 1
```

When this command is sent, the first command word is recognized as the root command (`:STAT`). When the next colon is detected, the path pointer moves to the next command level and executes the command. When the path pointer sees the colon after the semicolon (`;`), it resets to the root level.

Commands that are on the same command level can be executed without having to retype the entire command path. For example:

```
:STAT:OPER:ENAB 1; ENAB?
```

After the first command (`:ENAB`) is executed, the path pointer is at the third command level in the structure.

Command path rules

- Each new program message must begin with the root command unless it is optional, such as `[:SENSe]`. If the root is optional, treat the command word on the next level as the root.
- The colon (`:`) at the beginning of a program message is optional.
- The path pointer can only move down. It cannot be moved up a level. Executing a command at a higher level requires that you start over at the root command.

Using common commands and SCPI commands in the same message

You can use common commands and SCPI commands in the same message if they are separated by semicolons (`;`). A common command can be executed at any command level and does not affect the path pointer.

```
:STAT:OPER:ENAB 1; *ESE 1
```

Program message terminator (PMT)

Each program message must be terminated with a line feed (LF), end or identify (EOI), or LF+EOI. The bus hangs if your computer does not provide this termination. The following example shows how a program message must be terminated:

```
:ROUT:SCAN  (@1:5)  <PMT>
```

Command execution rules

- Commands execute in the order that they are presented in the program message.
- An invalid command generates an error and is not executed.
- Valid commands that precede an invalid command in a multiple command program message are executed.
- Valid commands that follow an invalid command in a multiple command program message are ignored.

Sending strings

If you are sending a string, it must begin and end with matching quotes (either single quotes or double quotes). To include a quote character as part of the string, type it twice with no characters in between.

Response messages

A response message is the message sent by the instrument to the computer in response to a query command program message.

After sending a query command, the response message is placed in the Output Queue. When the 2750 is addressed to talk, the response message is sent from the Output Queue to the computer.

Each response is terminated with a line feed (LF) and end or identify (EOI). The following example shows how a multiple response message is terminated:

```
0; 1; 1; 0;  <RMT>
```

Multiple response messages

If you send more than one query command in the same program message, the response messages for all the queries are sent to the computer when the 2750 is addressed to talk. The responses are sent in the order that the query commands were sent and are separated by semicolons (;). Items in the same query are separated by commas (,). The following example shows the response message for a program message that contains four single item query commands:

```
0; 1; 1; 0
```

Message exchange protocol

These rules summarize the message exchange protocol:

1. Always tell the 2750 what to send to the computer. To send information from the instrument to the computer:
   a. Send the appropriate query commands in a program message.
   b. Address the 2750 to talk.
2. The complete response message must be received by the computer before another program message can be sent to the 2750.

RS-232 serial interface

The 2750 has a built-in RS-232 serial interface. Over this interface, you can send program messages to the instrument and receive response messages from the instrument. You can also place the instrument in the talk-only mode, which allows you to send readings to an external listening device, such as a serial printer.

The serial port of the 2750 can be connected to the serial port of a computer for send/receive operation or to a listener (such as a serial printer) for talk-only operation.

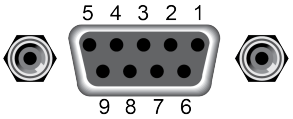
You can use the SCPI programming language over the RS-232 serial interface to communicate with the 2750.

RS-232 connections

The RS-232 serial port is connected to the serial port of a computer using a straight-through RS-232 cable terminated with DB-9 connectors. Do not use a null modem cable**.** If your computer uses a DB-25 connector for the RS-232 interface, you will need a cable or adapter with a DB-25 connector on one end and a DB-9 connector on the other, wired straight through (not null modem).

The serial port uses the transmit (TXD), receive (RXD), and signal ground (GND) lines of the RS-232 standard. It does not use the hardware handshaking lines, CTS and RTS. The following figure shows the rear panel connector for the RS-232 interface.

Figure 4: RS-232 interface connector



The following table shows the pinouts for the connector.

RS-232 connector pinouts

Pin number	Description
1	No connection
2	TXD, transmit data
3	RXD, receive data
4	No connection
5	GND, signal ground
6	Not used
7	Not used
8	Not used
9	No connection

The following table provides pinout identification for the 9-pin (DB-9) or 25-pin (DB-25) serial port connector on the computer.

Computer serial port pinouts

Signal	DB-9 pin number	DB-25 pin number
DCD, data carrier detect	1	8
RXD, receive data	2	3
TXD, transmit data	3	2
DTR, data terminal ready	4	20
GND, signal ground	5	7
DSR, data set ready	6	6
RTS, request to send	7	4
CTS, clear to send	8	5
RI, ring indicator	9	22

Select and configure the RS-232 interface

The following steps describe how to select RS-232 interface and set the baud rate, flow control, and terminator.

NOTE

Enabling the RS-232 interface disables the GPIB. Disabling the RS-232 interface enables the GPIB.

To select and configure the RS-232 interface:

1. Press the **SHIFT** key and then the **RS-232** key. The `RS 232 ON` or `RS 232 OFF` message is displayed.
2. If the RS-232 is already ON, press **ENTER** and proceed to step 3. Otherwise, press the ► key to place the cursor on `OFF`, press the ▲ or ▼ key to display the `ON` state, and then press **ENTER**.
3. To retain the presently displayed BAUD rate, press **ENTER** and proceed to step 4. Otherwise, press the 4 key to place the cursor on the baud rate value, use the ▲ or ▼ key to display the baud rate, and then press **ENTER**.
4. To retain the presently displayed FLOW control, press **ENTER** and proceed to step 5. Otherwise, press the ► key to place the cursor on the flow control setting, use the ▲ or ▼ key to display the flow control setting, and then press **ENTER**.
5. To retain the presently displayed terminator (`Tx TERM`), press **ENTER**. Otherwise, press the ► key to place the cursor on the terminator setting, use the▲ or ▼ key to display the terminator, and then press **ENTER**.

Sending and receiving data

The RS-232 interface transfers data using eight data bits, one stop bit, and no parity. Make sure the controller you connect to the multimeter also uses these settings.

You can break data transmissions by sending a `^C` (decimal 3) or `^X` (decimal 18) character string to the instrument. This clears any pending operation and discards any pending output.

You can break an RS-232 transmission of buffer readings by pressing LOCAL and then EXIT. The next command to send buffer data, such as `TRACe:DATA?`, will start at the beginning, rather than where the transmission was halted.

Baud rate

The baud rate is the rate at which the 2750 multimeter and the programming terminal communicate. Choose one these available rates:

- 19.2k
- 9600
- 4800
- 2400
- 1200
- 600
- 300

The factory selected baud rate is 4800.

When you choose a baud rate, make sure that the programming terminal that you are connecting to the 2750 can support the baud rate you selected. Both the multimeter and the other device must be configured for the same baud rate.

Signal handshaking (flow control)

Signal handshaking between the controller and the instrument allows the two devices to indicate being ready or not ready to receive data. The 2750 does not support hardware flow control.

Software flow control is in the form of `X_ON` and `X_OFF` characters and is enabled when `XonXoFF` is selected from the RS232 FLOW menu. When the input queue of the 2750 becomes more than 3/4 full, the instrument issues an `X_OFF` command. The control program responds to this and stops sending characters until the 2750 issues the `X_ON`, which it does when its input buffer has dropped below half-full.

The 2750 recognizes X_ON and X_OFF sent from the controller. An X_OFF causes the 2750 to stop outputting characters until it sees an X_ON. Incoming commands are processed after the <CR> character is received from the controller.

If NONE is the selected flow control, there is no signal handshaking between the controller and the 2750. Data will be lost if it is transmitted before the receiving device is ready.

NOTE

For RS-232 operation, *OPC or *OPC? should be used with slow responding commands. Refer to *OPC (on page 10-6) for additional detail.

XonXoFF is the factory and *RST default flow control setting.

NOTE

Even with XonXoFF selected, the computer may lose data from the 2750 if the return string is very large (approximately 30,000 or more characters), and one of the higher baud rates is selected. With no flow control (NONE selected), the error occurs with a much smaller return string. Your program could provide some type of error checking for these situations. You can also use the TRACe:DATA:SELected? (on page 9-201) command to return small portions of a large buffer.

Terminator

The 2750 can be configured to terminate each program message that it transmits to the controller with any of the following combinations of <CR> and <LF>.

- <CR>: Carriage return
- <CR+LF>: Carriage return and line feed
- <LF>: Line feed
- <LF+CR>: Line feed and carriage return

Error messages

Refer to the *Model 2750 User's Manual* for RS-232 error messages (+800 through +808).

Section 3

Switching and scanning

In this section:

Switching module operation 3-1
Plug-in switching modules..... 3-2
Pseudocards 3-3
Channel assignments..... 3-4
Identify installed modules..... 3-5
Closing and opening channels using system channel operation3-9
Remote programming — system channel control commands 3-12
Non-amp and non-measure switching modules 3-13
Multiple channel operation 3-15
Scanning fundamentals..... 3-25
Scan configuration 3-34
Scan operation..... 3-46
Scanning using remote programming 3-51
Scanning examples..... 3-52

Switching module operation

This section includes descriptions of the following:

- **Pseudocards:** Describes how to use remote programming to assign a pseudocard to an empty switching module slot.
- **Channel assignments:** Explains the format for specifying the mainframe channel assignment, which is made up of the slot number and switching module channel number.
- **Identifying installed modules and viewing closed channels:** Explains how to remotely identify installed modules and summarizes other query commands that can be used to acquire information about the installed modules.
- **System channel operation:** Provides information for using system channel operation using remote commands.
- **Multiple channel operation:** Provides detailed information for using multiple channel operation. Due to safety considerations, this operating mode should only be used by experienced test engineers.
- **Scanning fundamentals:** Explains channel assignments (slot/channel programming format), the difference between sequential and non-sequential scans, and the basic scan process. Block diagrams (known as trigger models) are provided to help explain the STEP and SCAN operations.

- **Scan configuration:** Provides the step-by-step procedures to configure a simple scan or an advanced scan. Describes other scan options, including delay, monitor, autochannel configuration, saving setups, and autoscan.
- **Scan operation:** Provides the step-by step procedures to perform a basic scan, a manual/external trigger scan, and a monitor scan.

For front-panel operation of switching module channels, refer to the *Model 2750 User's Manual*.

Plug-in switching modules

You can install up to five Keithley 77XX series switching modules in the 2750. A side-by-side comparison of the switching modules is provided in the following tables. Connection information is provided in the documentation for the switching module. Documentation is available at tek.com/keithley.

Model	**7700**	**7701**	**7702**	**7703**	**7705**
2-pole operation	20 channels	32 channels	40 channels	32 channels	Not applicable
4-pole operation	10 channel pairs	16 channel pairs	20 channel pairs	16 channel pairs	Not applicable
1-pole operation	Not applicable	Not applicable	Not applicable	Not applicable	40 channels
Measure volts	300 V maximum	150 V maximum	300 V maximum	300 V maximum	300 V maximum
Measure amps	Ch 21 and 22, 3 A maximum	No	Ch 41 and 42, 3 A maximum	No	No
Measure ohms	2-wire or 4-wire	2-wire or 4-wire	2-wire or 4-wire	2-wire or 4-wire	No
Cold junction for thermocouples	Yes	No	No	No	No
Relay type[1]	Latching electro-mechanical	Latching electro-mechanical	Latching electro-mechanical	Non-latching reed	Latching electro-mechanical
Connector type	Oversized screw terminals	1 female DB-50 1 female DB-25	Oversized screw terminals	2 female DB-50s	2 female DB-50s
Configuration[2]	Multiplexer	Multiplexer	Multiplexer	Multiplexer	Independent SPST channels
Unique features	All DMM functions	All DMM functions except amps	All DMM functions	All DMM functions except amps	Multiple channel operation only

[1] Latching relays hold their open/close state after the 2750 is turned off. When turned on, all relays open after a few seconds.
[2] All multiplexers can be configured as two independent multiplexers.

Model	7706	7707	7708	7709
2-pole operation	20 channels	10 channels	40 channels	8 channels
4-pole operation	10 channel pairs	5 channel pairs	20 channel pairs	4 channel pairs
1-pole operation	Not applicable	Not applicable	Not applicable	Not applicable
Measure volts	300 V maximum	300 V maximum	300 V maximum	300 V maximum
Measure amps	No	No	No	No
Measure ohms	2-wire or 4-wire	2-wire or 4-wire	2-wire or 4-wire	2-wire or 4-wire
Cold junction for thermocouples	Yes	No	Yes	No
Relay type[1]	Latching electromechanical	Latching electromechanical	Latching electromechanical	Latching electromechanical
Connector type	Mini screw terminals	1 female DB-50 1 female DB-25	Oversized screw terminals	1 female DB-50 1 female DB-25
Configuration[2]	Multiplexer	Multiplexer	Multiplexer	Matrix
Unique features	16 digital outputs, 2 analog outputs, one counter, totalizer	32 digital inputs/output	All DMM functions except amps	6 × 8 matrix; for system channel operation, rows 1 and 2 connect to DMM

[1] Latching relays hold their open/close state after the 2750 is turned off. When turned on, all relays open after a few seconds.
[2] All multiplexers can be configured as two independent multiplexers.

Pseudocards

You can perform most open, close, and scan operations and configure your system without having an actual switching module installed in your instrument. Using the remote interface, you can assign a pseudocard, allowing the instrument to operate as if a switching module were installed. This feature allows you exercise open, close, and scan operations or configure your system without having the actual switching module installed in the unit. There is a pseudocard for every Keithley Model 77XX series switching module.

A pseudocard cannot be configured from the front panel. However, once the remote configuration is complete, you can use the front panel to use the pseudocard. Press the LOCAL key to take the 2750 out of remote and use the front panel to use the pseudocard.

When a pseudocard is installed, the appropriate front-panel slot indicator turns on. When the instrument is turned off, the pseudocard is uninstalled.

Use the following commands to install pseudocards:

```
SYSTem:PCARd1 <name>   ' Install pseudocard in slot 1.
SYSTem:PCARd2 <name>   ' Install pseudocard in slot 2.
SYSTem:PCARd3 <name>   ' Install pseudocard in slot 3.
SYSTem:PCARd4 <name>   ' Install pseudocard in slot 4.
SYSTem:PCARd5 <name>   ' Install pseudocard in slot 5.
```

Where `<name>` = `C7700`, `C7701`, `C7702`, `C7703`, `C7705`, `C7706`, `C7707`, `C7708`, `C7709`, `C7710`, `C7711`, or `C7712`.

The following command sets up the 2750 to operate as if a Model 7700 switching module is installed in slot 2, which must be empty. You cannot assign a pseudocard to a slot that already has a switching module installed in it.

```
SYSTem:PCAR2 C7700  ' Install pseudocard 7700 for slot 2.
```

Refer to :SYSTem:PCARd (on page 9-193) for the command description.

Channel assignments

The 2750 has five slots for switching modules. To control the appropriate switching module, the slot number must be included with the switching module channel number when you specify a channel. The channel assignment is formatted as follows:

SCH

where:

- `S` is the slot number
- `CH` is the channel number

For example:

- `101` = Slot 1, Channel 1
- `210` = Slot 2, Channel 10
- `506` = Slot 5, Channel 6

Each switching module has a certain number of channels. For example, the Model 7700 switching module has 22 channels (1 through 22).

For remote operation, the three-digit channel assignment is included in the channel list parameter for the commands. The channel list is shown as `<clist>` and the format for the channel assignment is `(@SCH)`, where:

- `S` is the slot number (1, 2, 3, 4 or 5)
- `CH` is the switching module channel number (must be 2 digits)

Examples:

- `(@101)` = Slot 1, Channel 1
- `(@101, 203)` = Slot 1, Channel 1 and Slot 2, Channel 3
- `(@101:110)` = Slot 1, Channels 1 through 10

NOTE

The `<clist>` parameter is used to configure one or more channels for a scan. Each channel in the `<clist>` must be set to the function specified by the command. If not, a conflict error (−221) occurs. For example, `VOLTage:AC:DIGits 4.5, (@101)` is only valid if scan channel 101 is set for the ACV function.

Identify installed modules

You can use the CARD menu and remote query commands to identify modules installed in the mainframe.

On power-up, the model numbers of installed switching modules are displayed briefly. When the instrument is in the normal display state, red slot indicators on the right side of the display indicate which slots have a switching module or pseudocard installed.

NOTE

If a Model 7700, 7701, 7702, 7703, 7705, 7708, or 7709 switching module is removed when the 2750 is on, the slot indicator for that slot remains on and the instrument operates as if the module is installed. If a Model 7706 or 7707 is removed when the power is on, error +523, Card hardware error, occurs and the module is removed from the system.

In general, it is not recommended to install or remove switching modules with the power on.

CARD menu

The CARD menu identifies the switching modules installed in the mainframe. You can use this menu for the following operations:

- Configure digital inputs and outputs, and analog outputs for switching modules that have one or more of those capabilities, such as the Models 7706 and 7707.
- View the analog input channels that are presently closed. Also, read digital input and output ports, and analog output values for switching modules that have one or more of those capabilities.

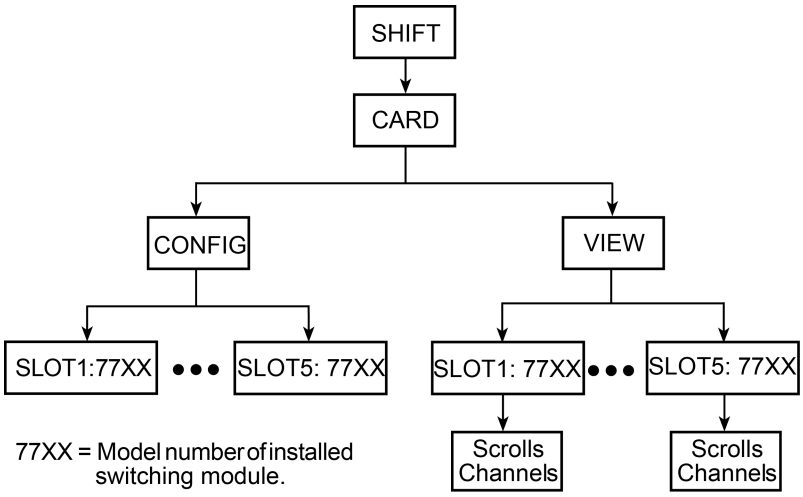
Once in the menu structure, the manual range keys (▲ and ▼**)** and the cursor keys (◄ and ►) display menu items and options and set parameter values. When the item, option, or setting is displayed, press the **ENTER** key to select it. You can cancel a pending selection and exit the menu structure by pressing the **EXIT** key.

To identify installed modules and pseudocards:

1. Press the **SHIFT** key and then the **CARD** key to display the **CARD** menu.
2. Select **CONFIG** or **VIEW** and use the ▲ or ▼ key to check each slot. When the menu structure is open, the slot indicator for the selected slot is on. The other indicators are off.
3. When finished, press **EXIT**.

The Card menu tree is shown in the following figure.

Figure 5: CARD menu tree



The items and options of the menu are defined in the following list.

CARD: CONFIG: This menu item is used to configure switching modules. The channels of the Model 7700 switching module and other similar type modules do not need to be configured.

SLOTX: 77xx: Use to configure the switching module in Slot X (where X = 1, 2, 3, 4, or 5). If configuration is not necessary, the instrument will exit from the menu when ENTER is pressed.

NOTE

For switching modules that require configuration, refer to the documentation for the module.

CARD: VIEW: This menu item is used to view all analog input channels that are presently closed. These include both measurement and non-measurement channels.

The channels are built into a string that scrolls the display. Four dots identify the end of the string. For example, if the Model 7700 is installed in slot 1, the Ω4 function is selected, and system channel 101 is closed, the following string scrolls across the display:

```
101, 111, 123, 124, 125 . . . .
```

Channels 101 and 111 are the paired channels for the 4-wire measurement. Channel 123 is the 4-pole relay setting, and channels 124 and 125 connect input and sense to the DMM of the 2750.

NOTE

Some switching modules have analog outputs, digital inputs, and digital outputs. The values for these channels are also displayed from the VIEW menu item. For details on a switching module, refer to the documentation for the module.

SLOTX: 77xx: Use to scroll the closed channels and channel settings (if applicable) for the switching module in slot *X* (where *X* = 1, 2, 3, 4, or 5).

Scrolling speed: The scrolling speed of the channel string is adjustable or can be paused. The ◄ key slows down scrolling speed and the ► key speeds it up. The ENTER key pauses scrolling. Press ENTER a second time to resume scrolling.

Exiting VIEW: To exit from VIEW, press the EXIT key. Pressing an instrument setting key also exits VIEW, but it also performs the operation associated with the key. For example, pressing Ω2 exits VIEW and selects the Ω2 function.

NOTE

When a command is received while the display is scrolling, the instrument exits from the CARD menu and the command is executed.

Switching module queries (remote operation)

There are commands to identify installed switching modules and closed channels. There are also commands to acquire general information about the installed modules.

The `*OPT?` command returns a list of the switching modules or pseudocards that are installed in the 2750. For additional detail, refer to *OPT? (on page 10-8).

The following commands return a list of closed channels:

- `ROUT:CLOS?`: Returns a list of closed measurement channels, including the paired channel for 4-wire measurements. It does not return non-measurement channels.
- `ROUT:MULT:CLOS?`: Returns all closed channels (measurement and non-measurement).
- `ROUT:MULT:CLOS:STAT?`: Returns the state (open or closed) of each specified channel. A `0` is returned for an open channel, and a `1` is returned for a closed channel.

Refer to ROUTe subsystem (on page 9-69) for detail on these commands.

The `SYSTem:CARD` commands acquire the following information about a switching module installed in the 2750:

- Return the serial number and firmware revision.
- Determine the maximum allowable voltage.
- Determine if the module supports multiplexer or isolated channels.
- Determine if the module has built-in temperature sensors for internal cold junction, thermocouple temperature measurements.
- Determine which channels are used for volts/2-wire measurements and which are used for amps.
- Determine which channels are used for analog or digital output.
- Determine the totalizer channel (Model 7706 only).

Refer to SYSTem subsystem (on page 9-174) for detail on the `SYSTem:CARD` commands.

If there is no card in the specified slot, error −241, Hardware Missing, occurs.

NOTE

The Model 2700 uses a different syntax to specify a slot for the `:CARD` command. Instead of using the `<NRf>` parameter, the slot number is included with the `:CARD` command word, for example, `SYST:CARD1:SNUM?`. To make the 2750 compatible with Model 2700 operation, the `:CARD` command for the 2750 accepts the syntax for slots 1 and 2.

Closing and opening channels using system channel operation

The following discussion assumes a multiplexing switching module, such as a 7700, installed in slot 1 of the mainframe. An alternative to installing a switching module is to use remote programming to assign slot 1 as a pseudocard. The instrument operates as if a switching module is installed in slot 1. To install a 7700 pseudocard in slot 1, send the following command:

```
SYST:PCAR1 C7700
```

NOTE

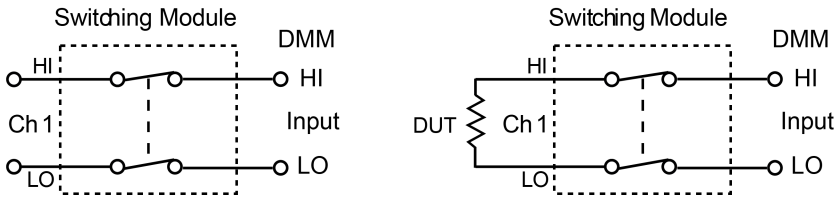
See "Switching module installation and connections" in the *Model 2750 User's Manual* for details on installing a switch module. See “Switch module channel operation” in the *Model 2750 User's Manual* for details on closing and opening switching module channels.

System channel operation is used to connect input channels to the DMM of the 2750.

For a 2-wire function such as DCV, closing a system channel connects the input to DMM Input of the 2750.

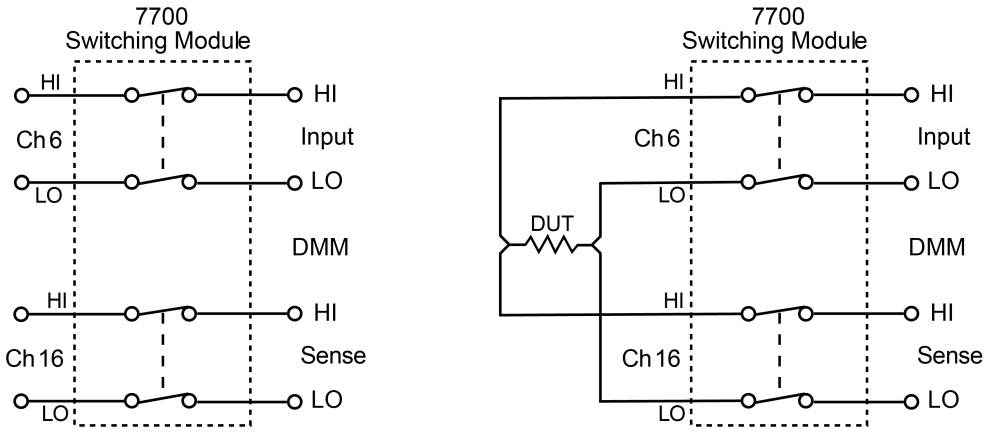
The following figure shows system channel 1 closed. For the Ω2 function, the resistance (DUT) is connected to DMM Input.

Figure 6: Connection to DMM for 2-wire function (system channel 101 closed)



For a 4-wire function such as Ω4, a channel pair is connected to the DMM when a system channel is closed. The system channel is connected to DMM Input and the paired channel is connected to DMM Sense. The following figure shows system channel 6 closed and how the DUT is connected to the DMM for the 4-wire function. For a 4-wire function, the paired channel also closes. For the Model 7700, channels 1 through 10 are paired to channels 11 through 20. When channel 6 is closed, channel 16 also closes.

Figure 7: Connection to DMM for 4-wire function (system channel 106 closed)



NOTE

The figures in this topic show simplified schematics of the switching module. They show that a single switch is closed to connect an input channel to the DMM. Multiple switching is used to make proper connections to the DMM. However, for system channel operation, the user does not need to be concerned about which switches in the module close.

NOTE

Switching module channels can also be controlled using multiple channel operation. This allows individual control of all module channels (switches). Multiple channel operation should only be used by experienced service personnel who recognize the dangers associated with multiple channel closures. Refer to Multiple channel operation (on page 3-15) for details.

Close and open operation

The following points on operation pertain to system channel operation only:

- Only one input channel (or channel pair) is closed at one time. When you close an input channel, the previously closed input channels open.
- When a system channel is closed, the channel number is displayed on the 2750. The slot number for the module is also displayed. For example, `103` indicates that system input channel 3 for a module in slot 1 is closed.
- The paired channel for a 4-wire function is not displayed. Only the system channel number is displayed. For example, in the figure “Connection to DMM for 4-wire function (system channel 106 closed)”, `106` is displayed with the Model 7707 installed in slot 1 of the mainframe.

Switching modules that have current measurement capability have separate channels reserved exclusively for the DCI and ACI functions. For example, the Model 7700 has channels 21 and 22 reserved for amps measurements. With the DCI or ACI function selected, only channels 21 and 22 can be closed. These channels cannot be accessed from any other function.

For remote programming, the following commands are used for basic system operation to open and close input channels:

```
ROUTe:CLOSe <clist>   ' Close specified system channel1.
ROUTe:CLOSe?          ' Query closed system channel2.
ROUT:OPEN:ALL         ' Open all channels.
```

Only one channel can be specified in the `<clist>`. For example, to close input channel 3 for a module in slot 1, the following command would be sent:

```
ROUTe:CLOSe (@103)
```

Only the closed system channel is returned by `ROUTe:CLOSe?`. The paired channel for a 4-wire function is not returned. For example, assume channel 2 in slot 1 is closed. The following response message is returned:

```
(@102)
```

For additional detail on commands for opening and closing channels, refer to ROUTe subsystem (on page 9-69).

Exercise 2 — Closing and opening channels (system channel operation)

The following code demonstrates a sequence to close and open channels of a Model 7700 installed in slot 1 of the mainframe.

This code:

- Opens all channels. It is a good, safe practice to start and end a switching sequence by opening all channels.
- Select the Ω2 function.
- Close system channel 101.
- Close system channel 102.
- Close system channel 106.
- Select Ω4 function.
- Open all channels.

The code example is:

```
ROUT:OPEN:ALL
FUNC 'RES'
ROUT:CLOS (@101)
ROUT:CLOS (@102)
ROUT:CLOS (@106)
FUNC 'FRES'
ROUT:OPEN:ALL
```

Remote programming — system channel control commands

The commands to close and open the system channel are listed in the following table.

When a system channel reading is returned, the system channel number is included in the data string if the CHANnel data element is selected. The FORMat:ELEMents command specifies the data elements to be included in the data string. Refer to :FORMat:ELEMents (on page 9-64) for detail.

The commands to close and open the system channels are listed in the following table.

System channel control commands

Commands	Description
`ROUTe:CLOSe <clist>`	Specify one measurement channel to close.
`ROUTe:CLOSe:STATe? <clist>`	Query closed channels in specified list (1 = closed).
`ROUTe:CLOSe?`	Returns a <clist> of closed measurement channels.
`ROUTe:OPEN:ALL`	Open all channels, and disable ratio and channel average.

Refer to ROUTe subsystem (on page 9-69) for detailed descriptions of the commands.

Non-amp and non-measure switching modules

There are Keithley switching modules that do not support current measurements and there are modules that do not support any measurements.

For switching modules that do not support current measurements, when an amps function is selected (DCI or ACI), you cannot use system channel operation to close channels on that module. Non-amp Keithley modules include the Models 7701, 7703, 7706, 7707, 7708, and 7709. For information the modules, refer to tek.com/keithley.

For modules that do not support any measurements, for front-panel operation, system channel operation cannot be used to close channels. For remote programming, system channel operation can be used, but only the one specified channel closes. All other channels on the module open.

Non-amps switching modules

A non-amp module does not support current measurements. System channel operation cannot be used to close channels when a current function (DCI or ICI) is selected.

If a current function (DCI or ACI) is selected and you attempt to close a system channel, the message `NO AMPS CHAN` is displayed briefly. For remote programming, error −222, Parameter data out of range, is generated.

For example:

```
SYST:PRES              ' Restores system preset defaults.
SENS:FUNC 'CURR:DC'    ' Selects DCI function.
ROUT:CLOS (@101)       ' Attempts to close system channel 101 – Generates
                       ' error -222.
```

If a system channel is already closed and you attempt to select the DCI or ACI function, the message `INVALID FUNC` is displayed briefly. For remote programming, error −221, Settings conflict, is generated.

For example:

```
SYST:PRES              ' Restores system preset defaults.
ROUT:CLOS (@101)       ' Close system channel 101.
SENS:FUNC 'CURR:DC'    ' Attempts to select DCI function – Generates
                       ' error -221.
```

Making amps measurements

To make amps measurements, you must use the front-panel inputs of the 2701 mainframe. You can use the non-amps module for other aspects of the test, but you must use multiple channel operation to close channels.

NOTE

To use the front-panel inputs, make sure the INPUT switch is in the out (F) position.

For example:

```
SYST:PRES              ' Restores system preset defaults.
ROUT:MULT:CLOS (@101) ' Closes channel 101.
SENS:FUNC 'CURR:DC'    '  Selects DCI function – Legal operation.
```

Non-measure switching modules

NOTE

Non-measure Keithley modules include the Models 7705, 7711, and 7712. You can check tek.com/keithley for changes to the available modules.

When using a non-measure module:

- For a non-measure card, no channels are connected to the internal DMM (the channels cannot be connected to the backplane).
- Multiple channel operation should be used to close channels on a non-measure module. For remote operation, use the `ROUT:MULT` commands to close channels.
- Front-panel system (single) channel operation cannot be used to close channels on a non-measure module. For front-panel operation, system channel operation causes message `NO MEAS CARD` to display.
- A non-measure module may have open/close operations that are specific only to that module. Refer to the appropriate module documentation for details on operation.

Multiple channel operation

The capability to individually control channels provides you with added flexibility in how you use a switching module. For example, assume you want to route a signal into channel 1 and out channel 20 of a Model 7700 switching module. You would do this by closing channels 1, 20, and 23. If you open channels 24 and 25, you will isolate the input signal from the DMM of 2750.

Multiple channel operation allows any channel (or channels) in the test system to be closed or opened. It allows more than one measurement channel to be closed at the same time. It also allows individual control of non-measurement channels, such as backplane isolation channels. Multiple channel operation should only be performed by experienced test system engineers.

WARNING

Careless multiple channel operation could create an electric shock hazard that could result in severe injury or death. Improper operation can also cause damage to the switching modules and external circuitry. Multiple channel operation should be restricted to experienced test engineers who recognize the dangers associated with multiple channel closures.

NOTE

Multiple channel operation cannot be used to perform thermocouple temperature measurements using the internal or external reference junction. The simulated reference junction will instead be used and the integrity of the temperature reading will be questionable (ERR annunciator turns on).

Some other key points for multiple channel operation include the following:

- Closing a channel using multiple channel operation has no affect on other closed channels. Whatever channels were previously closed, remain closed.
- A channel closed using multiple channel operation is not displayed on the 2750 and the CHAN annunciator does not turn on.
- Opening a channel using multiple channel operation has no affect on other closed channels. Only the specified channel opens.

NOTE

Use the VIEW option of the CARD menu to display closed channels (see CARD menu (on page 3-5)).

Controlling multiple channels

WARNING

When using multiple channel operation, you must be careful when switching hazardous voltages. If you inadvertently close the wrong channels, you could create a shock hazard and/or cause damage to the equipment.

Most switching modules use latching relays. That is, closed channels remain closed when the 2750 is turned off. Never handle a switching module that is connected to an external source that is turned on. Turn off all power sources before making or breaking connections to the module and before installing or removing the module into or from the 2750.

Avoiding corrupt measurements

Aside from the safety issues, improper use of multiple channel operation can result in corrupt measurements. For example, assume two Model 7700s installed in slots 1 and 2, and a 2-wire function selected. If you use multiple channel operation to close channels 201 and 225, you connect the input at channel 201 to the DMM for measurement.

If you then use system channel operation to close channel 101, channel 125 will also close to connect the input at channel 101 to the DMM. You now have two input channels (101 and 201) connected to DMM Input at the same time, inviting all sorts of problems.

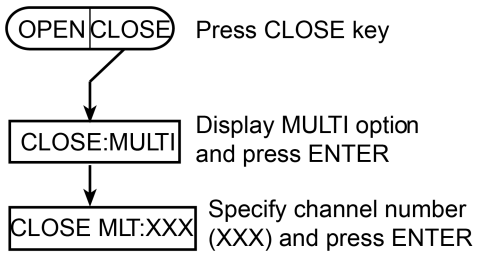
The above problem can be avoided by opening channels 201 and/or 225 before closing channel 101 (and 125) as demonstrated by the following sequence:

1. **Multiple channel operation:** Close channels 201 and 225 for connection to DMM.
2. **Multiple channel operation:** Open channels 201 and/or 225 to disconnect from DMM.
3. **System channel operation:** Close system channel 101 to connect to DMM.

CLOSE key (MULTI menu option)

The MULTI menu option for the CLOSE key can be used to close any individual channel in the mainframe as shown in the following figure.

Figure 8: Multiple channel operation — specifying a channel to close



NOTE

Channels closed by the MULTI option of the CLOSE key are not displayed. Use the VIEW option of the CARD menu to display closed channels (see CARD menu (on page 3-5)).

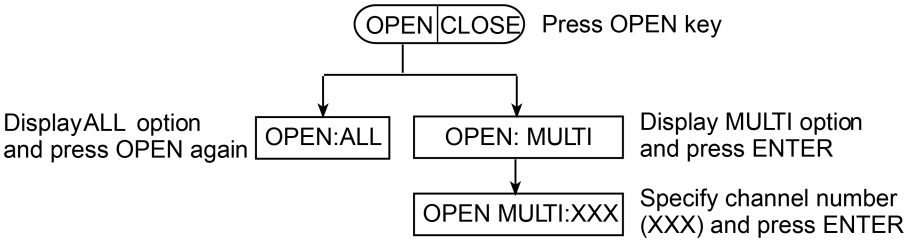
To close a channel:

1. Press the CLOSE key and then use the ▲ or ▼ key to display the "CLOSE:MULTI" message.
2. Press ENTER to display the prompt to close a channel (CLOSE MLT:XXX).
3. Using ▲, ▼, ◄, and ►, key in the three-digit channel you want to select.
4. Press ENTER to close the channel.
5. An invalid channel cannot be closed. The error messages associated with system channel operation also apply to multiple channel operation.

OPEN key

The OPEN key has two options to open channels: ALL and MULTI. The ALL option opens all channels in the mainframe. The MULTI option opens only the specified channel. All other closed channels remain closed. The following figure summarizes OPEN key operation.

Figure 9: Multiple channel operation — opening one or all channels



To open all channels in the instrument:

1. Press the **OPEN** key to display `OPEN: ALL`.
2. Press **OPEN** again (or press **ENTER**) to open all channels.

To open only the specified channel:

1. Press the **OPEN** key. The `OPEN: ALL` message is displayed.
2. Press the ▲ or ▼ key to display the `OPEN: MULTI` message.
3. Press **ENTER** to display the prompt to open a channel (`OPEN MLT:XXX`).
4. Using ▲, ▼, ◄, and ►, key in the three-digit channel you want to select.
5. Press **ENTER** to open the channel.

NOTE

If the channel you open using `OPEN: MULTI` is the system channel (the channel number displayed on the 2750), the channel opens, but the system channel number is still displayed. Refer to Multiple channel operation anomalies (on page 3-19).

Remote programming — Multiple channel control commands

The commands to close and open the multiple channel are listed in the following table.

Commands	Description
`ROUTe:MULTiple:CLOSe <clist>`	Specify one or more channels to close.
`ROUTe:MULTiple:OPEN <clist>`	Open channels specified in list. Unlisted channels not affected.
`ROUTe:OPEN:ALL`	Open all channels.
`ROUTe:MULTiple:CLOSe?`	Returns a list of all closed channels.
`ROUTe:MULTiple:CLOSe:STATe? <clist>`	Query closed channels in specified list (1 = closed).

Refer to ROUTe subsystem (on page 9-69) for detailed descriptions of the commands.

Remote programming example (multiple channel operation)

The following example assumes a Model 7700 installed in slot 1. This command sequence connects channel 101 to channel 111 (through channel 123). Note that these two closed channels will be internally isolated from the DMM since the backplane isolation channels (124 and 125) will be open.

```
ROUT:OPEN:ALL       ' Open all channels.
ROUT:MULT:CLOS (@101,111,123)       ' Close channels 101, 111 and 123.
```

When finished with multiple channel operation it is a good, safe practice to open all channels (`ROUT:OPEN:ALL`).

Multiple channel operation anomalies

Anomaly 1: When you use multiple channel operation to open the system channel, the channel will open but the system channel number will still be displayed on the 2750. For details, see Anomaly 1 example: Wrong channel displayed (on page 3-19).

Anomaly 2: For a 4-wire function, you can use multiple channel operation to open the paired channel. If you then use system channel operation to again select the already closed system channel, it will not reclose the paired channel. For details, see Anomaly 2 example: Opening the paired channel (on page 3-20).

NOTE

The following anomaly examples assume a Model 7700 installed in slot 1.

Anomaly 1 example: Wrong channel displayed

The following example closes channel 102 and connects it to the DMM Input. However, the 2750 does not display the measurement channel that is closed. It displays channel 101 instead of channel 102.

1. Use the ALL option for the OPEN key to open all channels in the mainframe.
   Remote programming: `ROUT:OPEN:ALL`
2. Press the ► key to close (and display) channel 101. This closes channel 101 (which is the system channel) and channel 125 to connect it to the DMM Input.
   Remote programming: `ROUT:CLOS (@101)`
3. Use the MULTI option for the CLOSE key to close channel 102. The system channel is not affected. Channels 101, 102, and 125 are now closed.
   Remote programming: `ROUT:MULT:CLOS (@102)`
4. Use the MULTI option for the OPEN key, open channel 101. Even though channel 101 is still being displayed on the 2750, it is channel 102 that is actually connected to the DMM Input (channels 102 and 125 closed).
   Remote programming: `ROUT:MULT:OPEN (@101)`

To correctly display the channel that is closed (channel 102) repeat step 1 above to open all channels, and then use the ► key or the `ROUT:CLOS (@102)` command to close (and display) channel 102. This closes channel 102 (which is the system channel) and channel 125 to connect it to the DMM Input.

Anomaly 2 example: Opening the paired channel

Assume 4-wire connections to a 1 kΩ resistor using channels 1 and 11 of the Model 7700 switching module. Also assume the Ω4 function is selected. The following procedure demonstrates how careless multiple channel operation can cause an overflow reading even though everything else from the front panel "looks right."

1. Use the ALL option for the OPEN key (OPEN: ALL) to open all channels in the mainframe.
   Remote programming: `ROUT:OPEN:ALL`
2. Press the ► key to close (and display) channel 101. The following channels close:
   - Channel 101 (system channel).
   - Channel 125 (connects channel 101 to DMM Input).
   - Channel 111 (paired channel for 4-wire measurements).
   - Channel 124 (connects channel 111 to DMM Sense).
   - Channel 123 (isolates channel 101 from channel 111).
   - The 2750 displays the 1 kΩ reading for system channel 101.
     Remote programming: `ROUT:CLOS (@101)`
3. Using the MULTI option for the OPEN key, open channel 111. This opens the connection to DMM Sense and causes an `OVRFLW` reading. Channel 101 is still closed and displayed as the system channel.
   Remote programming: `ROUT:MULT:OPEN (@111)`
4. In an attempt to clear the overflow reading problem, use the SINGLE option of the CLOSE key to close channel 101. Since channel 101 is still the system channel, selecting it again in this manner is a "no action." Channel 111 does not close.
   Remote programming: `ROUT:CLOS (@101)`

One way to resolve the above problem is to repeat step 1 to open all channels, and then repeat step 2 to close channel 101. All the listed channels in step 2 close to make the 4-wire connection to the 1 kΩ resistor.

Dual independent multiplexers

Using multiple channel operation, any multiplexer switching module can be configured as two independent multiplexers. For example, the Model 7700 is normally used as a single 1 × 20 multiplexer, but it can also be configured as two 1 × 10 multiplexers.

NOTE

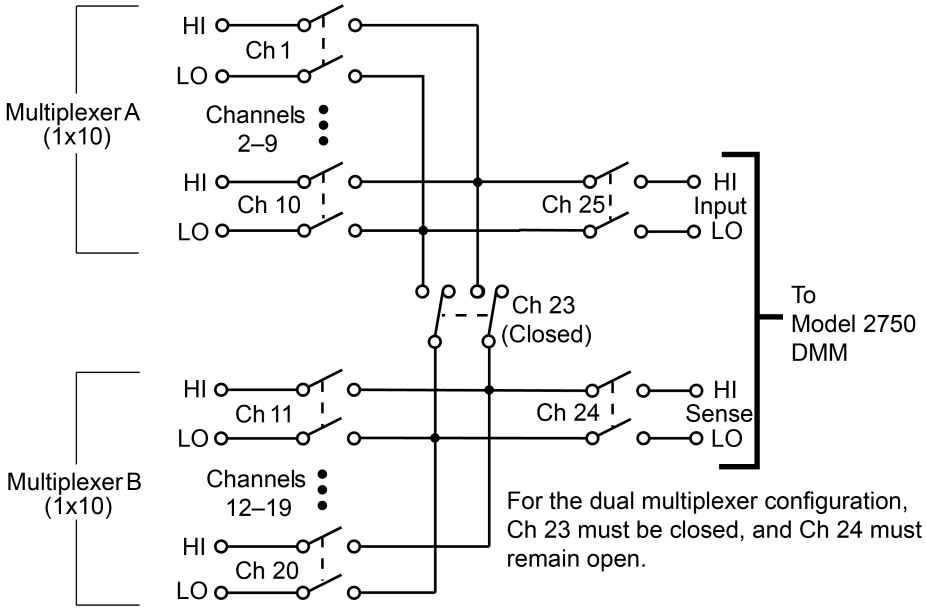
Thermocouple temperature measurements using the internal or external reference junction cannot be performed when using multiple channel operation to connect an input channel to the DMM. The simulated reference junction is used instead, resulting in invalid readings (ERR annunciator turns on).

A multiplexer switching module is configured as two multiplexers by using multiple channel operation to close the 2-pole/4-pole relay. The Model 7700 is configured as two independent multiplexers by closing channel 23. As shown in the following figure, the closed position of channel 23 isolates Multiplexer A (channels 1 through 10) from Multiplexer B (channels 11 through 20).

For the dual multiplexer configuration, only Multiplexer A channels can be internally connected to the DMM of the 2750. For the Model 7700, closing channel 25 allows channels 1 through 10 to be measured by the DMM.

When using the dual multiplexer configuration, the sense backplane isolation relay must be kept open to isolate Multiplexer B channels from the sense terminals of the DMM. For the Model 7700, channel 24 must be kept open, as shown in the following figure.

Figure 10: Dual multiplexer configuration (Model 7700)



Dual multiplexer application

This application demonstrates how to use the Model 7700 as a dual multiplexer to bias and measure 10 DUTs. An external source powers the DUTs while the DMM of the 2750 measures the output of the DUTs. To prevent overloading of the external source, each DUT is powered and measured separately.

The following figure shows the connections for this application. The external source is connected to the Sense terminals of the switching module and the DUTs are connected to channels 1 through 10. Channels 11 through 20 are used to connect external power to each DUT.

For this application, the 2-pole/4-pole relay and backplane isolation relays of the switching module are to be controlled as follows:

- Closing channel 23 isolates the input measurement channels (1 through 10) from the external source channels (11 through 20). It also connects the DUT to the external source. This channel must remain closed while testing DUT.
- Opening channel 24 isolates the external source from the backplane of the 2750. This channel must remain open while testing the DUT.
- Closing channel 25 connects an input channel (1 through 10) to the DMM.

In the following figure, channels 1 and 11 are closed to test DUT 1. A more detailed view of the test for DUT 1 is shown in Test procedure (on page 3-23). The test for the other DUTs is similar except that different source and measure channels are closed. Closed channels for each DUT test are listed in the following table.

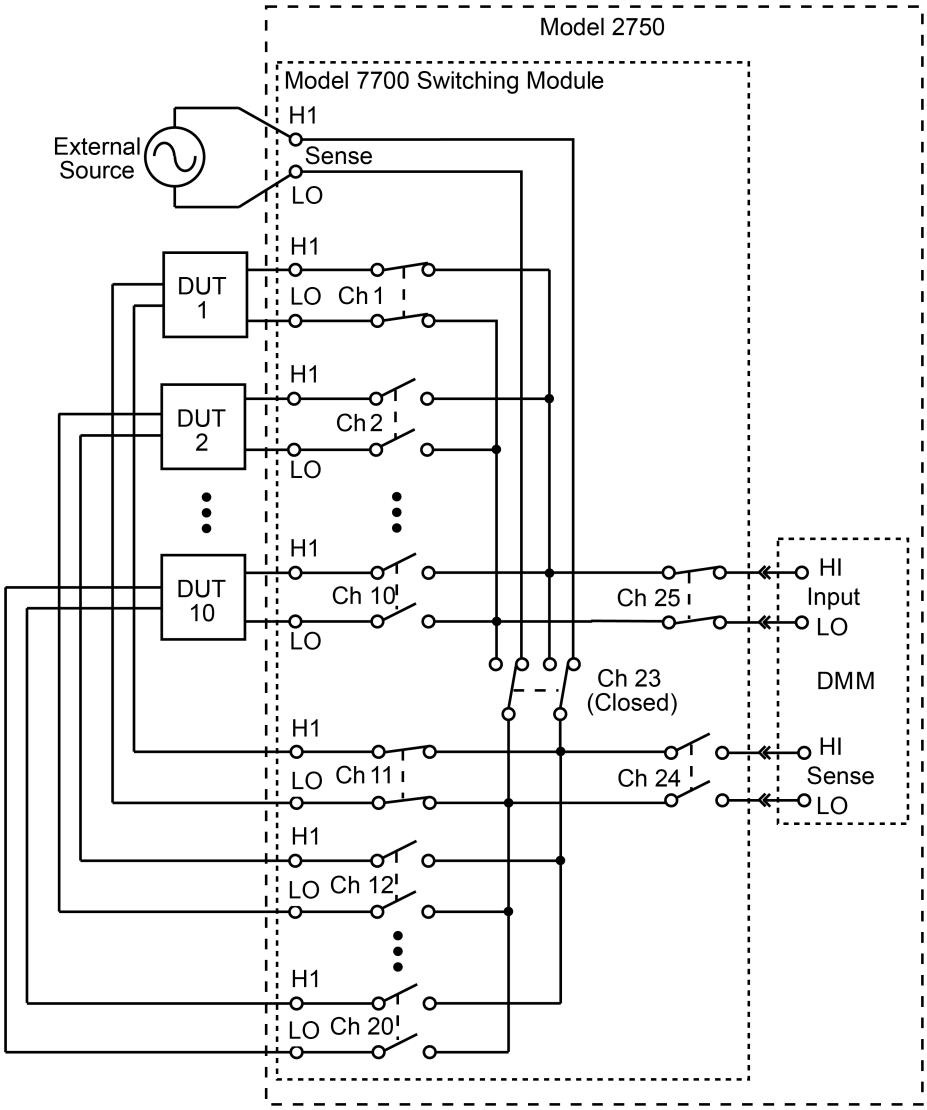
Tested device	Closed channels
DUT 1	1, 11, 23 and 25
DUT 2	2, 12, 23 and 25
DUT 3	3, 13, 23 and 25
DUT 4	4, 14, 23 and 25
DUT 5	5, 15, 23 and 25

Tested device	Closed channels
DUT 6	6, 16, 23 and 25
DUT 7	7, 17, 23 and 25
DUT 8	8, 18, 23 and 25
DUT 9	9, 19, 23 and 25
DUT 10	10, 20, 23 and 25

NOTE

Do not use this application to measure the temperature of the DUT using a thermocouple with the INTernal or EXTernal reference junction selected. The SIMulated reference junction is used, resulting in invalid readings. The ERR annunciator turns on to indicate that the integrity of the temperature reading is questionable.

Figure 11: Dual multiplexer application connections



Test procedure

The following test procedure assumes a Model 7700 switching module installed in slot 1 of the mainframe.

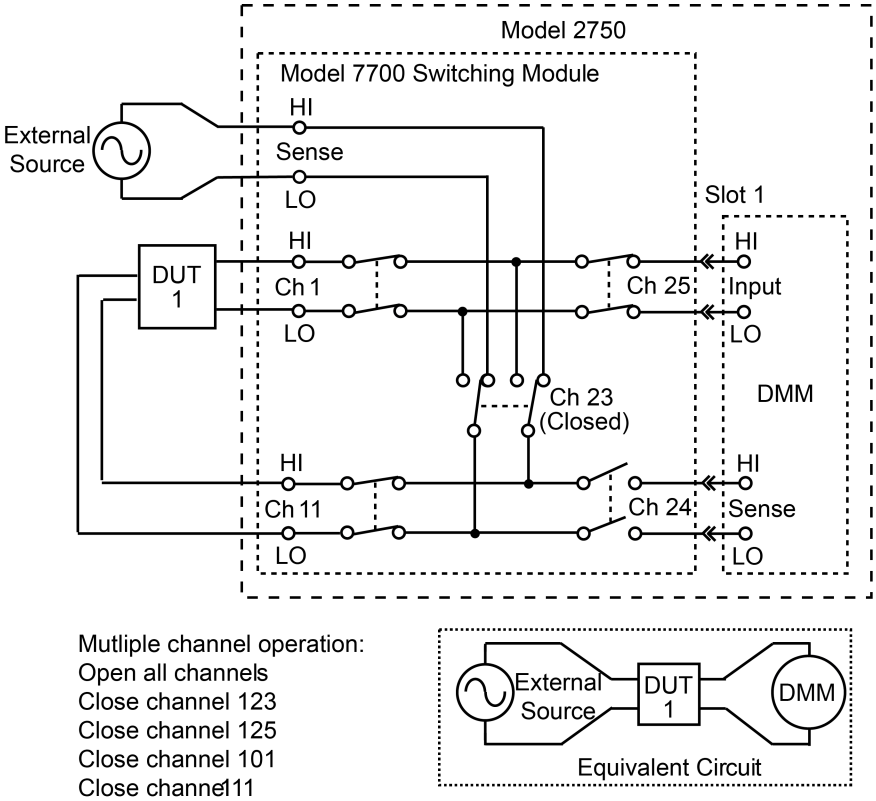
The procedure assumes that the instrument is operating in the continuous measurement (triggering) mode.

Do not use the following procedure to perform thermocouple temperature measurements with the internal or external reference junction selected. The simulated reference junction is used instead, resulting in invalid readings. The ERR annunciator turns on to indicate that the integrity of the temperature reading is questionable.

To run the dual independent multiplexer test procedure:

1. Open all channels. For most switching modules, channels remain closed after the 2750 is turned off. Therefore, it is good safe practice to open all channels at the start and end of the test.
   - Front-panel operation: Press OPEN > Display ALL > Press OPEN
   - Remote programming: `ROUT:OPEN:ALL`
2. Close channels 23 and 25.
   - Front-panel operation: Press CLOSE > Select MULTI > Key in 123 > press ENTER Press CLOSE > Select MULTI > Key in 125 > press ENTER
   - Remote programming: `ROUT:MULT:CLOS (@123,125)`
3. Close channels 1 and 11 to connect DUT 1 to the DMM and bias supply.
   - Front-panel operation: Press CLOSE > Select MULTI > Key in 101 > Press ENTER Press CLOSE > Select MULTI > Key in 111 > Press ENTER
   - Remote programming: `ROUT:MULT:CLOS (@101,111)`
4. Measure DUT 1.
   - Front-panel operation: Take reading from display
   - Remote programming: `DATA?`
5. Open channels 1 and 11.
   - Front-panel operation: Press OPEN > Select MULTI > Key in 101 > Press ENTER Press OPEN > Select MULTI > Key in 111 > Press ENTER
   - Remote programming: `ROUT:MULT:OPEN (@101,111)`
6. Modify steps 3, 4, and 5 to test DUT 2: Close channels 2 and 12, measure DUT 2, and then open channels 2 and 12.
7. Test the remaining eight DUTs in a similar manner: Close the appropriate channels for the DUT, make the measurement, and then open the channels.
8. After the last DUT is tested, open all channels.
   - Front-panel operation: Press OPEN > Display ALL > Press OPEN
   - Remote programming: `ROUT:OPEN:ALL`

Figure 12: Testing DUT 1



Scanning fundamentals

The 2750 can scan the channels of up to five installed Keithley switching modules. Each scan channel can have its own setup. Aspects of operation that can be uniquely set for each channel include function, range, rate, AC bandwidth, relative offset, filter, digits, math, ohms offset compensation, temperature transducers, limits, channel average, channel ratio, and volts dB.

Readings for scanned channels are automatically stored in the buffer. With buffer autoclear enabled (the default), the buffer clears when the scan is started. When disabled, scan readings are appended to the buffer. The `TRACe:CLEar:AUTO` command is used to enable or disable buffer autoclear.

Sequential and non-sequential scans

Only a sequential scan can be configured from the front panel. For a sequential scan, the scan proceeds from the lowest numbered channel to the highest. For example, assume channels 101, 102, 105, 108 and 109 are selected for a scan. The scan will run in this order: 101 > 102 > 105 > 108 > 109.

For remote programming, a non-sequential scan can be configured. Channels are scanned in the order that they are listed in the scan list. This allows backward scanning. For example, assume the following scan list:

(@101, 102, 104, 105, 103, 109)

The above scan will run in this order: 101 > 102 > 104 > 105 > 103 >109. Notice that after channel 105 is scanned, the 2750 returns to scan channel 103. It then proceeds forward to scan channel 109. Any scan that performs backward scanning is considered a non-sequential scan. For more information on non-sequential scanning, refer to :ROUTe:SCAN[:INTernal] (on page 9-78).

NOTE

Non-sequential scanning is only intended to be performed using remote programming. Unexpected results may occur if a non-sequential scan is run from the front panel.

Scan process

For functions that use 2-wire measurements, the basic scan process is as follows:

1. Open any closed channels.
2. Close a channel.
3. Make the measurement.

This process is repeated for each channel in the scan. The last scanned channel opens.

For the Ω4 and 4-wire RTD TEMP functions, which use 4-wire measurements, the scan process uses paired channels. The scan process is as follows:

1. Open any closed channels.
2. Close the paired channels.
3. Make the 4-wire measurement. The last scanned channel pair opens.

NOTE

For the Model 7700 switching module, primary channels 1 through 10 are paired to channels 11 through 20. Channel 1 is paired to channel 11, channel 2 is paired to channel 12, channel 3 is paired to channel 13, and so on.

Ratio and channel average performs measurements on two channels and then calculates and displays the result. Therefore, these 2-channel calculations also use paired channels. The scan process is as follows:

1. Open any closed channels.
2. Close the primary (displayed) channel and make a measurement.
3. Open the primary channel.
4. Close the paired channel and make a measurement.
5. Calculate and display the result.
6. Open the paired channel.

NOTE

When scanning, the displayed channel number, such as 101, is not necessarily the channel that is presently closed. If both a reading and a scan channel are displayed, the reading and annunciators pertain to that channel, but that channel is no longer closed. The next channel in the scan list is the one that is now closed. Therefore, the reading and annunciators pertain to the channel and do not necessarily indicate the present state of the 2750. If the display is blanked (- - - - - - -), the displayed channel is closed and has not been measured.

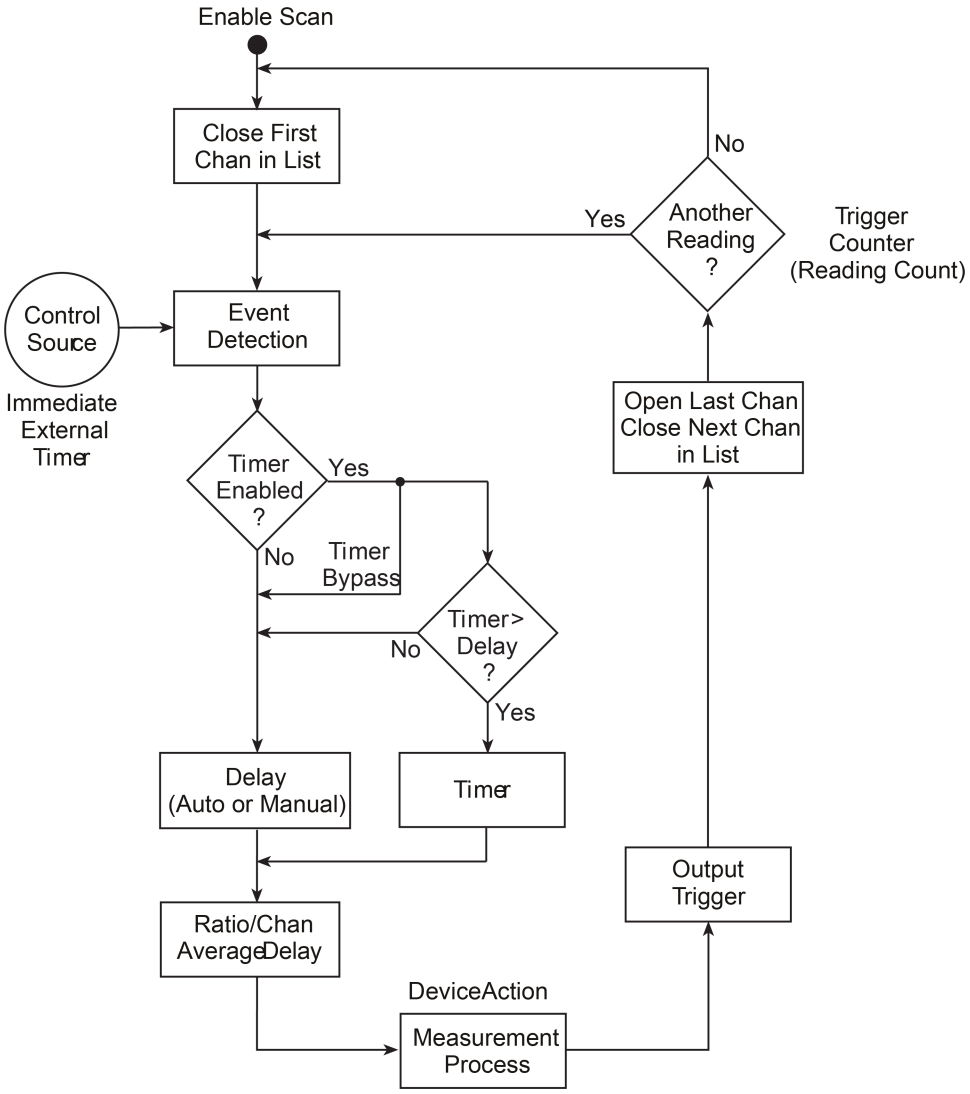
Trigger models when using the front panel

NOTE

The following information on trigger model operations applies specifically to front-panel operation.

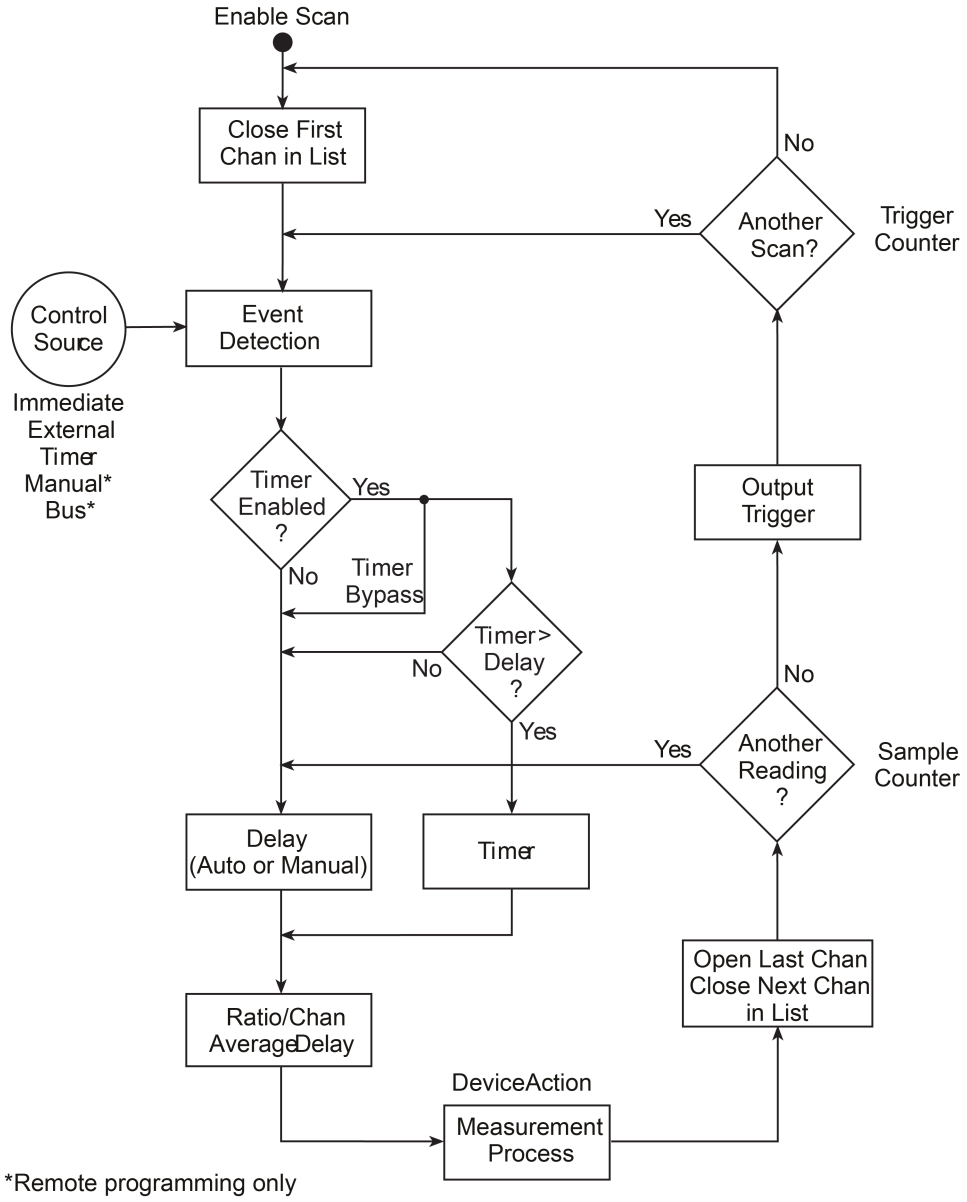
Block diagrams, known as trigger models, are used to show the two fundamental scan functions: STEP or SCAN. These two scan functions are enabled by the STEP and SCAN keys, respectively. The trigger models for scanning are shown in the following figures.

Figure 13: Trigger model with STEP function



The trigger model in the following figure also applies for bus operation. See Scanning using remote programming (on page 3-51) for differences between front-panel and remote scanning.

Figure 14: Trigger model with SCAN function



STEP operation overview: When the STEP key is pressed, the 2750 leaves the idle state, closes the first channel, and waits for the programmed trigger event. After the trigger is detected, the instrument may be subjected to one or more delays before performing the measurement.

After a reading is made and stored in the buffer, the 2750 outputs a trigger pulse, opens the closed channel, and then closes the next channel in the scan list. The instrument then waits for another trigger event to measure the channel. After the last channel is scanned, the instrument returns to the idle state with the first channel in the scan list closed.

SCAN operation overview: When the SCAN key is pressed, the 2750 leaves the idle state, closes the first channel, and waits for the programmed trigger event. After the trigger is detected, the instrument may be subjected to one or more delays before making the measurement.

After a reading is made and stored in the buffer, the 2750 opens the closed channel and then closes the next channel in the scan list. Operation loops to measure all channels in the scan list. After the last channel in the scan list is measured, the 2750 outputs a trigger pulse.

If programmed to again scan the channels in the scan list, the 2750 waits at the control source for another trigger event. After all the scan list channels are again measured, the 2750 outputs another trigger pulse. After all programmed scans are completed, the instrument returns to the idle state with the first channel in the scan list closed.

The individual components of the trigger models are explained in the following topics.

Idle

When a scan is enabled (STEP or SCAN annunciator on), operation goes into the idle state and immediately drops down to the control source. Note that after the last channel in the scan is measured, operation returns to the idle state, where measurements are halted and the first channel in the list is closed.

Control sources

For front-panel operation, the control sources to manage the scan are Immediate, Timer, and External Trigger. Operation is held at the selected control source until the appropriate trigger event is detected.

For step operation, when the trigger event is detected, a channel is measured. The scan pointer then loops to the control source and waits for the next trigger event to occur.

For scan operation, when the trigger event is detected, all the channels in the scan list are scanned. The scan pointer then returns to the control source and waits for the next trigger event to be detected.

Immediate control source

With immediate triggering, event detection is immediate allowing channels to be scanned.

Timer control source

With the timer source enabled, event detection is immediately satisfied. On the initial pass through the loop, the Timer Bypass is enabled, allowing operation to bypass the Timer and continue to the Delay block.

On each subsequent pass through the loop, the Timer Bypass is disabled. Operation is then delayed by the Timer or the Delay. If the user-set Timer interval is greater than the user-set Delay, the Timer controls the length of the delay. Otherwise, the length of the delay is controlled by the user-set Delay period.

The Timer interval can be set from 0 to 999999.999 seconds. The timer resets to its initial state when the scan is completed.

For step operation, the timer control source affects the timing between scanned channels.

For scan operation, the timer control source affects the timing between scans. It has no effect on the timing between scanned channels.

External trigger control source

Pressing the EX TRIG key places the instrument in the external trigger mode (TRIG annunciator on). When the STEP or SCAN key is then pressed, that scan is enabled. However, the scan does not start until an external trigger is received or the TRIG key is pressed. The external trigger or TRIG keypress satisfies event detection.

For step operation, each time an external trigger is received or the TRIG key is pressed, one channel is scanned.

For scan operation, each time an external trigger is received or TRIG key is pressed, one complete scan is performed.

Delays

As shown in the trigger models, operation may be subjected to one or more delays before a channel is measured.

If the timer control source is selected and its user-set interval is greater than the user-set delay, the timer interval supersedes the delay period after the first pass through the loop.

Refer to Delay (auto or manual) (on page 7-2) for detail on delays in the trigger model.

With ratio or channel average enabled, a delay is typically used to keep the channel relays from cycling too fast. The default delay period is 0.5 seconds but can be set from 0 to 999999.999 seconds using remote programming. Refer to Ratio and channel (on page 4-18) for detail.

NOTE

The ratio or channel average delay is in addition to the timer, automatic delay, or manual delay. It occurs after the timer interval or delay period elapses. The delay for ratio and channel average can only be set using remote programming. Refer to :TRIGger[:SEQuence[1]]:DELay (on page 9-209) and :TRIGger[:SEQuence[1]]:DELay:AUTO (on page 9-210) for additional information.

Setting an automatic or manual delay

To set an automatic or manual delay:

1. With the instrument in the normal measurement display state, press **SHIFT** and then **DELAY**.
2. Press ▲ and ▼ to display `AUTO` (auto delay) or `MAN` (manual delay) and press **ENTER**.
3. If you selected MAN, you are prompted to set the delay in the hour/minute/second time format. Use the ◄, ►, ▲, and ▼ keys to set the delay. Pressing the AUTO key sets the manual delay to 0.001 s.
4. With the manual delay displayed, press **ENTER**.

Device action

The channel measurement process is performed at this block. If repeat filter is enabled, the filter process is also performed.

Reading count

For both STEP and SCAN, the reading count specifies the number of readings to store in the buffer.

STEP operation: The reading count specifies the number of channels to scan. This can be equal to, less than, or greater than the number of channels in the scan list. The last scanned channel remains closed. If you start the scan again, it starts at the next channel.

If the reading count is set to infinity (INF), the scan continuously repeats until you stop it.

One counter is used for STEP operation. The reading count sets the Trigger Counter.

SCAN operation: When a scan is started, one or more complete scans are performed. The number of channels in the scan list determines the number of channels for each scan. The reading count determines the number of scans to perform and is best explained by an example. Assume there are 10 channels in the scan list. If you set the reading count to 10 or less, one scan of the 10 channels is performed. If you set the reading count to any value from 11 to 20, two scans are performed. A reading count from 21 to 30 gives you three scans.

If the reading count is set to infinity (INF), the scan continuously repeats until you stop it.

NOTE

As shown in Trigger models when using the front panel (on page 3-27), two counters are used for SCAN operation. The trigger counter controls the number of scans and the sample counter controls the number of channels for each scan. The number of channels in the scan list and the programmed reading count automatically sets the Trigger Counter and the Sample Counter. The Sample Count is equal to the scan list length. For example, if channels 101, 102, and 103 are programmed to be scanned, the sample count is 3.

Output trigger

During step operation, after each channel is scanned, an output trigger is applied to the rear panel Trigger Link connector.

During scan operation, after all channels in the scan list are scanned, an output trigger is applied to the rear panel Trigger Link connector.

Trigger model using remote operations

The trigger model for bus operation is shown in Trigger models when using the front panel (on page 3-27), in the figure "Trigger model with SCAN function." Bus operation is similar to front-panel SCAN operation, with the following significant differences:

- **Idle:** The instrument goes into the idle state (measurements halted) after the last scan channel is measured. For front-panel operation, the instrument stays in idle until the next scan is started. For bus operation, the instrument will not stay in idle unless continuous initiation is disabled. There are two commands to disable continuous initiation:

```
INITitate:CONTinuous OFF   ' Disable continuous initiation.
*RST                       ' Restore *RST defaults.
```

  The instrument remains in idle until it receives an initiate command. Typical commands to initiate one scan cycle include:

```
INITiate   ' Initiate one scan cycle.
READ?      ' Initiate one scan cycle and request sample readings.
```

More information on using these commands is provided in Command reference (on page 9-1).

- **Control sources:** For bus operation, there are two additional control sources: Bus and Manual. For the Bus control source, scan operation is controlled by bus triggers (such as *TRG) or by using the TRIG key. For the Manual control source, event detection is controlled solely by the TRIG key. The instrument must be in local to use the TRIG key. The LOCAL key takes the instrument out of remote.
- **Trigger and sample counters:** For front-panel SCAN operation, the number of channels in the scan list and the programmed reading count automatically sets the trigger and sample counters. For remote operation, these two counters are set by the TRIGGer:COUNt and SAMPle:COUNt commands.

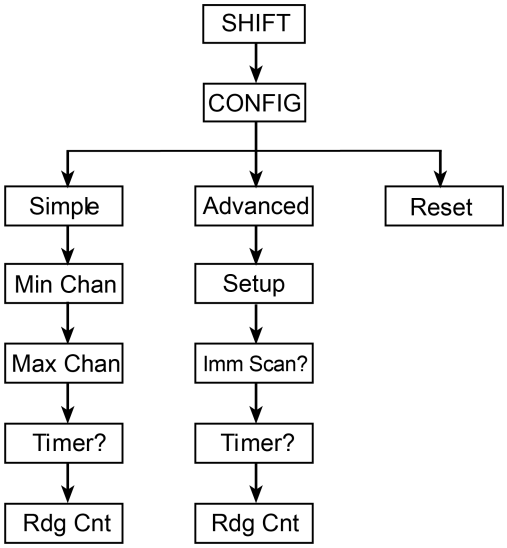
NOTE

To set sample count >1, continuous initiation must be disabled (refer to Idle (on page 3-30)). Only sample count readings are stored in the buffer. Refer to Triggering (on page 7-1) for detailed information on the trigger model.

Scan configuration

A scan is configured from the scan configuration menu, which is accessed by pressing SHIFT and then CONFIG. The following figure shows the basic flowchart to configure a scan. After entering the menu structure, you can configure a simple scan, an advanced scan, or reset the configuration to the default setup for a simple scan. Refer to the following figure for the following discussions.

Figure 15: Scan configuration flowchart



NOTE

Only a sequential scan can be configured from the front panel. For a sequential scan, the scan proceeds from the lowest numbered channel to the highest. Non-sequential scanning is only intended for remote programming. Unexpected results may occur if a non-sequential scan is run from the front panel. For more information, see Sequential and non-sequential scans (on page 3-26).

The instrument is always configured to run a scan. On power-up, each available channel uses the power-on default setup. For example, with factory power-on default settings and two Model 7700s installed, the instrument scans channels 101 through 220 when the scan is run.

There are two scan configurations: Simple and advanced. When you configure the simple scan, the instrument uses the present instrument setup for each channel in the scan. For the advanced scan, each channel can have its own setup.

Channel setup considerations

To use an acquired relative offset value for an advanced scan channel, the relative offset value must be acquired with the instrument in the normal measurement state. To set relative offset values for scan channels, refer to Relative offset (on page 4-12).

The moving filter cannot be used in a scan; only the repeating filter can be used. If you configure channels to use the moving filter, the filter is set to off when the scan is run. Refer to Digital filters (on page 4-9) for details on filtering.

In the scan setup menu, use the edit keys (◄, ►, ▲, and ▼) to make selections and set values. Displayed selections and settings are entered by pressing the ENTER key.

Effects of function changes on the scan list

To avoid problems with scans, it is recommended that the scan list (`ROUT:SCAN`) be created after scan channel functions are selected (`SENS:FUNC`).

Changing from a 2-wire function to a 4-wire function changes the scan list. This is demonstrated in the following example.

The following commands show the proper sequence to configure a simple 20-channel DCV scan using a Model 7700 installed in slot 1:

```
SENS:FUNC 'VOLT',(@101:120)   ' Set channels for DCV.
ROUT:SCAN (@101:120)          ' Specify scan list.
```

When the scan is changed to a 4-wire function, the scan list changes. For example, assume the above scan is changed to the Ω4 function as follows:

```
SENS:FUNC 'FRES',(@101:110)
```

For the 4-wire resistance function, channels 101 through 110 is paired to channels 111 through 120. `ROUT:SCAN?` returns the following scan list:

```
(@101:110)
```

Now assume the scan is returned to DCV function as follows:

```
SENS:FUNC 'VOLT',(@101:120)
```

The above command sets channels 101 through 120 for DCV. However, it does not affect the scan list. `ROUT:SCAN?` still returns a 10-channel scan list:

```
(@101:110)
```

The following command sets the scan list for 20 channels:

```
ROUT:SCAN (@101:120)
```

Saving the configured scan

You can save the configured scan in a user-saved setup. For scan configured from the front panel, the timer state (NO or YES), timer value, and reading count value are also saved. However, if the settings for a user setup or power-on setup do not match the switching module type presently installed when the setting is recalled, error +520, Saved setup scancard mismatch, occurs. The scan resets to the factory default settings, and all channels open. The saved setup is still retained in memory and can be restored when the matching switching module is later installed.

There is an anomaly with the displayed front-panel CONFIG menu when a saved front-panel scan is recalled. The reading count value in the menu may not reflect the actual reading count of the scan. For example, assume a Model 7700 module is configured for a 10-channel scan and a reading count of 30. For this configuration, the instrument scans through the scan list three times. Now assume the scan setup is saved in SAV1, and SAV1 is the power-on default. After cycling power, press SCAN or STEP to run the scan. The scan runs properly and scans 10 channels three times. However, if you check the reading count in the front-panel CONFIG menu, the reading count is 10 instead of 30 because the scan list length suggests the reading count value during the setup process.

NOTE

For more information on saving and recalling, refer to the *Model 2750 User's Manual*.

Scan reset

From the scan configuration menu, you can reset the scan configuration to the default setup for a simple scan.

For the Model 7700 switching module, channels 21 and 22 are turned off (not used), and channels 1 through 20 are configured as follows:

- Function - DCV
- Range - Auto
- Rate - Slow

All other multimeter features and functions are disabled.

When the scan is run by pressing STEP or SCAN, channels 1 through 20 will be scanned and the 20 DCV readings will be stored in the buffer.

To reset the scan configuration:

1. Press SHIFT and then CONFIG to enter the scan configuration menu.
2. Press the ▲ or ▼key to display INT: RESET and press ENTER.

After briefly displaying "LIST RESET," the instrument returns to the normal measurement state.

Simple scan

For a simple scan, you specify a starting channel (MIN CHAN) and an ending channel (MAX CHAN) for the scan. These settings determine the number of channels in the scan. For example, if you set MIN CHAN to 101 and MAX CHAN to 110, there are 10 channels in the scan list.

The starting channel number must be lower than the ending channel number. If you enter an invalid value, the message `TOO SMALL` or `TOO LARGE` is displayed briefly. The displayed channel number defaults to the lowest available channel, typically 101, or the highest available channel. such as 220.

To configure a simple scan:

1. In the normal measurement mode, set up the instrument for your test. This setup is used for all selected channels in the scan.
2. Press SHIFT and then CONFIG to access the scan setup menu.
3. Press the ▲ or ▼ key to display `INT: SIMPLE` and press ENTER.
4. Set the minimum channel (MIN CHAN) for the scan and press ENTER.
5. Set the maximum channel (MAX CHAN) for the scan and press ENTER.
6. Enable (Yes) or disable (No) the timer and press ENTER.
7. If you enabled the timer, set the timer interval using the hour/minute/second format. The timer can be set from 0.001 s (00H:00M:00.001S) to 99 hrs, 99 min, 99.999 s (99H:99M:99.999S). Note that pressing the AUTO key sets the timer to 0.001 s. With the interval displayed, press ENTER.
8. The displayed reading count (RDG CT) sets the number of channels to scan (STEP) or the number of scans to run (SCAN). You can change the reading count to any finite value from 2 to 110000, or you can select infinite (continuous scanning). To select infinite, set the reading count to 000000 to display INF. With the reading count setting displayed, press ENTER to return to the normal measurement display state.

Simple scanning example

With at least one multiplexer switching module, such as a 7700, installed in the mainframe, the instrument can scan channels that are valid for the selected function.

Any valid switching module channel can be included in the scan list. List them from the lowest numbered channel to the highest. For example, to scan channels 1 through 8 of a Model 7700 installed in slot 1, send the following command to define the scan list:

```
ROUTe:SCAN (@101:108)
```

The following code example performs a simple scan:

```
ROUTe:SCAN C7700           '  Define scan list.
TRIGger:COUNt <NRf>        '  Specify number of scans
                           '  (1 to 11000 or INFinity).
SAMPle:COUNt <NRf>         '  Specify number of channels
                           '  to scan (1 to 11000).
ROUTe:SCAN:LSELect <name> '  Enable (INT) or disable
                           '  (NONE) scan.
```

Simple scanning programming example

The programming example in the following steps assumes a Model 7700 installed in slot 1 of the mainframe. The scan uses default settings (DCV) to scan eight channels and store the readings in the buffer.

This code example:

- Clears the buffer.
- Restores defaults. The factory and `*RST` defaults open all channels, select the DCV function, and set `TRIG:COUN` to `1`. The trigger count specifies the number of scans to be performed.
- Configures the scan.
- Enables and starts the scan. `ROUT:SCAN:LSEL INT` enables the scan. `INIT` triggers the start of the scan.
- Stops the scanner.
- Recalls the eight stored readings.
- Opens all channels.

The code example is:

```
TRAC:CLE
*RST
ROUT:SCAN (@101:108)
SAMP:COUN 8
ROUT:SCAN:LSEL INT
INIT
ROUT:SCAN:LSEL NONE
CALC1:DATA?
```

Advanced scan

For an advanced scan, each enabled channel can have its own setup. Channels that are disabled are excluded from the scan list.

When you enter the channel setup menu, the displayed information indicates the present setup for the selected channel. The position of the decimal point in the "SETUP" message indicates present range. Examples are shown in the following table.

SETUP	V:101	1 V range for channel 101. If the AC annunciator is off, the function is DCV. If it is on, ACV is selected.
SETUP	KΩ:102	10 kΩ range for channel 102. If the 4-wire annunciator is off, the function is Ω2. If it is on, Ω4 is selected.
SETUP	mA:121	100 mA range for channel 121. If the AC annunciator is off, the function is DCI. If it is on, ACI is selected.
SETUP	°C:103	TEMP function selected for channel 103.
SETUP	HZ:104	FREQ function selected for channel 104.
SETUP	S:105	PERIOD function selected for channel 105.
SETUP	PR:111	For the Model 7700, channel 111 is paired to channel 101, and cannot be changed. Channel pairing occurs when Ratio or Channel Average is enabled, or when a 4-wire function (Ω4 or 4-wire RTD TEMP) is selected.

The annunciators indicate which of the other instrument settings are enabled for the selected channel. When you edit settings for the selected channel, such as autorange, relative offset, and rate, the related annunciators turn on or off.

Advanced scan setup notes

The CHAN annunciator is on when the instrument is in the scan setup menu.

For some channel-specific setups, you must configure them from a menu. For example, to set up and enable mx+b, you must use the MATH menu. In that menu, the CHAN annunciator flashes to indicate that you are editing the mx+b math setup for that channel in the scan list. When you exit from the mx+b setup menu, the CHAN annunciator stops flashing.

A paired channel function or operation can only be selected for a primary channel. For the Model 7700, channels 1 through 10 are the primary channels. If you try to select a paired channel function or operation for channels 11 through 22, the message `INVALID CHAN` is displayed.

When you press a function key, the selected channel assumes the mainframe setup for that function. Also, available channels for the specified slot that follow also assume that setup. For example, for the Model 7700, if you press DCV for channel 101, channels 101 through 120 assume the DCV setup. Note that channels for other slots are not affected.

When you press the Ω4 function key for a primary channel, the subsequently paired channels are displayed briefly. For example, if you are using a Model 7700 and you press Ω4 for channel 108, channels 109 and 110 also assume the Ω4 function. The message `118-120 PRD` is displayed to indicate the paired channels.

A channel that is paired to a primary channel is not affected by function changes. For example, assume channel 102 of a 7700 is paired to channel 112. Select channel 103 and press DCV. All following channels, except channel 112, assume the DCV setup. Channel 112 remains paired to channel 102. However, if you select channel 101 and press DCV,

channel 102 changes to DCV and is not be paired to channel 112 anymore. Therefore, all 20 channels assume the DCV setup.

When you press a key to change a setting such as range, relative offset, or digits, only the selected scan channel is affected. For example, if you make a range change on the 7700 for channel 103, the range settings for other channels are not affected.

There can only be one USER RTD per scan list.

Advanced scan setup procedure

Step 1: Select the advanced scan configuration menu

1. Press **SHIFT** and then **CONFIG** to access the scan setup menu.
2. Press the ▲ or ▼ key to display `INT: ADVANCED` and press **ENTER**.

Step 2: Edit scan channels

1. Use the ◄ or ► key to select channel 101 (the factory default):
   `SETUP V:101`

NOTE

You can also use the CLOSE key to select a scan channel to be edited. Press CLOSE, use the ▲ or ▼ keys to display the channel, and then press ENTER.

2. You can disable the channel or use it in the scan. Do one of the following:
   - If you do not want to use the channel, press **SHIFT** and then **CH-OFF** to disable the channel. Available channels that follow are also disabled. Channels for the other slots are not affected.
   - If you want the channel in the scan, you can either use the presently selected function or select a valid measurement function. When you press a function key such as DCV, the channel assumes the setup of the selected function. Available channels that follow assume the same setup. Channels for the other slots are not affected.

NOTE

For the Model 7700, DCI and ACI cannot be selected for module channels 1 through 20. DCI and ACI are the only functions that can be set for module channels 21 and 22.

3. If you did not disable the channel, make your setup changes (if any) for the selected function. These changes do not affect the following channels.

4. Using the ◄ or ► keys or the CLOSE key to select the channel, repeat the previous two steps to set other channels.
5. When finished setting up channels, press **ENTER** to proceed to set up triggering.

NOTE

If there are not at least two channels in the scan list (two or more channels enabled), the message `INVALID LIST` message appears briefly. You cannot exit from the scan configure menu or finish the scan setup until you enable at least two channels.

The remaining steps are used to check or change the setups for triggering, timer, and reading count. If you are not going to make changes to any of those setups, you can exit from the scan setup menu by pressing **EXIT** twice. The instrument returns to the normal measurement mode.

Step 3: Enable immediate scan

The present state of immediate scan (IMM SCAN) is displayed, Y (yes, which is the factory and *RST default) or N (no). With immediate scan enabled, the scan starts when you press the STEP or SCAN key. Use the ▲ or ▼ key to display IMM SCAN: Y, and press ENTER.

If you want to use a monitored reading limit to trigger the start of the scan, disable immediate scan (IMM SCAN: N). More information on using this technique to start a scan is provided in Scan operation (on page 3-46).

Step 4: Timer controlled scan

The present state of the TIMER is displayed, OFF or ON. If you do not want to use the timer, use the ▲ or ▼ key to display `TIMER? OFF` (which is the factory and *RST default) and press **ENTER**.

To use the timer, use the ▲ or ▼ key to display `TIMER? ON` and press **ENTER**. The timer interval is displayed in the hour/minute/second format. The timer can be set from 0.001 s (00H:00M:00.001S) to 99 hr, 99 min, 99.999 s (99H:99M:99.999S). Pressing the AUTO key sets the timer to 0.001 s. With the interval displayed, press **ENTER**.

Step 5: Set the reading count

The displayed reading count (RDG CT) sets the number channels to scan (STEP) or the number of scans to run (SCAN). You can change the reading count to any finite value from 2 to 110000, or you can select infinite (continuous scanning). To select infinite, set the reading count to 00000 to display INF. With the reading count setting displayed, press ENTER. The instrument returns to the normal measurement mode.

Monitor channel

When a scan list channel is in the normal measurement state, it can be used to monitor readings. When a channel is selected to be the monitor, it assumes the setup of the scan list channel.

NOTE

If you change the setup when a monitor channel is closed, that setup is copied to that channel in the scan list.

When a scan is started, the first channel in the scan list is briefly displayed. While the scan is in progress, the display only shows the readings for the monitor channel. After the last channel is scanned, the scan is disables with the monitor channel closed.

Monitor can be used with limit testing to trigger the start of a scan. When the monitor detects that a set reading limit has been reached, the scan is triggered to start. A detailed example of performing a monitor scan is provided in Monitor scan (on page 3-54).

NOTE

An overflow reading (`OVRFLW` message displayed) is interpreted by the 2750 as a positive reading, even if the input signal is negative. This could inadvertently trigger a monitor scan. If the monitor channel is removed from the scan list, the lowest channel in the scan list becomes the monitor channel.

To set a channel to monitor, the instrument can be in the normal measurement state or enabled in the advanced scan menu. You can use one of the following methods to select a monitor channel:

- Select monitor when a channel is closed
- Select monitor with no channels closed

NOTE

The monitor channel must be a channel that is in the scan list.

To select monitor when a channel is closed:

1. Use the **CLOSE** key or the ◄ and ► keys to close the channel that you want to be the monitor.
2. Press **SHIFT** and then **MONITOR** (MON annunciator turns on).

Select monitor with no channels closed:

1. If a channel is closed, press **OPEN** to open it.
2. Press **SHIFT** and then **MONITOR**.
3. Use the ◄, ►, ▲, and ▼ keys to display the monitor channel (for example, `MONITOR 101`) and press **ENTER**. The monitor channel closes and the MON annunciator turns on.

To disable monitor, again press **SHIFT** and then **MONITOR** (MON annunciator turns off).

Once enabled, you can change the monitor channel using the CLOSE key or the ◄ and ► keys. If you open the monitor channel, the monitor is not disabled but it does become inactive (MON annunciator turns off). When a channel is closed, monitor becomes active (MON annunciator turns on).

In the normal measurement state, the present monitor channel dictates which channel in the scan list is the monitor. Therefore, if you change the monitor channel, the scan list monitor channel also changes.

When you change the monitor channel when it is in the normal measurement state, the instrument setup does not change. If you want the monitor channel to assume the setup of the scan list channel, you must disable the monitor and then re-enable it.

Autochannel configuration

Autochannel configuration allows you to recall scan list setups. With autochannel configuration enabled, a closed channel assumes the scan list setup. With this feature, you can inspect the channel setups of the scan, or manually scan channels. When a scan channel is disabled (not in scan list), it cannot be closed with autochannel configuration enabled.

As with normal operation, when you use the ◄, ►, or CLOSE keys to close a channel (or channel pair), any other closed channels are first opened.

To enable or disable autochannel configuration.

1. Press SHIFT and then SETUP.
2. Use the ▲ or ▼ key to display the autochannel configuration (CH AUTOCFG) setting, N (no) or Y (yes).
3. Press the ► key to place the cursor on the present setting (N or Y), and press the ▲ or ▼ key to change the setting.
4. Press ENTER to return to the normal measurement state.

NOTES:

Autochannel configuration cannot be enabled if there is a non-scan channel presently closed. For example, assume the scan list consists of channels 105 through 110 and channel 101 (a non-scan channel) is presently closed. When you attempt to enable autochannel configuration from the front panel, the message `NOT IN SCAN` is briefly displayed. For remote operation, error −221, settings conflict, occurs.

With autochannel configuration enabled, the ◄ and ► keys do not properly step through a non-sequential scan list. Therefore, do not use autochannel configuration for a non-sequential scan list. For information on non-sequential scans, see Sequential and non-sequential scans (on page 3-26).

For remote operation, the ROUT:CLOS:ACON (on page 9-69) command is used to enable or disable autochannel configuration.

Saving setup

Up to three instrument setups can be saved in memory using the SHIFT > SAVE menu (SAV0, SAV1, and SAV2). A user-saved setup can also be used as the power-on setup. A user-saved setup can be restored from the SHIFT > SETUP menu. Details on user setups are described in the *Model 2750 User's Manual*.

Autoscan

When autoscan is enabled, the scan operation is saved in memory. If power to the 2750 is interrupted, the scan resumes when power is restored. With auto scan enabled, the last scan setup becomes the power-on setup. It takes precedence over the factory, *RST, or user-saved power-on setup.

To set auto scan.

1. In the normal measurement state, press **SHIFT** and then **SETUP**.
2. Use the ▲ and ▼ keys to display the autoscan (AUTOSCAN) setting, N (no) or Y (yes).
3. To change the setting, press ► to place the cursor on the setting (N or Y) and press ▲ and ▼ to change the setting.
4. Press **ENTER** to exit from the menu structure.

NOTE

With autoscan enabled, do not save the present setup as the power-on default setup. If you do, an interrupted scan will not resume. During the power-up sequence, if the 2750 detects a card ID change for any slot, autoscan configuration is disabled and an interrupted scan is not resumed. Error +517, cannot resume scan, occurs to indicate that the scan has been disabled. The instrument assumes the normal power-on setup.

The Model 7706 does not support autoscan. Trying to enable autoscan with a Model 7706 card installed causes error −221, settings conflict.

Scan operation

A basic scan is controlled solely by the STEP and SCAN keys. When one of these keys is pressed, the STEP or SCAN operation is performed. For the manual/external trigger scan, the TRIG key or triggers received from another instrument starts the STEP or SCAN operation. For the monitor scan, a channel monitors readings. When a set reading limit is reached, STEP or SCAN will start.

Basic scan

When the scan is enabled (STEP or SCAN annunciator on), most front-panel keys are inoperative and cause the message `HALT SCANNER` to display.

When running a scan, the Timer Delay for STEP occurs between channels and the Timer Delay for SCAN occurs between scans. If the timer is off, both scans run at virtually the same speed.

To run the presently configured scan:

1. To start the scan, press **STEP** or **SCAN**.
2. The STEP or SCAN annunciator turns on and channels are scanned from the lowest to highest numbered channel. Channels that are turned off are not scanned.
3. With reading count set to a finite value, the last channel scanned opens and the first channel in the scan list closes. The scan is still enabled (STEP or SCAN annunciator on). When you press STEP or SCAN, the scan continues, starting with the next channel.

   With reading count set to infinite, the scan keeps repeating.
4. To recall scanned readings stored in the buffer, press **RECALL** and use the ◄, ►, ▲, and ▼ keys to navigate through the buffer. You can read the buffer while the instrument is storing readings. To exit from buffer recall, press the **EXIT** key.
5. To disable the scan, press **SHIFT** and then **HALT**.

Manual/external trigger scan

The only difference between a manual/external trigger scan and the basic scan is control. The basic scan runs as soon as the STEP or SCAN key is pressed. The manual/external trigger scan is controlled by the front-panel TRIG key or by triggers received from another instrument.

For the following procedure, the 2750 can be triggered by pressing the TRIG key or by receiving a trigger pulse from another instrument. Refer to Triggering (on page 7-1) for details on triggering.

NOTE

Channels for an advanced scan can be configured using different mx+b units, such as Ω and °, temperature sensors, such as 4-wire RTD and thermistor, and measurement type, such as offset-compensated ohms and DCV. However, when readings are recalled from the buffer, the display may not indicate the correct mx+b units symbol or annunciator for each channel. For example, assume one channel used offset-compensated ohms and a second used Ω2. When the readings are recalled, the OCOMP annunciator may remain on for both channels.

This display anomaly is due to memory limitations. Preserving the mx+b units and annunciators for each channel would reduce the number of readings that could be stored in the buffer.

To run a manual/external scan:

1. If the scanner is presently enabled (STEP or SCAN annunciator on), press **SHIFT** and then **HALT** to disable it.
2. Press **EX TRIG** to place the instrument in the external triggering mode. The TRIG annunciator turns on and the reading is blanked (- - - - - - -).
3. Press **STEP** or **SCAN** to enable the scan (STEP or SCAN annunciator turns on).
4. The **TRIG** key or input triggers control the scan as follows:

   STEP operation: In general, each time the **Model 2750** is triggered, one channel is scanned. When the STEP key is pressed to enable the scan, the first channel in the scan list closes. When the first trigger occurs, a measurement is made, the channel opens and the next channel closes. This process continues for each channel in the scan. After the last channel in the scan list is scanned, the first channel in the scan list closes.

   The reading count determines how many channel measurements will be performed during the scan sequence. If the reading count is greater than the scan list length, operation loops back to the beginning of the scan list and continues.

After a scan sequence (as determined by the reading count) is completed, the scan remains enabled (STEP annunciator on), but the 2750 goes into the idle state. To repeat the scan sequence, take the 2750 out of idle. This can be done by pressing the **STEP** (or **TRIG**) key.

SCAN operation: In general, when the 2750 is triggered, a complete scan of all the channels in the scan list is performed. When the SCAN key is pressed to enable the scan, the first channel in the scan list closes. When a trigger occurs, one scan of the scan list channels is performed. After the last channel is scanned, the first channel in the scan list will close.

The reading count determines how many scans are performed (see Trigger models when using the front panel (on page 3-27)). If programmed for another scan, it starts when another trigger occurs.

After the last scan is completed, the scan remains enabled (SCAN annunciator on), but the 2750 goes into the idle state. To repeat the scans, take the 2750 out of idle. This can be done by pressing the **SCAN** (or **TRIG**) key.

5. When finished, press **SHIFT** and then **HALT** to disable the scan, and press **EX TRIG** to take the 2750 out of the external triggering mode.

Monitor scan (analog trigger)

A channel can be assigned as a monitor channel. When the monitor channel detects that a reading limit has been reached, the scan is triggered to start.

There are four reading limits that can be used to trigger the start of the scan:

- Low limit 1 (LLIM1)
- High limit 1 (HLIM1)
- Low limit 2 (LLIM2)
- High limit 2 (HLIM2)
- The scan starts when any enabled reading limit event is detected by the monitor channel. Refer to Limits (on page 6-1) for detail on limits.

NOTE

An overflow reading (`OVRFLW` message displayed) is interpreted by the 2750 as a positive reading, even if the input signal is negative. This could inadvertently trigger a monitor scan. For example, assume the monitor channel is monitoring a negative input signal, and the instrument is configured to trigger a monitor scan if a positive input signal is detected. If for some reason, the negative input signal exceeds the measurement range, the overflow reading is interpreted as positive and trigger the start of the scan.

To run a monitor scan:

NOTE

The last enabled scan function (STEP or SCAN) is used for the monitor scan.

1. Press **SHIFT** and then **CONFIG** to access the scan setup menu.
2. Press the ▲ or ▼ key to display `INT: ADVANCED` and press **ENTER**.
3. Use the ◄ or ► key to select channel 101 (the factory default):
   `SETUP V:101`

NOTE

You can also use the CLOSE key to select a scan channel to be edited. Press CLOSE, use the ▲ or ▼ keys to display the channel, and then press ENTER.

4. You can disable the channel or use it in the scan. Do one of the following:
   - If you do not want to use the channel, press **SHIFT** and then **CH-OFF** to disable the channel. Available channels that follow are also disabled. Channels for the other slots are not affected.
   - If you want the channel in the scan, you can either use the presently selected function or select a valid measurement function. When you press a function key such as DCV, the channel assumes the setup of the selected function. Available channels that follow assume the same setup. Channels for the other slots are not affected.

NOTE

For the Model 7700, DCI and ACI cannot be selected for module channels 1 through 20. DCI and ACI are the only functions that can be set for module channels 21 and 22.

5. If you did not disable the channel, make your setup changes (if any) for the selected function. These changes do not affect the following channels.
6. Using the ◄ or ► keys or the CLOSE key to select the channel, repeat the previous two steps to set other channels.
7. When finished setting up channels, press **ENTER** to proceed to set up triggering.

NOTE

If there are not at least two channels in the scan list (two or more channels enabled), the message `INVALID LIST` message appears briefly. You cannot exit from the scan configure menu or finish the scan setup until you enable at least two channels.

The remaining steps are used to check or change the setups for triggering, timer, and reading count. If you are not going to make changes to any of those setups, you can exit from the scan setup menu by pressing **EXIT** twice. The instrument returns to the normal measurement mode.

8. With the channel to be used as the monitor selected, set and enable limits as follows. Note that you only need to set values for limits that are going to be used.
   a. Press SHIFT and then LIMITS to access the limits menu. Note that the CHAN annunciator flashes to indicate that the menu is being used to set up a scan channel.
   b. Use the ◄, ►, ▲, and ▼ keys to set high limit 1 (HI1) and press **ENTER**.
   c. Set low limit 1 (LO1) and press **ENTER**.
   d. Set high limit 2 (HI2) and press **ENTER**.
   e. Set low limit 2 (LO2) and press **ENTER**. The instrument returns to the scan setup menu.
   f. Press **SHIFT** and then **ON/OFF** to display the present state of LIMITS (ON or OFF). The CHAN annunciator flashes to indicate that the menu is for a scan channel.
   g. Press the ▲ or ▼ key to display `LIMITS:ON` and press **ENTER**. The instrument returns to the scan setup menu. Note that the HIGH and LOW annunciators are on to indicate that limits are enabled.
   h. After all scan channels are set up, press **ENTER**. The present state of IMM SCAN is Y (yes) or N (no).
9. Press the ▲ or ▼ key to display `IMM SCAN: N` and press **ENTER**.
   a. Press the ▲ or ▼ key to enable or disable low limit 1 (LLIM1 SCAN:N/Y), and press **ENTER**.
   b. Press the ▲ or ▼ key to enable or disable high limit 1 (HLIM1 SCAN:N/Y), and press **ENTER**.
   c. Press the ▲ or ▼ key to enable or disable low limit 2 (LLIM2 SCAN:N/Y), and press **ENTER**.
   d. Press the ▲ or ▼ key to enable or disable high limit 2 (HLIM2 SCAN:N/Y), and press **ENTER**.

10. To use the timer, use the ▲ or ▼ key to display `TIMER? ON` and press **ENTER**. The timer interval is displayed in the hour/minute/second format. The timer can be set from 0.001 s (00H:00M:00.001S) to 99 hr, 99 min, 99.999 s (99H:99M:99.999S). Press the **AUTO** key to set the timer to 0.001 s. With the interval displayed, press **ENTER**.
11. The displayed reading count (RDG CT) sets the number channels to scan (STEP) or the number of scans to run (SCAN). You can change the reading count to any finite value from 2 to 110000, or you can select infinite (continuous scanning). To select infinite, set the reading count to 00000 to display INF. With the reading count setting displayed, press ENTER. The instrument returns to the normal measurement mode.

NOTE

For a remote programmed monitor scan, use the ROUTe:MONitor:POINts (on page 9-73) command to specify the number of channels to scan.

12. In the normal measurement state, select and enable the monitor channel as explained in Monitor channel (on page 3-43).

    When the reading limit for the monitor channel is reached, the scan is triggered to start. When the monitor channel is scanned, the display shows the reading that triggered the scan.

    If the reading limit event is still present on the monitor channel when the scan finishes, the scan is triggered to run again. You can also run the scan by pressing STEP or SCAN.
13. To disable the monitor scan:
    a. To disable monitor, press **SHIFT** and then **MONITOR** (MON annunciator turns off).
    b. To disable limits, press **SHIFT** and then **ON/OFF**. Press ▲ or ▼ to display `LIMITS: OFF` and press **ENTER**.

Scanning using remote programming

NOTE

Scanning examples (remote programming and front-panel operation) are provided in Scanning examples (on page 3-52).

Channel setup

The `<clist>` parameter is used to define scan channels. For example, the following examples show how to set up scan channel 101:

```
FUNC 'VOLT', (@101)       ' Set 101 for DCV.
VOLT:RANG 10, (@101)      ' Set 101 for 10 V range.
VOLT:DIG 4.5, (@101)      ' Set 101 for 4½ digit resolution.
VOLT:NPLC 3, (@101)       ' Set 101 rate for 3 PLC.
```

NOTE

In the above command sequence, channel 101 is first set for DCV before sending the other commands to set range, digits, and rate. If channel 101 was set to a different function, such as RESistance, the VOLT commands to set range, digits, and rate would generate error +700, Invalid function in scanlist.

Buffer

For front-panel scanning, the reading count specifies the number of readings to store in the buffer. For remote scanning, the sample count specifies the number of readings to store in the buffer.

Readings stored in the buffer by the TRACe command or by front-panel data store operation must be cleared before sending `INITiate` or `READ?` to take the instrument out of idle. The following command clears the buffer:

```
TRACe:CLEar              ' Clear buffer.
```

Scanning examples

The following scanning examples assume that the Model 7700 switching module is installed in slot 1 of the mainframe.

Tables are used for the procedure steps to configure and run scan examples. The left side of the table provides the front-panel procedure. The right side shows the equivalent remote programming commands. Where appropriate, menu sequences are provided to summarize a front-panel operation or selection. For example:

SHIFT SETUP > RESTORE: FACT

For the above menu sequence, press **SHIFT** and then **SETUP** to access the menu, use the edit keys (◄, ►, ▲, and ▼) to display `RESTORE: FACT`, and then press **ENTER** to select it.

External trigger scan

For this example, an external instrument is used to trigger the start of the 2-channel scan. Trigger pulse requirements and trigger cable connections are described in Triggering (on page 7-1).

For this example, the front-panel TRIG key can be used in place of an external input trigger. Each time the TRIG key is pressed, the 2-channel scan will run.

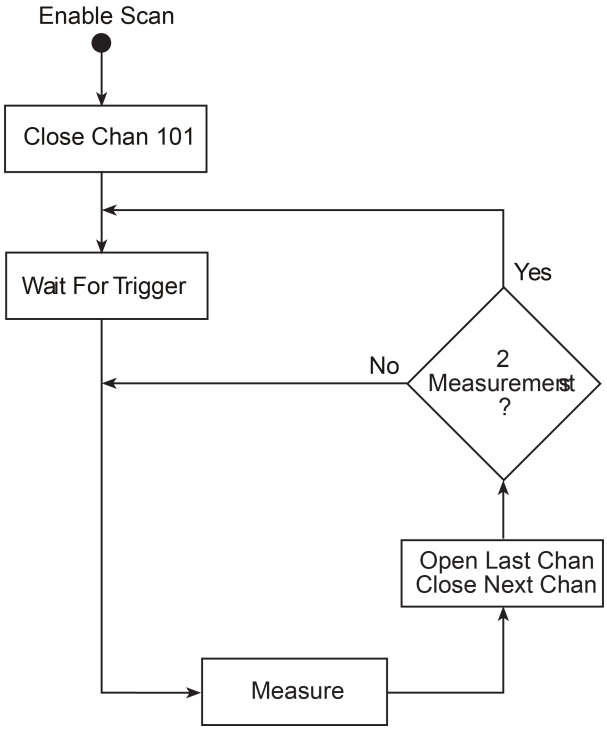
One channel (101) measures temperature and the other channel (102) measures resistance. The two readings are stored in the buffer. Each time the scan is run, the two readings will be appended (added) to the buffer.

A type K thermocouple is used to measure temperature. Since the internal cold (reference) junction of the Model 7700 is being used, the thermocouple can be connected directly to the screw terminals of the switching module.

Operation

A simplified model of external trigger scan operation is shown in the following example. The steps and programming commands are listed in the following table.

Figure 16: External trigger scan example



As shown in the operation model, when the scan is enabled, channel 101 closes and the 2750 waits for an external trigger. When the trigger is received, channels 101 and 102 are measured. Operation then returns to the control source where it waits for another trigger.

After the scan is enabled, you can use the TRIG key to trigger the scan.

External trigger scan example

	Front-panel operation	Remote programming
1	Restore defaults:	
	SHIFT SETUP > RESTORE: FACT.	`*RST`
2	For front-panel operation, proceed to step 3.	
	For remote programming, clear buffer and disable buffer autoclear:	`TRAC:CLE` `TRAC:CLE:AUTO OFF`
3	Configure advanced scan:	
	(SHIFT CONFIG > ADVANCED):	
a	Channel 101:	
	Select TEMP function.	`FUNC 'TEMP',(@101)`
	Configure temperature (SHIFT SENSOR):	
	Select thermocouple sensor (SENS: TCOUPLE).	`TEMP:TRAN TC,(@101)`
	Select type K thermocouple (TYPE: K).	`TEMP:TC:TYPE K,(@101)`
	Select internal reference junction (JUNC: INT).	`TEMP:RJUN:RSEL INT,(@101)`
b	Channel 102:	
	Select Ω2 function.	`FUNC 'RES',(@102)`
	Select 1 MΩ range.	`RES:RANG 1e6,(@102)`
c	Disable (off) channels 103 through 122 (SHIFT CH-OFF).	`ROUT:SCAN (@101,102)`
d	Enable immediate scan (IMM SCAN: Y).	`ROUT:SCAN:TSO IMM`
e	Disable timer (TIMER? NO).	
f	Set reading count to infinity (RDG CT: INF).	`TRIG:COUN INF`
		`SAMP:COUN 2`
4	Select external trigger control source:	
	Press EX TRIG.	`TRIG:SOUR EXT`
5	Enable scan:	
	Press SCAN.	`ROUT:SCAN:LSEL INT` `INIT`

Monitor scan

For this example, channel 101 of the Model 7700 is used to monitor temperature. When the temperature reading reaches 30°C, it starts the scan. For this 4-channel scan, channel 101 measures temperature and channels 102, 103, and 104 measure DCV.

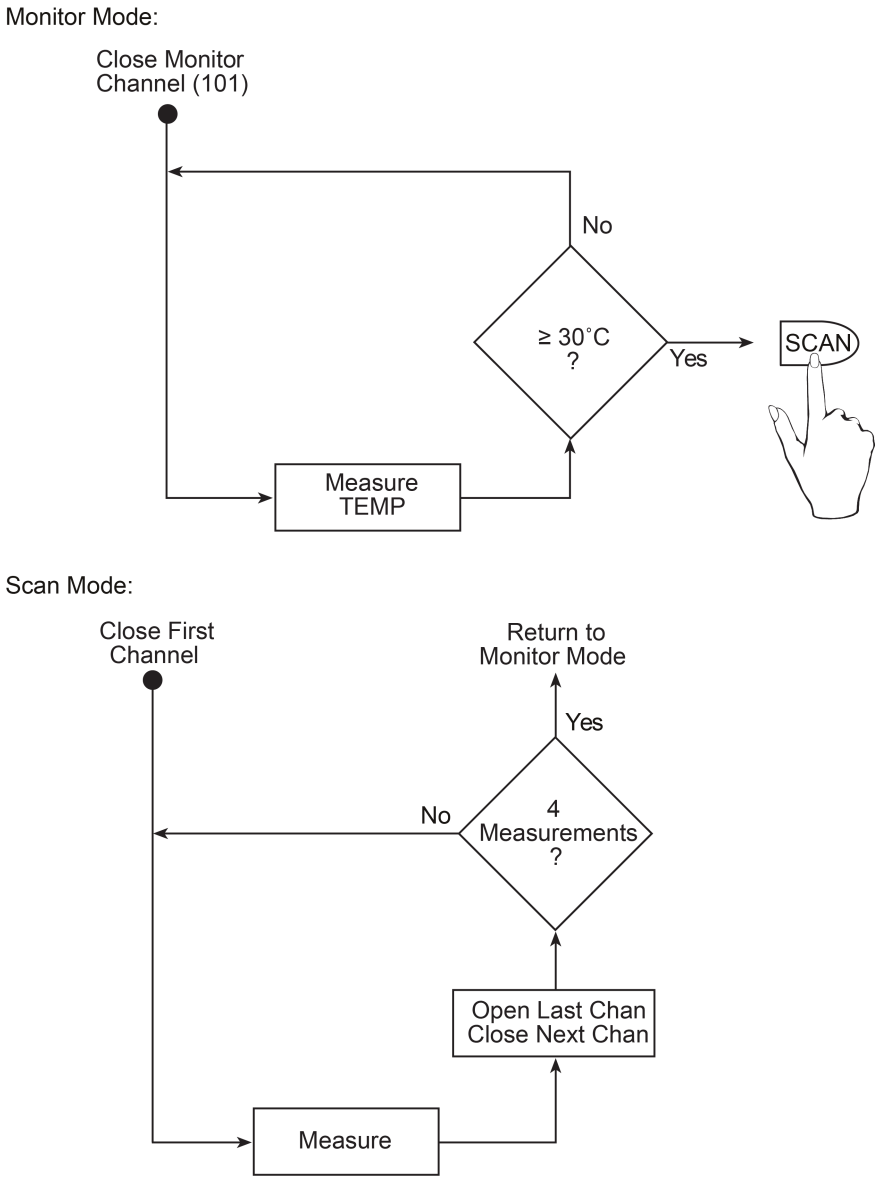
This example uses the Channel Average feature to measure temperature. With channel average enabled, two temperature measurements will be taken: one at channel 101 and another at its paired channel (111). The two measured readings are then averaged to yield a single reading. It is this averaged temperature reading that starts the scan when it reaches 30°C. Refer to Ratio and channel average (on page 4-18) for details on channel average.

Two type K thermocouples measure temperature. Since the internal cold (reference) junction of the Model 7700 is being used, the thermocouples can be connected directly to the screw terminals of the switching module.

Operation

A simplified model of monitor scan operation is shown in the following figure. The procedure steps and programming commands are listed in the following table.

Figure 17: Monitor scan example



In the previous figure, notice that there are two modes of operation. When the 2750 is in the monitor mode, it continuously performs temperature measurements. Since channel average is used, each temperature reading is the average of two temperature measurements (one on channel 101 and one on channel 111). As long as the average temperature reading remains below 30°C, the instrument remains in monitor mode.

When the temperature reading reaches 30°C, the 2750 switches over to the scan mode. This is similar to pressing the SCAN key when the monitor detects that the average temperature is at or above 30°C.

The instrument is configured to scan four channels. The monitor TEMP channel reading and three DCV channel readings are stored in the buffer. After the fourth channel is measured, operation returns to the monitor mode, to measure temperature again. Note that if the average temperature is still at or above 30°C, the scan runs again.

Monitor scan example

	Front-panel operation	**Remote programming**
1	Restore defaults (SHIFT SETUP > RESTORE: FACT).	`SYST:PRES`
2	For front-panel operation, proceed to step 3.	
	For remote programming, clear the buffer:	`TRAC:CLE`
3	Configure advanced scan:	
	SHIFT CONFIG > ADVANCED:	
a	Channel 101:	
	Select TEMP function.	`FUNC 'TEMP',(@101)`
	Configure temperature (SHIFT SENSOR):	
	Select Thermocouple sensor (SENS: TCOUPLE).	`TEMP:TRAN TC,(@101)`
	Select type K thermocouple (TYPE: K).	`TEMP:TC:TYPE K,(@101)`
	Select internal reference junction (JUNC: INT).	`TEMP:RJUN:RSEL INT,(@101)`
b	Set and enable high limit 1:	
	Set limit to 30 (SHIFT LIMITS > HI1:+30.00000).	`CALC3:LIM1:UPP 30,(@101)`
	Enable (on) limit (SHIFT OFF/ON > LIMITS: ON).	`CALC3:LIM1:STAT ON,(@101)`
	Enable Channel Average (SHIFT CH AVG).	`CAV ON,(@101)`
c	Channel 102, 103, and 104:	
	Select DCV function.	`FUNC 'VOLT',(@102:104)`
	Select 10V range.	`VOLT:RANG 10,(@102:104)`
	Set filter count to 20 (SHIFT TYPE > 020 RDGS).	`VOLT:AVER:COUN 20,(@102:104)`
	Enable filter (FILTER).	`VOLT:AVER:STAT ON,(@102:104)`
d	Disable (off) channels 105 through 222 (SHIFT CH OFF).	`ROUT:SCAN (@101:104)`
e	Disable immediate scan (IMM SCAN: N), and enable high limit 1 (HLIM1 SCAN:Y).	`ROUT:SCAN:TSO HLIM1`
f	Disable timer (TIMER? OFF).	
g	Set reading count to 4.	`ROUT:MON:POIN 4`
4	Select and enable monitor channel (SHIFT MONITOR >101).	`ROUT:MON (@101)` `ROUT:MON:STAT ON`

Section 4

Making measurements

In this section:

Performance considerations..... 4-1
Measurement range 4-7
Digital filters..... 4-9
Relative offset 4-12
Math 4-14
Ratio and channel average 4-18
dB..... 4-21
Low-level voltage measurement considerations..... 4-22
Remote DMM measurements using the front-panel inputs 4-23
Signal processing sequence 4-25

Performance considerations

The following topics provide information on issues that can affect measurement performance.

Autozero

To help maintain stability and accuracy over time and changes in temperature, the Model 2750 periodically measures internal voltages corresponding to offsets (zero) and amplifier gains. For thermocouple temperature measurements using the internal reference junction, such as when the Model 7700 switching module is installed, the internal temperature is also measured. These measurements are used in the algorithm to calculate the reading of the input signal. This process is known as autozeroing.

When autozero is disabled, the offset, gain, and internal temperature measurements are not performed. This increases the measurement speed. However, the zero, gain, and temperature reference points will eventually drift, resulting in inaccurate readings of the input signal. It is recommended that autozero only be disabled for short periods.

When autozero is enabled after being off for a long period, the internal reference points are not updated immediately. This initially results in inaccurate measurements, especially if the ambient temperature has changed by several degrees. To force a rapid update of the internal reference points, set the integration rate to 0.01 PLC, and then back to the preferred rate (for example, 1.0 PLC).

You can use remote programming to enable or disable autozero. Autozero cannot be disabled from the front panel. However, it can be enabled by restoring factory default conditions.

To set autozero, refer to :SYSTem:AZERo:STATe (on page 9-174).

The NPLC commands to set the integration rate are described in SENSe[1] subsystem (on page 9-79).

Line cycle synchronization

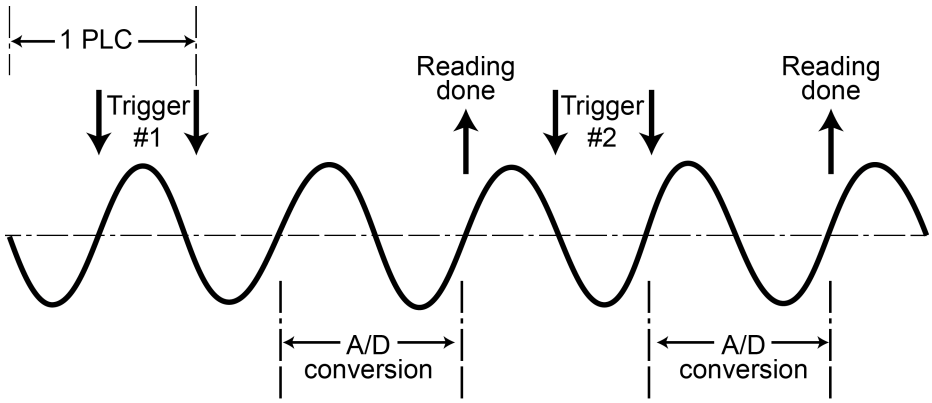
Synchronizing A/D conversions with the frequency of the power line increases common mode and normal mode noise rejection. When line cycle synchronization is enabled, the measurement is initiated at the first positive-going zero crossing of the power line cycle after the trigger.

The following figure shows a measurement process that consists of two A/D conversions. If the trigger occurs during the positive cycle of the power line (Trigger #1), the A/D conversion starts with the positive-going zero crossing of the power line cycle. If the next trigger (Trigger #2) occurs during the negative cycle, then the measurement process also starts with the positive-going zero crossing.

NOTE

Line synchronization is not available for the AC functions (ACV, ACI, FREQ, or PERIOD) and for integration rates <1 PLC, regardless of the LSYNC setting.

Figure 18: Line cycle synchronization



To enable or disable line cycle synchronization using the front panel:

1. Press SHIFT and then LSYNC to display the present state of line synchronization (OFF or ON).
2. Use the ▲ or ▼ key to display "LINESYNC ON" or "LINESYNC OFF."
3. Press ENTER. The instrument returns to the normal display state.

To enable or disable line cycle synchronization using remote commands:

Refer to :SYSTem:LSYNc:STATe (on page 9-192).

Channel list parameter (<clist>)

You can scan the channels of one or more switching modules installed in the 2750. Each scan channel can have its own setup. For example, a channel could be set to measure DCV on the 10 V range and another channel can be set to measure ACV on the 1 V range.

From the front panel, scan channels are configured from the scan configuration menu as described in Switching and scanning (on page 3-1). For remote programming, the `<clist>` parameter is used to define the configure scan channels. Refer to Channel assignments (on page 3-4) for detail on defining `<clist>`.

Throughout this manual, you will encounter commands that can use the `<clist>` parameter. The `<clist>` indicates that the associated command can be used to configure a scan channel. For example:

```
SENSe:FUNCtion 'VOLTage:AC'          ' Select ACV function.
   SENSe:FUNCtion 'VOLTage:AC', (@101)     ' Configure scan channel 101 for ACV.
```

In the normal measurement display state, the first command selects the ACV function. The second command configures channel 101 to measure ACV when it is scanned.

DCV input divider

Normally, the input resistance for the 100 mVDC, 1 VDC, and 10 VDC ranges is >10GΩ and the input resistance of the 100 VDC and 1000 VDC ranges is 10 MΩ. You can also set the input resistance for the three lower DCV ranges to 10 MΩ by enabling the input divider.

With the input resistance lowered, a more stable 0 V reading is achieved with an open input. Also, some external devices (such as a high voltage probe) must be terminated to a 10 MΩ load.

The input divider cannot be enabled from the front panel.

For remote programming, refer to [:SENSe[1]]:VOLTage[:DC]:IDIVider (on page 9-160).

Crest factor

AC voltage (and current) accuracies are affected by the crest factor of the waveform, the ratio of the peak value to the RMS value. The following table lists the fundamental frequencies at which the corresponding crest factor must be taken into account for accurate calculations.

Crest factor limitations

Crest factor	Fundamental frequency
2	50 kHz
3	3 kHz
4 to 5	1 kHz

Low-level considerations

For sensitive measurements, external considerations affect the accuracy. Effects that are not noticeable when working with higher voltages are significant in microvolt signals. The 2750 reads only the signal received at its input; therefore, it is important that this signal be properly transmitted from the source. The following paragraphs indicate factors that affect accuracy, including stray signal pick-up and thermal offsets.

Shielding

AC voltages that are extremely large compared with the dc signal to be measured may produce an erroneous output. To minimize AC interference, the circuit should be shielded, with the shield connected to the 2750 input low (particularly for low-level sources). Improper shielding can cause the 2750 to behave in one or more of the following ways:

- Unexpected offset voltages
- Inconsistent readings between ranges
- Sudden shifts in readings

To minimize pick-up, keep the voltage source and the 2750 away from strong AC magnetic sources. The voltage induced due to magnetic flux is proportional to the area of the loop formed by the input leads. Minimize the loop area of the input leads and connect each signal at only one point.

Thermal EMFs

Thermoelectric EMFs (thermoelectric potentials) are generated by temperature differences between the junctions of dissimilar metals. These can be large compared to the signals that the 2750 can measure. Thermoelectric EMFs can cause the following conditions:

- Instability or zero offset is much higher than expected.
- The reading is sensitive to (and responds to) temperature changes. This effect can be demonstrated by touching the circuit, by placing a heat source near the circuit, or by a regular pattern of instability (corresponding to changes in sunlight or the activation of heating and air conditioning systems).

To minimize the drift caused by thermoelectric EMFs, use copper leads to connect the circuit to the 2750.

For front-panel inputs, a banana plug generates a few microvolts. A clean, oxidized-free, copper conductor such as #10 bus wire is ideal. For switching modules, use #20 AWG copper wire to make connections. The leads to the 2750 may be shielded or unshielded, as necessary.

Widely varying temperatures within the circuit can also create thermoelectric EMFs. Therefore, maintain constant temperatures to minimize these thermoelectric EMFs. A shielded enclosure around the circuit under test also helps by minimizing air currents.

The relative offset control can be used to null out constant offset voltages.

For information on shielding, refer to Shielding (on page 5-8).

Thermoelectric coefficients

The following table shows the magnitude of thermoelectric EMFs that are generated for different materials. Best results are obtained with clean copper-to-copper connections.

Material thermoelectric coefficients	
Material	**Thermoelectric potential**
Copper-to-copper	0.2 µV/°C
Copper-to-silver	0.3 µV/°C
Copper-to-gold	0.3 µV/°C
Copper-to-cadmium/tin	0.3 µV/°C
Copper-to-lead/tin	1 µV/°C to 3 µV/°C
Copper-to-Kovar®	40 µV/°C to 75 µV/°C
Copper-to-silicon	400 µV/°C
Copper-to-copper oxide	1000 µV/°C

Minimizing thermoelectric EMFs

To minimize thermoelectric EMF generation:

- Construct circuits that use the same material for all conductors. For example, connections made by crimping copper sleeves or lugs on copper wires result in copper-to-copper junctions, which generate minimal thermoelectric EMFs.
- Keep connections clean and free of oxides.
- Use low-thermoelectric cables and connections.
- Keep the two materials forming the junction at the same temperature.
- Keep the two junctions close together.
- Allow test equipment to warm up and reach thermal equilibrium in a constant ambient temperature.
- Keep all junctions away from air currents; in some cases, it may be necessary to thermally insulate sensitive junctions to minimize temperature variations.
- When making a copper-to-copper connection, apply sufficient pressure to ensure the connection is gastight to prevent future oxidation.
- In some cases, you may need to connect the two thermal junctions together with good thermal contact to a common heat sink. Unfortunately, most good electrical insulators are poor heat conductors. In cases where low thermal conductivity may be a problem, you can use special insulators that combine high electrical insulating properties with high thermal conductivity. Some examples of these materials include hard anodized aluminum, sapphire, and diamond.

Using relative offset to minimize thermoelectric EMFs

Some systems may still have residual thermoelectric offsets after following the guidelines in Minimizing thermoelectric EMFs (on page 4-6). If the offsets are relatively constant, you can use the relative offset feature in the 2750 to cancel them. Refer to Relative offset (on page 4-12) for information.

AC voltage offset

The 2750, at 5½ digits resolution, typically displays 100 counts of offset on AC volts with the input shorted. This offset is caused by the offset of the T_{RMS} converter. This offset does not affect reading accuracy and should not be zeroed out using the relative offset feature. The following equation expresses how this offset (V_{OFFSET}) is added to the signal input (V_{IN}):

$$Displayed\ reading = \sqrt{(V_{IN})^2 + (V_{OFFSET})^2}$$

Example:
Range= 1 VAC, Offset = 100 counts (1.0 mV), Input = 100 mV_{RMS}

$$Displayed\ reading = \sqrt{(100\text{mV})^2 + (1.0\text{mV})^2}$$

Therefore, the displayed reading is 0.100005 V.

The offset is seen as the last digit, which is not displayed. Therefore, the offset is negligible. If relative offset were used to zero the display, the 100 counts of offset would be subtracted from V_{IN}, resulting in an error of 100 counts in the displayed reading.

Measurement range

The measurement range affects the accuracy of the measurement and the maximum signal that can be measured. The measurement ranges for each function, except frequency, period, and temperature, are listed in the following table.

Function	Ranges	Maximum reading
DCV	100 mV, 1 V, 10 V, 100 V, 1000 V	±1010 V
ACV	100 mV, 1 V, 10 V, 100 V, 750 V	757.5 V
DCI	20 mA, 100 mA, 1 A, 3 A	±3.1 A
ACI	1 A, 3 A	3.1 A
2Ω	10 Ω , 100 Ω, 1 kΩ, 10 kΩ, 100 kΩ, 1 MΩ, 10 MΩ, 100 MΩ	120 MΩ
4Ω*	1 Ω , 10 Ω, 100 Ω, 1 kΩ, 10 kΩ, 100 kΩ, 1 MΩ, 10 MΩ, 100 MΩ	120 MΩ
* Offset-compensated ohms and dry circuit ohms: OCOMP: Offset-compensated ohms can be performed on the 1 Ω, 10 Ω, 100 Ω, 1 kΩ, and 10 kΩ ranges. DRYCKT: Available ranges for dry circuit ohms include the 1 Ω , 10 Ω, 100 Ω, and 1 kΩ ranges.		

Frequency and period ranges: Frequency measurements from 3 Hz to 500 kHz and period measurements from 2 µs to 333 µs can be made on the ACV ranges.

Temperature: There is no range selection for temperature measurements. Temperature measurements are performed on a single fixed range. Depending on which type of sensor is being used, the maximum temperature readings range from –200 °C to 1820 °C.

Input values that exceed the maximum readings cause the message `OVERFLOW` to display. To resolve this issue, select a higher range until an on-range reading is displayed. Use the lowest range possible without causing an overflow to assure best accuracy and resolution.

The range setting is saved for each measurement function. When you select a function, the instrument returns to the last range setting for that function.

Manual ranging

To change the range using the front panel, press the RANGE ▲ or ▼ key. The instrument changes one range per key press. The selected range is displayed for one second. The manual range keys have no effect on temperature (TEMP).

To change the range using remote commands, specify the expected reading as an absolute value using the <n> parameter for the appropriate `:RANGe` command. The 2750 goes to the most sensitive range for that expected reading. For example, if you expect a reading of approximately 3 V, set the parameter to 3 to select the 10 V range.

Autoranging

To enable autorange, press the **AUTO** key. The AUTO annunciator turns on when autoranging is selected. When autoranging is enabled, the instrument automatically selects the best range to measure the applied signal. Autoranging should not be used when optimum speed is required.

The range is increased at 120% of range. The range is decreased when the reading is <10% of nominal range.

Autoranging has no effect on temperature (TEMP).

To disable autoranging, press **AUTO**. This leaves the instrument on the present range. You can also disable autoranging by pressing the ▲ or ▼ key, which can cause a range change.

The `:RANGe:AUTO` commands are coupled to the commands to select the range manually (`:RANGe` <n>). When autorange is enabled, the parameter value for `:RANGe` <n> changes to the automatically selected range value. When autorange is disabled, the instrument remains at the selected range. When a valid `:RANGe` <n> command is sent, autoranging is disabled.

Ranges when scanning

When a simple scan is configured, the present function and range setting applies to all channels in the scan. When an advanced scan is configured, each channel can have its own range setting. Details to configure and run a scan are provided in Switching and scanning (on page 3-1).

For remote programming, the `<clist>` parameter configures channels for a scan.

Digital filters

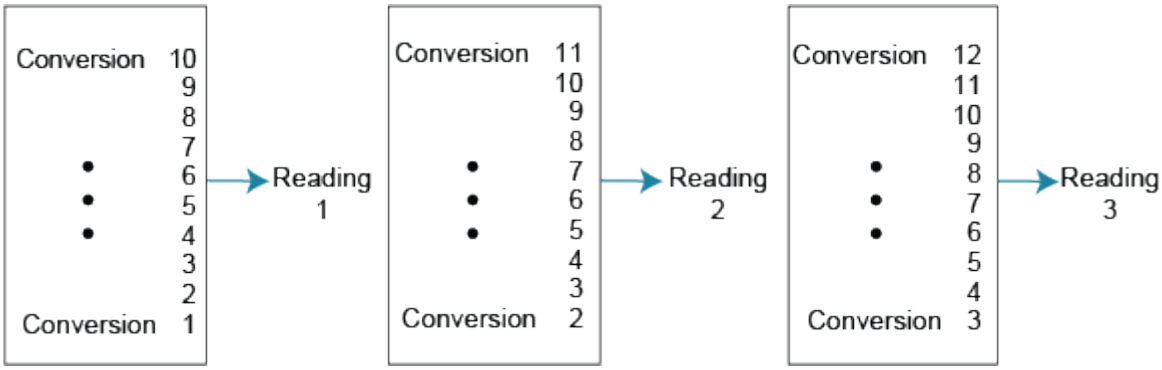
The digital filter stabilizes noisy measurements. The displayed, stored, or transmitted reading is a windowed-average of reading conversions (from 1 to 100).

In general, the digital filter places a specified number of consecutive A/D conversions (filter count) into a memory stack. These A/D conversions must occur consecutively within a selected reading (filter) window. The readings in the stack are then averaged to yield a single filtered reading. You can set the filter to use a moving or repeating average.

When the moving average filter is selected, the measurements are added to the stack continuously on a first-in, first-out basis. As each measurement is made, the oldest measurement is removed from the stack. A new averaged sample is produced using the new measurement and the data that is now in the stack.

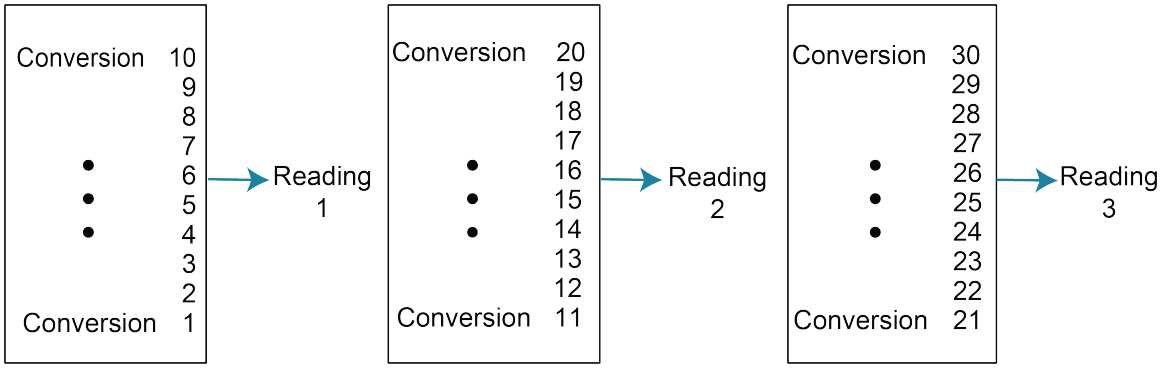
When the moving average filter is first selected, the stack is empty. When the first measurement is made, it is copied into all the stack locations to fill the stack. A true average is not produced until the stack is filled with new measurements. See the following figure for an example of a moving average filter.

Figure 19: Moving average filter



When the repeating average filter is selected, a set of measurements are made. These measurements are stored in a measurement stack and averaged together to produce the averaged sample. Once the averaged sample is produced, the stack is flushed, and the next set of data is used to produce the next averaged sample. This type of filter is the slowest, since the stack must be completely filled before an averaged sample can be produced, but it provides more stable results. See the following figure for an example of a repeating average filter.

Figure 20: Repeating average filter



For either method, the greater the number of measurements that are averaged, the slower the averaged sample rate, but the lower the noise error. Trade-offs between speed and noise are normally required to tailor the instrumentation to your measurement application.

Filter characteristics

You can set the filter type, filter count, and filter window for the digital filter. You can set the filter type to moving or repeating.

NOTE

If you are scanning, you must use the repeating filter. If a scan channel is set up to use the moving filter, the filter will not turn on. The filter window is also not used when scanning. When a simple scan is configured, the present filter count and state will apply to all channels in the scan. For the advanced scan, filter state (on or off) and count can be set for each channel. You cannot set unique filter count, type, and window settings from the advanced scan setup menu. For remote programming, the `<clist>` parameter sets the filter count and state for each channel in the scan.

The filter count specifies how many consecutive A/D conversions in the filter window to place in the memory stack. When the stack is full, the A/D conversions are averaged to calculate the final filtered reading. The filter count can be set from 1 to 100. If the filter count is 1, no averaging is done. Only readings in the filter window are displayed, stored, or transmitted.

The digital filter uses a window to control the filter threshold. While the input signal remains in the selected window, A/D conversions continue to be placed in the stack. If the signal changes to a value outside the window, the filter resets, and the filter starts processing again, starting with a new initial conversion value from the A/D converter.

Setting 0 sets the filter window to NONE.

From the front panel, you can select a window of 0.01%, 0.1%, 1%, or 10% of range, or NONE (no window). With remote commands, you can set the window to any value from 0% to 10%.

For the voltage, current, and resistance functions, the filter window is expressed as a percent of range. For example, on the 10 V range, a 10% window means that the filter window is ±1 V. For temperature, the filter window is expressed as a percent of the maximum temperature reading. The maximum temperature depends on which thermocouple is used. For example, for a Type J thermocouple, the maximum reading is 760 °C, so a 10% window creates a filter window of ±76 °C.

Filter control and configuration

The FILTER key toggles the state of filters. When filter is enabled, the FILT annunciator is on. The FILT annunciator flashes when the filter is not settled.

You can configure the filter when filter is enabled or disabled. If filter is enabled, changes to the configuration take effect as soon as they are made. With filter disabled (FILT annunciator off), changes to the configuration take effect the next time the filter is enabled.

During the filtering operation, the FILT annunciator blinks. Readings continue to be processed, but they may be questionable. When the FILT annunciator stops blinking, the filter has settled.

The filter setup is stored with each measurement function. When you select a function, the instrument returns to the last filter setup for that function. Filters are reset when the function or range changes. The filter settings change to the state (enabled or disabled) and configuration for that function or range.

Filters are applied in a sequential manner. Refer to Signal processing sequence (on page 4-25) for details.

To configure the filter using the front panel:

1. Select the function.
2. Press **SHIFT** and then **TYPE**. The present window setting is displayed.
3. Use the **RANGE** ▲ or ▼ key to select the window setting (0.01%, 0.1%, 1%, 10%, or NONE) .
4. Press **ENTER**. The present digital filter count is displayed.
5. To change the count, use the ◄ and ► keys and ▲ and ▼ keys to display the count setting (1 to 100).
6. Press **ENTER**. The present filter type (moving or repeating) is displayed.
7. To change the filter type, place the cursor on the type name and press the **RANGE** ▲ or ▼ key.
8. Press **ENTER**. The instrument returns to the normal measurement display state.

Digital filter example

Filter Count = 10
Filter Window = 0.01% of range
Filter Type = Moving

Ten readings fill the stack to yield a filtered reading. Now assume the next reading (which is the 11th) is outside the window. A reading is processed and displayed; however, the stack is loaded with that same reading. Each subsequent valid reading then displaces one of the loaded readings in the stack. The FILT annunciator flashes until 10 new readings fill the stack.

NOTE

Bit 8 of the Operation Event Status Register sets when the filter window has settled. See Status structure (on page 11-1) for details.

Relative offset

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

When a relative offset value is established for a measure function, the value is the same for all ranges for that measure function. Refer to Signal processing sequence (on page 4-25) for more information.

NOTE

You can perform the equivalent of relative offset manually by using the mx+b math function. Set `m` to 1 and b to the value of the offset.

Setting the relative offset for non-scan measurements

NOTE

If you are using the switching module inputs, make sure the front-panel INPUTS switch is set to the REAR position (in). If you are using the front-panel inputs, the switch must be in the FRONT position (out).

To set the relative offset:

1. Select the measurement function and an appropriate range setting.
2. Apply the signal that requires a relative offset to a switching channel input or to the front-panel inputs.
3. If you are using a switching module, use the ◄ or ► key to select (close) the input channel. If you are using the front-panel inputs (FRONT inputs selected), you do not need to close the switching channel.
4. Press the **REL** key to set the relative offset value. The display zeroes and the REL annunciator turns on.
5. Apply the signal to be measured.

To disable relative offset:

Press the REL key to disable relative offset. The REL annunciator turns off.

Setting the relative offset for scanning

When a simple scan is configured, the present relative offset setting applies to all channels in the scan. When an advanced scan is configured, each channel can have a unique relative offset setting.

For details on how to configure and run a scan, refer to Switching and scanning (on page 3-1).

For an advanced scan, the following general procedure shows how to configure a scan channel to use relative offset:

1. In the normal measurement state, select the appropriate function and close the appropriate channel. For example, if you are configuring scan channel 101 for DCV with a relative offset, select DCV and close channel 101.
2. Apply the DCV signal that requires a relative offset to the closed channel. This could be an offset or a baseline level.
3. Press the **REL** key to enable relative offset (REL annunciator on). The input signal level is used as the relative offset value.
4. When configuring the advanced scan, select the channel, press DCV, and then press REL (REL annunciator on).
5. When the channel is scanned, relative offset is enabled and uses the relative offset value established in step 3.

Math

The 2750 allows you to apply the following math operations to the measurement:

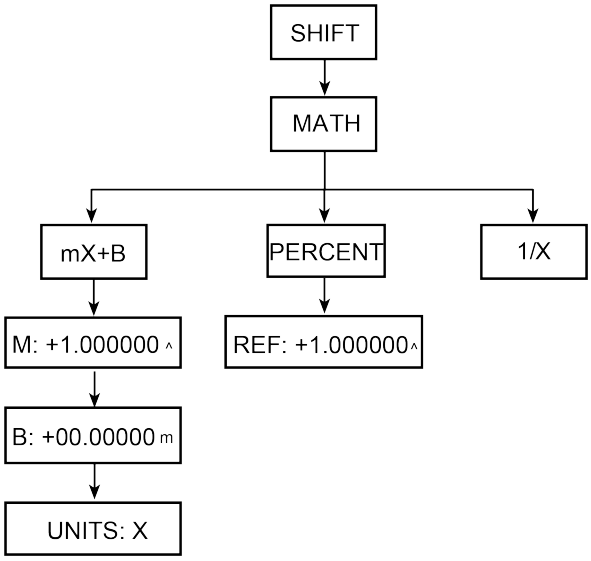
- mx+b
- percent
- reciprocal (1/X)

Math calculations are applied to the input signal after relative offset and before limit tests. For more detail on the order of operations, see Signal processing sequence (on page 4-25).

A math operation can be used with the ratio and channel average calculation. The ratio or channel average reading is used in the calculation for the selected math function.

The following figure shows the MATH menu tree. The settings shown in the menu tree are the factory defaults.

Figure 21: MATH menu tree



Calculations are applied to all measurement functions.

When a simple scan is configured, the present math calculation applies to all channels in the scan. When an advanced scan is configured, each channel can have its own math setting. Details to configure and run a scan are provided in Switching and scanning (on page 3-1).

WARNING

When using relative offset, math, ratio, channel average, and dB functions, the display may indicate a non-hazardous voltage, but hazardous voltage may be present on the input connectors.

mx+b

The mx+b math operation lets you manipulate normal display readings (x) mathematically based on the following calculation:

mx + b = y

Where:

- **m** is a user-defined constant for the scale factor
- **x** is the measurement reading (if you are using a relative offset, this is the measurement with relative offset applied)
- **b** is a user-defined constant for the offset factor
- **y** is the displayed result

If you are using relative offset, the input signal with relative offset applied is used by the mx+b calculation.

mx+b configuration

To set up mx+b using the front panel:

1. Press **SHIFT** and then **MATH** to display the math menu.
2. Press the **RANGE** ▲ and ▼ keys to display `mX+b` and press **ENTER** to display the present scale factor:

   `M: +1.000000 ▲`
3. Key in the scale factor value. The ◄ and ► keys control cursor position, and the ▲ and ▼ keys increment and decrement the digit value. To change the value, place the cursor on the multiplier and use the ▲ and ▼ keys (m = ×0.001, Δ = ×1, K = ×1000, and M = ×1,000,000). With the cursor on the polarity sign, the ▲ and ▼ keys toggle polarity.
4. Press **ENTER** to enter the M value and display the offset (b) value:

   `b: +00.00000 m`
5. Key in the offset value.
6. Press **ENTER** to enter the b value and display the one-character units designator:

   `UNITS:  X`
7. To change the unit designator, use the cursor keys and the ▲ and ▼. The character can be any letter in the alphabet (A through Z), the degree symbol (°), or the ohms symbol (Ω).
8. Press **ENTER**. The MATH annunciator turns on and the result of the calculation is displayed.
9. Select the measurement function.

10. Apply the signal to be measured to a switching channel input or to the front-panel inputs.
11. If you are using a switching module (REAR inputs selected), use the ◄ or ► key to select (close) the input channel. If you are using the front-panel inputs (FRONT inputs selected), you do not need close the channels.

    The result of the math calculation is displayed.
12. To disable mx+b, press **SHIFT** and then **MATH**. The MATH annunciator turns off.

Set the relative offset using mx+b

You can use the mx+b function to manually establish a relative offset value. To do this, set the scale factor (m) to 1 and set the offset (b) to the offset value. Each subsequent reading is the difference between the actual input and the offset value.

Percent

The percent math function displays measurements as percent deviation from a specified reference constant. The percent calculation is:

$$\text{Percent} = \left(\frac{\text{input - reference}}{\text{reference}}\right) \times 100\%$$

Where:

- **Percent** = The result
- **Input** = The measurement (if relative offset is being used, this is the relative offset value)
- **Reference** = The user-specified constant

The result of the percent calculation is positive when the input is more than the reference. The result is negative when the input is less than the reference.

If relative offset is applied, the relative offset reading of the input signal is used by the percent calculation.

Percent configuration

To set up percent using the front panel:

1. Press SHIFT and then MATH to display the math menu.
2. Press the RANGE ▲ or ▼ key to display "PERCENT" and press ENTER to display the present reference value:

   `REF +1.000000 Δ`

3. Key in the reference value. The ◄ and ► keys control cursor position, and the ▲ and ▼ keys increment and decrement the digit value. To change range, place the cursor on the multiplier and use the ▲ or ▼ keys (m = ×0.001, Δ = ×1, K = ×1000, and M = ×1,000,000). With the cursor on the polarity sign, the ▲ and ▼ keys toggle polarity.
4. Press ENTER. The MATH annunciator turns on and the result of the calculation is displayed.
5. Select the measurement function.
6. Apply the signal to be measured to a switching channel input or to the front-panel inputs.
7. If you are using a switching module (REAR inputs selected), use the ◄ or ► key to select (close) the input channel. If you are using the front-panel inputs (FRONT inputs selected), you do not need close the channels.

   The result of the math calculation is displayed.
8. To disable percent, press SHIFT and then MATH. The MATH annunciator turns off.

NOTE

The result of the percent calculation may be displayed in exponential notation. For example, a displayed reading of +2.500E+03% is equivalent to 2500% (2.5K%).

Reciprocal (1/X)

You can set math operation to reciprocal to display the reciprocal of a reading.

The reciprocal is 1/X, where X is the reading. If relative offset is on, the 1/X calculation uses the input signal with the relative offset applied.

The displayed units designator for reciprocal readings is "R." This units designator cannot be changed.

The result of the 1/X calculation may be displayed in exponential notation. For example, a displayed reading of +2.500E+03 R is equivalent to 2500 R.

If you are using relative offset, the input signal with relative offset applied is used by the reciprocal calculation.

For example, assume the normal displayed reading is 002.5000 Ω. The reciprocal of resistance is conductance. When the reciprocal math function is enabled, the following conductance reading is 0.400000 R.

Reciprocal (1/X) configuration

To set the reciprocal from the front panel:

1. Press SHIFT and then MATH to display the math menu.
2. Press the RANGE ▲ or ▼ key to display 1/X and press ENTER. The MATH annunciator turns on and the result of the calculation is displayed.
3. Select the measurement function.
4. Apply the signal to be measured to a switching channel input or to the front-panel inputs.
5. If you are using a switching module (REAR inputs selected), use the ◄ or ► key to select (close) the input channel. If you are using the front-panel inputs (FRONT inputs selected), you do not need close the channels.

   The result of the math calculation is displayed.
6. To disable 1/X, press SHIFT and then MATH. The MATH annunciator turns off.

Ratio and channel average

With a switching module installed in the 2750, the ratio or average of two channels can be calculated and displayed. The ratio calculation can be done on the DCV function, and the channel average calculation can be done on the DCV and TEMP (thermocouples only) functions.

Ratio and channel average are calculated as follows:

$$\text{Ratio} = \frac{\text{Channel A}}{\text{Channel B}}$$

$$\text{Channel average} = \frac{\text{Channel A} + \text{Channel B}}{2}$$

Where:

- Channel A is the selected (closed) channel.
- Channel B is the paired channel for the installed switching module.
- Ratio and Channel Average are the displayed results of the respective calculations.

Paired channels are used for ratio and channel average. For example, the Model 7700 switching module has 20 channels that can use ratio and channel average. The primary channels (1 through 10) are linked to the paired channels (11 through 20). Channel 1 is paired to channel 11, channel 2 is paired to channel 12, and so on.

When ratio or channel average is enabled, the 2750 measures the closed primary channel. It then opens the primary channel and closes and measures the paired channel. Ratio or

channel average is then calculated from the two readings and displayed. If the 2750 is configured for continuous measurements, the two-channel scan continues to repeat and refresh the display with each new calculated reading.

The ratio or channel average calculation can only be enabled if a valid switching channel is closed. If no channel is closed when you attempt to enable one of these calculations, the message `CLOSE A CHAN` is displayed to remind you to first close a valid channel.

A primary channel must be closed before you can enable ratio or channel average. If a paired channel is instead closed, message `INVALID CHAN` is displayed to indicate the settings conflict.

NOTE

Instrument operations, including Ratio or Channel Average, are performed on the input signal in a sequential manner. Refer to Signal processing sequence (on page 4-25) for details.

Basic ratio or channel average operation

NOTE

Make sure the INPUTS switch is set to the REAR position (in).

The only value function for ratio is DCV. The only valid functions for channel average are TEMP with a thermocouple and DCV.

To calculate and display the ratio or average of two channels:

1. Select and configure a valid measurement function.
2. Use the ◄ or ► key to select (close) a primary channel (101 through 110 for the Model 7700). The CLOSE key can also be used.
3. Apply one signal to the selected primary channel, and apply the other signal to the paired channel. For the Model 7700, if the closed primary channel is 101, the paired channel is 111.
4. Enable Ratio or Channel Average:
   - **Ratio:** Press SHIFT and then RATIO. The RATIO annunciator will turn on to indicate that the displayed readings are the result of the ratio calculation.
   - **Channel Average:** Press SHIFT and then CHA-AVG. The DELTA annunciator will turn on to indicate that the displayed readings are the result of the channel average calculation. To disable channel average, again press SHIFT and then CH-AVG.

To disable the calculation, do one of the following:

- Press the OPEN key. The calculation will disable and the channel will open.
- Press SHIFT and then RATIO to disable ratio, or press SHIFT and then CHA-AVG to disable channel average. The calculation will disable, but the channel will remain closed.

NOTE

The paired channel number is not displayed when it is measured. Only the primary channel is displayed during the 2-channel scan for the calculation.

Enabling ratio disables channel average and conversely, enabling channel average disables ratio.

If either of the channel readings over range ("OVRFLW"), the result of the calculation is also "OVRFLW."

When using limits with ratio or channel average, the limit values are compared to the result of the calculation and not to the individual channels.

With ratio or channel average enabled, pressing a function key will display the "EXIT RATIO" or "EXIT CHA-AVG" message to indicate that the calculation must first be disabled as explained in the above procedure.

Ratio and channel average when scanning

Ratio and channel average can be used in an advanced scan. The 2-channel scan for the calculation is performed for every primary channel that is scanned. For example, assume the Model 7700 is installed in slot 1 and is configured to perform the ratio calculation for 10 channels. When channel 101 is scanned, measurements are performed on channels 101 and on its paired channel (111). The calculation is performed and the result is displayed. When the next channel (102) is scanned, measurements are performed on that channel (102) and on its paired channel (112). The calculation is performed and the result is displayed. This process continues for each scanned channel.

When an advanced scan is configured, each channel can have its own setup. That is, one or more channels can use ratio, and other channels can use channel average.

Details to configure and run a scan are provided in Switching and scanning (on page 3-1).

Advanced scan configuration notes

When a calculation (ratio or channel average) is enabled for a primary scan channel, the following setup actions occur:

- The calculation enables for the paired channel.
- The primary channel setup (for example, the function, range, and relative offset) is copied to the paired channel.

The filter setup for both scan channels is controlled by the primary channel.

After the calculation is enabled, the range setting can be independently set for both the primary and paired channel.

Before the calculation is enabled, the relative offset can be independently set for both the primary and paired channel. In general, set up relative offset from the normal measurement state, then go into the advanced menu and enable relative offset for the primary and/or paired channel. See Relative offset (on page 4-12) for details.

Settings such as NPLC, aperture, bandwidth, and offset compensation are ignored on the paired channel. These settings are controlled by the primary channel.

dB

Expressing DC or AC voltage in dB makes it possible to compress a large range of measurements into a much smaller scope. The relationship between dB and voltage is defined by the following equation:

$$|\,dB\,| = 20\log \frac{V_{IN}}{V_{REF}}$$

Where:

- V_{IN} is the DC or AC input signal.
- V_{REF} is the specified voltage reference level.

The instrument reads 0 dB when the reference voltage level is applied to the input.

If a relative value is in effect when dB is selected, the value is converted to dB, then the relative offset is applied to dB. If the relative offset is applied after dB has been selected, dB has relative offset applied to it.

NOTE

The largest negative value of dB is −160 dB. This accommodates a ratio of V_{IN} = 1 µV and V_{REF} = 1000 V.

Remote programming must be used to configure the 2750 for dB measurements. It cannot be configured from the front panel.

dB and scanning

Typically, a scan using dB is configured and run using remote programming. However, once dB is selected using remote programming, a simple dB scan can be configured and run from the front panel. When the simple scan is configured, it uses the dB measurement setup for each channel in the scan.

Details on configuring and running a scan are provided in Switching and scanning (on page 3-1).

Low-level voltage measurement considerations

Low-level voltage measurements can be adversely affected by noise or other unwanted signals that can make it difficult to get accurate voltage readings. Some of the phenomena that can cause unwanted noise include thermoelectric effects (thermocouple action), source resistance noise, magnetic fields, and radio frequency interference. The following paragraphs discuss the most important of these effects and ways to minimize them.

NOTE

For comprehensive information on low-level measurements, see the *Low Level Measurements Handbook*, which is available from tek.com/keithley.

Magnetic fields

When a conductor loop cuts through magnetic lines of force, a very small current is generated. This phenomenon can cause unwanted signals to occur in the test leads of a test system. If the conductor has sufficient length or cross-sectional area, even weak magnetic fields can create signals that affect low-level measurements.

To reduce these effects:

- Reduce the lengths of the connecting cables.
- Minimize the exposed circuit area.
- Change the orientation of the leads or cables.
- Minimize cable loop area or introduce cable twisting.

In extreme cases, you may require magnetic shielding. Special metal with high permeability at low flux densities (such as mu metal) is effective at reducing these effects.

Even when the conductor is stationary, you may have problems with magnetically-induced signals. Fields can be produced by sources such as the AC power line voltage and large inductors, such as power transformers. Keep the 2750 voltage source and connecting cables away from these potential noise sources.

Radio frequency interference

Radio frequency interference (RFI) is a general term used to describe electromagnetic interference over a wide range of frequencies across the spectrum. RFI creates problems at low signal levels, but it can also affect measurements at high levels if the fields are of sufficient magnitude.

RFI can be caused by steady-state sources, such as radio or TV signals, or some types of electronic equipment, such as microprocessors and high-speed digital circuits. It can also result from impulse sources, as in the case of arcing in high-voltage environments. The effect on the measurement can be considerable if enough of the unwanted signal is present.

You can minimize RFI in several ways:

- Keep the 2750 voltage source and signal leads away from RFI sources.
- Shield the instrument, signal leads, sources, and other measuring instruments.
- In extreme cases, a specially constructed screen room may be required to sufficiently attenuate the RFI signal.

In some situations, the 2750 digital filter may help to reduce RFI effects, but additional external filtering may be required. Filtering may have detrimental effects, such as increased settling time on the signal.

Remote DMM measurements using the front-panel inputs

For remote programming, the instrument is typically used in a non-continuous measurement mode. In this mode, the user uses remote commands to specify the number of measurements to perform. *RST defaults place the instrument in a non-continuous measurement mode. Most of the other settings for factory and *RST defaults are the same.

For remote programming, the following command is used to select the function.

NOTE

Items in brackets ([ ]) are optional and do not need to be included. Upper-case characters are required. Lower-case characters are optional and need not be included.

```
[SENSe[1]]:FUNCtion <func>    '  Select measurement function.
```

Where <func> can be:

```
'VOLTage[:DC]'   DCV
'VOLTage:AC'   ACV
'CURRent[:DC]'   DCI
'CURRent:AC'   ACI
'RESistance'   Ω2
'FRESistance'   Ω4
'FREQuency'   FREQ
'PERiod'   PERIOD
'TEMPerature'   TEMP
```

Each function can have its own setup configuration, such as a different range, digits, and speed. For example, the following command words select range and digits:

```
RANGe[:UPPer] <n>      ' Specify expected reading.
RANGe:AUTO <b>       ' Enable (ON) or disable (OFF) autorange.
DIGits      ' Set display resolution; 3.5, 4.5, 5.5 or 6.5 (digits).
```

The following examples demonstrate how to include the function name in the command string for configuration commands.

```
VOLT:RANG 10      '  Select 10 V range for DCV.
RES:RANG:AUTO ON      '  Enable autorange for Ω2.
CURR:DIG 4.5      '  Set DCI for 4½ digit resolution.
```

Exercise: Basic DMM measurements using remote commands

The following code example measures ACV on the 10 V range and stores 15 readings in the buffer.

These commands:

- Clear the buffer. To avoid problems with remote programming, it is good practice to routinely clear the buffer (TRAC:CLE) at the beginning of a program that performs multiple measurements (SAMP:COUN >1). Restoring *RST or factory defaults does not clear the buffer.
- Restore the defaults. This example uses *RST, which puts the instrument in non-continuous measurement mode. Factory defaults puts the instrument in continuous measurement mode.
- Select the AC voltage function.
- Select the 10 V range.
- Select the buffer to store 15 readings.
- Recall the readings from the buffer.

Statistics for buffer readings are also stored in the buffer. `CALC1:DATA?` only returns the readings that were stored. It does not return buffer statistics. You can use `CALC2` commands to calculate and return buffer statistics.

To make a DMM measurement, send the following code:

```
TRAC:CLE
*RST
FUNC 'VOLT:AC'
VOLT:AC:RANG 10
SAMP:COUN 15
READ?
CALC1:DATA?
```

Signal processing sequence

The instrument operations, such as filters, calculations, relative offset, and limits, are performed on the input signal in a sequential manner. The following section provides information on the sequence for front-panel and remote operation.

The instrument operations that are in the sequence include:

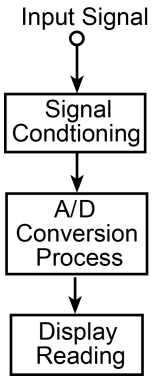
- Offset-compensated ohms
- Filter
- Output trigger pulse (VMC)
- Relative offset
- Math calculations
- Limits
- Buffer
- Ratio or channel average

Basic signal processing

The signal is applied to the multimeter input through front-panel input terminals or a switching module. When a channel is closed or scanned, the signal connected to that channel (or channel-pair for 4-wire measurements) is connected to the input.

The following figure shows the basic processing sequence of an input signal. With all features of the 2750 disabled, the input signal is conditioned and measured (A/D conversion process). The reading is then displayed on the 2750.

Figure 22: Basic signal processing



Based on the selected measurement function and range, signal conditioning transforms the input signal into a dc voltage that is applied to the A/D converter.

The A/D conversion process measures the dc signal voltage and internal voltages that correspond to offsets (zero) and amplifier gains. For thermocouple temperature measurements using a switching module that has an internal reference junction (for example, the Model 7700), the internal temperature is also measured. These measurements are used in an algorithm to accurately calculate the reading of the input signal. The voltage, current, resistance, frequency (or period), or temperature reading is then displayed by the 2750.

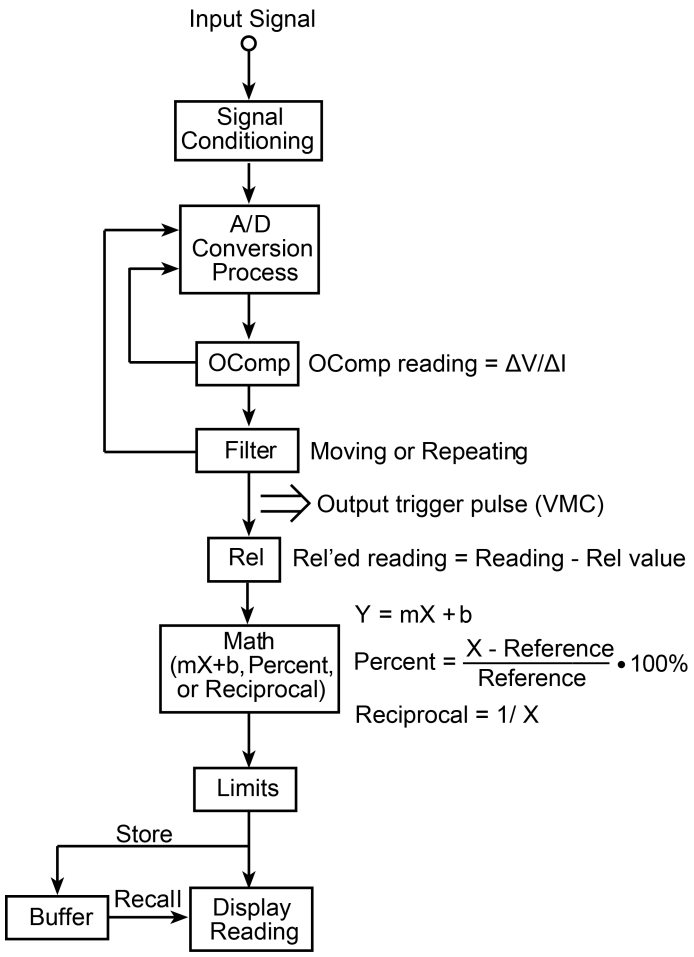
NOTE

The multiple measurement process used by the A/D converter is known as autozeroing. It can be disabled to increase speed (only the signal is measured). However, stability and accuracy will be affected over time and changes in temperature. Refer to Autozero (on page 4-1) for additional information.

Signal processing using instrument features

The following figure shows the processing sequence for an input signal with instrument features enabled. If a feature is not enabled, the reading skips that step and goes to the next enabled feature or to the display.

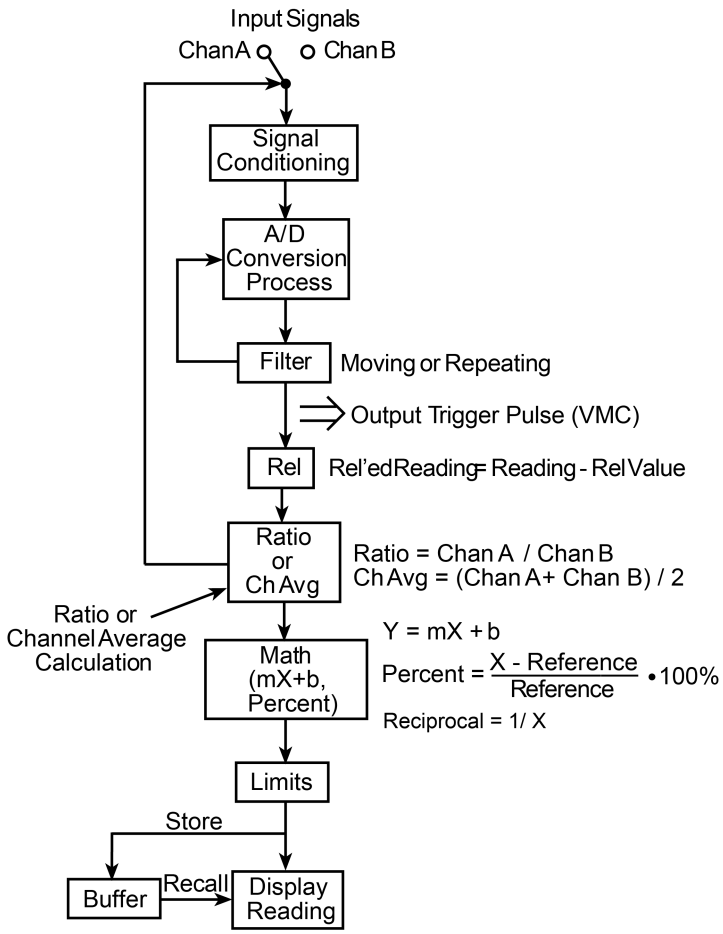
Figure 23: Signal processing using instrument features



Signal processing using Ratio or Ch Avg

When a switching module is installed, the ratio or average of two channels can be calculated. The following figure shows where Ratio or Ch Avg is calculated in the signal processing sequence.

Figure 24: Signal processing using Ratio or Channel Average



With a channel closed and Ratio or Ch Avg enabled, the reading that is applied to the Ratio or Ch Avg block in the flowchart is used as the Chan A value for the calculation. The paired channel then closes, and that reading is used as the Chan B value for the calculation. Ratio or Ch Avg is then calculated as shown in the previous figure.

The result of Ratio or Ch Avg can be used by an enabled math operation.

NOTE

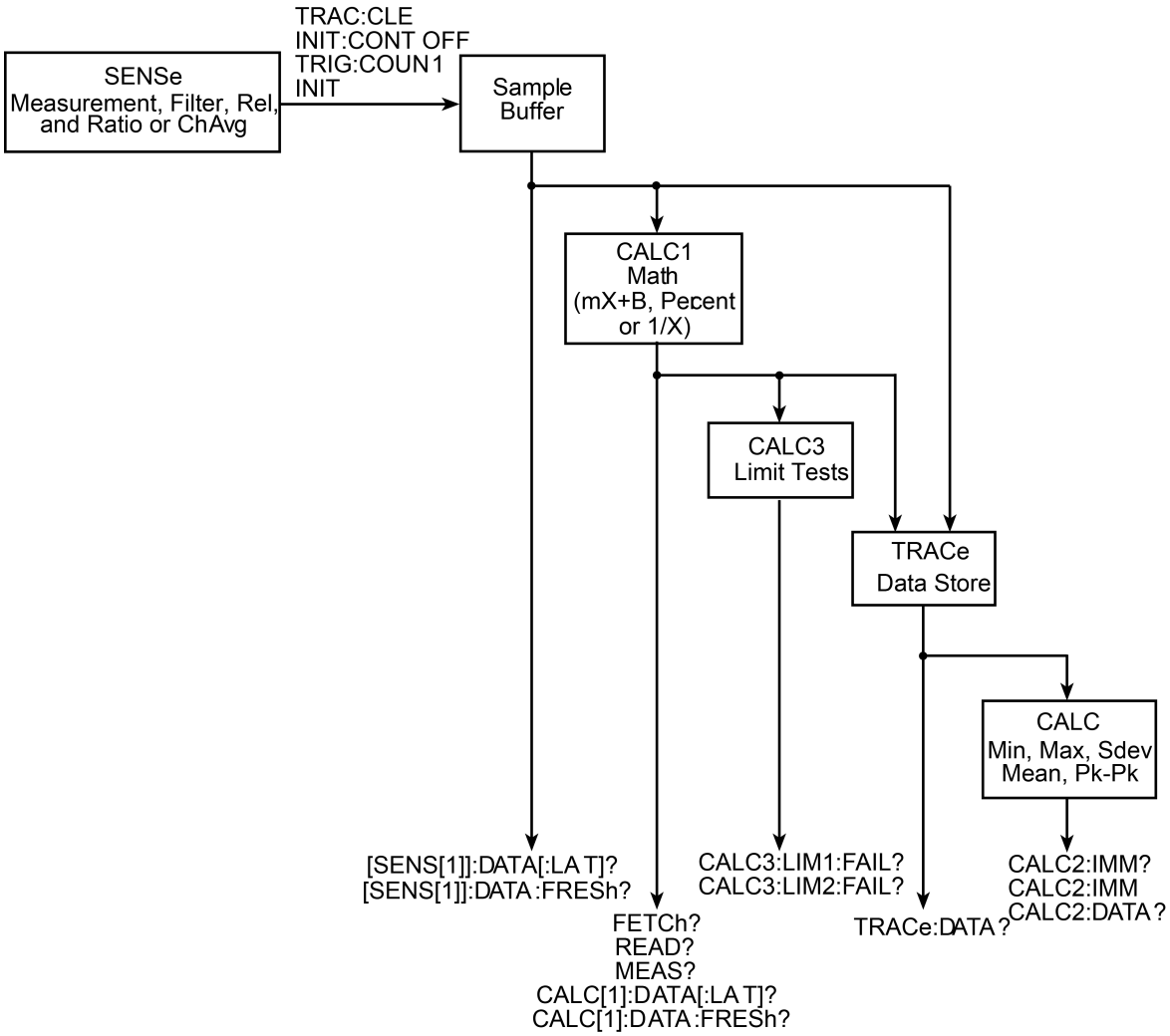
For details on these calculations, refer to Ratio and channel average (on page 4-18).

Data flow (remote operation)

Remote operation can be used with triggering configured to perform a specified number of measurements and then stop. The read commands (SENS:DATA?, FETCh?, READ?, MEAS?, CALC2:DATA?, TRACe:DATA?, and CALC1:DATA?) return the data arrays acquired during the measurement cycle.

The data flow for this triggering configuration is summarized in the following figure. Refer to this block diagram for the following discussion.

Figure 25: Data flow for remote operation



For the following discussion, a "data array" is defined as the group of data elements that are included with each measured reading. Each data array includes the reading and the channel, reading number, units, timestamp, and limits result. Refer to :FORMat:ELEMents (on page 9-64) for details.

For example, assume the selected data elements to be returned by a read command include the reading, units designator, and reading number. Now assume a 1 V dc input and the READ? command is sent to trigger two readings and return the two data arrays. The two returned data arrays look like this:

```
+1.00000000E+00VDC, +00000RDNG#, +1.00000000E+00VDC. +00001RDNG#
          (Data Array #1)     (Data Array #2)
```

Continuous measurement mode

With continuous initiation enabled (`INIT:CONT ON`), the instrument continuously performs and displays measurements. Data flow is the same except that only one data array is stored in the sample buffer at a time. The single data array is then fed to the other enabled data flow blocks. When the next measurement occurs, that data array overwrites the previous data array in the sample buffer. The new data is then fed to the other data flow blocks. When SENS:DATA?, FETCh?, READ?, or CALC1:DATA? is sent, the latest data array will be returned.

NOTE

The READ? command tries to perform an INIT operation. This causes error −213, Init ignored, since the instrument is already initiating measurements.

NOTE

When the instrument is not in continuous measurement mode, the INIT:CONT ON command can be sent to enable continuous initiation. However, if the sample count is >1, error −221, setting conflict, will occur. Set the sample count to 1 (SAMP:COUN 1) and then send INIT:CONT ON.

SENSe and sample buffer

The TRACe:CLEar command clears the data store, INITiate:CONTinuous OFF command disables continuous initiation, and TRIGger:COUNt 1 configures the instrument to perform one measurement cycle. The INIT command can then be used to initiate the measurement cycle. When the INIT command is sent, the programmed number of measurements (set by the SAMPle:COUNt command) are performed and the respective data is temporarily stored in the sample buffer.

For example, if 20 measurements were performed (SAMP:COUN 20), then 20 data arrays will be stored in the sample buffer. Data from this buffer is then routed to other enabled data flow blocks. The data in the sample buffer remains there until data from another measurement cycle overwrites the buffer.

NOTE

The trigger count (TRIG:COUN) determines how many measurement cycles are performed. However, only the data arrays for the last measurement cycle end up in the sample buffer. For example, assume TRIG:COUN 2 and SAMP:COUN 20. The first measurement cycle stores 20 data arrays in the sample buffer. The second measurement cycle then overwrites the 20 data arrays in the sample buffer.

Scanning

For remote operation, scanning is normally performed with continuous initiation disabled (`INIT:CONT OFF`). The sample count (`SAMP:COUNt`) specifies the number of channels to scan and store in the buffers (sample buffer and data store), and the trigger count (`TRIG:COUNt`) specifies the number of scans to perform. If the trigger count is >1, the data for each subsequent scan overwrites the data stored in the sample buffer and data store.

Once the scan is properly configured, `INIT` or `READ?` starts the scan. `READ?` also returns the scanned readings (data arrays) from the sample buffer or the CALC1 block if Math is enabled. `FETCh?` will not start a scan, but it will return the readings already stored.

While the scan is in process, `SENS:DATA?` and `CALC1:DATA?` commands can be used to return the latest data array. When used after the scan is finished, they return the data array for the last stored reading.

Section 5

Measurement considerations

In this section:

Introduction 5-1
Thermoelectric potentials 5-1
Thermoelectric generation..... 5-2
Source resistance noise 5-4
Magnetic fields 5-5
Radio frequency interference 5-6
Ground loops..... 5-6
Shielding 5-8
Meter loading 5-9

Introduction

Low-level voltage measurements made using the 2750 can be adversely affected by various types of noise or other unwanted signals that can make it difficult to obtain accurate voltage readings. Some of the phenomena that can cause unwanted noise include thermoelectric effects (thermocouple action), source resistance noise, magnetic fields, and radio frequency interference. This section describes the most important of these effects and ways to minimize them.

NOTE

For comprehensive information on low-level measurements, see the *Low Level Measurements Handbook*, which is available from tek.com/keithley.

Thermoelectric potentials

Thermoelectric potentials (thermal EMFs) are small electric potentials generated by differences in temperature at the junction of dissimilar metals. The following paragraphs discuss how such thermals are generated and ways to minimize their effects.

Thermoelectric coefficients

The following table shows the magnitude of thermoelectric EMFs that are generated for different materials. Best results are obtained with clean copper-to-copper connections.

Material thermoelectric coefficients	
Material	**Thermoelectric potential**
Copper-to-copper	0.2 µV/°C
Copper-to-silver	0.3 µV/°C
Copper-to-gold	0.3 µV/°C
Copper-to-cadmium/tin	0.3 µV/°C
Copper-to-lead/tin	1 µV/°C to 3 µV/°C
Copper-to-Kovar®	40 µV/°C to 75 µV/°C
Copper-to-silicon	400 µV/°C
Copper-to-copper oxide	1000 µV/°C

Thermoelectric generation

The following figure represents how thermal EMFs are generated. The test leads are made of the A material and the source under test is the B material. The temperatures between the junctions are shown as T1 and T2. To determine the thermal EMF generated, the following relationship may be used:

$E_T = Q_{AB} (T_1 - T_2)$

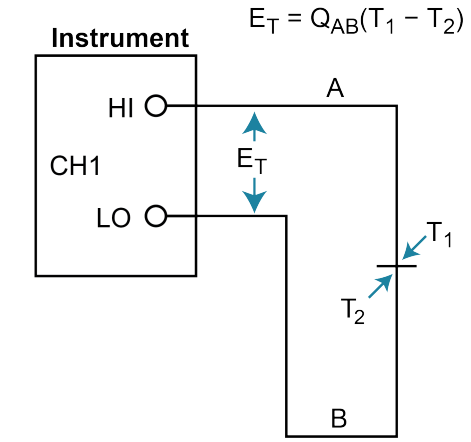
Where:

- E_T = Generated thermal EMF
- Q_{AB} = Thermoelectric coefficient of material A with respect to material B (µV/°C)
- T_1 = Temperature of B junction (°C or K)
- T_2 = Temperature of A junction (°C or K)

In the unlikely event that the two junction temperatures are identical, no thermal EMFs are generated. More often, the two junction temperatures differ and considerable thermal EMFs are generated.

A typical test setup may have several copper-to-copper junctions. Each junction can have a thermoelectric coefficient as high as 0.2 µV/°C. Since the two materials frequently have a several degree temperature differential, several microvolts of thermal potentials can be generated even if reasonable precautions are taken.

Figure 26: Thermal EMF generation



Minimizing thermal EMFs

To minimize thermal EMFs, use only copper wires, lugs, and test leads for the entire test setup. Also, it is imperative that all connecting surfaces are kept clean and free of oxides. As noted in Thermoelectric generation (on page 5-2), copper-to-copper oxide junctions can result in thermal EMFs as high as 1 mV/°C.

Even when low-thermal cables and connections are used, thermal EMFs can still be a problem in some cases. It is especially important to keep the materials that form the junction at the same temperature. One way to minimize thermal problems is to keep the two junctions close together. Also, keep all junctions away from air currents; in some cases, it may be necessary to thermally insulate sensitive junctions to minimize temperature variations. When a copper-to-copper connection is made, sufficient pressure must be applied to ensure the connection is gastight to prevent future oxidation.

In some cases, connecting the two thermal junctions together with good thermal contact to a common heat sink may be required. Unfortunately, most good electrical insulators are poor conductors of heat. In cases where such low thermal conductivity may be a problem, special insulators that combine high electrical insulating properties with high thermal conductivity may be used. Some examples of these materials include hard anodized aluminum, sapphire, and diamond.

Canceling residual thermal offsets

Even if all reasonable precautions are taken, some residual thermal offsets may still be present. These offsets can be minimized by using the 2750 relative offset feature to cancel them out.

For more information on using relative offset, refer to Relative offset (on page 4-12).

To use relative offset to cancel the residual thermal offsets:

1. Place the instrument on the 3 mV range.
2. Disconnect the cable from the source to avoid shorting out the source.
3. Short the end of the connecting cable nearest the measured source.
4. Allow the reading to settle.
5. Press the **REL** key.
6. Select the appropriate range and make your measurement as usual.

Source resistance noise

Noise present in the source resistance is often the limiting factor in the ultimate resolution and accuracy of 2750 measurements. The following topics discuss the generation of Johnson noise and ways to minimize such noise.

Johnson noise equation

The amount of noise present in a given resistance is defined by the Johnson noise equation as follows:

$$E_{RMS} = \sqrt{4kTRF}$$

Where:

- E_{RMS} = The RMS value of the noise voltage
- k = Boltzmann constant (1.38×10^{-23} joules/Kelvin)
- T = Temperature (K)
- R = Source resistance (ohms)
- F = Noise bandwidth (Hz)

At a room temperature of 293 K (20 °C), the equation simplifies to:

$$E_{RMS} = 1.27 \times 10^{-10} \sqrt{RF}$$

Since the peak-to-peak noise is five times the RMS value 99% of the time, the peak-to-peak noise can be equated as follows:

$$E_{p\text{-}p} = 6.35 \times 10^{-10} \sqrt{RF}$$

For example, with a source resistance of 10 kΩ, the noise over a 0.5 Hz bandwidth at room temperature is:

$$E_{p\text{-}p} = 6.35 \times 10^{-10} \sqrt{(10 \times 10^{3})(0.5)}$$
$$E_{p\text{-}p} = 45 \text{ nV}$$

Minimizing source resistance noise

You can reduce the noise in several ways:

- Reduce the source resistance
- Lower the temperature
- Narrow the bandwidth

Of these, lowering the resistance is the least practical because the signal voltage is reduced more than the noise. For example, decreasing the resistance of a current shunt by a factor of 100 also reduces the voltage by a factor of 100, but the noise is decreased only by a factor of 10.

Cooling the source is often the only practical method available to reduce noise. Again, however, the available reduction is not as large as it might seem because the reduction is related to the square root of the change in temperature. For example, to cut the noise in half, the temperature must be decreased from 293 K to 73.25 K.

Magnetic fields

When a conductor loop cuts through magnetic lines of force, a very small current is generated. This phenomenon can cause unwanted signals to occur in the test leads of a test system. If the conductor has sufficient length or cross-sectional area, even weak magnetic fields can create signals that affect low-level measurements.

To reduce these effects:

- Reduce the lengths of the connecting cables.
- Minimize the exposed circuit area.
- Change the orientation of the leads or cables.
- Minimize cable loop area or introduce cable twisting.

In extreme cases, you may require magnetic shielding. Special metal with high permeability at low flux densities (such as mu metal) is effective at reducing these effects.

Even when the conductor is stationary, you may have problems with magnetically-induced signals. Fields can be produced by sources such as the AC power line voltage and large inductors, such as power transformers. Keep the 2750 voltage source and connecting cables away from these potential noise sources.

Radio frequency interference

Radio frequency interference (RFI) is a general term used to describe electromagnetic interference over a wide range of frequencies. RFI can be particularly troublesome at low signal levels, but it can also affect measurements at high levels if the fields are of sufficient magnitude.

Some causes of RFI are:

- Steady-state sources, such as radio or TV signals.
- Some types of electronic equipment, such as microprocessors and high-speed digital circuits.
- Impulse sources, such as arcing in high-voltage environments.

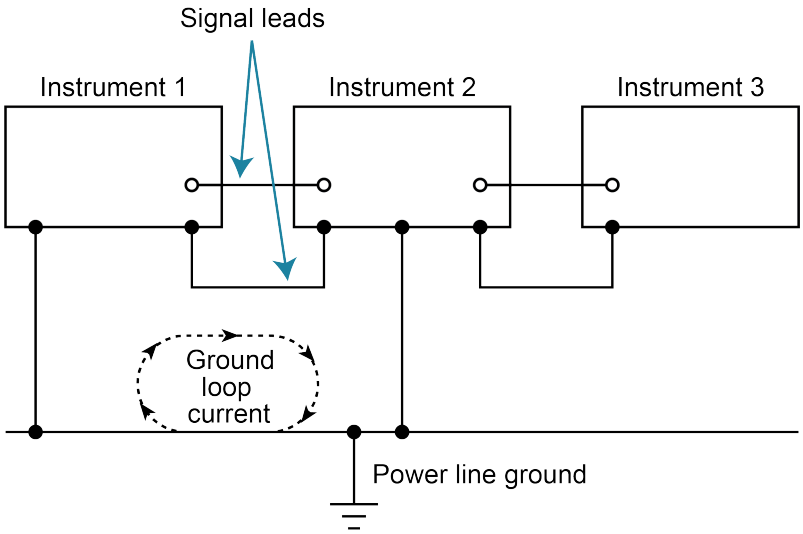
RFI can be minimized in several ways. The most obvious method is to keep the 2750 voltage source and signal leads as far away from the RFI source as possible. Additional shielding of the instrument, signal leads, sources, and other measuring instruments will often reduce RFI to an acceptable level. In extreme cases, a specially-constructed screen room may be required to sufficiently attenuate the RFI signal.

The 2750 digital filter may help to reduce RFI effects in some situations. In other cases, additional external filtering may also be required. However, filtering may have detrimental effects on the signal, such as increasing settling time.

Ground loops

When two or more instruments are connected, avoid unwanted signals caused by ground loops. Ground loops usually occur when sensitive instrumentation is connected to other instrumentation with more than one signal return path, such as power line ground. As shown in the following figure, the resulting ground loop causes current to flow through the instrument LO signal leads and then back through power line ground. This circulating current develops a small voltage between the LO terminals of the two instruments. This voltage is added to the source voltage, affecting the accuracy of the measurement.

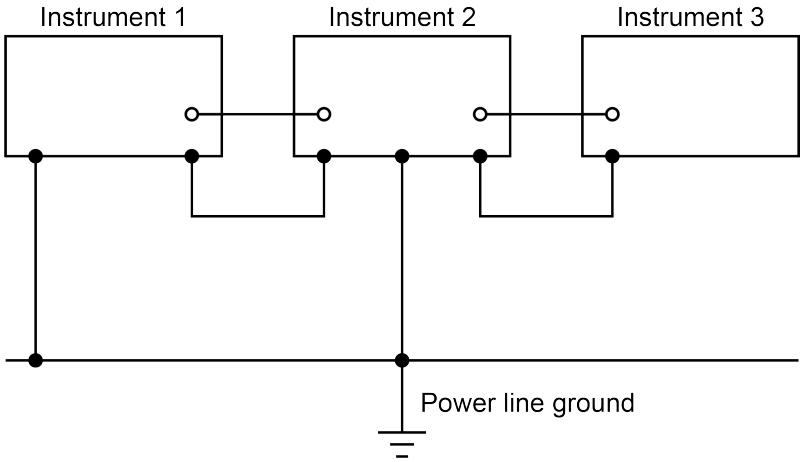
Figure 27: Power-line ground loops



The following figure shows how to connect several instruments together to eliminate this type of ground loop problem. Here, only one instrument is connected to power line ground.

Ground loops are not normally a problem with instruments like the 2750 that have isolated LO terminals. However, other instruments in the test setup may not have isolated terminals. Consult the documentation for each instrument in the test setup.

Figure 28: Eliminating ground loops



Shielding

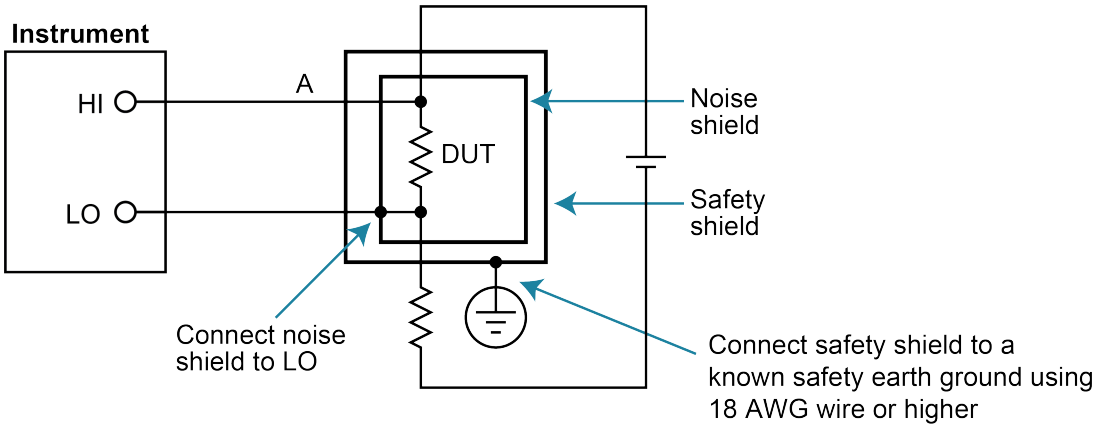
Proper shielding of all signal paths and sources being measured is important to minimize noise pickup in virtually any low-level measurement situation. Otherwise, interference from noise sources such as line frequency and RF fields can seriously corrupt measurements.

To minimize noise, a closed metal shield surrounding the source may be necessary, as shown in the following figure. In most cases, this shield should be connected to input LO. In some situations, better noise performance may result with the shield connected to chassis ground. The metal shield should enclose the test circuit.

WARNING

Do not float input LO more than 30 V_{RMS}, 42.4 V_{PEAK} above earth ground with an exposed shield connected to input LO. To avoid a possible shock hazard, surround the LO shield with a second safety shield that is insulated from the inner shield. Connect this safety shield to safety earth ground using #18 AWG minimum wire before use. Failure to adhere to these guidelines can result in personal injury or death due to electric shock.

Figure 29: Shielding example

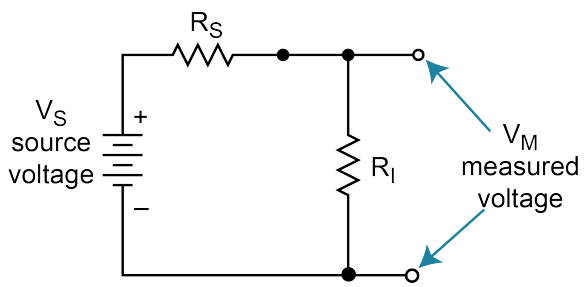


Meter loading

Loading of the voltage source by the 2750 becomes a consideration for high-source resistance values. As the source resistance increases, the error caused by meter loading increases.

The following figure shows the method used to determine the percent error due to meter loading. The voltage source, V_S, has a source resistance, R_S. The input resistance of the 2750 is R_I. The voltage measured by the nanovoltmeter is V_M.

Figure 30: Meter loading



The voltage actually measured by the meter is attenuated by the voltage divider action of R_S and R_I. It can be calculated as follows:

$$V_M = \frac{V_S R_L}{R_I + R_S}$$

This relationship can be modified to directly compute for percent error:

$$\text{Percent error} = \frac{100 R_S}{R_I + R_S}$$

From this equation, the input resistance of the 2750 must be at least 999 times the value of source resistance to keep loading error within 0.1%.

Section 6

Limits

In this section:

Introduction 6-1
Limits..... 6-1
Basic limits operation 6-4
Digital I/O 6-5
Digital outputs 6-5
Application: Sorting resistors..... 6-11

Introduction

This sections describes how to perform limit tests on measured readings and how the five digital outputs respond to the results of limit tests.

With Limits enabled, the reading is tested against two sets of high and low limits. Along with the displayed reading, annunciators and messages are used to indicate the result of the limits testing.

It also includes an example application for sorting resistors. This application tests the tolerances of 100 Ω resistors and provides the digital output response to the pass/fail combinations of the limit tests.

Limits

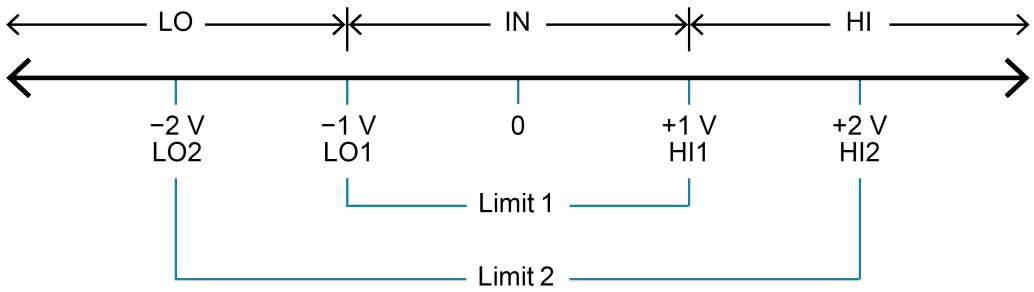
Limit 1 and Limit 2 operations set and control the values that determine the HI, IN, or LO status of subsequent measurements. The limit test is performed on the result of an enabled relative offset, math, ratio, or channel average operation. Limits are tested after other operations are applied and before readings are stored in the buffer. Refer to Signal processing using instrument features (on page 4-27) for detail.

The HIGH/IN/LOW status indication applies to the first limit (limit 1 or limit 2) that fails. The following figure illustrates the factory default values of the limits.

Limit 1: HI1 = +1 V and LO1 = −1 V

Limit 2: HI2 = +2 V and LO2 = −2 V

Figure 31: Default limits



When a reading is within both limits, the message `IN` is displayed. When the reading is high or low, the HIGH or LOW annunciator turns on, and the number `1` or `2` replaces the `IN` message. `1` indicates that Limit 1 failed. `2` indicates that Limit 2 failed. If the reading is outside both limits, `1` is displayed.

For the limits shown in the figure above, a reading of 1.5 V is outside Limit 1 (which is the primary limit). Therefore, the message `1` is displayed and the HIGH annunciator turns on. For a reading of +2.5 V, which is outside both Limit 1 and Limit 2, the same status indication (HIGH, `1`) is displayed since Limit 1 takes precedence.

The limit value for Limit 2 does not have to exceed the Limit 1 value. For example, Limit 2 can be set to ±1 V and Limit 1 can be set to ±2 V. In this case, Limit 2 fails before Limit 1.

A beeper is also available for limit testing. The beeper has the following options:

- **NEVER:** Disables the beeper.
- **OUTSIDE:** The beeper sounds when the reading is outside (HIGH or LOW) Limit 1. In the figure above, a 1.5 V reading is outside (HIGH) Limit 1, so the beeper sounds.
- **INSIDE:** The beeper sounds when the reading is inside Limit 1 or Limit 2. If the reading is inside Limit 1, the beeper sounds raspy. If the reading is outside Limit 1 but inside Limit 2, the beeper sounds at a lower pitch. The beeper does not sound for readings that are outside both limits. For the limits shown in the figure above, a 0.5 V reading sounds the beeper at its normal pitch, a 0.5 V reading sounds the beeper at a lower pitch, and a 2.5 V reading does not sound the beeper.

NOTE

Limits cannot be used with the CONT function.

When a switching module channel is closed, the message `I` replaces the message `IN` to indicate that the reading is inside both Limit 1 and Limit 2.

For limit test readings that get stored in the buffer, the limits status indicators are displayed for each recalled reading.

When a limit test reading is returned using remote programming, limit test status can be included with the reading. See FORMat subsystem (on page 9-63) for details.

When using Limits with Ratio or Ch Avg, the limit values are compared to the result of the calculation and not to the individual channels.

Tips to use Limit 2 test

Limits 1 < Limits 2: When the set limits for Limit 1 are less than the limits for Limit 2, use the INSIDE beeper.

Limits 1 > Limits 2: When the set limits for Limit 1 are greater than the limits for Limit 2, use the OUTSIDE beeper.

When the reading is between Limit 1 and Limit 2, the beeper will sound raspy.

Overflow readings

A reading that exceeds the present measurement range causes the `OVRFLW` message to display. The `IN`, `1`, and `2` messages are not displayed while in the overflow condition. The HIGH annunciator turns on to indicate an out of limits reading.

The LOW annunciator is not used for an overflow reading. An overflow reading is interpreted by the 2750 as a positive reading, even if the input signal is negative.

Scanning and limits

When a simple scan is configured, the present limit values and state apply to all channels in the scan. When an advanced scan is configured, each channel can have its own limits configuration. Details to configure and run a scan are provided in Switching and scanning (on page 3-1).

For remote programming, the `<clist>` parameter is used to configure channels for a scan.

Basic limits operation

The limits configuration is the same for all functions. For example, a reading limit that is set to 1 equates to 1 V for a voltage function, 1 A for a current function, and 1 Ω for a resistance function.

Setting limits

To set the limits from the front panel:

1. Press **SHIFT** and then **LIMITS** to display the high limit for Limit 1 (HI1).
2. Use ◄, ►, ▲, and ▼ to key in the HI1 limit and press **ENTER**.
3. When editing a reading, use the range designator (▲ , K, or M) as a multiplier. With the cursor on the range designator, each press of ▲ or ▼ increases or decreases the reading by a factor of 10.
4. Key in the low limit for Limit 1 (LO1) and press **ENTER**.
5. Key in the high limit for Limit 2 (HI2) and press **ENTER**.
6. Key in the low limit for Limit 2 (LO2) and press **ENTER**. The instrument returns the normal measurement state.

Beeper settings

Remote programming cannot be used to set the beeper for the limits. It can only be set from the front panel. You can disable the beeper using remote commands. Refer to :SYSTem:BEEPer[:STATe] (on page 9-175) for details.

To configure the beeper:

1. Press SHIFT and then OUTPUT.
2. Use the ▲ or ▼ key to display the present beeper (BEEP) setting: NEVER, INSIDE, or OUTSIDE.
3. Press ► to position the cursor on the present beeper setting, use the ▲ or ▼ key to display the setting, and press ENTER. The instrument will return to the normal measurement state.

Enabling and disabling limits

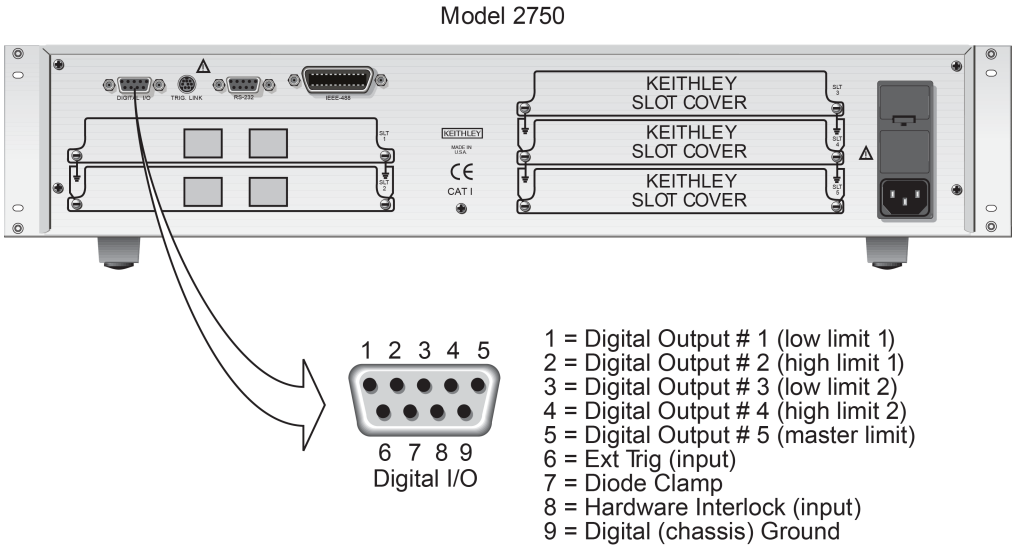
To change the state of the limits:

1. Press SHIFT and then ON/OFF to display the present state (off or on) of limits. To enable limits, use the ∆ or ∇ key to display "LIMITS: ON" and press ENTER.
2. To disable limits, press SHIFT and then ON/OFF, select "LIMITS: OFF" and press ENTER.

Digital I/O

The digital I/O port of the 2750 is accessed at a DB-9 plug on the rear panel. The connector location and pin designations are shown in the following figure.

Figure 32: Digital I/O port



Digital input (trigger link input)

When enabled, the Trigger In (pin 6) and Digital Ground (pin 9) can be used as the trigger link input for external triggering. Pin 6 is physically connected to the input line (pin 2) of the TRIG LINK connector.

Pin 8 of the digital I/O is used to enable or disable Trigger In. Trigger In is enabled by leaving pin 8 open, or pulling it high (+5 V). Trigger In is disabled by setting pin 8 low (0 V).

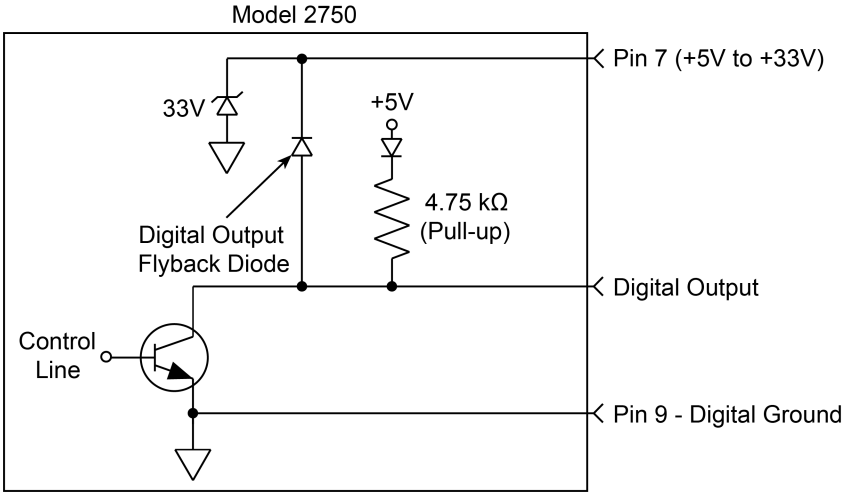
NOTE

External triggering is described in External triggering (on page 7-4).

Digital outputs

The digital I/O port has five digital outputs. Each digital output can be used as a sink to control devices (such as relays) or as a source to provide input to external logic (TTL or CMOS) circuitry. The simplified schematic for the digital outputs is shown in the following figure. This illustration shows the schematic for one digital output. All five digital output circuits are identical.

Figure 33: Digital I/O port simplified schematic



The five digital output lines (pins 1 through 5) are controlled by limit operations. Each of these five outputs correspond to the following limit operations:

Digital Output 1: Low Limit 1 (LO1)

Digital Output 2: High Limit 1 (HI1)

Digital Output 3: Low Limit (LO2)

Digital Output 4: High Limit 2 (HI2)

Digital Output 5: Master Limit (logical OR of the four above limits)

When a limit (LO1, HI2, LO2, or HL2) is reached, the digital output line for that limit is pulled high or low. When a reading is within the limit, the output line is released. Digital output 5 is the logical OR of the four limits. Therefore, if any of the four limits are reached or exceeded, output 5 is pulled high or low.

NOTE

When the reading is made and a limit has been reached, there is a short delay before the digital output line is active. As measured from the output trigger (TLINK), the delay is about 10 ms when closing a channel and about 2 ms without a channel closure. For temperature readings, the delay may be up to 10 times longer because of the additional time needed for data conversion. Allow for this delay when designing test systems.

Logic sense

The selected logic sense (active high or active low) determines if an output is pulled high or low when the limit is reached. If logic sense is set high, the output line will be pulled high when the reading reaches or exceeds the limit. If logic sense is set low, the output line will be pulled low to 0 V when the reading reaches or exceeds the limit.

Pulse option

Pulse option is available for the digital outputs. When enabled, an output line pulses high or low (depending on the logic sense setting) for each reading that reaches or exceeds the limit. The factory default time duration for the pulse is 2 ms (maximum), but can be set from 0.001 to 99999.999 seconds using remote programming. Pulse time cannot be set from the front panel.

NOTE

The commands to set pulse time and enable/disable pulse output are in CALCulate3 subsystem (on page 9-50).

The pulse time does not affect measurement speed. If a subsequent in-limit reading occurs while the output line is being pulsed, the line is released immediately (pulse terminated).

Master limit latch

The master limit line is pulled high or low when one or more of the other four limits are reached or exceeded. The master limit line can be programmed to release when a reading is inside all four limits, or the master limit can be latched when a failure occurs. When latched, the master limit line will not release until operation within the trigger model returns to and passes the control source. Refer to Triggering (on page 7-1) for details on triggering.

When scanning, the latched master limit line will not release until the scan is finished and another scan is started. For example, if after testing a resistor network the master limit line did set, then the network has passed all tests.

Controlling external devices using sink mode

Each output can be operated from an external supply. The high current sink capacity of the output driver allows direct control of relays, solenoids, and lamps (no additional circuitry needed).

As shown in the figure in Digital outputs (on page 6-5), each of the digital open-collector outputs includes a built-in pull up resistor connected to +5 V. The output transistor can sink 250 mA from voltages up to +33 V. Each output channel contains a fly-back diode for protection when switching inductive loads (such as a low power solenoid or relay coils). To

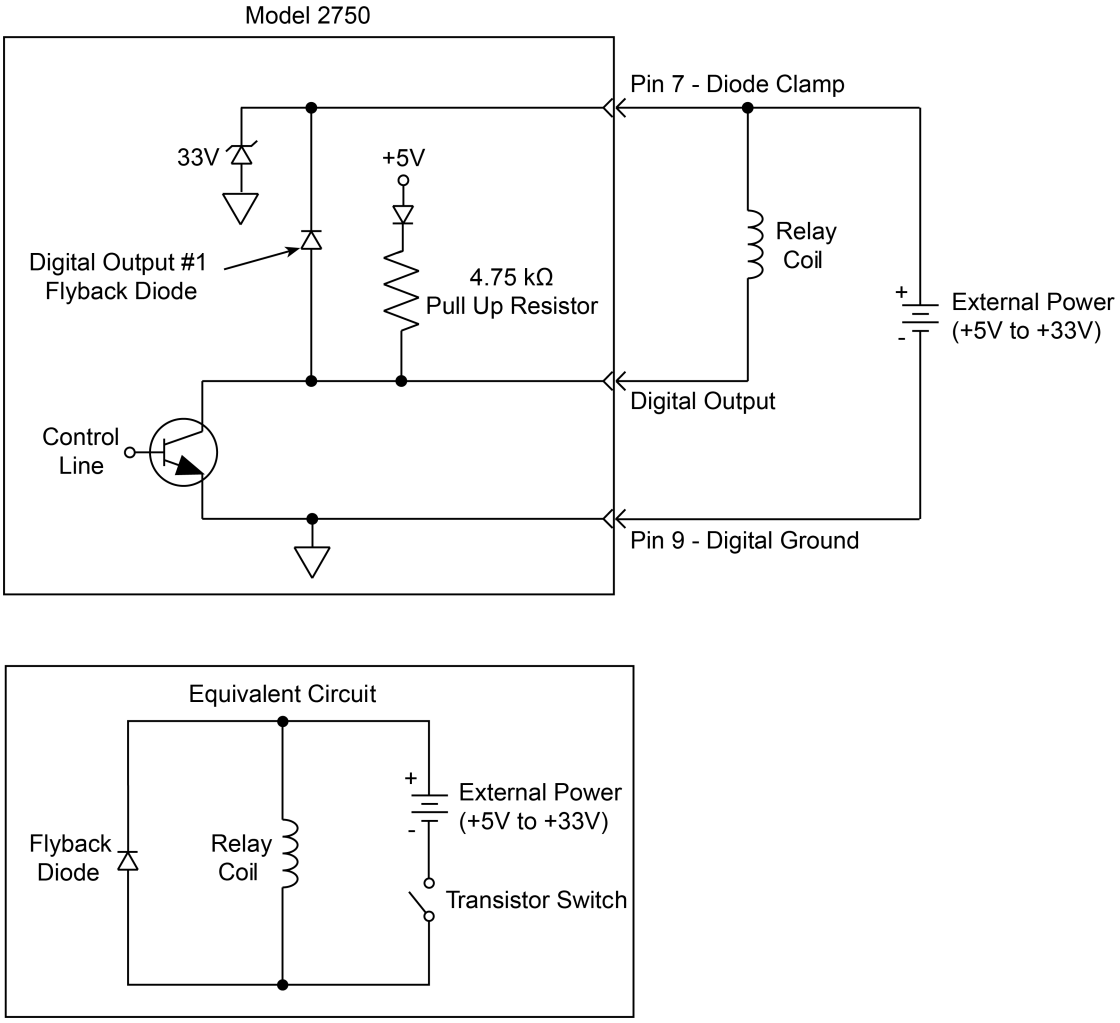
use these fly-back diodes, connect the external supply voltage to pin 7 of the digital I/O port. Make sure the external supply voltage is between +5 V and +33 V and the current required by the device does not exceed 250 mA.

CAUTION

On pin 7, do not exceed +33 V. For the output lines, do not exceed the maximum sink current. The maximum sink current for an output line is 250 mA. Exceeding these limits may cause damage to the instrument that is not covered by the warranty.

An externally powered relay connected to the digital output port is shown in the following figure. Other externally powered devices can be similarly connected by replacing the relay with the device. When the output line is pulled low (0 V), the output transistor sinks current through the external device. In the high state, the output transistor is off (transistor switch open). This interrupts current flow through the external device.

Figure 34: Controlling externally powered relays



Using logic control as the source mode

The digital outputs can be used as logic inputs to active TTL, low-power TTL, or CMOS inputs. For this mode of operation, the output lines can source up to 200 µA.

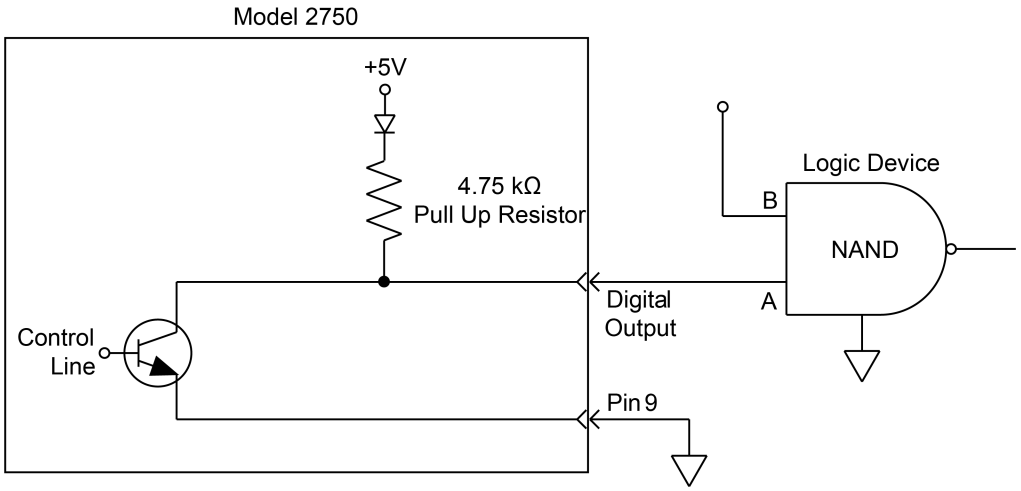
CAUTION

Exceeding 200 µA may cause damage to 2750 that is not covered by the warranty.

The following figure shows how to connect a logic device to one of the output lines. When the output line is pulled high, the transistor turns off (transistor switch open) to provide a reliable logic high output (>3.75 V). When the output line goes low, the transistor turns on (transistor switch closed) to route current to digital ground. As a result, a low logic output (0 V) is provided at the output.

If the second input (B) of the NAND gate is connected to another output line of the port, the output of the NAND gate goes to logic 0 when both digital outputs are set high.

Figure 35: NAND gate control



Setting digital output

The OUTPUT menu controls and configures the digital outputs. Menu items for the digital output include:

- **DOUTPUT:** Use to enable (ON) or disable (OFF) the digital outputs.
- **PULSE:** Use to enable (YES) or disable (NO) the pulse option for the digital outputs.

NOTE

The factory default pulse time is 2 ms maximum. Using remote programming, pulse time can be set from 0.001 s to 99999.999 s. It cannot be set from the front panel.

- **LSENSE:** Use to select the logic sense, active HIGH or active LOW. With active high selected, an output is at approximately +5 V when a reading is at or exceeds the limit. Conversely, with active low selected, an output is at 0 V when a reading reaches or exceeds the limit.
- **MASTR LATCH:** Use to enable (Y) or disable (N) the master limit latch. When enabled, the master limit remains latched when a reading limit is reached or exceeded. When disabled, the master limit line releases immediately when the reading is inside all four limits.

OUTPUT menu

Menu item	Setting	Description
DOUTPUT	ON or OFF	Enable/disable digital outputs.
PULSE	YES or NO	Enable/disable digital pulse output.
LSENSE	HIGH or LOW	Select logic sense.
BEEP	NEVER, INSIDE, or OUTSIDE	Set beeper for limits. See Limits (on page 6-1) for details.
MASTR LATCH	Y or N	Enable/disable master limit latch.

To enable and configure digital outputs:

1. Press **SHIFT** and then **OUTPUT**.
2. If the digital output is already on (`DOUTPUT: ON`), proceed to step 3. Otherwise, press ► to move the cursor to the right, press ▲ or ▼ key to display `ON`, and press **ENTER**.
3. Use the ▼ key to display the master limit latch (`MASTR LATCH`) setting, N (no) or Y (yes).
4. To retain the present master limit setting, proceed to step 5. Otherwise, press ► to move the cursor to the right, press ▲ or ▼ key to display `Y` or `N`, and press **ENTER**.
5. Use the ▼ key to display the present logic sense (LSENSE) setting: HIGH or LOW.
6. To retain the present logic sense setting, proceed to step 7. Otherwise, press ► to move the cursor to the right, press ▲ or ▼ key to display `HIGH` or `LOW`, and press **ENTER**.
7. Use the ▼ key to display the present PULSE mode setting, `NO` or `YES`.
8. To retain the present pulse mode setting, press **ENTER**. Otherwise, press ► to move the cursor to the right, press ▲ or ▼ key to display `NO` or `YES`, and press **ENTER**.

Scanning and the digital output

You can configure limits for each scan channel, but you cannot configure the digital output. For all scan channels that are set to use limits, the digital output functions according to how the 2750 is set up when the scan is run.

Application: Sorting resistors

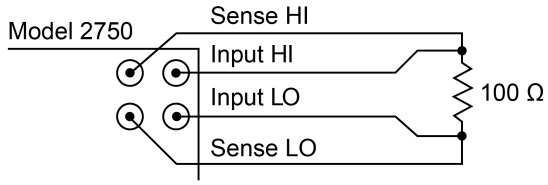
This application sorts a batch of 100 Ω resistors into three bins. Bin 1 is for resistors that are within 1% of the nominal value. Bin 2 is for resistors that exceed 1% tolerance, but are within 5%. Bin 3 is for resistors that exceed 5% tolerance.

The digital outputs of the 2750 can be used to further automate the test system by controlling a compatible component handler to perform the binning operations.

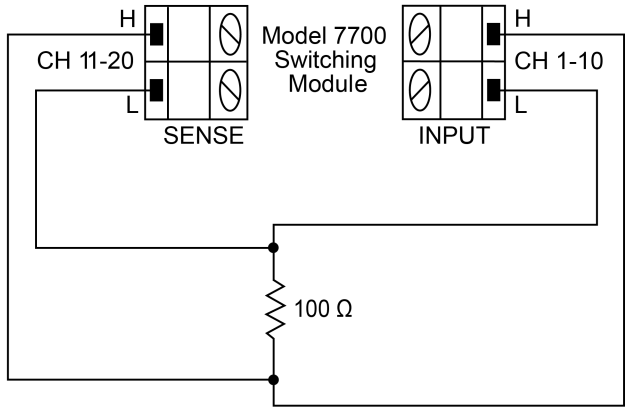
Limits

Limit testing is used to test resistor tolerances. The following figure shows a basic setup using 4-wire offset compensated ohms to test 100 Ω resistors.

Figure 36: Setup to test 100 Ω resistors



A. Front panel inputs



B. Model 7700

Limit 1 will be used to test for the 1% tolerance and Limit 2 will be used to test for the 5% tolerance.

The resistance values for the 1% and 5% tolerances are calculated as follows:

$R_{1\%}$ = 100 Ω × 1%
= 100 Ω × 0.01
= 1 Ω

$R_{5\%}$ = 100 Ω × 5%
= 100 Ω × 0.05
= 5 Ω

The high and low limits are then calculated as follows:

HI Limit1 = 100 Ω + $R_{1\%}$
= 100 Ω + 1 Ω
= 101 Ω

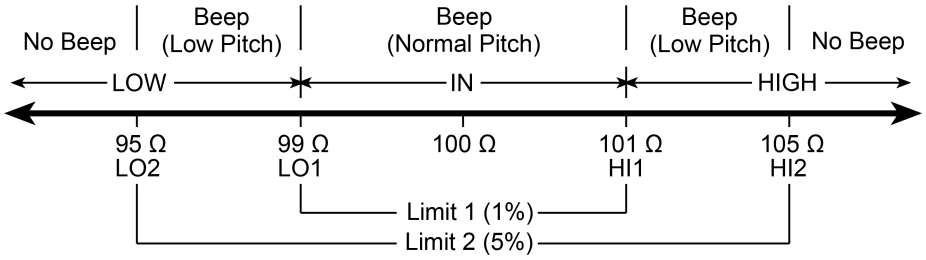
LO Limit 1 = 100 Ω − $R_{1\%}$
= 100 Ω − 1 Ω
= 99 Ω

HI Limit 1 = 100 Ω + $R_{5\%}$
= 100 Ω + 5 Ω
= 5 Ω

LO Limit 2 = 100 Ω − $R_{5\%}$
= 100 Ω − 5 Ω
= 95 Ω

The limits are illustrated in the following figure.

Figure 37: Limits to sort 100 Ω resistors (1%, 5%, and >5%)



For front-panel operation, the INSIDE beeper mode must be used. A normal pitch beep and the message IN indicates that the resistor is within the 1% tolerance limit. This 1% resistor belongs in Bin 1. A raspy beep and the 1 message indicate that the resistor is >1% tolerance but <5% tolerance. This 5% resistor belongs in Bin 2. For resistors >5%, there is no beep. Place these resistors in Bin 3.

For remote operation, make sure both Limit 1 and Limit 2 are enabled.

The following table evaluates the possible pass/fail combinations for this example.

Limit 1 result	Limit 2 result	Resistor tolerance	Bin assignment
Pass	Pass	>1%	1
Fail	Pass	>5%	2
Fail	Fail	>5%	3

A fail condition must be reset before testing the next resistor. Fail can be reset manually or automatically. Refer to :CALCulate3:LIMit1:CLEar:AUTO (on page 9-50) and :CALCulate3:LIMit1:CLEar[:IMMediate] (on page 9-51).

Digital outputs

With the digital outputs of the 2750 enabled, the digital outputs respond as shown in the following table for each resistor reading.

LO limit 2	LO limit 1	HI limit 1	HI limit 2	Resistor tolerance	Bin	Affected outputs*
Pass	Pass	Pass	Pass	1%	1	None
Pass	Fail	Pass	Pass	5%	2	1 and 5
Pass	Pass	Fail	Pass	5%	2	2 and 5
Fail	Fail	Pass	Pass	>5%	3	1, 3, and 5
Pass	Pass	Fail	Fail	>5%	3	2, 4, and 5
*Affected outputs are pulled (or pulsed) high or low when a limit test fails.						

Section 7

Triggering

In this section:

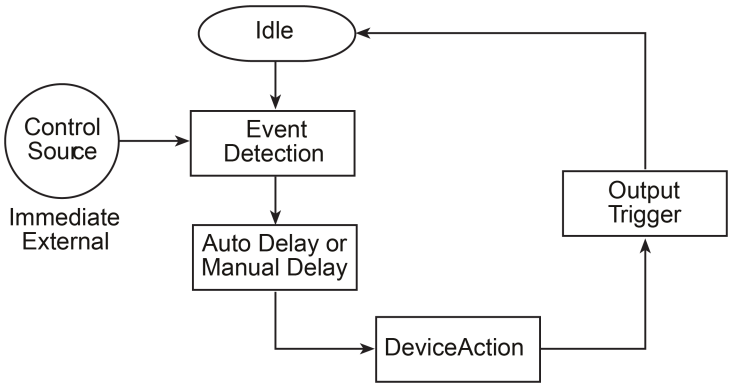
Triggering 7-1
Programming examples for triggering 7-14
Relay closure count..... 7-17

Triggering

This section explains the components of the front-panel trigger model, which controls the triggering operations of the instrument. It also explains external triggering, which allows the 2750 to trigger and be triggered by other instruments.

The flowchart in the following figure summarizes triggering as viewed from the front panel. It is called a trigger model because it is modeled after the SCPI commands used to control triggering.

Figure 38: Front-panel trigger model (without scanning)



For scanning, the trigger model has additional control blocks, such as a timer. These are described in Trigger models when using the front panel (on page 3-27). The complete trigger model, which is based on bus operation, is shown and discussed in Triggering using remote programming (on page 7-10).

Idle

When not scanning and in the continuous trigger mode (factory default setup), the instrument will not stay in idle. Operation will continuously fall through the idle state and proceed to the Event Detection block of the trigger model. When in the one-shot trigger mode (*RST default setup), the TRIG key must be pressed to take the instrument out of idle. After each measurement, the instrument returns to idle and requires the TRIG key to be pressed to continue. The FACT (factory) default setup or *RST default setup is selected from the SHIFT > SETUP menu.

When scanning, the unit is considered idle at the end of a scan operation when the reading for the last channel remains displayed. To restore triggers, press SHIFT and then HALT. See Scanning fundamentals (on page 3-25) for details on scanning.

Control source and event detection

The control source pauses operation until the programmed event occurs and is detected. You can choose an immediate or external control source.

When the control source is immediate, event detection is immediately satisfied and operation continues.

When the control source is external, an event must occur before trigger operation continues. Events include:

- An input trigger through the Trigger Link line EXT TRIG.
- Press of the front-panel TRIG key. The 2750 must be taken out of remote before it responds to the TRIG key. Use the LOCAL key or send GTL over the bus.
- A trigger command (`*TRG` or `GET`) received over the bus.

Delay (auto or manual)

A programmable delay is available after event detection. You can set the delay manually or set it to be automatic. With autodelay selected, the instrument automatically selects a delay period that provides sufficient settling for function and autorange changes and multi-phase measurements.

In the normal measurement state, when autodelay is selected and the External or Bus control source is selected, the 2750 selects a delay based on the selected voltage range. The autodelay period cannot be adjusted by the user. The autodelays are listed in the following table. With one of the other control sources selected, the autodelay is 0.000 s for all functions and ranges.

When the instrument is scanning, the nominal delay is long enough to allow each switching module channel relay to settle before making the measurement. When scanning, the autodelay times in the following are valid for all control sources (immediate, external, timer, manual, or bus).

Autodelay settings

Function	Range and delay								
DCV	100 mV 1 ms	1 V 1 ms	10 V 1 ms	100 V 5 ms	1000 V 5 ms				
ACV	100 mV 400 ms	1 V 400 ms	10 V 400 ms	100 V 400 ms	750 V 400 ms				
FREQ and PERIOD	100 mV 1 ms	1 V 1 ms	10 V 1 ms	100 V 1 ms	750 V 1 ms				
DCI	20 mA 2 ms	100 mA 2 ms	1 A 2 ms	3 A 2 ms					
ACI			1 A 400 ms	3 A 400 ms					
Ω2, Ω4[1]	1 Ω[2] 3ms	10 Ω 3 ms	100 Ω 3 ms	1 kΩ 3 ms	10 kΩ 13 ms	100 kΩ 25 ms	1 MΩ 100 ms	10 MΩ 150 ms	100 MΩ 250 ms
Dry circuit ohms[3]	1 Ω 3 ms	10 Ω 3 ms	100 Ω 13 ms	1 kΩ 13 ms					
Continuity		1 kΩ 3 ms							
TEMP	The autodelay for thermocouples is 1 ms. For thermistors and 4-wire RTDs, the autodelay period is the same as the delay for the resistance range that is used for the measurement.								

1. Delay times also apply for offset-compensated ohms (OCOMP).
2. 1 Ω range not available for the Ω2 function.
3. State (on or off) of offset-compensated ohms (OCOMP) does not affect delay times.

The delay function is accessed by pressing SHIFT and then DELAY. The present delay setting (AUTO or MANual) is displayed. Press the ▲ or ▼ key to display the setting and press ENTER.

If MANual is chosen, also enter the duration of the delay in the hour/minute/second format using the ◄, ►, ▲, and ▼ keys. The maximum is 99 hours, 99 minutes, and 99 seconds. Pressing the AUTO key sets the delay to 0.001 s. Press ENTER to accept the delay or EXIT for no change. If you set a delay shorter than the corresponding autodelay period, measurement uncertainty increases (noisy and/or unsettled readings may result).

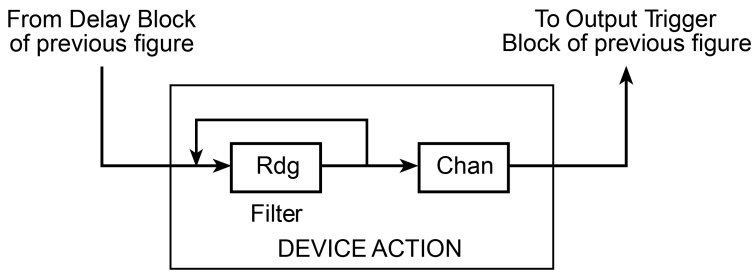
NOTE

If the timer control source is selected, the delay period is only in effect for the first pass through the loop.

Device action

The primary device action is a measurement. However, the device action block could include the following additional actions, as shown in the following figure.

Figure 39: Device action



Filtering: If the repeating filter is enabled, the instrument samples the specified number of reading conversions to yield single filtered reading. Only one reading conversion is performed if the filter is disabled or after the specified number of reading conversions for a moving average filter is reached. After a reading (Rdg) is made, operation proceeds to Channel Closure.

Channel Closure: When scanning, the last device action is channel control. The 2750 opens the presently closed channel and then closes the next channel in the scan.

Output trigger

You can use an output trigger pulse from the 2750 to trigger an external instrument to perform an operation. In general, a trigger pulse is output at this point in the flowchart for each processed reading. An exception is the SCAN function for scanning. For the SCAN function, an output trigger is not output until after the specified number of channels (as set by the sample counter) are scanned.

The output trigger is available at the rear-panel Trigger Link connector. You can use this trigger to trigger another instrument to perform an operation, such as selecting the next channel for an external scan.

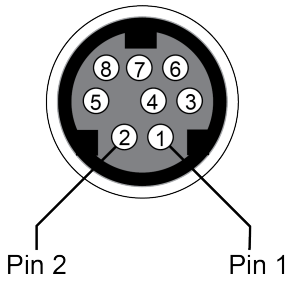
External triggering

The EX TRIG key selects triggering from the Trigger Link, digital I/O, and the TRIG key. When EX TRIG is pressed, the TRIG annunciator lights and dashes are displayed to indicate the instrument is waiting for an external trigger. From the front panel, press the TRIG key to trigger a single reading. Press the EX TRIG key again to toggle to continuous triggers.

The 2750 uses two lines of the Trigger Link rear-panel connector as external trigger (EXT TRIG) input and voltmeter complete (VMC) output. The EXT TRIG line allows the 2750 to be triggered by other instruments. The VMC line allows the 2750 to trigger other instruments.

At the factory, line 1 is configured as VMC and line 2 as EXT TRIG. The connector pinout is shown in the following figure.

Figure 40: Rear-panel pinout



Pin number	Description
1	Voltmeter complete output
2	External trigger input
3	No connection
4	No connection
5	No connection
6	No connection
7	Signal ground
8	Signal ground

Digital I/O

Pin 6 (Ext Trig) of the Digital I/O can also be used as the external trigger input for the 2750. Line 2 of the TRIG LINK is physically connected to pin 6 of the Digital I/O connector.

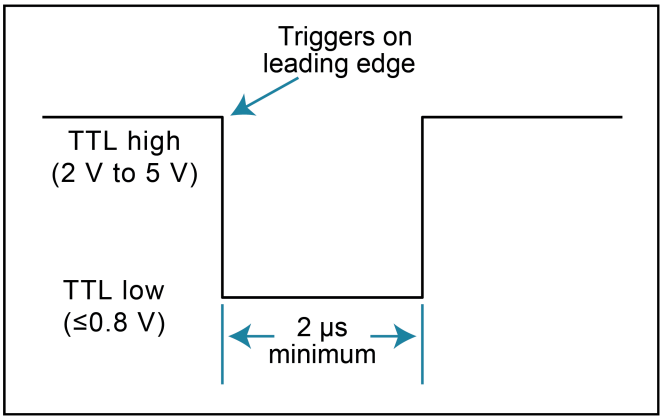
The Digital I/O has a hardware interlock line (pin 8) that allows the use of an external circuit to control input triggers. When that line is left open or pulled high (+5 V), input triggers are enabled. When pulled low to 0 V, input triggers are disabled. When disabled, the2750 will not respond to an input trigger.

Details on the Digital I/O are provided in Limits and digital I/O (on page 6-1).

External trigger

The EXT TRIG input requires a falling-edge TTL-compatible pulse with the specifications shown in the following figure. In general, you can use external triggers to control measure operations. For the 2750 to respond to external triggers, the trigger model must be configured for it.

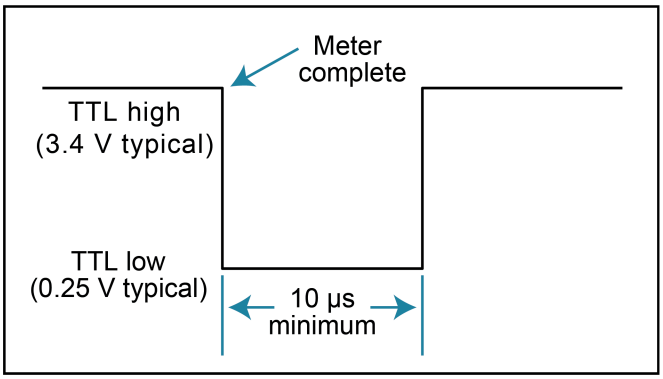
Figure 41: Trigger Link input pulse specifications (EXT TRIG)



Voltmeter complete

The voltmeter complete (VMC) output provides a TTL-compatible output pulse that you can use to trigger other instruments. The specifications for this trigger pulse are shown in following figure. Typically, you have the 2750 to output a trigger after the settling time of each measurement.

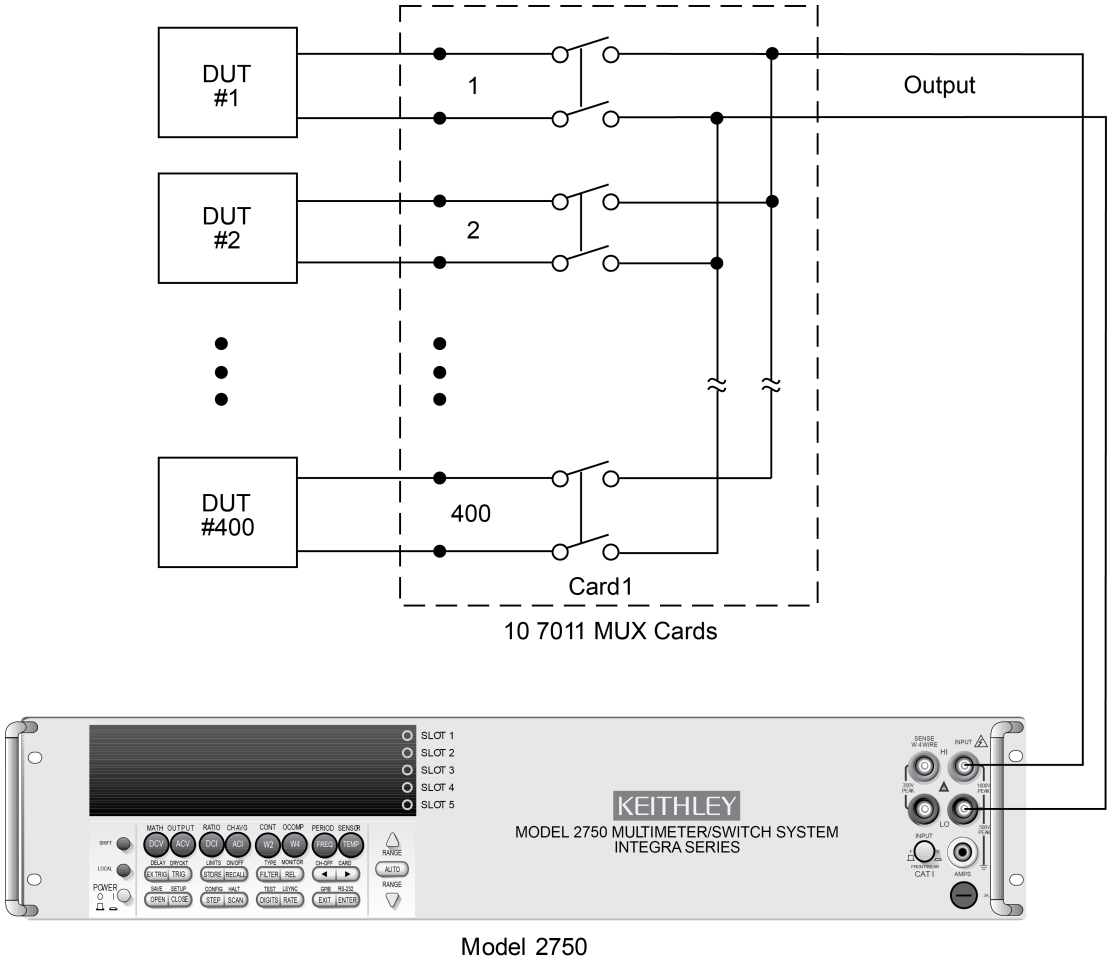
Figure 42: Trigger link output pulse specifications (VMC)



External triggering example

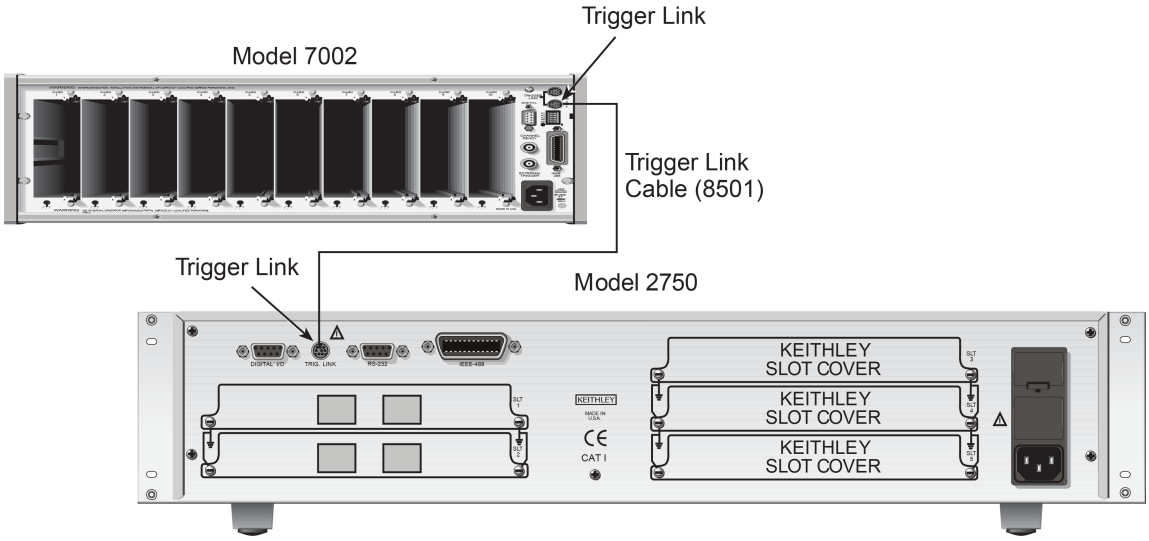
For a test system that requires a large number of switching channels, the 2750 can be used with external scanners such as the Keithley Models 7001 and 7002. For example, ten Model 7011 cards installed in the Model 7002 can provide up to 400 2-pole channels, as shown in the following figure.

Figure 43: DUT test system



The Trigger Link connections for this test system are shown in the following figure.

Figure 44: Trigger link connections



The Trigger Link of the 2750 is connected to Trigger Link (either IN or OUT) of the Model 7002. When the 7002 is configured for the default trigger settings, line 1 is an input and line 2 is an output. This complements the trigger lines on the 2750.

External triggering example configuration

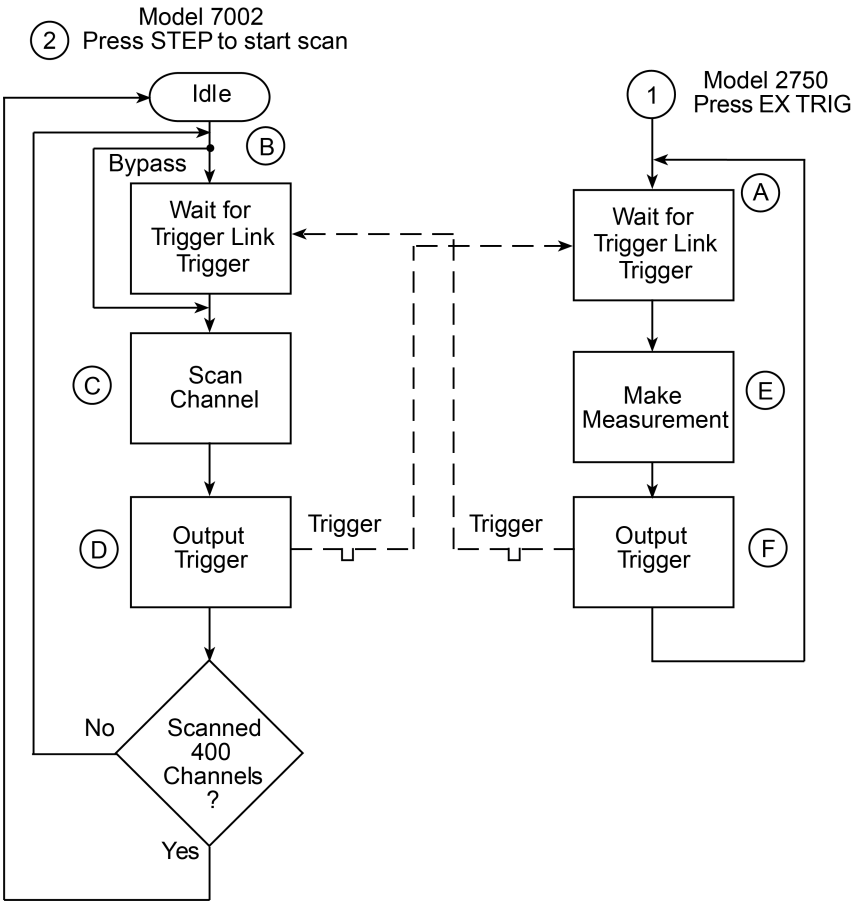
For this example, the 2750 and 7002 are configured as follows:

- 2750:
  Factory defaults restored (accessed from SHIFT-SETUP).
  External triggers (accessed from EX TRIG).
  Buffer enabled and set to store 400 readings.
- Factory defaults restored
  Scan list = 1!1-1!400
  Number of scans = 1
  Channel spacing = TrigLink.

To configure the example:

1. Press **EX TRIG** to place the 2750 in the external trigger mode.
2. Press **STEP** on the 7002 to take it out of idle and start the scan. The output pulse of the scanner triggers the 2750 to make a reading, store it, and send a trigger pulse. The following explanation on operation is referenced to the operation model shown in the following figure.

Figure 45: Operation model for triggering example



A. Pressing EX TRIG on the 2750 places it at point A in the flowchart, where it is waiting for an external trigger.

B. Pressing STEP on the 7002 takes it out of the idle state and places operation at point B in the flowchart.

C. For the first pass through the model, the scanner does not wait at point B for a trigger. Instead, it closes the first channel.

D. After the relay settles, the 7002 outputs a Channel Ready pulse. Since the instrument is programmed to scan 400 channels, operation loops back to point B, where it waits for an input trigger.

E and F. 2750 operation is at point A waiting for a trigger. The output Channel Ready pulse from the 7002 triggers the 2750 to measure DUT 1 (point E). After the measurement is complete, the 2750 outputs a completion pulse (point F) and then loops back to point A where it waits for another input trigger.

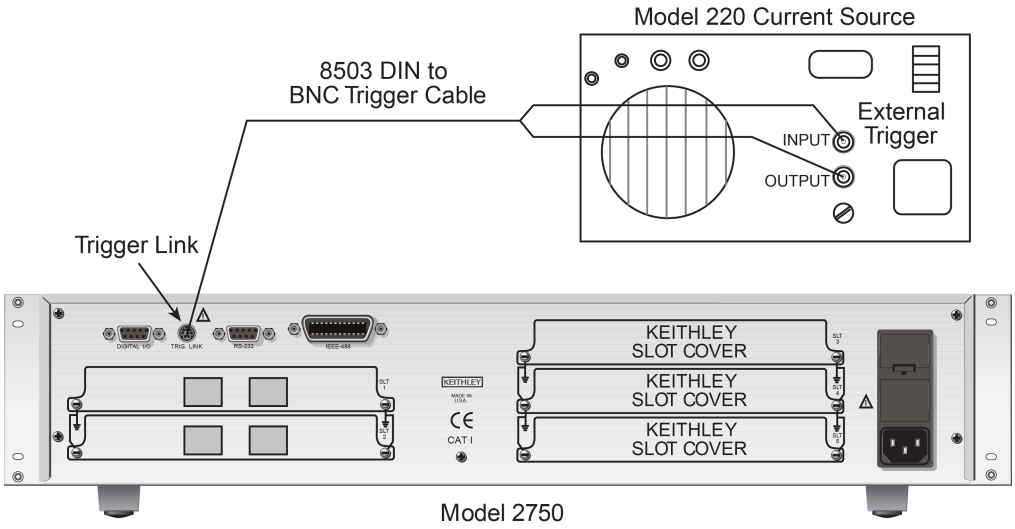
The trigger applied to the 7002 from the 2750 closes the next channel in the scan. This triggers the 2750 to measure the next DUT. The process continues until all 400 channels are scanned, measured, and stored in the buffer.

External triggering with BNC connections

An adapter cable is available to connect the micro-DIN Trigger Link of the 2750 to instruments with BNC trigger connections. The Model 8503 DIN to BNC Trigger Cable has a micro-DIN connector at one end and two BNC connectors at the other end. The BNC cables are labeled VMC (trigger line 1) and EXT TRIG (trigger line 2).

The following figure shows how to connect a current source to the Trigger Link of the 2750 using the adapter cable. When used with the STEP mode of the Model 6220, you can perform synchronized source-measure operations without the use of a computer. Whenever the Model 6220 receives a trigger from the 2750, it steps to the next current source value.

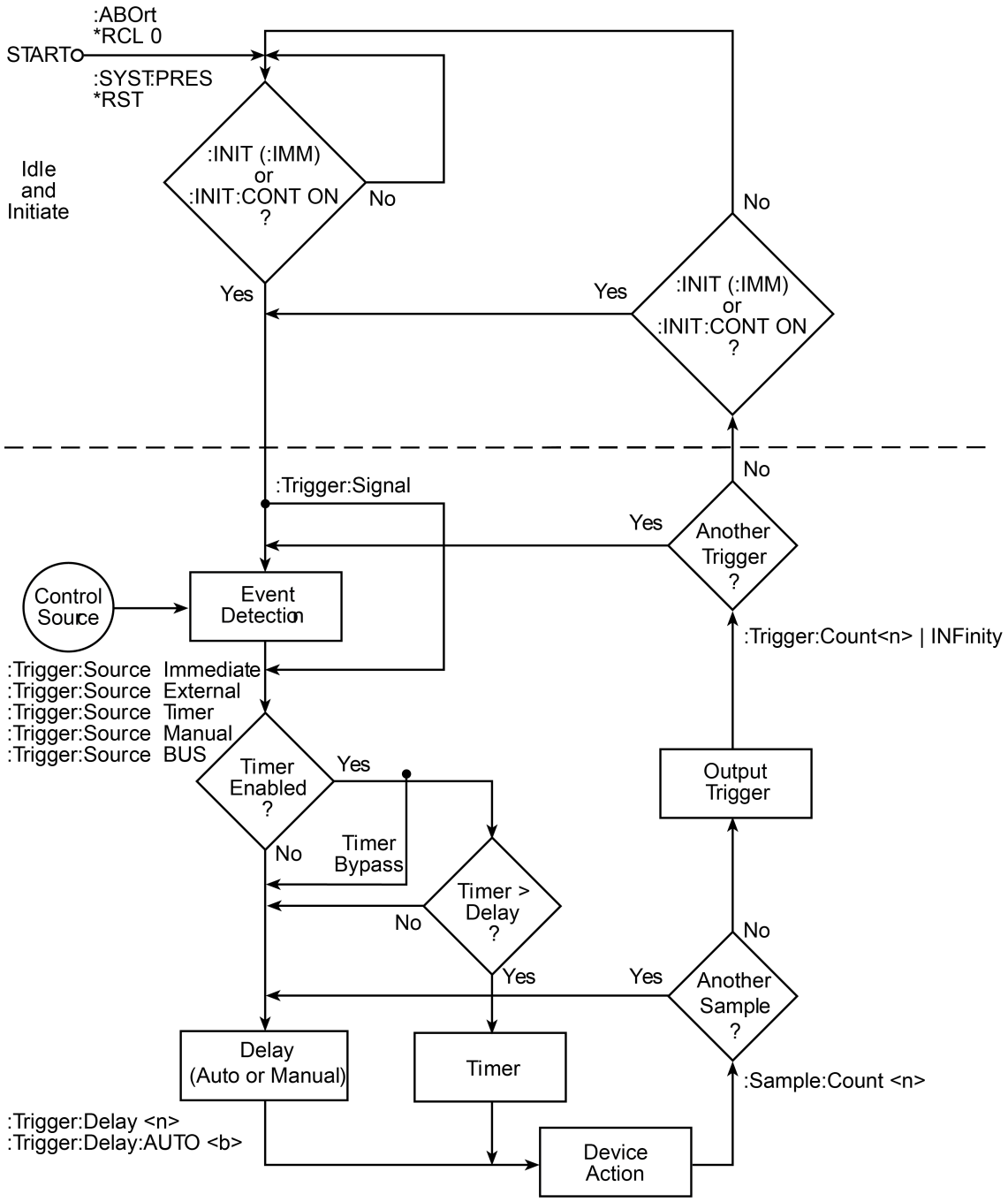
Figure 46: DIN to BNC trigger cable



Triggering using remote programming

The following topics describe how the 2750 operates for remote operation. The following figure summarizes operation over the bus. The flow chart is called the trigger model because operation is controlled by SCPI commands from the trigger subsystem. Key SCPI commands are included in the trigger model.

Figure 47: Trigger model (remote operation)



Idle and initiate

The instrument is considered to be in the idle state whenever operation is at the top of the trigger model. As shown in the figure in Triggering using remote programming (on page 7-10), initiation needs to be satisfied to take the instrument out of idle. In the idle state, the instrument cannot perform any measure or step and scan operations.

The following commands return operation to idle at the START point of the trigger model:

- `:ABORt`
- `:SYSTem:PRESet`
- `*RCL 0`, `1`, or `2`
- `*RST`

The next actions depend on the state of initiation. If continuous initiation is already enabled, the instrument leaves the idle state.

- `SYSTem:PRESet` enables continuous initiation. Therefore, operation immediately leaves the idle state when it is sent.
- The `*RCL 0` command also enables continuous initiation if `INITiate:CONTinuous ON` is a user-saved default.
- `*RST` disables continuous initiation and the instrument remains in the idle state.

Either of the following two initiate commands takes the instrument out of the idle state:

- `:INITiate`
- `:INITiate:CONTinuous ON`

NOTE

When the instrument is in remote, press the LOCAL key to restore continuous front-panel operation.

Trigger model operation

Once the instrument is taken out of idle, operation proceeds through the trigger model down to the device action. In general, the device action includes a measurement and, when scanning, closes the next channel.

Control Source: As shown in the figure in Triggering using remote programming (on page 7-10), a control source is used to hold up operation until the programmed event occurs. The control source options are as follows:

- **IMMediate:** Event detection is immediately satisfied allowing operation to continue.
- **MANual:** Event detection is satisfied by pressing the TRIG key. The 2750 must be in LOCAL mode for it to respond to the TRIG key. Press the LOCAL key or send GTL over the bus to remove the instrument from the remote mode.
- **TIMer:** With the timer source enabled (selected), event detection is immediately satisfied. On the initial pass through the loop, the Timer Bypass is enabled allowing operation to bypass the Timer and continue on to the Delay block.

On each subsequent pass through the loop, the Timer Bypass is disabled. Operation is then delayed by the Timer or the Delay. If the user-set Timer interval is larger than the user-set Delay, the Timer will control the length of the delay. Otherwise, the length of the delay is controlled by the user-set Delay period.

The Timer interval can be set from 0 to 999999.999 seconds. The timer source is only available during scan operation. The timer resets to its initial state when the instrument goes into the normal mode of operation or into the idle state.

- **EXTernal:** Event detection is satisfied when an input trigger via the TRIG LINK connector is received by the 2750.
- **BUS:** Event detection is satisfied when a bus trigger (GET or *TRG) is received by the 2750.

Delay and Device Action: These blocks of the trigger model operate the same for both front-panel and GPIB operation. See Trigger models when using the front panel (on page 3-27), for operating information on these trigger model blocks.

Counters: Programmable counters are used to repeat operations within the trigger model. For example, if performing a 10-channel scan, the sample counter would be set to 10. Operation continues until all 10 channels are scanned and measured. To repeat the scan three times, set the trigger counter to three.

For a sample count value >1, the sample readings are automatically stored in the buffer. For example, with sample count set to 5, the five measured readings are stored in the buffer. If the trigger model is configured to repeat the sample readings (for example, the trigger count is set 2), those five new readings overwrite the original five readings in the buffer.

Output Trigger: The 2750 sends one or more output triggers. The output trigger is applied to the Trigger Link connector on the rear panel. It can be used to trigger an external instrument to perform an operation.

The trigger model can be configured to output a trigger after the completion of a series of measurements, or after every measurement. For example, with the sample counter set to 10 and the trigger counter set to one, a trigger is sent after the 10 measurements are made. If instead, the trigger counter is set to 10 and the sample counter is set to 1, a trigger is sent after each measurement.

Programming examples for triggering

There are several commands used to trigger and return readings. The proper commands and sequence to use depend on trigger state (continuous or non-continuous) and what you are trying to accomplish. The following examples provide sample code for common scenarios.

NOTE

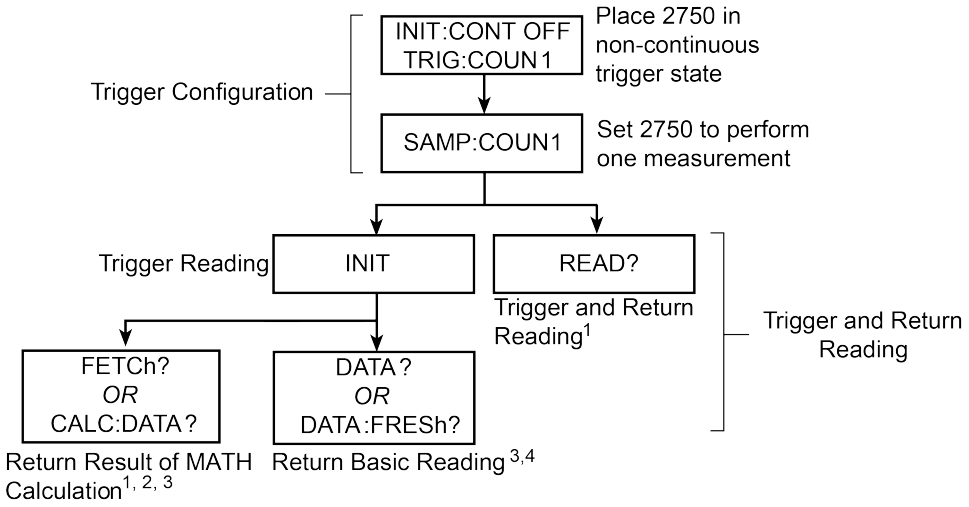
Each exercise includes the commands used to configure triggering. Once triggering is configured, the commands to trigger and return readings can be repeated as often as needed unless noted otherwise.

Example: Trigger and return a single reading

The instrument is typically used in a non-continuous trigger mode. In this mode, commands are used to trigger one or more readings. After the specified number of readings are completed, the measurement process stops.

The following figure provides a command sequence to trigger and return one reading.

Figure 48: Trigger and return a single reading



Notes on the figure:

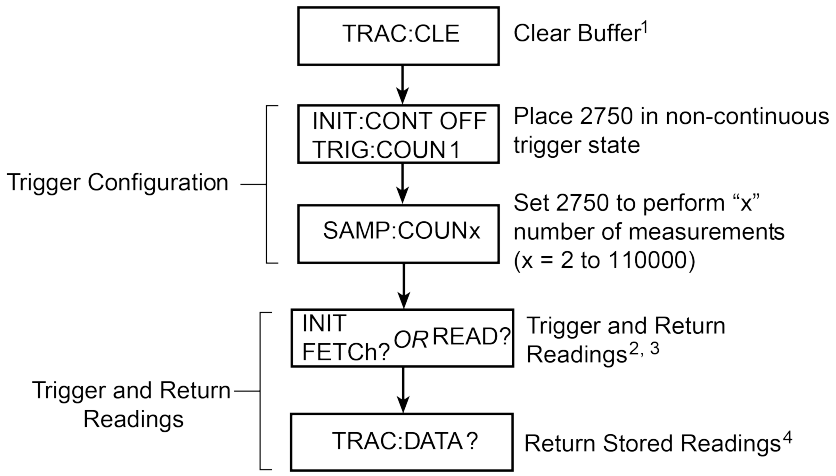
1. If a MATH function (mx+b, percent, or 1/X) is enabled, the result of the calculation is returned. Refer to Math (on page 4-14) for information on the MATH functions.
2. If there is no MATH function enabled, `FETCh?` and `CALC:DATA?` return the basic reading.
3. `FETCh?`, `CALC:DATA?`, and `DATA?` do not trigger readings. They return the last reading. If you send one of these commands before triggering a new reading, the old reading is returned.
4. `DATA:FRESh?` can only be used once to return the same reading. Sending it again without triggering a new reading causes error −230, data corrupt or stale.

Example: Trigger and return multiple readings

The instrument is typically used in a non-continuous trigger mode. In this mode, commands are used to trigger one or more readings. After the specified number of readings are completed, the measurement process stops.

The following figure provides a command sequence to trigger and return multiple readings.

Figure 49: Trigger and return multiple readings example



Notes on the figure:

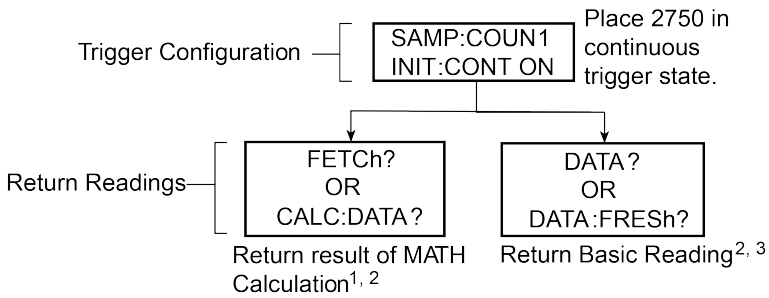
1. To trigger and return multiple readings, the buffer must be cleared of readings that were stored by the TRACe command or front-panel operation. Refer to Reading buffers (on page 8-1) for details on buffer operation.
2. `INIT` triggers the measurements and `FETCh?` returns the readings. Sending `FETCh?` without sending `INIT` returns old readings.

3. `READ?` performs an `INIT` to trigger the measurements. Use `FETCh?` to return the readings.
4. Triggered readings are automatically stored in the buffer. Statistics for buffer readings are also stored in the buffer. CALC2 commands are used to calculate and return buffer statistics.

Example: Return a single reading (continuous triggering)

Readings can be returned when the instrument is in the continuous measurement (trigger) mode. Each time a read command is sent, the latest reading is returned. The following figure shows a command sequence that returns a single reading when the instrument is in the continuous trigger state.

Figure 50: Return a single reading (continuous triggering)



Notes on figure:

1. If a MATH function (mx+b, percent, or 1/X) is enabled, the result of the calculation is returned. If there is no MATH function enabled, `FETCh?` and `CALC:DATA?` return the basic reading. Refer to Math (on page 4-14) for detail on the math functions.
2. None of these read commands trigger measurements. They return the latest reading. If `FETCH?`, `CALC:DATA?`, or `DATA?` is sent before a new reading is triggered, the previous reading is returned.
3. `DATA:FRESh?` can only be used once to return the same reading. Sending it again before a new reading is triggered causes error −230, data corrupt or stale.

Relay closure count

The 2750 keeps an internal count of the number of times each module relay has been closed. The total number of relay closures are stored in EEPROM on the card. This count helps you determine if and when any relays require replacement (see module contact life specifications).

Relay closures are counted only when a relay cycles from open to closed state. If you send multiple close commands to the same channel without sending an open command, only the first closure is counted.

Relay closure count can only be read using remote operation. The relay closure count commands are:

- :ROUTe:CLOSe:COUNt:INTerval (on page 9-70)
- :ROUTe:CLOSe:COUNT? (on page 9-71)

NOTE

You can reset the relay closure count to zero. For details, see the *Model 2750 Service Manual*, "Plug-in module relay closure count."

Section 8

Reading buffers

In this section:

Buffer overview 8-1
Autoclear 8-1
Timestamps 8-2
Storing readings 8-4
Recalling readings 8-5
Buffer statistics 8-6
Remote programming and the buffer 8-7

Buffer overview

The 2750 has a data store (buffer) that can store from 2 to 110,000 readings. The instrument stores the readings that are displayed during the storage process. Each timestamped reading includes the buffer location number and a timestamp.

The data store also provides statistical data on the measured readings stored in the buffer. These include minimum, maximum, average, peak-to-peak, and standard deviation.

When scanning, the readings are automatically stored in the buffer.

Instrument operations, including buffer operation, are performed on the input signal in a sequential manner. Buffer storage is one of the last steps in the sequence. Refer Signal processing sequence (on page 4-25) for detail.

Autoclear

When buffer autoclear is enabled, the buffer is cleared (readings lost) before a new storage operation starts. The buffer can be manually cleared by setting the number of readings to store (buffer size) to `000000`.

NOTE

When editing the reading count over the front panel, press AUTO to reset the count to 000000. You can then press <ENTER> to clear the buffer.

When buffer autoclear is disabled, the buffer is not cleared and the buffer size is set to 110000. Each subsequent storage operation appends the readings to the buffer. When the buffer fills with 110,000 readings, the storage process stops. The 110,000 readings are cleared before the next storage operation starts.

With buffer autoclear disabled, the only two valid buffer size values are 110000 and 000000. Buffer size 000000 clears the buffer. Entering any other buffer size value resets the buffer size to 110000.

If the buffer is empty when the 2750 is turned off, buffer autoclear is enabled when it is turned back on. If the buffer is not empty, the instrument powers up to the last autoclear setting. If the instrument powers up with buffer autoclear off, buffer size is fixed at 110000. You must enable autoclear to change the buffer size.

The autoclear setting is not affected by `SYSTem:PRESet` or `*RST`. However, buffer autoclear is enabled by the front-panel `FACTory` defaults.

Enabling and disabling buffer autoclear

If you change the state of buffer autoclear, the buffer is cleared.

To enable buffer autoclear:

1. Press SHIFT and then SETUP.
2. Use the▲ and ▼ keys to display the present state of buffer autoclear:

   `(BUF AUTOCLR): Y (yes) or N (no)`
3. To retain the present state of buffer autoclear, press ENTER or EXIT.
4. To change the state of buffer autoclear, press ► to place the cursor on the present setting (Y or N).
5. Use the ▲ or ▼ key to display the setting (Y=enabled, N=disabled), and press ENTER.

For remote programming, autoclear is enabled or disabled using :TRACe:CLEar:AUTO (on page 9-200). You can also use :TRACe:CLEar[:IMMediate] (on page 9-201) to clear the buffer.

Timestamps

Each stored reading is referenced to either a real-time clock timestamp or to a relative timestamp.

When the real-time clock is selected, each stored reading is timestamped with the time and date. For the time, the seconds reading has 0.01 s resolution.

When the relative timestamp is relative selected, there is an absolute and delta timestamp types for each reading. The absolute timestamp (S) references each stored reading to zero seconds. Therefore, the first reading in the buffer has an absolute timestamp of zero seconds. The delta timestamp (dS) indicates the time (in seconds) between the displayed reading and the reading before it. The resolution for each timestamp is 0.001 s.

When autoclear is disabled and relative timestamp is selected, every stored reading is referenced to the first reading (#0), even if the buffer is stopped and started again. For example, assume you stored 10 readings in the buffer, and one hour later, you store 10 more readings. The timestamps for all 20 readings are referenced to the first reading. Therefore, the timestamp for the 11th reading (#10) is one hour (3600 seconds).

When the 2750 is turned off, the relative timestamp resets to 0 s when the instrument is turned back on. If you have readings stored in the buffer and autoclear is disabled when the unit is turned off, subsequent stored readings are appended to the old group of readings. However, the relative timestamps for the new readings are referenced to 0 s.

When recalling stored readings from the front panel, both absolute and delta timestamps are provided. For remote operation, the absolute or delta timestamp is returned with each buffer reading. The TRACe:TSTamp:FORMat (on page 9-207) command selects the relative timestamp type.

Setting time and date

For the real-time clock, the time and date are set at the factory, but can be changed as needed.

To set the time:

1. Press **SHIFT** and then **SETUP**.
2. Use the ▲ and ▼ keys to display `SET TIME` and press **ENTER**. The displayed clock is running in the hour/minute/second AM/PM format.
3. Use the edit keys (◄, ►, ▲, and ▼) to set the hour, minute, and AM/PM. Seconds cannot be set. Press **ENTER**.

To set the date:

1. Press **SHIFT** and then **SETUP**.
2. Use the ▲ and ▼ keys to display `SET DATE` and press **ENTER** to display the date in the month/day/year format.
3. Use the edit keys (◄, ►, ▲, and ▼ to set the date (month/day/year).
4. Press **ENTER**.

Selecting timestamp

Changing the timestamp will clear the buffer if a storage is in process. The message "BUF CLEARED" will be displayed to indicate the buffer readings were lost. If no storage is in process, changing the timestamp will not clear the buffer.

To select either the real-time clock timestamp or the relative timestamp:

1. Press SHIFT and then SETUP.
2. Use the ▲ or ▼ keys to display "TSTAMP."
3. Press the ► key to place the cursor on the presently selected timestamp, REL or RTCL.
4. Use the ▲ or ▼ key to display the relative (REL) or real-time clock (RTCL), and press ENTER.

Storing readings

To store readings using the front panel:

1. Set up the 2750 for the measurement.
2. Press the **STORE** key.
3. Use the ◄, ►, ▲, and ▼ keys to specify the number of readings to store in the buffer (2 to 110000). Pressing the **AUTO** key sets the reading count to 000000.

NOTE

When buffer autoclear is disabled, the only valid buffer size values are 110000 and 000000 (which clears the buffer). Any other buffer size value is ignored.

4. Press **ENTER**. The asterisk (`*`) annunciator turns on to indicate the buffer is enabled. It turns off when the storage is finished.
5. The buffer can be stopped at any time by pressing EXIT.

NOTE

Stored readings are not lost when the instrument is turned off. To clear the buffer, set the reading value to `000000` and press ENTER.

For remote programming, continuous storage mode can be selected. After the buffer fills, operation wraps around to the beginning of the buffer (location #0) and starts to overwrite old reading data. Refer to :TRACe:FEED:CONTrol (on page 9-203) for additional information.

Recalling readings

Readings stored in the buffer are displayed by pressing the RECALL key. The readings are on the left side of the display and the buffer location number (reading number) and timestamps are on the right side.

To view stored readings and buffer statistics:

1. Press **RECALL**. The BUFFER annunciator indicates that stored readings are displayed. The double-arrow annunciator indicates that more data can be viewed with the ◄, ►, ▲, and ▼ keys.
2. As shown in the following figures, use the ◄, ►, ▲, and ▼ keys to navigate through the reading numbers, reading values, statistics, and timestamps. If buffer statistics are enabled, such as standard deviation, average, peak-to-peak, minimum, and maximum, the STAT annunciator is on.

NOTE

The longer you hold in the ▲ or ▼ key, the faster you scroll through the buffer. Scrolling speed increases to incrementing or decrementing the buffer reading number by 100, and then by 500. When you get close to the reading number, release the ▲ or ▼ key. Press and hold in the ▲ or ▼ key again to scroll one reading at a time.

Figure 51: Recalling buffer data: Relative timestamp

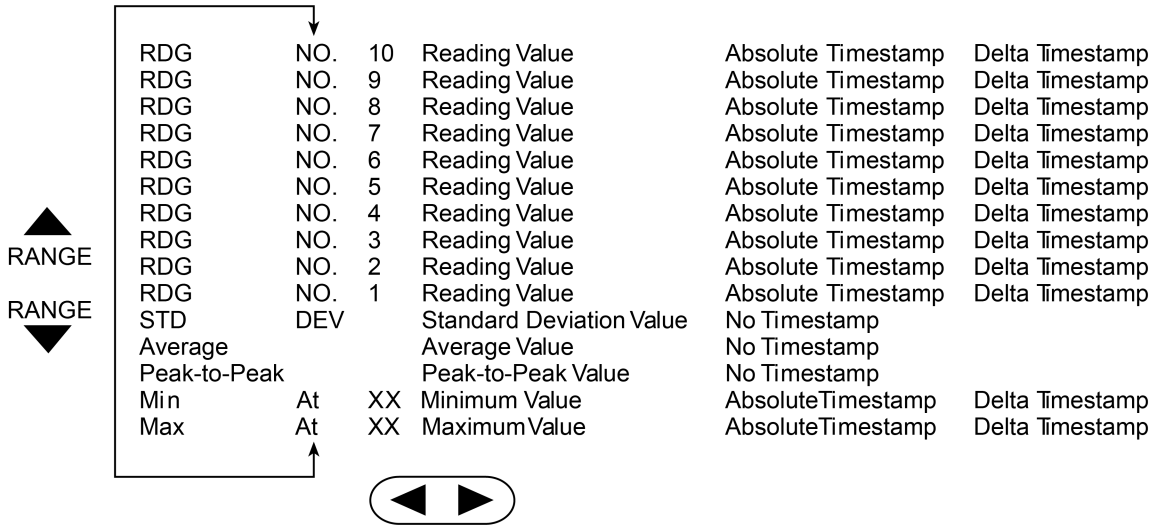
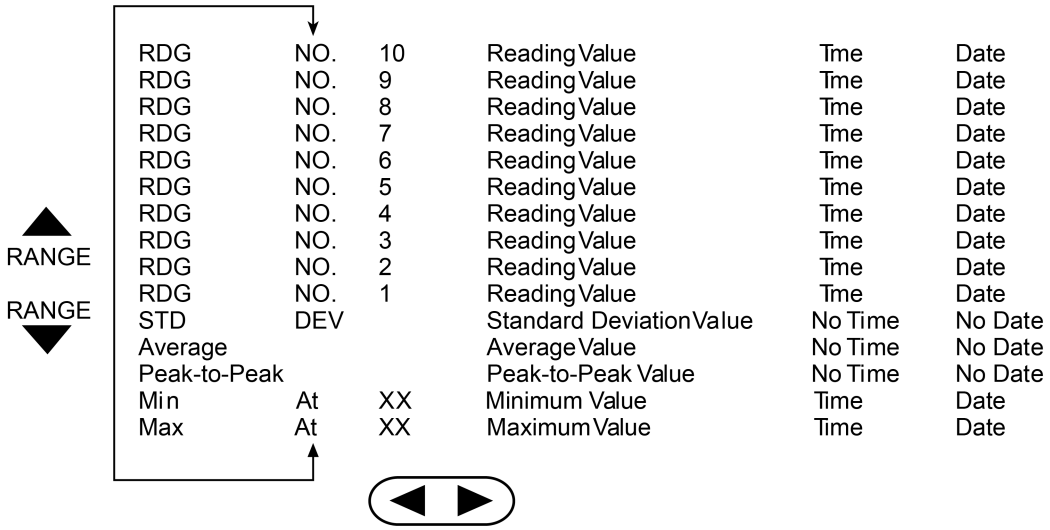


Figure 52: Recalling buffer data: Real-time clock timestamp



Buffer statistics

The minimum (MIN) and maximum (MAX) locations return the minimum and maximum readings stored in the buffer. It also indicates the buffer location of these readings.

Peak-to-peak is the absolute value of the difference between the MIN and MAX readings. It is calculated as follows:

Peak-to-Peak = |MAX − MIN|

Average is the mean of the buffer readings. Mean is calculated as follows:

$$y = \frac{\sum_{i=1}^{n} X_i}{n}$$

where:

- X_i is a stored reading.
- n is the number of stored readings.

Standard deviation is the standard deviation of the buffered readings. The equation used to calculate the standard deviation is:

$$y = \sqrt{\frac{\sum_{i=1}^{n} X_i^2 - \left(\frac{1}{n}\left(\sum_{i=1}^{n} X_i\right)^2\right)}{n-1}}$$

where:

- X_i is a stored reading.
- n is the number of stored readings.

NOTE

If the standard deviation calculation is being performed on a buffer that has more than 1000 readings, the `CALCULATING` message flashes to indicate that the 2750 is busy. The front-panel keys do not operate while the message is displayed.

Remote programming and the buffer

`TRACe` subsystem commands store and recall readings in the buffer. Refer to the TRACe subsystem (on page 9-200) for information.

`CALCulate2` commands obtain statistics from the buffer data. Refer to the CALCulate2 subsystem (on page 9-47) for information. There is no SCPI command to obtain the peak-to-peak statistic. To get the peak-to-peak statistic, your program must calculate it from the maximum and minimum statistics

The SYStem commands allow you to set the time, date, and timestamp type. Refer to the SYSTem subsystem (on page 9-174) for information.

The elements for the TRACe:DATA? are set by :FORMat:ELEMents (on page 9-64).

The measurement event register can be read to check when the buffer becomes ¼, ½, ¾, or completely full. Status registers are described in Status model (on page 11-1).

NOTE

When readings are stored in the buffer by the TRACe command or by the front-panel data store operation), `INIT` and multi-sample `READ?` queries are locked out. If you have readings in the buffer that were stored in that manner, you cannot use the `INIT` or `READ?` command if sample count is >1. If you attempt to stored readings, the error message −225, out of memory, is generated.

Section 9

Command reference

In this section:

Using the SCPI command reference 9-1
SCPI reference tables 9-6
:CONFigure? 9-30
:FETCh? 9-32
:INITiate:CONTinuous 9-33
:INITiate[:IMMediate] 9-34
:MEASure? 9-35
:READ? 9-37
CALCulate subsystem 9-39
CALCulate2 subsystem 9-47
CALCulate3 subsystem 9-50
DISPlay subsystem 9-61
FORMat subsystem 9-63
ROUTe subsystem 9-69
SENSe[1] subsystem 9-79
STATus subsystem 9-166
SYSTem subsystem 9-174
TRACe subsystem 9-200
TRIGger subsystem 9-208
UNIT subsystem 9-213

Using the SCPI command reference

The SCPI command reference contains detailed descriptions of each of the SCPI commands that you can use to control your instrument. Each command description is broken into several standard subsections. The following figure shows an example of a command description.

Figure 53: SCPI command description example

:EXAMple:COMMand:STATe

This command is an example of a typical SCPI command that turns an instrument feature on or off.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle	Save settings	1 (ON)

Usage

```
:EXAMple:COMMand:STATe <state>
:EXAMple:COMMand:STATE?
```

<state>	Disable the example feature: 0 or OFF Enable the example feature: 1 or ON

Details

This command is an example of a typical SCPI command that enables or disables a feature.

Example

:EXAMple:COMMand:STATe ON	Turn the example feature on.

Also see

:EXAMple:COMMand:UNIT (on page 6-100)

Each command listing is divided into five subsections that contain information about the command:

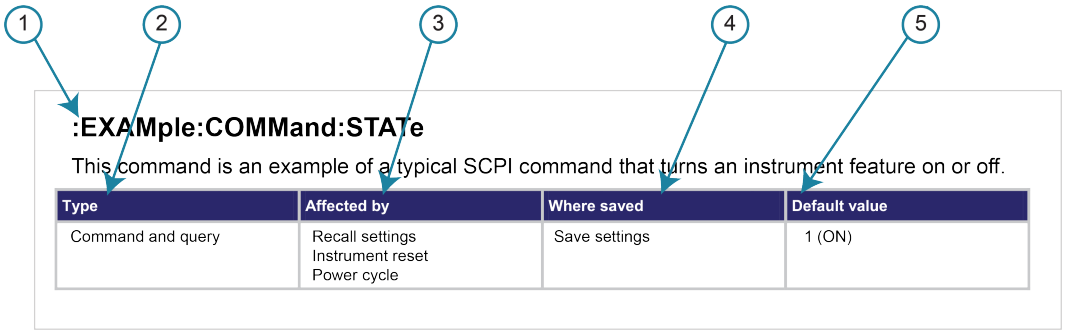
- Command name and summary table
- Usage
- Details
- Example
- Also see

The content of each of these subsections is described in the following topics.

Command name and summary table

Each instrument command description starts with the command name, followed by a table with relevant information for each command. Definitions for the numbered items are listed following the figure.

Figure 54: SCPI command name and summary table



1 **Instrument command name**. Signals the beginning of the command description and is followed by a brief description of what the command does.

2 **Type of command**. Options are:

- **Command only**. There is a command but no query option for this command.
- **Command and query**. The command has both a command and query form.
- **Query only**. This command is a query.

3 **Affected by**. Commands or actions that have a direct effect on the instrument command.

- **Recall settings.** If you send `*RCL` to recall the system settings, this setting is changed to the saved value.
- **Instrument reset.** When you reset the instrument, this command is reset to its default value. Reset can be done from the front panel or when you send `*RST`.
- **Power cycle.** When you power cycle the instrument, this command is reset to its default value.

4 **Where saved**. Indicates where the command settings reside once they are used on an instrument. Options include:

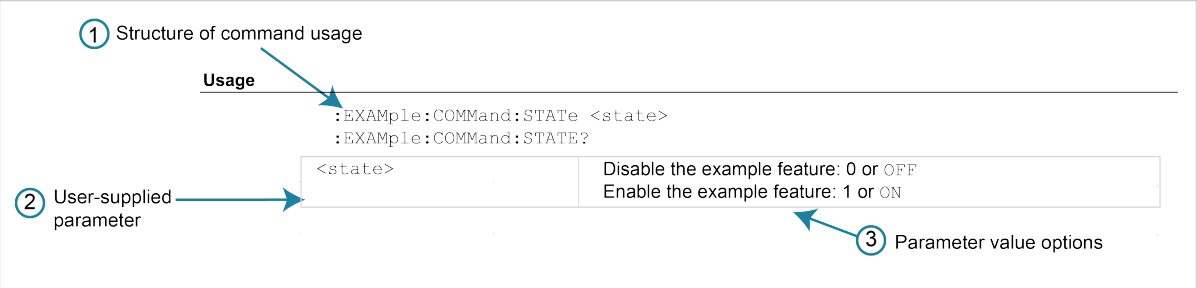
- **Not saved.** Command is not saved and must be sent each time you use it.
- **Nonvolatile memory.** The command is stored in a storage area in the instrument where information is saved even when the instrument is turned off.
- **Save settings.** This command is saved when you send the `*SAV` command.

5 **Default value:** Lists the default value for the command. The parameter values are defined in the Usage or Details sections of the command description.

Command usage

The Usage section of the remote command listing shows how to properly structure the command. Each line in the Usage section is a separate variation of the command usage; all possible command usage options are shown here.

Figure 55: SCPI command description usage identification



1. **Structure of command usage:** Shows the organization of the parts of the command.
2. **User-supplied parameters:** Indicated by angle brackets (< >).
3. **Parameter value options:** Descriptions of options that are available for the parameter.

Command details

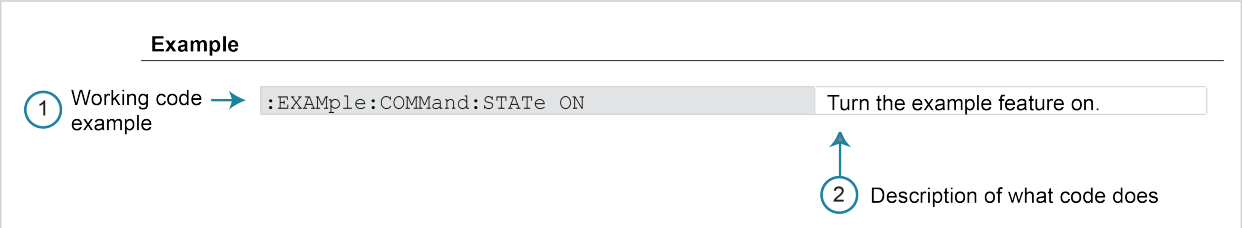
This section lists additional information you need to know to successfully use the command.

Figure 56: Details section of command listing

Example section

The Example section of the command description shows some simple examples of how the command can be used.

Figure 57: SCPI command description code examples



1. Example code that you can copy from this table and paste into your own application. Examples are generally shown using the short forms of the commands.
2. Description of the code and what it does. This may also contain the output of the code.

Related commands list

The **Also see** section of the remote command description provides links to commands that are related to the command.

Figure 58: SCPI related commands list example

Channel assignments

The 2750 has five slots for switching modules. To control the appropriate switching module, the slot number must be included with the switching module channel number when you specify a channel. The channel assignment is formatted as follows:

SCH

where:

- `S` is the slot number
- `CH` is the channel number

For example:

- `101` = Slot 1, Channel 1
- `210` = Slot 2, Channel 10
- `506` = Slot 5, Channel 6

Each switching module has a certain number of channels. For example, the Model 7700 switching module has 22 channels (1 through 22).

For remote operation, the three-digit channel assignment is included in the channel list parameter for the commands. The channel list is shown as `<clist>` and the format for the channel assignment is `(@SCH)`, where:

- `S` is the slot number (1, 2, 3, 4 or 5)
- `CH` is the switching module channel number (must be 2 digits)

Examples:

- `(@101)` = Slot 1, Channel 1
- `(@101, 203)` = Slot 1, Channel 1 and Slot 2, Channel 3
- `(@101:110)` = Slot 1, Channels 1 through 10

NOTE

The `<clist>` parameter is used to configure one or more channels for a scan. Each channel in the `<clist>` must be set to the function specified by the command. If not, a conflict error (−221) occurs. For example, `VOLTage:AC:DIGits 4.5, (@101)` is only valid if scan channel 101 is set for the ACV function.

SCPI reference tables

The following topics summarize the commands to operate the 2750. The subsystems are:

- CALCulate command summary
- DISPlay command summary
- FORMat command summary
- ROUTe command summary
- SENSe command summary
- STATus command summary
- SYSTem command summary
- TRACe command summary
- Trigger command summary
- UNIT command summary

General notes:

- Brackets (`[ ]`) denote optional character sets. These optional characters do not have to be included in the program message. Do not use brackets in the program message.
- Angle brackets (`< >`) indicate parameter type. Do not use angle brackets in the program message.

- The Boolean parameter (<b>) enables or disables an instrument operation. `1` or `ON` enables the operation. `0` or `OFF` disables the operation.
- Upper case characters indicate the short-form version for each command word.
- The listed default parameters are both the `*RST` and `SYSTem:PRESet` defaults, unless noted otherwise. Parameter notes are provided at the end of each table.
- The reference (Ref) column indicates where to find detailed information on the commands.
- A “yes” in the SCPI column indicates that the command and its parameters are SCPI confirmed. An unmarked command indicates that it is an SCPI command, but does not conform to the SCPI standard set of commands. It is not a recognized command by the SCPI consortium. SCPI confirmed commands that use one or more non-SCPI parameters are explained by notes.

CALCulate command summary

Command	Description	Default parameter	SCPI
`CALCulate[1]`	Subsystem to control CALC 1:		yes
`:FORMat <name> [<, clist>]`	Select math format (NONE, MXB, PERCent, or RECiprocal).	PERCent	yes
`:FORMat? [<clist>]`	Query math format.		yes
`:KMATh`	Path to configure math calculations:		
`:MMFactor <NRf> [, <clist>]`	Set "m" factor for mx+b (−4294967295 to +4294967295).[1]	1	
`:MA1Factor <NRf> [, <clist>]`	Set "m" factor for mx+b (−4294967295 to +4294967295).[1]	1	
`:MMFactor? [<clist>]`	Query "m" factor.[1]		
`:MA1Factor? [<clist>]`	Query "m" factor.[1]		
`:MBFactor <NRf> [, <clist>]`	Set "b" factor for mx+b (−4294967295 to +4294967295).[2]	0	
`:MA0Factor <NRf> [, <clist>]`	Set "b" factor for mx+b (−4294967295 to +4294967295).[2]	0	
`:MBFactor? [<clist>]`	Query "b" factor.[2]		
`:MA0Factor? [<clist>]`	Query "b" factor.[2]		
`:MUNits <char> [, <clist>]`	Specify units for mx+b reading.[3]	'X'	
`:MUNits? [, <clist>]`	Query "mx+b" units.		

Command	Description	Default parameter	SCPI
:PERCent <NRf> [, <clist>]	Set target value for PERCent calculation (−4294967295 to +4294967295).	1	
:ACQuire	Use input signal as target value.		
:PERCent? [<clist>]	Query percent.		
:STATe <b> [, <clist>]	Enable or disable kmath calculation.	(Note 4)	yes
:STATe? [< clist>]	Query state of kmath function.		yes
:DATA?	Read result of kmath calculation.		yes
:DATA:FRESh?	Read result of kmath calculation.		
CALCulate2	Subsystem to control CALC 2:		yes
:FORMat <name>	Select math format: (MEAN, SDEViation, MAXimum, MINimum, PKPK, or NONE).	NONE	yes
:FORMat?	Query math format.		yes
:STATe <b>	Enable or disable calculation.	(Note 4)	yes
:STATe?	Query state of math function.		yes
:IMMediate	Recalculate raw input data in buffer.		yes
:IMMediate?	Perform calculation and read result.		yes
:DATA?	Read math result of CALC 2.		yes
CALCulate3	Subsystem to control CALC 3 (limit test):		yes
:MLIMit	Path for master limit command:		
:LATChed <b>	Enable or disable master limit latch.	OFF	
:OUTPut	Path for limit output commands:		
[:STATe] <b>	Enable or disable limit outputs.	OFF	
[:STATe]?	Query state of limit outputs.		
:PULSe	Path to control limit output pulsing:		
[:STATe] <b>	Enable or disable limit output pulsing.	OFF	
[:STATe]?	Query state of limit output pulsing.		
:TIME <NRf>	Set output pulse time in seconds (0.001 to 99999.999).	0.002	
:TIME?	Query output pulse time.		
:LSENse <name>	Set logic sense of all limit lines (AHIGh or ALOW).	AHIGh	
:LSENse?	Query logic sense of limit lines.		
:LIMit1	Path to control LIMIT 1 test:		yes
:UPPer	Path to configure upper limit:		yes
[:DATA] <n> [, <clist>]	Set upper limit (−4294967295 to +4294967295).	1	yes
[:DATA]? [<clist>]	Query upper limit.		yes
:LOWer	Path to configure lower limit:		yes
[:DATA] <n> [, <clist>]	Set lower limit (−4294967295 to +4294967295).	−1	yes
[:DATA]? [<clist>]	Query lower limit.		yes
:STATe <b> [, <clist>]	Enable or disable limit test.	OFF	yes
:STATe? [<clist>]	Query state of limit test.		yes

Command	Description	Default parameter	SCPI
:FAIL?	Query test result (0 = failing).		yes
:CLEAR	Path to clear events:		yes
[:IMMediate]	Clear high and low events.		yes
:AUTO <b>	Enable or disable autoclear.	ON	yes
:AUTO?	Query autoclear.		yes
:LIMit2	Path to control LIMIT 2 test:		yes
:UPPer	Path to configure upper limit:		yes
[:DATA] <n> [, <clist>]	Set upper limit (−4294967295 to +4294967295).	2	yes
[:DATA]? [<clist>]	Query upper limit.		yes
:LOWer	Path to configure lower limit:		yes
[:DATA] <n> [, <clist>]	Set lower limit (−4294967295 to +4294967295).	−2	yes
[:DATA]? [<clist>]	Query lower limit.		yes
:STATe <b> [, <clist>]	Enable or disable limit test.	OFF	yes
:STATe? [<clist>]	Query state of limit test.		yes
:FAIL?	Query test result (1 = failing).		yes
:CLEAR	Path to clear events:		yes
[:IMMediate]	Clear high and low events.		yes
:AUTO <b>	Enable or disable autoclear.	ON	yes
:AUTO?	Query autoclear.		yes

Notes:
1. The :MMFactor and :MA1Factor commands perform the same operations.
2. The :MBFactor and :MA0Factor commands perform the same operations.
3. For mx+b units, <char> = one character; A through Z, degrees symbol (°), or ohms symbol (Ω).
4. ON is the *RST default parameter. OFF is the SYSTem:PRESet default.

DISPlay command summary

Command	Description	Default parameter	SCPI
DISPlay		(see Note)	
[:WINDow[1]]			yes
:TEXT	Path to control user text messages.		yes
:DATA <a>	Define ASCII message "a" (up to 12 characters).	(none)	yes
:DATA?	Query text message.		yes
:STATe <b>	Enable or disable message mode.	OFF	yes
:STATe?	Query text message state.		yes
:ENABle <b>	Enable or disable the front-panel display.	OFF	yes
:ENABle?	Query state of the display.		yes

Note: *RST and SYSTem:PRESet have no effect on commands in this subsystem. The listed defaults are power-on defaults.

FORMat command summary

Command	Description	Default parameter	SCPI
FORMat			
[:DATA] <type> [,<length>]	Select data format (ASCii, SREal, or DREal).	ASCii	
[:DATA]?	Query data format.		yes
:ELEMents <item list>	Specify data elements (READing, CHANnel, UNITs, RNUMber, TSTamp, and LIMits).	(see Note)	yes
:ELEMents?	Query data elements.		yes
:BORDer <name>	Select binary byte order (NORMal or SWAPped).	SWAPped	yes
:BORDer?	Query byte order.		yes
Note: The SYSTem:PRESet and *RST default is READ, UNIT, RNUM, and TST.			

ROUTe command summary

Command	Description	Default parameter	SCPI
ROUTe			
:MONitor <clist>	Specify one channel to be monitored.		
:STATe <b>	Enable or disable channel monitoring.	OFF	
:STATe?	Query state of channel monitoring.		
:DATA?	Returns the most recent monitor reading.		
:POINts <NRf>	For a monitor scan, specify number of channels to scan (2 to 110000).	(Note 1)	
:POINts?	For a monitor scan, query number of channels to scan.		
:MONitor?	Query the channel to be monitored.		
:CLOSe <clist>	Close the one specified channel (all others will open).		
:STATe? <clist>	Query closed channels in specified list; 1 = closed.		
:ACONfigure <b>	Enable or disable autoconfigure.	(Note 2)	
:ACONfigure?	Query state of autoconfigure.		
:COUNt? <clist>	Query closure count for specified channels.		
:INTerval <NRf>	Set count update interval in minutes (10 to 1440).	(Note 3)	
:INTerval?	Query relay count update interval.		
:CLOSe?	Returns a list of closed channels.		
:OPEN:ALL	Open all channels.		
:MULTiple	Path to control multiple channels:		
:OPEN <clist>	Open channels specified in list. Unlisted channels not affected.		

Command	Description	Default parameter	SCPI
:CLOSe <clist>	Close channels specified in list[4]. Unlisted channels not affected.		
:STATe? <clist>	Query closed channels in specified list; 1 = closed.		
:CLOSe?	Return list of all closed channels.		
ROUTe			
:SCAN	Path to configure scan:		yes
[:INTernal] <clist>	Specify list of channels to be scanned.		yes
[:INTernal]?	Query scan list.		yes
:TSOurce <list>	Select trigger source to start the scan (IMM, or HLIM1, HLIM2, LLIM1, and LLIM2).	IMM	
:TSOurce?	Query trigger source for scan.		
:NVOLatile <b>	Enable or disable nonvolatile memory for scanning (autoscan).	(Note 2)	
:NVOLatile?	Query nonvolatile memory setting.		
:LSELect <name>	Enable (INTernal) or disable (NONE) scan.	NONE	
:LSELect?	Query state of scan.		

Notes:
1. Default value depends on which switching module is installed.
2. Not affected by *RST and SYSTem:PRESet. Front-panel factory default is OFF.
3. Not affected by *RST and SYSTem:PRESet. Interval set to 15 minutes at the factory.
4. The ROUT:MULT:CLOS command cannot be used to measure thermocouple temperature using the internal or external reference junction. The simulated reference junction is used.

SENSe command summary

The following tables provide the commands in the SENSE subsystem, organized by measurement function.

[SENSe[1]] general commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:FUNCtion <name> [, <clist>]	Select function: 'VOLTage[:DC]', 'VOLTage :AC', 'CURRent[:DC]', 'CURRent:AC', 'RESistance', 'FRESistance', 'TEMPerature', 'FREQuency', 'PERiod', 'CONTinuity'.	VOLT:DC	yes
:FUNCtion? [<clist>]	Query function.		yes
:DATA[:LATest]?	Return the last reading.		yes
:DATA:FRESh?	Return the last fresh reading.		
:CAVerage <b> [, <clist>]	Channel average calculation:		
:DELay <NRf> [, <clist>]	Set delay between the two measurements in seconds (0 to 99999.999).[1]	0.5	
:DELay? [<clist>]	Query delay.		

Command	Description	Default parameter	SCPI
[:STATe] <b> [, <clist>]	Enable or disable channel average.	OFF	
[:STATe]? [, <clist>]	Query state of channel average.		
:RATio <b> [, <clist>]	Channel ratio calculation:	OFF	
:DELay <NRf> [, <clist>]	Set delay between the two measurements in seconds (0 to 99999.999).[1]	0.5	
:DELay? [<clist>]	Query delay.		
[:STATe] <b> [, <clist>]	Enable or disable channel ratio.	OFF	
[:STATe]? [, <clist>]	Query state of channel ratio.		
Note: CAVerage:DELay and RATio:DELay are coupled. Changing the delay for channel average also changes the delay for channel ratio, and vice versa.			

[SENSe[1]]:VOLTage[:DC] commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:VOLTage[:DC]	Path to configure DC voltage.		yes
:APERture <NRf> [, <clist>]	Set integration rate in seconds (60 Hz, 1.67e−4 to 1; 50 Hz, 2e−4 to 1).	(Note 2)	
:APERture? [<clist>]	Query aperture integration rate.		
:NPLCycles <n> [, <clist>]	Set integration rate in line cycles (60 Hz, 0.01 to 60; 50 Hz, 0.01 to 50).	5.0	yes
:NPLCycles? [<clist>]	Query line cycle integration rate.		yes
:RANGe	Path to set measurement range:		yes
[:UPPer] <n> [, <clist>]	Select range (0 to 1010).	1000	yes
[:UPPer]? [<clist>]	Query range.		yes
:AUTO <b> [, <clist>]	Enable or disable autorange.	ON	yes
:AUTO? [<clist>]	Query state of autorange.		yes
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	7	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference (−1010 to 1010).	0	yes
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	yes
:STATe? [<clist>]	Query state of reference.		yes
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		yes

Command	Description	Default parameter	SCPI
:AVERage	Path to configure and control filter:		
:TCONtrol <name>	Select filter type: (MOVing or REPeat).	(Note 3)	
:TCONtrol?	Query filter type.		
:WINDow <NRf>	Set filter window in % of range (0 to 10).	0.1	
:WINDow?	Query filter window.		
:COUNt <n> [, <clist>]	Specify filter count (1 to 100).	10	
:COUNt? [<clist>]	Query filter count.		
[:STATe] <b> [, <clist>]	Enable or disable filter.	(Note 4)	
[:STATe]? [<clist>]	Query state of digital filter.		
:IDIVider <b>	Enable or disable 10 MΩ input divider.	OFF	
:IDIVider?	Query state of input divider.		

Notes:
2. For 60 Hz line power, the default for aperture is 16.67 ms. For 50 Hz, the default is 20 ms.
3. REPeat is the *RST default. MOVing is the SYSTem:PRESet default. From the front panel, the factory default is MOVing.
4. OFF is the *RST default. ON is the SYSTem:PRESet default.

[SENSe[1]]:VOLTage:AC commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:VOLTage:AC	Path to configure AC voltage.		yes
:APERture <NRf> [, <clist>]	Set integration rate in seconds (60 Hz, 1.67e−4 to 1; 50Hz, 2e−4 to 1).	(Note 2)	
:APERture? [<clist>]	Query aperture integration rate.		
:NPLCycles <n> [, <clist>]	Set integration rate in line cycles (60 Hz, 0.01 to 60; 50 Hz, 0.01 to 50).	5.0	yes
:NPLCycles? [<clist>]	Query line cycle integration rate.		yes
:RANGe	Path to set measurement range:		yes
[:UPPer] <n> [, <clist>]	Select range (0 to 757.5).	750	yes
[:UPPer]? [<clist>]	Query range.		yes
:AUTO <b> [, <clist>]	Enable or disable autorange.	ON	yes
:AUTO? [<clist>]	Query state of autorange.		yes
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	6	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference (−757.5 to 757.5).	0	yes

Command	Description	Default parameter	SCPI
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	yes
:STATe? [<clist>]	Query state of reference.		yes
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		yes
:AVERage	Path to configure and control filter:		
:TCONtrol <name>	Select filter type: (MOVing or REPeat).	(Note 3)	
:TCONtrol?	Query filter type.		
:WINDow <NRf>	Set filter window in % of range (0 to 10).	0.1	
:WINDow?	Query filter window.		
:COUNt <n> [, <clist>]	Specify filter count (1 to 100).	10	
:COUNt? [<clist>]	Query filter count.		
[:STATe] <b> [, <clist>]	Enable or disable filter.	(Note 4)	
[:STATe]? [<clist>]	Query state of digital filter.		
:DETector	Path to set bandwidth:		
:BANDwidth <NRf> [, <clist>]	Set AC detector bandwidth in Hertz (3 to 3e5).	30	
:BANDwidth? [<clist>]	Query bandwidth.		

Notes:
2. For 60 Hz line power, the default for aperture is 16.67 ms. For 50 Hz, the default is 20 ms.
3. REPeat is the *RST default. MOVing is the SYSTem:PRESet default. From the front panel, the factory default is MOVing.
4. ON is the *RST default parameter, and OFF is the SYSTem:PRESet default.

[SENSe[1]]:CURRent[:DC] commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:CURRent[:DC]	Path to configure DC current.		yes
:APERture <NRf> [, <clist>]	Set integration rate in seconds (60 Hz, 1.67e-4 to 1; 50 Hz, 2e-4 to 1).	(Note 2)	
:APERture? [<clist>]	Query aperture integration rate.		
:NPLCycles <n> [, <clist>]	Set integration rate in line cycles (60Hz, 0.01 to 60; 50Hz, 0.01 to 50).	5.0	yes
:NPLCycles? [<clist>]	Query line cycle integration rate.		yes

Command	Description	Default parameter	SCPI
:RANGe	Path to set measurement range:		yes
[:UPPer] <n> [, <clist>]	Select range (0 to 3.1).	3	yes
[:UPPer]? [<clist>]	Query range.		yes
:AUTO <b> [, <clist>]	Enable or disable autorange.	ON	yes
:AUTO? [<clist>]	Query state of autorange.		yes
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	7	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference (-3.1 to 3.1).	0	yes
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	yes
:STATe? [<clist>]	Query state of reference.		yes
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		yes
:AVERage	Path to configure and control filter:		
:TCONtrol <name>	Select filter type: (MOVing or REPeat).	(Note 3)	
:TCONtrol?	Query filter type.		
:WINDow <NRf>	Set filter window in % of range (0 to 10).	0.1	
:WINDow?	Query filter window.		
:COUNt <n> [, <clist>]	Specify filter count (1 to 100).	10	
:COUNt? [<clist>]	Query filter count.		
[:STATe] <b> [, <clist>]	Enable or disable filter.	(Note 4)	
[:STATe]? [<clist>]	Query state of digital filter.		

Notes:
2. For 60 Hz line power, the default for aperture is 16.67 ms. For 50 Hz, the default is 20 ms.
3. REPeat is the *RST default. MOVing is the SYSTem:PRESet default. From the front panel, the factory default is MOVing.
4. OFF is the *RST default. ON is the SYSTem:PRESet default.

[SENSe[1]]:CURRent:AC commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:CURRent:AC	Path to configure AC current.		yes
:APERture <NRf> [, <clist>]	Set integration rate in seconds (60 Hz, 1.67e-4 to 1; 50 Hz, 2e-4 to 1).	(Note 2)	
:APERture? [<clist>]	Query aperture integration rate.		
:NPLCycles <n> [, <clist>]	Set integration rate in line cycles (60Hz, 0.01 to 60; 50Hz, 0.01 to 50).	5.0	yes
:NPLCycles? [<clist>]	Query line cycle integration rate.		yes
:RANGe	Path to set measurement range:		yes
[:UPPer] <n> [, <clist>]	Select range (0 to 3.1).	3	yes
[:UPPer]? [<clist>]	Query range.		yes
:AUTO <b> [, <clist>]	Enable or disable autorange.	ON	yes
:AUTO? [<clist>]	Query state of autorange.		yes
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	6	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference (-3.1 to 3.1).	0	yes
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	yes
:STATe? [<clist>]	Query state of reference.		yes
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		yes
:AVERage	Path to configure and control filter:		
:TCONtrol <name>	Select filter type: (MOVing or REPeat).	(Note 3)	
:TCONtrol?	Query filter type.		
:WINDow <NRf>	Set filter window in % of range (0 to 10).	0.1	
:WINDow?	Query filter window.		
:COUNt <n> [, <clist>]	Specify filter count (1 to 100).	10	
:COUNt? [<clist>]	Query filter count.		
[:STATe] <b> [, <clist>]	Enable or disable filter.	(Note 4)	
[:STATe]? [<clist>]	Query state of digital filter.		

Command	Description	Default parameter	SCPI
:DETector	Path to set bandwidth:		
:BANDwidth <NRf> [, <clist>]	Set AC detector bandwidth in Hertz (3 to 3e5).	30	
:BANDwidth? [<clist>]	Query bandwidth.		
:RESistance	Path to configure resistance.		yes
:APERture <NRf> [, <clist>]	Set integration rate in seconds (60Hz, 1.67e-4 to 1; 50Hz, 2e-4 to 1).	(Note 2)	
:APERture? [<clist>]	Query aperture integration rate.		
:NPLCycles <n> [, <clist>]	Set integration rate in line cycles (60Hz, 0.01 to 60; 50Hz, 0.01 to 50).	5.0	yes
:NPLCycles? [<clist>]	Query line cycle integration rate.		yes
:RANGe	Path to set measurement range:		yes
[:UPPer] <n> [, <clist>]	Select range (0 to 120e6).	120e6	yes
[:UPPer]? [<clist>]	Query range.		yes
:AUTO <b> [, <clist>]	Enable or disable autorange.	ON	yes
:AUTO? [<clist>]	Query state of autorange.		yes
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	7	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference (0 to 120e6).	0	yes
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	yes
:STATe? [<clist>]	Query state of reference.		yes
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		yes
:AVERage	Path to configure and control filter:		
:TCONtrol <name>	Select filter type: (MOVing or REPeat).	(Note 3)	
:TCONtrol?	Query filter type.		
:WINDow <NRf>	Set filter window in % of range (0 to 10).	0.1	
:WINDow?	Query filter window.		
:COUNt <n> [, <clist>]	Specify filter count (1 to 100).	10	
:COUNt? [<clist>]	Query filter count.		

Command	Description	Default parameter	SCPI
[:STATe] <b> [, <clist>]	Enable or disable filter.	(Note 4)	
[:STATe]? [<clist>]	Query state of digital filter.		

Notes:
2. For 60 Hz line power, the default for aperture is 16.67 ms. For 50 Hz, the default is 20 ms.
3. REPeat is the *RST default. MOVing is the SYSTem:PRESet default. From the front panel, the factory default is MOVing.
4. OFF is the *RST default. ON is the SYSTem:PRESet default.

[SENSe[1]]:RESistance commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:RESistance	Path to configure resistance.		yes
:APERture <NRf> [, <clist>]	Set integration rate in seconds (60 Hz, 1.67e-4 to 1; 50 Hz, 2e-4 to 1).	(Note 2)	
:APERture? [<clist>]	Query aperture integration rate.		
:NPLCycles <n> [, <clist>]	Set integration rate in line cycles (60 Hz, 0.01 to 60; 50 Hz, 0.01 to 50).	5.0	yes
:NPLCycles? [<clist>]	Query line cycle integration rate.		yes
:RANGe	Path to set measurement range:		yes
[:UPPer] <n> [, <clist>]	Select range (0 to 120e6).	120e6	yes
[:UPPer]? [<clist>]	Query range.		yes
:AUTO <b> [, <clist>]	Enable or disable autorange.	ON	yes
:AUTO? [<clist>]	Query state of autorange.		yes
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	7	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference (0 to 120e6).	0	yes
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	yes
:STATe? [<clist>]	Query state of reference.		yes
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		yes
:AVERage	Path to configure and control filter:		
:TCONtrol <name>	Select filter type: (MOVing or REPeat).	(Note 3)	
:TCONtrol?	Query filter type.		

Command	Description	Default parameter	SCPI
:WINDow <NRf>	Set filter window in % of range (0 to 10).	0.1	
:WINDow?	Query filter window.		
:COUNt <n> [, <clist>]	Specify filter count (1 to 100).	10	
:COUNt? [<clist>]	Query filter count.		
[:STATe] <b> [, <clist>]	Enable or disable filter.	(Note 4)	
[:STATe]? [<clist>]	Query state of digital filter.		

Notes:
2. For 60 Hz line power, the default for aperture is 16.67 ms. For 50 Hz, the default is 20 ms.
3. REPeat is the *RST default. MOVing is the SYSTem:PRESet default. From the front panel, the factory default is MOVing.
4. OFF is the *RST default. ON is the SYSTem:PRESet default.

[SENSe[1]]:FRESistance commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:FRESistance	Path to configure four-wire resistance.		yes
:APERture <NRf> [, <clist>]	Set integration rate in seconds (60 Hz, 1.67e-4 to 1; 50 Hz, 2e-4 to 1).	(Note 2)	
:APERture? [<clist>]	Query aperture integration rate.		
:NPLCycles <n> [, <clist>]	Set integration rate in line cycles (60 Hz, 0.01 to 60; 50 Hz, 0.01 to 50).	5.0	yes
:NPLCycles? [<clist>]	Query line cycle integration rate.		yes
:RANGe	Path to set measurement range:		yes
[:UPPer] <n> [, <clist>]	Select range (0 to 120e6).	120e6	yes
[:UPPer]? [<clist>]	Query range.		yes
:AUTO <b> [, <clist>]	Enable or disable autorange.	ON	yes
:AUTO? [<clist>]	Query state of autorange.		yes
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	7	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference (0 to 120e6).	0	yes
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	yes
:STATe? [<clist>]	Query state of reference.		yes

Command	Description	Default parameter	SCPI
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		yes
:AVERage	Path to configure and control filter:		
:TCONtrol <name>	Select filter type: (MOVing or REPeat).	(Note 3)	
:TCONtrol?	Query filter type.		
:WINDow <NRf>	Set filter window in % of range (0 to 10).	0.1	
:WINDow?	Query filter window.		
:COUNt <n> [, <clist>]	Specify filter count (1 to 100).	10	
:COUNt? [<clist>]	Query filter count.		
[:STATe] <b> [, <clist>]	Enable or disable filter.	(Note 4)	
[:STATe]? [<clist>]	Query state of digital filter.		
:OCOMpensated <b> [, <clist>]	Enable or disable offset compensation.	OFF	
:OCOMpensated? [<clist>]	Query state of offset compensation.		
:DCIRcuit<b> [, <clist>]	Enable or disable dry circuit ohms.	OFF	
:DCIRcuit? [, <clist>]	Query state of dry circuit ohms.		

Notes:
2. For 60 Hz line power, the default for aperture is 16.67 ms. For 50 Hz, the default is 20 ms.
3. REPeat is the *RST default. MOVing is the SYSTem:PRESet default. From the front panel, the factory default is MOVing.
4. OFF is the *RST default. ON is the SYSTem:PRESet default.

[SENSe[1]]:TEMPerature commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:TEMPerature	Path to configure temperature:		yes
:APERture <NRf> [, <clist>]	Set integration rate in seconds (60 Hz, 1.67e−4 to 1; 50 Hz, 2e−4 to 1).	(Note 2)	
:APERture? [<clist>]	Query aperture integration rate.		
:NPLCycles <n> [, <clist>]	Set integration rate in line cycles (60 Hz, 0.01 to 60; 50 Hz, 0.01 to 50).	5.0	yes
:NPLCycles?	Query line cycle integration rate.		yes
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	6	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference in °C (−200 to 1820).	0	yes

Command	Description	Default parameter	SCPI
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	yes
:STATe? [<clist>]	Query state of reference.		yes
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		yes
:AVERage	Path to configure and control filter:		
:TCONtrol <name>	Select filter type: (MOVing or REPeat).	(Note 3)	
:TCONtrol?	Query filter type.		
:WINDow <NRf>	Set filter window in % of range (0 to 10).	0.1	
:WINDow?	Query filter window.		
:COUNt <n> [, <clist>]	Specify filter count (1 to 100).	10	
:COUNt? [<clist>]	Query filter count.		
[:STATe] <b> [, <clist>]	Enable or disable filter.	(Note 4)	
[:STATe]? [<clist>]	Query state of digital filter.		
:TRANsducer <name> [, <clist>]	Select temperature transducer (TCouple, FRTD, or THERmistor).	TCouple	
:TRANsducer? [<clist>]	Query transducer type.		
:TCouple	Path to configure thermocouple:[5]		
:TYPE <type> [, <clist>]	Select T/C type (B, E, J, K, N, R, S, or T.	K	
:TYPE? [<clist>]	Query T/C type.		
:ODETect <b>	Enable or disable T/C open detector.	OFF	
:ODETect?	Query state of T/C open detector.		
:RJUNction	Path to configure reference junction:[5, 6]		
:RSELect <name> [, <clist>]	Select reference junction (SIMulated, INTernal, or EXTernal).	(Note 7)	
:RSELect? [<clist>]	Query reference junction.		
:SIMulated <n> [, <clist>]	Set simulated reference temperature; 0 to 65 (°C), 32 to 149 (°F), or 273 to 338 (K).	23	
:SIMulated? [<clist>]	Query simulated reference temperature.		
:THERmistor	Path to configure thermistor:		
[:TYPE] <NRf> [, <clist>]	Set thermistor type in ohms (1950 to 10050).	5000	
[:TYPE]? [<clist>]	Query thermistor type.		

Command	Description	Default parameter	SCPI
:FRTD	Path to configure 4-wire RTD;		
:TYPE <name> [, <clist>]	Select FRTD type (PT100, D100, F100, PT3916, PT385, or USER.	PT100	
:TYPE? [<clist>]	Query FRTD type.		
:RZERo <NRf> [, <clist>]	Specify constant for USER type (0 to 10000).	100	
:RZERo? [<clist>]	Query rzero.		
:ALPHa <NRf> [, <clist>]	Specify constant for USER type (0 to 0.01).	0.00385	
:ALPHa? [<clist>]	Query alpha.		
:BETA <NRf> [, <clist>]	Specify constant for USER type (0 to 1.00).	0.111	
:BETA? [<clist>]	Query beta.		
:DELTa <NRf> [, <clist>]	Specify constant for USER type (0 to 5.00).	1.507	
:DELTa? [<clist>]	Query delta.		

Notes:

2. For 60 Hz line power, the default for aperture is 16.67 ms. For 50 Hz, the default is 20 ms.
3. REPeat is the *RST default. MOVing is the SYSTem:PRESet default. From the front panel, the factory default is MOVing.
4. OFF is the *RST default. ON is the SYSTem:PRESet default.
5. The following commands can instead be used to select the reference junction and set the simulated reference temperature:

   TEMPerature:RJUNction:RSELect <name> [, <clist>]. Select reference junction: SIMulated, INTernal, or EXTernal.

   TEMPerature:RJUNction:RSELect? [<clist>]. Query reference junction.

   TEMPerature:RJUNction:SIMulated <n> [, <clist>]. Set simulated reference temperature: 0 to 50 (°C), 32 to 122 (°F), or 273 to 323 (K).

   TEMPerature:RJUNction:SIMulated? [<clist>]. Query simulated reference temperature.
6. When using multiple channel operation (ROUT:MULT command) to connect a switching module channel to the DMM for thermocouple temperature measurements, the SIMulated reference junction is used if the INTernal or EXTernal reference junction is selected.
7. With a Model 7700, 7706, or 7708 installed, the default sensor junction is Internal. Otherwise, the simulated (23 °C) junction is selected.

[SENSe[1]]:FREQuency commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:FREQuency	Path to configure frequency.		
:APERture <NRf> [, <clist>]	Sets gate time for frequency measurements in seconds (0.01 to 1.0).	1.0	
:APERture? [<clist>]	Query frequency gate time.		
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	7	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference (0 to 1.5e7).	0	
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	
:STATe? [<clist>]	Query state of reference.		
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		
:THReshold	Path to select the threshold voltage range:		yes
:VOLTage			yes
:RANGe <n> [, <clist>]	Select threshold range (0 to 1010).	10	
:RANGe? [<clist>]	Query threshold range.		

[SENSe[1]]:PERiod commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:PERiod	Path to configure period:		
:APERture <NRf> [, <clist>]	Sets gate time for period measurements in seconds (0.01 to 1.0).	1.0	
:APERture? [<clist>]	Query period gate time.		
:DIGits <n> [, <clist>]	Specify measurement resolution (4 to 7).	7	
:DIGits? [<clist>]	Query resolution.		
:REFerence <n> [, <clist>]	Specify reference (0 to 1).	0	
:STATe <b> [, <clist>]	Enable or disable reference.	OFF	
:STATe? [<clist>]	Query state of reference.		

Command	Description	Default parameter	SCPI
:ACQuire [, <clist>]	Use input signal as reference.		
:REFerence? [<clist>]	Query reference value.		
:THReshold	Path to select the threshold voltage range:		
:VOLTage			
:RANGe <n> [, <clist>]	Select threshold range (0 to 1010).	10	
:RANGe? [<clist>]	Query threshold range.		

[SENSe[1]]:CONTinuity commands

Command	Description	Default parameter	SCPI
[SENSe[1]]			
:CONTinuity	Path to configure continuity test:		
:THReshold <NRf>	Set threshold resistance in ohms (1 to 1000).	10	
:THReshold?	Query threshold resistance.		

STATus command summary

Command	Description	Default parameter	SCPI
STATus		(Note 1)	yes
:MEASurement	Path to control measurement event registers:		
[:EVENt]?	Read the event register.	(Note 2)	
:ENABle <NRf>	Program the enable register.	(Note 3)	
:ENABle?	Read the enable register.		
:CONDition?	Read the condition register.		
:OPERation	Path to control operation status registers:		yes
[:EVENt]?	Read the event register.	(Note 2)	yes
:ENABle <NRf>	Program the enable register.	(Note 3)	yes
:ENABle?	Read the enable register.		yes
:CONDition?	Read the condition register.		yes
:QUEStionable	Path to control questionable status registers:		yes
[:EVENt]?	Read the event register.	(Note 2)	yes
:ENABle <NRf>	Program the enable register.	(Note 3)	yes
:ENABle?	Read the enable register.		yes
:CONDition?	Read the condition register.		yes
:PRESet	Return status registers to default states.		yes

Command	Description	Default parameter	SCPI
:QUEue	Path to access error queue:		yes
[:NEXT]?	Read the most recent error message.	(Note 4)	yes
:ENABle <list>	Specify error and status messages for queue.	(Note 5)	yes
:ENABle?	Read the enabled messages.		yes
:DISable <list>	Specify messages not to be placed in queue.	(Note 5)	
:DISable?	Read the disabled messages.		
:CLEar	Clears all messages from Error Queue.		

Notes:
1. Commands in this subsystem are not affected by *RST and SYSTem:PRESet. The effects of cycling power, *CLS and STATus:PRESet are explained by the following notes.
2. Event Registers: Power-up and *CLS clear all bits of the registers. STATus:PRESet has no effect.
3. Enable Registers: Power-up and STATus:PRESet clear all bits of the registers. *CLS has no effect.
4. Error Queue: Power-up and *CLS clear the Error Queue. STATus:PRESet has no effect.
5. Enable/Disable Error Queue Messages. Power-up clears list of messages. *CLS and STATus:PRESet have no effect.

SYSTem command summary

Command	Description	Default parameter	SCPI
SYSTem			
:PRESet	Return to defaults optimized for front-panel operation.		yes
:POSetup <name>	Select power-on setup: (RST, PRESet, SAV0, SAV1, or SAV2).	RST	
:POSetup?	Query power-on setup.		
:FRSWitch?	Query INPUTS switch (0=rear, 1=front).		
:BEEPer	Path to control beeper.		
[:STATe] <b>	Enable or disable beeper.	ON	yes
[:STATe]?	Query state of beeper.		yes
:KCLick <b>	Turn the keyclick on/off.	ON	
:KCLick?	Query the keyclick status.		
:KEY <NRf>	Simulate key-press (1 to 31).		yes
:KEY?	Query the last pressed key.		yes
:AZERo	Path to set up autozero.		
:STATe <b>	Enable or disable autozero.	ON	
:STATe?	Query autozero.		
:LSYNc	Path to control line synchronization:		
[:STATe] <b>	Enable or disable line-sync.	OFF	
[:STATe]?	Query state of line-sync.		
:LFRequency?	Query power line frequency.		
:PCARdX <name>	Set up an empty slot (X = 1 to 5) as a pseudocard (C7700, C7701, C7702, C7703, C7705, C7706, C7707, 7708, C7709).		

Command	Description	Default parameter	SCPI
SYSTem			
:CARD	Path to query switching module in specified slot.[1, 2] For the following :CARD commands, <NRf> = slot number for module.		
:SNUMber? <NRf>	Request serial number[3].		
:SWRevision? <NRf>	Request firmware revision[3].		
:VMAX? <NRf>	Request maximum allowable voltage.		
:MUX? <NRf>	Support multiplexer channels?; 1 = yes, 0 = no.		
:ISOLated? <NRf>	Support isolated channels?; 1 = yes, 0 = no.		
:TCOMpensated? <NRf>	Built-in temperature sensors for T/C cold junction?; 1 = yes, 0 = no.		
:VCHannel	Path to query volts/2-wire channels:		
[:STARt]? <NRf>	Request lowest numbered volts/2-wire channel (usually 1); 0 = voltage measurements not supported.		
:END? <NRf>	Request highest numbered volts/2-wire channel. 0 = voltage measurements not supported.		
:ACHannel	Path to query amps channels:		
[:STARt]? <NRf>	Request lowest numbered amps channel; 0 = amps measurements not supported.		
:END? <NRf>	Request highest numbered amps channel; 0 = amps measurements not supported.		
:AOUTput	Path to query analog output channels:		
[:STARt]? <NRf>	Request highest numbered analog output channel; 0 = analog output not supported.		
:END? <NRf>	Request lowest numbered analog output channel; 0 = analog output not supported.		
:DOUTput	Path to query digital output channels:		
[:STARt]? <NRf>	Request lowest numbered digital output channel; 0 = digital output not supported.		
:END? <NRf>	Request highest numbered digital output channel; 0 = digital output not supported.		
:TCHannel? <NRf>	Query the totalizer channel.		
:DINPut	Path to query digital input channels:		
[:STARt]?	Request lowest numbered digital input port; 0=digital input not supported.		
:END	Request highest numbered digital output channel; 0 = digital input not supported.		
:SNOpen?	Query whether card is single no-open type, such as the 7711: 1 = yes, 0 = no.		
:BANKs?	For a single no-open card, query the number of banks. If the card is not the single no-open type, error −221, settings conflict, results.		
:SWOpen?	Query whether the card is single with-open type such as 7711; 1 = yes, 0 = no.		
:BANKs?	For single with-open card, query number of banks.		
:CSOhms?	Query whether card supports commonside 4-wire ohms, such as the 7701.		

Command	Description	Default parameter	SCPI
SYSTem			
:TIME <hr, min, sec>	Set system time using 24-hour format.		
:TIME?	Query system time.		
:DATE <yr, mo, day>	Set system date (yr = 1999 or 20xx).		
:DATE?	Query system date.		
:TSTamp	Path to set timestamp:		
:TYPE <name>	Select timestamp type (RELative or RTCLock).	REL	
:TYPE?	Query timestamp type that will be used for the next buffer storage.		
:RELative	Path to reset relative timestamp:		
:RESet	Reset relative timestamp to 0.		
:RNUMber	Path to reset reading number:		
:RESet	Reset reading number. Next reading is #1.		
:ERRor?	Query (read) Error Queue.	(Note 4)	yes
:CLEar	Clears messages from the Error Queue.		
:VERSion?	Query rev level of SCPI standard.		yes
:LOCal	Take 2750 out of remote and restore operation of front-panel controls (RS-232 only).		
:REMote	Place 2750 in remote (RS-232 only).		
:RWLock	Lockout front-panel controls (RS-232 only).		

Note:
1. If there is no card in the specified slot, error -241, Hardware Missing, occurs.
2. The Model 2700 uses a different syntax to specify a slot for the :CARD command. Instead of using the <NRf> parameter, the slot number is included with the :CARD command word (for example, SYST:CARD1:SNUM?). To make the 2750 compatible with 2700 operation, the :CARD command for the 2750 accepts the syntax for slots 1 and 2.
3. If a pseudocard is installed in the slot, the message ???????? is returned when querying the serial number or firmware revision.
4. Power-up and *CLS clears the error queue. *RST, SYSTem:PRESet, and STATus:PRESet has no effect on the error queue.

TRACe command summary

Command	Description	Default parameter*	SCPI
TRACe|:DATA	Use TRACe or DATA as root command.		
:CLEar	Path to clear the buffer.		
[:IMMediate]	Clear the buffer.		
:AUTO <b>	Enable or disable buffer autoclear.	ON	
:AUTO?	Query state of buffer autoclear.		
:FREE?	Query bytes available and bytes in use.		yes
:POINts <NRf>	Specify size of buffer (2 to 110000).	100	yes
:POINts?	Query buffer size.		yes
:NOTify <NRf>	Specify number of stored readings that will set Trace Notify bit (B6) of measurement event register (2 to 109999). Must be less than TRACe:POINts value.	50	
:NOTify?	Query trace notify value.		

Command	Description	Default parameter*	SCPI
:NEXT?	Query buffer location for next stored reading.		
:TSTamp	Path to set timestamp format:		
:FORMat <name>	Select timestamp format (ABSolute or DELTa).	ABS	
:FORMat?	Query timestamp format.		
:TYPE?	Query timestamp type for readings presently in buffer.		
:FEED <name>	Select source of readings (SENSe[1], CALCulate[1], or NONE).	CALC	yes
:CONTrol <name>	Select buffer control mode (NEVer, NEXT, or ALWays).	NEV	yes
:CONTrol?	Query buffer control mode.		yes
:FEED?	Query source of readings for buffer.		yes
:DATA?	Read all readings in the buffer.		yes
:DATA:SELected? <start>, <count>	Specify readings to be returned; specify starting point (first reading is #0) and number of readings (count).		
*SYSTem:PRESet and *RST have no effect on commands in this subsystem. The listed defaults are defaults set at the factory.			

TRIGger command summary

Command	Description	Default parameter	SCPI
INITiate	Subsystem command path:		yes
[:IMMediate]	Initiate one trigger cycle.		yes
:CONTinuous <b>	Enable or disable continuous initiation.	(Note 1)	
:CONTinuous?	Query continuous initiation.		yes
ABORt	Reset trigger system.		yes
TRIGger[:SEQuence[1]]	Path to program Trigger Layer:		yes
:COUNt <n>	Set measure count (1 to 110000 or INFinity).	(Note 2)	yes
:COUNt?	Query measure count.		yes
:DELay <n>	Set delay (0 s to 999999.999 s).	0	yes
:AUTO <b>	Enable or disable autodelay.	OFF	
:AUTO?	Query state of delay.		
:DELay?	Query delay.		yes
:SOURce <name>	Select control source (IMMediate, TIMer, MANual, BUS, or EXTernal).	IMM	yes
:SOURce?	Query control source.		yes
:TIMer <n>	Set timer interval (0.001 s to 999999.999 s).	0.1	yes
:TIMer?	Request the programmed timer interval.		yes
:SIGNal	Loop around control source.		yes

Command	Description	Default parameter	SCPI
SAMPle			
:COUNt <NRf>	Specify sample count (1 to 110000).	1	
:COUNt?	Query sample count.		
Notes: 1. Defaults for continuous initiation: SYSTem:PRESet enables continuous initiation. *RST disables continuous initiation. 2. Defaults for count: SYSTem:PRESet sets the count to INF (infinity). *RST sets the count to 1.			

UNIT command summary

Command	Description	Default parameter	SCPI
UNIT			
:TEMPerature <UnitofMeasure>	Select temperature units (C, CEL, F, FAR, or K).	C	yes
:TEMPerature?	Query temperature units.		yes
:VOLTage	Path to configure voltage units.		
[:DC] <name> [, <clist>]	Select DCV measurement units (V or DB).	V	
:DB	Path to set DB reference voltage:		
:REFerence <n>	Specify reference in volts (1e−7 to 1000).	1	
:REFerence?	Query reference.		
[:DC]? [<clist>]	Query DCV units.		
:AC <name> [, <clist>]	Select ACV measurement units (V or DB).	V	
:DB	Path to set DB reference voltage.		
:REFerence <n>	Specify reference in volts (1e−7 to 1000).	1	
:REFerence?	Query DB reference.		
:AC? [<clist>]	Query ACV units.		

:CONFigure?

This command places the instrument in one-shot measurement mode.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Not applicable

Usage

```
:CONFigure?
:CONFigure:<function>?
:CONFigure:<function>? <rang>
:CONFigure:<function>? <rang>, <res>
:CONFigure:<function>? <rang>, <clist>
```

`<function>`	The function: ▪ DC voltage: `VOLTage[:DC]` ▪ AC voltage: `VOLTage:AC` ▪ DC current: `CURRent[:DC]` ▪ AC current: `CURRent:AC` ▪ Ω2 resistance: `RESistance` ▪ Ω4 resistance: `FRESistance` ▪ Frequency: `FREQuency` ▪ Period: `PERiod` ▪ Temperature: `TEMPerature` ▪ Continuity: `CONTinuity?`
`<rang>`	Range parameter for the specified function; refer to **Details**
`<res>`	Resolution of the reading; refer to **Details**
`<clist>`	The channel to be closed and measured; single channels only; refer to Channel assignments (on page 3-4)

Details

This command places the instrument in a one-shot measurement mode. You can then use the `READ?` command to trigger a measurement and acquire a reading.

When this command is sent without the `<clist>` parameter, the 2750 is configured as follows:

- The function specified by this command is selected. If specified, range and resolution are also set.
- All controls related to the selected function are defaulted to the `*RST` values.
- Continuous initiation is disabled (`INITiate:CONTinuous OFF`).
- The control source of the trigger model is set to Immediate.
- The count values of the trigger model are set to one.
- The delay of the trigger model is set to zero.
- The 2750 is placed in the idle state.

- All math calculations are disabled.
- Buffer operation is disabled. A storage operation presently in process is aborted.
- Autozero is enabled.

The range sets the range of the function. For example, if the range parameter is set to 10, the 10 V range is selected.

The resolution and the range set the number of display digits. Some examples are shown in the following table.

Resolution setting	Example	Equivalent digits
0.1	100.0	3½
0.01	10.00	3½
0.001	1.000	3½
0.0001	1.0000	4½
0.00001	1.00000	5½
0.000001	1.000000	6½

As shown in the table, with the 100 V range selected and resolution set to 0.1, a 100 V reading is displayed as 100.0 V (3½ digits).

The display defaults to 3½ digits when using parameter values that attempt to set the display to less than 3½ digits. For example, a 10 V reading with a resolution set to 0.1 for the 10 V range is displayed as 10.00 V, not 10.0 V.

For the TEMPerature and CONTinuity functions, you cannot use the `<rang>` and `<res>` parameters. The command is ignored and causes error −108, parameter not allowed.

A command that uses parameter values that attempt to set the display above 7½ digits is ignored and generates error −221, settings conflict error.

This command cannot be used when scanning. The command is ignored and causes error −221, settings conflict.

When the channel list parameter (`<clist>`) is included, the specified channels for the scanlist assumes the `*RST` default settings for the specified function. Range can also be set for the channels by including the <rang> parameter. If the resolution parameter (`<res>`) is included, it is ignored. Resolution for the scanlist channels is determined by the present setting for the specified function and by the present resolution setting for the specified function. The present measurement function and the trigger model settings are not affected when the CONFigure command is sent with the `<clist>` parameter.

Example 1

This example configures scanlist channels 101 through 105 for 4-wire resistance measurements on the 1 MΩ range.

```
CONF:FRES 1e6, (@101:105)
```

Example 2

This example selects the dc voltage function using the 10 V range with 3½ digit display resolution.

```
MEAS:VOLT? 10, 0.01
```

Also see

:FETCh? (on page 9-32)
:READ? (on page 9-37)

:FETCh?

This command requests the latest post-processed reading.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:FETCh?
```

Details

This command is typically used with the instrument in one-shot measurement mode to trigger and acquire a specified number of readings. The `SAMPle:COUNt` command is used to specify the number of readings.

After sending this command and addressing the 2750 to talk, the reading is sent to the computer. This command does not affect the instrument setup. This command does not trigger a measurement. The command requests the last available reading. The reading includes any operations, such as math calculations, that were applied.

Similar to `DATA[:LATest]?`, `FETCh?` can be used to return the last reading. However, it can also be used to return more than one reading. When returning more than one reading, the readings are automatically stored in the buffer. To return multiple reading strings, continuous initiation must be disabled (INIT:CONT OFF) so that the sample count (SAMPle:COUNt), which specifies the number of measurements to be made, can be set to more than 1. After INITiate is sent to trigger the measurements, `FETCh?` returns the reading strings.

If the instrument does not have a reading available (indicated by dashes in the display), sending this command generates the error –230, Data corrupt or stale.

This query does not trigger a reading or wait for a result if a reading is in progress. It is possible to get the same reading repeatedly using this query. It continues to give the same result until one of the following things has happened:

- A new reading has been triggered
- The old reading has been invalidated by changing ranges or by changing function.

`DATA:FRESh?` is often a better query to use. If `FETCh?` is used, the following conditions should be met:

- A reading has been triggered, by either continuous measurements (`:INIT:CONT ON` and `:TRIG:SOUR IMM`), by some event such as a bus trigger (`*TRG`), or by an external trigger (`:TRIG:SOUR EXT`).

- It is confirmed that the reading is completed by either the setting of the RAV bit in the status model or by allowing sufficient time to pass for the reading to complete.

`FETCh?` is automatically asserted when the `READ?` or `MEASure?` command is sent.

When this command is sent, the following commands execute in this order:

1. `ABORt`
2. `INITiate`
3. `FETCh?`

When `ABORt` is executed, the instrument goes into the idle state if continuous initiation is disabled. If continuous initiation is enabled, the operation restarts at the beginning of the trigger model.

NOTE

`FETCh?` can repeatedly return the same reading. Until there is a new reading, this command continues to return the old reading.

When an instrument setting that is relevant to the readings in the sample buffer is changed, the `FETCh?` command causes error −230, data corrupt or stale, or a bus timeout to occur. To get `FETCh?` working again, a new reading must be triggered.

Also see

:INITiate:CONTinuous (on page 9-33)
:MEASure? (on page 9-35)
:READ? (on page 9-37)
[:SENSe[1]]:DATA:FRESh? (on page 9-99)
:TRIGger[:SEQuence[1]]:SAMPle:COUNt (on page 9-210)

:INITiate:CONTinuous

This command enables or disables continuous initiation.

Type	Affected by	Where saved	Default value
Command and query	SYSTem:PRESet Reset	Not applicable	See **Details**

Usage

```
:INITiate:CONTinuous <b>
:INITiate:CONTinuous?
```

`<b>`	Disable continuous initiation: `OFF` Enable continuous initiation: `ON`

Details

When continuous initiation is selected (`ON`), the instrument is taken out of the idle state. The instrument continuously performs and displays measurements. At the conclusion of all programmed operations, the instrument returns to Arm Layer 1.

With continuous initiation enabled, you cannot use the READ? command or set sample count (SAMPle:COUNt) greater than one.

Defaults for continuous initiation:

- `SYSTem:PRESet` enables continuous initiation.
- `*RST` disables continuous initiation.

SCPI compliant.

Also see

None

:INITiate[:IMMediate]

This command takes the 2750 out of the idle state and initiates a trigger cycle.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:INITiate[:IMMediate]
```

Details

After all programmed operations are completed, the instrument leaves the trigger layers and returns to the idle state if continuous initiation is disabled.

This command is only available when `:INITiate:CONTinuous` is disabled (OFF).

This command cannot be used if the sample count is more than 1.

Sending this command when the instrument is making measurements generates error message −213, init ignored.

SCPI compliant.

Also see

:INITiate:CONTinuous (on page 9-33)

:MEASure?

This command combines the signal-oriented measurement commands to perform a one-shot measurement and acquire the post-processed reading.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:MEASure?
:MEASure:<function>?
:MEASure:<function>? <range>
:MEASure:<function>? <range>, <res>
:MEASure:<function>? <range>, <res>, <clist>
```

`<function>`	The function: ▪ DC voltage: `VOLTage[:DC]` ▪ AC voltage: `VOLTage:AC` ▪ DC current: `CURRent[:DC]` ▪ AC current: `CURRent:AC` ▪ Ω2 resistance: `RESistance` ▪ Ω4 resistance: `FRESistance` ▪ Frequency: `FREQuency` ▪ Period: `PERiod` ▪ Temperature: `TEMPerature` ▪ Continuity: `CONTinuity?`
`<range>`	Range parameter for the specified function; refer to **Details**
`<res>`	Resolution of the reading; refer to **Details**
`<clist>`	The channel to be closed and measured; single channels only; refer to Channel assignments (on page 3-4)

Details

This query reconfigures the instrument to the function specified in the query, sets the trigger source for immediate, sets the trigger count to 1, and configures the measurement parameters to *RST defaults. It then triggers a single reading, and returns the result. This command performs an `:ABORt`, `:CONFigure:<function>`, and a `:READ?`. If specified, range and resolution are also set. If the `<clist>` parameter is included, the specified channel closes before making the measurement.

If a function is not specified, the command executes as if the present function is specified. For example, assume the Ω2 function is presently selected. When MEAS? is sent, the instrument resets to the *RST defaults for the Ω2 function, and then makes a measurement.

When ABORt is executed, the instrument goes into the idle state if continuous initiation is disabled. If continuous initiation is enabled, operation restarts at the beginning of the trigger model.

This query is much slower than a :READ? or :FETCh? query because it has to reconfigure the instrument each time it is sent. It resets the NPLC, autoranging, and averaging to default settings.

This is an ideal command for taking one-shot measurements if the default settings for a measurement are appropriate and speed is not a requirement.

The range sets the range of the function. For example, if the range parameter is set to 10, the 10 V range is selected.

The resolution and the range set the number of display digits. Some examples are shown in the following table.

Resolution setting	Example	Equivalent digits
0.1	100.0	3½
0.01	10.00	3½
0.001	1.000	3½
0.0001	1.0000	4½
0.00001	1.00000	5½
0.000001	1.000000	6½

As shown in the table, with the 100 V range selected and resolution set to 0.1, a 100 V reading is displayed as 100.0 V (3½ digits).

The display defaults to 3½ digits when using parameter values that attempt to set the display to less than 3½ digits. For example, a 10 V reading with a resolution set to 0.1 for the 10 V range is displayed as 10.00 V, not 10.0 V.

For the TEMPerature and CONTinuity functions, you cannot use the `<rang>` and `<res>` parameters. The command is ignored and causes error −108, parameter not allowed.

A command that uses parameter values that attempt to set the display above 7½ digits is ignored and generates error −221, settings conflict error.

This command cannot be used when scanning. The command is ignored and causes error −221, settings conflict.

Example 1

Make a measurement using the presently selected function.

```
:MEAS?
```

Example 2

This example measures dc voltage channel 101 using the 10 V range with 3½ digit display resolution.

```
MEAS:VOLT? 10, 0.01, (@101)
```

Example 3

This example measures dc voltage on the 100 V range.

```
MEAS:VOLT? 100
```

Also see

:CONFigure? (on page 9-30)
:FETCh? (on page 9-32)
:READ? (on page 9-37)

:READ?

This command performs an ABORt, an INIT to trigger measurements, and then a FETCh? to retrieve the reading strings.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
:READ?
```

Details

This command resets the trigger model to the idle layer (equivalent to the :ABORt command), takes the trigger model out of idle (equivalent to the :INIT command), and returns a reading (equivalent to a FETCh? query). This command always returns a new reading, since aborting the trigger model invalidates any old readings and triggers a new one. This query waits for a new reading to become available before the instrument returns the results. For scans, it initiates on scan cycle and requests sample readings.

With continuous initiation disabled (INITiate:CONTinuous OFF), you can use the READ? command to trigger and return readings. The sample count determines the number of reading strings to be returned. With a sample count that is more than 1, the returned readings are automatically stored in the buffer.

This command does not work if the trigger source is set for BUS or EXTERNAL. This causes the error –214, Trigger deadlock. Under this condition, use a :FETCh? query or a :DATA:FRESh? query. If the trigger model is continuously initiating (:INIT:CONT ON), sending this query may cause error –213, Init ignored, but it still returns a new reading. It can be disabled by sending INITiate:CONTinuous OFF.

If the instrument receives a *RST command, it defaults to :INIT:CONT OFF, :TRIG:SOUR IMM, and :TRIG:COUNT 1. Sending a :READ? query under these conditions triggers a new reading. It also places the 2750 in one-shot measurement mode.

The following conditions must be met to use READ?:

- Continuous initiation must be disabled. It can be disabled by sending *RST or INIT:CONT OFF.
- If there are readings stored in the data store, the sample count (SAMP:COUN) must be set to 1.
- To use a sample count >1, the data store must be cleared (empty). It can be cleared by sending TRAC:CLE.

NOTE

When readings are stored in the buffer by the TRACe command or by front-panel data store operation, `INIT` and multi-sample `READ?` queries are locked out. With readings in the buffer that were stored in that manner, you cannot use the `INIT` or `READ?` command if the sample count is more than 1. If you use one of the commands, you receive the error message −225, out of memory. Refer to Reading buffers (on page 8-1) for more information.

The buffer of the 2750 is nonvolatile. Therefore, readings stored in the buffer are not lost when the instrument is turned off, or when `*RST` or `SYSTem:PRESet` is sent. When writing test programs that perform multi-sample measurements (`SAMPle:COUNt` >1), you may want to add the `TRACe:CLEar` command at the beginning to clear the buffer. However, be careful not to inadvertently clear stored readings that are needed.

Example

```
*RST
:INITiate:CONTinuous OFF;
:ABORt
:SENSe:FUNCtion 'VOLTage:DC'
:SENSe:VOLTage:DC:RANGe 10  // Use fixed range for fastest readings.
:SENSe:VOLTage:DC:NPLC 0.01 // Use lowest NPLC setting for fastest
                            // readings.
:DISPlay:ENABle OFF         // Turn off display to increase speed.
:SYSTem:AZERo:STATe OFF     // Disable autozero to increase speed, but may cause
                            // drift over time.
:SENSe:VOLTage:DC:AVERage:STATe OFF // Turn off filter for speed.
:TRIGger:COUNt 1
:READ?
```

This example makes a one-shot reading for dc voltage with no trigger at the fastest rate.

Also see

:INITiate:CONTinuous (on page 9-33)
[:SENSe[1]]:DATA:FRESh? (on page 9-99)
:FETCh? (on page 9-32)

CALCulate subsystem

The commands in this subsystem configure and control the math and statistic operations.

:CALCulate[1]:DATA:FRESh?

This command returns the last reading string.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:CALCulate[1]:DATA:FRESh?
```

Details

This command does not trigger a reading. The reading reflects the result of the calculation.

While the instrument is making measurements, you can use this command to return the last reading. `CALC:DATA:FRESh?` can only be used once to return the same reading string. Sending this command again to retrieve the same reading string generates error −230 (data corrupt or stale) or causes the GPIB to time out. To use `CALC:DATA:FRESh?`, a new reading must be triggered.

If math is disabled (`CALCulate:FORMat NONE` or `CALCulate:STATe OFF`), the raw reading is returned.

SCPI compliant.

Also see

:CALCulate[1]:DATA[:LATest]? (on page 9-39)
[:SENSe[1]]:DATA:FRESh? (on page 9-99)

:CALCulate[1]:DATA[:LATest]?

This command returns the last calculation result.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:CALCulate[1]:DATA[:LATest]?
```

Details

This command does not trigger a reading; it returns the last reading string. The reading reflects the result of the calculation.

You can use this command while the instrument is making measurements. If the instrument is not making measurements, `CALC:DATA?` keeps returning the same reading string.

If math is disabled (`CALCulate:FORMat NONE` or `CALCulate:STATe OFF`), the raw reading is retrieved by `CALC:DATA?`.

This query returns the last reading, regardless of issues that may have invalidated that reading, such as changing ranges or functions.

SCPI compliant.

Also see

[:SENSe[1]]:DATA:FRESh? (on page 9-99)

:CALCulate[1]:FORMat

This command specifies which math operation is performed on measurements when math operations are enabled.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	PERCent

Usage

```
:CALCulate[1]:FORMat <name>
:CALCulate[1]:FORMat <name>, <clist>
:CALCulate[1]:FORMat?
:CALCulate[1]:FORMat? <clist>
```

`<name>`	The math operation: ■ No math operation: `NONE` ■ y = mx+b: `MXB` ■ Percent: `PERCent` ■ Reciprocal: `RECiprocal`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This specifies which math operation is performed on measurements for the selected measurement function.

You can choose one of the following math operations:

- y = mx+b: Manipulate normal display readings by adjusting the m and b factors.
- Percent: Displays measurements as the percentage of deviation from a specified reference constant.
- Reciprocal: 1/X.

The query returns the selected format.

SCPI compliant.

Example

The following command sequence performs the mx+b calculation for channels 101 and 102 of the Model 7700. After `CALC:DATA?` is sent, the 2750 has to be addressed to talk to send the math result to the computer.

```
CALC:FORM MXB      ' Select mx+b calculation.
CALC:KMAT:MMF 2    ' Set 'm' factor to 2.
CALC:KMAT:MBF 100  ' Set 'b' factor to 100.
CALC:STAT ON       ' Enable math calculation.
ROUT:CLOS (@101)   ' Close channel 101.
CALC:DATA?         ' Read mx+b result for channel 101.
ROUT:CLOS (@102)   ' Close channel 102.
CALC:DATA?         ' Read mx+b result for channel 102.
```

Also see

:CALCulate[1]:KMATh:MBFactor (on page 9-41)
:CALCulate[1]:KMATh:MMFactor (on page 9-43)
:CALCulate[1]:KMATh:MUNits (on page 9-44)
:CALCulate[1]:KMATh:PERCent (on page 9-45)

:CALCulate[1]:KMATh:MBFactor

This command specifies the offset, b, for the y = mx + b operation.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
:CALCulate:KMATh:MBFactor <NRf>
:CALCulate:KMATh:MBFactor <NRf>, <clist>
:CALCulate:KMATh:MBFactor?
:CALCulate:KMATh:MBFactor? <clist>
:CALCulate:KMATh:MA0Factor <NRf>
:CALCulate:KMATh:MA0Factor <NRf>, <clist>
:CALCulate:KMATh:MA0Factor?
:CALCulate:KMATh:MA0Factor? <clist>
```

`<NRf>`	The offset factor; the valid range is −4294967295 to +4294967295
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This attribute specifies the offset (b) for an mx+b operation.

The mx+b math operation lets you manipulate normal display readings (x) mathematically based on the calculation:

$$y = mx + b$$

Where:

- *y* is the displayed result
- *m* is a user-defined constant for the scale factor
- *x* is the measurement reading (if you are using a relative offset, this is the measurement with relative offset applied)
- *b* is the user-defined constant for the offset factor

The `:MBFactor` and `:MA0Factor` commands perform the same operations.

Example

The following command sequence performs the mx+b calculation for channels 101 and 102 of the Model 7700. After `CALC:DATA?` is sent, the 2750 has to be addressed to talk to send the math result to the computer.

```
CALC:FORM MXB     ' Select mx+b calculation.
CALC:KMAT:MMF 2   ' Set 'm' factor to 2.
CALC:KMAT:MBF 100 ' Set 'b' factor to 100.
CALC:STAT ON      ' Enable math calculation.
ROUT:CLOS (@101)  ' Close channel 101.
CALC:DATA?        ' Read mx+b result for channel 101.
ROUT:CLOS (@102)  ' Close channel 102.
CALC:DATA?        ' Read mx+b result for channel 102.
```

Also see

:CALCulate:FORMat (on page 9-40)

:CALCulate[1]:KMATh:MMFactor

This command specifies the scale factor, m, for the y = mx+b math operation.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	1

Usage

```
:CALCulate:KMATh:MMFactor <NRf>
:CALCulate:KMATh:MMFactor <NRf>, <clist>
:CALCulate:KMATh:MMFactor?
:CALCulate:KMATh:MMFactor? <clist>
:CALCulate:KMATh:MA1Factor <NRf>
:CALCulate:KMATh:MA1Factor <NRf>, <clist>
:CALCulate:KMATh:MA1Factor?
:CALCulate:KMATh:MA1Factor? <clist>
```

`<NRf>`	The scale factor; the valid range is −4294967295 to +4294967295
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command sets the scale factor (m) for an mx+b operation for the selected measurement function.

The mx+b math operation lets you manipulate normal display readings (x) mathematically according to the following calculation:

$$y = mx + b$$

Where:

- *y* is the displayed result
- *m* is a user-defined constant for the scale factor
- *x* is the measurement reading (if you are using a relative offset, this is the measurement with relative offset applied)
- *b* is the user-defined constant for the offset factor

The :MMFactor and :MA1Factor commands perform the same operations.

Example

The following command sequence performs the mx+b calculation for channels 101 and 102 of the Model 7700. After `CALC:DATA?` is sent, the 2750 has to be addressed to talk to send the math result to the computer.

```
CALC:FORM MXB      ' Select mx+b calculation.
CALC:KMAT:MMF 2    ' Set 'm' factor to 2.
CALC:KMAT:MBF 100  ' Set 'b' factor to 100.
CALC:STAT ON       ' Enable math calculation.
ROUT:CLOS (@101)   ' Close channel 101.
CALC:DATA?         ' Read mx+b result for channel 101.
ROUT:CLOS (@102)   ' Close channel 102.
CALC:DATA?         ' Read mx+b result for channel 102.
```

Also see

:CALCulate:FORMat (on page 9-40)

:CALCulate[1]:KMATh:MUNits

This command specifies the units for the mx+b result.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	'X'

Usage

```
:CALCulate[1]:KMATh:MUNits "<char>"
:CALCulate[1]:KMATh:MUNits "<char>", <clist>
:CALCulate[1]:KMATh:MUNits?
:CALCulate[1]:KMATh:MUNits? <clist>
```

`<name>`	Specify units for mx+b; see **Details**
`<clist>`	Channel list parameter

Details

Specify the units for the mx+b result, up to 2 characters. You can specify:

- Alphabetic characters: A to Z
- Degree symbol (°): \
- Ohms symbol (Ω): [

Example

`CALCulate:KMATh:MUNits '['`	Use ohms symbol (Ω) as the units designator.
`CALCulate:KMATh:MUNits '\'`	Use degree symbol (°) as the units designator.

Also see

:CALCulate:FORMat (on page 9-40)

:CALCulate[1]:KMATh:PERCent

This command specifies the reference constant that is used when math operations are set to percent.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	1

Usage

```
:CALCulate[1]:KMATh:PERCent <NRf>
:CALCulate[1]:KMATh:PERCent <NRf>, <clist>
:CALCulate[1]:KMATh:PERCent?
:CALCulate[1]:KMATh:PERCent? <clist>
```

`<NRf>`	The target value for the percent calculation: `-4294967295` to `+4294967295`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command is coupled to the :CALCulate[1]:KMATh:PERCent:ACQuire command. When a reference value is set, the :PERCent? query command returns the programmed value. When reference is set using :ACQuire, the :PERCent? query command returns the acquired reference value.

Example

The following command sequence configures channels 101 through 110 of the Model 7700 to perform the percent calculation when they are scanned.

```
CALC:FORM PERC,(@101:110)        ' Select percent calculation.
CALC:KMAT:PERC 100,(@101:110)    ' Set reference to 100.
CALC:STAT ON,(@101:110)          ' Enable math calculation.
```

Also see

:CALCulate[1]:FORMat (on page 9-40)
:CALCulate[1]:KMATh:PERCent:ACQuire (on page 9-45)

:CALCulate[1]:KMATh:PERCent:ACQuire

This command sets the input signal to be the reference value when math operation is set to percent.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:CALCulate:KMATh:PERCent:ACQuire
```

Details

This command is only functional if a reading is available. If the instrument is overflowed (“OVERFLOW”) or a reading has not been triggered ("`-------`"), an execution error (−200) occurs when :ACQuire is sent.

Also see

:CALCulate[1]:FORMat (on page 9-40)
:CALCulate[1]:KMATh:PERCent (on page 9-45)

:CALCulate[1]:STATe

This command enables or disables the selected math operation.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	System reset: ON Preset: OFF

Usage

```
:CALCulate:STATe <b>
:CALCulate:STATe <b>, <clist>
:CALCulate:STATe?
:CALCulate:STATe <clist>
```

`<b>`	Disable math operations: `OFF` Enable math operations: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Example

The following command sequence configures channels 101 through 110 of the Model 7700 to perform the percent calculation when they are scanned.

```
CALC:FORM PERC,(@101:110)        ' Select percent calculation.
CALC:KMAT:PERC 100,(@101:110)    ' Set reference to 100.
CALC:STAT ON,(@101:110)          ' Enable math calculation.
```

Also see

:CALCulate[1]:FORMat (on page 9-40)

CALCulate2 subsystem

The commands in this path configure and control math calculations on buffer data. For additional information, see Buffer operations (on page 8-1).

:CALCulate2:DATA?

This command reads the result of the last statistic calculation.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:CALCulate2:DATA?
```

Details

The `:CALCulate2:DATA?` command does not initiate a calculate operation. If new data is stored in the buffer, you must send the `:CALCulate2:IMMediate` or `:CALCulate2:IMMediate?` command to recalculate the statistic from that new data.

If you calculate a statistic for an empty buffer, the number 9.910000E+37 is returned when it is read.

If you perform a calculation with no statistic selected (CALC2:FORM NONE) or CALC2 disabled (CALC2:STAT OFF), the result of the last statistic calculation is returned. If there was no calculation previously performed, the number 9.910000E+37 is returned.

SCPI compliant.

Also see

:CALCulate2:IMMediate (on page 9-49)

:CALCulate2:FORMat

This command selects the statistics that are applied to the data in the buffer.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	NONE

Usage

```
:CALCulate2:FORMat <name>
:CALCulate2:FORMat?
```

`<name>`	The buffer statistic: ▪ Lowest reading in buffer: `MINimum` ▪ Largest reading in buffer: `MAXimum` ▪ Mean value of readings in buffer: `MEAN` ▪ Standard deviation of readings in buffer: `SDEViation` ▪ Peak-to-peak: `PKPK` ▪ No calculation: `NONE`

Details

The peak-to-peak value is calculated as:

$$\text{PKPK} = MAX - MIN$$

Where: *MAX* is the largest reading in the buffer and *MIN* is the lowest reading in the buffer.

If the standard deviation calculation is being performed on a buffer that has more than 1000 readings, the `CALCULATING` message flashes to indicate that the 2750 is busy. While the instrument is busy with the calculation, remote programming commands will not execute.

SCPI compliant.

Example

This program fragment stores 20 readings into the buffer and then calculates the mean average of the buffer readings. After sending a query command, the 2750 must be addressed to talk.

```
' Store readings:
TRAC:CLE:AUTO ON     ' Enable buffer autoclear.
TRAC:POIN 20         ' Set buffer size to 20.
TRAC:FEED SENS       ' Select raw readings for storage.
TRAC:FEED:CONT NEXT  ' Start storage process.
TRAC:DATA?           ' Request all stored readings.

' Calculate the mean:
CALC2:FORM MEAN      ' Select mean calculation.
CALC2:STAT ON        ' Enable mean calculation.
CALC2:IMM?           ' Perform calculation and request result.
```

Also see

Buffer statistics (on page 8-6)

:CALCulate2:IMMediate

This command recalculates raw input data in the buffer.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	Not applicable

Usage

```
:CALCulate2:IMMediate
:CALCulate2:IMMediate?
```

Details

The query performs a calculation and reads the result.

After the selected statistic is enabled, `IMMediate` or `IMMediate?` must be sent to calculate the statistic from the data in the buffer.

When the calculation is selected (using the CALC2:FORMat command), and CALC2 is enabled (CALC2:STATe ON), use the CALC2:IMM? or CALC2:IMM command to perform the calculation:

- When CALC2:IMM? is used, the statistic is calculated and result is returned.
- When CALC2:IMM is used to calculate the statistic, the CALC2:DATA? command is used to return the result.

If you perform a calculation with no statistic selected (CALC2:FORM NONE) or CALC2 disabled (CALC2:STAT OFF), the result of the last statistic calculation is returned.

If you calculate a statistic for an empty buffer or if no calculation was made, the number `9.910000E+37` is returned when it is read.

Use *OPC or *OPC? with CALC2:IMM and CALC2:IMM? when performing the standard deviation calculation on a large buffer.

SCPI compliant.

Also see

:CALCulate2:DATA? (on page 9-48)
:CALCulate2:FORMat (on page 9-48)
*OPC (on page 10-6)

:CALCulate2:STATe

This command enables or disables the statistic calculation.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Reset: ON Preset: OFF

Usage

```
:CALCulate2:STATe <b>
:CALCulate2:STATe?
```

<b>	Disable math operations: `OFF` or `0` Enable math operations: `ON` or `1`

Details

When enabled, the selected `CALC 2` format is calculated when the `:CALCulate2:IMMediate` or `:CALCulate2:IMMediate?` command is executed.

SCPI compliant.

Also see

:CALCulate2:FORMat (on page 9-48)
:CALCulate2:IMMediate (on page 9-49)

CALCulate3 subsystem

The commands in the CALCulate3 subsystem to configure and control limit testing. Refer to Limits (on page 6-1) for more information on limit testing.

:CALCulate3:LIMit1:CLEar:AUTO

This command enables or disables autoclear for limit 1 when instrument operation enters the idle state.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
:CALCulate3:LIMit1:CLEar:AUTO <b>
:CALCulate3:LIMit1:CLEar:AUTO?
```

<b>	Disable autoclear: `OFF` Enable autoclear: `ON`

Details

With autoclear enabled, the fail message (`0`) is cleared when the instrument returns to the idle state. If the test is programmed not to return to idle, you can manually clear the fail condition by sending the `CLEar[:IMMediate]` command. If autoclear is disabled, the fail condition must be cleared manually.

SCPI compliant.

Also see

:CALCulate3:LIMit1:CLEar[:IMMediate] (on page 9-51)

:CALCulate3:LIMit1:CLEar[:IMMediate]

This command clears the high and low events for LIMIT 1.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:CALCulate3:LIMit1:CLEar[:IMMediate]
```

Details

Use this command to clear the test results when the limit autoclear option is turned off. Both the high and low test results are cleared.

SCPI compliant.

Also see

None

:CALCulate3:LIMit1:FAIL?

This command queries the results of a limit test.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:CALCulate3:LIMit1:FAIL?
```

Details

Query test result:

- Passing (reading within the high and low limits): 0
- Failing (reading has reached or exceeded the high or low limit): 1

The response message only tells you if a limit test passed or failed. It does not tell you whether the high or low limit failed. To determine which limit failed, you must read the measurement event register. Refer to Status model (on page 11-1) for more information.

You can include the limits result in the read commands (`SENS:DATA?`, `FETCh?`, `READ?`, `MEAS?`, `CALC2:DATA?`, `TRACe:DATA?`, and `CALC1:DATA?`) using `FORMat:ELEMents`.

SCPI compliant.

Also see

:CALCulate3:LIMit1:CLEar (on page 9-51)
:FORMat:ELEMents (on page 9-64)

:CALCulate3:LIMit1:LOWer[:DATA]

This command specifies the lower limit for a limit test.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	−1

Usage

```
:CALCulate3:LIMit1:LOWer[:DATA] <n>
:CALCulate3:LIMit1:LOWer[:DATA] <n>, <clist>
:CALCulate3:LIMit1:LOWer[:DATA]?
:CALCulate3:LIMit1:LOWer[:DATA]? <clist>
```

`<value>`	The value of the lower limit: `−4294967295` to `+4294967295`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command sets the low limit for the selected measurement function. The actual limit depends on the measurement function that is presently selected. For example, a limit value of 0.000001 is 1 µA for the current function and 1 µC for the charge function.

A limit value is not range sensitive. A limit of 2 for volts is 2 V on all measurement ranges.

SCPI compliant.

Also see

:CALCulate3:LIMit1:STATe (on page 9-53)
:CALCulate3:LIMit1:UPPer[:DATA] (on page 9-54)

:CALCulate3:LIMit1:STATe

This command enables or disables the limit 1 test.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
:CALCulate3:LIMit1:STATe <b>
:CALCulate3:LIMit1:STATe <b>, <clist>
:CALCulate3:LIMit1:STATe?
:CALCulate3:LIMit1:STATe? <clist>
```

`<b>`	Disable the limit 1 test: `OFF` Enable the limit 1 test: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When controlled using remote commands, Limit 1 and Limit 2 can be controlled separately. The front-panel limit indicators are affected as follows:

- Limit 1 enabled: The front-panel HIGH/IN/LOW indicators work the same as they do for front-panel operation.
- Limit 1 disabled and Limit 2 enabled: The status indicators pertain to Limit 2. When the reading is within Limit 2, the message `I2` is displayed. When the reading reaches or exceeds the high or low limit, the HIGH or LOW annunciator turns on and the number 2 is displayed.

NOTE

When limits are disabled from the front panel, both Limit 1 and Limit 2 disable for remote operation.

SCPI compliant.

Also see

:CALCulate3:LIMit1:LOWer[:DATA] (on page 9-52)
:CALCulate3:LIMit1:UPPer[:DATA] (on page 9-54)

:CALCulate3:LIMit1:UPPer[:DATA]

This command sets high limit 1 for monitor.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	1.0

Usage

```
:CALCulate3:LIMit1:UPPer[:DATA] <n>
:CALCulate3:LIMit1:UPPer[:DATA] <n>, <clist>
:CALCulate3:LIMit1:UPPer[:DATA]?
:CALCulate3:LIMit1:UPPer[:DATA]? <clist>
```

`<n>`	The value of the upper limit: −4294967295 to +4294967295
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command sets the high limit for the selected measurement function. The actual limit depends on the measurement function that is presently selected. For example, a limit value of 0.000001 is 1 µA for the current function and 1 µC for the charge function.

A limit value is not range sensitive. A limit of 2 for volts is 2 V on all measurement ranges.

SCPI compliant.

Also see

:CALCulate3:LIMit1:LOWer (on page 9-52)
:CALCulate3:LIMit1:STATe (on page 9-53)

:CALCulate3:LIMit2:CLEar:AUTO

This command enables or disables autoclear for limit 2 when instrument operation enters the idle state.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
:CALCulate3:LIMit2:CLEar:AUTO <b>
:CALCulate3:LIMit2:CLEar:AUTO?
```

`<b>`	Disable autoclear: `OFF` Enable autoclear: `ON`

Details

With autoclear enabled, the fail message (`0`) is cleared when the instrument returns to the idle state. If the test is programmed not to return to idle, you can manually clear the fail condition by sending the `CLEar[:IMMediate]` command. If autoclear is disabled, the fail condition must be cleared manually.

SCPI compliant.

Also see

:CALCulate3:LIMit2:CLEar (on page 9-55)

:CALCulate3:LIMit2:CLEar[:IMMediate]

This command clears the high and low events for LIMIT 2.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:CALCulate3:LIMit2:CLEar[:IMMediate]
```

Details

Use this command to clear the test results when the limit autoclear option is turned off. Both the high and low test results are cleared.

SCPI compliant.

Also see

:CALCulate3:LIMit2:CLEar:AUTO (on page 9-54)

:CALCulate3:LIMit2:FAIL?

This command queries the results of a limit test.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:CALCulate3:LIMit2:FAIL?
```

Details

Query test result:

- Passing (reading within the high and low limits): 0
- Failing (reading has reached or exceeded the high or low limit): 1

The response message only tells you if a limit test passed or failed. It does not tell you whether the high or low limit failed. To determine which limit failed, you must read the measurement event register. Refer to Status model (on page 11-1) for more information.

You can include the limits result in the read commands (`SENS:DATA?`, `FETCh?`, `READ?`, `MEAS?`, `CALC2:DATA?`, `TRACe:DATA?`, and `CALC1:DATA?`) using `FORMat:ELEMents`.

SCPI compliant.

Also see

:CALCulate3:LIMit2:CLEar (on page 9-55)
:FORMat:ELEMents (on page 9-64)

:CALCulate3:LIMit2:LOWer[:DATA]

This command specifies the lower limit for a limit test.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	−2

Usage

```
:CALCulate3:LIMit2:LOWer[:DATA] <n>
:CALCulate3:LIMit2:LOWer[:DATA] <n>, <clist>
:CALCulate3:LIMit2:LOWer[:DATA]?
:CALCulate3:LIMit2:LOWer[:DATA]? <clist>
```

`<n>`	The value of the lower limit: −4294967295 to +4294967295
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command sets the low limit for the selected measurement function. The actual limit depends on the measurement function that is presently selected. For example, a limit value of 0.000001 is 1 µA for the current function and 1 µC for the charge function.

A limit value is not range sensitive. A limit of 2 for volts is 2 V on all measurement ranges.

SCPI compliant.

Also see

:CALCulate3:LIMit2:STATe (on page 9-57)
:CALCulate3:LIMit2:UPPer[:DATA] (on page 9-58)

:CALCulate3:LIMit2:STATe

This command enables or disables the limit 2 test.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
:CALCulate3:LIMit2:STATe <b>
:CALCulate3:LIMit2:STATe <b>, <clist>
:CALCulate3:LIMit2:STATe?
:CALCulate3:LIMit2:STATe? <clist>
```

`<b>`	Disable the limit 2 test: `OFF` Enable the limit 2 test: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When enabled, the test sequence for limits is performed every time the instrument makes a measurement. Testing is performed in the following sequence: Low Limit 1, High Limit 1, Low Limit 2, and High Limit 2. Only limit tests that are enabled are performed.

When a limit test is enabled, the digital output port cannot be controlled from the `:SOURce` subsystem.

A fail indication for LIMIT 2 is cleared when the respective limit test is disabled.

SCPI compliant.

Also see

:CALCulate3:LIMit2:LOWer (on page 9-56)
:CALCulate3:LIMit2:UPPer (on page 9-58)

:CALCulate3:LIMit2:UPPer[:DATA]

This command specifies the upper limit for a limit test.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	2.0

Usage

```
:CALCulate3:LIMit2:UPPer[:DATA] <n>
:CALCulate3:LIMit2:UPPer[:DATA] <n>, <clist>
:CALCulate3:LIMit2:UPPer[:DATA]?
:CALCulate3:LIMit2:UPPer[:DATA]? <clist>
```

`<value>`	The value of the upper limit: `-4294967295` to `+4294967295`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command sets the high limit for the selected measurement function. The actual limit depends on the measurement function that is presently selected. For example, a limit value of 0.000001 is 1 µA for the current function and 1 µC for the charge function.

A limit value is not range sensitive. A limit of 2 for volts is 2 V on all measurement ranges.

SCPI compliant.

Also see

:CALCulate3:LIMit2:LOWer[:DATA] (on page 9-56)
:CALCulate3:LIMit2:STATe (on page 9-57)

:CALCulate3:MLIMit:LATChed

This command enables or disables the master limit latch.

Type	Affected by	Where saved	Default value
Command only	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
CALCulate3:MLIMit:LATChed <b>
```

`<b>`	Disable: `OFF` Enable: `ON`

Details

The master limit line is pulled high or low when one or more of the other four limits are reached or exceeded. The master limit line can be programmed to release when a reading is inside all four limits, or the master limit can be latched when a failure occurs. When latched, the master limit line will not release until operation within the trigger model returns to and passes the control source. Refer to Triggering (on page 7-1) for details on triggering.

When scanning, the latched master limit line will not release until the scan is finished and another scan is started. For example, if after testing a resistor network the master limit line did set, then the network has passed all tests.

Also see

None

:CALCulate3:OUTPut:LSENse

This command sets the logic sense of all limit lines.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	AHIGh

Usage

```
:CALCulate3:OUTPut:LSENse <name>
:CALCulate3:OUTPut:LSENse?
```

`<name>`	Set logic sense: `AHIGh` or `ALOW`

Also see

None

:CALCulate3:OUTPut:PULSe:TIMe

This command sets the output pulse time.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.002

Usage

```
:CALCulate3:OUTPut:PULSe:TIMe <NRf>
:CALCulate3:OUTPut:PULSe:TIMe?
```

`<NRf>`	Output pulse time in seconds: 0.001 to 99999.999

Also see

:CALCulate3:OUTPut:PULSe[:STATe] (on page 9-60)

:CALCulate3:OUTPut:PULSe[:STATe]

This command enables or disables limit output pulsing.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
:CALCulate3:OUTPut:PULSe[:STATe] <b>
:CALCulate3:OUTPut:PULSe[:STATe]?
```

<b>	Disable: OFF Enable: ON

Details

When enabled, an output line will pulse high or low (depending on the logic sense setting) for each reading that reaches or exceeds the limit.

Also see

None

:CALCulate3:OUTPut[:STATe]

This command enables or disables digital outputs.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
:CALCulate3:OUTPut[:STATe] <b>
:CALCulate3:OUTPut[:STATe]?
```

<b>	Disable: OFF Enable: ON

Example

The following command sequence configures the 2750 to perform Limit 1 test on a DCV reading. If the 100 mV limit is reached, digital output 2 is pulled low. If the −100 mV limit is reached, digital output 1 is pulled low.

```
*RST                 ' One-shot measurement mode (DCV).
CALC3:LIM1:UPP 0.1   ' Set HI1 limit to 100 mV.
CALC3:LIM1:LOW -0.1  ' Set LO1 limit to -100 mV.
CALC3:LIM1:STAT ON   ' Enable Limit 1.
CALC3:OUTP:LSEN ALOW ' Set logic sense to active low.
CALC3:OUTP ON        ' Enable digital outputs.
READ?                ' Trigger and request reading.
CALC3:LIM1:FAIL?     ' Request result of limit 1 test.
```

Also see

None

DISPlay subsystem

This subsystem contains commands that control the front-panel display.

:DISPlay[:WINDow[1]:TEXT:DATA

This command defines the text that is displayed on the front-panel display.

Type	Affected by	Where saved	Default value
Command and query	Power cycle	Not applicable	Not applicable

Usage

```
:DISPlay[:WINDow[1]:TEXT:DATA "<string>"
:DISPlay[:WINDow[1]:TEXT:DATA?
```

`<string>`	String that contains the message; up to 12 characters

Details

This command defines a text message for the display.

A space counts as a character. If there are too many characters, an error is generated.

The text must be enclosed in either single quotes (`' '`), or double quotes (`" "`).

NOTE

A user-defined text message remains displayed only when the instrument is in remote operation. Taking the instrument out of remote operation cancels the message.

Example

`DISP:TEXT:DATA 'TESTING'` `DISP:TEXT:STAT ON`	Define the text message `TESTING`. Enable text message mode to display the message.

Also see

:DISPlay:TEXT:STATe (on page 9-62)

:DISPlay[:WINDow[1]]:TEXT:STATe

This command enables or disables message mode.

Type	Affected by	Where saved	Default value
Command and query	Power cycle	Not applicable	OFF

Usage

```
:DISPlay[:WINDow[1]]:TEXT:STATe <b>
:DISPlay[:WINDow[1]]:TEXT:STATe?
```

<b>	Disable text message mode: OFF or 0 Enable text message mode: ON or 1

Details

When enabled, a defined text message is displayed. When disabled, the message is removed from the display. Messages are disabled when power is cycled.

A user-defined text message remains displayed only when the instrument is in remote. Taking the instrument out of remote (by pressing the LOCAL key or sending GTL) cancels the message and disables the text message mode.

Example

DISP:TEXT:DATA 'TESTING' DISP:TEXT:STAT ON	Define the text message TESTING. Enable text message mode to display the message.

Also see

DISPlay:TEXT:DATA (on page 9-61)

:DISPlay:ENABle

This command turns the front-panel display on or off.

Type	Affected by	Where saved	Default value
Command and query	Power cycle	Nonvolatile memory	ON

Usage

```
:DISPlay:ENABle <b>
:DISPlay:ENABle?
```

<b>	Disable the front-panel display: OFF or 0 Enable the front-panel display: ON or 1

Details

When the front-panel display is disabled, the instrument operates at a higher speed.

Example

`DISP:ENABle OFF`	Turn off the display.

Also see

None

FORMat subsystem

The commands for this subsystem select the data format that is used to transfer instrument readings over the remote interface.

:FORMat:BORDer

This command sets the byte order for the IEEE Std 754 binary formats.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	SWAPped

Usage

```
:FORMat:BORDer <name>
:FORMat:BORDer?
```

`<name>`	The binary byte order: ▪ Normal byte order: `NORMal` ▪ Reverse byte order for binary formats: `SWAPped`

Details

This attribute selects the byte order in which data is written.

The ASCII data format can only be sent in the normal byte order. If the ASCII format is selected, the `SWAPped` selection is ignored.

When you select `NORMal` byte order, the data format for each element is sent as follows:

`Byte 1 Byte 2 Byte 3 Byte 4`

(Single precision)

`Byte 1  Byte 2  ... Byte 8`

(Double precision)

When you select `SWAPped`, the data format for each element is sent as follows:

`Byte 4 Byte 3 Byte 2 Byte 1`

(Single precision)

`Byte 8  Byte 7 ... Byte 1`

(Double precision)

The `#0` header is not affected by this command. The header is always sent at the beginning of the data string for each measurement conversion.

This command affects readings that are transferred from the buffer. Commands that are not transferred from the buffer, such as `:SENSe:DATA?` and `:CALC:DATA?`, are always sent in ASCII.

SCPI compliant.

Also see

:FORMat[:DATA] (on page 9-66)

:FORMat:ELEMents

This command specifies the elements to include in a data string for each measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	READing, UNITs, RNUMber, TSTamp

Usage

```
:FORMat:ELEMents <item list>
:FORMat:ELEMents?
```

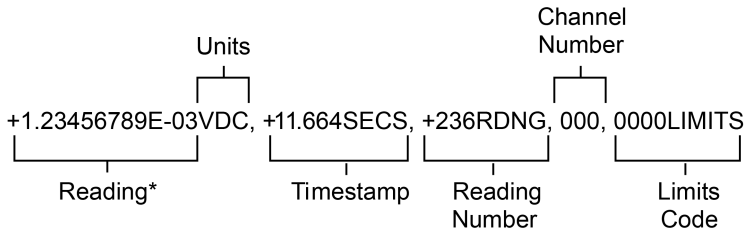
`<item list>`	Data elements to include: ■ Reading: `READing` ■ Reading number: `RNUMber` ■ The measurement function (such as VDC): `UNITs` ■ Channel number: `CHANnel` ■ Timestamp: `TSTamp` ■ Limits reading: `LIMits`

Details

You can specify any combination of the `<item list>` parameters in any order. Each item in the list must be separated by a comma. The specified elements are included in the data string in response to `FETCh?`, `READ?`, `MEASure?`, and `TRACe:DATA?`.

The elements for the ASCII format are shown in the following figure.

Figure 59: ASCII data format



Units:

VDC	=	DC Volts	HZ	=	Frequency
VAC	=	AC Volts	SECS	=	Period
ADC	=	DC Current	C	=	Temperature in C˚
AAC	=	AC Current	F	=	Temperature in F˚
OHM	=	2-Wire Resistance or Continuity	K	=	Temperature in K˚
OHM4W	=	4-Wire Resistance			

*An overflow reading is displayed as +9.9E37 with no limits.

An overflow reading is returned as `+9.9e37`. When a specified data element has invalid data associated with it, NAN (not a number) is the response. NAN is also returned as `+9.9E37`.

`RNUMber` returns the number of the reading. The instrument keeps track of the number of readings it has made. This reading counter starts at zero when the instrument is turned on or when `:SYSTem:RNUMber:RESet` is sent over the bus. The instrument also keeps track of the number of readings it has stored in the buffer. When buffer readings are sent over the bus (`:TRACe:DATA?`), each reading number is referenced to the first reading, which is #0, stored in the buffer.

The channel number indicates the switching module channel for the reading. Channel number 000 indicates that no channel was closed.

`TSTamp` includes the timestamp of the measurement. The timestamp is either real-time or relative, as defined by `:SYSTem:TSTamp:TYPE`. `TSTamp` is also available for buffer readings. `TSTamp` can be referenced to the first reading stored in the buffer (absolute format) that is timestamped at 0 seconds or can provide the time between each reading (delta format). The `:TRACe:TSTamp:FORMat` command selects the absolute or delta format.

NOTE

The real-time clock timestamp can only be returned for the ASCII data format. For the binary formats, timestamp is not sent with the real-time clock selected.

The relative timestamp operates as a timer that starts at zero seconds when the instrument is turned on, or when the relative timestamp is reset (`SYSTem:TSTamp:RELative:RESet`). After 99,999.99 seconds, the timer resets back to zero and starts over.

For buffer readings recalled from the front panel, the relative timestamp is referenced to the first reading stored in the buffer (absolute format) which is timestamped at 0 seconds and to the time between each stored reading (delta format). For remote programming, you can only return the absolute or delta timestamp. Use the `TRACe:TSTamp:FORMat` command to select the relative timestamp format for the buffer.

For the ASCII data format, limit test results are returned as a 4-bit binary number `abcd` where:

a = High limit 2

b = Low limit 2

c = High limit 1

d = Low limit 1

A `0` in the bit position indicates that the limit has passed. A `1` indicates that the limit has failed.

For the binary data formats, the limits information must be decoded from the returned value (0 to 15). Convert the value to its binary equivalent for `abcd` where `d` is the LSD and `a` is the MSD. For example, the value 10 converted to its binary equivalent is 1010. That means High Limit 2 and High Limit 1 have failed.

Example

`FORMat:ELEMents READing, UNITs, RNUMber`
This example includes the units and reading number with the reading.

Also see

:FORMat[:DATA] (on page 9-66)
:SYSTem:TSTamp:TYPE (on page 9-198)
:SYSTem:RNUMber:RESet (on page 9-196)
:SYSTem:TSTamp:RELative:RESet (on page 9-198)
:TRACe:TSTamp:FORMat (on page 9-207)

:FORMat[:DATA]

This command specifies the data format that is used when transferring readings over the remote interface.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle	Save settings	ASCii

Usage

```
:FORMat[:DATA] <type>
:FORMat[:DATA] REAL, <length>
:FORMat[:DATA]?
```

`<type>`	The data format: ■ ASCII: `ASCii` ■ IEEE Std 754 single-precision: `REAL` ■ IEEE Std 754 single-precision: `SREal` ■ IEEE Std 754 double-precision: `DREal`
`<length>`	32 or 64 (default 32)

Details

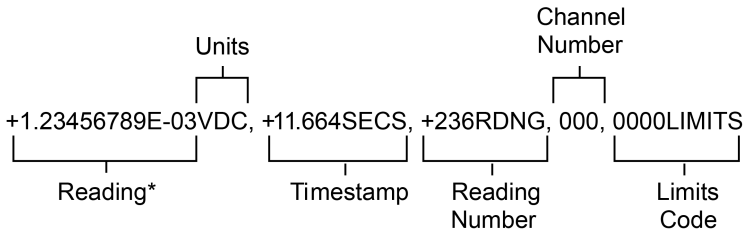
NOTE

The RS-232 interface only supports the ASCII data format. If the RS-232 interface is selected, the data format defaults to ASCII and cannot be changed.

The response to READ?, FETCh?, MEASure?, TRACe:DATA?, CALC1:DATA?, or CALC2:DATA? over the GPIB can be returned in either the ASCII or binary format. All other queries are returned in ASCII, regardless of the selected format. Regardless of which data format for output strings is selected, the instrument will only respond to input commands using the ASCII format.

The ASCII data format is in a direct readable form for the operator. Most BASIC languages convert ASCII mantissa and exponent to other formats. However, some speed is compromised to accommodate the conversion. The following figure shows the ASCII format that includes all the data elements.

Figure 60: ASCII data format



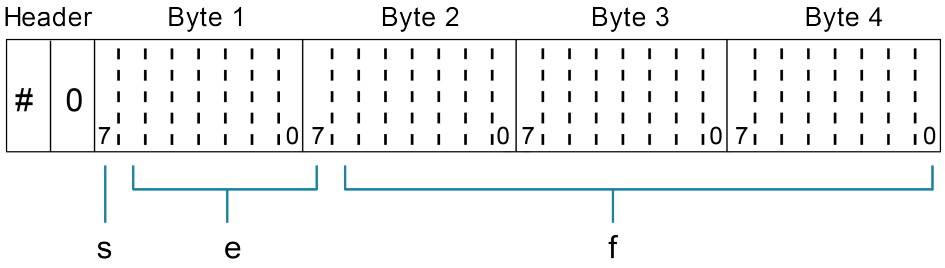
Units:

VDC	=	DC Volts	HZ	=	Frequency
VAC	=	AC Volts	SECS	=	Period
ADC	=	DC Current	C	=	Temperature in C°
AAC	=	AC Current	F	=	Temperature in F°
OHM	=	2-Wire Resistance or Continuity	K	=	Temperature in K°
OHM4W	=	4-Wire Resistance			

*An overflow reading is displayed as +9.9E37 with no limits.

`REAL,32` or `SREal` selects the binary IEEE Std 754 single-precision data format and is shown in the following figure. The figure shows the normal byte order format for each data element. For example, if three valid elements are specified, the data string for each reading conversion is made up of three 32-bit data blocks. The data string for each reading conversion is preceded by a 2-byte header that is the binary equivalent of an ASCII # sign and 0.

Figure 61: IEEE Std 754 single-precision data format (32 data bits)



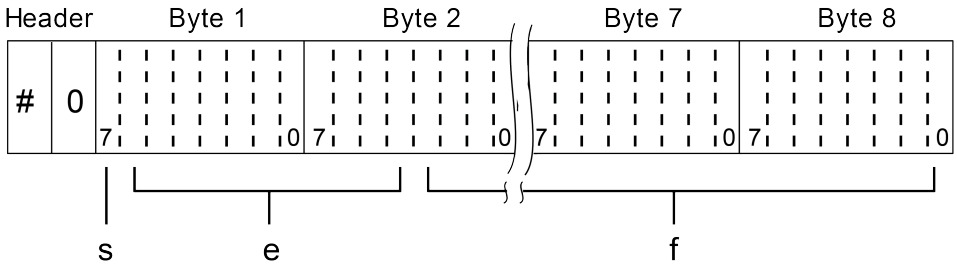
This figure shows:

- s = sign bit (0 = positive, 1 = negative)
- e = exponent bits (8)
- f = fraction bits (23)

The normal byte order is shown. For the swapped byte order, the bytes are sent in the order: Header, Byte 4, Byte 3, Byte 2, Byte 1. The Header is only sent once for each measurement conversion. The figure does not show the byte for the terminator that is attached to the end of each data string. You can use FORMat:BORDer to send the byte order of the data string in reverse order.

`REAL,64` or `DREal` selects the binary IEEE Std 754 double-precision data format and is shown in the following figure (normal byte order shown). This format is similar to the single-precision format except that it is 64 bits long.

Figure 62: IEEE Std 754 double precision data format (64 data bits)



This figure shows:

- s = sign bit (0 = positive, 1 = negative)
- e = exponent bits (11)
- f = fraction bits (52)

Bytes 3, 4, 5, and 6 are not shown. Normal byte order shown. For swapped byte order, the bytes are sent in the order Header, Byte 8, Byte 7 … Byte 1. The Header is only sent once for each measurement conversion.

Also see

:FORMat:BORDer (on page 9-63)
:FORMat:ELEMents (on page 9-64)

ROUTe subsystem

The commands in this subsystem configure and control switching and are summarized in ROUTe command summary (on page 9-10).

:ROUTe:CLOSe

This command closes the specified channels on a card installed in the 2750.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	Not applicable

Usage

```
:ROUTe:CLOSe <clist>
:ROUTe:CLOSe? <clist>
```

`<clist>`	The channels to be closed; refer to Channel assignments (on page 3-4)

Details

The query returns a list of closed measurement channels, including the paired channel for 4-wire measurements. It does not return nonmeasurement channels, such as backplane isolation channels and the pole-mode channel.

This command functions the same as the front-panel CLOSE key (SINGLE menu option) to select the system channel. Only one measurement channel can be specified in the `<clist>`. If you try to close an invalid channel (such as a non-measurement channel) with this command, the error −222, Parameter data out of range, is generated.

Example

`:ROUTe:CLOSe? (@101, 104, 107, 102)`
This example closes channels 1, 4, 7, and 2 on the module in slot 1.

Also see

None

:ROUTe:CLOSe:ACONfigure

This command enables or disables automatic channel configuration.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	See Details

Usage

```
:ROUTe:CLOSe:ACONfigure <b>
:ROUTe:CLOSe:ACONfigure?
```

`<b>`	Enable autoconfigure: ON Disable autoconfigure: OFF

Details

Not affected by `*RST` and `SYSTem:PRESet`. Front-panel factory default is OFF.

Also see

Autochannel configuration (on page 3-44)
Relative offset (REL) (on page 4-12)

:ROUTe:CLOSe:COUNt:INTerval

This command sets the count update interval.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	See **Details**

Usage

```
:ROUTe:CLOSe:COUNt:INTerval <NRf>
:ROUTe:CLOSe:COUNt:INTerval?
```

`<NRf>`	Count update interval in minutes: 10 to 1440

Details

Relay closure counts are updated in temporary RAM every time a channel is closed, regardless of whether it was closed by a SCPI command, front-panel control, or during a scan. These counts are permanently written to the EEPROM on the card only at a user-set time interval or whenever the counts are queried.

The lower the interval, the less chance there is of losing relay counts due to power failures. However, writing to the EEPROM more often may reduce scanning throughput. The higher the interval, the less scanning throughput is reduced, but more relay counts may be lost in the event of a power failure.

NOTE

If the 2750 is turned off before the updated count is written to EEPROM, the relay counts are lost. It is good practice to add the `ROUT:CLOS:COUN? <clist>` command at the end of a program to manually update the count.

The relay count interval set to 15 minutes at the factory. SYSTem:PRESet and *RST have no effect on the set interval.

Example

`ROUT:CLOS:COUN:INT 30`
This example sets the interval to 30 minutes.

Also see

Relay closure count (on page 7-17)

:ROUTe:CLOSe:COUNT?

This command queries the closure count for the specified channels.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:ROUTe:CLOSe:COUNT? <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The 2750 keeps an internal count of the number of times each module relay has been closed. The total number of relay closures are stored in EEPROM on the card. This count helps you determine if and when any relays require replacement (see module contact life specifications).

Relay closures are counted only when a relay cycles from open to closed state. If you send multiple close commands to the same channel without sending an open command, only the first closure is counted.

Relay closure count can only be read using remote operation.

Example 1

```
:ROUTe:CLOSe:COUNt? (@101, 104)
```

This example returns the closure count of channels 1 and 4 of a module in slot 1.

Example 2

```
:ROUTe:CLOSe:COUNt? (@101:110)
```

This example returns the closure count of channels 1 through 10 of a module in slot 1.

Also see

Relay closure count (on page 7-17)

:ROUTe:CLOSe:STATe?

This command queries the closed channels in the list.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:ROUTe:CLOSe:STATe? <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The query returns `0` (open) or `1` (closed) for every measurement channel specified in `<clist>`.

The state of non-measurement channels cannot be checked with this command.

Example

```
:ROUT:CLOS:STAT? (@101, 104, 107, 102)
```

If 107 is closed and the other channels are open, the response message is:
`0, 0, 1, 0`

Also see

None

:ROUTe:MONitor

This command specifies one channel to be monitored.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	Not applicable

Usage

```
:ROUTe:MONitor <clist>
:ROUTe:MONitor?
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The channel that you specify as the monitor must be a channel that is in the scan list. If it is not, the first channel in the scan list will automatically become the monitor channel.

If the `<clist>` has more than one channel, error −223 (too much data) occurs and the command is not executed.

Also see

Relative offset (REL) (on page 4-12)

:ROUTe:MONitor:DATA?

This command returns the most recent monitor reading.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:ROUTe:MONitor:DATA?
```

Also see

None

:ROUTe:MONitor:POINts

This command specifies the number of channels for a monitor scan.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Depends on which switching module is installed

Usage

```
:ROUTe:MONitor:POINts <NRf>
:ROUTe:MONitor:POINts?
```

`<NRf>`	The number of channels to scan: 2 to 110000

Details

Use this command to specify the number of channels to scan each time the monitor scan is triggered to start. For example, assume the monitor scan list has 10 channels. To scan that list once, send ROUT:MON:POIN 10. To scan that list twice, use parameter value 20. For three scans, send parameter value 30, and so on.

Also see

Relative offset (REL) (on page 4-12)

:ROUTe:MONitor:STATe

This command enables or disables channel monitoring.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
:ROUTe:MONitor:STATe <b>
:ROUTe:MONitor?
```

`<b>`	Enable channel monitoring: ON Disable channel monitoring: OFF

Details

Also see

Relative offset (REL) (on page 4-12)

:ROUTe:MULTiple:CLOSe

This command closes the channels specified in the list.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	

Usage

```
:ROUTe:MULTiple:CLOSe <clist>
:ROUTe:MULTiple:CLOSe?
```

`<clist>`	Channel assignments (on page 3-4)

Details

This command closes channels specified in channel assignments. Unlisted channels are not affected and channel pairing is disabled. This command functions like the front-panel CLOSE key (MULTI menu option).

The query returns list of all closed channels, including nonmeasurement channels and paired channels for 4-wire functions.

Channels closed by `ROUT:MULT:CLOS` are not displayed.

If the `<clist>` is large when using RS-232 operation and in some cases, GPIB operation, use `*OPC` or `*OPC?` with `:ROUT:MULT:CLOS`.

NOTE

The `ROUT:MULT:CLOS` command cannot be used to measure thermocouple temperature using the internal or external reference junction. The simulated reference junction is used instead. See Temperature measurements for details.

Also see

*OPC (on page 10-6)

:ROUTe:MULTiple:CLOSe:STATe?

This command returns the open or closed state for every channel specified in the list.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:ROUTe:MULTiple:CLOSe:STATe? <clist>
```

`<clist>`	Channel assignments (on page 3-4)

Details

This query returns a `0` (open) or `1` (closed) for every channel specified in the `<clist>`. It is valid for both measurement and nonmeasurement channels.

Example

```
:ROUTe:MULTiple:CLOSe:STATe? (@101, 104, 125)
```

In this example, assume channel 125 is closed. The response message returns `0, 0, 1` to indicate that channels 101 and 104 are open and channel 125 is closed

Also see

None

:ROUTe:MULTiple:OPEN

This command opens the channels specified in the list.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:ROUTe:MULTiple:OPEN <clist>
```

`<clist>`	The channels to open

Details

Opens the channels specified in `<clist>`. Unlisted channels are not affected.

Because only one channel can be closed, `<list>` for this command can only consist of one channel. You can also use the `ALL` parameter, which opens the closed channel.

The query command determines the state (opened or not opened) of each channel specified by the list parameter. See `:ROUTe:CLOSe` for examples of how to express a channel list. A return of `1` indicates that the channel is open. `0` indicates that the channel is not open.

If the `<clist>` is large when using RS-232 operation and in some cases, GPIB operation, use `*OPC` or `*OPC?` with `:ROUT:MULT:OPEN`.

SCPI compliant.

Also see

*OPC (on page 10-6)
:ROUTe:CLOSe (on page 9-69)

:ROUTe:OPEN:ALL

This command opens all channels.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:ROUTe:OPEN:ALL
```

Details

This command functions the same as the front-panel OPEN key (ALL menu option). It opens all channels, including non-measurement channels.

Example

```
ROUT:OPEN:ALL       ' Open all channels.
ROUT:CLOS (@101)    ' Close channels 101, 111, 123, 124, and 125.
```

This example assumes a 7700 installed in slot 1 with the Ω4 function of the 2750 selected. This command sequence connects channel 101 and its paired channel (111) to DMM Input and Sense.

Also see

:ROUTe:CLOSe (on page 9-69)

:ROUTe:SCAN:LSELect

This command enables or disables scan.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	NONE

Usage

```
:ROUTe:SCAN:LSELect <name>
:ROUTe:SCAN:LSELect?
```

`<name>`	The scan operation: ■ Disable all scan operations: `NONE` ■ Enable an internal scan: `INTernal`

Details

When `INTernal` is selected, the 2750 scans the channels of the internal switching card according to how the scan is configured.

The `NONE` selection disables all operations associated with a scan.

Also see

:ROUTe:SCAN[:INTernal] (on page 9-78)

:ROUTe:SCAN:NVOLatile

This command enables or disables nonvolatile memory for scanning (autoscan).

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	See **Details**

Usage

```
:ROUTe:SCAN:NVOLatile <b>
:ROUTe:SCAN:NVOLatile?
```

`<b>`	Disable autoscan: OFF Enable autoscan: ON

Details

Not affected by `*RST` and `SYSTem:PRESet`. Front-panel factory default is OFF.

Also see

:INITiate:CONTinuous (on page 9-33)
:INITiate[:IMMediate] (on page 9-34)

:ROUTe:SCAN:TSOurce

This command selects and queries the trigger source that starts the scan.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	IMM

Usage

```
:ROUTe:SCAN:TSOurce <list>
:ROUTe:SCAN:TSOurce?
```

`<list>`	Immediate control source: `IMMediate`
	`HLIMit1`
	`HLIMit2`
	`LLIMit1`
	`LLIMit2`

Details

The scan can start immediately when it is enabled and triggered or it can be started by a reached reading limit detected by the monitor channel. For immediate, the IMMediate command must be the only parameter in the list. To use reading limits, each limit must be separated by a comma (`,`).

Example

```
ROUT:SCAN:TSO IMM           ' Start scan when it is enabled and triggered.
ROUT:SCAN:TSO HLIM1,LLIM1   ' Enable high limits 1 and low limits 1.
```

Any reached limit will start the scan.

Also see

:INITiate:CONTinuous (on page 9-33)
:INITiate[:IMMediate] (on page 9-34)

:ROUTe:SCAN[:INTernal]

This command specifies the list of channels to be scanned.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	Not applicable

Usage

```
:ROUTe:SCAN[:INTernal] <clist>
:ROUTe:SCAN[:INTernal]?
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

Channels are scanned in the order that they are listed. Channel do not need to be sequential.

Non-sequential scanning is only intended to be performed using remote programming. Unexpected results may occur if a non-sequential scan is run from the front panel.

There must be at least two channels in the scan list. Creating a scan list that has only one channel generates error −221 (settings conflict).

Example 1

In this example, the scan runs, starting with the lowest numbered channel (101) and then sequencing up (forward) to the highest numbered channel (206).

```
ROUT:SCAN (@101:110,201,204,206)
```

Example 2

In this example, after channels 101 to 105 are scanned, the instrument scans channel 103, then proceeds to scan channels 106 through 110.

```
ROUT:SCAN (@101:105,103,106:110)
```

Example 3

In this example, the scan starts with channel 110, then proceeds backward to channel 101.

```
ROUT:SCAN (@110:101)
```

Also see

None

SENSe[1] subsystem

The SENSe[1] subsystem commands configure and control the measurement functions of the instrument.

[:SENSe[1]]:CAVerage:DELay

This command sets the delay when channel average is enabled.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.5 s

Usage

```
[:SENSe[1]]:CAVerage:DELay <NRf>
[:SENSe[1]]:CAVerage:DELay <NRf>, <clist>
[:SENSe[1]]:CAVerage:DELay?
[:SENSe[1]]:CAVerage:DELay? <clist>
```

`<NRf>`	Delay in seconds: 0 to 99999.999
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

`:RATio:DELay` and `:CAVerage:DELay` set the delay between two channel measurements for the enabled calculation. This delay is applied after the trigger delay in the trigger model. This delay cannot be set from the front panel. The 0.5 s default delay keeps the relays from cycling too fast. Setting a shorter delay may shorten the life of the relays.

It does not matter which of the two commands you use to set the delay. The set delay affects both the ratio and channel average.

Example

```
FUNC 'VOLT',(@103)   ' Set 103 for DCV.
RAT ON,(@103)        ' Set 103 for ratio on.
FUNC 'TEMP',(@105)   ' Set 105 for TEMP.
CAV ON,(@105)        ' Set 105 for channel average on.
```

This command sequence configures channels 103 and 105 for the ratio and channel average calculations. When channel 103 is scanned, the ratio calculation is based on DCV measurements of channels 103 and 113. When channel 105 is scanned, the channel average calculation is based on the TEMP measurements of channels 105 and 115.

Also see

Ratio and channel average (on page 4-18)
Triggering (on page 7-1)
[:SENSe[1]]:CAVerage[:STATe] (on page 9-80)
[:SENSe[1]]:RATio[:STATe] (on page 9-122)

[:SENSe[1]]:CAVerage[:STATe]

This command enables or disables channel average.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:CAVerage[:STATe] <b>
[:SENSe[1]]:CAVerage[:STATe] <b>, <clist>
[:SENSe[1]]:CAVerage[:STATe]?
[:SENSe[1]]:CAVerage[:STATe]? <clist>
```

`<b>`	Enable: ON Disable: OFF
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Example

```
FUNC 'VOLT',(@103)   ' Set 103 for DCV.
RAT ON,(@103)        ' Set 103 for ratio on.
FUNC 'TEMP',(@105)   ' Set 105 for TEMP.
CAV ON,(@105)        ' Set 105 for channel average on.
```

This command sequence configures channels 103 and 105 for the ratio and channel average calculations. When channel 103 is scanned, the ratio calculation is based on DCV measurements of channels 103 and 113. When channel 105 is scanned, the channel average calculation is based on the TEMP measurements of channels 105 and 115.

Also see

Ratio and channel average (on page 4-18)
:SENSe[1]:CAVerage:DELay (on page 9-79)

[:SENSe[1]]:CONTinuity:THReshold

This command selects the threshold resistance for continuity testing.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:CONTinuity:THReshold <NRf>
[:SENSe[1]]:CONTinuity:THReshold <n>, <clist>
[:SENSe[1]]:CONTinuity:THReshold?
[:SENSe[1]]:CONTinuity:THReshold? <clist>
```

`<n>`	Threshold resistance in ohms: 0 to 1000
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Also see

None

[:SENSe[1]]:CURRent:AC:APERture

This command sets the aperture for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	60 Hz: 16.67 ms 50 Hz: 20 ms

Usage

```
[:SENSe[1]]:CURRent:AC:APERture <NRf>
[:SENSe[1]]:CURRent:AC:APERture <NRf>, <clist>
[:SENSe[1]]:CURRent:AC:APERture?
[:SENSe[1]]:CURRent:AC:APERture? <clist>
```

`<NRf>`	Integration rate in seconds: ▪ 60 Hz: 1.67e−4 to 1 ▪ 50 Hz: 2e−4 to 1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command is only valid if bandwidth is set to 300 (300 Hz to 300 kHz).

SCPI compliant.

Also see

[:SENSe[1]]:CURRent:AC:NPLCycles (on page 9-86)

[:SENSe[1]]:CURRent:AC:AVERage:COUNt

This command selects the filter count for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:CURRent:AC:AVERage:COUNt <n>
[:SENSe[1]]:CURRent:AC:AVERage:COUNt <n>, <clist>
[:SENSe[1]]:CURRent:AC:AVERage:COUNt?
[:SENSe[1]]:CURRent:AC:AVERage:COUNt? <clist>
```

`<n>`	The filter count: `1` to `100`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The filter count is the number of readings that are acquired and stored in the filter buffer for the averaging calculation. The larger the filter count, the more filtering that is performed.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:CURRent:AC:AVERage:TCONtrol

This command selects the filter mode for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Reset: REPeat System preset: MOVing

Usage

```
[:SENSe[1]]:CURRent:AC:AVERage:TCONtrol <name>
[:SENSe[1]]:CURRent:AC:AVERage:TCONtrol?
```

`<name>`	Repeating filter mode: `REPeat` Moving filter mode: `MOVing`

Details

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:CURRent:AC:AVERage:WINDow

This command specifies a filter window to control the filter threshold.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.1

Usage

```
[:SENSe[1]]:CURRent:AC:AVERage:WINDow <NRf>
[:SENSe[1]]:CURRent:AC:AVERage:WINDow?
```

`<NRf>`	The filter window (in percent of range): `0` to `10`

Details

The digital filter uses a window to control the filter threshold. While the input signal remains in the selected window, A/D conversions continue to be placed in the stack. If the signal changes to a value outside the window, the filter resets, and the filter starts processing again, starting with a new initial conversion value from the A/D converter.

Setting 0 sets the filter window to NONE.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:CURRent:AC:AVERage[:STATe]

This command enables or disables the digital averaging filter for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	*RST: OFF SYSTem:PRESet: ON

Usage

```
[:SENSe[1]]:CURRent:AC:AVERage[:STATe] <b>
[:SENSe[1]]:CURRent:AC:AVERage[:STATe] <b>, <clist>
[:SENSe[1]]:CURRent:AC:AVERage[:STATe]?
[:SENSe[1]]:CURRent:AC:AVERage[:STATe]? <clist>
```

`<b>`	Disable the digital filter: `OFF` Enable the digital filter: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When enabled, readings are filtered using the filter configuration.

If the state is set to `ON`, averaging type is set to `NONE`, and the median state is set to `OFF`, a settings conflict error occurs.

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:CURRent:AC:DETector:BANDwidth

This command selects the detector bandwidth for AC current and AC voltage measurements.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle Measure configuration list	Save settings Measure configuration list	30

Usage

```
[:SENSe[1]]:CURRent:AC:DETector:BANDwidth <NRf>
[:SENSe[1]]:CURRent:AC:DETector:BANDwidth <NRf>, <clist>
[:SENSe[1]]:CURRent:AC:DETector:BANDwidth?
[:SENSe[1]]:CURRent:AC:DETector:BANDwidth? <clist>
```

`<NRf>`	The AC detector bandwidth in Hertz: 3 to 3e5
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

You can set the detector bandwidth to improve measurement accuracy. Select the bandwidth that contains the lowest frequency component of the input signal. For example, if the lowest frequency component of your input signal is 40 Hz, use a bandwidth setting of 30 Hz.

The instrument accepts a parameter value up to 10e6, but it defaults to 3e5.

Example

`FUNC "VOLT:AC"` `VOLT:AC:DET:BAND 30`	Set the measure function to AC volts. Set the detector bandwidth for AC volts to 30 Hz.

Also see

[:SENSe[1]]:CURRent:AC:APERture (on page 9-81)
[:SENSe[1]]:CURRent:AC:NPLCycles (on page 9-86)

[:SENSe[1]]:CURRent:AC:DIGits

This command determines the number of digits that are displayed for measurements on the front panel.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	6

Usage

```
[:SENSe[1]]:CURRent:AC:DIGits <n>
[:SENSe[1]]:CURRent:AC:DIGits <n>, <clist>
[:SENSe[1]]:CURRent:AC:DIGits?
[:SENSe[1]]:CURRent:AC:DIGits? <clist>
```

`<n>`	Display digits: ■ 6½: 7 or `6.5` ■ 5½: 6 or `5.5` ■ 4½: 5 or `4.5` ■ 3½: 4 or `3.5`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command affects how the reading for a measurement is displayed on the front panel of the instrument. It does not affect the number of digits returned in a remote command reading. It also does not affect the accuracy or speed of measurements.

The display digits setting is saved with the function setting, so if you use another function, then return to the function for which you set display digits, the display digits setting you set previously is retained.

The change in digits occurs the next time a measurement is made.

Example

`CURR:AC:DIG 4`	Select 3½ digits for ACI.

Also see

None

[:SENSe[1]]:CURRent:AC:NPLCycles

This command sets the time that the input signal is measured.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	5.0

Usage

```
[:SENSe[1]]:CURRent:AC:NPLCycles <NRf>
[:SENSe[1]]:CURRent:AC:NPLCycles <NRf>, <clist>
[:SENSe[1]]:CURRent:AC:NPLCycles?
[:SENSe[1]]:CURRent:AC:NPLCycles <clist>
```

`<NRf>`	Integration rate in line cycles: ▪ 60 Hz: 0.01 to 60 ▪ 50 Hz: 0.01 to 50
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command is only valid if bandwidth is set to 300 (300 Hz to 300 kHz).

SCPI compliant.

Also see

[:SENSe[1]]:CURRent:AC:APERture (on page 9-81)

[:SENSe[1]]:CURRent:AC:RANGe:AUTO

This command determines if the measurement range is set manually or automatically for the selected measure function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
[:SENSe[1]]:CURRent:AC:RANGe:AUTO <b>
[:SENSe[1]]:CURRent:AC:RANGe:AUTO <b>, <clist>
[:SENSe[1]]:CURRent:AC:RANGe:AUTO?
[:SENSe[1]]:CURRent:AC:RANGe:AUTO? <clist>
```

`<b>`	Set the measurement range manually: `OFF` Set the measurement range automatically: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When autorange is enabled, the instrument automatically goes to the most sensitive AC current range to make the measurement.

SCPI compliant.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:CURRent:AC:RANGe[:UPPer] (on page 9-87)

[:SENSe[1]]:CURRent:AC:RANGe[:UPPer]

This command sets the measurement range for the AC current function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	3

Usage

```
[:SENSe[1]]:CURRent:AC:RANGe[:UPPer] <n>
[:SENSe[1]]:CURRent:AC:RANGe[:UPPer] <n>, <clist>
[:SENSe[1]]:CURRent:AC:RANGe[:UPPer]?
[:SENSe[1]]:CURRent:AC:RANGe[:UPPer]? <clist>
```

`<n>`	Range in amps: 0 to 3.1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Example

`CURR:RANG 0.1`	Select 100 mA range for DCI.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:CURRent:AC:RANGe:AUTO (on page 9-86)

[:SENSe[1]]:CURRent:AC:REFerence

This command contains the relative offset value for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
[:SENSe[1]]:CURRent:AC:REFerence <n>
[:SENSe[1]]:CURRent:AC:REFerence <clist>
[:SENSe[1]]:CURRent:AC:REFerence?
[:SENSe[1]]:CURRent:AC:REFerence? <clist>
```

`<n>`	The reference value: −3.1 to 3.1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

On the resistance function, the input signal is the computed resistance value seen at the input. The final result is the algebraic difference between the input resistance and the reference. You can establish a reference for the current component of a resistance measurement.

When `:SENSe[1]:CURRent:AC:REFerence:ACQuire` is sent, this is changed to the acquired value.

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:CURRent:AC:REFerence:STATe

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:CURRent:AC:REFerence:STATe <b>
[:SENSe[1]]:CURRent:AC:REFerence:STATe <b>, <clist>
[:SENSe[1]]:CURRent:AC:REFerence:STATe?
[:SENSe[1]]:CURRent:AC:REFerence:STATe? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:CURRent:AC:REFerence:ACQuire

This command acquires a measurement and stores it as the relative offset value.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:CURRent:AC:REFerence:ACQuire
[:SENSe[1]]:CURRent:AC:REFerence:ACQuire <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command uses the input signal on the selected channel as the relative offset value.

This command is functional only if the instrument is set to the selected measurement function. Sending this command when another function is selected generates an error.

`[:SENSe[1]]:CURRent:AC:REFerence` is set to the acquired value.

`[:SENSe[1]]:CURRent:AC:REFerence:ACQuire` is SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:CURRent[:DC]:APERture

This command sets the aperture for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	60 Hz: 16.67 ms 50 Hz: 20 ms

Usage

```
[:SENSe[1]]:CURRent[:DC]:APERture <NRf>
[:SENSe[1]]:CURRent[:DC]:APERture <NRf>, <clist>
[:SENSe[1]]:CURRent[:DC]:APERture?
[:SENSe[1]]:CURRent[:DC]:APERture? <clist>
```

`<NRf>`	Integration rate in seconds: ■ 60 Hz: 1.67e−4 to 1 ■ 50 Hz: 2e−4 to 1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:CURRent[:DC]:NPLCycles (on page 9-95)

[:SENSe[1]]:CURRent[:DC]:AVERage:COUNt

This command selects the filter count for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:CURRent[:DC]:AVERage:COUNt <n>
[:SENSe[1]]:CURRent[:DC]:AVERage:COUNt <n>, <clist>
[:SENSe[1]]:CURRent[:DC]:AVERage:COUNt?
[:SENSe[1]]:CURRent[:DC]:AVERage:COUNt? <clist>
```

`<n>`	The filter count: `1` to `100`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The filter count is the number of readings that are acquired and stored in the filter buffer for the averaging calculation. The larger the filter count, the more filtering that is performed.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:CURRent[:DC]:AVERage:TCONtrol

This command selects the filter mode for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Reset: REPeat System preset: MOVing

Usage

```
[:SENSe[1]]:CURRent[:DC]:AVERage:TCONtrol <name>
[:SENSe[1]]:CURRent[:DC]:AVERage:TCONtrol?
```

`<name>`	Select the repeating filter mode: `REPeat` Select the moving filter mode: `MOVing`

Details

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Example

```
CURR:TCON MOV          ' Select the moving filter.
CURR:AVER:WIND 0.01    ' Set filter window to 0.01%.
CURR:AVER:COUN 10      ' Set to filter 10 readings.
CURR:AVER ON           ' Enable filter.
```

This example configures filtering for the dc current function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:CURRent[:DC]:AVERage:WINDow

This command specifies a filter window to control filter threshold.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.1

Usage

```
[:SENSe[1]]:CURRent[:DC]:AVERage:WINDow <NRf>
[:SENSe[1]]:CURRent[:DC]:AVERage:WINDow?
```

`<NRf>`	The filter window (in percent of range): `0` to `10`

Details

The digital filter uses a window to control the filter threshold. While the input signal remains in the selected window, A/D conversions continue to be placed in the stack. If the signal changes to a value outside the window, the filter resets, and the filter starts processing again, starting with a new initial conversion value from the A/D converter.

Setting 0 sets the filter window to NONE.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:CURRent[:DC]:AVERage[:STATe]

This command enables or disables the digital averaging filter for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	*RST: OFF SYSTem:PRESet: ON

Usage

```
[:SENSe[1]]:CURRent[:DC]:AVERage[:STATe] <b>
[:SENSe[1]]:CURRent[:DC]:AVERage[:STATe] <b>, <clist>
[:SENSe[1]]:CURRent[:DC]:AVERage[:STATe]?
[:SENSe[1]]:CURRent[:DC]:AVERage[:STATe]? <clist>
```

`<b>`	Disable the digital filter: `OFF` Enable the digital filter: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When enabled, readings are filtered using the filter configuration.

If the state is set to `ON`, averaging type is set to `NONE`, and the median state is set to `OFF`, a settings conflict error occurs.

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:CURRent[:DC]:DIGits

This command determines the number of digits that are displayed for measurements on the front panel.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	7

Usage

```
[:SENSe[1]]:CURRent[:DC]:DIGits <n>
[:SENSe[1]]:CURRent[:DC]:DIGits <n>, <clist>
[:SENSe[1]]:CURRent[:DC]:DIGits?
[:SENSe[1]]:CURRent[:DC]:DIGits? <clist>
```

`<n>`	Display digits: ■ 6½: 7 or `6.5` ■ 5½: 6 or `5.5` ■ 4½: 5 or `4.5` ■ 3½: 4 or `3.5`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command affects how the reading for a measurement is displayed on the front panel of the instrument. It does not affect the number of digits returned in a remote command reading. It also does not affect the accuracy or speed of measurements.

The display digits setting is saved with the function setting, so if you use another function, then return to the function for which you set display digits, the display digits setting you set previously is retained.

The change in digits occurs the next time a measurement is made.

Example

`CURR:DIG 6`	Select 5½ digits for DCI.

Also see

None

[:SENSe[1]]:CURRent[:DC]:NPLCycles

This command sets the time that the input signal is measured.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	5.0

Usage

```
[:SENSe[1]]:CURRent[:DC]:NPLCycles <NRf>
[:SENSe[1]]:CURRent[:DC]:NPLCycles <NRf>, <clist>
[:SENSe[1]]:CURRent[:DC]:NPLCycles?
[:SENSe[1]]:CURRent[:DC]:NPLCycles <clist>
```

`<NRf>`	Integration rate in line cycles: ▪ 60 Hz: 0.01 to 60 ▪ 50 Hz: 0.01 to 50
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:CURRent[:DC]:APERture (on page 9-90)

[:SENSe[1]]:CURRent[:DC]:RANGe:AUTO

This command determines if the measurement range is set manually or automatically for the selected measure function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
[:SENSe[1]]:CURRent[:DC]:RANGe:AUTO <b>
[:SENSe[1]]:CURRent[:DC]:RANGe:AUTO <b>, <clist>
[:SENSe[1]]:CURRent[:DC]:RANGe:AUTO?
[:SENSe[1]]:CURRent[:DC]:RANGe:AUTO? <clist>
```

`<b>`	Set the measurement range manually: `OFF` Set the measurement range automatically: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When autorange is enabled, the instrument automatically goes to the most sensitive dc current range to make the measurement.

SCPI compliant.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:CURRent[:DC]:RANGe[:UPPer] (on page 9-96)

[:SENSe[1]]:CURRent[:DC]:RANGe[:UPPer]

This command sets the measurement range for dc current function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	3

Usage

```
[:SENSe[1]]:CURRent[:DC]:RANGe[:UPPer] <n>
[:SENSe[1]]:CURRent[:DC]:RANGe[:UPPer] <n>, <clist>
[:SENSe[1]]:CURRent[:DC]:RANGe[:UPPer]?
[:SENSe[1]]:CURRent[:DC]:RANGe[:UPPer]? <clist>
```

`<n>`	Range in amps: 0 to 3.1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:CURRent[:DC]:RANGe:AUTO (on page 9-95)

[:SENSe[1]]:CURRent[:DC]:REFerence

This command contains the relative offset value for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
[:SENSe[1]]:CURRent[:DC]:REFerence <n>
[:SENSe[1]]:CURRent[:DC]:REFerence <clist>
[:SENSe[1]]:CURRent[:DC]:REFerence?
[:SENSe[1]]:CURRent[:DC]:REFerence? <clist>
```

`<n>`	The reference value in amps: −3.1 to 3.1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

On the resistance function, the input signal is the computed resistance value seen at the input. The final result is the algebraic difference between the input resistance and the reference. You can establish a reference for the current component of a resistance measurement.

When `:SENSe[1]:CURRent[:DC]:REFerence:ACQuire` is sent, this is changed to the acquired value.

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:CURRent[:DC]:REFerence:ACQuire

This command acquires a measurement and stores it as the relative offset value.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:CURRent[:DC]:REFerence:ACQuire
[:SENSe[1]]:CURRent[:DC]:REFerence:ACQuire <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command uses the input signal on the selected channel as the relative offset value.

This command is functional only if the instrument is set to the selected measurement function. Sending this command when another function is selected generates an error.

`[:SENSe[1]]:CURRent[:DC]:REFerence` is set to the acquired value.

`[:SENSe[1]]:CURRent[:DC]:REFerence:ACQuire` is SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:CURRent[:DC]:REFerence:STATe

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:CURRent[:DC]:REFerence:STATe <b>
[:SENSe[1]]:CURRent[:DC]:REFerence:STATe <b>, <clist>
[:SENSe[1]]:CURRent[:DC]:REFerence:STATe?
[:SENSe[1]]:CURRent[:DC]:REFerence:STATe? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:DATA:FRESh?

This command returns the last 2750 reading.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

`[:SENSe[1]]:DATA:FRESh?`

Details

This command returns a new reading. The reading reflects what is applied to the input.

When the instrument is making measurements, you can use this command to return the last reading. This command does not trigger a new reading.

This is a better choice than `:FETCh?` because it cannot return the same reading twice. This is a good query to use when triggering by BUS or EXTERNAL, because it waits for a reading to complete if a reading is in progress.

The `:CALC:DATA:FRESh?` query is similar to the `:DATA:FRESh?` query but applies to readings that have math applied to them.

`DATA:FRESh?` can only be used once to return the same reading string. Sending this command again to retrieve the same reading string generates error −230, data corrupt or stale, or causes the GPIB to time out. To use `DATA:FRESh?` again, a new reading must be triggered.

Example 1

```
*RST
:INITiate:CONTinuous OFF;
:ABORt
:TRIGger:SOURce BUS
:SENSe:FUNCtion 'VOLTage:DC'
:SENSe:VOLTage:DC:RANGe:AUTO ON
:TRIGger:COUNt 1
:INITiate
*TRG -or- GET    // Triggers reading (GET is a GPIB general bus command).
:SENSe:DATA:FRESh?
```

This example makes a one-shot reading for DC voltage with a bus trigger using autoranging.

Example 2

```
*RST
:INITiate:CONTinuous OFF;
:ABORt
:TRIGger:SOURce EXTernal;
:TRIGger:DELay:AUTO ON   // Note: Auto trigger delay only takes effect with
                         // trigger source set for BUS or EXTernal.
:SENSe:FUNCtion 'VOLTage:DC'
:SENSe:VOLTage:DC:RANGe:AUTO ON
:INITiate                // External trigger occurs.
:SENSe:DATA:FRESh?
```

This example makes a one-shot reading for DC voltage with an external trigger using autoranging and autodelay.
If a trigger does not occur, this will time out.

Also see

:CALCulate[1]:DATA:FRESh? (on page 9-39)
[:SENSe[1]]:DATA[:LATest]? (on page 9-100)

[:SENSe[1]]:DATA[:LATest]?

This command returns the latest 2750 reading.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:DATA[:LATest]?
```

Details

This command does not trigger a reading. It returns the last reading string. The reading reflects signal processing applied to the input. This query can return old meaningless data.

When the instrument is making measurements, you can use this command to return the last reading. If the instrument is not performing measurements, `DATA[:LATest]?` returns the same reading string.

This command does not affect data in the sample buffer.

When using this command, it is good practice to include reading numbers in the data elements. Reading numbers that do not change indicate that the same data array is being returned.

SCPI compliant.

Also see

:FORMat:ELEMents (on page 9-64)

[:SENSe[1]]:FREQuency:APERture

This command sets the gate time for the frequency.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	1.0

Usage

```
[:SENSe[1]]:FREQuency:APERture <NRf>
[:SENSe[1]]:FREQuency:APERture <NRf>, <clist>
[:SENSe[1]]:FREQuency:APERture?
[:SENSe[1]]:FREQuency:APERture? <clist>
```

`<NRf>`	Gate time for frequency measurements in seconds: 0.01 to 1.0
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

On the front panel, the rate annunciators indicate the following aperture settings:

- SLOW = 1 s
- MED = 0.1 s
- FAST = 0.01 s

For all other aperture times, the rate annunciators are turned off.

SCPI compliant.

Also see

None

[:SENSe[1]]:FREQuency:DIGits

This command determines the number of digits that are displayed for measurements on the front panel.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	7

Usage

```
[:SENSe[1]]:FREQuency:DIGits <n>
[:SENSe[1]]:FREQuency:DIGits <n>, <clist>
[:SENSe[1]]:FREQuency:DIGits?
[:SENSe[1]]:FREQuency:DIGits? <clist>
```

`<n>`	Display digits: ■ 6½: 7 or `6.5` ■ 5½: 6 or `5.5` ■ 4½: 5 or `4.5` ■ 3½: 4 or `3.5`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command affects how the reading for a measurement is displayed on the front panel of the instrument. It does not affect the number of digits returned in a remote command reading. It also does not affect the accuracy or speed of measurements.

The display digits setting is saved with the function setting, so if you use another function, then return to the function for which you set display digits, the display digits setting you set previously is retained.

The change in digits occurs the next time a measurement is made.

Also see

None

[:SENSe[1]]:FREQuency:REFerence

This command contains the relative offset value for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
[:SENSe[1]]:FREQuency:REFerence <n>
[:SENSe[1]]:FREQuency:REFerence <n>, <clist>
[:SENSe[1]]:FREQuency:REFerence?
[:SENSe[1]]:FREQuency:REFerence? <clist>
```

`<n>`	The reference value in Hertz: 0 to 1.5e7
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

On the resistance function, the input signal is the computed resistance value seen at the input. The final result is the algebraic difference between the input resistance and the reference. You can establish a reference for the current component of a resistance measurement.

When `:SENSe[1]:FREQuency:ACQuire` is sent, this is changed to the acquired value.

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:FREQuency:REFerence:ACQuire

This command acquires a measurement and stores it as the relative offset value.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:FREQuency:REFerence:ACQuire
[:SENSe[1]]:FREQuency:REFerence:ACQuire <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command uses the input signal on the selected channel as the relative offset value.

This command is functional only if the instrument is set to the selected measurement function. Sending this command when another function is selected generates an error.

`[:SENSe[1]]:FREQuency:REFerence` is set to the acquired value.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:FREQuency:REFerence:STATe

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:FREQuency:REFerence:STATe <b>
[:SENSe[1]]:FREQuency:REFerence:STATe <b>, <clist>
[:SENSe[1]]:FREQuency:REFerence:STATe?
[:SENSe[1]]:FREQuency:REFerence:STATe? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:FREQuency:THReshold:VOLTage:RANGe

This command selects the threshold range.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:FREQuency:THReshold:VOLTage:RANGe <n>
[:SENSe[1]]:FREQuency:THReshold:VOLTage:RANGe <n>, <clist>
[:SENSe[1]]:FREQuency:THReshold:VOLTage:RANGe?
[:SENSe[1]]:FREQuency:THReshold:VOLTage:RANGe? <clist>
```

`<n>`	Threshold range: 0 to 1010
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command specifies the expected input level. The instrument automatically selects the most sensitive current or voltage threshold range.

Also see

None

[:SENSe[1]]:FRESistance:APERture

This command sets the aperture for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	60 Hz: 16.67 ms 50 Hz: 20 ms

Usage

```
[:SENSe[1]]:FRESistance:APERture <NRf>
[:SENSe[1]]:FRESistance:APERture <NRf>, <clist>
[:SENSe[1]]:FRESistance:APERture?
[:SENSe[1]]:FRESistance:APERture? <clist>
```

`<NRf>`	Integration rate in seconds: ■ 60 Hz: 1.67e−4 to 1 ■ 50 Hz: 2e−4 to 1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:FRESistance:NPLCycles (on page 9-110)

[:SENSe[1]]:FRESistance:AVERage:COUNt

This command selects the filter count for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:FRESistance:AVERage:COUNt <n>
[:SENSe[1]]:FRESistance:AVERage:COUNt <n>, <clist>
[:SENSe[1]]:FRESistance:AVERage:COUNt?
[:SENSe[1]]:FRESistance:COUNt? <clist>
```

`<n>`	The filter count: `1` to `100`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The filter count is the number of readings that are acquired and stored in the filter buffer for the averaging calculation. The larger the filter count, the more filtering that is performed.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:FRESistance:AVERage:TCONtrol

This command selects the filter mode for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Reset: REPeat System preset: MOVing

Usage

```
[:SENSe[1]]:RESistance:AVERage:TCONtrol <name>
[:SENSe[1]]:RESistance:AVERage:TCONtrol?
```

`<name>`	Repeating filter mode: `REPeat` Moving filter mode: `MOVing`

Details

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:FRESistance:AVERage:WINDow

This command specifies a filter window to control filter threshold.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle	Save settings	0.1

Usage

```
[:SENSe[1]]:FRESistance:AVERage:WINDow <NRf>
[:SENSe[1]]:FRESistance:AVERage:WINDow?
```

`<NRf>`	The filter window (in percent of range): `0` to `10`

Details

The digital filter uses a window to control the filter threshold. While the input signal remains in the selected window, A/D conversions continue to be placed in the stack. If the signal changes to a value outside the window, the filter resets, and the filter starts processing again, starting with a new initial conversion value from the A/D converter.

Setting 0 sets the filter window to NONE.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:FRESistance:AVERage[:STATe]

This command enables or disables the digital averaging filter for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	*RST: OFF SYSTem:PRESet: ON

Usage

```
[:SENSe[1]]:FRESistance:AVERage[:STATe] <b>
[:SENSe[1]]:FRESistance:AVERage[:STATe] <b>, <clist>
[:SENSe[1]]:FRESistance:AVERage[:STATe]?
[:SENSe[1]]:FRESistance:AVERage[:STATe]? <clist>
```

`<b>`	Disable the digital filter: `OFF` Enable the digital filter: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When enabled, readings are filtered using the filter configuration.

If the state is set to `ON`, averaging type is set to `NONE`, and the median state is set to `OFF`, a settings conflict error occurs.

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:FRESistance:DCIRcuit

This command enables or disables dry circuit ohms.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:FRESistance:DCIRcuit <b>
[:SENSe[1]]:FRESistance:DCIRcuit <b>, <clist>
[:SENSe[1]]:FRESistance:DCIRcuit?
[:SENSe[1]]:FRESistance:DCIRcuit? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

Enabling dry circuit limits the open-circuit voltage to below 20 mV, which is often required with low-glitch measurements, such as measuring switch and relay contact resistance.

When dry circuit is enabled, offset compensation is automatically enabled.

The instrument does not have to be on the Ω4 function to enable dry circuit ohms. When Ω4 is selected, dry circuit ohms is enabled if `FRES:OCOM` is set to ON.

Also see

None

[:SENSe[1]]:FRESistance:DIGits

This command determines the number of digits that are displayed for measurements on the front panel.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	7

Usage

```
[:SENSe[1]]:FRESistance:DIGits <n>
[:SENSe[1]]:FRESistance:DIGits <n>, <clist>
[:SENSe[1]]:FRESistance:DIGits?
[:SENSe[1]]:FRESistance:DIGits? <clist>
```

`<n>`	Display digits: ■ 6½: 7 or `6.5` ■ 5½: 6 or `5.5` ■ 4½: 5 or `4.5` ■ 3½: 4 or `3.5`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command affects how the reading for a measurement is displayed on the front panel of the instrument. It does not affect the number of digits returned in a remote command reading. It also does not affect the accuracy or speed of measurements.

The display digits setting is saved with the function setting, so if you use another function, then return to the function for which you set display digits, the display digits setting you set previously is retained.

The change in digits occurs the next time a measurement is made.

Also see

None

[:SENSe[1]]:FRESistance:NPLCycles

This command sets the time that the input signal is measured.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	5.0

Usage

```
[:SENSe[1]]:FRESistance:NPLCycles <NRf>
[:SENSe[1]]:FRESistance:NPLCycles <NRf>, <clist>
[:SENSe[1]]:FRESistance:NPLCycles?
[:SENSe[1]]:FRESistance:NPLCycles <clist>
```

`<NRf>`	Integration rate in line cycles: ▪ 60 Hz: 0.01 to 60 ▪ 50 Hz: 0.01 to 50
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:FRESistance:APERture (on page 9-106)

[:SENSe[1]]:FRESistance:OCOMpensated

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:FRESistance:OCOMpensated <b>
[:SENSe[1]]:FRESistance:OCOMpensated <b>, <clist>
[:SENSe[1]]:FRESistance:OCOMpensated?
[:SENSe[1]]:FRESistance:OCOMpensated? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

The instrument does not have to be on the Ω4 function to enable offset-compensated ohms. When Ω4 is selected, offset-compensated ohms is enabled if FRES:OCOM is set to ON.

When using the `<clist>` parameter to configure a scan channel for offset-compensated ohms, that channel must already be set for Ω4. If it is set for another function, the error −221, settings conflict, occurs.

SCPI compliant.

Example

```
*RST              ' One-shot measurement mode (INIT:CONT OFF).
FUNC 'FRES'       ' Select Ω4 function.
FRES:RANG 1e3     ' Select 1 kΩ range.
FRES:OCOM ON      ' Enable offset-compensated ohms.
READ?             ' Trigger and return one reading.
```

This example places the 2750 in a one-shot trigger mode to measure offset-compensated ohms. Whenever READ? is sent, a measurement is triggered, and the measured reading is sent to the computer when the 2750 is addressed to talk.

Also see

[:SENSe[1]]:FRESistance:REFerence (on page 9-113)
[:SENSe[1]]:FRESistance:REFerence:ACQuire (on page 9-113)

[:SENSe[1]]:FRESistance:RANGe:AUTO

This command determines if the measurement range is set manually or automatically for the selected measure function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
[:SENSe[1]]:FRESistance:RANGe:AUTO <b>
[:SENSe[1]]:FRESistance:RANGe:AUTO <b>, <clist>
[:SENSe[1]]:FRESistance:RANGe:AUTO?
[:SENSe[1]]:FRESistance:RANGe:AUTO? <clist>
```

`<b>`	Set the measurement range manually: `OFF` Set the measurement range automatically: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When autorange is enabled for Ω4 measurements, the instrument automatically goes to the most sensitive range to make the measurement.

SCPI compliant.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:FRESistance:RANGe[:UPPer] (on page 9-112)

[:SENSe[1]]:FRESistance:RANGe[:UPPer]

This command sets the measurement range for the Ω4 function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	120e6

Usage

```
[:SENSe[1]]:FRESistance:RANGe[:UPPer] <n>
[:SENSe[1]]:FRESistance:RANGe[:UPPer] <n>, <clist>
[:SENSe[1]]:FRESistance:RANGe[:UPPer]?
[:SENSe[1]]:FRESistance:RANGe[:UPPer]? <clist>
```

`<n>`	Range: 0 to 120e6
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:FRESistance:RANGe:AUTO (on page 9-111)

[:SENSe[1]]:FRESistance:REFerence

This command contains the relative offset value for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
[:SENSe[1]]:FRESistance:REFerence <n>
[:SENSe[1]]:FRESistance:REFerence <n>, <clist>
[:SENSe[1]]:FRESistance:REFerence?
[:SENSe[1]]:FRESistance:REFerence? <clist>
```

`<n>`	The reference value in ohms: 0 to 120e6
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

On the resistance function, the input signal is the computed resistance value seen at the input. The final result is the algebraic difference between the input resistance and the reference. You can establish a reference for the current component of a resistance measurement.

When `[:SENSe[1]]:FREFerence:ACQuire` is sent, this is changed to the acquired value.

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:FRESistance:REFerence:ACQuire

This command acquires a measurement and stores it as the relative offset value.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:FRESistance:REFerence:ACQuire
[:SENSe[1]]:FRESistance:REFerence:ACQuire <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command uses the input signal on the selected channel as the relative offset value.

This command is functional only if the instrument is set to the selected measurement function. Sending this command when another function is selected generates an error.

`[:SENSe[1]]:FRESistance:REFerence` is set to the acquired value.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:FRESistance:REFerence:STATe

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:FRESistance:REFerence:STATe <b>
[:SENSe[1]]:FRESistance:REFerence:STATe <b>, <clist>
[:SENSe[1]]:FRESistance:REFerence:STATe?
[:SENSe[1]]:FRESistance:REFerence:STATe? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:FUNCtion

This command selects the active measure function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	VOLT

Usage

```
[:SENSe[1]]:FUNCtion "<function>"
[:SENSe[1]]:FUNCtion "<function>", clist>
[:SENSe[1]]:FUNCtion?
[:SENSe[1]]:FUNCtion? <clist>
```

`<function>`	A string that contains the measure function: ▪ `VOLTage[:DC]` ▪ `VOLTage:AC` ▪ `CURRent[:DC]` ▪ `CURRent:AC` ▪ `RESistance` ▪ `FRESistance` ▪ `TEMPerature` ▪ `CONTinuity` ▪ `FREQuency` ▪ `PERiod` ▪ `CONTinuity`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

Enclose the `<function>` parameter in single or double quotes. For example, these two commands are equivalent:

```
:func 'volt'
:func "volt"
```

Each measurement function stores its own setup configuration, such as settings for range, speed, resolution, filter, and relative offset. This eliminates the need to reprogram setup conditions when you switch from one function to another.

NOTE

A function does not have to be selected to set the configuration. When the function is later selected, it uses the programmed states.

If you are using the `<clist>` command to configure a scan channel, the scan channel must be set to the appropriate function before sending other commands to configure it.

SCPI compliant.

Example 1

`SENS:FUNC "RES"`	Make the resistance function the active function.

Example 2

```
FUNC 'FRES', (@101)   ' Set scan channel 101 to Ω4 function.
FRES:OCOM ON, (@101)  ' Enable offset compensated ohms for scan
                      ' channel 101.
```

This example sets scan channel 101 to use offset compensated ohms. If scan channel 101 was not first set for the Ω4 function, error +700, Invalid function in scanlist, occurs when trying to enable offset compensated ohms for that channel. Details on scanning are provided in Switching and scanning (on page 3-1).

Example 3

```
SYST:PRES         ' Continuous measurement mode (INIT:CONT ON).
FUNC 'VOLT:AC'    ' Select ACV function.
DATA?             ' Request last measured reading.
```

This example places the 2750 in a continuous trigger mode to measure ACV. Whenever `DATA?` is sent, the last measured reading is sent to the computer when the 2750 is addressed to talk.

Example 4

```
FUNC 'VOLT',(@101)   ' Configure scan channel 101 for DCV.
FUNC 'RES',(@102)    ' Configure scan channel 102 for Ω2.
FUNC 'CURR',(@121)   ' Configure scan channel 121 for DCI.
```

This example configures scan channels 101, 102, and 121 of a Model 7700 installed in slot 1. When channel 101 is scanned, DCV is selected. When channel 102 is scanned, Ω2 is selected. When channel 121 is scanned, DCI is selected.

Also see

None

[:SENSe[1]]:PERiod:APERture

This command sets the gate time for the period.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	1.0

Usage

```
[:SENSe[1]]:PERiod:APERture <NRf>
[:SENSe[1]]:PERiod:APERture <NRf>, <clist>
[:SENSe[1]]:PERiod:APERture?
[:SENSe[1]]:PERiod:APERture? <clist>
```

`<NRf>`	Gate time for period measurements in seconds: 0.01 to 1.0
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

On the front panel, the rate annunciators indicate the following aperture settings:

- SLOW = 1 s
- MED = 0.1 s
- FAST = 0.01 s

For all other aperture times, the rate annunciators are turned off.

SCPI compliant.

Also see

None

[:SENSe[1]]:PERiod:DIGits

This command determines the number of digits that are displayed for measurements on the front panel.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	7

Usage

```
[:SENSe[1]]:PERiod:DIGits <n>
[:SENSe[1]]:PERiod:DIGits <n>, <clist>
[:SENSe[1]]:PERiod:DIGits?
[:SENSe[1]]:PERiod:DIGits? <clist>
```

`<n>`	Display digits: ■ 6½: 7 or `6.5` ■ 5½: 6 or `5.5` ■ 4½: 5 or `4.5` ■ 3½: 4 or `3.5`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command affects how the reading for a measurement is displayed on the front panel of the instrument. It does not affect the number of digits returned in a remote command reading. It also does not affect the accuracy or speed of measurements.

The display digits setting is saved with the function setting, so if you use another function, then return to the function for which you set display digits, the display digits setting you set previously is retained.

The change in digits occurs the next time a measurement is made.

Also see

None

[:SENSe[1]]:PERiod:REFerence

This command contains the relative offset value for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
[:SENSe[1]]:PERiod:REFerence <n>
[:SENSe[1]]:PERiod:REFerence <n>, <clist>
[:SENSe[1]]:PERiod:REFerence?
[:SENSe[1]]:PERiod:REFerence? <clist>
```

`<n>`	The reference value in seconds: 0 to 1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

On the resistance function, the input signal is the computed resistance value seen at the input. The final result is the algebraic difference between the input resistance and the reference. You can establish a reference for the current component of a resistance measurement.

When `:SENSe[1]:PERiod:REFerence:ACQuire` is sent, this is changed to the acquired value.

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:PERiod:REFerence:ACQuire

This command acquires a measurement and stores it as the relative offset value.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:PERiod:REFerence:ACQuire
[:SENSe[1]]:PERiod:REFerence:ACQuire <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command uses the input signal on the selected channel as the relative offset value.

This command is functional only if the instrument is set to the selected measurement function. Sending this command when any other function is selected causes an error.

`[:SENSe[1]]:PERiod:REFerence` is set to the acquired value.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:PERiod:REFerence:STATe

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:PERiod:REFerence:STATe <b>
[:SENSe[1]]:PERiod:REFerence:STATe <b>, <clist>
[:SENSe[1]]:PERiod:REFerence:STATe?
[:SENSe[1]]:PERiod:REFerence:STATe? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:PERiod:THReshold:VOLTage:RANGe

This command selects the threshold range.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:PERiod:THReshold:VOLTage:RANGe <n>
[:SENSe[1]]:PERiod:THReshold:VOLTage:RANGe <n>, <clist>
[:SENSe[1]]:PERiod:THReshold:VOLTage:RANGe?
[:SENSe[1]]:PERiod:THReshold:VOLTage:RANGe? <clist>
```

`<n>`	Threshold range: 0 to 1010
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command specifies the expected input level. The instrument automatically selects the most sensitive current or voltage threshold range.

Also see

Measurement range (on page 4-7)

[:SENSe[1]]:RATio:DELay

This command sets the channel average.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.5 s

Usage

```
[:SENSe[1]]:RATio:DELay <NRf>
[:SENSe[1]]:RATio:DELay <NRf>, <clist>
[:SENSe[1]]:RATio:DELay?
[:SENSe[1]]:RATio:DELay? <clist>
```

`<NRf>`	Delay in seconds: 0 to 99999.999
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

`:RATio:DELay` and `:CAVerage:DELay` set the delay between two channel measurements for the enabled calculation. This delay is applied after the trigger delay in the trigger model. This delay cannot be set from the front panel. The 0.5 s default delay keeps the relays from cycling too fast. Setting a shorter delay may shorten the life of the relays.

It does not matter which of the two commands you use to set the delay. The set delay affects both the ratio and channel average.

Also see

:SENSe[1]:CAVerage:DELay (on page 9-79)

[:SENSe[1]]:RATio[:STATe]

This command enables or disables channel ratio.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:RATio[:STATe] <b>
[:SENSe[1]]:RATio[:STATe] <b>, <clist>
[:SENSe[1]]:RATio[:STATe]?
[:SENSe[1]]:RATio[:STATe]? <clist>
```

`<b>`	Enable: `ON` Disable: `OFF`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

Enabling ratio disables channel average and conversely, enabling channel average disables ratio.

Example

```
*RST                ' One-shot measure mode.
FUNC 'VOLT'         ' Select DCV function.
ROUT:CLOS (@102)    ' Close channel 102.
RAT ON              ' Enable the ratio calculation.
READ?               ' Read the result of the calculation.
```

This command sequence performs the ratio calculation using primary channel 102 of the Model 7700. After READ? is sent, the 2750 must be addressed to talk to return the result of the calculation.

Also see

:SENSe[1]:CAVerage[:STATe] (on page 9-80)
:SENSe[1]:RATio:DELay (on page 9-121)

[:SENSe[1]]:RESistance:APERture

This command sets the aperture for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	60 Hz: 16.67 ms 50 Hz: 20 ms

Usage

```
[:SENSe[1]]:RESistance:APERture <NRf>
[:SENSe[1]]:RESistance:APERture <NRf>, <clist>
[:SENSe[1]]:RESistance:APERture?
[:SENSe[1]]:RESistance:APERture? <clist>
```

`<NRf>`	Integration rate in seconds: ■ 60 Hz: 1.67e−4 to 1 ■ 50 Hz: 2e−4 to 1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:RESistance:NPLCycles (on page 9-127)

[:SENSe[1]]:RESistance:AVERage:COUNt

This command selects the filter count for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:RESistance:AVERage:COUNt <n>
[:SENSe[1]]:RESistance:AVERage:COUNt <n>, <clist>
[:SENSe[1]]:RESistance:AVERage:COUNt?
[:SENSe[1]]:RESistance:COUNt? <clist>
```

`<n>`	The filter count: `1` to `100`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The filter count is the number of readings that are acquired and stored in the filter buffer for the averaging calculation. The larger the filter count, the more filtering that is performed.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:RESistance:AVERage:TCONtrol

This command selects the filter mode for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Reset: REPeat System preset: MOVing

Usage

```
[:SENSe[1]]:RESistance:AVERage:TCONtrol <name>
[:SENSe[1]]:RESistance:AVERage:TCONtrol?
```

`<name>`	Repeating filter mode: `REPeat` Moving filter mode: `MOVing`

Details

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:RESistance:AVERage:WINDow

This command specifies a filter window to control filter threshold.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.1

Usage

```
[:SENSe[1]]:RESistance:AVERage:WINDow <NRf>
[:SENSe[1]]:RESistance:AVERage:WINDow?
```

`<NRf>`	The filter window (in percent of range): `0` to `10`

Details

The digital filter uses a window to control the filter threshold. While the input signal remains in the selected window, A/D conversions continue to be placed in the stack. If the signal changes to a value outside the window, the filter resets, and the filter starts processing again, starting with a new initial conversion value from the A/D converter.

Setting 0 sets the filter window to NONE.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:RESistance:AVERage[:STATe]

This command enables or disables the digital averaging filter for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	*RST: OFF SYSTem:PRESet: ON

Usage

```
[:SENSe[1]]:RESistance:AVERage[:STATe] <b>
[:SENSe[1]]:RESistance:AVERage[:STATe] <b>, <clist>
[:SENSe[1]]:RESistance:AVERage[:STATe]?
[:SENSe[1]]:RESistance:AVERage[:STATe]? <clist>
```

`<b>`	Disable the digital filter: `OFF` or `0` Enable the digital filter: `ON` or `1`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When enabled, readings are filtered using the filter configuration.

If the state is set to `ON`, averaging type is set to `NONE`, and the median state is set to `OFF`, a settings conflict error occurs.

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:RESistance:DIGits

This command determines the number of digits that are displayed for measurements on the front panel.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	7

Usage

```
[:SENSe[1]]:RESistance:DIGits <n>
[:SENSe[1]]:RESistance:DIGits <n>, <clist>
[:SENSe[1]]:RESistance:DIGits?
[:SENSe[1]]:RESistance:DIGits? <clist>
```

`<n>`	Display digits: ■ 6½: 7 or `6.5` ■ 5½: 6 or `5.5` ■ 4½: 5 or `4.5` ■ 3½: 4 or `3.5`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command affects how the reading for a measurement is displayed on the front panel of the instrument. It does not affect the number of digits returned in a remote command reading. It also does not affect the accuracy or speed of measurements.

The display digits setting is saved with the function setting, so if you use another function, then return to the function for which you set display digits, the display digits setting you set previously is retained.

The change in digits occurs the next time a measurement is made.

Example

`FUNC 'RES' (@101:110)` `RES:DIG 4.5,(@101:110)`	Select the Ω2 function for channels 101 through 110 of the Model 7700. Set the scan channels to 4½ digits.

Also see

None

[:SENSe[1]]:RESistance:NPLCycles

This command sets the time that the input signal is measured.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	5.0

Usage

```
[:SENSe[1]]:RESistance:NPLCycles <NRf>
[:SENSe[1]]:RESistance:NPLCycles <NRf>, <clist>
[:SENSe[1]]:RESistance:NPLCycles?
[:SENSe[1]]:RESistance:NPLCycles <clist>
```

`<NRf>`	Integration rate in line cycles: ▪ 60 Hz: 0.01 to 60 ▪ 50 Hz: 0.01 to 50
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:RESistance:APERture (on page 9-123)

[:SENSe[1]]:RESistance:RANGe:AUTO

This command determines if the measurement range is set manually or automatically for the selected measure function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
[:SENSe[1]]:RESistance:RANGe:AUTO <b>
[:SENSe[1]]:RESistance:RANGe:AUTO <b>, <clist>
[:SENSe[1]]:RESistance:RANGe:AUTO?
[:SENSe[1]]:RESistance:RANGe:AUTO? <clist>
```

`<b>`	Set the measurement range manually: `OFF` Set the measurement range automatically: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When autorange is enabled, the instrument automatically goes to the most sensitive range to perform the 2Ω measurement.

SCPI compliant.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:RESistance:RANGe[:UPPer] (on page 9-128)

[:SENSe[1]]:RESistance:RANGe[:UPPer]

This command sets the measurement range for Ω2 function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	120e6

Usage

```
[:SENSe[1]]:RESistance:RANGe[:UPPer] <n>
[:SENSe[1]]:RESistance:RANGe[:UPPer] <n>, <clist>
[:SENSe[1]]:RESistance:RANGe[:UPPer]?
[:SENSe[1]]:RESistance:RANGe[:UPPer]? <clist>
```

`<n>`	Range: 0 to 120e6
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Example

`RES:RANG 2e3`	Select 10 kΩ range for Ω2.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:RESistance:RANGe:AUTO (on page 9-127)

[:SENSe[1]]:RESistance:REFerence

This command contains the relative offset value for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
[:SENSe[1]]:RESistance:REFerence <n>
[:SENSe[1]]:RESistance:REFerence <n>, <clist>
[:SENSe[1]]:RESistance:REFerence?
[:SENSe[1]]:RESistance:REFerence? <clist>
```

`<n>`	The reference value: 0 to 120e6
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

On the resistance function, the input signal is the computed resistance value seen at the input. The final result is the algebraic difference between the input resistance and the reference. You can establish a reference for the current component of a resistance measurement.

When `:SENSe[1]:RESistance:REFerence:ACQuire` is sent, this is changed to the acquired value.

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:RESistance:REFerence:ACQuire

This command acquires a measurement and stores it as the relative offset value.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:RESistance:REFerence:ACQuire
[:SENSe[1]]:RESistance:REFerence:ACQuire <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command uses the input signal on the selected channel as the relative offset value.

This command is functional only if the instrument is set to the selected measurement function. Sending this command when any other function is selected causes an error.

`[:SENSe[1]]:RESistance:REFerence` is set to the acquired value.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:RESistance:REFerence:STATe

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:RESistance:REFerence:STATe <b>
[:SENSe[1]]:RESistance:REFerence:STATe <b>, <clist>
[:SENSe[1]]:RESistance:REFerence:STATe?
[:SENSe[1]]:RESistance:REFerence:STATe? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:TEMPerature:APERture

This command sets the aperture for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	60 Hz: 16.67 ms 50 Hz: 20 ms

Usage

```
[:SENSe[1]]:TEMPerature:APERture <NRf>
[:SENSe[1]]:TEMPerature:APERture <NRf>, <clist>
[:SENSe[1]]:TEMPerature:APERture?
[:SENSe[1]]:TEMPerature:APERture? <clist>
```

`<NRf>`	Integration rate in seconds: ■ 60 Hz: 1.67e−4 to 1 ■ 50 Hz: 2e−4 to 1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:TEMPerature:NPLCycles (on page 9-140)

[:SENSe[1]]:TEMPerature:AVERage:COUNt

This command selects the filter count for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:TEMPerature:AVERage:COUNt <n>
[:SENSe[1]]:TEMPerature:AVERage:COUNt <n>, <clist>
[:SENSe[1]]:TEMPerature:AVERage:COUNt?
[:SENSe[1]]:TEMPerature:COUNt? <clist>
```

`<n>`	The filter count: `1` to `100`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The filter count is the number of readings that are acquired and stored in the filter buffer for the averaging calculation. The larger the filter count, the more filtering that is performed.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:TEMPerature:AVERage:TCONtrol

This command selects the filter mode for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Reset: REPeat System preset: MOVing

Usage

```
[:SENSe[1]]:TEMPerature:AVERage:TCONtrol <name>
[:SENSe[1]]:TEMPerature:AVERage:TCONtrol?
```

`<name>`	Repeating filter mode: `REPeat` Moving filter mode: `MOVing`

Details

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:TEMPerature:AVERage:WINDow

This command specifies a filter window to control filter threshold.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.1

Usage

```
[:SENSe[1]]:TEMPerature:AVERage:WINDow <NRf>
[:SENSe[1]]:TEMPerature:AVERage:WINDow?
```

`<NRf>`	The filter window (in percent of range): `0` to `10`

Details

The digital filter uses a window to control the filter threshold. While the input signal remains in the selected window, A/D conversions continue to be placed in the stack. If the signal changes to a value outside the window, the filter resets, and the filter starts processing again, starting with a new initial conversion value from the A/D converter.

Setting 0 sets the filter window to NONE.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:TEMPerature:AVERage[:STATe]

This command enables or disables the digital averaging filter for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	*RST: OFF SYSTem:PRESet: ON

Usage

```
[:SENSe[1]]:TEMPerature:AVERage[:STATe] <b>
[:SENSe[1]]:TEMPerature:AVERage[:STATe] <b>, <clist>
[:SENSe[1]]:TEMPerature:AVERage[:STATe]?
[:SENSe[1]]:TEMPerature:AVERage[:STATe]? <clist>
```

`<b>`	Disable the digital filter: `OFF` Enable the digital filter: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When enabled, readings are filtered using the filter configuration.

If the state is set to `ON`, averaging type is set to `NONE`, and the median state is set to `OFF`, a settings conflict error occurs.

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:TEMPerature:DIGits

This command determines the number of digits that are displayed for measurements on the front panel.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	6

Usage

```
[:SENSe[1]]:TEMPerature:DIGits <n>
[:SENSe[1]]:TEMPerature:DIGits <n>, <clist>
[:SENSe[1]]:TEMPerature:DIGits?
[:SENSe[1]]:TEMPerature:DIGits? <clist>
```

`<n>`	Display digits: ■ 6½: 7 or `6.5` ■ 5½: 6 or `5.5` ■ 4½: 5 or `4.5` ■ 3½: 4 or `3.5`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command affects how the reading for a measurement is displayed on the front panel of the instrument. It does not affect the number of digits returned in a remote command reading. It also does not affect the accuracy or speed of measurements.

The display digits setting is saved with the function setting, so if you use another function, then return to the function for which you set display digits, the display digits setting you set previously is retained.

The change in digits occurs the next time a measurement is made.

Also see

None

[:SENSe[1]]:TEMPerature:FRTD:ALPHa

This command contains the alpha value of a user-defined RTD.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle Measure configuration list	Save settings Measure configuration list	0.00385

Usage

```
[:SENSe[1]]:TEMPerature:FRTD:ALPHa <NRf>
[:SENSe[1]]:TEMPerature:FRTD:ALPHa <NRf>, <clist>
[:SENSe[1]]:TEMPerature:FRTD:ALPHa?
[:SENSe[1]]:TEMPerature:FRTD:ALPHa? <clist>
```

`<NRf>`	The alpha constant of the user-defined RTD: `0` to `0.01`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When this command is selected, the USER RTD type is automatically selected. This attribute is only valid when the function is set to temperature.

Example

`:FUNC "TEMP"`	Set the measure function to temperature.
`:TEMP:TRANsducer FRTD`	Set the transducer type to 3-wire RTD.
`:TEMP:FRTD:TYPE USER`	Set the RTD type to User.
`:TEMP:FRTD:ALPH 0.00390`	Set the alpha RTD value to 0.00390.

Also see

[:SENSe[1]]:TEMPerature:FRTD:TYPE (on page 9-139)
[:SENSe[1]]:TEMPerature:TRANsducer (on page 9-147)

[:SENSe[1]]:TEMPerature:FRTD:BETA

This command contains the beta value of a user-defined RTD.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.111

Usage

```
[:SENSe[1]]:TEMPerature:FRTD:BETA <NRf>
[:SENSe[1]]:TEMPerature:FRTD:BETA <NRf>, <clist>
[:SENSe[1]]:TEMPerature:FRTD:BETA?
[:SENSe[1]]:TEMPerature:FRTD:BETA? <clist>
```

`<NRf>`	The beta constant of the user-defined RTD: `0` to 1.00
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When this command is selected, the USER RTD type is automatically selected. This attribute is only valid when the function is set to temperature.

Example

`:FUNC "TEMP"`	Set the measure function to temperature.
`:TEMP:TRANsducer FRTD`	Set the transducer type to RTD.
`:TEMP:FRTD:TYPE USER`	Set the RTD type to User.
`:TEMP:FRTD:ALPH 0.005`	Set the alpha RTD value to 0.005.
`:TEMP:FRTD:BETA 0.115`	Set the beta RTD value to 0.115.
`:TEMP:FRTD:ZERO 120`	Set the zero RTD value to 120.

Also see

[:SENSe[1]]:TEMPerature:FRTD:TYPE (on page 9-139)
[:SENSe[1]]:TEMPerature:TRANsducer (on page 9-147)

[:SENSe[1]]:TEMPerature:FRTD:DELTa

This command contains the delta value of a user-defined RTD.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle Measure configuration list	Save settings Measure configuration list	1.507

Usage

```
[:SENSe[1]]:TEMPerature:FRTD:DELTa <NRf>
[:SENSe[1]]:TEMPerature:FRTD:DELTa <NRf>, <clist>
[:SENSe[1]]:TEMPerature:FRTD:DELTa?
[:SENSe[1]]:TEMPerature:FRTD:DELTa? <clist>
```

`<NRf>`	The beta constant of the user-defined RTD: `0` to 5.00
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When this command is selected, the USER RTD type is automatically selected. This attribute is only valid when the function is set to temperature.

Example

`:FUNC "TEMP"`	Set the measure function to temperature.
`:TEMP:TRANsducer FRTD`	Set the transducer type to RTD.
`:TEMP:FRTD:TYPE USER`	Set the RTD type to User.
`:TEMP:FRTD:ALPH 0.005`	Set the alpha RTD value to 0.005.
`:TEMP:FRTD:DELT 0.00385`	Set the delta RTD value to 0.00385.
`:TEMP:FRTD:ZERO 120`	Set the zero RTD value to 120.

Also see

[:SENSe[1]]:TEMPerature:FRTD:TYPE (on page 9-139)
[:SENSe[1]]:TEMPerature:TRANsducer (on page 9-147)

[:SENSe[1]]:TEMPerature:FRTD:RZERo

This command contains the zero value of a user-defined RTD.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	100

Usage

```
[:SENSe[1]]:TEMPerature:FRTD:RZERo <NRf>
[:SENSe[1]]:TEMPerature:FRTD:RZERo <NRf>, <clist>
[:SENSe[1]]:TEMPerature:FRTD:RZERo?
[:SENSe[1]]:TEMPerature:FRTD:RZERo? <clist>
```

`<NRf>`	The zero value of the user-defined RTD: `0` to `10000`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command sets the ohms at 0°C.

When this command is selected, the USER RTD type is automatically selected. This attribute is only valid when the function is set to temperature.

Example

`:FUNC "TEMP"`	Set the measure function to temperature.
`:TEMP:TRANsducer FRTD`	Set the transducer type to RTD.
`:TEMP:FRTD:TYPE USER`	Set the RTD type to User.
`:TEMP:FRTD:ALPH 0.00385`	Set the alpha RTD value to 0.00385.
`:TEMP:FRTD:ZERO 120`	Set the zero RTD value to 120.

Also see

[:SENSe[1]]:TEMPerature:FRTD:TYPE (on page 9-139)
[:SENSe[1]]:TEMPerature:TRANsducer (on page 9-147)

[:SENSe[1]]:TEMPerature:FRTD:TYPE

This command contains the type of 4-wire RTD that is being used.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	PT100

Usage

```
[:SENSe[1]]:TEMPerature:FRTD:TYPE <name>
[:SENSe[1]]:TEMPerature:FRTD:TYPE <name>, <clist>
[:SENSe[1]]:TEMPerature:FRTD:TYPE?
[:SENSe[1]]:TEMPerature:FRTD:TYPE? <clist>
```

`<type>`	The type of 4-wire RTD: ■ PT100: `PT100` ■ PT385: `PT385` ■ PT3916: `PT3916` ■ D100: `D100` ■ F100: `F100` ■ User-specified type: `USER`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The transducer type must be set to temperature and the transducer must be set to 4-wire RTD before you can set the RTD type.

Example

`:FUNC "TEMP"`
`:TEMP:TRANsducer FRTD`
`:TEMP:FRTD:TYPE PT3916`
	Set the measure function to temperature.
	Set the transducer type to 4-wire RTD.
	Set the RTD type to PT3916.

Also see

[:SENSe[1]]:TEMPerature:TRANsducer (on page 9-147)

[:SENSe[1]]:TEMPerature:NPLCycles

This command sets the time that the input signal is measured.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	5.0

Usage

```
[:SENSe[1]]:TEMPerature:NPLCycles <NRf>
[:SENSe[1]]:TEMPerature:NPLCycles <NRf>, <clist>
[:SENSe[1]]:TEMPerature:NPLCycles?
[:SENSe[1]]:TEMPerature:NPLCycles <clist>
```

`<NRf>`	Integration rate in line cycles: ▪ 60 Hz: 0.01 to 60 ▪ 50 Hz: 0.01 to 50
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:TEMPerature:APERture (on page 9-131)

[:SENSe[1]]:TEMPerature:REFerence

This command contains the relative offset value for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
[:SENSe[1]]:TEMPerature:REFerence <n>
[:SENSe[1]]:TEMPerature:REFerence <n>, <clist>
[:SENSe[1]]:TEMPerature:REFerence?
[:SENSe[1]]:TEMPerature:REFerence? <clist>
```

`<n>`	Specify reference in °C: −200 to 1820
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

On the resistance function, the input signal is the computed resistance value seen at the input. The final result is the algebraic difference between the input resistance and the reference. You can establish a reference for the current component of a resistance measurement.

When `:SENSe[1]:TEMPerature:REFerence:ACQuire` is sent, this is changed to the acquired value.

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:TEMPerature:REFerence:ACQuire

This command acquires a measurement and stores it as the relative offset value.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:TEMPerature:REFerence:ACQuire
[:SENSe[1]]:TEMPerature:REFerence:ACQuire <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command uses the input signal on the selected channel as the relative offset value.

This command is functional only if the instrument is set to the selected measurement function. Sending this command when any other function is selected causes an error.

`[:SENSe[1]]:TEMPerature:REFerence` is set to the acquired value.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:TEMPerature:REFerence:STATe

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:TEMPerature:REFerence:STATe <b>
[:SENSe[1]]:TEMPerature:REFerence:STATe <b>, <clist>
[:SENSe[1]]:TEMPerature:REFerence:STATe?
[:SENSe[1]]:TEMPerature:REFerence:STATe? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:TEMPeratureTCouple:ODETect

This command determines if the detection of open leads is enabled or disabled.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle Measure configuration list	Save settings Measure configuration list	OFF

Usage

```
[:SENSe[1]]:TEMPerature:TCouple:ODETect <b>
[:SENSe[1]]:TEMPerature:TCouple:ODETect <b>, <clist>
[:SENSe[1]]:TEMPerature:TCouple:ODETect?
[:SENSe[1]]:TEMPerature:TCouple:ODETect? <clist>
```

`<state>`	Disable: `OFF` or `0` Enable: `ON` or `1`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

For temperature measurements, this is only available when the transducer is set to a thermocouple or one of the RTDs.

Long lengths of thermocouple wire can have a large amount of capacitance, which is seen at the input of the DMM. If an intermittent open occurs in the thermocouple circuit, the capacitance can cause an erroneous on-scale reading. The open thermocouple detection circuit, when enabled, applies a 100 µA pulse of current to the thermocouple before the start of each temperature measurement.

Example

`TEMP:TRAN TC`	Set the transducer type to thermocouple.
`TEMP:TC:TYPE K`	Set the thermocouple type to K.
`UNIT:TEMP CELS`	Set the units to Celsius.
`TEMP:TC:ODET OFF`	Turn open lead detection off.

Also see

None

[:SENSe[1]]:TEMPerature:TCouple:RJUNction:RSELect

This command selects the type of the thermocouple reference junction.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	INTernal

Usage

```
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:RSELect <type>
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:RSELect <type>, <clist>
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:RSELect?
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:RSELect? <clist>
:TEMPerature:RJUNction:RSELect <type>
:TEMPerature:RJUNction:SIMulated <type>, <clist>
:TEMPerature:RJUNction:SIMulated?
:TEMPerature:RJUNction:SIMulated? <clist>
```

`<type>`	The type of reference junction: ▪ `SIMulated` ▪ `INTernal` ▪ `EXTernal`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

To use the external reference junction, the scan channel (101, 201, 301, 401, or 501) to be used for the measurement must already be configured to use a thermistor or 4-wire RTD transducer. Otherwise, error −221, Settings Conflict Error, occurs. The `[:SENSe[1]]:TEMPerature:TRANsducer` command selects the transducer.

If a Model 7700, 7706, or 7708 is installed, the default sensor junction is internal. Otherwise, the simulated (23 °C) junction is selected.

When using multiple channel operation (`ROUT:MULT` command) to connect a switching module channel to the DMM for thermocouple temperature measurements, the simulated reference junction is used if the internal or external reference junction is selected. The ERR annunciator turns on to indicate that the integrity of the temperature reading is questionable.

Also see

[:SENSe[1]]:TEMPerature:TCouple:RJUNction:SIMulated (on page 9-144)
[:SENSe[1]]:TEMPerature:TRANsducer (on page 9-147)

[:SENSe[1]]:TEMPerature:TCouple:RJUNction:SIMulated

This command sets the simulated reference temperature of the thermocouple reference junction.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	23 °C

Usage

```
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:SIMulated <n1>
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:SIMulated <n1>, <clist>
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:SIMulated?
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:SIMulated? <clist>
:TEMPerature:RJUNction:SIMulated? <n2>
:TEMPerature:RJUNction:SIMulated? <n2>, <clist>
:TEMPerature:RJUNction:SIMulated?
:TEMPerature:RJUNction:SIMulated? <clist>
```

`<n1>`	The temperature: ■ °C: `0` to `65` ■ °F: `32` to `149` ■ K: `273` to `338`
`<n2>`	The temperature: ■ °C: `0` to `50` ■ °F: `32` to `122` ■ K: `273` to `323`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command applies to the temperature function when the reference junction is set to simulated. It allows you to set the simulated reference temperature value.

The units for the simulated reference temperature depend on the present temperature measurement units as set by UNIT:TEMPerature.

Example

```
*RST                  ' One-shot measurement mode (INIT:CONT OFF).
FUNC 'TEMP'           ' Select TEMP function.
UNIT:TEMP F           ' Select °F TEMP units.
TEMP:TRAN TC          ' Select thermocouple transducer.
TEMP:TC:TYPE J        ' Select type J thermocouple.
TEMP:RJUN:RSEL SIM    ' Select simulated reference junction.
TEMP:RJUN:SIM 32      ' Set reference temperature to 32°F (ice point).
ROUT:CLOS (@101)      ' Close channel 101.
INIT                  ' Trigger one measurement.
DATA?                 ' Return measured reading.
```

This example assumes a Model 7700 switching module is installed in slot 1. It places the 2750 in one-shot trigger mode to perform a thermocouple temperature measurement at channel 101. With channel 101 closed, the `INIT` command triggers one measurement, and the `DATA?` command sends the measured reading to the computer when it is addressed to talk.

Also see

:SENSe:TEMPerature:TCouple:RJUNction:RSELect (on page 9-143)
:UNIT:TEMPerature (on page 9-213)

[:SENSe[1]]:TEMPerature:TCouple:TYPE

This command indicates the thermocouple type.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle Measure configuration list	Save settings Measure configuration list	K

Usage

```
[:SENSe[1]]:TEMPerature:TCouple:TYPE <type>
[:SENSe[1]]:TEMPerature:TCouple:TYPE <type>, <clist>
[:SENSe[1]]:TEMPerature:TCouple:TYPE?
[:SENSe[1]]:TEMPerature:TCouple:TYPE? <clist>
```

`<type>`	B, E, J, K, N, R, S, or T
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command is only applicable when the transducer type is set to thermocouple.

Example

`FUNC "TEMP"`	Set the transducer type to thermocouple.
`TEMP:TRAN TC`	Set the thermocouple type to K.
`TEMP:TC:TYPE K`	Set the units to Celsius.
`TEMP:UNIT CELS`	Set the simulated reference temperature to 30.
`TEMP:TC:RJUN:SIM 30`	

Also see

"Temperature measurements" in the *Model 2750 User's Manual*
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:SIMulated (on page 9-144)
[:SENSe[1]]:TEMPerature:TCouple:RJUNction:RSELect (on page 9-143)

[:SENSe[1]]:TEMPerature:THERmistor[:TYPE]

This command describes the type of thermistor.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle Measure configuration list	Save settings Measure configuration list	5000

Usage

```
[:SENSe[1]]:TEMPerature:THERmistor[:TYPE] <NRf>
[:SENSe[1]]:TEMPerature:THERmistor[:TYPE] <NRf>, <clist>
[:SENSe[1]]:TEMPerature:THERmistor[:TYPE]?
[:SENSe[1]]:TEMPerature:THERmistor[:TYPE]? <clist>
```

`<n>`	The thermistor type in ohms: `1950` to `10050`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command is only applicable when the transducer type is set to thermistor.

Example

`FUNC "TEMP"`
`TEMP:TRAN THER`
`TEMP:THER 2252`
	Set measurement function to temperature.
	Set the transducer type to thermistor.
	Set the thermistor type to 2252.

Also see

"Temperature measurements" in the *Model 2750 User's Manual*

[:SENSe[1]]:TEMPerature:TRANsducer (on page 9-147)

[:SENSe[1]]:TEMPerature:TRANsducer

This command sets the transducer type.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	TCouple

Usage

```
[:SENSe[1]]:TEMPerature:TRANsducer <name>
[:SENSe[1]]:TEMPerature:TRANsducer <name>, <clist>
[:SENSe[1]]:TEMPerature:TRANsducer?
[:SENSe[1]]:TEMPerature:TRANsducer? <clist>
```

`<name>`	The thermocouple type: ▪ Thermocouple: `TCouple` ▪ RTD: `FRTD` ▪ Thermistor: `THERmistor`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Example

`:FUNC "TEMP"`
`:TEMP:TRANsducer FRTD`
`:TEMP:FRTD:TYPE PT3916`
	Set the measure function to temperature.
	Set the transducer type to 4-wire RTD.
	Set the RTD type to PT3916.

Also see

None

[:SENSe[1]]:VOLTage:AC:APERture

This command sets the aperture for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	60 Hz: 16.67 ms 50 Hz: 20 ms

Usage

```
[:SENSe[1]]:VOLTage:AC:APERture <NRf>
[:SENSe[1]]:VOLTage:AC:APERture <NRf>, <clist>
[:SENSe[1]]:VOLTage:AC:APERture?
[:SENSe[1]]:VOLTage:AC:APERture? <clist>
```

`<NRf>`	Integration rate in seconds: ▪ 60 Hz: 1.67e−4 to 1 ▪ 50 Hz: 2e−4 to 1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command is only valid if bandwidth is set to 300 (300 Hz to 300 kHz).

SCPI compliant.

Also see

[:SENSe[1]]:VOLTage:AC:NPLCycles (on page 9-152)

[:SENSe[1]]:VOLTage:AC:AVERage:COUNt

This command selects the filter count for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:VOLTage:AC:AVERage:COUNt <n>
[:SENSe[1]]:VOLTage:AC:AVERage:COUNt <n>, <clist>
[:SENSe[1]]:VOLTage:AC:AVERage:COUNt?
[:SENSe[1]]:VOLTage:AC:AVERage:COUNt? <clist>
```

`<n>`	The filter count: `1` to `100`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The filter count is the number of readings that are acquired and stored in the filter buffer for the averaging calculation. The larger the filter count, the more filtering that is performed.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:VOLTage:AC:AVERage:TCONtrol

This command selects the filter mode for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Reset: REPeat System preset: MOVing

Usage

```
[:SENSe[1]]:VOLTage:AC:AVERage:TCONtrol <name>
[:SENSe[1]]:VOLTage:AC:AVERage:TCONtrol?
```

`<name>`	Repeating filter mode: `REPeat` Moving filter mode: `MOVing`

Details

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:VOLTage:AC:AVERage:WINDow

This command specifies a filter window to control filter threshold.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.1

Usage

```
[:SENSe[1]]:VOLTage:AC:AVERage:WINDow <NRf>
[:SENSe[1]]:VOLTage:AC:AVERage:WINDow?
```

`<NRf>`	The filter window (in percent of range): `0` to `10`

Details

The digital filter uses a window to control the filter threshold. While the input signal remains in the selected window, A/D conversions continue to be placed in the stack. If the signal changes to a value outside the window, the filter resets, and the filter starts processing again, starting with a new initial conversion value from the A/D converter.

Setting 0 sets the filter window to NONE.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:VOLTage:AC:AVERage[:STATe]

This command enables or disables the digital averaging filter for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	*RST: OFF SYSTem:PRESet: ON

Usage

```
[:SENSe[1]]:VOLTage:AC:AVERage[:STATe] <b>
[:SENSe[1]]:VOLTage:AC:AVERage[:STATe] <b>, <clist>
[:SENSe[1]]:VOLTage:AC:AVERage[:STATe]?
[:SENSe[1]]:VOLTage:AC:AVERage[:STATe]? <clist>
```

`<b>`	Disable the digital filter: `OFF` or `0` Enable the digital filter: `ON` or `1`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When enabled, readings are filtered using the filter configuration.

If the state is set to `ON`, averaging type is set to `NONE`, and the median state is set to `OFF`, a settings conflict error occurs.

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:VOLTage:AC:DETector:BANDwidth

This command selects the detector bandwidth for AC current and AC voltage measurements.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	30

Usage

```
[:SENSe[1]]:VOLTage:AC:DETector:BANDwidth <NRf>
[:SENSe[1]]:VOLTage:AC:DETector:BANDwidth <NRf>, <clist>
[:SENSe[1]]:VOLTage:AC:DETector:BANDwidth?
[:SENSe[1]]:VOLTage:AC:DETector:BANDwidth? <clist>
```

`<NRf>`	The AC detector bandwidth in Hertz: 3 to 3e5, see Details
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

You can set the detector bandwidth to improve measurement accuracy. Select the bandwidth that contains the lowest frequency component of the input signal. For example, if the lowest frequency component of your input signal is 40 Hz, use a bandwidth setting of 30 Hz.

The instrument accepts parameter values up to 10e6, but it defaults to 3e5.

Example

```
FUNC "VOLT:AC"
VOLT:AC:DET:BAND 300
VOLT:AC:NPLC 5
```

This example sets the ACV rate to 5 PLC. To set the rate for an AC function, the bandwidth must be set to 300 Hz.

Also see

[:SENSe[1]]:VOLTage:AC:APERture (on page 9-147)
[:SENSe[1]]:VOLTage:AC:NPLCycles (on page 9-152)

[:SENSe[1]]:VOLTage:AC:DIGits

This command determines the number of digits that are displayed for measurements on the front panel.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	6

Usage

```
[:SENSe[1]]:VOLTage:AC:DIGits <n>
[:SENSe[1]]:VOLTage:AC:DIGits <n>, <clist>
[:SENSe[1]]:VOLTage:AC:DIGits?
[:SENSe[1]]:VOLTage:AC:DIGits? <clist>
```

`<n>`	Display digits: ■ 6½: 7 or `6.5` ■ 5½: 6 or `5.5` ■ 4½: 5 or `4.5` ■ 3½: 4 or `3.5`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command affects how the reading for a measurement is displayed on the front panel of the instrument. It does not affect the number of digits returned in a remote command reading. It also does not affect the accuracy or speed of measurements.

The display digits setting is saved with the function setting, so if you use another function, then return to the function for which you set display digits, the display digits setting you set previously is retained.

The change in digits occurs the next time a measurement is made.

Also see

None

[:SENSe[1]]:VOLTage:AC:NPLCycles

This command sets the time that the input signal is measured.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	5.0

Usage

```
[:SENSe[1]]:VOLTage:AC:NPLCycles <NRf>
[:SENSe[1]]:VOLTage:AC:NPLCycles <NRf>, <clist>
[:SENSe[1]]:VOLTage:AC:NPLCycles?
[:SENSe[1]]:VOLTage:AC:NPLCycles <clist>
```

`<NRf>`	Integration rate in line cycles: ▪ 60 Hz: 0.01 to 60 ▪ 50 Hz: 0.01 to 50
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command is only valid if bandwidth is set to 300 (300 Hz to 300 kHz).

SCPI compliant.

Also see

[:SENSe[1]]:VOLTage:AC:APERture (on page 9-147)

[:SENSe[1]]:VOLTage:AC:RANGe:AUTO

This command determines if the measurement range is set manually or automatically for the selected measure function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
[:SENSe[1]]:VOLTage:AC:RANGe:AUTO <b>
[:SENSe[1]]:VOLTage:AC:RANGe:AUTO <b>, <clist>
[:SENSe[1]]:VOLTage:AC:RANGe:AUTO?
[:SENSe[1]]:VOLTage:AC:RANGe:AUTO? <clist>
```

`<b>`	Set the measurement range manually: `OFF` Set the measurement range automatically: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When autorange is enabled, the instrument automatically goes to the most sensitive range to make the AC voltage measurement.

SCPI compliant.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:VOLTage:AC:RANGe[:UPPer] (on page 9-153)

[:SENSe[1]]:VOLTage:AC:RANGe[:UPPer]

This command sets the measurement range for selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	750

Usage

```
[:SENSe[1]]:VOLTage:AC:RANGe[:UPPer] <n>
[:SENSe[1]]:VOLTage:AC:RANGe[:UPPer] <n>, <clist>
[:SENSe[1]]:VOLTage:AC:RANGe[:UPPer]?
[:SENSe[1]]:VOLTage:AC:RANGe[:UPPer]? <clist>
```

`<n>`	Range in volts: 0 to 757.5
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:VOLTage:AC:RANGe:AUTO (on page 9-152)

[:SENSe[1]]:VOLTage:AC:REFerence

This command contains the relative offset value for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
[:SENSe[1]]:VOLTage:AC:REFerence <n>
[:SENSe[1]]:VOLTage:AC:REFerence <n>, <clist>
[:SENSe[1]]:VOLTage:AC:REFerence?
[:SENSe[1]]:VOLTage:AC:REFerence? <clist>
```

`<n>`	The reference value: −757.5 to 757.5
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

On the resistance function, the input signal is the computed resistance value seen at the input. The final result is the algebraic difference between the input resistance and the reference. You can establish a reference for the current component of a resistance measurement.

`[:SENSe[1]]:VOLTage:AC:REFerence` is SCPI compliant.

Also see

Relative offset (on page 4-12)
[:SENSe[1]]:VOLTage:AC:REFerence:ACQuire (on page 9-154)
[:SENSe[1]]:VOLTage:AC:REFerence:STATe (on page 9-155)

[:SENSe[1]]:VOLTage:AC:REFerence:ACQuire

This command acquires a measurement and stores it as the relative offset value.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:VOLTage:AC:REFerence:ACQuire
[:SENSe[1]]:VOLTage:AC:REFerence:ACQuire <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command uses the input signal on the selected channel as the relative offset value.

This command is functional only if the instrument is set to the selected measurement function. Sending this command when the instrument is set to any other function causes an error.

`[:SENSe[1]]:VOLTage:AC:REFerence` is set to the acquired value.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:VOLTage:AC:REFerence:STATe

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:VOLTage:AC:REFerence:STATe <b>
[:SENSe[1]]:VOLTage:AC:REFerence:STATe <b>, <clist>
[:SENSe[1]]:VOLTage:AC:REFerence:STATe?
[:SENSe[1]]:VOLTage:AC:REFerence:STATe? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

SCPI compliant.

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:VOLTage[:DC]:APERture

This command sets the aperture for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	60 Hz: 16.67 ms 50 Hz: 20 ms

Usage

```
[:SENSe[1]]:VOLTage[:DC]:APERture <NRf>
[:SENSe[1]]:VOLTage[:DC]:APERture <NRf>, <clist>
[:SENSe[1]]:SENSe[1]:VOLTage[:DC]:APERture?
[:SENSe[1]]:VOLTage[:DC]:APERture? <clist>
```

`<NRf>`	Integration rate in seconds: ■ 60 Hz: 1.67e−4 to 1 ■ 50 Hz: 2e−4 to 1
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:VOLTage[:DC]:NPLCycles (on page 9-160)

[:SENSe[1]]:VOLTage[:DC]:AVERage:COUNt

This command selects the filter count for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	10

Usage

```
[:SENSe[1]]:VOLTage[:DC]:AVERage:COUNt <n>
[:SENSe[1]]:VOLTage[:DC]:AVERage:COUNt <n>, <clist>
[:SENSe[1]]:VOLTage[:DC]:AVERage:COUNt?
[:SENSe[1]]:VOLTage[:DC]:AVERage:COUNt? <clist>
```

`<n>`	The filter count: `1` to `100`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

The filter count is the number of readings that are acquired and stored in the filter buffer for the averaging calculation. The larger the filter count, the more filtering that is performed.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:VOLTage[:DC]:AVERage:TCONtrol

This command selects the filter mode for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Reset: REPeat System preset: MOVing

Usage

```
[:SENSe[1]]:VOLTage[:DC]:AVERage:TCONtrol <name>
[:SENSe[1]]:VOLTage[:DC]:AVERage:TCONtrol?
```

`<name>`	Select the repeating filter mode: `REPeat` Select the moving filter mode: `MOVing`

Details

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Example

```
FUNC 'VOLT'                      ' Select DCV function.
VOLT:AVER:TCON REP               ' Select the repeating filter.
VOLT:AVER:COUN 20,(@101:115)     ' Set to filter 20 readings.
VOLT:AVER ON,(@101:115)          ' Enable filter.
```

This example configures channels 101 through 115 of the Model 7700 to use the repeat filter when they are scanned.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:VOLTage[:DC]:AVERage:WINDow

This command specifies a filter window to control filter threshold.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.1

Usage

```
[:SENSe[1]]:VOLTage[:DC]:AVERage:WINDow <NRf>
[:SENSe[1]]:VOLTage[:DC]:AVERage:WINDow?
```

`<NRf>`	The filter window (in percent of range): `0` to `10`

Details

The digital filter uses a window to control the filter threshold. While the input signal remains in the selected window, A/D conversions continue to be placed in the stack. If the signal changes to a value outside the window, the filter resets, and the filter starts processing again, starting with a new initial conversion value from the A/D converter.

Setting 0 sets the filter window to NONE.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:VOLTage[:DC]:AVERage[:STATe]

This command enables or disables the digital averaging filter for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	*RST: OFF SYSTem:PRESet: ON

Usage

```
[:SENSe[1]]:VOLTage[:DC]:AVERage[:STATe] <b>
[:SENSe[1]]:VOLTage[:DC]:AVERage[:STATe] <b>, <clist>
[:SENSe[1]]:VOLTage[:DC]:AVERage[:STATe]?
[:SENSe[1]]:VOLTage[:DC]:AVERage[:STATe]? <clist>
```

`<b>`	Disable the digital filter: `OFF` Enable the digital filter: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When enabled, readings are filtered using the filter configuration.

If the state is set to `ON`, averaging type is set to `NONE`, and the median state is set to `OFF`, a settings conflict error occurs.

Using the moving filter during a scan or test sequence generates a settings conflict error. To prevent this error, the filter mode is automatically set to `REPeat` during a scan or test sequence if the filter is `ON` for the active function.

Also see

Digital filters (on page 4-9)

[:SENSe[1]]:VOLTage[:DC]:DIGits

This command determines the number of digits that are displayed for measurements on the front panel.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	7

Usage

```
[:SENSe[1]]:VOLTage[:DC]:DIGits <n>
[:SENSe[1]]:VOLTage[:DC]:DIGits <n>, <clist>
[:SENSe[1]]:VOLTage[:DC]:DIGits?
[:SENSe[1]]:VOLTage[:DC]:DIGits? <clist>
```

`<n>`	Display digits: ■ 6½: 7 or `6.5` ■ 5½: 6 or `5.5` ■ 4½: 5 or `4.5` ■ 3½: 4 or `3.5`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command affects how the reading for a measurement is displayed on the front panel of the instrument. It does not affect the number of digits returned in a remote command reading. It also does not affect the accuracy or speed of measurements.

The display digits setting is saved with the function setting, so if you use another function, then return to the function for which you set display digits, the display digits setting you set previously is retained.

The change in digits occurs the next time a measurement is made.

Example

`VOLT:DIG 5.5`	Select 5½ digits for VDC.

Also see

None

[:SENSe[1]]:VOLTage[:DC]:IDIVider

This command enables or disables the 10 MΩ input divider.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:VOLTage[:DC]:IDIVider <b>
[:SENSe[1]]:VOLTage[:DC]:IDIVider?
```

`<b>`	Disable the input divider: `OFF` Enable the input divider: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

Normally, the input resistance for the 100 mVDC, 1 VDC, and 10 VDC ranges is >10GΩ and the input resistance of the 100 VDC and 1000 VDC ranges is 10 MΩ. You can also set the input resistance for the three lower DCV ranges to 10 MΩ by enabling the input divider.

With the input resistance lowered, a more stable 0 V reading is achieved with an open input. Also, some external devices (such as a high voltage probe) must be terminated to a 10 MΩ load.

The input divider cannot be enabled from the front panel.

Also see

None

[:SENSe[1]]:VOLTage[:DC]:NPLCycles

This command sets the time that the input signal is measured.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	5

Usage

```
[:SENSe[1]]:VOLTage[:DC]:NPLCycles <NRf>
[:SENSe[1]]:VOLTage[:DC]:NPLCycles <NRf>, <clist>
[:SENSe[1]]:VOLTage[:DC]:NPLCycles?
[:SENSe[1]]:VOLTage[:DC]:NPLCycles <clist>
```

`<NRf>`	Integration rate in line cycles: ▪ 60 Hz: 0.01 to 60 ▪ 50 Hz: 0.01 to 50
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Also see

[:SENSe[1]]:VOLTage[:DC]:APERture (on page 9-156)

[:SENSe[1]]:VOLTage[:DC]:RANGe:AUTO

This command determines if the measurement range is set manually or automatically for the selected measure function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
[:SENSe[1]]:VOLTage[:DC]:RANGe:AUTO <b>
[:SENSe[1]]:VOLTage[:DC]:RANGe:AUTO <b>, <clist>
[:SENSe[1]]:VOLTage[:DC]:RANGe:AUTO?
[:SENSe[1]]:VOLTage[:DC]:RANGe:AUTO? <clist>
```

`<b>`	Set the measurement range manually: `OFF` Set the measurement range automatically: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When autorange is enabled, the instrument automatically goes to the most sensitive dc voltage range to make the measurement.

SCPI compliant.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:VOLTage[:DC]:RANGe[:UPPer] (on page 9-162)

[:SENSe[1]]:VOLTage[:DC]:RANGe[:UPPer]

This command sets the measurement range for the dc voltage function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	1000

Usage

```
[:SENSe[1]]:VOLTage[:DC]:RANGe[:UPPer] <n>
[:SENSe[1]]:VOLTage[:DC]:RANGe[:UPPer] <n>, <clist>
[:SENSe[1]]:VOLTage[:DC]:RANGe[:UPPer]?
[:SENSe[1]]:VOLTage[:DC]:RANGe[:UPPer]? <clist>
```

`<n>`	Range: 0 to 1010
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

SCPI compliant.

Example 1

`VOLT:RANG 0.5`	Select 1 V range for DCV.

Example 2

`FUNC 'VOLT',(@101)` `VOLT:RANG 1.5,(@101)`	Set channel 101 for the DCV function. Set 101 for 10 V range.

Also see

Measurement range (on page 4-7)
[:SENSe[1]]:VOLTage[:DC]:RANGe:AUTO (on page 9-161)

[:SENSe[1]]:VOLTage[:DC]:REFerence

This command contains the relative offset value for the selected function.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
[:SENSe[1]]:VOLTage[:DC]:REFerence <n>
[:SENSe[1]]:VOLTage[:DC]:REFerence <clist>
[:SENSe[1]]:VOLTage[:DC]:REFerence?
[:SENSe[1]]:VOLTage[:DC]:REFerence? <clist>
```

`<n>`	The reference value: −1010 to 1010
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When making measurements, you may need to subtract an offset value from a measurement.

The relative offset feature subtracts a set value or a baseline reading from measurement readings. When you enable relative offset, all measurements are recorded as the difference between the actual measured value and the relative offset value. The formula to calculate the offset value is:

Displayed value = Actual measured value − Relative offset value

On the resistance function, the input signal is the computed resistance value seen at the input. The final result is the algebraic difference between the input resistance and the reference. You can establish a reference for the current component of a resistance measurement.

When `[:SENSe[1]]:VOLTage[:DC]:REFerence:ACQuire` is sent, this is changes to the acquired value.

SCPI compliant.

Example

This example configures channel 101 of the Model 7700 to enable relative offset and uses a 1 V relative offset value when it is scanned.

```
FUNC 'VOLT', (@101)        ' Select DCV for channel 101.
VOLT:REF 1, (@101)         ' Set 1 V relative offset value.
VOLT:REF:STAT ON, (@101)   ' Enable relative offset.
```

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:VOLTage[:DC]:REFerence:ACQuire

This command acquires a measurement and stores it as the relative offset value.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
[:SENSe[1]]:VOLTage[:DC]:REFerence:ACQuire
[:SENSe[1]]:VOLTage[:DC]:REFerence:ACQuire <clist>
```

`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

This command uses the input signal on the selected channel as the relative offset value.

This command is functional only if the instrument is set to the selected measurement function. Sending this command when any other function is selected causes an error.

`[:SENSe[1]]:VOLTage[:DC]:REFerence` is set to the acquired value.

`[:SENSe[1]]:VOLTage[:DC]:REFerence:ACQuire` is SCPI compliant.

Example

This example configures channel 101 of the Model 7700 to zero correct the DCV input when it is scanned.

```
FUNC 'VOLT', (@101)        ' Select DCV for channel 101.
ROUT:CLOS (@101)           ' Close channel 101.
VOLT:REF:ACQ (@101)        ' Use input to channel 101 as the relative offset
   value.
VOLT:REF:STAT ON, (@101)   ' Enable relative offset.
```

Also see

Relative offset (on page 4-12)

[:SENSe[1]]:VOLTage[:DC]:REFerence:STATe

This command enables or disables the application of a relative offset value to the measurement.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
[:SENSe[1]]:VOLTage[:DC]:REFerence:STATe <b>
[:SENSe[1]]:VOLTage[:DC]:REFerence:STATe <b>, <clist>
[:SENSe[1]]:VOLTage[:DC]:REFerence:STATe?
[:SENSe[1]]:VOLTage[:DC]:REFerence:STATe? <clist>
```

`<b>`	Disable the relative offset: `OFF` Enable the relative offset: `ON`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Details

When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value.

Each returned measured relative reading is the result of the following calculation:

Displayed reading = Actual measured reading - Relative offset value

SCPI compliant.

Example

This example zeroes the display for the DCV measurement function.

```
FUNC 'VOLT'         ' Select DCV.
VOLT:REF:ACQ        ' Use input level as the relative offset value for DCV.
VOLT:REF:STAT ON    ' Enable relative offset.
```

Also see

Relative offset (on page 4-12)

STATus subsystem

The STATus subsystem controls the status registers of the 2750. For additional information on the status model, see Status model (on page 11-1).

:STATus:MEASurement:CONDition?

This command reads condition registers.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:STATus:MEASurement:CONDition?
```

Also see

:STATus:MEASurement[:EVENt]? (on page 9-167)

:STATus:MEASurement:ENABle

This command sets or reads the contents of the Measurement Event Enable Register of the status model.

Type	Affected by	Where saved	Default value
Command and query	STATus:PRESet Power cycle	Not applicable	See **Details**

Usage

```
:STATus:MEASurement:ENABle <NRf>
:STATus:MEASurement:ENABle?
```

`<NRf>`	The status of the Measurement Status Register; see **Details**

Details

Power-up and the `:STATus:PRESet` command clear all bits of the registers. The `*CLS` command has no effect.

Example

```
STAT:MEAS:ENAB 512     ' Enable BFL (buffer full).
STAT:MEAS:COND?        ' Read Measurement Condition Register.
STAT:MEAS?             ' Read Measurement Event Register.
```

This example programs and reads the measurement register set. The 2750 has to be addressed to talk after a query command.

Also see

Measurement Event Status Register (on page 11-14)

:STATus:MEASurement[:EVENt]?

This command reads the Measurement Event Register of the status model.

Type	Affected by	Where saved	Default value
Query only	Power-up *CLS	Not applicable	Not applicable

Usage

```
:STATus:MEASurement[:EVENt]?
```

Details

Power-up and *CLS clear all bits of the register. STATus:PRESet has no effect.

Also see

None

:STATus:OPERation:CONDition?

This command reads condition registers.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:STATus:OPERation:CONDition?
```

Also see

:STATus:OPERation[:EVENt]? (on page 9-167)

:STATus:OPERation:ENABle

This command sets or reads the contents of the Operation Event Enable Register of the status model.

Type	Affected by	Where saved	Default value
Command and query	STATus:PRESet Power cycle	Not applicable	See **Details**

Usage

```
:STATus:OPERation:ENABle <NRf>
:STATus:OPERation:ENABle?
```

`<NRf>`	The status of the Operation Status Register; see **Details**

Details

To clear the operation event enable register, set <NRf> to 0. For example:

```
STAT:OPER:ENAB 0
```

Power-up and the `:STATus:PRESet` command clear all bits of the registers. The `*CLS` command has no effect.

Also see

Event enable registers (on page 11-18)

:STATus:OPERation[:EVENt]?

This command reads the Operation Event Register of the status model.

Type	Affected by	Where saved	Default value
Query only	Power-up *CLS	Not applicable	Not applicable

Usage

```
:STATus:OPERation[:EVENt]?
```

Details

Power-up and *CLS clear all bits of the register. STATus:PRESet has no effect.

Also see

None

:STATus:PRESet

This command resets all bits of the event registers to 0.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:STATus:PRESet
```

Details

This resets the following enable and event registers to 0:

- Operation Event Enable Register
- Measurement Event Enable Register
- Questionable Event Enable Register

The Standard Event Enable Register is not reset by `STATus:PRESet` or `*CLS`. To reset all bits of that enable register to 0, send the following command:

```
*ESE 0
```

`STATus:PRESet` has no effect on the Error Queue.

SCPI compliant.

Also see

*ESE (on page 10-4)

:STATus:QUEStionable:CONDition?

This command reads condition registers.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:STATus:QUEStionable:CONDition?
```

Also see

:STATus:OPERation[:EVENt]? (on page 9-167)

:STATus:QUEStionable:ENABle

This command sets or reads the contents of the Operation Event Enable Register of the status model.

Type	Affected by	Where saved	Default value
Command and query	STATus:PRESet Power cycle	Not applicable	See **Details**

Usage

```
:STATus:QUEStionable:ENABle <NRf>
:STATus:QUEStionable:ENABle?
```

`<NRf>`	The status of the Operation Status Register; see **Details**

Details

Power-up and the `:STATus:PRESet` command clear all bits of the registers. The `*CLS` command has no effect.

Also see

Questionable event register (on page 11-16)

:STATus:QUEStionable[:EVENt]?

This command reads the Questionable Event Register of the status model.

Type	Affected by	Where saved	Default value
Query only	Power-up *CLS	Not applicable	Not applicable

Usage

```
:STATus:QUEStionable[:EVENt]?
```

Details

Power-up and *CLS clear all bits of the register. STATus:PRESet has no effect.

Also see

None

:STATus:QUEue:CLEar

This function clears all messages from the error queue.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:STATus:QUEue:CLEar
```

Also see

Queues (on page 11-19)

:STATus:QUEue:DISable

This function specifies the messages that are not to be placed in the error queue.

Type	Affected by	Where saved	Default value
Command and query	Power cycle	Not applicable	Not applicable

Usage

```
:STATus:QUEue:DISable <list>
:STATus:QUEue:DISable?
```

`<list>`	The messages to disable; see **Details**

Details

The query reads the disabled messages.

On power-up, all error messages are enabled and go into the error queue as they occur. This command specifies the messages you want to disable. Disabled messages are prevented from entering the queue. To disable all messages from entering the error queue, send the command:

```
:STAT:QUE:DIS ()
```

NOTE

Status messages are not enabled and do not go into the queue.

Messages are specified by number. Refer to “Status and error messages” in the *Model 2750 User's Manual* for message numbers.

Power-up clears the list of messages. `*CLS` and `:STATus:PRESet` have no effect.

Example

`:STATus:QUEue:DISable -110`	Disable single message.
`:STATus:QUEue:DISable -110, -140, -222`	Disable multiple messages (separated by commas).
`:STATus:QUEue:DISable -110:-222`	Disable a range of messages (start and end separated by a colon).
`:STATus:QUEue:DISable -110:-222, -230`	Disable a range (separated by a colon) and a single entry (separated by a comma).

Also see

Queues (on page 11-19)
:STATus:QUEue:ENABle (on page 9-172)

:STATus:QUEue:ENABle

This function returns the most recent error queue message.

Type	Affected by	Where saved	Default value
Command and query	Power cycle	Not applicable	Not applicable

Usage

```
:STATus:QUEue:ENABle <list>
:STATus:QUEue:ENABle?
```

`<list>`	A specified list of messages to enable for the error queue; see **Details**

Details

On power-up, all error messages are enabled and go into the error queue as they occur. This command specifies which messages you want to enable. Messages that are not specified are disabled and prevented from entering the queue.

NOTE

Status messages are not enabled and do not go into the queue.

When this command is sent, all messages are disabled, then the specified messages are enabled. The query contains all the enabled messages. To prevent all messages from entering the error queue, send the command with the null list parameter:

```
:STAT:QUE:ENAB ()
```

Messages are specified by numbers. A list of messages is provided in “Status and error messages” in the *Model 2750 User's Manual*.

The list of messages is cleared on power up. `*CLS` and `:STATus:PRESet` have no effect.

Example

`:STATus:QUEue:ENABle -110`	Enable single message.
`:STATus:QUEue:ENABle -110, -140, -222`	Enable multiple messages (separated by commas).
`:STATus:QUEue:ENABle -110:-222`	Enable a range of messages (start and end separated by a colon).
`:STATus:QUEue:ENABle -110:-222, -230`	Enable a range (separated by a colon) and a single entry (separated by a comma).

Also see

Queues (on page 11-19)
:STATus:QUEue:DISable (on page 9-171)

:STATus:QUEue[:NEXT]?

This function returns the oldest error queue message from the Event Queue and removes it.

Type	Affected by	Where saved	Default value
Command and query	Power cycle	Not applicable	See **Details**

Usage

```
:STATus:QUEue[:NEXT]?
```

Details

As error and status messages occur, they are placed in the error queue. This command reads those messages. The error queue is a first-in, first-out (FIFO) register that can hold up to 10 messages. Each time you read the queue, the oldest message is read, and that message is then removed from the queue.

If the queue is full, the `350, Queue Overflow` message occupies the last memory location in the register. When the error queue is empty, the message `0, No error` is placed in the error queue.

The messages in the queue are preceded by a number. Negative (–) numbers indicate SCPI-defined messages and positive (+) numbers indicate messages defined by Keithley. Refer to “Status and error messages” in the *Model 2750 User's Manual* for the list of messages.

Power-up and `*CLS` clears the list of messages. `:STATus:PRESet` has no effect.

SCPI compliant.

NOTE

The `:STATus:QUEue[:NEXT]?` query command performs the same function as the `:SYSTem:ERRor?` query command.

Example

```
STAT:QUE?
Returns information on the oldest error in the event log.
```

Also see

:SYSTem:ERRor? (on page 9-188)

SYSTem subsystem

This subsystem contains commands that affect the overall operation of the instrument, such as autozero and beeper. It also includes commands to retrieve card information.

:SYSTem:AZERo:STATe

This command enables and disables autozero.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
:SYSTem:AZERo:STATe <b>
:SYSTem:AZERo:STATe?
```

<b>	Disable autozero: OFF Enable autozero: ON

Details

After enabling autozero, you can update the internal reference points immediately by setting the integration rate to 0.01 PLC and then back to the appropriate setting.

Example

`:SYST:AZER OFF`	Disables autozero.

Also see

[:SENSe[1]]:CURRent:AC:NPLCycles (on page 9-86)
[:SENSe[1]]:FRESistance:NPLCycles (on page 9-110)
[:SENSe[1]]:RESistance:NPLCycles (on page 9-127)
[:SENSe[1]]:TEMPerature:NPLCycles (on page 9-140)
[:SENSe[1]]:VOLTage:AC:NPLCycles (on page 9-152)
[:SENSe[1]]:VOLTage[:DC]:NPLCycles (on page 9-160)

:SYSTem:BEEPer[:STATe]

This command allows you to turn the beeper on or off.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
:SYSTem:BEEPer[:STATe] <b>
:SYSTem:BEEPer[:STATe]?
```

`<b>`	Disable the beeper: `ON` Enable the beeper: `OFF`

Details

You can disable the beeper for limits and continuity tests. However, when limits or CONT is selected again, the beeper is automatically enabled.

SCPI compliant.

Also see

None

:SYSTem:CARD:ACHannel:END?

This command indicates the last channel in the specified slot that supports amps measurements.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:ACHannel:END? <NRf>
```

`<NRf>`	The slot number: `1` to `5`

Details

The channels that support amps measurements are grouped, so you can use the start and end channel numbers to identify a group of channels that supports amps measurements. If the card supports amps measurements, the returned value is the number of the end channel. If only one channel on the card supports amps measurements, the end channel matches the start channel. If the channel does not support amps measurements, the return is `0`.

Example

`syst:card:ach? 1`
`syst:card:ach:end? 1`
	These commands return the amps measurement start channel and end channel. For the 7700, the returns are:
	`7700; "21"`
	`7700; "22"`

Also see

:SYSTem:CARD:ACHannel[:STARt]? (on page 9-176)

:SYSTem:CARD:ACHannel[:STARt]?

This command indicates the first channel in the specified slot that supports current measurements.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

`:SYSTem:CARD:ACHannel[:STARt]? <NRf>`

`<NRf>`	The slot number: 1 to 5

Details

The channels that support current measurements are grouped, so you can use the start and end channel numbers to identify the group of channels that supports current measurements. If the card supports current measurements, the returned value is the number of the start channel. If only one channel on the card supports current measurements, the start channel matches the end channel. If the channel does not support current measurements, the return is `0`.

Example

`syst:card:ach? 1` `syst:card:ach:end? 1`	These commands return the amps measurement start channel and end channel.

Also see

:SYSTem:CARD:ACHannel:END? (on page 9-175)

:SYSTem:CARD:AOUTput:END?

This command indicates the last channel in the specified slot that supports analog outputs.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

`:SYSTem:CARD:AOUTput:END? <NRf>`

`<NRf>`	The slot number

Details

The channels that support analog outputs are grouped, so you can use the start and end channel numbers to identify the group of channels. If the card supports analog outputs, the returned value is the number of the last channel. If only one channel on the card supports analog outputs, the start channel matches the end channel. If the channel does not support analog outputs, the return is `0`.

Example

`syst:card:aout? 1` `syst:card:aout:end? 1`	If the 2750 contains a 7706 switching module in slot 1, these commands return 23 for the start channel and 24 for the end channel.

Also see

:SYSTem:CARD:AOUTput[:STARt]? (on page 9-177)

:SYSTem:CARD:AOUTput[:STARt]?

This command indicates the first channel in the specified slot that supports analog outputs.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:AOUTput[:STARt]? <NRf>
```

`<NRf>`	The slot number

Details

The channels that support analog outputs are grouped, so you can use the start and end channel numbers to identify the group of channels. If the card supports analog outputs, the returned value is the number of the start channel. If only one channel on the card supports analog outputs, the start channel matches the end channel. If the channel does not support analog outputs, the return is `0`.

Example

`syst:card:aout? 1` `syst:card:aout:end? 1`	If the 2750 contains a 7706 switching module in slot 1, these commands return 23 for the start channel and 24 for the end channel.

Also see

:SYSTem:CARD:AOUTput:END? (on page 9-176)

:SYSTem:CARD:BANKs?

This command returns the number of banks of a single no-open or single with-open card.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:BANKs? <NRf>
```

`<NRf>`	The slot number: 1 to 5

Details

If the card is not a single no-open type or single with-open type, error −221, settings conflict, is returned.

Also see

None

:SYSTem:CARD:CSOhms?

This command indicates whether a card in the specified slot supports commonside channels for 4-wire resistance measurements.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:CSOhms? <NRf>
```

`<NRf>`	The slot number

Details

If commonside 4-wire resistance channels are supported, the returned value is `1`.

If commonside 4-wire resistance channels are not supported, the return value is `0`.

Example

`SYST:CARD:CSO? 1`	Query to verify that slot 1 supports commonside 4-wire resistance channels.

Also see

None

:SYSTem:CARD:DINPut:END?

This command indicates the last channel in the specified slot that supports digital inputs.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:DINPut:END? <NRf>
```

`<NRf>`	The slot number

Details

The channels that support digital inputs are grouped, so you can use the start and end channel numbers to identify the group of channels. If the card supports digital inputs, the returned value is the number of the last channel. If only one channel on the card supports digital inputs, the start channel matches the end channel. If the channel does not support digital inputs, the return is `0`.

Example

`syst:card:dinp? 1` `syst:card:dinp:end? 1`	Returns the start and end channels that support digital inputs.

Also see

:SYSTem:CARD:DINPut[STARt]? (on page 9-179)

:SYSTem:CARD:DINPut[:STARt]?

This command indicates the first channel in the specified slot that supports digital inputs.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:DINPut[:STARt]? <NRf>
```

`<NRf>`	The slot number

Details

The channels that support digital inputs are grouped, so you can use the start and end channel numbers to identify the group of channels. If the card supports digital inputs, the returned value is the number of the start channel. If only one channel on the card supports digital inputs, the start channel matches the end channel. If the channel does not support digital inputs, the return is `0`.

Example

`syst:card:dinp? 1` `syst:card:dinp:end? 1`	Returns the start and end channels that support digital inputs.

Also see

:SYSTem:CARD:DINPut:END? (on page 9-179)

:SYSTem:CARD:DOUTput:END?

This command indicates the last channel in the specified slot that supports digital outputs.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:DOUTput:END? <NRf>
```

`<NRf>`	The slot number

Details

The channels that support digital outputs are grouped, so you can use the start and end channel numbers to identify the group of channels. If the card supports digital outputs, the returned value is the number of the end channel. If only one channel on the card supports digital outputs, the start channel matches the end channel. If the channel does not support digital outputs, the return is `0`.

Example

`syst:card:dout? 1` `syst:card:dout:end? 1`	Returns the start and end channels that support digital outputs.

Also see

:SYSTem:CARD:DOUTput[:STARt]? (on page 9-180)

:SYSTem:CARD:DOUTput[:STARt]?

This command indicates the first channel in the specified slot that supports digital outputs.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:DOUTput[:STARt]? <NRf>
```

`<NRf>`	The slot number

Details

The channels that support digital outputs are grouped, so you can use the start and end channel numbers to identify the group of channels. If the card supports digital outputs, the returned value is the number of the start channel. If only one channel on the card supports digital outputs, the start channel matches the end channel. If the channel does not support digital outputs, the return is `0`.

Example

`syst:card:dout? 1` `syst:card:dout:end? 1`	Returns the start and end channels that support digital outputs.

Also see

:SYSTem:CARD:DOUTput:END? (on page 9-176)

:SYSTem:CARD:ISOLated?

This command indicates if isolated channels are supported.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:ISOLated? <NRf>
```

`<NRf>`	The slot number: `1` to `5`

Details

If isolated channels are supported, the return is `1`.

If not, the return is `0`.

Also see

None

:SYSTem:CARD:MUX?

This command indicates if multiplexer channels are supported.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:MUX? <NRf>
```

`<NRf>`	The slot number: `1` to `5`

Details

If multiplexer channels are supported, the return is `1`.

If not, the return is `0`.

Also see

None

:SYSTem:CARD:SNOpen?

This command indicates if an installed card is a single no-open type (for example, 7711).

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:SNOpen? <NRf>
```

`<NRf>`	The slot number: `1` to `5`

Details

If the card is a single no-open type, the return is `1`.

If not, the return is `0`.

Also see

None

:SYSTem:CARD:SNUMber?

This command returns the serial number of a card.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:SNUMber? <NRf>
```

`<NRf>`	The slot number

Details

If a pseudocard is installed in the slot, the message `????????` is returned when querying the serial number.

Example

`syst:card:snum? 1`	Returns the serial number of the card in slot 1.

Also see

None

:SYSTem:CARD:SWOpen?

This command indicates if an installed card is a single with-open type (for example, 7711).

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:SWOpen? <NRf>
```

`<NRf>`	The slot number: `1` to `5`

Details

If the card is a single with-open type, the return is `1`.

If not, the return is `0`.

Also see

None

:SYSTem:CARD:SWRevision?

This command returns the firmware revision of a card.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:SWRevision? <NRf>
```

`<NRf>`	The slot number

Details

If a pseudocard is installed in the slot, the message ???????? is returned when querying the firmware revision.

Example

`syst:card:swr? 1`	Returns the firmware version of the card in slot 1.

Also see

None

:SYSTem:CARD:TCHannel?

This command returns the totalizer channel.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:TCHannel? <NRf>
```

`<NRf>`	The slot number: `1` to `5`

Also see

None

:SYSTem:CARD:TCOMpensated?

This command indicates the card includes built-in temperature sensors for thermocouple cold junction. supported.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:TCOMpensated? <NRf>
```

`<NRf>`	The slot number: `1` to `5`

Details

If the card includes built-in temperature sensors, the return is `1`.

If not, the return is `0`.

Also see

None

:SYSTem:CARD:VCHannel:END?

This command indicates the last channel in the specified slot that supports voltage or 2-wire measurements.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:VCHannel:END? <NRf>
```

`<NRf>`	The slot number: `1` to `5`

Details

The channels that support voltage or 2-wire measurements are grouped, so you can use the start and end channel numbers to identify the group. If the card supports voltage or 2-wire measurements, the returned value is the number of the last channel. If only one channel on the card supports voltage or 2-wire measurements, the start channel matches the end channel. If the channel does not support voltage or 2-wire measurements, the return is `0`.

Example

`syst:card:vch? 1`
`syst:card:vch:end? 1`
	These commands return the voltage or 2-wire measurement start channel and end channel. For the 7700, the returns are:
	`7700, "1"`
	`7700, "20"`

Also see

:SYSTem:CARD:VCHannel[:STARt]? (on page 9-199)

:SYSTem:CARD:VCHannel[:STARt]?

This command indicates the first channel in the specified slot that supports voltage or 2-wire measurements

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:VCHannel[:STARt]? <NRf>
```

`<NRf>`	The slot number: 1 to 5

Details

The channels that support voltage or 2-wire measurements are grouped, so you can use the start and end channel numbers to identify the group. If the card supports voltage or 2-wire measurements, the returned value is the number of the start channel. If only one channel on the card supports voltage or 2-wire measurements, the start channel matches the end channel. If the channel does not support voltage or 2-wire measurements, the return is `0`.

Example

`syst:card:vch? 1` `syst:card:vch:end? 1`	These commands return the voltage or 2-wire measurement start channel and end channel.

Also see

:SYSTem:CARD:VCHannel:END? (on page 9-185)

:SYSTem:CARD:VMAX?

This command returns the maximum allowable voltage.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:VMAX? <NRf>
```

`<NRf>`	The slot number: 1 to 5

Also see

None

:SYSTem:CLEar

This command clears messages from the error queue.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CLEar
```

Details

SCPI compliant.

Also see

:SYSTem:ERRor? (on page 9-188)

:SYSTem:DATE

This command sets the calendar of the instrument.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Nonvolatile memory	Not applicable

Usage

```
:SYSTem:DATE <year>, <month>, <day>
:SYSTem:DATE?
```

`<year>`	Year: 1999 to 2099
`<month>`	Month: 1 to 12
`<day>`	Day: 1 to 31

Details

The query command returns the present date in the format:

`<year>,<month>,<day>`

Example

`:SYST:DATE 2023, 5, 12`	Set the system date to May 12, 2023.

Also see

:SYSTem:TIME (on page 9-197)

:SYSTem:ERRor?

This command returns the oldest unread error message from the event log and removes it from the log.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:ERRor?
```

Details

The error queue is cleared at power-up and when `*CLS` is sent. `:SYSTem:PRESet` and `:STATus:PRESet` have no effect on the error queue.

Example

`SYST:ERR?`	
Returns information on the oldest error in the event log.	

Also see

Queues (on page 11-19)
:STATus:QUEue[:NEXT]? (on page 9-173)

:SYSTem:FRSWitch?

This command returns the state of the INPUTS switch.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:FRSWitch?
```

Details

If the INPUTS switch is set to REAR, `0` is returned.

If the INPUTS switch is set to FRONT, `1` is returned.

Example

`SYSTem:FRSwitch?`	Query the front-panel INPUTS switch.

Also see

None

:SYSTem:KCLick

This command allows you to enable or disable the key click sound.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle	Save settings	ON

Usage

```
:SYSTem:KCLick <state>
:SYSTem:KCLick?
```

`<state>`	Disable the key click sound: `OFF` or `0` Enable the key click sound: `ON` or `1`

Details

This command enables or disables the key click sound. When enabled, when you press a key on the front-panel, a sound is output.

NOTE

Keyclick ON is the default for FACTORY, *RST, and SYSTem:PRESet.

Example

`:SYSTem:KCLick OFF`	Disable the key clicks.

Also see

None

:SYSTem:KEY

This command simulates a front-panel key press using remote commands.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	Not applicable

Usage

```
:SYSTem:KEY <NRf>
:SYSTem:KEY?
```

`<NRf>`	The number that corresponds to the key press; see **Details**

Details

For example, to select DCV (dc volts) you can send the following command to simulate pressing the DCV key:

```
:syst:key 2
```

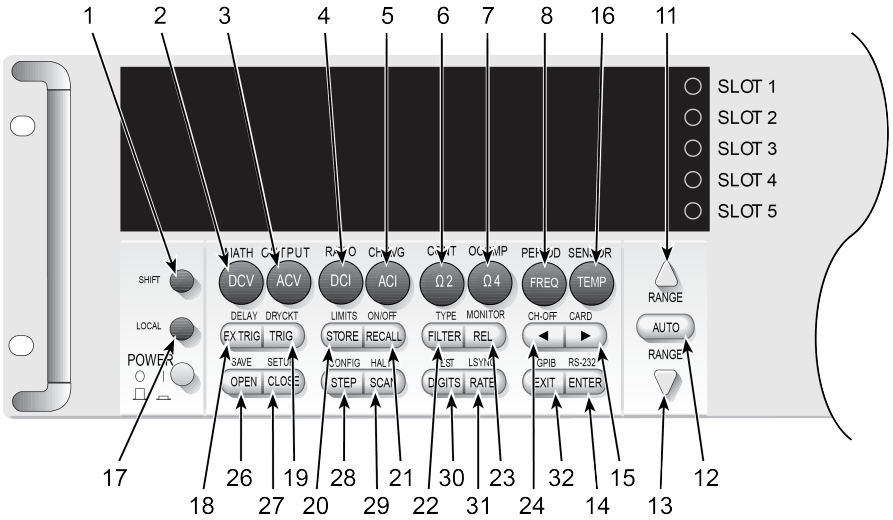
The queue for `:SYSTem:KEY?` holds one key-press value. When `:SYSTem:KEY?` is sent over the bus and the 2750 is addressed to talk, the key-press code number for the last key pressed (either physically or with `:SYSTem:KEY?`) is sent to the computer.

The value of `<NRf>` that corresponds to each key is listed in the following table. The `<NRf>` values are also shown in the figure that follows the table.

<NRf>	Key
1	SHIFT
2	DCV
3	ACV
4	DCI
5	ACI
6	Ω2
7	Ω4
8	FREQ
11	Range ▲
12	AUTO
13	Range ▼
14	ENTER
15	►
16	TEMP
17	LOCAL

<NRf>	Key
18	EX TRIG
19	TRIG
20	STORE
21	RECALL
22	FILTER
23	REL
24	◄
26	OPEN
27	CLOSE
28	STEP
29	SCAN
30	DIGITS
31	RATE
32	EXIT

Figure 63: Key-press codes



Also see

None

:SYSTem:LFRequency?

This query returns the line frequency that was automatically detected when the instrument was turned on.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:LFRequency?
```

Details

The instrument automatically detects the power line frequency when the instrument is powered on. Power line frequency can be 50 Hz or 60 Hz.

Example

`:SYST:LFR?`	Return the line frequency.

Also see

None

:SYSTem:LOCal

This command takes 2750 out of remote control.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:LOCal
```

Details

This command takes the 2750 out of the remote state and enables the operation of front-panel keys.

This command can only be sent over the RS-232 interface.

Also see

:SYSTem:RWLock (on page 9-197)

:SYSTem:LSYNc:STATe

This command enables or disables line synchronization.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	OFF

Usage

```
:SYSTem:LSYNc:STATe <b>
:SYSTem:LSYNc:STATe?
```

<b>	Disable line cycle synchronization: OFF Enable line cycle synchronization: ON

Details

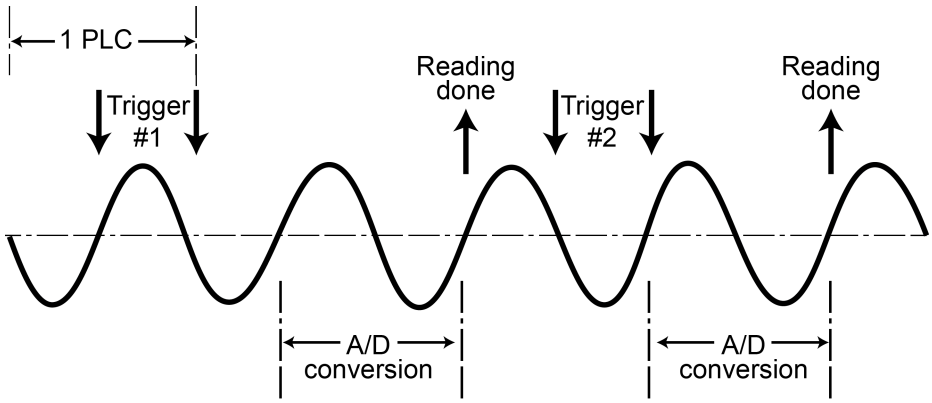
Synchronizing A/D conversions with the frequency of the power line increases common mode and normal mode noise rejection. When line cycle synchronization is enabled, the measurement is initiated at the first positive-going zero crossing of the power line cycle after the trigger.

The following figure shows a measurement process that consists of two A/D conversions. If the trigger occurs during the positive cycle of the power line (Trigger #1), the A/D conversion starts with the positive-going zero crossing of the power line cycle. If the next trigger (Trigger #2) occurs during the negative cycle, then the measurement process also starts with the positive-going zero crossing.

NOTE

Line synchronization is not available for the AC functions (ACV, ACI, FREQ, or PERIOD) and for integration rates <1 PLC, regardless of the LSYNC setting.

Figure 64: Line cycle synchronization



Also see

None

:SYSTem:PCARd

This command specifies a pseudocard to implement.

Type	Affected by	Where saved	Default value
Command only	Recall settings Instrument reset Power cycle	Save settings	0

Usage

```
:SYSTem:PCARd<slot> C<cardNumber>
```

`<slot>`	Slot number: 1 to 5
`<cardNumber>`	The card number: ■ `0`: No pseudocard ■ `7700`: 20-channel Differential Multiplexer Module ■ `7701`: 32-channel Differential Multiplexer Module ■ `7702`: 40-channel Differential Multiplexer Module with Screw Terminals ■ `7703`: 32-channel High Speed Differential Multiplexer Module ■ `7705`: 40-channel Single-pole Control Module ■ `7706`: 20-channel All-in-One I/O Module ■ `7707`: 32-channel Digital I/O Module with 10-channel Differential Multiplexer ■ `7708`: 40-channel Differential Multiplexer Module with Automatic CJC and Screw Terminals ■ `7709`: 6 × 8 Matrix Module

	■ 7710: 20-channel Solid-state Differential Multiplexer with Automatic CJC ■ 7711: 2 GHz 50 Ohm RF Module ■ 7712: 3.5 GHz 50 Ohm RF Module

Details

Pseudocards allow you to configure your system without having an actual switching module installed in your system. You can perform open, close, and scan operations and configure your system with pseudocards.

This command is only applicable to a slot that does not have a switching module or pseudocard installed. If a pseudocard is presently assigned to the slot, you must set the slot to no pseudocard before assigning the new pseudocard.

After assigning a pseudocard, you can use valid commands for the switching module for that slot.

Changing the pseudocard assignment from a pseudocard to no pseudocard invalidates scan lists that include that slot.

Example

`syst:pcar2 C0` `syst:pcar2 C7700`	Remove any existing pseudocards from slot 2. Install a pseudocard 7700 in slot 2.

Also see

Pseudocards

:SYSTem:POSetup

This command selects the defaults when you power on the instrument.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Nonvolatile memory	RST

Usage

```
:SYSTem:POSetup <name>
:SYSTem:POSetup?
```

`<name>`	Which setup to restore when you power on the instrument: ■ `*RST` defaults: `RST` ■ `:SYSTem:PRESet` defaults: `PRESet` ■ Stored setup: `SAV0`, `SAV1`, or `SAV2`

Details

When you select `*RST`, the instrument restores settings to their default values when the instrument is powered on.

When you select `PRESet`, the instrument restores settings to the `:SYSTem:PRESet` default conditions when the instrument is powered on.

When you select a `*SAV` setting, the settings in the saved setup are applied when the instrument is powered on. The settings are saved using the `*SAV` command.

Default settings are listed in the SCPI command descriptions.

Example

```
*SAV 2          ' Save the present setup in memory location 2.
SYST:POS SAV2   ' Specify SAV2 setup as the power-on setup.
*RST            ' Return 2750 to RST defaults.
*RCL 2          ' Return 2750 to setup stored in memory location 2.
```

Also see

*RST (on page 10-10)
*SAV (on page 10-10)
:SYSTem:PRESet (on page 9-195)

:SYSTem:PRESet

This command returns the instrument to states optimized for front-panel operation.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:PRESet
```

Details

The default values that are restored are listed in SCPI reference tables (on page 9-6).

For RS-232 operation and in some cases, GPIB operation, `*OPC` or `*OPC?` should be used with `SYST:PRES`, which is a slow responding command. Details on `*OPC` and `*OPC?` are provided in Common commands (on page 10-1).

SCPI compliant.

Example

`:SYST:PRES`	Set the instrument to states optimized for front-panel operation.

Also see

*RST (on page 10-10)

:SYSTem:REMote

This command places the 2750 in remote control.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:REMote
```

Details

In remote, the front-panel keys are locked out if local lockout is enabled.

This command can only be sent over the RS-232 interface.

Also see

:SYSTem:RWLock (on page 9-197)

:SYSTem:RNUMber:RESet

This command resets the reading number to zero.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:RNUMber:RESet
```

Details

The next reading will be 1.

Also see

None

:SYSTem:RWLock

This command locks out the front-panel controls.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:RWLock
```

Details

When this command is sent, all front-panel keys are disabled. The instrument can only be controlled using the remote interface.

This command can only be sent over the RS-232 interface.

Example

`:SYST:RWLock`	Disables the front-panel keys.

Also see

:SYSTem:LOCal (on page 9-192)

:SYSTem:TIME

This command sets the time for the real-time clock.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:TIME <hr>, <min>, <sec>
:SYSTem:TIME?
```

`<hr>`	Specify hour (24-hour format): `0` to `23`
`<min>`	Specify minute: `0` to `59`
`<sec>`	Specify second (rounded to 1/100 seconds): `0.00` to `59.9`

Details

The time must be set using the 24-hour format (for example, hour 13 is 1 pm). Setting an invalid time results in an error and the previous time is retained.

`SYST:TIME?` returns the time to the nearest hundredth of a second.

SCPI compliant.

Example

`:SYST:TIME 13, 23, 36` `:SYST:TIME 3, 25, 28.5`	Set time to 1:23:36 pm. Set time to 3:25:28.5 am.

Also see

None

:SYSTem:TSTamp:RELative:RESet

This command resets the relative timestamp to 0 seconds.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:TSTamp:RELative:RESet
```

Also see

:SYSTem:TSTamp:TYPE (on page 9-198)

:SYSTem:TSTamp:TYPE

This command selects the timestamp type.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	REL

Usage

```
:SYSTem:TSTamp:TYPE <name>
:SYSTem:TSTamp:TYPE?
```

`<name>`	Select relative-time timestamp: `RELative` Select real-time timestamp: `RTClock`

Details

`RELative` selects the relative-time timestamp.

`RTClock` selects the real-time clock timestamp.

Changing the timestamp clears the buffer if a storage is in process. If no storage is in process, changing the timestamp does not clear the buffer.

`SYSTem:TSTamp:TYPE?` queries the timestamp that is used for the next storage operation. `TRACe:TSTamp:TYPE?` queries the timestamp for readings that are presently stored in the buffer.

Also see

:SYSTem:TSTamp:RELative:RESet (on page 9-198)
:TRACe:TSTamp:TYPE? (on page 9-207)

:SYSTem:VERSion?

This command returns the present SCPI version.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:VERSion?
```

Details

This query command returns the SCPI version.

Example

`SYSTem:VERSion?`	Query the version. An example of a return is: `1996.0`

Also see

None

SYSTem:CARD:VCHannel[:STARt]?

This command indicates the first channel in the slot that supports voltage or 2-wire measurements.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:SYSTem:CARD:VCHannel[:STARt]? <NRf>
```

`<NRf>`	The slot number

Details

The channels that support voltage or 2-wire measurements are grouped, so you can use the start and end channel numbers to identify the group. If the card supports voltage or 2-wire measurements, the returned value is the number of the start channel. If only one channel on the card supports voltage or 2-wire measurements, the start channel matches the end channel. If the channel does not support voltage or 2-wire measurements, the return is `0`.

NOTE

To provide compatibility with the 2700 SCPI commands, you can include the slot number with the `CARD` command word for slots 1 and 2. For example, `SYST:CARD1:VCH?` is equivalent to `SYST:CARD1:VCH? 1`.

Example

`syst:card:vch? 1` `syst:card:vch:end? 1`	These commands return the voltage or 2-wire measurement start channel and end channel.

Also see

:SYSTem:CARD:VCHannel:END? (on page 9-185)

TRACe subsystem

The commands in this subsystem configure and control data storage into the buffer.

`SYSTem:PRESet` and `*RST` have no effect on TRACe commands. The listed defaults for `TRACe` commands are set at the factory.

NOTE

You can use either `:TRACe` or `:DATA` as the root command for this subsystem. In most cases, the documentation in this manual uses `:TRACe`. If you prefer to use `:DATA`, replace the `:TRACe` command words with `:DATA`.

:TRACe:CLEar:AUTO

This command enables or disables buffer autoclear.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	See Details

Usage

```
:TRACe:CLEar:AUTO <b>
:TRACe:CLEar:AUTO?
:DATA:CLEar:AUTO <b>
:DATA:CLEar:AUTO?
```

`<n>`	Enable buffer autoclear: `ON` Disable buffer autoclear: `OFF`

Details

When autoclear is enabled, the buffer automatically clears when the storage process starts. When disabled, readings are appended to old readings in the buffer until the buffer becomes full (110,000 readings) or the storage process is stopped.

Disabling autoclear automatically sets the buffer size to 110,000.

SYSTem:PRESet and *RST have no effect on commands in this subsystem. The default set at the factory is ON.

Also see

None

:TRACe:CLEar[:IMMediate]

This command clears readings from the buffer.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:TRACe:CLEar[:IMMediate]
:DATA:CLEar[:IMMediate]
```

Details

If you do not clear the buffer, a subsequent store overwrites the old readings. If the subsequent store is aborted before the buffer becomes full, some old readings may still be in the buffer.

When `TRAC:CLE` is sent while the instrument is displaying stored readings, the message `BUF CLEARED` is briefly displayed and the instrument returns to the normal measurement state.

Buffer readings are not cleared when the 2750 is turned off.

Also see

:TRACe:CLEar:AUTO (on page 9-200)

:TRACe:DATA:SELected?

This command specifies the readings to be returned.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:TRACe:DATA:SELected? <start>, <count>
:DATA:DATASELected? <start>, <count>
```

`<start>`	The starting point in the buffer for the readings
`<count>`	Number of readings

Details

Specify the readings to be returned by defining the starting point and the number of readings. The first reading in the buffer is #0.

SCPI compliant.

Example

Assume the buffer is configured for continuous storage and the buffer size is 100. Stop the storage process to return all the readings that were stored since the last time the buffer filled. This example returns the buffer location for the next stored reading. Assuming TRAC:NEXT? returned a value of 37, you can use that value as the `<count>` parameter for the TRAC:DATA:SEL? command to return the 37 readings (0 through 36):

```
TRAC:NEXT?                  ' Query buffer location for next stored reading.
TRAC:DATA:SEL? 0, 36        ' Return 37 buffer readings (0 through 36).
```

Also see

FORMat subsystem (on page 9-63)
:FORMat:ELEMents (on page 9-64)

:TRACe:DATA?

This command reads all readings in the buffer.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:TRACe:DATA?
:DATA:DATA?
```

Details

When this command is sent and the 2750 is addressed to talk, all the readings stored in the buffer are sent to the computer. The format in which the readings are sent over the bus is controlled by the `FORMat` subsystem.

Subsequent TRACe:DATA? commands will not retrieve readings already returned. However, once the buffer has filled and you have retrieved all buffer readings, you can again send TRACe:DATA? to retrieve all the stored readings.

NOTE

The buffer elements selected by :TRACe:ELEMents must match the bus elements selected by :FORMat:ELEMents. Otherwise, the error +313, Buffer & format element mismatch, occurs when using this command to send buffer readings over the bus.

SCPI compliant.

Also see

FORMat subsystem (on page 9-63)
:FORMat:ELEMents (on page 9-64)

:TRACe:FEED

This command specifies the source of readings for the buffer.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	CALC

Usage

```
:TRACe:FEED <name>
:TRACe:FEED?
:DATA:FEED <name>
:DATA:FEED?
```

`<name>`	The source of readings: ■ Store the reading before a math operation is applied: `[SENSe[1]` ■ Store the result of a math operation in the buffer: `CALCulate[1]` ■ Do not store readings in the buffer: `NONE`

Details

SCPI compliant.

Also see

Buffer (data store) (on page 8-1)
:TRACe:POINts (on page 9-206)

:TRACe:FEED:CONTrol

This command specifies the type of buffer control.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	See **Details**

Usage

```
:TRACe:FEED:CONTrol <name>
:TRACe:FEED:CONTrol?
:DATA:FEED:CONTrol <name>
:DATA:FEED:CONTrol?
```

`<name>`	■ Disable buffer storage: `NEVer` ■ Fill buffer and stop: `NEXT` ■ Continuously store readings in buffer: `ALWays`

Details

When `NEVer` is selected, storage into the buffer is disabled.

When `NEXT` is selected, the storage process starts, fills the buffer, and then stops. The buffer size is specified by the `:TRACe:POINts` command.

When `ALWays` is selected, the storage process starts and continues to store readings even after the buffer fills. After the buffer fills, readings overwrite the previously stored readings.

SYSTem:PRESet and *RST have no effect on commands in this subsystem. At the factory, this is set to `NEV`.

SCPI compliant.

Also see

Buffer (data store) (on page 8-1)
:TRACe:POINts (on page 9-206)

:TRACe:FREE?

This command reads the status of storage memory.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:TRACe:FREE?
:DATA:FREE?
```

Details

After sending this command and addressing the 2750 to talk, two values separated by commas are sent to the computer. The first value indicates how many bytes of memory are available, and the second value indicates how many bytes are reserved to store readings.

SCPI compliant.

Also see

None

:TRACe:NEXT?

This command queries the buffer location for the next stored reading.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:TRACe:NEXT?
:DATA:NEXT?
```

Details

When the storage process is aborted, you can use `TRACe:NEXT?` to determine the buffer location for the next stored reading. For example, if the last reading is stored at memory location 36, `TRACe:NEXT?` returns `37`. This query is useful when using the buffer in the continuous storage mode (`TRACe:FEED:CONTrol ALWays`).

When using the RS-232 interface, the `TRAC:DATA:SEL?` command should be used when recalling more than 100 points of buffer data. For large buffers, the computer may lose synchronization and data can be lost. To avoid this, use this query command to recall buffer data in 100 point chunks.

Example

Assume the buffer is configured for continuous storage and the buffer size is 100. Stop the storage process to return all the readings that were stored since the last time the buffer filled. This example returns the buffer location for the next stored reading. Assuming TRAC:NEXT? returned a value of 37, you can use that value as the `<count>` parameter for the TRAC:DATA:SEL? command to return the 37 readings (0 through 36):

```
TRAC:NEXT?                  ' Query buffer location for next stored reading.
TRAC:DATA:SEL? 0, 36        ' Return 37 buffer readings (0 through 36).
```

Also see

FORMat subsystem (on page 9-63)
:FORMat:ELEMents (on page 9-64)

:TRACe:NOTify

This command sets the number of readings a buffer can store.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	See **Details**

Usage

```
:TRACe:NOTify <NRf>
:TRACe:NOTify?
:DATA:NOTify <NRf>
:DATA:NOTify?
```

`<NRf>`	Number of readings: 2 to `109999`

Details

Specify number of stored readings that set the trace notify bit (B6) of the measurement event register. This value must be less than `TRACe:POINts` value.

The maximum valid parameter value for this command is one less than the present buffer size (which is set by the `TRACe:POINTs` command). For example, `TRACe:POINts 110000` sets the buffer size to 110,000 readings. For this buffer size, the maximum valid parameter value for `TRACe:NOTify` is 109999 (110000 − 1).

When an invalid parameter value is specified, the command is ignored and causes error −222, Parameter data out of range.

`SYSTem:PRESet` and `*RST` have no effect on commands in this subsystem. The value set at the factory is 50.

Also see

Measurement event register (on page 11-14)
Status model (on page 11-1)
:TRACe:POINts (on page 9-206)

:TRACe:POINts

This command sets the number of readings a buffer can store.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	100

Usage

```
:TRACe:POINts <NRf>
:TRACe:POINts?
:DATA:POINts <NRf>
:DATA:POINts?
```

`<n>`	Number of readings: 2 to `110000`

Details

With buffer autoclear enabled, you can set the buffer to store from 2 to 110,000 readings. A buffer size of zero or one is not valid (error −222).

With buffer autoclear disabled, you cannot use this command to set buffer size (error −221) because it is fixed at 110,000.

NOTE

The ¼ full and ½ buffer full measurement events are not intended to be used with buffer size smaller than four readings.

SCPI compliant.

Also see

None

:TRACe:TSTamp:FORMat

This command selects the timestamp format for buffer readings.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	Not applicable

Usage

```
:TRACe:TSTamp:FORMat <name>
:TRACe:TSTamp:FORMat?
:DATA:TSTamp:FORMat <name>
:DATA:TSTamp:FORMat?
```

`<name>`	Reference to first buffer reading: `ABSolute` Time between buffer readings: `DELTa`

Details

With `ABSolute` selected, each timestamp is referenced to the first reading stored in the buffer. With `DELTa` selected, timestamps provide the time between each buffer reading.

Changing the timestamp format clears the buffer.

The timestamp is only be included with a returned buffer reading if it is specified as a data element using FORMat:ELEMents.

Set to ABS at the factory.

Also see

:FORMat:ELEMents (on page 9-64)
:SYSTem:TSTamp:RELative:RESet (on page 9-198)
:SYSTem:TSTamp:TYPE (on page 9-198)

:TRACe:TSTamp:TYPE?

This command queries the timestamp type for readings presently in the buffer.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
:TRACe:TSTamp:TYPE?
:DATA:TSTamp:TYPE?
```

Details

When ABS is selected, each timestamp is referenced to the first reading stored in the buffer. With `DELT` selected, timestamps provide the time between each buffer reading.

`SYSTem:TSTamp:TYPE?` queries the timestamp that is used for the next storage operation. `TRACe:TSTamp:TYPE?` queries the timestamp for readings that are presently stored in the buffer.

Also see

:FORMat:ELEMents (on page 9-64)
:SYSTem:TSTamp:RELative:RESet (on page 9-198)
:SYSTem:TSTamp:TYPE (on page 9-198)

TRIGger subsystem

The commands in this subsystem configure and control the trigger model for the trigger layer.

:ABORt

This command aborts the operation.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:ABORt
```

Details

When this command is sent, the 2750 aborts the operation and returns to the top of the trigger model. If continuous initiation is disabled, the instrument goes into the idle state. If continuous initiation is enabled, operation continues at the top of the trigger model.

SCPI compliant.

Also see

None

:TRIGger[:SEQuence[1]]:COUNt

This command sets the measure count or the number of scans to perform.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	See **Details**

Usage

```
:TRIGger[:SEQuence[1]]:COUNt <NRf>
:TRIGger[:SEQuence[1]]:COUNt?
```

`<NRf>`	Count: `1` to `110000` or set to infinite (`INF`)

Details

This command specifies how many times operation loops around the measure layer. For example, if the count is set to 10, operation continues to loop around in the measure layer until 10 measurements are made. After the 10th measurement, operation proceeds to the scan layer.

`:SYSTem:PRESet` sets the count to `INF`. The `*RST` command sets the count to `1`.

SCPI compliant.

Also see

None

:TRIGger[:SEQuence[1]]:DELay

This command sets the measure layer delay.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0

Usage

```
:TRIGger[:SEQuence[1]]:DELay <n>
:TRIGger[:SEQuence[1]]:DELay?
```

`<n>`	Specify delay in seconds: `0` to `999999.999`

Details

This command delays operation in the measure layer. After the programmed event occurs, the instrument waits until the delay period expires before proceeding in the trigger model.

SCPI compliant.

Example

```
*RST            ' Restore *RST defaults.
TRAC:CLE        ' Clear buffer.
TRIG:DEL 0.5    ' Set delay for 0.5sec.
SAMP:COUN 10    ' Set sample count to 10.
READ?           ' Trigger, store, and request readings.
```

This example triggers 10 readings and stores them in the buffer. To send the readings to the computer, you must address the 2750 to talk after sending `READ?`.

Also see

Trigger model (on page 3-33)

:TRIGger[:SEQuence[1]]:DELay:AUTO

This command enables or disables autodelay.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	ON

Usage

```
:TRIGger[:SEQuence[1]]:DELay:AUTO <b>
:TRIGger[:SEQuence[1]]:DELay:AUTO?
```

`<n>`	Enable autodelay: `ON` Disable autodelay: `OFF`

Details

With autodelay selected, the instrument automatically selects a delay period that will provide sufficient settling for function and autorange changes and multi-phase measurements.

Disabling autodelay sets the delay time to 0.

Autodelay times are listed in Delay (auto or manual) (on page 7-2).

Also see

Delay (auto or manual) (on page 7-2)

:TRIGger[:SEQuence[1]]:SAMPle:COUNt

This command sets the number of measurements per trigger event.

Type	Affected by	Where saved	Default value
Command and query	Recall settings Instrument reset Power cycle	Save settings	1

Usage

```
:TRIGger[:SEQuence[1]]:SAMPle:COUNt <NRf>
:TRIGger[:SEQuence[1]]:SAMPle:COUNt?
```

`<NRf>`	The number of measurements: 1 to 110000

Details

This command specifies the number of readings to scan and store in the buffer.

If the sample count is greater than the number of channels in the scan list (scan list length), operation wraps around to the beginning of the scan list and continues. For example, assume the scan list is made up of channels 101, 102, and 103, and the sample count is set to 4. After channels 101, 102, and 103 are scanned, operation loops around to scan channel 101 again. The first and last readings in the buffer are channel 101.

When performing multiple scans (trigger count >1), sample readings overwrite the readings stored for the previous scan.

Continuous initiation must be disabled in order to set the sample count to more than 1.

Example

`:TRIG:SAMP:COUN 10`	Set the sample count to 10.

Also see

Trigger model (on page 3-33)

:TRIGger[:SEQuence[1]]:SIGNal

This command bypasses the measure control source.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
:TRIGger[:SEQuence[1]]:SIGNal
```

Details

This command allows you to bypass the specified control source or programmed event. The instrument must be waiting for the event when the command is sent. Otherwise, an error occurs and this command is ignored.

SCPI compliant.

Also see

None

:TRIGger[:SEQuence[1]]:SOURce

This command specifies the event control source for the measure layer.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	IMMediate

Usage

```
:TRIGger[:SEQuence[1]]:SOURce <name>
:TRIGger[:SEQuence[1]]:SOURce?
```

`<name>`	■ Pass operation through specified layer: `IMMediate` ■ Select manual event: `MANual` ■ Select GPIB trigger as event: `BUS` ■ Select external triggering as event: `EXTernal` ■ Select timer as event: `TIMer`

Details

When `IMMediate` is selected, operation immediately passes through the measure layer.

When `BUS` is selected, you can send a GPIB trigger (`GET` or `*TRG`) to continue operation through the measure layer.

With `TIMer` selected for the measure layer, the event occurs at the beginning of the timer interval and every time it times out. For example, if the measure layer timer is programmed for a 30-second interval, the first pass through the measure layer control source occurs immediately. Subsequent scan events occur every 30 seconds. The interval for the timer is set using the `:TRIGger[:SEQuence[1]]:TIMer` command.

SCPI confirmed.

Example

```
TRAC:CLE                ' Clear buffer.
INIT:CONT OFF           ' Disable continuous initiation.
TRIG:SOUR IMM           ' Select the immediate control source.
TRIG:COUN 1             ' Set to perform one scan.
SAMP:COUN 10            ' Set to scan 10 channels.
ROUT:SCAN (@101:110)    ' Set scan list channels; 101 through 110.
ROUT:SCAN:TSO IMM       ' Start scan when enabled and triggered.
ROUT:SCAN:LSEL INT      ' Enable scan.
READ?                   ' Trigger scan and request the readings.
```

This example program scans 10 channels (101 through 110).

Also see

:TRIGger[:SEQuence[1]]:TIMer (on page 9-212)

:TRIGger[:SEQuence[1]]:TIMer

This command sets the interval for the measure layer.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	0.1

Usage

```
:TRIGger[:SEQuence[1]]:TIMer <n>
:TRIGger[:SEQuence[1]]:TIMer <DEF|MIN|MAX>
:TRIGger[:SEQuence[1]]:TIMer?
:TRIGger[:SEQuence[1]]:TIMer? <DEF|MIN|MAX>
```

`<n>`	Timer interval in seconds: `0.001` to `999999.999`

Details

The timer is in effect only if the timer is the selected control source.

SCPI compliant.

Also see

:TRIGger[:SEQuence[1]]:SOURce (on page 9-211)

UNIT subsystem

The `UNIT` subsystem selects measurement units for readings.

:UNIT:TEMPerature

This command sets the unit of measurement used for temperature readings.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	C

Usage

```
:UNIT:TEMPerature <unitOfMeasure>
:UNIT:TEMPerature?
```

`<unitOfMeasure>`	The unit of measurement: ▪ Celsius: `C` or `CEL` ▪ Fahrenheit: `F` or `FAR` ▪ Kelvin: `K`

Details

This command selects the units for temperature readings (internal and external). This command controls the temperature reading units for the displays, buffer readings, and readings sent over the bus.

Also see

None

:UNIT:VOLTage[:DC]

This command sets the unit of measurement used for DC voltage readings.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	V

Usage

```
:UNIT:VOLTage[:DC] <unitOfMeasure>
:UNIT:VOLTage[:DC] <unitOfMeasure>, <clist>
:UNIT:VOLTage[:DC]?
:UNIT:VOLTage[:DC]? <clist>
```

`<unitOfMeasure>`	The unit of measurement: ■ Voltage: V ■ DB reference: DB
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Example

```
FUNC 'VOLT'          ' Select DCV function.
UNIT:VOLT DB         ' Select DCV dB.
UNIT:VOLT:DB:REF 1   ' Set dB reference to 1 V.
```

The following command sequence configures the 2750 to perform DCV dB measurements. A 1 V input is measured as 0 dB.

Also see

None

:UNIT:VOLTage[:DC]:DB:REFerence

This command sets the DB reference voltage for DC voltage readings.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	1

Usage

```
:UNIT:VOLTage[:DC]:DB:REFerence <n>
:UNIT:VOLTage[:DC]:DB:REFerence?
```

`<n>`	Specify reference in voltage: 1e−7 to 1000

Example

```
FUNC 'VOLT'          ' Select DCV function.
UNIT:VOLT DB         ' Select DCV dB.
UNIT:VOLT:DB:REF 1   ' Set dB reference to 1 V.
```

The following command sequence configures the 2750 to perform DCV dB measurements. A 1 V input is measured as 0 dB.

Also see

None

:UNIT:VOLTage:AC

This command sets the unit of measurement used for ac voltage readings.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	V

Usage

```
:UNIT:VOLTage:AC <unitOfMeasure>
:UNIT:VOLTage:AC <unitOfMeasure>, <clist>
:UNIT:VOLTage:AC?
:UNIT:VOLTage:AC? <clist>
```

`<unitOfMeasure>`	The unit of measurement: ■ Voltage: `V` ■ dB reference: `DB`
`<clist>`	Channel list parameter (refer to Channel assignments (on page 3-4))

Example

```
FUNC 'VOLT:AC', (@101, 105)   ' Set 101 and 105 for ACV.
UNIT:VOLT:AC DB, (@101, 105)  ' Set 101 and 105 for dB.
UNIT:VOLT:AC:DB:REF 10        ' Set 101 and 105 for 10 VAC dB reference.
```

This example configures channels 101 and 105 of the Model 7700 to perform ACV dB measurements when they are scanned. A 10 V input is measured as 0 dB.

Also see

None

:UNIT:VOLTage:AC:DB:REFerence

This command sets the dB reference voltage for AC voltage readings.

Type	Affected by	Where saved	Default value
Command and query	Recall settings SYSTem:PRESet Instrument reset Power cycle	Save settings	1

Usage

```
:UNIT:VOLTage:AC:DB:REFerence <n>
:UNIT:VOLTage:AC:DB:REFerence?
```

<n>	Specify reference in voltage: 1e−7 to 1000

Example

```
FUNC 'VOLT:AC', (@101, 105)   ' Set channels 101 and 105 for ACV.
UNIT:VOLT:AC DB, (@101, 105)  ' Set channels 101 and 105 for dB.
UNIT:VOLT:AC:DB:REF 10        ' Set channels 101 and 105 for 10 V ac dB reference.
```

This example configures channels 101 and 105 of the Model 7700 to perform ACV dB measurements when they are scanned. A 10 V input is measured as 0 dB.

Also see

None

Section 10

Common commands

In this section:

Common command summary 10-1
Common command descriptions..... 10-2

Common command summary

The IEEE Std 488.2 common commands that are supported by the 2750 are summarized in the following table. Although commands are shown in uppercase, common commands are not case sensitive, so you can use either uppercase or lowercase. These commands are designated and defined by the IEEE Std 488.2 standard.

Command	Name	Description
`*CLS`	Clear status	Clears all event registers and Error Queue. For detailed information, including status commands, see the Status model (on page 11-1).
`*ESE <NRf>`	Event enable command	Program the standard event status enable register. For detailed information, including status commands, see the Status model (on page 11-1).
`*ESE?`	Event enable query	Read the standard event status enable register and clear it. For detailed information, including status commands, see the Status model (on page 11-1).
`*ESR?`	Event status register query	Read/clear the standard event enable register. For detailed information, including status commands, see the Status model (on page 11-1).
`*IDN?`	Identification query	Returns the manufacturer, model number, serial number, and firmware revision levels of the unit. For information, see *IDN? (on page 10-6).
`*OPC`	Operation complete command	Set the operation complete bit in the standard event register after all pending commands have completed. For information, see *OPC (on page 10-6).
`*OPC?`	Operation complete query	Places an ASCII 1 into the output queue when all selected device operations have completed. For information, see *OPC (on page 10-6).
`*OPT?`	Option query	Returns the model numbers of the switching modules installed in the 2750. Returns `NONE` if a slot is empty. Refer to *OPT? (on page 10-8).
`*RCL <NRf>`	Recall command	Returns the 2750 to the user-saved setup (0, 1, or 2).
`*RST`	Reset command	Returns the 2750 to default conditions. For detailed information, see *RST (on page 10-10).
`*SAV`	Save command	Saves the present setup as the user-saved setup (0, 1, or 2). For information, see *SAV (on page 10-10).
`*SRE <NRf>`	Service request enable command	Programs the service request enable register. For detailed information, including status commands, see the Status model (on page 11-1).
`*SRE?`	Service request enable query	Reads the service request enable register. For detailed information, including status commands, see the Status model (on page 11-1).
`*STB?`	Status byte query	Reads the status byte register. For detailed information, including status commands, see the Status model (on page 11-1).
`*TRG`	Trigger command	Sends a bus trigger to the 2750. For information, see *TRG (on page 10-14).
`*TST?`	Self-test query	Performs a checksum test on ROM and returns the result. See *TST? (on page 10-15).
`*WAI`	Wait-to-continue command	Waits until all previous commands have completed. For detailed information, see *WAI (on page 10-15) .

Common command descriptions

Information for each common command is provided in the following topics.

*CLS

This command clears the event registers and queues.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
*CLS
```

Details

Use the `*CLS` command to clear (reset to 0) the bits of the following registers in the 2750:

- Operation Event Register
- Error Queue
- Questionable Event Register
- Standard Event Register
- Measurement Event Register

The Standard Event Enable Register is not reset by *CLS. Send the 0 parameter value with *ESE to reset all bits of that enable register to 0.

STATus:PRESet has no effect on the Error Queue.

Example

`*CLS`
Clear the event registers and queues.

Also see

Status model (on page 11-1)
*ESE (on page 10-4)
*ESR? (on page 10-5)
*STB? (on page 10-12)

*ESE

This command sets and queries bits in the status enable register of the standard event register.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	See **Details**

Usage

```
*ESE <NRf>
*ESE?
```

`<NRf>`	The value of the status enable register: ▪ `0`: Clear register ▪ `1`: Set OPC (B0) ▪ `4`: Set QYE (B2) ▪ `8`: Set DDE (B3) ▪ `16`: Set EXE (B4) ▪ `32`: Set CME (B5) ▪ `64`: Set URQ (B6) ▪ `128`: Set PON (B7) ▪ `255`: Set all bits

Details

The `*ESE` command programs the standard event enable register. This command is sent with the decimal equivalent of the binary value that determines the state (0 or 1) of the bits in the register. This register is cleared on power-up.

This register is used as a mask for the standard event status register. When a standard event is masked, the occurrence of that event does not set the event summary bit (ESB) in the status byte register. Conversely, when a standard event is unmasked (enabled), the occurrence of that event sets the ESB bit.

A cleared bit (`0`) in the enable register prevents (masks) the ESB bit in the status byte register from setting when the corresponding standard event occurs. A set bit (`1`) in the enable register enables the ESB bit to set when the corresponding standard event occurs.

The standard event enable register is shown in the following table and includes the decimal weight of each bit. The sum of the decimal weights of the bits to be set is the parameter value that is sent with the `*ESE` command.

If a command error (`CME`) occurs, bit B5 of the standard event status register sets. If a query error (`QYE`) occurs, bit B2 of the standard event status register sets. Because both events are unmasked (enabled), the occurrence of any one of them causes the ESB bit in the status byte register to set.

The standard event status register can be read by using the `*ESE?` query command.

Standard event enable register

Bit position	B7	B6	B5	B4	B3	B2	B1	B0
Event	PON	URQ	CME	EXE	DDE	QYE	—	OPC
Decimal weighting	128 (2^7)	64 (2^6)	32 (2^5)	16 (2^4)	8 (2^3)	4 (2^2)	—	1 (2^0)
Value	0/1	0/1	0/1	0/1	0/1	0/1	—	0/1

Bits B8 through B15 are not shown because they are not used.

Values are:

- 1 = Enable Standard Event
- 0 = Disable (Mask) Standard Event

Events are:

- PON = Power On
- URQ = User Request
- CME = Command Error
- EXE = Execution Error
- DDE = Device-dependent Error
- QYE = Query Error
- OPC = Operation Complete

Example

`*ESE 36`	Sets the CME and QYE bits of the Standard Event Register, where: ▪ CME (bit B5) = Decimal 32 ▪ QYE (bit B2) = Decimal 4 ▪ The Status Enable register is 36

Also see

*CLS (on page 10-3)
Status model (on page 11-1)

*ESR?

This command reads the standard event status register and clears it.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
*ESR?
```

Also see

Standard event status register (on page 11-11)

*IDN?

This command retrieves the identification string of the instrument.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
*IDN?
```

Details

The identification string includes the manufacturer, model number, serial number, and firmware revision of the instrument. The string is formatted as follows:

```
KEITHLEY INSTRUMENTS,MODEL nnnn,xxxxxxxx,yyyyyy/zzzzz
```

Where:

- *nnnn* is the model number
- *xxxxxxxx* is the serial number
- *yyyyyy/zzzzz* is the firmware revision levels of the digital board ROM and display board ROM

Example

`*IDN?`	Output: `KEITHLEY INSTRUMENTS,MODEL 2750,01234567/1.0.0i`

Also see

None

*OPC

This command sets the operation complete (OPC) bit after all pending commands, including overlapped commands, have been executed.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	Not applicable

Usage

```
*OPC
*OPC?
```

Details

When `*OPC` is sent, the OPC bit (bit 0) in the Status Event Status Register is set after all pending command operations have been executed.

After the *OPC command is sent, the Operation Complete bit (bit B0) of the Standard Event Status Register sets immediately after the last pending command is completed. If the corresponding bit (bit B0) in the Standard Event Enable Register and bit 5 (Event Summary Bit) of the Service Request Enable Register is set, the RQS/MSS (Request for Service/Master Summary Status) bit in the Status Byte Register is set.

When used with the initiate immediately command (:INITiate), the OPC bit in the Standard Event Status Register is not set until the 2750 returns to the idle state. The :INIT command operation is not considered to be finished until the 2750 returns to the idle state. See the description of *WAI for more information on command execution.

*OPC? places an ASCII 1 in the output queue after all pending operations are completed. When the 2750 is then addressed to talk, the 1 in the Output Queue is sent to the computer. The 1 in the Output Queue sets the Message Available (MAV) bit (B4) of the Status Byte Register. If the corresponding bit (B4) in the Service Request Enable Register is set, the Request for Service/Master Summary Status (RQS/MSS) bit in the Status Byte Register is set. The execution of OPC? is not completed until it has placed the 1 in the Output Queue.

If *OPC? is used with the initiate immediately command (:INITiate), the 1 is not placed into the Output Queue until the 2750 returns to the idle state. The :INIT command operation is not considered finished until the 2750 goes back into the idle state. See the description of *WAI for more information on command execution.

NOTE

The following commands take a long time to process and may benefit from using *OPC or *OPC?:

- *RST and SYST:PRES
- *RCL and *SAV
- ROUT:MULT:CLOS and ROUT:MULT:OPEN if the <clist> is long.
- CALC2:IMM and CALC2:IMM? when performing the standard deviation calculation on a large buffer. A 10,000 point buffer takes about 5.75 seconds.

Example 1

```
SYST:PRES   ' Returns 2750 to default setup.
INIT:CONT OFF   ' Disables continuous initiation.
ABORt    ' Aborts operation. Places 2750 in idle.
INIT:IMM   ' Initiate one trigger cycle.
*OPC   ' Sends the OPC command
*ESR?   ' Reads the Standard Event Status Register.

ABORt   ' Aborts operation. Places 2750 in idle.
*ESR?   ' Reads the Standard Event Status Register.

SYST:PRES   ' Returns 2750 to default setup.
```

The first group of commands sends the `*OPC` command after the `:INITiate` command and verifies that the OPC bit in the Standard Event Status Register does not set while the instrument continues to make measurements (not in idle). The second group of commands returns the 2750 to the idle state and verifies that the OPC bit was set.

After addressing the 2750 to talk, the returned value of 0 denotes that the bit (bit 0) is not set, indicating that the `:INITiate` operation is not complete.

After addressing the 2750 to talk, the returned value of 1 denotes that the bit (bit 1) is set, indicating that the `:INITiate` operation is complete.

Example 2

```
SYST:PRES       ' Returns 2750 to default setup.
INIT:CONT OFF   ' Disables continuous initiation.
ABORt           ' Aborts operation. Places 2750 in idle.
TRIG:COUN 1     ' These two commands configure the 2750
SAMP:COUN 5     ' to perform five measurements.
INIT            ' Starts the measurement process.
*OPC?           ' Sends the OPC? command.

SYST:PRES       ' Returns 2750 to default setup.
```

The example sequence demonstrates how to use `*OPC?` to signal the end of a measurement process.

After all five measurements are made and the instrument returns to the idle state, an ASCII `1` is placed in the Output Queue. After addressing the 2750 to talk, the `1` from the Output Queue is sent to the computer.

Also see

:INITiate[:IMMediate] (on page 9-34)
*WAI (on page 10-15)

*OPT?

This command determines which switching modules and pseudocards are installed in the 2750.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
*OPT?
```

Details

Returns the model numbers of the switching modules or pseudocards. The model number of an installed pseudocard is returned as if a real card is installed.

Example

*OPT?
If a Model 7700 is installed in slot 1 and the other slots are empty, the response message is: `7700, NONE, NONE, NONE, NONE`

Also see

Identify installed modules (on page 3-5)
Pseudocards (on page 2-1)

*RCL

This command returns the instrument to the setup that was stored in a specified memory location.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
*RCL <NRf>
```

`<NRf>`	An integer that represents the saved setup: `0`, `1`, or 2

Details

This command returns the 2750 to a setup configuration stored at a memory location. The `*SAV` command stores a setup configuration at a memory location.

The 2750 is shipped from the factory with `:SYSTem:PRESet` defaults loaded into the available setup memories. If a recall error occurs, the setup memories default to the `:SYSTem:PRESet` values.

NOTE

For RS-232 operation, `*OPC` or `*OPC?` should be used with `*SAV` and `*RCL`, which are slow responding commands. Refer to *OPC (on page 10-6) for additional information.

Example

```
*SAV 2   ' Save present setup in memory location 2.
*RST     ' Return 2750 to RST defaults.
*RCL 2   ' Return (recall) 2750 to setup stored in location 2.
```

Also see

"Saving setups" in the *Model 2750 User's Manual*
:SYSTem:PRESet (on page 9-195)
*SAV (on page 10-10)

*RST

This command resets the instrument settings to their default values.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
*RST
```

Details

Returns the instrument to default settings, cancels all pending commands, and cancels the response to any previously received `*OPC` and `*OPC?` commands.

This places the 2750 in the idle state.

The default values that are restored are listed in SCPI reference tables (on page 9-6).

NOTE

For RS-232 operation (and in some cases, GPIB operation), use `*OPC` or `*OPC?` with `*RST`, which is a command that responds slowly.

Also see

*OPC (on page 10-6)

*SAV

This command saves the present instrument settings as a user-saved setup.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Nonvolatile memory	Not applicable

Usage

```
*SAV <NRf>
```

`<NRf>`	The memory location: `0`, `1`, or 2

Details

Save the present instrument settings as a user-saved setup. You can restore the settings with the `*RCL` command.

Any control affected by `*RST` is saved by the `*SAV` command.

You can save up to three user-saved setups. A saved setup is approximately 4 kilobytes in size.

The 2750 ships from the factory with `SYSTem:PRESet` defaults loaded into the available setup memory. If a recall error occurs, the setup memory defaults to the `SYSTem:PRESet` values.

NOTE

For RS-232 operation, `*OPC` or `*OPC?` should be used with `*SAV` and `*RCL`, which are slow responding commands. Refer to *OPC (on page 10-6) for additional information.

Example

```
*SAV 2   ' Save present setup in memory location 2.
*RST     ' Return 2750 to RST defaults.
*RCL 2   ' Return (recall) 2750 to setup stored in location 2.
```

Also see

"Saving setups" in the *Model 2750 User's Manual*

*RCL (on page 10-9)

*SRE

This command sets or clears the bits of the service request enable register.

Type	Affected by	Where saved	Default value
Command and query	Not applicable	Not applicable	See Details

Usage

```
*SRE <NRf>
*SRE?
```

`<NRf>`	`0`: Clear enable register `1`: Set MSB bit (bit 0) `4`: Set EAV bit (bit 2) `8`: Set QSB bit (bit 3) `16`: Set MAV bit (bit 4) `32`: Set ESB bit (bit 5) `128`: Set OSB bit (bit 7) `255`: Set all bits

Details

The `*SRE` command programs the service request enable register. This command is sent with the decimal equivalent of the binary value that determines the state (0 or 1) of each bit in the register. This register is cleared on power-up.

This enable register is used with the status byte register to generate service requests (SRQ). With a bit in the service request enable register set, an SRQ occurs when the corresponding bit in the status byte register is set by an appropriate event. For details on register structure, refer to Status model (on page 11-1).

The service request enable register is shown in the following table. The sum of the decimal weights of the bits to set is the value that is sent with the `*SRE` command.

You can read the contents of the service request enable register using the `*SRE?` query command.

Service request enable register

Bit position	B7	B6	B5	B4	B3	B2	B1	B0
Event	OSB	—	ESB	MAV	QSB	EAV	—	MSB
Decimal weighting	128 (2^7)	—	32 (2^5)	16 (2^4)	8 (2^3)	4 (2^2)	—	1 (2^0)
Value	0/1	—	0/1	0/1	0/1	0/1	—	0/1

Values are:

- 1 = Enable Service Request Event
- 0 = Disable (Mask) Service Request Event

Events are:

- OSB = Operation Summary Bit
- ESB = Event Summary Bit
- MAV = Message Available
- QSB = Questionable Summary Bit
- EAV = Error Available
- MSB = Measurement Summary Bit

`*CLS` and `STATus:PRESet` have no effect on the Service Request Enable Register.

Example

`*SRE 48`
Set the ESB and MAV bits of the service request enable register, where: ▪ ESB (bit B5) = `32` ▪ MAV (bit B4) = `16` ▪ `<NRf>` = `48`

Also see

None

*STB?

This command reads the status byte register.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

`*STB?`

Details

The `*STB?` query command acquires the value of the status byte register. The status byte register is shown in the following table. The binary equivalent of the decimal value determines which bits in the register are set.

All bits except bit 6 in this register are set by other event registers and queues. Bit 6 sets when one or more enabled conditions occur.

The `*STB?` query command does not clear the status byte register. This register can only be cleared by clearing the related registers and queues. Register and queue structure are explained in Status structure (on page 11-1).

For example, for an acquired decimal value of 48, the binary equivalent is 00110000. This binary value indicates that bits 4 and 5 of the status byte register are set.

Status byte register

Bit position	B7	B6	B5	B4	B3	B2	B1	B0
Event	OSB	MSS, RQS	ESB	MAV	QSB	EAV	—	MSB
Decimal weighting	128 (2^7)	64 (2^6)	32 (2^5)	16 (2^4)	8 (2^3)	4 (2^2)	—	1 (2^0)
Value	0/1	0/1	0/1	0/1	0/1	0/1	—	0/1

Values are:

- 1 = Event Bit Set
- 0 = Event Bit Cleared

Events are:

- OSB = Operation Summary Bit
- MSS = Master Summary Status
- RQS = Request Service
- ESB = Event Summary Bit
- MAV = Message Available
- QSB = Questionable Summary Bit
- EAV = Error Available
- MSB = Measurement Summary Bit

The bits of the status byte register are described as follows:

Bit 0, measurement status (MSB): Set bit indicates that a measurement event has occurred. The event can be identified by reading the measurement event status register using the `:STATus:MEASurement?` command (see SCPI command reference (on page 9-1)).

Bit 1: Not used.

Bit 2, error available (EAV): Set bit indicates that an error or status message is present in the error queue. The message can be read using one of the following SCPI commands (see SCPI command reference (on page 9-1)):

`:SYSTem:ERRor?`

`:STATus:QUEue?`

Bit 3, questionable summary bit (QSB): Set bit indicates that a calibration error has occurred.

Bit 4, message available (MAV): Set bit indicates that a message is present in the output queue. The message is sent to the computer when the 2750 is addressed to talk.

Bit 5, event summary bit (ESB): Set bit indicates that an enabled standard event has occurred. The event can be identified by reading the standard event status register using the `*ESE?` query command.

Bit 6, master summary status (MSS) / request service (RQS): Set bit indicates that one or more enabled status byte conditions have occurred. The MSS bit can be read using the `STB?` query command, or the occurrence of a service request (RQS bit set) can be detected by performing a serial poll.

Bit 7, operation summary bit (OSB): Set bit indicates that an enabled operation event has occurred. The event can be identified by reading the Operation Event Status Register using the `:STATus:OPERation?` query command (see SCPI command reference (on page 9-1) for details).

Also see

None

*TRG

This command sends a bus trigger to the 2750.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

`*TRG`

Details

Use the `*TRG` command to issue a GPIB trigger to the 2750. It has the same effect as a group execute trigger (`GET`).

Use the `*TRG` command as an event t
o control operation. The instrument reacts to this trigger if `BUS` is the programmed arm control source. The control source is programmed from the TRIGger subsystem.

Example

```
*RST            ' Restore RST defaults.
TRIG:SOUR BUS   ' Select BUS control source.
TRIG:COUN INF   ' Set trigger layer count to infinity.
INIT            ' Take 2750 out of idle.
*TRG            ' Trigger one measurement.
```

This command sequence configures 2750 to be controlled by bus triggers. The last line, which sends a bus trigger, triggers one measurement. Each subsequent bus trigger also triggers a single measurement.

Also see

Status model (on page 11-1)
TRIGger subsystem (on page 9-208)

*TST?

This command runs self test and reads the result.

Type	Affected by	Where saved	Default value
Query only	Not applicable	Not applicable	Not applicable

Usage

```
*TST?
```

Details

This query command performs a checksum test on ROM and places the coded result (0 or 1) in the output queue. When the 2750 is addressed to talk, the coded result is sent from the output queue to the computer.

A returned value of zero (0) indicates that the test passed. A value of one (1) indicates that the test has failed.

Also see

None

*WAI

This command postpones the execution of subsequent commands until all previous overlapped commands are finished.

Type	Affected by	Where saved	Default value
Command only	Not applicable	Not applicable	Not applicable

Usage

```
*WAI
```

Details

There are two types of instrument commands:

- **Overlapped commands:** Commands that allow the execution of subsequent commands while instrument operations of the overlapped command are still in progress.
- **Sequential commands:** Commands whose operations must finish before the next command is executed.

The `*WAI` command suspends the execution of commands until the instrument operations of all previous overlapped commands are finished. The `*WAI` command is not needed for sequential commands.

The 2750 has three overlapped commands:

- `:INITiate`
- `:INITiate:CONTinuous ON`
- `*TRG`

The `:INITiate` commands remove the 2750 from the idle state. The device operations of `:INITiate` do not complete until the 2750 returns to idle. You can send the `*WAI` command after the `:INITiate` command to wait until the 2750 returns to idle before executing subsequent commands.

The `*TRG` command issues a bus trigger that can be used to provide the arm, scan, and measure events for the trigger model. By sending the `*WAI` command after the `*TRG` command, subsequent commands execute when the trigger model moves in response to `*TRG` and settles at its next state.

Example

```
SYST:PRES        ' Returns the instrument to default setup.
INIT:CONT OFF    ' Disables continuous initiation.
ABORt            ' Aborts operation. Places the instrument in idle.
TRIG:COUN 1      ' These two commands configure the instrument
SAMP:COUN 30     ' to perform 30 measurements.
INIT             ' Starts the measurement process.
*WAI             ' Sends the WAI command. Program waits for
                 ' the instrument to go into idle before executing
                 ' the next command.
DATA?            ' Requests one reading.
```

This command sequence shows how to use the WAI command to allow the 2750 to wait for the programmed measurements to be completed before requesting a reading.

Also see

*OPC (on page 10-6)
*OPC? (on page 10-6)
*TRG (on page 10-14)

Section 11

Status model

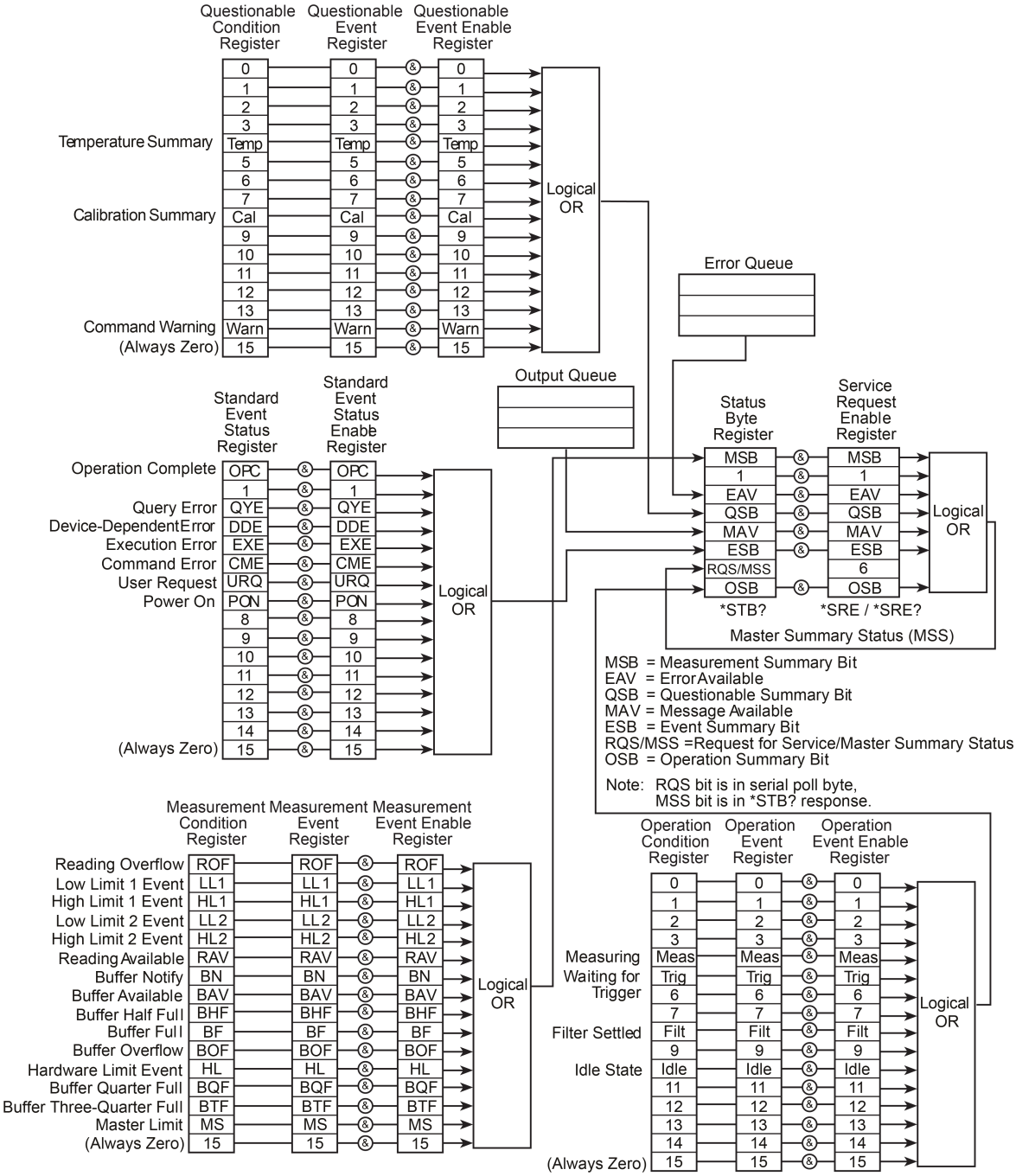
In this section:

Status structure 11-1
Clearing registers and queues 11-4
Programming enable registers 11-4
Reading registers 11-5
Status byte and service request (SRQ) 11-6
Status register sets 11-11
Queues 11-19

Status structure

The 2750 provides a series of status registers and queues that allow the operator to monitor and manipulate instrument events. The status structure is shown in the following figure. The heart of the status structure is the Status Byte Register. This register can be read by the user's test program to determine if a service request (SRQ) has occurred and what event caused it.

Figure 65: Model 2750 status register structure



Status byte and SRQ

The Status Byte Register receives the summary bits of four status register sets and two queues. The register sets and queues monitor the various instrument events. When an enabled event occurs, it sets a summary bit in the Status Byte Register. When a summary bit of the Status Byte is set and its corresponding enable bit is set (as programmed by the user), the RQS/MSS bit will set to indicate that an SRQ has occurred.

Status register sets

A typical status register set is made up of a condition register, an event register and an event enable register. A condition register is a read-only register that constantly updates to reflect the present operating conditions of the instrument.

When an event occurs, the appropriate event register bit sets to 1. The bit remains latched to 1 until the register is reset. When an event register bit is set and its corresponding enable bit is set (as programmed by the user), the output (summary) of the register will set to 1, which in turn sets the summary bit of the Status Byte Register.

Queues

The 2750 uses an Output Queue and an Error Queue. The response messages to query commands are placed in the Output Queue. As various programming errors and status messages occur, they are placed in the Error Queue. When a queue contains data, it sets the appropriate summary bit of the Status Byte Register.

Clearing registers and queues

When the 2750 is turned on, the bits of all registers in the status structure are cleared (reset to 0) and the two queues are empty. The commands to reset the event and event enable registers are:

- *CLS (on page 10-3)
- :STATus:PRESet (on page 9-169)

The commands to clear the error queue are:

- *CLS (on page 10-3)
- :STATus:QUEue:CLEar (on page 9-170)
- :SYSTem:CLEar (on page 9-187)

The Standard Event Enable Register is not reset by `STATus:PRESet` or `*CLS`. Send `*ESE 0` to reset all bits of that enable register to `0`. Refer to *ESE (on page 10-4) for details.

In addition to these commands, you can reset any enable register by sending the `0` parameter value with the individual command to program the register.

NOTE

`SYSTem:PRESet` and `*RST` have no effect on status structure registers and queues.

Programming enable registers

The only registers that you can program are the enable registers. All other registers in the status structure are read-only registers. The following explains how to ascertain the parameter values for the various commands used to program enable registers.

A command to program an event enable register is sent with a decimal parameter value that determines the state (0 or 1) of each bit in the appropriate register. The bit positions of the register indicate the binary parameter value. For example, if you wish to set bits B4, B3, and B1, the binary value would be 11010 (where B4=1, B3=1, B1=1, and all other bits are 0).

The binary value is then converted to its decimal equivalent:

Binary 11010 = Decimal 26

The following figure shows the decimal weight for each register bit. For example, to set bits B4, B3, and B1, the decimal parameter value is the sum of the decimal weights for those bits (16 + 8 + 2 = 26).

Figure 66: 16-bit status register

Bit Position	B7	B6	B5	B4	B3	B2	B1	B0
Binary Value	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
Decimal Weights	128 (2^7)	64 (2^6)	32 (2^5)	16 (2^4)	8 (2^3)	4 (2^2)	2 (2^1)	1 (2^0)

A. Bits 0 through7

Bit Position	B15	B14	B13	B12	B11	B10	B9	B8
Binary Value	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
Decimal Weights	32768 (2^{15})	16384 (2^{14})	8192 (2^{13})	4096 (2^{12})	2048 (2^{11})	1024 (2^{10})	512 (2^9)	256 (2^8)

B. Bits 8 through 15

Reading registers

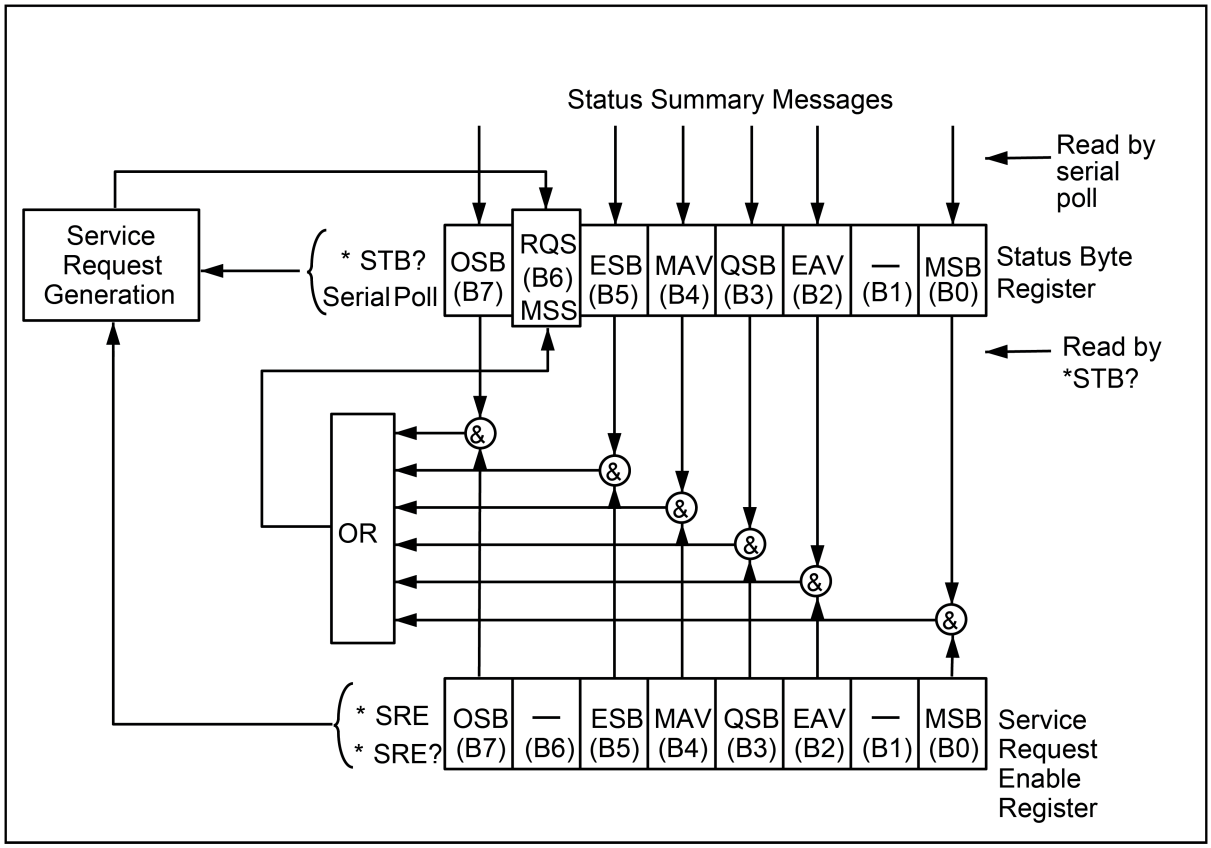
Any register in the status structure can be read by using the appropriate query (?) command. The following explains how to interpret the returned value (response message).

The response message for a register query is a decimal value. This decimal value has to be converted to its binary equivalent. For example, decimal 19 in binary is 10011. This binary value indicates that bits B0, B1, and B4 are set (1).

Status byte and service request (SRQ)

Service request is controlled by the 8-bit registers Status Byte Register and Service Request Enable Register. The following figure shows the structure of these registers.

Figure 67: Status byte and service request



Status byte and service request definitions

OSB	Operation Summary bit	&	Logical AND
MSS	Master Summary Status	OR	Logical OR
RQS	Request for Service		
ESB	Event Summary bit		
MAV	Message available		
QSB	Questionable Summary bit		
EAV	Error available		
MSB	Measurement Summary bit		

Status byte register

The summary messages from the status registers and queues set or clear the appropriate bits (B0, B2, B3, B4, B5, and B7) of the Status Byte Register. These bits do not latch, and their states (0 or 1) are solely dependent on the summary messages (0 or 1). For example, if the Standard Event Status Register is read, its register clears. As a result, its summary message resets to 0, which clears the ESB bit in the Status Byte Register.

The bits of the Status Byte Register are:

- **Bit B0, Measurement Summary Bit (MSB):** Set summary bit indicates that an enabled measurement event has occurred.
- **Bit B1:** Not used.
- **Bit B2, Error Available (EAV):** Set summary bit indicates that an error or status message is present in the Error Queue.
- **Bit B3, Questionable Summary Bit (QSB):** Set summary bit indicates that an enabled questionable event has occurred.
- **Bit B4, Message Available (MAV):** Set summary bit indicates that a response message is present in the Output Queue.
- **Bit B5, Event Summary Bit (ESB):** Set summary bit indicates that an enabled standard event has occurred.
- **Bit B6, Request Service (RQS)/Master Summary Status (MSS):** Set bit indicates that an enabled summary bit of the Status Byte Register is set.
- **Bit B7, Operation Summary (OSB):** Set summary bit indicates that an enabled operation event has occurred.

Bit B6 in the Status Byte Register is one of the following:

- The Master Summary Status (MSS) bit, sent in response to the `*STB?` command, indicates the status of any set bits with corresponding enable bits set.
- The Request for Service (RQS) bit, sent in response to a serial poll, indicates which device was requesting service by polling on the SRQ line. Refer to Serial polling and SRQ (on page 11-8) for details.

Service request enable register

The generation of a service request is controlled by the Service Request Enable Register. This register is programmed by you and serves as a mask for the Status Summary Message bits (B0, B2, B3, B4, B5, and B7) of the Status Byte Register. When masked, a set summary bit in the Status Byte Register cannot set bit B6 (MSS/RQS) of the Status Byte Register. Conversely, when unmasked, a set summary bit in the Status Byte Register sets bit B6.

A Status Summary Message bit in the Status Byte Register is masked when the corresponding bit in the Service Request Enable Register is cleared (0). When the masked summary bit in the Status Byte Register sets, it is ANDed with the corresponding cleared bit in the Service Request Enable Register. The logic 1 output of the AND gate is applied to the input of the OR gate, so it sets the MSS/RQS bit in the Status Byte Register.

The individual bits of the Service Request Enable Register can be set or cleared by using the `*SRE <NRf>` common command.

To read the Service Request Enable Register, use the `*SRE?` query command. The Service Request Enable Register clears when power is cycled or when `*SRE 0` is sent.

Serial poll and SRQ

Any enabled event summary bit that goes from 0 to 1 sets RQS and generates a service request (SRQ). In your test program, you can periodically read the Status Byte Register to check if a service request (SRQ) occurred and what caused it. If an SRQ occurred, the program can, for example, branch to an appropriate subroutine that services the request. Typically, service requests (SRQs) are managed by the serial poll sequence of the 2750. If an SRQ did not occur, bit B6 (RQS) of the Status Byte Register remains cleared, and the program proceeds normally after the serial poll is performed. If an SRQ did occur, bit B6 of the Status Byte Register sets, and the program can branch to a service subroutine when the SRQ is detected by the serial poll.

The serial poll automatically resets RQS of the Status Byte Register. This allows subsequent serial polls to monitor bit B6 for an SRQ occurrence generated by other event types. After a serial poll, the same event can cause another SRQ, even if the event register that caused the first SRQ has not been cleared.

A serial poll clears RQS but does not clear MSS. The MSS bit stays set until all Status Byte event summary bits are cleared.

SPE, SPD (serial polling)

When the instrument detects the serial polling enable (SPE) and serial polling disable (SPD) events, it sends the status byte of the instrument. This contains the serial poll byte of the instrument.

The serial poll byte contains information about internal functions. Generally, the serial polling sequence is used by the controller to determine which of several instruments has requested service with the SRQ line.

Status byte and service request commands

The following commands program and read the Status Byte Register and Service Request Enable Register:

- *STB? (on page 10-12): Reads the status byte register.
- *SRE (on page 10-11): Program and read the service request enable register (0 to 255).

To reset the bits of the Service Request Enable Register to 0, send:

```
*SRE 0
```

The following programming example sets MSS (B6) when an error occurs. The second command in the sequence enables EAV (error available). When an invalid command is sent (line 3), bits B2 (EAV) and B6 (MSS) of the Status Byte Register set to 1. The last command reads the Status Byte Register. You must address the 2750 to talk after sending a query command. To determine the exact nature of the error, you must read the error queue. Refer to Queues (on page 11-3) for details.

```
*CLS       ' Clear Error Queue
*SRE 4     ' Enable EAV.
*XYZ       ' Generate error.
*STB?      ' Read Status Byte Register.
```

For details on programming and reading registers, see Programming enable registers (on page 11-4) and Reading registers (on page 11-5).

Serial poll programming example

This example is written specifically for the KPCI-488.2 GPIB card and QuickBasic/VisualBasic with the appropriate IEEE libraries. Other types of cards and languages may have different function calls that are equivalent to the `initialize()`, `transmit()`, `send()`, `srq`, and `spoll()` calls used in the following code.

The following program stores 2000 readings in the buffer. When the buffer fills with 500 readings (quarter full), an SRQ occurs and a message is displayed on the computer to

indicate that event. An SRQ and message also occurs when the 1000th (half full), 1500th (three-quarter full), 1750th (buffer notify), and 2000th (full) reading is stored.

```
' $INCLUDE: 'ieeeqb.bi'

CLS   ' Clear computer output screen
CONST addr = 16   ' Set instrument address.

'
' Init GPIB.
'
CALL initialize(21, 0)
CALL transmit("unt unl listen " + STR$(addr) + " sdc unl", status%) ' Send Device
   Clear.

CALL send(addr, "*rst", status%)      ' Restore *rst defaults.
CALL send(addr, "trac:cle", status%)      ' Clear buffer.
CALL send(addr, "trig:coun inf", status%)      ' Infinite trigger count.
CALL send(addr, "trac:poin 2000", status%)      ' Set buffer size to 2000.
CALL send(addr, "trac:not 1750", status%)      ' Set Trace Notify bit on 1750th
   reading.
CALL send(addr, "trac:feed:cont next", status%)      ' Enable buffer.
CALL send(addr, "stat:pres", status%)      ' Reset measure enable bits.
CALL send(addr, "*cls", status%)      ' Clear all event registers.
CALL send(addr, "stat:meas:enab 13120", status%) ' Enable buffer bits; B6, B8, B9,
   B12, B13.
CALL send(addr, "*ese 0", status%)      ' Disable standard events.
CALL send(addr, "*sre 1", status%)      ' Enable measurement events.
CALL send(addr, "init", status%)      ' Start measure/store process.

N = 0      ' Initialize quarter buffer counter.

WaitSRQ:
   WHILE srq = 0: WEND   ' Wait for GPIB SRQ line to go true.
   CALL spoll(addr, poll%, status%)   ' Clear rqs/mss bit in status byte
         register.
   CALL send(addr, "*cls", status%)   ' Clear all event registers.
   N = N + 1   ' Increment buffer counter.
   IF N = 1 THEN GOTO QtrFull   ' Branch when buffer ¼ full.
   IF N = 2 THEN GOTO HalfFull   ' Branch when buffer ½ full.
   IF N = 3 THEN GOTO ThreeQtrFull   ' Branch when buffer ¾ full.
   IF N = 4 THEN GOTO 1750thReading   ' Branch when 1750th reading stored.
   PRINT "BUFFER FULL"   ' Display buffer full message.
END

QtrFull: PRINT "Buffer ¼ Full"    ' Display ¼ full message.
GOTO WaitSRQ   ' Return to WaitSRQ.
HalfFull: PRINT "Buffer ½ Full"   ' Display ½ full message.
GOTO WaitSRQ   ' Return to WaitSRQ.

ThreeQtrFull: PRINT "Buffer ¾ Full"   ' Display ¾ full message.
GOTO WaitSRQ      ' Return to WaitSRQ.

1750thReading: PRINT "1750th reading stored"      ' Display 1750th reading message.
GOTO WaitSRQ      ' Return to WaitSRQ.
```

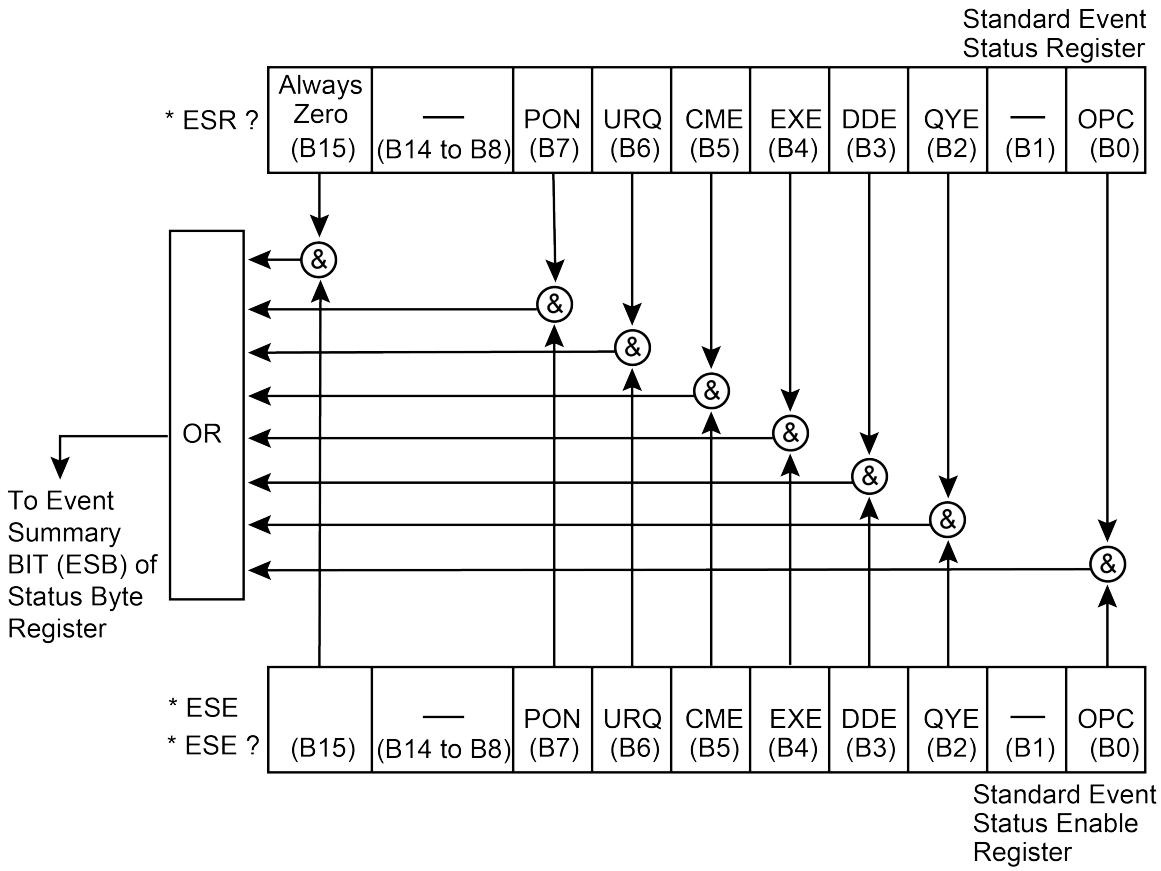
Status register sets

As shown in Status structure (on page 11-1), the register sets in the status structure of the 2750 include the Standard Event Status, Operation Event Status, Measurement Event Status, and Questionable Event Status.

Standard event status register

The status event register is shown in the following figure.

Figure 68: Standard event status registers



PON	Power on	&	Logical AND
URQ	User request	OR	Logical OR
CME	Command error		
EXE	Execution error		
DDE	Device-dependent error		
QYE	Query error		
OPC	Operation complete		

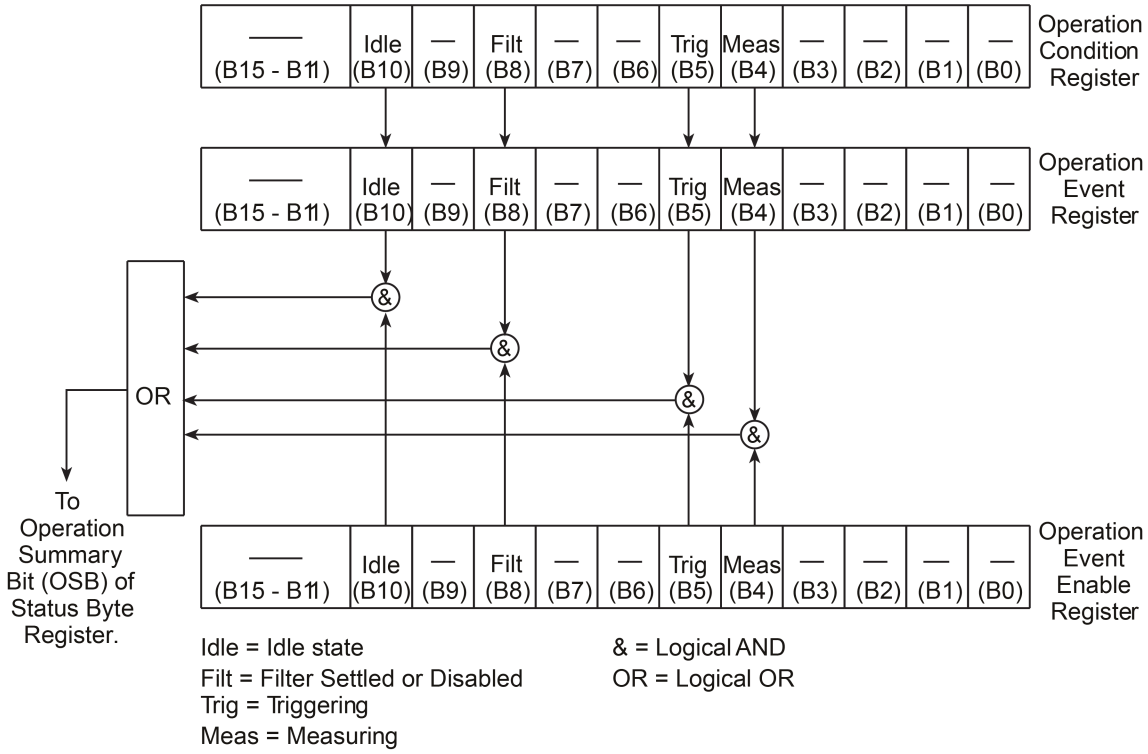
The used bits of the Standard Event Register are described in the following text.

- **Bit B0, Operation Complete (OPC):** Set bit indicates that all pending selected device operations are completed and the 2750 instrument is ready to accept new commands. The bit is set in response to an *OPC command. The `opc()` function can be used in place of the *OPC command. See Common commands (on page 10-1) for details on the *OPC command.
- **Bit B1:** Not used.
- **Bit B2, Query Error (QYE):** Set bit indicates that you attempted to read data from an empty output queue.
- **Bit B3, Device-Dependent Error (DDE):** Set bit indicates that an instrument operation did not execute properly due to some internal condition. Some of the errors specific to the 2750 that set this bit include:
  - **Error +516: Battery backed RAM error:** Data stored in RAM has been lost. Replace the battery if frequent failures occur.
  - **Error +517: Cannot resume scan:** Due to a card ID change, autoscan has disabled. The scan will not resume after a power interruption. For details, refer to Scan configuration (on page 3-34) and Auto scan (on page 3-45).
  - **Error +520: Saved setup scancard mismatch:** Settings for a user setup or power-on setup do not match the switching module types presently installed.
  - **Error +523: Card hardware error:** Communication with the microprocessor on a switching module card has been lost.
  - **Error +524: Unsupported card detected:** The 2750 has detected an installed Model 77XX switching module that is not supported by the current version of firmware.
- **Bit B4, Execution Error (EXE):** Set bit indicates that the 2750 detected an error when trying to execute a command.
- **Bit B6, User Request (URQ):** Set bit indicates that the LOCAL key on the 2750 instrument front panel was pressed.
- **Bit B7, Power ON (PON):** Set bit indicates that the 2750 instrument has been turned off and turned back on since the last time this register was read.

Operation event register

The operation event status register is shown in the following figure.

Figure 69: Operation event status



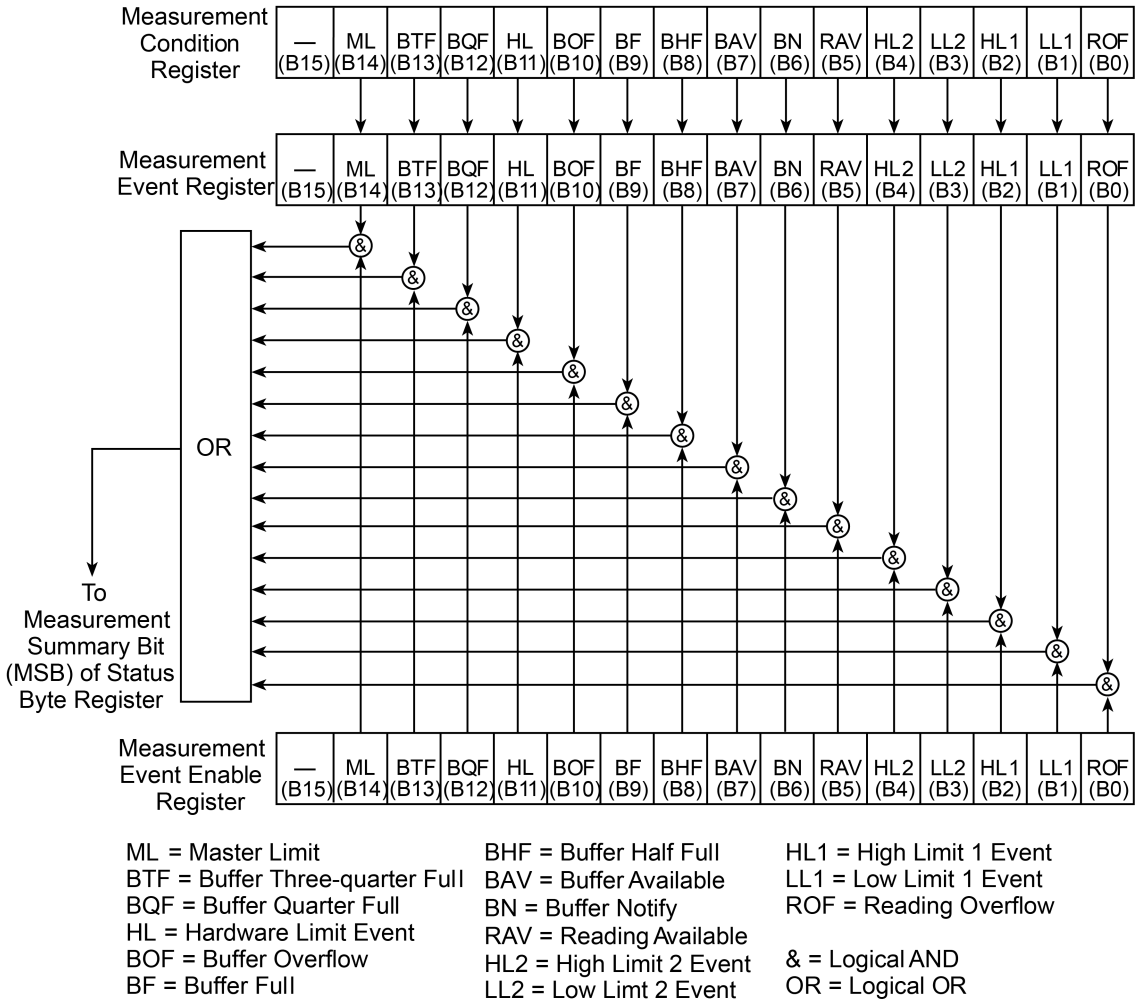
The bits of the Operation Event Register are described in the following text.

- **Bits B0 through B3:** Not used.
- **Bit B4, Measuring (Meas):** Set bit indicates that the instrument is making a measurement.
- **Bit B5, Waiting for Trigger (Trig):** Set bit indicates that the 2750 is in the trigger layer waiting for a trigger event to occur.
- **Bits B6 and B7:** Not used.
- **Bits B8, Filter Settled (Filt):** Set bit indicates that the filter has settled or the filter is disabled.
- **Bit B9:** Not used.
- **Bit B10, Idle State (Idle):** Set bit indicates the 2750 is in the idle state.
- **Bits B11 through B15:** Not used.

Measurement event register

The measurement event register is shown in the following figure.

Figure 70: Measurement event register



The used bits of the measurement event register are described in the following text.

- **Bit B0, Reading Overflow (ROF):** Set bit indicates that the reading exceeds the measurement range of the instrument.
- **Bit B1, Low Limit 1 Event (LL1):** Set bit indicates that a reading has reached or exceeded Low Limit 1.

- **Bit B2, High Limit 1 Event (HL1):** Set bit indicates that a reading has reached or exceeded High Limit 1.
- **Bit B3, Low Limit 2 Event (LL2):** Set bit indicates that a reading has reached or exceeded Low Limit 2.
- **Bit B4, High Limit 2 Fail (HL2):** Set bit indicates that a reading has reached or exceeded High Limit 2.
- **Bit B5, Reading Available (RAV):** Set bit indicates that a reading was made and processed.
- **Bit B6, Buffer Notify (BN):** Set bit is a notification that the user-specified number of readings have been stored in the buffer. The `TRACe:NOTify` command specifies the number of stored readings that will set this bit. Refer to :TRACe:NOTify (on page 9-205) for details.
- **Bit B7, Buffer Available (BAV):** Set bit indicates that there are at least two readings in the buffer.
- **Bit B8, Buffer Half Full (BHF):** Set bit indicates that the trace buffer is half full.
- **Bit B9, Buffer Full (BF):** Set bit indicates that the trace buffer is full.
- **Bit B10, Buffer Overflow (BOF):** Set bit indicates that the filled buffer has wrapped and written over previously stored readings.
- **Bit B11, Hardware Limit Event (HL):** Set bit indicates that a reading has exceeded the hardware limit.
- **Bit B12, Buffer Quarter Full (BQF):** Set bit indicates that the trace buffer is one-quarter full.

NOTE

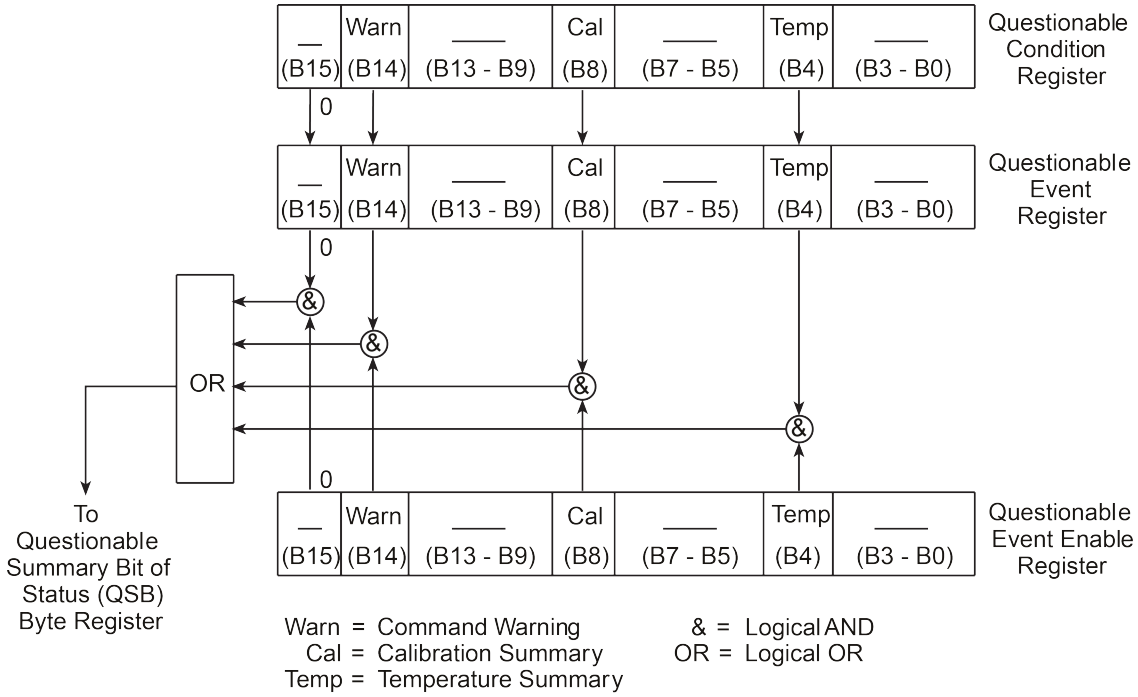
Bits B12 (¼ full) and B13 (¾ full) are not intended to be used with buffer sizes smaller than four readings.

- **Bit B13, Buffer Three-Quarter Full (BTF):** Set bit indicates that the trace buffer is three-quarters full.
- **Bit B14, Master Limit (ML):** Set bit indicates that one or more of the other limits have been reached or exceeded.
- **Bit B15:** Not used.

Questionable event register

The questionable event register is shown in the following figure.

Figure 71: Questionable event register



The used bits of the questionable event register are described in the following text.

- **Bits B0 through B3:** Not used.
- **Bit B4, Temperature Summary (Temp):** Set bit indicates that an invalid reference junction measurement has occurred for thermocouple temperature measurements.
- **Bits B5, B6 and B7:** Not used.
- **Bit B8, Calibration Summary (Cal):** Set bit indicates that an invalid calibration constant was detected during the power-up sequence. The instrument will instead use a default calibration constant. This error will clear after successful calibration of the instrument.
- **Bits B9 through B13:** Not used.
- **Bit B14, Command Warning (Warn):** Set bit indicates that a Signal Oriented Measurement Command parameter has been ignored.
- **Bit B15:** Not used.

NOTE

Whenever a questionable event occurs, the ERR annunciator turns on. The annunciator turns off when the questionable event clears.

Condition registers

As shown in the figures, all status register sets, except the standard event status register set, have a condition register. A condition register is a real-time read-only register that is constantly updated to reflect the present operating conditions of the instrument. For example, while a calculation is performed, Bit B10 (Idle) of the Operation Condition Register is set. When the calculation is completed, Bit B10 clears.

To read the condition registers, use the `:CONDition?` query for the status register set:

- :STATus:MEASurement:CONDition? (on page 9-166)
- :STATus:OPERation:CONDition? (on page 9-167)
- :STATus:QUEStionable:CONDition? (on page 9-169)

For details on reading registers, see Reading registers (on page 11-5).

Event registers

As shown in the figures, each status register set has an event register. An event register is a latched, read-only register whose bits are set by the corresponding condition register and transition filter. Once a bit in an event register is set, it remains set (latched) until the register is cleared by a specific clearing operation. The bits of an event register are logically ANDed with the bits of the corresponding enable register and applied to an OR gate. The output of the OR gate is applied to another register set or to the status byte register.

The `*ESR?` common command reads the standard event register (see Common commands (on page 10-1)).

An event register is cleared when it is read. The following operations clear all event registers:

- Cycling power
- Sending `*CLS`

STATus:PRESet has no effect.

Use the following commands to read the event registers:

- *ESR? (on page 10-5)
- :STATus:MEASurement[:EVENt]? (on page 9-167)
- :STATus:QUEStionable[:EVENt]? (on page 9-170)
- :STATus:OPERation[:EVENt]? (on page 9-168)

For details on reading registers, see Reading registers (on page 11-5).

Event enable registers

Each status register set has an enable register, as shown in the figures in Status structure (on page 11-1). An enable register is programmed by you and serves as a mask for the corresponding event register. An event bit is masked when the corresponding bit in the enable register is cleared (`0`). When masked, a set bit in an event register cannot set a bit in the Status Byte Register (1 AND 0 = 0).

To use the Status Byte Register to detect events (serial poll), you must unmask the events by setting the appropriate bits of the enable registers to `1`.

To program and query the Standard Event Status Register, use `*ESE` and `*ESE?`. All other enable registers are programmed and queried using the `:ENABle` and `:ENABle?` commands in the `STATus` subsystem.

An enable register is not cleared when it is read. The following operations affect the enable registers:

- Cycling power: Clears all enable registers
- `:STATus:PRESet` clears the following enable registers:
  - Operation Event Enable Register
  - Questionable Event Enable Register
  - Measurement Event Enable Register
- `*ESE 0`: Clears the Standard Event Status Enable Register

The following commands program and read the event enable registers:

- *ESE (on page 10-4)
- :STATus:OPERation:ENABle (on page 9-168)
- :STATus:MEASurement:ENABle (on page 9-166)
- :STATus:QUEStionable:ENABle (on page 9-170)

`*CLS` has no effect on the event enable registers.

See STATus subsystem (on page 9-166) for more information. For details on programming and reading registers, refer to Programming enable registers (on page 11-4) and Reading registers (on page 11-5).

NOTE

You can reset the bits of any enable register by sending the `0` parameter value with the appropriate enable command. For example:

```
STATus:OPERation:ENABle 0
```

Queues

The 2750 queues are first-in, first-out (FIFO) registers. The Output Queue holds reading and response messages. The Error Queue holds error and status messages.

Refer to Status structure (on page 11-1) for a figure that shows how the queues are structured with the other registers.

The output queue holds data that pertains to the normal operation of the instrument. For example, when a query command is sent, the response message is placed in the output queue.

When data is placed in the output queue, the Message Available (MAV) bit in the Status Byte Register is set. A data message is cleared from the output queue when it is read. The output queue is considered cleared when it is empty. An empty output queue clears the MAV bit in the Status Byte Register.

Read a message from the output queue by addressing the 2750 to talk after the appropriate query is sent.

The error queue holds error and status messages. When an error or status event occurs, a message that defines the error or status is placed in the error queue. This queue holds up to ten messages.

When a message is placed in the error queue, the Error Available (EAV) bit in the Status Byte Register is set. An error message is cleared from the error queue when it is read. The error queue is considered cleared when it is empty. An empty error queue clears the EAV bit in the Status Byte Register. To read an error message from the error queue, send either of the following SCPI query commands and then address the 2750 to talk:

- `:SYSTem:ERRor?`
- `:STATus:QUEue?`

On power-up, the Error Queue is empty. When empty, the message 0, No Error, is placed in the queue. All error messages are enabled and go into the Error Queue as they occur. Status messages are not enabled and do not go into the queue. If the queue becomes full, the message 350, Queue Overflow, occupies the last memory location.

Messages in the Error Queue are preceded by a code number. Negative (−) numbers are used for SCPI-defined messages. Positive (+) numbers are used for Keithley-defined messages. The messages are listed in the *Model 2750 User's Manual*.

The following commands control the messages and the error queue:

- :STATus:QUEue:CLEar (on page 9-170)
- :STATus:QUEue:DISable (on page 9-171)
- :STATus:QUEue[:NEXT]? (on page 9-173)
- :SYSTem:ERRor? (on page 9-188)
- :SYSTem:CLEar (on page 9-187)

Section 12

Temperature equations

In this section:

Temperature equations 12-1
Thermocouple equation 12-1
Thermistor equation 12-6
RTD equation 12-7

Temperature equations

The following topics contain information you can use when making temperature measurements.

- Thermocouple equation (on page 12-1): Documents the ITS-90 inverse function polynomial and the coefficients to calculate thermocouple temperature.
- Thermistor equation (on page 12-6): Documents the Steinhart-Hart equation, which is used to calculate thermistor temperature.
- RTD equations: Documents the Callendar-Van Dusen equation, which is used to calculate the temperature versus resistance readings listed in the RTD reference tables.

Thermocouple equation

The 2750 uses the ITS-90 inverse function coefficients for the polynomial to calculate thermocouple temperature. The 2750 measures the thermocouple voltage and then calculates temperature (in °C) as follows:

$$t_{90} = c_0 + c_1E + c_2E^2 + c_3E^3 \ldots c_iE^i$$

Where:

- $\mathbf{t}_{90}$ is the calculated temperature in °C.
- $\mathbf{c}_0$, $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$ ... $\mathbf{c}_i$ are the coefficients for the thermocouple type.
- **E** is the thermocouple voltage in microvolts (µV).

The coefficients for each thermocouple type are listed in the following tables.

Type B inverse function polynomial

	250 ºC to 700 ºC (291 µV to 2,431 µV)	**700º C to 1,820 ºC (2,431 µV to 13,820 µV)**
c_0 =	$9.842\,332\,1 \times 10^{1}$	$2.131\,507\,1\,1 \times 10^{2}$
c_1 =	$6.997\,150\,0 \times 10^{-1}$	$2.851\,050\,4\,1 \times 10^{-1}$
c_2 =	$-8.476\,530\,4 \times 10^{-4}$	$-5.274\,288\,7\,1 \times 10^{-5}$
c_3 =	$1.005\,264\,4 \times 10^{-6}$	$9.916\,080\,4\,1 \times 10^{-9}$
c_4 =	$-8.334\,595\,2 \times 10^{-10}$	$-1.296\,530\,3\,1 \times 10^{-12}$
c_5 =	$4.550\,854\,2 \times 10^{-13}$	$1.119\,587\,0\,1 \times 10^{-16}$
c_6 =	$-1.552\,303\,7 \times 10^{-16}$	$-6.062\,519\,9\,1 \times 10^{-21}$
c_7 =	$2.988\,675\,0 \times 10^{-20}$	$1.866\,169\,6\,1 \times 10^{-25}$
c_8 =	$-2.474\,286\,0 \times 10^{-24}$	$-2.487\,858\,5\,1 \times 10^{-30}$
Error:	0.03 °C to –0.02 °C	0.02 °C to –0.01 °C

$t_{90} = c_0 + c_1E + c_2E^2 + c_3E^3 \ldots c_iE^i$
Where:
t_{90} is the calculated temperature in °C.
E is the measured voltage in microvolts.

Type E inverse function polynomial

	–200 °C to 0 °C (–8,825 µV to 0 µV)	**0 °C to 1,000 °C (0 µV to 76,373 µV)**
c_0 =	0.0	0.0
c_1 =	$1.697\,728\,8 \times 10^{-2}$	$1.705\,703\,5 \times 10^{-2}$
c_2 =	$-4.351\,497\,0 \times 10^{-7}$	$-2.330\,175\,9 \times 10^{-7}$
c_3 =	$-1.585\,969\,7 \times 10^{-10}$	$6.543\,558\,5 \times 10^{-12}$
c_4 =	$-9.250\,287\,1 \times 10^{-14}$	$-7.356\,274\,9 \times 10^{-17}$
c_5 =	$-2.608\,431\,4 \times 10^{-17}$	$-1.789\,600\,1 \times 10^{-21}$
c_6 =	$-4.136\,019\,9 \times 10^{-21}$	$8.403\,616\,5 \times 10^{-26}$
c_7 =	$-3.403\,403\,0 \times 10^{-25}$	$-1.373\,587\,9 \times 10^{-30}$
c_8 =	$-1.156\,486\,0 \times 10^{-29}$	$1.062\,982\,3 \times 10^{-35}$
c_9 =		$-3.244\,708\,7 \times 10^{-41}$
Error:	0.03 °C to –0.01 °C	0.02 °C to –0.02 °C

$t_{90} = c_0 + c_1E + c_2E^2 + c_3E^3 \ldots c_iE^i$
Where:
t_{90} is the calculated temperature in °C.
E is the measured voltage in microvolts.

Type J inverse function polynomial

	–210 °C to 0 °C (–8,095 µV to 0 µV)	**0 °C to 760 °C (0 µV to 42,919 µV)**	**760 °C to 1,200 °C (42,919 µV to 69,553 µV)**
c_0 =	0.0	0.0	$-3.113\ 581\ 87 \times 10^{3}$
c_1 =	$1.952\ 826\ 8 \times 10^{-2}$	$1.978\ 425 \times 10^{-2}$	$3.005\ 436\ 84 \times 10^{-1}$
c_2 =	$1.228\ 618\ 5 \times 10^{-6}$	$2.001\ 204 \times 10^{-7}$	$9.947\ 732\ 30 \times 10^{-6}$
c_3 =	$1.075\ 217\ 8 \times 10^{-9}$	$1.036\ 969 \times 10^{-11}$	$1.702\ 766\ 30 \times 10^{-10}$
c_4 =	$5.908\ 693\ 3 \times 10^{-13}$	$2.549\ 687 \times 10^{-16}$	$1.430\ 334\ 68 \times 10^{-15}$
c_5 =	$1.725\ 671\ 3 \times 10^{-16}$	$3.585\ 153 \times 10^{-21}$	$4.738\ 860\ 84 \times 10^{-21}$
c_6 =	$2.813\ 151\ 3 \times 10^{-20}$	$5.344\ 285 \times 10^{-26}$	
c_7 =	$2.396\ 337\ 0 \times 10^{-24}$	$5.099\ 890 \times 10^{-31}$	
c_8 =	$8.382\ 332\ 1 \times 10^{-29}$		
Error:	0.03 °C to –0.05 °C	0.04 °C to –0.04 °C	0.03 °C to –0.05 °C

$t_{90} = c_0 + c_1E + c_2E^2 + c_3E^3 \ldots c_iE^i$
Where:
t_{90} is the calculated temperature in °C.
E is the measured voltage in microvolts.

Type K inverse function polynomial

	–200 °C to 0 °C (–5,891 µV to 0 µV)	**0 °C to 500 °C (0 µV to 20,644 µV)**	**500 °C to 1,372 °C (20,644 µV to 54,886 µV)**
c_0 =	0.0	0.0	$-1.318\ 058 \times 10^{2}$
c_1 =	$2.517\ 346\ 2 \times 10^{-2}$	$2.508\ 355\ 2 \times 10^{-2}$	$4.830\ 222 \times 10^{-2}$
c_2 =	$-1.166\ 287\ 8 \times 10^{-6}$	$7.860\ 106\ 2 \times 10^{-8}$	$-1.646\ 031 \times 10^{-6}$
c_3 =	$-1.083\ 363\ 8 \times 10^{-9}$	$-2.503\ 131\ 2 \times 10^{-10}$	$5.464\ 731 \times 10^{-11}$
c_4 =	$-8.977\ 354\ 0 \times 10^{-13}$	$8.315\ 270\ 2 \times 10^{-14}$	$-9.650\ 715 \times 10^{-16}$
c_5 =	$-3.734\ 237\ 7 \times 10^{-16}$	$-1.228\ 034\ 2 \times 10^{-17}$	$8.802\ 193 \times 10^{-21}$
c_6 =	$-8.663\ 264\ 3 \times 10^{-20}$	$9.804\ 036\ 2 \times 10^{-22}$	$-3.110\ 810 \times 10^{-26}$
c_7 =	$-1.045\ 059\ 8 \times 10^{-23}$	$-4.413\ 030\ 2 \times 10^{-26}$	
c_8 =	$-5.192\ 057\ 7 \times 10^{-28}$	$1.057\ 734\ 2 \times 10^{-30}$	
c_9 =		$-1.052\ 755\ 2 \times 10^{-35}$	
Error:	0.04 °C to –0.02 °C	0.04 °C to –0.05 °C	0.06 °C to –0.05 °C

$t_{90} = c_0 + c_1E + c_2E^2 + c_3E^3 \ldots c_iE^i$
Where:
t_{90} is the calculated temperature in °C.
E is the measured voltage in microvolts.

Type N inverse function polynomial

	–200 °C to 0 °C (–3,990 µV to 0 µV)	**0 °C to 600 °C (0 µV to 20,613 µV)**	**600 °C to 1,300 °C (20,613 µV to 47,513 µV)**
c_0 =	0.0	0.0	$1.972\,485 \times 10^{1}$
c_1 =	$3.843\,684\,7 \times 10^{-2}$	$3.868\,96 \times 10^{-2}$	$3.300\,943 \times 10^{-2}$
c_2 =	$1.101\,048\,5 \times 10^{-6}$	$-1.082\,67 \times 10^{-6}$	$-3.915\,159 \times 10^{-7}$
c_3 =	$5.222\,931\,2 \times 10^{-9}$	$4.702\,05 \times 10^{-11}$	$9.855\,391 \times 10^{-12}$
c_4 =	$7.206\,052\,5 \times 10^{-12}$	$-2.121\,69 \times 10^{-18}$	$-1.274\,371 \times 10^{-16}$
c_5 =	$5.848\,858\,6 \times 10^{-15}$	$-1.172\,72 \times 10^{-19}$	$7.767\,022 \times 10^{-22}$
c_6 =	$2.775\,491\,6 \times 10^{-18}$	$5.392\,80 \times 10^{-24}$	
c_7 =	$7.707\,516\,6 \times 10^{-22}$	$-7.981\,56 \times 10^{-29}$	
c_8 =	$1.158\,266\,5 \times 10^{-25}$		
c_9 =	$7.313\,886\,8 \times 10^{-30}$		
Error:	0.03 °C to –0.02 °C	0.03 °C to –0.02 °C	0.02 °C to –0.04 °C

$t_{90} = c_0 + c_1E + c_2E^2 + c_3E^3 \ldots c_iE^i$
Where:
t_{90} is the calculated temperature in °C.
E is the measured voltage in microvolts.

Type R inverse function polynomial

	–50 °C to 250 °C (–226 µV to 1,923 µV)	**250 °C to 1,200 °C (1,923 µV to 13,228 µV)**	**1,064 °C to 1,664.5 °C (11,361 µV to 19,739 µV)**	**1,664.5 °C to 1,768.1 °C (19,739 µV to 21,103 µV)**
c_0 =	0.0	$1.334\,584\,505 \times 10^{1}$	$-8.199\,599\,416 \times 10^{1}$	$3.406\,177\,836 \times 10^{4}$
c_1 =	$1.889\,138\,0 \times 10^{-1}$	$1.472\,644\,573 \times 10^{-1}$	$1.553\,962\,042 \times 10^{-1}$	$-7.023\,729\,171$
c_2 =	$-9.383\,529\,0 \times 10^{-5}$	$-1.844\,024\,844 \times 10^{-5}$	$-8.342\,197\,663 \times 10^{-6}$	$5.582\,903\,813 \times 10^{-4}$
c_3 =	$1.306\,861\,9 \times 10^{-7}$	$4.031\,129\,726 \times 10^{-9}$	$4.279\,433\,549 \times 10^{-10}$	$-1.952\,394\,635 \times 10^{-8}$
c_4 =	$-2.270\,358\,0 \times 10^{-10}$	$-6.249\,428\,360 \times 10^{-13}$	$-1.191\,577\,910 \times 10^{-14}$	$2.560\,740\,231 \times 10^{-13}$
c_5 =	$3.514\,565\,9 \times 10^{-13}$	$6.468\,412\,046 \times 10^{-17}$	$1.492\,290\,091 \times 10^{-19}$	
c_6 =	$-3.895\,390\,0 \times 10^{-16}$	$-4.458\,750\,426 \times 10^{-21}$		
c_7 =	$2.823\,947\,1 \times 10^{-19}$	$1.994\,710\,149 \times 10^{-25}$		
c_8 =	$-1.260\,728\,1 \times 10^{-22}$	$-5.313\,401\,790 \times 10^{-30}$		
c_9 =	$3.135\,361\,1 \times 10^{-26}$	$6.481\,976\,217 \times 10^{-35}$		
c_{10} =	$-3.318\,776\,9 \times 10^{-30}$			
Error:	0.02 °C to –0.02 °C	0.005 °C to –0.005 °C	0.001 °C to –0.0005 °C	0.002 °C to –0.001 °C

$t_{90} = c_0 + c_1E + c_2E^2 + c_3E^3 \ldots c_iE^i$
Where:
t_{90} is the calculated temperature in °C.
E is the measured voltage in microvolts.

Type S inverse function polynomial

	−50 °C to 250 °C (−235 µV to 1,874 µV)	**250 °C to 1,200 °C (1,874 µV to 11,950 µV)**	**1,064 °C to 1,664.5 °C (10,332 µV to 17,536 µV)**	**1,664.5 °C to 1,768.1 °C (17,536 µV to 18,693 µV)**
c_0 =	0.0	$1.291\ 509\ 199 \times 10^{1}$	$-8.087\ 801\ 117 \times 10^{1}$	$5.333\ 875\ 126 \times 10^{4}$
c_1 =	$1.849\ 494\ 60 \times 10^{-1}$	$1.466\ 298\ 863 \times 10^{-1}$	$1.621\ 573\ 104 \times 10^{-1}$	$-1.235\ 892\ 298 \times 10^{1}$
c_2 =	$-8.005\ 050\ 62 \times 10^{-5}$	$-1.534\ 713\ 402 \times 10^{-5}$	$-8.536\ 869\ 453 \times 10^{-6}$	$1.092\ 657\ 613 \times 10^{-3}$
c_3 =	$1.022\ 374\ 30 \times 10^{-7}$	$3.145\ 945\ 973 \times 10^{-9}$	$4.719\ 686\ 976 \times 10^{-10}$	$-4.265\ 693\ 686 \times 10^{-8}$
c_4 =	$-1.522\ 485\ 92 \times 10^{-10}$	$-4.163\ 257\ 839 \times 10^{-13}$	$-1.441\ 693\ 666 \times 10^{-14}$	$6.247\ 205\ 420 \times 10^{-13}$
c_5 =	$1.888\ 213\ 43 \times 10^{-13}$	$3.187\ 963\ 771 \times 10^{-17}$	$2.081\ 618\ 890 \times 10^{-19}$	
c_6 =	$-1.590\ 859\ 41 \times 10^{-16}$	$-1.291\ 637\ 500 \times 10^{-21}$		
c_7 =	$8.230\ 278\ 80 \times 10^{-20}$	$2.183\ 475\ 087 \times 10^{-26}$		
c_8 =	$-2.341\ 819\ 44 \times 10^{-23}$	$-1.447\ 379\ 511 \times 10^{-31}$		
c_9 =	$2.797\ 862\ 60 \times 10^{-27}$	$8.211\ 272\ 125 \times 10^{-36}$		
Error:	0.02 °C to –0.02 °C	0.01 °C to –0.01 °C	0.0002 °C to –0.0002 °C	0.002 °C to –0.002 °C

$t_{90} = c_0 + c_1E + c_2E^2 + c_3E^3 \ldots c_iE^i$
Where:
t_{90} is the calculated temperature in °C.
E is the measured voltage in microvolts.

Type T inverse function polynomial

	–200 °C to 0 °C (–5,603 µV to 0 µV)	**0 °C to 400 °C (0 µV to 20,872 µV)**
c_0 =	0.0	0.0
c_1 =	$2.594\ 919\ 2 \times 10^{-2}$	$2.592\ 800 \times 10^{-2}$
c_2 =	$2.131\ 696\ 7 \times 10^{-7}$	$7.602\ 961 \times 10$
c_3 =	$7.901\ 869\ 2 \times 10^{-10}$	$7\ 4.637\ 791 \times 10^{-11}$
c_4 =	$4.252\ 777\ 7 \times 10^{-13}$	$2.165\ 394 \times 10^{-15}$
c_5 =	$1.330\ 447\ 3 \times 10^{-16}$	$6.048\ 144 \times 10^{-20}$
c_6 =	$2.024\ 144\ 6 \times 10^{-20}$	$7.293\ 422 \times 10^{-25}$
c_7 =	$1.266\ 817\ 1 \times 10^{-24}$	
Error:	0.04 °C to –0.02 °C	0.03 °C to –0.03 °C

$t_{90} = c_0 + c_1E + c_2E^2 + c_3E^3 \ldots c_iE^i$
Where:
t_{90} is the calculated temperature in °C.
E is the measured voltage in microvolts.

Thermistor equation

Temperature (in Kelvin) is calculated using the Steinhart-Hart equation as follows:

$$T_K = \frac{1}{A + (B\ln R) + [C(\ln R)^3]}$$

Where:

T_K = The calculated temperature in Kelvin.

lnR = The natural log of the measured resistance of the thermistor.

A, **B**, and **C** =The curve fitting constants.

The constants for the three thermistor types used by the 2750 are listed below.

2750 curve fitting constants for thermistors

Constant	2,252 Ω at 25 °C (Series 44004)	5,000 Ω at 25 °C (Series 44007)	10 kΩ at 25 °C (Series 44006)
A	0.0014733	0.001288	0.0010295
B	0.0002372	0.0002356	0.0002391
C	1.074e−7	9.557e−8	1.568e−7

Selecting a thermistor

The thermistor manufacturer's specified curve fitting values (A, B, and C) may not be exactly the same as the ones used by the 2750. If they are not exactly the same, perform the following steps to select a thermistor to use with the 2750.

NOTE

The specified thermistor temperature measurement accuracy of the 2750 is based on the curve fitting constants listed in Thermistor equation (on page 12-6). If the thermistor manufacturer's curve fitting constants are not exactly the same as the ones listed in Thermistor equation (on page 12-6), accuracy will be affected.

1. Choose the thermistor type to be used: 2,252 Ω, 5 kΩ, or 10 kΩ.
2. Compare the A, B, and C constants from the thermistor manufacturer with those used by the 2750 (see Thermistor equation (on page 12-6)). 2750 specifications are available for download from the Keithley Product Support and Downloads web page (tek.com/support/product-support).
3. Select a thermistor that closely matches the A, B, and C constants in Thermistor equation (on page 12-6).
4. Analyze the differences between the two sets of curve fitting constants to determine the affect on measurement accuracy.

Converting Kelvin to °C

The temperature in Kelvin can be converted to °C as follows:

$$T_{°C} = T_K - 273.15$$

Where:

- **$T_{°C}$** is the temperature in °C.
- **T_K** is the calculated Kelvin temperature.

Example

Calculate the temperature for a Series 44007 thermistor that measures 5 kΩ (R):

lnR = ln(5000) = 8.5172

A = 0.001288

B = 0.0002356

C = 9.557e-8

$$\begin{aligned}
\mathbf{T_K} &= 1 / \{A + (B\ln R) + [(C)(\ln R)^3]\} \\
&= 1 / \{0.001288 + [(0.0002356)(8.5172)] + [(9.557e{-}8)(8.5172^3)]\} \\
&= 1 / (0.001288 + 0.002007 + 0.000059) \\
&= 1 / 0.003354 \\
&= 298.15 \\
\mathbf{T_{°C}} &= T_K - 273.15 \\
&= 298.15 - 273.15 \\
&= 25\ °C
\end{aligned}$$

RTD equation

The temperature versus resistance readings listed in the RTD reference tables are calculated using the Callendar-Van Dusen equation:

$$R_{RTD} = R_0\,[1 + AT + BT^2 + CT^3(T-100)]$$

Where:

R_{RTD} = The calculated resistance of the RTD

R_0 = The known RTD resistance at 0 °C

T = The temperature in °C

A = Alpha [1 + (delta/100)]

B = –1 (alpha)(delta)(1e−4)

C = –1 (alpha)(beta)(1e−8)

The alpha, beta, and delta values are listed in the following table.

RTD parameters

Type	Standard	Alpha	Beta	Delta	Ω at 0 °C
PT100	ITS-90	0.003850	0.10863	1.49990	100 Ω
D100	ITS-90	0.003920	0.10630	1.49710	100 Ω
F100	ITS-90	0.003900	0.11000	1.49589	100 Ω
PT385	IPTS-68	0.003850	0.11100	1.50700	100 Ω
PT3916	IPTS-68	0.003916	0.11600	1.50594	100 Ω

For example, calculate the resistance of a PT100 RTD at 100°C (T). The following R_0 (Ω at 0°C), alpha, beta, and delta values are used for the PT100 RTD.

T = 100 °C

R_0 (Ω at 0 °C) = 100 Ω

alpha = 0.003850

delta = 1.49990

Using the above alpha and delta values, A and B are calculated as follows:

A = 0.00385 [1 + (1.4999/100)]
= 0.00385 (1.014999)
= 0.003907746

B = –1 (0.00385)(1.4999)(1e-4)
= –1 (0.005774615)(1e-4)
= –5.774615e-7

The resistance of the RTD at 100 °C (R_{100}) is then calculated as follows:

R_{100} = R_0 [1 + AT + BT^2]
= 100 {1 + [(0.003907746)(100)] + [(–5.774615e-7)(100^2)]}
= 100 [1 + 0.3907746 + (-0.005774615)]
= 100 (1.385)
= 138.5 Ω

Section 13

IEEE-488 bus overview

In this section:

Introduction 13-1
Bus description..... 13-2
Bus lines..... 13-4
Bus commands 13-6
Interface function codes 13-13

Introduction

The IEEE-488 bus is a communication system between two or more electronic devices. A device can be either an instrument or a computer. When a computer is used on the bus, it serves to supervise the communication exchange between all the devices and is known as the controller. Supervision by the controller consists of determining which device talks and which device listens. As a talker, a device outputs information and as a listener, a device receives information. To simplify the task of keeping track of the devices, a unique address number is assigned to each one.

On the bus, only one device can talk at a time and is addressed to talk by the controller. The device that is talking is known as the active talker. The devices that need to listen to the talker are addressed to listen by the controller. Each listener is then referred to as an active listener. Devices that do not need to listen are instructed to unlisten. The reason for the unlisten instruction is to optimize the speed of bus information transfer since the task of listening takes up bus time. Through the use of control lines, a handshake sequence takes place in the transfer process of information from a talker to a listener. This handshake sequence helps ensure the credibility of the information transfer. The basic handshake sequence between an active controller (talker) and a listener is as follows:

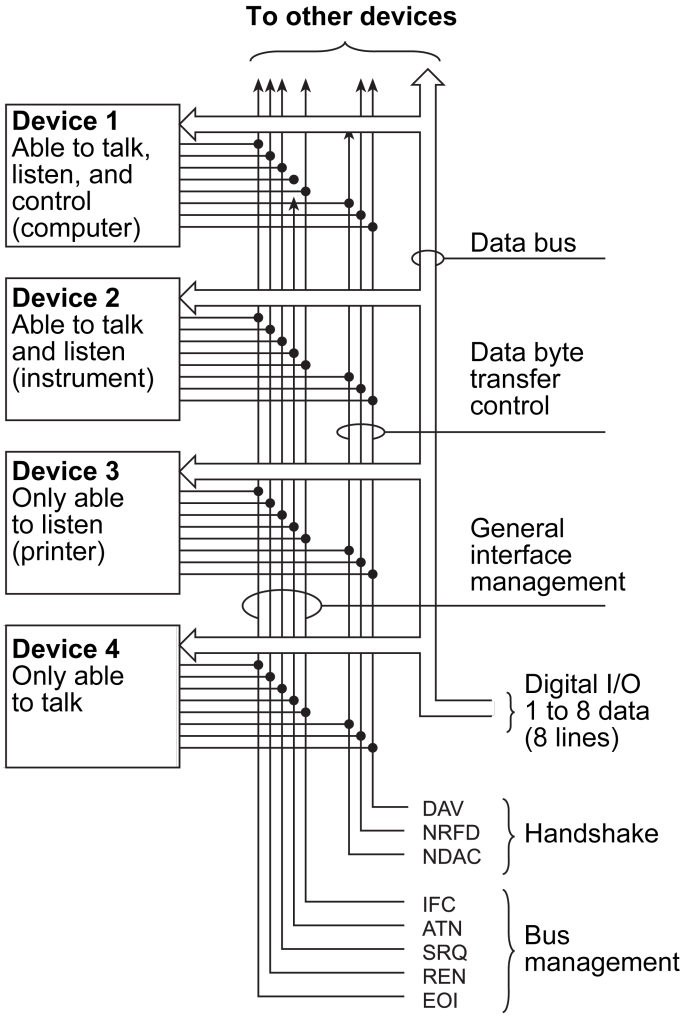
1. The listener indicates that it is ready to listen.
2. The talker places the byte of data on the bus and indicates that the data is available to the listener.
3. The listener, aware that the data is available, accepts the data and indicates that the data has been accepted.
4. The talker, aware that the data has been accepted, stops sending data and indicates that data is not being sent.
5. The listener, aware that there is no data on the bus, indicates that it is ready for the next byte of data.

Bus description

The IEEE-488 bus, which is also referred to a general purpose interface bus (GPIB), is a parallel transfer medium that optimizes data transfer without using an excessive number of bus lines. In keeping with this goal, the bus has only eight data lines that are used for both data and with most commands. Five bus management lines and three handshake lines round out the complement of bus signal lines.

A typical setup for controlled operation is shown in the following figure. Generally, a system contains one controller and a number of other instruments to which the commands are given. Device operation is categorized into three operators: Controller, talker, and listener. The controller controls the instruments on the bus. The talker sends data and a listener receives data. Depending on the type of instrument, any particular device can be a talker only, a listener only, or both a talker and listener.

Figure 72: IEEE-488 bus configuration



There are system controllers and basic controllers. Both can control other instruments, but only the system controller has absolute authority in the system. In a system with more than one controller, only one controller may be active at any given time. Certain protocol is used to pass control from one controller to another.

The IEEE-488 bus is limited to 15 devices, including the controller. Therefore, any number of talkers and listeners up to that limit may be present on the bus at one time. Although several devices may be commanded to listen simultaneously, the bus can have only one active talker.

A device is placed in the talk or listen state by sending an appropriate talk or listen command. These talk and listen commands are derived from the primary address of the instrument. The primary address may have any value between 0 and 31. The actual listen address value sent out over the bus is obtained by performing an OR operation of the primary address with `$20`. For example, if the primary address is `$27`, the actual listen

address is $47 ($47 = $27 + $20). The talk address is obtained by performing an OR operation of the primary address with $40. With the present example, the talk address derived from a primary address of 27 decimal is $67 ($67 = $27 + $40).

The IEEE-488 standards also include another addressing mode called secondary addressing. Secondary addresses are in the range of $60 to $7F. The 2750 does not use secondary addressing.

Once a device is addressed to talk or listen, the appropriate bus transactions take place. For example, if the instrument is addressed to talk, it places its data string on the bus one byte at a time. The controller reads the information and the appropriate software can be used to direct the information to the correct location.

Bus lines

The signal lines on the IEEE-488 bus are grouped into data lines, management lines, and handshake lines. The data lines handle bus data and commands. The management and handshake lines ensure that proper data transfer and operation takes place. Each bus line is active low, with approximately zero volts representing a logic 1 (true). The following topics describe the operation of these lines.

Data lines

The IEEE-488 bus uses eight data lines that transfer data one byte at a time. DIO1 (data input/output) through DIO8 (data input/output) are the eight bidirectional data lines used to transmit both data and multiline commands. The data lines operate with low true logic.

Bus management lines

The bus management lines help to ensure proper interface control and management. These lines send the uniline commands.

ATN (attention): The ATN line is one of the more important management lines because the state of this line determines how information on the data bus is interpreted.

IFC (interface clear): The IFC line controls clearing of instruments from the bus.

REN (remote enable):The REN line places the instrument on the bus in the remote mode.

EOI (end or identify): The EOI is usually used to mark the end of a multibyte data transfer sequence.

SRQ (service request): This line is used by devices when they require service from the controller.

Handshake lines

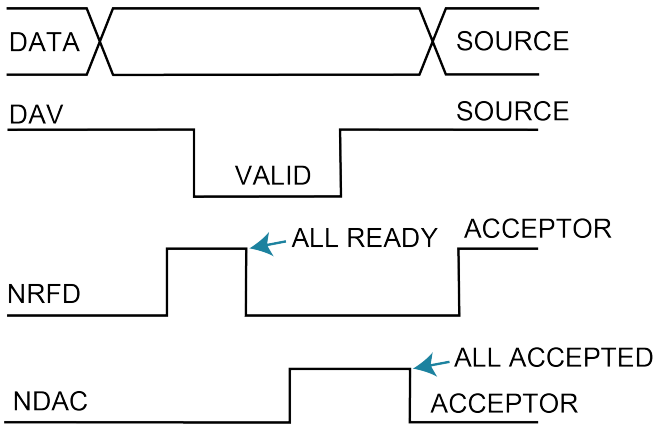
The bus handshake lines operate in an interlocked sequence. This method ensures reliable data transmission regardless of the transfer rate. Generally, data transfer occurs at a rate determined by the slowest active device on the bus.

One of the three handshake lines is controlled by the source (the talker sending information). The other two lines are controlled by accepting devices (the listener or listeners receiving the information). The three handshake lines are:

- DAV (data valid): The source controls the state of the DAV line, which indicates whether data bus information is valid for any listening lines.
- NRFD (not ready for data): The acceptor controls the state of NRFD. It signals the transmitting device to pause the byte transfer sequence until the accepting device is ready.
- NDAC (not data accepted): NDAC is also controlled by the accepting device. The state of NDAC tells the source whether or not the device has accepted the data byte.

The complete handshake sequence for one data byte is shown in the following figure. Once data is placed on the data lines, the source checks to see that NRFD is high, indicating that all active devices are ready. At the same time, NDAC should be low from the previous byte transfer. If these conditions are not met, the source must wait until NDAC and NRFD have the correct status. If the source is a controller, NRFD and NDAC must be stable for at least 100 ns after ATN is set true. Because of the possibility of a bus interruption, many controllers have timeout routines that display messages if the transfer sequence stops for any reason.

Figure 73: IEEE-488 handshake sequence



Once all NDAC and NRFD are properly set, the source sets DAV low, signaling to accepting devices that the byte on the data lines is now valid. NRFD then goes low, and NDAC goes

high once all devices have accepted the data. Each device releases NDAC at its own rate, but NDAC is not released to go high until all devices have accepted the data byte.

The sequence described above is used to transfer data, talk and listen addresses, and multiline commands. The state of the ATN line determines whether the data bus contains data, addresses, or commands.

Bus commands

The instrument may be given a number of special bus commands through the IEEE-488 interface. This section briefly describes the purpose of the bus commands, which are grouped into the following categories:

- Uniline commands: Sent by setting the associated bus lines true. For example, to assert REN (remote enable), the REN line is set low (true).
- Multiline commands: General bus commands that are sent over the data lines with the ATN line true (low).
- Common commands: Commands that are common to all devices on the bus; sent with ATN high (false).
- SCPI commands: Commands that are particular to each device on the bus; sent with ATN (false).

These bus commands and their general purposes are summarized in the following table.

IEEE-488 bus command summary

Command type	Command	State of ATN line	Comments
Uniline	REN (remote enable)	X	Sets up devices for remote operation
	EOI	X	Marks end of transmission
	IFC (interface clear)	X	Clears interface
	ATN (attention)	Low	Defines data bus contents
	SRQ	X	Controlled by external device
Multiline Universal	LLO (local lockout)	Low	Locks out local operation
	DCL (device clear)	Low	Returns device to default conditions
	SPE (serial enable)	Low	Enables serial polling
	SPD (serial poll disable)	Low	Disables serial polling
Addressed	SDC (selective device clear)	Low	Returns instrument to default conditions
	GTL (go to local)	Low	Returns device to local
Unaddressed	UNL (unlisten)	Low	Removes all listeners from the bus
	UNT (untalk)	Low	Removes any talkers from the bus
Common	—	High	Programs IEEE-488.2 compatible instruments for common operations
SCPI	—	High	Programs SCPI compatible instruments for specific operations

Uniline commands

ATN, IFC, and REN are asserted only by the controller. SRQ is asserted by an external device. EOI may be asserted either by the controller or other devices, depending on the direction of data transfer. The following is a description of each command. Each command is sent by setting the corresponding bus line true.

REN (remote enable): REN is sent to set up instruments on the bus for remote operation. When REN is true, devices are removed from the local mode. Depending on device configuration, all front-panel controls except the LOCAL key may be locked out when REN is true. Generally, send REN before attempting to program instruments over the bus.

EOI (end or identify): EOI positively identifies the last byte in a multibyte transfer sequence, allowing data words of various lengths to be transmitted easily.

IFC (interface clear): IFC clears the interface and returns all devices to the talker and listener idle states.

ATN (attention): The controller sends ATN when transmitting addresses or multiline commands.

SRQ (service request): SRQ is asserted by a device when it requires service from a controller.

Universal multiline commands

Universal commands are multiline commands that require no addressing. All devices equipped to implement such commands do so simultaneously when the commands are transmitted. All multiline commands are transmitted with ATN true.

LLO (local lockout): LLO is sent to the instrument to lock out the LOCAL key and all front-panel controls of the instrument.

DCL (device clear): DCL returns instruments to a default state. Usually, instruments return to their power-up conditions.

SPE (serial poll enable): SPE is the first step in the serial polling sequence that determines which device has requested service.

SPD (serial poll disable): SPD is used by the controller to remove all devices on the bus from the serial poll mode and is generally the last command in the serial polling sequence.

Addressed multiline commands

Addressed commands are multiline commands that must be preceded by the device listen address before that instrument responds to the command. Only the addressed device responds to these commands. Both the commands and the address preceding it are sent with ATN true.

SDC (selective device clear): The SDC command performs essentially the same function as the DCL command, except that only the addressed device responds. Generally, instruments return to their power-up default conditions when responding to the SDC command.

GTL (go to local): The GTL command removes instruments from the remote mode. With some instruments, GTL also unlocks front-panel controls if they were previously locked out with the LLO command.

GET (group execute trigger): The GET command triggers devices to perform a specific action that depends on device configuration (for example, make a reading). Although GET is an addressed command, many devices respond to GET without addressing.

Addressed commands

Addressed commands include two primary command groups and a secondary address group. ATN is true when these commands are asserted. The commands include:

- LAG (listen address group): These listen commands are derived from the primary address of the instrument and address devices to listen. The actual command byte is obtained by performing an OR of the primary address with $20.
- TAG (talk address group): The talk commands are derived from the primary address by performing an OR of the address with $40. Talk commands address devices to talk.
- SCG (secondary command group): Commands in this group provide additional addressing capabilities. Many devices (including the 2750) do not use these commands.

Unaddressed commands

The controller uses the unaddressed commands to remove any talkers or listeners from the bus. ATN is true when these commands are asserted.

UNL (unlisten): Listeners are placed in the listener idle state by the UNL command.

UNT (untalk): Any previously commanded talkers are placed in the talker idle state by the UNT command.

Common commands

Common commands are commands that are common to all devices on the bus. These commands are designated and defined by the IEEE-488.2 standard.

Generally, these commands are sent as one or more ASCII characters that tell the device to perform a common operation, such as reset. The IEEE-488 bus treats these commands as data because ATN is false when the commands are transmitted.

SCPI commands

SCPI commands are commands that are particular to each device on the bus. These commands are designated by the instrument manufacturer and are based on the instrument model defined by the Standard Commands for Programmable Instruments (SCPI) Consortium's SCPI standard.

Generally, these commands are sent as one or more ASCII characters that tell the device to perform a particular operation, such as setting a range or closing a relay. The IEEE-488 bus treats these commands as data because ATN is false when the commands are transmitted.

Command codes

Hexadecimal and decimal values for the commands that use the data lines are listed in the following table.

Command	Hexadecimal value	Decimal value
GTL	01	1
SDC	04	4
GET	08	8
LLO	11	17
DCL	14	20
SPE	18	24
SPD	19	25
LAG	20 through 3F	32 through 63
TAG	40 through 5F	64 through 95
SCG	60 through 7F	96 through 127
UNL	3F	63
UNT	5F	95

Command codes for the various commands that use the data lines are summarized in the following figure.

Figure 74: Command codes

D_7 D_6 D_5 D_4	X 0 0 0 0	Command	X 0 0 0 1	Command	X 0 1 0	Primary Address	X 0 1 1	Primary Address	X 1 0 0	Primary Address	X 1 0 1	Primary Address	X 1 1 0		X 1 1 1	
Bits	0 (A)	0 (B)	1 (A)	1 (B)	2 (A)	2 (B)	3 (A)	3 (B)	4 (A)	4 (B)	5 (A)	5 (B)	6 (A)	6 (B)	7 (A)	7 (B)
D_3 →	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
D_2 →	0	0	0	0	1	1	1	1	0	0	1	1	0	0	1	1
D_1 →	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
D_0 →	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
Column → Row ↓																
0	NUL		DLE		SP	0	0	16	@	0	P	16				
1	SOH	GTL	DC1	LLO	!	1	1	17	A	1	Q	17	a		p	
2	STX		DC2		"	2	2	18	B	2	R	18	b		q	
3	ETX		DC3		#	3	3	19	C	3	S	19	c		r	
4	EOT	SDC	DC4	DCL	$	4	4	20	D	4	T	20	d		s	
5	ENQ	PPC*	NAK	PPU*	%	5	5	21	E	5	U	21	e		t	
6	ACK		SYN		&	6	6	22	F	6	V	22	f		u	
7	BEL		ETB		'	7	7	23	G	7	W	23	g		v	
8	BS	GET	CAN	SPE	(	8	8	24	H	8	X	24	h		w	
9	HT	TCT*	EM	SPD	)	9	9	25	I	9	Y	25	i		x	
10	LF		SUB		*	10	:	26	J	10	Z	26	j		y	
11	VT		ESC		+	11	;	27	K	11	[	27	k		z	
12	FF		FS		,	12	<	28	L	12	\	28	l		{	
13	CR		GS		—	13	=	29	M	13	]	29	m		:	
14	SO		RS		.	14	>	30	N	14	^	30	n		}	
15	SI		US		/	15	?	UNL	O	15	_	UNT	o		≅	DEL

ADDRESSED COMMAND GROUP (ACG)

UNIVERSAL COMMAND GROUP (UCG)

LISTEN ADDRESS GROUP (LAG)

TALK ADDRESS GROUP (TAG)

PRIMARY COMMAND GROUP (PCG)

SECONDARY COMMAND GROUP (SDC)

*PPC (PARALLEL POLLCONFIGURE) PPU (PARALLEL POLL UNCONFIGURE), (PCG) and TCT (TAKE CONTROL) not implemented by Model 2750.

Note: D_0 = D101 ...D_7 = D108; X = Don't Care.

Typical command sequences

For the multiline commands, a specific bus sequence must take place to properly send the command. In particular, the correct listen address must be sent to the instrument before it responds to addressed commands. The following table lists a typical bus sequence for sending the addressed multiline commands. In this instance, the SDC command is sent to the instrument. UNL is generally sent as part of the sequence to ensure that no other active listeners are present. ATN is true for both the listen command and the SDC command byte.

Typical addressed command sequence

Step	Command	ATN state	Data bus		
			ASCII	**Hex**	**Decimal**
1	UNL	Set low	?	3F	63
2	LAG*	Stays low	;	3B	59
3	SDC	Stays low	EOT	04	4
4		Returns high			
*Assumes primary address = 27					

The following table gives a typical common command sequence. In this instance, ATN is true while the instrument is being addressed, but it is set high while sending the common command string.

Typical common command sequence

Step	Command	ATN state	Data bus		
			ASCII	**Hex**	**Decimal**
1	UNL	Set low	?	3F	63
2	LAG*	Stays low	;	3B	59
3	Data	Set high	*	2A	42
4	Data	Stays high	R	52	82
5	Data	Stays high	S	53	83
6	Data	Stays high	T	54	84
*Assumes primary address = 27					

IEEE command groups

Command groups supported by the 2750 are listed in the following table. Common commands and SCPI commands are not included in this list.

Handshake command group	
	NDAC = Not data accepted NRFD = Not ready for data DAV = Data valid
Universal command group	
	ATN = Attention DCL = Device clear IFC = Interface clear REN = Remote enable SPD = Serial poll disable SPE = Serial poll enable
Address command group	
LISTEN TALK	LAG = Listen address group MLA = My listen address UNL = Unlisten TAG = Talk address group MTA = My talk address UNT = Untalk OTA = Other talk address
Addressed command group	
	ACG = Addressed command group GTL = Go to local SDC = Selective device clear
Status command group	
	RQS = Request service SRQ = Serial poll request STB = Status byte EOI = End

Interface function codes

The interface function codes, which are part of the IEEE-488 standards, define the ability of an instrument to support interface functions and should not be confused with programming commands found elsewhere in this manual. The interface function codes for the 2750 are listed in the following table.

2750 interface function codes and descriptions

Code	Interface function	Code description
SH1	Source handshake capability	Defines the ability of the instrument to initiate the transfer of message/data over the data bus.
AH1	Acceptor handshake capability	Defines the ability of the instrument to guarantee proper reception of message/data transmitted over the data bus.
T5	Talker	Basic talker, talk-only, serial poll, unaddressed to talk on listen address group (LAG). This function provides the ability of the instrument to send data over the bus to other devices. Instrument talker capabilities (T5) exist only after the instrument has been addressed to talk.
L4	Listener	Basic listener, unaddressed to listen on TAG. This function provides the ability for the instrument to receive device-dependent data over the bus from other devices. Listener capabilities (L4) of the instrument exist only after it has been addressed to listen.
SR1	Service request capability	Defines the ability of the instrument to request service from the controller.
RL1	Remote/local capability	Defines the ability of the instrument to be placed in the remote or local modes.
PP0	No parallel poll capability	The instrument does not have parallel polling capabilities.
DC1	Device clear capability	Defines the ability of the instrument to be cleared (initialized).
DT1	Device trigger capability	Defines the ability of the 2750 to have readings triggered.
C0	No controller capability	The instrument does not have controller capabilities.
E1	Open collector bus drivers	The instrument has open-collector bus drivers.
TE0	No extended talker capability	The instrument does not have extended talker capabilities.
LE0	No extended listener capability	The instrument does not have extended listener capabilities.

Specifications are subject to change without notice.
All Keithley trademarks and trade names are the property of Keithley Instruments.
All other trademarks and trade names are the property of their respective companies.

Keithley Instruments • 28775 Aurora Road • Cleveland, Ohio 44139 • 1-800-833-9200 • tek.com/keithley

KEITHLEY
A Tektronix Company

04/2022