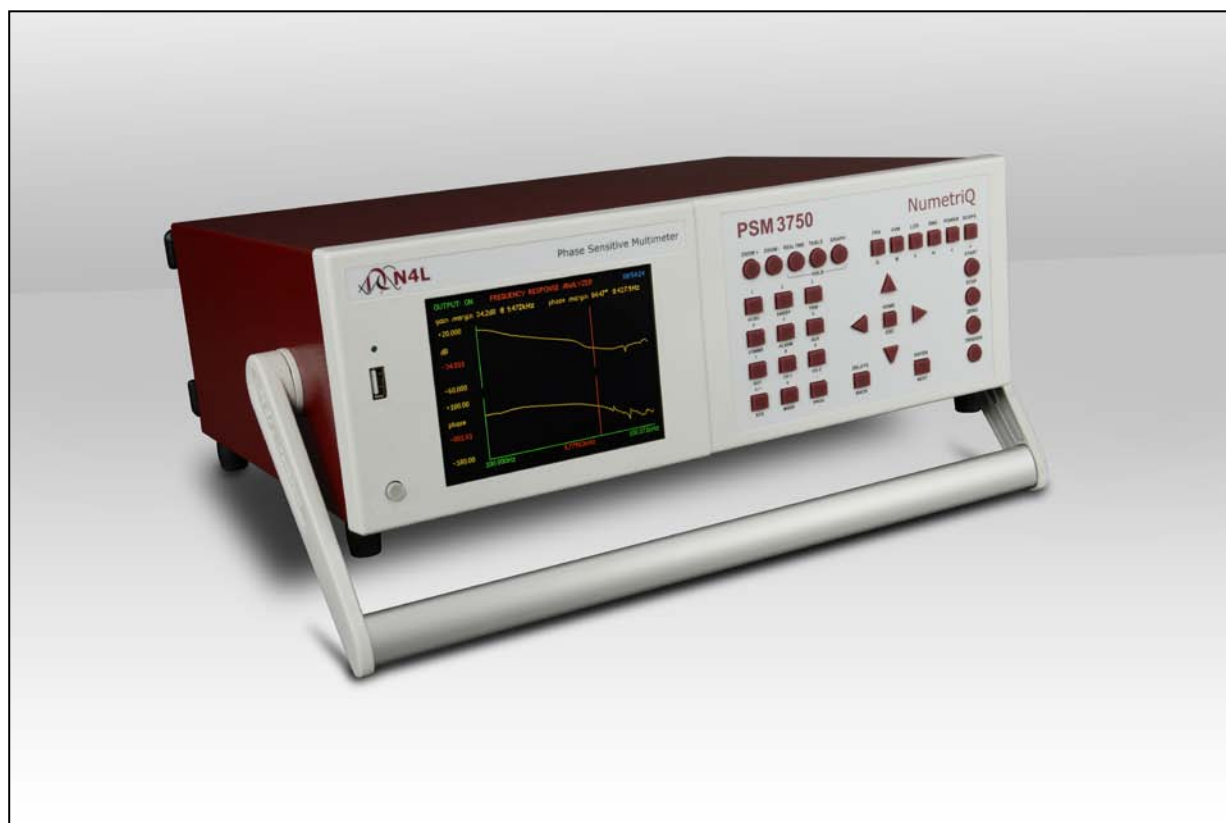




# PSM3750 - *NumetriQ*

## COMMUNICATIONS MANUAL

技术支持：13377795928



*" Do not be hasty when making measurements. "*

**NumetriQ** is a precision instrument that provides you with the tools to make a wide variety of measurements accurately, reliably, and efficiently - but good metrology practice must be observed. Take time to read this manual and familiarise yourself with the features of the instrument in order to use it most effectively.

**DANGER OF ELECTRIC SHOCK**

Only qualified personnel should install this equipment, after reading and understanding this user manual. If in doubt, consult your supplier.

**RISQUE D'ELECTROCUTION**

L'installation de cet équipement ne doit être confiée qu'à un personnel qualifié ayant lu et compris le présent manuel d'utilisation. Dans le doute, s'adresser au fournisseur.

**GEFAHR VON ELEKTRISCHEM SCHOCK**

Nur entsprechend ausgebildetes Personal ist berechtigt, diese Ausrüstung nach dem Lesen und Verständnis dieses Anwendungshandbuches zu installieren. Falls Sie Zweifel haben sollten, wenden Sie sich bitte an Ihren Lieferanten.

**RISCHIO DI SCARICHE ELETTRICHE**

Solo personale qualificato può installare questo strumento, dopo la lettura e la comprensione di questo manuale. Se esistono dubbiconsultate il vostro rivenditore.

**PELIGRO DE DESCARGA ELÉCTRICA**

Solo personal cualificado debe instalar este instrumento, después de la lectura y comprensión de este manual de usuario. En caso de duda, consultar con su suministrador.

## **IMPORTANT SAFETY INSTRUCTIONS**

This equipment is designed to comply with BSEN 61010-1 (Safety requirements for electrical equipment for measurement, control, and laboratory use) – observe the following precautions:

- This appliance ***must*** be earthed. Ensure that the instrument is powered from a properly grounded supply.
- The input connectors are High Voltage safety types for use up to 500V peak input from earth, overvoltage category II. Do not exceed 500V peak on any input connection. Only use test leads that are fitted with approved High Voltage safety connectors when working with hazardous voltages.
- The inputs must not be connected to signals greater than is indicated on the front panel.
- Keep the ventilation holes on the underneath and sides free from obstruction.
- Do not operate or store under conditions where condensation may occur or where conducting debris may enter the case.
- There are no user serviceable parts inside the instrument – do not attempt to open the instrument, refer service to the manufacturer or his appointed agent.

**Note: Newtons4th Ltd. shall not be liable for any consequential damages, losses, costs or expenses arising from the use or misuse of this product however caused.**

## **ABOUT THIS MANUAL**

This manual gives details of the communication commands recognized by the PSM3750 instrument over RS232, USB, LAN or GPIB. For more general operating instructions for the instrument refer to the specific user manual.

Each command is listed alphabetically with details of any arguments and reply. A one line summary of each command is given in the appendix. Although most of the commands apply to all instruments in the range there are some commands that are specific to one instrument or another.

The information in this manual is believed to be accurate and complete but Newtons4th Ltd cannot accept any liability whatsoever for any consequential damage or losses arising from any errors, inaccuracies, or omissions.

Revision 1.12

This manual is copyright © 2012 - 2013 Newtons4th Ltd. and all rights are reserved. No part may be copied or reproduced in any form without prior written consent.

10 September 2013

## CONTENTS

1	Using remote control .....	1-1
1.1	Standard event status register.....	1-3
1.2	Serial Poll status byte .....	1-4
1.3	RS232 connections.....	1-5
2	Communication commands .....	2-6
	*CLS .....	2-6
	*ESE .....	2-7
	*ESR?.....	2-8
	*IDN?.....	2-9
	*OPC? .....	2-10
	*RST .....	2-11
	*SRE .....	2-12
	*STB?.....	2-14
	*TRG .....	2-15
	*TST?.....	2-16
	*WAI .....	2-17
	ABORT .....	2-18
	ACTRIM .....	2-19
	ALARM .....	2-20
	ALARM1 .....	2-22
	ALARM2 .....	2-23
	ALARME .....	2-24
	AMPLIT .....	2-26
	AUXILI .....	2-27
	BANDWI .....	2-28
	BEEP .....	2-29
	BLANKI .....	2-30
	CONFIG .....	2-31
	COUPLI .....	2-33
	CYCLES.....	2-34
	DATALO .....	2-35
	DAV? .....	2-37
	DAVER.....	2-38
	DELAY .....	2-40
	FAST .....	2-41
	FILTER.....	2-42
	FRA .....	2-43
	FREQUE .....	2-45
	FSWEEP .....	2-46

## PSM3750 communications manual

GAINPH .....	2-47
HARMON .....	2-49
HOLD .....	2-51
INPUT .....	2-52
KEYBOA .....	2-53
LCR .....	2-54
LOWFRE .....	2-56
MARKER .....	2-57
MODE .....	2-58
MULTIL .....	2-59
MULTIL? .....	2-60
NEWLOC .....	2-61
NOOVER .....	2-62
PFCONV .....	2-63
OFFSET .....	2-64
OUTPUT .....	2-65
PAV .....	2-66
PHASEM .....	2-68
PHCONV .....	2-70
PHREF .....	2-71
POWER .....	2-72
PROGRA .....	2-74
RANGE .....	2-76
RESOLU .....	2-77
RESULT .....	2-78
REZERO .....	2-80
SCALE .....	2-81
SCOPE? .....	2-82
SCREEN? .....	2-83
SETUP .....	2-84
SETUP? .....	2-85
SHUNT .....	2-86
SMOOTH .....	2-87
SPEED .....	2-88
START .....	2-89
STATUS? .....	2-90
STOP .....	2-91
SUSPEND .....	2-92
TAGREP .....	2-93
TFA .....	2-94
USER? .....	2-96
VARCON .....	2-97
VECTOR .....	2-98
VERSIO? .....	2-100

## PSM3750 communications manual

VRMS .....	2-101
WAVEFO .....	2-103
WIRING .....	2-104
ZERO.....	2-105
ZOOM.....	2-106





## 1 Using remote control

The instrument is fitted with an RS232 serial communications port as standard, and may have an IEEE488 (GPIB) interface or LAN interface fitted as an option. All the interfaces use the same ASCII protocol with the exception of the end of line terminators:

	Rx expects	Tx sends
RS232, LAN, USB	carriage return (line feed ignored)	carriage return and line feed
IEEE488	carriage return or line feed or EOI	carriage return with EOI

All the functions of the instrument can be programmed via either interface, and results read back. When the IEEE488 interface is set to 'remote' the RS232 port is ignored.

The commands are not case sensitive and white space characters are ignored (e.g. tabs and spaces). Replies from the instrument are always upper case, delimited by commas, without spaces.

Only the first six characters of any command are important – any further characters will be ignored. For example, the command to set the generator frequency is FREQUE but the full word FREQUENCY may be sent as the redundant NCY at the end will be ignored.

Fields within a command are delimited by comma, multiple commands can be sent on one line delimited with a semi-colon. Eg.

AMPLIT,1.5;OUTPUT,ON

Mandatory commands specified in the IEEE488.2 protocol have been implemented, (e.g. \*IDN?, \*RST) and all commands that expect a reply (query commands) are terminated with a question mark.

The instrument maintains an error status byte consistent with the requirements of the IEEE488.2 protocol (called the standard event status register) that can be read by the mandatory command \*ESR? (see section 5.1).

The instrument also maintains a status byte consistent with the requirements of the IEEE488.2 protocol, that can be read either with the IEEE488 serial poll function or by the mandatory command \*STB? over RS232 or IEEE or LAN (see section 5.2).

The IEEE address defaults to 23 and can be changed via the COMMS menu.

The keyboard is disabled when the instrument is set to "remote" using the IEEE. Press HOME to return to "local" operation.

RS232 data format is: start bit, 8 data bits (no parity), 1 stop bit. Flow control is RTS/CTS (see section 5.2), baud rate is selectable via the COMMS menu.

A summary of the available commands is given in the Appendix. Details of each command are given in the communication command section of the manual.

Commands are executed in sequence except for two special characters that are immediately obeyed:

- Control T (20) – reset interface (device clear)
- Control U (21) – warm restart

## 1.1 Standard event status register

PON		CME	EXE	DDE	QYE		OPC
-----	--	-----	-----	-----	-----	--	-----

- bit 0 OPC (operation complete)  
cleared by most commands  
set when data available or sweep complete
- bit 2 QYE (unterminated query error)  
set if no message ready when data read
- bit 3 DDE (device dependent error)  
set when the instrument has an error
- bit 4 EXE (execution error)  
set when the command cannot be executed
- bit 5 CME (command interpretation error)  
set when a command has not been recognised
- bit 7 PON (power on event)  
set when power first applied or unit has reset

The bits in the standard event status register except for OPC are set by the relevant event and cleared by specific command (\*ESR?, \*CLS, \*RST). OPC is also cleared by most commands that change any part of the configuration of the instrument (such as MODE or START).

## 1.2 Serial Poll status byte

		ESB	MAV	ALM		SDV	RDV
--	--	-----	-----	-----	--	-----	-----

- bit 0 RDV (result data available)  
set when results are available to be read as enabled by DAVER
- bit 1 SDV (sweep data available)  
set when sweep results are available to be read as enabled by DAVER
- bit 3 ALA (alarm active)  
set when an alarm is active and enabled by ALARMER
- bit 4 MAV (message available)  
set when a message reply is waiting to be read
- bit 5 ESB (standard event summary bit)  
set if any bit in the standard event status register is set as well as the corresponding bit in the standard event status enable register (set by \*ESE).

## 1.3 RS232 connections

The RS232 port on the instrument uses the same pinout as a standard 9 pin serial port on a PC or laptop (9-pin male 'D' type).

Pin	Function	Direction
1	DCD	in (+ weak pull up)
2	RX data	in
3	TX data	out
4	DTR	out
5	GND	
6	DSR	not used
7	RTS	out
8	CTS	in
9	RI	not used

The instrument will only transmit when CTS (pin 8) is asserted, and can only receive if DCD (pin 1) is asserted. The instrument constantly asserts (+12V) DTR (pin 4) so this pin can be connected to any unwanted modem control inputs to force operation without handshaking. The instrument has a weak pull up on pin 1 as many null modem cables leave it open circuit. In electrically noisy environments, this pin should be driven or connected to pin 4.

To connect the instrument to a PC, use a 9 pin female to 9 pin female null modem cable:

1 & 6	-	4
2	-	3
3	-	2
4	-	1 & 6
5	-	5
7	-	8
8	-	7

## 2 Communication commands

**\*CLS**

**\*CLS**

Function: Clear status

Description: Clears the *standard event status register*.

Format: \*CLS

Arguments: none

Reply: none

Example: \*CLS  
\*ESR?  
0

Notes:

**\*ESE**

**\*ESE**

Function: Set standard event status enable register.

Description: Enable which bits of the *standard event status register* set the ESB bit in the serial poll status byte..

Format: \*ESE, value

Arguments: decimal equivalent of bits in standard event status enable register

Reply: can be read by \*ESE?

Example: \*ESE, 60

Notes: The following bits in the standard event status enable register have been implemented:

bit 0 OPC (operation complete)  
 bit 2 QYE (unterminated query error)  
 bit 3 DDE (device dependent error)  
 bit 4 EXE (execution error)  
 bit 5 CME (command interpretation error)  
 bit 7 PON (power on event)

For example, \*ESE, 60 enables all the error bits so that the ESB bit in the serial poll status byte is set in the event of any error.



**\*ESR?**

**\*ESR?**

Function:	Standard event status register query
Description:	Returns the contents of the <i>standard event status register</i> and clears it.
Format:	*ESR?
Arguments:	none
Reply:	decimal equivalent of bits in standard event status register
Example:	*ESR? 33
Notes:	<p>The following bits in the standard event status register have been implemented:</p> <ul style="list-style-type: none"> <li>bit 0 OPC (operation complete)</li> <li>bit 2 QYE (unterminated query error)</li> <li>bit 3 DDE (device dependent error)</li> <li>bit 4 EXE (execution error)</li> <li>bit 5 CME (command interpretation error)</li> <li>bit 7 PON (power on event)</li> </ul> <p>For example, if a command is sent incorrectly and is not recognised, the CME bit will be set and the value of 33 will be returned.</p>

**\*IDN?**

**\*IDN?**

Function:	Identify query
Description:	Returns a standard format identification string.
Format:	*IDN?
Arguments:	none
Reply:	An ASCII string in the IEEE488.2 format: manufacturer,model,serial no,version
Example:	*IDN? NEWTONS4TH,PSM3750,01234,1.00
Notes:	

**\*OPC?**

**\*OPC?**

Function:	Test for operation complete
Description:	Returns 1 if previous operation is completed, 0 if not.
Format:	*OPC?
Arguments:	none
Reply:	0 or 1
Example:	START *OPC? 0 *OPC? 0 *OPC? 1
Notes:	*OPC? can be used to indicate when data is available or when a frequency sweep has completed.

**\*RST**

**\*RST**

Function:	Reset
Description:	Resets the instrument to the default state and clears the <i>standard event status register</i> .
Format:	*RST
Arguments:	none
Reply:	none
Example:	*RST
Notes:	The *RST command loads the default configuration. This is the same as loading the default configuration via the PROGRAM menu.

Any preceding setup commands will be overwritten.

\*RST should be followed by an end of line not a message separator. It may be helpful to follow it with a short pause to allow the new configuration to become active before sending further commands.

**\*SRE**

**\*SRE**

Function: Set service request enable register.

Description: Enable which bits of the *status byte register* initiate a service request.

Format: \*SRE, value

Arguments: decimal equivalent of bits in status byte register

Reply: can be read by \*SRE?

Example: \*SRE, 1  
generate a service request when data available.

Notes:

**\*SRE?**

**\*SRE?**

Function: Read service request enable register.

Description: Read back the present setting of the service request enable register.

Format: \*SRE?

Arguments:

Reply: decimal equivalent of bits in status byte register that would generate a service request.

Example: \*SRE?  
1

Notes:

**\*STB?**

**\*STB?**

Function: Read serial poll status byte

Description: Returns the decimal value of the serial poll status byte.

Format: \*STB?

Arguments: none

Reply: decimal value of the serial poll status byte

Example: \*STB?  
1

Notes: The following bits in the serial poll status register have been implemented:

- bit 0 RDV (results data available)
- bit 1 SDV (sweep data available)
- bit 3 ALA (alarm active)
- bit 4 MAV (message available)
- bit 5 ESB (standard event summary bit)

**\*TRG**

**\*TRG**

Function:	Trigger
Description:	Initiates a new measurement, resets the ranging and filtering.
Format:	*TRG
Arguments:	none
Reply:	none
Example:	MODE,VRMS *TRG VRMS?
Notes:	



**\*TST?**

**\*TST?**

Function: Self test query

Description: Returns the results of self test

Format: \*TST?

Arguments: none

Reply: single integer  
bit 0 – set if uncalibrated  
bit 1 – set if error with analogue zero  
> 15 – major system error

Example: \*TST?  
0

Notes:

**\*WAI**

**\*WAI**

Function:	Wait for operation complete
Description:	Suspends communication until the previous operation has completed
Format:	*WAI
Arguments:	none
Reply:	none
Example:	GAINPH START *WAI GAINPH,SWEEP?
Notes:	In the example, the query command GAINPH,SWEEP? can be sent immediately after the *WAI command and the sweep data will be returned as soon as the sweep has completed.

## **ABORT**

## **ABORT**

Function: Abort sweep

Description: Abort an active sweep

Format: ABORT

Arguments: none

Reply: none

Example: FSWEEP,50,1000,1e6  
OUTPUT,ON  
START  
ABORT

Notes:

## ACTRIM

## ACTRIM

Function:	Set ac control parameters
Description:	Sets the specified signal level, tolerance and input channel. for the ac control (amplitude compression).
Format:	<i>ACTRIM,channel,level,tolerance</i>
Arguments:	channel: DISABL CH1 CH2 CH3 level: required ac level in V or A or dBm tolerance: required accuracy in percent
Reply:	none
Example:	ACTRIM,CH1,1.0,5        (1.0V, 5%)
Notes:	The level should be set in dBm if dBm mode is selected (OUTPUT,DBM)  It is not necessary to send all the arguments but those that are sent must be in the correct sequence.

## ALARM

## ALARM

Function: Set common controls for alarm1 and alarm2.

Description: Set the alarm latch and sounder control.

Format: *ALARM,latch,sounder*

Arguments: latch:  
              ON  
              OFF  
          sounder:  
              ENABLED  
              DISABLED

Reply: none

Example: ALARM,ON,DISABLED

Notes:

## ALARM?

## ALARM?

Function: Read alarm status.

Description: Reads the status of the measurements and 2 alarms.

Format: ALARM?

Arguments: none

Reply: single integer  
           bit 0 data available  
           bit 1 data error  
           bit 2 alarm 1  
           bit 3 alarm 2

Example: ALARM?  
           1

Notes: An alarm is present if bit 0 is high (data is available) and either alarm 1 or alarm 2 bits are high.

## ALARM1

## ALARM1

Function:	Set parameters for alarm1.
Description:	Set alarm1 type and thresholds.
Format:	ALARM1,DISABLED ALARM1, <i>type,data,high,low</i>
Arguments:	type: HIGH LOW INSIDE OUTSIDE LINEAR data 1-4 high: high threshold low: low threshold
Reply:	None
Example:	ALARM1,HIGH,1,2,0 ALARM1,DISABLED
Notes:	DISABLED does not have any further arguments otherwise both thresholds must be sent even if only one is used.

## ALARM2

## ALARM2

Function:	Set parameters for alarm2.
Description:	Set alarm2 type and thresholds.
Format:	ALARM2,DISABLED ALARM2, <i>type,data,high,low</i>
Arguments:	type: HIGH LOW INSIDE OUTSIDE data 1-4 for zoom data high: high threshold low: low threshold
Reply:	None
Example:	ALARM2,LOW,3,0,0.5
Notes:	DISABLED does not have any further arguments otherwise both thresholds must be sent even if only one is used. There is no LINEAR option for alarm 2.



## ALARME

## ALARME

Function:	Set alarm status enable register
Description:	Sets bits in the alarm status enable register to control which alarm bit if any set the alarm active bits in the status byte.
Format:	ALARME, <i>value</i>
Arguments:	decimal equivalent of alarm bits bit2 set bit 3 of status byte when alarm 1 is active bit3 set bit 3 of status byte when alarm 2 is active
Reply:	none
Example:	ALARME, 12 *SRE,8 set bit 3 in status byte when either alarm 1 or alarm 2 is active and generate a service request
Notes:	default value is 0

**ALARME?**

**ALARME?**

Function: Read alarm status enable register

Description: Read back present bits in the alarm status enable register which controls the alarm active bit in the status byte.

Format: ALARME?

Arguments: none

Reply: decimal equivalent of alarm bits

Example: ALARME?  
12

Notes:

## AMPLIT

## AMPLIT

Function:	Set output amplitude
Description:	Sets the output amplitude in Volts or dBm for the generator.
Format:	<i>AMPLIT,amplitude</i>
Arguments:	peak amplitude in Volts or amplitude in dBm
Reply:	none
Example:	AMPLIT,0.5 (set peak amplitude to 0.5V)
Notes:	dBm mode is selected by OUTPUT,DBM

## AUXILI

## AUXILI

Function:	Controls the auxiliary device
Description:	Controls the auxiliary device connected to the extension port.
Format:	<i>AUXILI,device,value</i>
Arguments:	device: IAI value: LOW NORMAL HIGH VHIGH
Reply:	None
Example:	AUXILI,IAI,HIGH
Notes:	Arguments will be extended to include other auxiliary devices as they become available

## **BANDWI**

## **BANDWI**

**Function:** Select bandwidth or selective (heterodyning) measurements.

**Description:** Selective measurement automatically starts at around 10kHz for those functions that support it. It can be disabled by forcing the bandwidth to "wide". For low noise measurements at low frequency the bandwidth can be restricted to "low".

**Format:** BANDWI, *type*

**Arguments:** type:  
                   AUTO  
                   WIDE  
                   LOW

**Reply:** none

**Example:** BANDWI,WIDE

**Notes:** In wide bandwidth mode the frequency range is limited to 5MHz.  
 In low bandwidth mode, the frequency is restricted to 30kHz

## **BEEP**

## **BEEP**

Function: Sound the buzzer

Description: Makes a “beep” from the instrument.

Format: BEEP

Arguments: none

Reply: none

Example: BEEP

Notes:

## **BLANKI**

## **BLANKI**

Function: Select blanking

Description: Enable or disable low value blanking.

Format: BLANKI,*value*

Arguments: value:  
            ON  
            OFF

Reply: none

Example: BLANKI,OFF

Notes:

## CONFIG

## CONFIG

Function:	Direct access of configuration parameters
Description:	Sets configuration parameter for which there may not be a direct command.
Format:	CONFIG, <i>index,data</i>
Arguments:	index is the number of the parameter data is the data for that parameter
Reply:	none
Example:	CONFIG,6,1    (set phase convention)
Notes:	The list of configurable parameters is given in the appendix. CONFIG goes through the same limit checking as when entering data from the menus.



## CONFIG?

## CONFIG?

Function:	Configurable parameter query
Description:	Reads the present value of a single parameter.
Format:	CONFIG, <i>index</i> ?
or:	CONFIG? <i>index</i>
Arguments:	<i>index</i> is the parameter number
Reply:	Value of parameter, real or integer as appropriate.
Example:	CONFIG,6?    (read phase convention) 0 CONFIG,6,1 CONFIG,6? 1
Notes:	The list of configurable parameters is given in the appendix.

## COUPLI

## COUPLI

Function: Set ac or dc coupling.

Description: Selects the input coupling for a given input channel.

Format: COUPLI,*channel,coupling*

Arguments: channel:  
                   CH1  
                   CH2  
           coupling:  
                   AC+DC  
                   ACONLY

Reply: none

Example: COUPLI,CH2,AC+DC

Notes:

## CYCLES

## CYCLES

Function: Set the minimum number of cycles for a measurement.

Description: The measurement window is normally set according to a time value but subject to a whole cycle of the frequency. Setting a minimum number of cycles to a value greater than 1 extends the measurement window at frequencies where the periodic time is longer than the set window time.

Format: CYCLES,*cycles*

Arguments: minimum number of cycles

Reply: none

Example: CYCLES,4

Notes:

## DATALO

## DATALO

Function: Set up datalog

Description: Sets datalog parameters or accesses datalog non-volatile store.

Format: *DATALO,function,interval*

Arguments: function:  
                   DISABLE  
                   RAM  
                   NONVOL  
                   RECALL  
                   DELETE  
           interval:  
                   datalog interval in seconds

Reply: none

Example: *DATALOG,NONVOL,10*

Notes:

## DATALO?

## DATALO?

Function:	Read back datalog results
Description:	Return datalog values, one record per line
Format:	DATALO, <i>start,records</i>
Arguments:	start: first record to return records: number of records to return
Reply:	3 to 6 data values depending on settings: index 1-n elapsed time in hours data1 data2 (if stored) data3 (if stored) data4 (if stored) one record per line
Example:	DATALOG,NONVOL,36 START wait for datalog STOP DATALOG,20,4? 20,1.9000E-1,1.2345E0 21,2.0000E-1,1.2345E0 22,2.1000E-1,5.6789E3 23,2.2000E-1,1.2345E0
Notes:	if no arguments are sent then DATALOG? returns all data in the same format

## DAV?

## DAV?

Function:	Data available query
Description:	Returns data availability status.
Format:	DAV?
Arguments:	none
Reply:	Decimal equivalent of data available bits: bit0 new data available bit1 data available bit2 new full sweep data available bit3 sweep data available bit5 harmonic data available bit6 integration data available bit7 datalog data available
Example:	START (trigger sweep) DAV? 0 DAV? 11 (first data available) DAV? 11 DAV? 11 DAV? 15 (full sweep data available)
Notes:	DAV? does not modify the status bits.

## DAVER

## DAVER

Function:	Set data available enable register
Description:	Sets bits in the data available enable register to control which status bits set the data available bits in the status byte.
Format:	DAVER,value
Arguments:	decimal equivalent of data available bits bit0 set bit 0 of status byte when new data available bit1 set bit 0 of status byte when data available bit2 set bit 1 of status byte when new full sweep data available bit3 set bit 1 of status byte when sweep data available
Reply:	none
Example:	DAVER, 4 set bit 1 in status byte only when full sweep data is ready
Notes:	default value is 6: bit 0 of status byte is set whenever data is available bit 1 of status byte is set when full sweep data is available.

**DAVER?**

**DAVER?**

Function: Read data available enable register

Description: Read back present setting of the data available enable register, which controls the status bits that set the data available bits in the status byte.

Format: DAVER?

Arguments: none

Reply: decimal equivalent of bits

Example: DAVER?  
4

Notes:



## DELAY

## DELAY

Function: Set a delay time between frequency points

Description: Applies a settling time when changing frequency for systems which need some settling time after the frequency changes before a measurement should be made.

Format: DELAY, *time*

Arguments: delay time in seconds from 1 to 60

Reply: none

Example: DELAY,1

Notes: Whole seconds only

## **FAST**

## **FAST**

Function: Set fast communications mode.

Description: Disables the screen drawing for high speed operation.

Format: FAST, *value*

Arguments: value:  
              ON  
              OFF

Reply: none

Example: FAST, ON

Notes: FAST mode does not suppress the data acquisition which continues in the background. See SUSPEND to disable all non-communication functions.

## **FILTER**

## **FILTER**

Function:	Select the filtering
Description:	Sets the filter time constant and dynamic response.
Format:	<code>FILTER, <i>type</i>, <i>dynamics</i></code>
Arguments:	type: NONE NORMAL SLOW dynamics: AUTO FIXED
Reply:	none
Example:	<code>FILTER,NORMAL,FIXED</code> <code>FILTER,NONE</code>
Notes:	It is not necessary to send both parameters if it is only required to set the type. Both arguments must be sent to set the dynamics.

## **FRA**

## **FRA**

Function: Set frequency response analyser mode.

Description: Set frequency response analyser mode.

Format: FRA

Arguments:

Reply: none

Example: FRA

Notes: This command has the same effect as  
MODE,GAINPH.  
FRA, GAINPH, TFA are aliases for the  
same command.

## FRA?

## FRA?

Function:	frequency response analyser query
Description:	Read frequency response analyser results. Sets frequency response analyser mode if not already set. Waits for next unread data if necessary. Clears new data available bit read by DAV?
Format:	FRA?
or:	FRA,SWEEP?
Arguments:	none, or SWEEP
Reply:	6 data values separated by commas freq,mag1,mag2,db,phase,delay + 4 values if 3 channels in use mag3,db3,phase3,delay3 one line per result for sweep data
Example:	OUTPUT,ON FRA FSWEEP,20,10,20E3 START DAV? 3 DAV? 15 FRA?SWEEP data returned
Notes:	FRA? waits for next unread data. FRA?SWEEP does not wait for new data. FRA, GAINPH, TFA are aliases for the same command

## **FREQUE**

## **FREQUE**

Function: Set the output frequency

Description: Sets the generator output frequency in Hz.

Format: `FREQUE, frequency`

Arguments: frequency in Hz

Reply: none

Example: `FREQUE,5e4` (set frequency to 50kHz)

Notes:

## FSWEEP

## FSWEEP

Function:	Set the frequency sweep parameters
Description:	Sets the start frequency in Hz, the end frequency, the number of steps and log/linear for the selected function.
Format:	<i>FSWEEP,steps,start,end,type</i>
Arguments:	steps: number of steps start: start frequency in Hz end: end frequency in Hz type: LOGARI LINEAR
Reply:	none
Example:	MODE,GAINPH FSWEEP,50,1000,1e6 (set 50 steps between 1kHz and 1MHz)
Notes:	It is not necessary to send all the arguments, but if they must be in the specified order. The action at the end of the sweep is specified in the OUTPUT command.

## **GAINPH**

## **GAINPH**

Function: Set gain/phase analyser mode.

Description: Set gain/phase analyser mode.

Format: GAINPH

Arguments:

Reply: none

Example: GAINPH

Notes: This command has the same effect as  
MODE,GAINPH.  
FRA, GAINPH, TFA are aliases for the  
same command.



## **GAINPH?**

## **GAINPH?**

Function: Gain/phase query

Description: Read gain/phase analyser results.

Format: GAINPH?  
or: GAINPH,SWEEP?

Arguments: none, or SWEEP

Reply: See FRA?

Example: OUTPUT,ON  
GAINPH  
FSWEEP,20,10,20E3  
START  
DAV?  
3  
DAV?  
15  
GAINPH?SWEEP  
data returned

Notes: GAINPH? is the same as FRA?

## HARMON

## HARMON

Function:	Set harmonic analyser mode.
Description:	Set harmonic analyser mode and parameters.
Format:	<i>HARMON,scan,parameter,harmonic,max</i>
Arguments:	scan: SINGLE THDD THDS parameter: PERCEN DB harmonic: single harmonic 2-100 for display max: harmonic series 2-100 for series thd
Reply:	none
Example:	HARMON,SINGLE,PERCEN,3
Notes:	It is not necessary to send any arguments, but if any are sent they must be in the specified order.

## HARMON?

## HARMON?

Function:	Harmonic analyser query
Description:	Read harmonic results. Sets harmonic analyser mode if not already set. Waits for next unread data if necessary. Clears new data available bit read by DAV?
Format:	HARMON?
or:	HARMON,SWEEP?
or:	HARMON,SERIES?
Arguments:	none, or SWEEP, or SERIES
Reply:	7 data values separated by commas:
single:	freq,mag1,mag2,h1,h2,harm1,harm2
thd:	freq,mag1,mag2,thd1,thd2,harm1,harm2 1 line per result for sweep data
series:	6 data values separated by commas: mag1,%1,phase1,mag2,%2,phase2
Example:	HARMON? data returned
Notes:	HARMON? waits for next unread data. HARMON?SWEEP does not wait for new data – data can be read multiple times.

## **HOLD**

## **HOLD**

Function: Set/clear HOLD mode

Description: HOLD mode stops the instrument from updating the measured values

Format: HOLD, *value*

Arguments: value:  
ON  
OFF

Reply: none

Example: HOLD,ON

Notes:

## INPUT

## INPUT

Function:	Set input mode
Description:	Selects the input type of the instrument
Format:	INPUT, <i>channel,type</i>
Arguments:	channel: CH1 CH2 CH3 type: VOLTAGE CURRENT
Reply:	None
Example:	INPUT,CH1,CURRENT SHUNT,CH1,0.1
Notes:	Current input needs an external shunt

## KEYBOA

Function: Disable front panel keyboard.

Description: The front panel keyboard can be disabled to prevent accidental operation.

Format: KEYBOARD, *value*

Arguments: value:  
            ENABLE  
            DISABLE

Reply: none

Example: KEYBOARD,DISABLE

Notes: The keyboard can be re-enabled from the front panel only by pressing the HOME key.

## KEYBOA

## LCR

## LCR

Function:	Set LCR meter mode.
Description:	Set LCR mode and conditions.
Format:	<i>LCR,conditions,parameter,head</i>
Arguments:	conditions: AUTO MANUAL parameter: AUTO CAPACITANCE INDUCTANCE IMPEDANCE ADMITTANCE head: NONE LOW (only valid for IAI) NORMAL HIGH VHIGH
Reply:	none
Example:	LCR,AUTO,IMPEDA,NORMAL
Notes:	It is not necessary to send any arguments, but if any are sent they must be in the specified order.

## LCR?

## LCR?

Function:	LCR meter query
Description:	Read LCR meter results. Sets LCR meter mode if not already set. Waits for next unread data if necessary. Clears new data available bit read by DAV?
Format:	LCR?
or:	LCR?SWEEP
or:	LCR,SWEEP?
Arguments:	none, or SWEEP
Reply:	14 data values separated by commas: freq, mag1, mag2, impedance, phase, series R, series C, series L, //R, //C, //L, $\tan\delta$ , Q, reactance
or	11 data values separated by commas: freq, mag1, mag2, impedance, phase, resistance, reactance, admittance, phase, conductance, susceptance
sweep reply:	8 data values per line per sweep result: freq,Q, $\tan\delta$ ,impedance,phase,L,C,R
Example:	OUTPUT,ON LCR? data returned
Notes:	LCR? waits for next unread data. LCR?SWEEP does not wait for new data – data can be read multiple times.



## LOWFRE

## LOWFRE

Function:	Set low frequency mode
Description:	Sets the low frequency option for external frequency measurement.
Format:	LOWFRE, <i>value</i>
Arguments:	value: ON OFF
Reply:	none
Example:	LOWFRE,ON
Notes:	LOWFRE is mainly used for measuring low frequencies when not using the instrument generator for the frequency reference. However, as it applies digital filtering, it may also be useful when analysing any signals below a few hundred Hertz.

## MARKER

## MARKER

Function:	Set frequency marker
Description:	Enable or disable frequency marker.
Format:	MARKER, <i>value</i> , <i>frequency</i>
Arguments:	value: ON OFF frequency: marker frequency in Hz
Reply:	none
Example:	MARKER,OFF MARKER,ON,25e3
Notes:	It is not necessary to send the frequency when enabling the marker if it has already been set.

## MODE

## MODE

Function:	Set mode														
Description:	Sets the fundamental operating mode of the instrument.														
Format:	MODE, <i>type</i>														
Arguments:	type: <table> <tr> <td>VRMS</td><td>(rms voltmeter)</td></tr> <tr> <td>GAINPH</td><td>(gain/phase analyser)</td></tr> <tr> <td>VECTOR</td><td>(vector voltmeter)</td></tr> <tr> <td>POWER</td><td>(power meter)</td></tr> <tr> <td>LCR</td><td>(LCR meter)</td></tr> <tr> <td>HARMON</td><td>(harmonic analyser)</td></tr> <tr> <td>SCOPE</td><td>(oscilloscope)</td></tr> </table>	VRMS	(rms voltmeter)	GAINPH	(gain/phase analyser)	VECTOR	(vector voltmeter)	POWER	(power meter)	LCR	(LCR meter)	HARMON	(harmonic analyser)	SCOPE	(oscilloscope)
VRMS	(rms voltmeter)														
GAINPH	(gain/phase analyser)														
VECTOR	(vector voltmeter)														
POWER	(power meter)														
LCR	(LCR meter)														
HARMON	(harmonic analyser)														
SCOPE	(oscilloscope)														
Reply:	none														
Example:	MODE,GAINPH														
Notes:	MODE sets the measurement mode of the instrument														

## MULTIL

## MULTIL

Function: Selects data for multi string reply

Description: Selects data values across phases and functions that can be read in a single string.

Format: *MULTILOG,index,phase,function*

Arguments:

index:		
0	clear all	
1-30	select data 1-30	
phase:		
1-3	phase 1-3	
4	sum	
5	neutral	
function:		
1-99	see appendix	

Reply: none

Example:

```
MULTIL,0
MULTIL,1,1,2      (phase 1 Watts)
MULTIL,2,2,2      (phase 2 watts)
MULTIL,3,4,3      (sum VA)
MULTIL?
3 data values returned
```

Notes:

## MULTIL?

## MULTIL?

Function:	Reads multi string reply
Description:	Waits for data to be available then returns selected results.
Format:	MULTILOG?
or:	MULTILOG, <i>lines</i> ?
Arguments:	Lines: Integer
Reply:	Up to 60 data values as selected by the MULTILOG command in a single reply string OR Up to 60 data values as selected by the MULTILOG command in a single reply string, replying " <i>lines</i> " times.
Example:	MULTIL,0 MULTIL,1,1,2 (phase 1 Watts) MULTIL,2,2,2 (phase 2 watts) MULTIL,3,4,3 (sum VA) MULTIL? 3 data values returned MUTLIL,5? Replies 5 times, each containing 3 data values
Notes:	The MULTILOG, <i>lines</i> ? command will reply each time a new data point is available.

## **NEWLOC**

## **NEWLOC**

Function:	Waits for new data then holds so that multiple commands can be used on the same data set.
Description:	Reads multiple sets of data
Format:	NEWLOC
Arguments:	None
Reply:	none
Example:	NEWLOC;HARMON?SERIES;HPOWER? Harmonic series and Power data returned
Notes:	After the command the data will still be held so to release the lock send SUSPEND,OFF

## NOOVER

## NOOVER

Function:	Disable overranging
Description:	Prevents an overrange error from blanking out results in manual ranging.
Format:	NOOVER, <i>value</i>
Arguments:	value: ON OFF
Reply:	none
Example:	NOOVER,ON
Notes:	This can be useful when testing devices in a noisy environment. The range can be set to the correct range for the signal to be measured even if sporadic noise spikes would push it up on to the next range.

## PFCONV

## PFCONV

Function:	Set power factor sign convention.
Description:	Fundamental power factor is given a sign depending convention either: negative if lagging current negative if leading current
Format:	PFCONV, <i>type</i>
Arguments:	type: NEGLAG NEGLEA
Reply:	none
Example:	PFCONV,NEGLAG
Notes:	An inductive load would have a lagging current, a capacitive load would have a leading current. The sign given to VAR can be independently set: see VARCON



## OFFSET

## OFFSET

Function: Set the output offset

Description: Sets the output generator offset in Volts.

Format: `OFFSET,offset`

Arguments: offset in Volts

Reply: none

Example: `OFFSET,5e-3` (set offset to 5mV)

Notes:

## OUTPUT

## OUTPUT

Function:	Set output
Description:	Turns the output on or off, or sets the level mode to dBm or voltage. Also specifies the action at the end of a sweep
Format:	OUTPUT, <i>command,sweep,phase</i>
Arguments:	command: OFF ON DCONLY VOLT DBM sweep: OFF ON DCONLY
Reply:	none
Example:	OUTPUT,ON
Notes:	For safety, the output defaults to off and must be turned on explicitly. It is not necessary to send all the arguments, but if they are sent they must be in the specified order

## PAV

## PAV

Function:	Set phase angle voltmeter mode.
Description:	Set phase angle voltmeter mode and parameter.
Format:	<i>PAV,parameter,lvdt scale</i>
Arguments:	<p>parameter:</p> <p>INPHAS QUADR TANPHI MAGNIT POLAR A2/1 RMS2 RMS2/1 LVDT-D LVDT-R</p> <p>lvdt scale:</p> <p>scale factor in m for lvdt applications</p>
Reply:	none
Example:	PAV,LVDT-D,0.1
Notes:	<p>It is not necessary to send any arguments, but those that are sent must be in the specified order.</p> <p>PAV and VECTOR are aliases for the same command.</p>

## PAV?

## PAV?

Function:	Phase angle voltmeter query
Description:	<p>Read phase angle voltmeter results.</p> <p>Sets phase angle voltmeter mode if not already set.</p> <p>Waits for next unread data if necessary.</p> <p>Clears new data available bit read by DAV?</p>
Format:	PAV?
or:	PAV,RMS?
or:	PAV,SWEEP?
Arguments:	none, or SWEEP
Reply:	<p>7 data values separated by commas:</p> <p>freq,mag1,mag2,ratio,phase,a,b</p> <p>+5 data values if CH3 enabled</p> <p>mag3,ratio3,phase3,a3,b3</p> <p>+ rms values if PAV,RMS?</p> <p>1 line per result for sweep data</p>
Example:	<p>FREQ,3300</p> <p>OUTPUT,ON</p> <p>PAV?</p> <p>data returned</p>
Notes:	<p>PAV? waits for next unread data.</p> <p>PAV?SWEEP does not wait for new data – data can be read multiple times.</p> <p>PAV and VECTOR are aliases for the same command.</p>

## **PHASEM**

## **PHASEM**

Function: Set phase meter mode.

Description: Select phase meter mode.

Format: PHASEM

Arguments: as FRA

Reply: none

Example: PHASEM

Notes: PHASEM is an alias for FRA to support early instruments

**PHASEM?**

**PHASEM?**

Function:	Phase meter query
Description:	Reads phase meter results
Format:	PHASE?
Arguments:	none
Reply:	as FRA?
Example:	PHASE? Data returned
Notes:	PHASEM is an alias for FRA to support early instruments.

## PHCONV

## PHCONV

Function: Set phase convention

Description: Set phase convention

Format: PHCONV,*convention*

Arguments: convention:  
              180: -180 to +180  
              -360: 0 to -360  
              +360: 0 to +360

Reply: none

Example: PHCONV, -360

Notes:

## PHREF

## PHREF

Function:	Set phase reference
Description:	Select measurement of phase as CH2 relative to CH1 or as CH1 relative to CH2
Format:	PHREF, <i>channel</i>
Arguments:	channel: CH1: phase = ch2 wrt ch1 CH2: phase = ch1 wrt ch2
Reply:	none
Example:	PHREF, CH2
Notes:	This parameter influences the phase meter mode and the phase angle voltmeter mode



## POWER

## POWER

Function:	Set up power meter mode.
Description:	Configure power meter with integration type
Format:	POWER, <i>integration type</i>
Arguments:	integration type: MAGNITUDE SIGNED
Reply:	none
Examples:	POWER,SIGNED POWER
Notes:	It is not necessary to send the integration type argument.

## POWER?

## POWER?

Function: Read power meter results

Description: Reads back latest power meter results.  
Sets power meter mode if not already set.  
Waits for next unread data if necessary.  
Clears new data available bit read by DAV?

Format: POWER?  
or: POWER?*results*  
or: POWER,*results?*

Arguments: results:  
                  WATTS  
                  RMS  
                  INTEGR

Reply: WATTS:  
          9 data values separated by commas  
          W,W.f,VA,VA.f,pf,pf.f,Wdc,W.h,freq  
RMS:  
          8 data values separated by commas  
          rms1,2,dc1,2,f1,2,phase1,2  
INTEGR:  
          9 data values separated by commas  
          Wh,Wh.f,VAh,VAh.f,avpf,avpf.f,  
          Ah,Ah.f,time  
no argument:  
          26 data values separated by commas  
          WATTS, RMS, INTEGR

Example: POWER?WATTS

Notes:

## PROGRA

## PROGRA

Function:	Access non volatile program stores.
Description:	Recall, store or delete non-volatile program store.
Format:	PROGRA, <i>function,number</i>
Arguments:	function: RECALL STORE DELETE number 0-999
Reply:	none
Example:	PROGRA,RECALL,13
Notes:	Number 0 represents factory default, which can only be recalled.

**PROGRA?**

**PROGRA?**

Function: Identify program.

Description: Reads the name of the last program to be loaded or recalled or a program is memory.

Format: PROGRA,NAME?  
PROGRA,NAME,*number*?  
PROGRA,FILES?

Arguments: number  
0-999

Reply: NAME: text string  
FILES: 1 text string per stored program:  
number,name,date

Example: PROGRA,NAME?  
factory default

Notes:

## RANGE

## RANGE

Function: Set channel ranging.

Description: Select minimum range and range control for a given input channel.

Format: *RANGE,channel,ranging,range*

Arguments: channel:  
                   CH1  
                   CH2  
           ranging:  
                   AUTO  
                   UPAUTO  
                   MANUAL  
           range:  
                   nominal range value

Reply: none

Example: RANGE,CH2,MANUAL,3V

Notes:

**RESOLU****RESOLU**

Function:	Set the data resolution		
Description:	Data is returned in scientific format with exponent and mantissa. The resolution of the mantissa may be selected to be 5 digit (NORMAL) or 6 digit (HIGH).		
Format:	RESOLU, <i>format</i>		
Arguments:	format:		
	NORMAL	(5 digit mantissa)	
	HIGH	(6 digit mantissa)	
	BINARY	(raw binary format)	
Reply:	none		
Example:	RESOLU,HIGH		
Notes:	<p>The resolution only changes the real number replies.</p> <p>Data format for NORMAL is: [-]1.2345E[-]00</p> <p>Data format for HIGH is: [-]1.23456E[-]00</p> <p>The signs of the mantissa and exponent, shown as [-] in the above examples, are only sent if they are negative.</p> <p>Data format for BINARY is a proprietary floating point format which returns raw data in a minimum number of data bytes.</p>		

## RESULT

## RESULT

Function: Access non volatile result stores.

Description: Recall, store or delete non-volatile result.

Format: *RESULT,function,number*

Arguments: function:  
              RECALL  
              STORE  
              DELETE  
              number  
              0-999

Reply: none

Example: RESULT,RECALL,13

Notes:

**RESULT?**

**RESULT?**

Function: Identify available results.

Description: Reads the name of the stored results.

Format: RESULT,NAME,*number*?  
RESULT,FILES?

Arguments: number  
0-999

Reply: NAME: text string  
FILES: 1 text string per stored result:  
number,name,date

Example: RESULT,NAME,13?  
PSU stability sweep #3

Notes:



## **REZERO**

## **REZERO**

Function: Rezero front end

Description: Request the DSP to re-compensate for dc offset and compute a new autozero

Format: REZERO

Arguments: none

Reply: none

Example: REZERO

Notes:

## SCALE

## SCALE

Function: Set channel scale factor.

Description: Set a multiplying scale factor for a given input channel.

Format: *SCALE,channel,factor*

Arguments: channel:  
                   CH1  
                   CH2  
                   CH3  
           factor:  
                   multiplying scale factor

Reply: none

Example: SCALE,CH2,10

Notes:

## SCOPE?

## SCOPE?

Function: Fetch raw scope data.

Description: Read back raw oscilloscope data.

Format: SCOPE,*channel?*  
SCOPE,*phase,channel?*

Arguments: phase:  
                PHASE1  
                PHASE2  
                PHASE3  
                NEUTRA  
            channel:  
                VOLTAGE  
                CURRENT

Reply: 252 signed integers:  
            range  
            trigger  
            250 x data

Example: HOLD,ON  
SCOPE,PHASE1,VOLTAGE?  
read data  
SCOPE,PHASE2,VOLTAGE?  
read data  
SCOPE,PHASE3,VOLTAGE?  
read data  
HOLD,OFF

Notes:

**SCREEN?**

**SCREEN?**

Function: Read the screen data

Description: Returns a bit map of screen pixel display in ascii and hex format

Format: SCREEN?

Arguments: none

Reply: Multiple data bit values

Example: SCREEN?  
data returned

Notes: SCREEN? response:

ASCII coded Hex  
(2 characters for each byte)  
240 lines of 40 bytes (each line represents one line of the display)  
preceded by #H  
Each byte represents 8 dots where the lsb is the leftmost dot of the display  
The bit is set for on and cleared for off

## SETUP

## SETUP

Function:	Upload instrument set up
Description:	All the settings within the instrument may be read by SETUP?. The same settings may then be stored by ending the same data back to the instrument. As it sends all settings in a compressed format it is quicker than setting individual parameters.
Format:	SETUP,index,data
Arguments:	index: 0-15 data: ASCII hex as returned by SETUP?
Reply:	none
Example:	SETUP? Read 16 lines of data SETUP,00,data00 SETUP,01,data01 . . SETUP,15,data15
Notes:	The settings are only updated when the 16 <sup>th</sup> line has been received and the checksum has been verified.

## **SETUP?**

## **SETUP?**

Function: Read instrument set up

Description: All the settings within the instrument may be read by SETUP?. The same settings may then be stored by ending the same data back to the instrument. As it sends all settings in a compressed format it is quicker than setting individual parameters.

Format: SETUP?

Arguments: none

Reply: 16 lines of ASCII data

Example: SETUP?  
Read 16 lines of data

Notes:

## SHUNT

## SHUNT

Function:	Set channel shunt value
Description:	Set the resistance factor of a current shunt to be divided into the measured voltage for a given input channel.
Format:	SHUNT, <i>channel,resistance</i>
Arguments:	channel: CH1 CH2 CH3 resistance: shunt resistance in Ohms
Reply:	none
Example:	SHUNT,CH1,10
Notes:	The SHUNT command is still accepted if the channel has not been configured for current. The value stored will be used when the channel is configured for current.

## SMOOTH

## SMOOTH

Function:	Select the smoothing
Description:	Sets the filter time constant and dynamic response.
Format:	SMOOTH, <i>type</i> , <i>dynamics</i>
Arguments:	type: NONE NORMAL SLOW dynamics: AUTO FIXED
Reply:	none
Example:	SMOOTH,NORMAL,FIXED SMOOTH,NONE
Notes:	It is not necessary to send both parameters if it is only required to set the type. Both arguments must be sent to set the dynamics. FILTER is an alias for SMOOTH



## **SPEED**

## **SPEED**

Function:	Sets the measurement speed
Description:	Sets the minimum window size for the measurement.
Format:	SPEED, <i>value</i> SPEED,WINDOW, <i>time</i>
Arguments:	value: FAST MEDIUM SLOW VSLOW WINDOW
Reply:	none
Example:	SPEED,SLOW SPEED,WINDOW,0.1
Notes:	

## **START**

## **START**

Function: Start sweep

Description: Initiate sweep in those functions that have a sweep or resets filtering in others.

Format: START

Arguments: none

Reply: none

Example: MODE,GAINPH (set gain phase analyser)  
START

Notes:

## STATUS?

## STATUS?

Function:	Read back channel ranging status.
Description:	Read back condition of selected channel: range number (1-16) range text overflow/underflow status
Format:	STATUS?
or:	STATUS, <i>channel</i> ?
Arguments:	channel: CH1 CH2 CH3
Reply:	If no channel specified: OVER if any channel overflow LOW if any channel underflow OK if all channels in range Else, range number,range text,status 1-16 range as per RANGE command OVER if overflow LOW if underflow OK if in range
Example:	STATUS,CH1? 6,3V,OK
Notes:	

## STOP

## STOP

Function: Stop sweep

Description: Stop an active sweep, or data streaming.

Format: STOP

Arguments: none

Reply: none

Example: MODE,PHASE,STREAM,0.01  
START  
*read data values as required*  
STOP  
*read remaining data values*

Notes:

## SUSPEND

## SUSPEND

Function:	Suspend data acquisition
Description:	Suspends the background data acquisition to maximise the communications speed.
Format:	SUSPEND, <i>command</i>
Arguments:	command OFF ON
Reply:	none
Example:	DATALOG, NONVOL, 36 START wait for datalog STOP SUSPEND, ON DATALOG? data, data, data, data, ..... SUSPEND, OFF
Notes:	

## **TAGREP**

## **TAGREP**

Function:	Set up a reply tag
Description:	Select a reply tag to identify the instrument in a multi-instrument environment
Format:	TAGREP, <i>on/off</i>
Arguments:	on/off: ON OFF
Reply:	none
Example:	TAGREP,ON *ESR? PPA5530:00635:1
Notes:	When "tag reply" is turned on every reply string has a prefix of an identification string comprising the model and serial number

**TFA**

**TFA**

Function: Set transfer function analyser mode.

Description: Set transfer function analyser mode.

Format: TFA

Arguments:

Reply: none

Example: TFA

Notes: This command has the same effect as  
MODE,GAINPH.  
FRA, GAINPH, TFA are aliases for the  
same command.

**TFA?**

**TFA?**

Function: transfer function analyser query

Description: Read transfer function analyser results.

Format: TFA?  
or: TFA,SWEEP?

Arguments: none, or SWEEP

Reply: As for FRA?

Example: OUTPUT,ON  
TFA  
FSWEEP,20,10,20E3  
START  
DAV?  
3  
DAV?  
15  
TFA?SWEEP  
data returned

Notes: FRA, GAINPH, TFA are aliases for the same command



**USER?**

**USER?**

Function: Read the user data

Description: Returns up to 3 lines of user data

Format: USER?

Arguments: none

Reply: 3 lines of ASCII terminated by CR

Example: USER?  
          Newtons4th Ltd  
          R&D department  
          PSM3750 #4

Notes:

## **VARCON**

## **VARCON**

Function:	Set VAr sign convention.
Description:	Fundamental VAr measurement is given a sign depending convention either: negative if lagging current negative if leading current
Format:	VARCON, <i>type</i>
Arguments:	type: NEGLAG NEGLEA
Reply:	none
Example:	VARCON,NEGLAG
Notes:	An inductive load would have a lagging current, a capacitive load would have a leading current. The sign given to power factor can be independently set: see PFCONV

## VECTOR

## VECTOR

Function:	Set vector voltmeter mode.
Description:	Set vector voltmeter mode and parameter.
Format:	VECTOR, <i>parameter,lvdt scale</i>
Arguments:	As PAV
Reply:	none
Example:	VECTOR,LVDT-D,0.1
Notes:	PAV and VECTOR are aliases for the same command.

**VECTOR?**

**VECTOR?**

Function: Vector voltmeter query

Description: Read vector voltmeter results.

Format: VECTOR?  
or: VECTOR,SWEEP?

Arguments: none, or SWEEP

Reply: As PAV

Example: FREQ,3300  
OUTPUT,ON  
VECTOR?  
data returned

Notes: PAV and VECTOR are aliases for the same command.

**VERSIO?**

**VERSIO?**

Function:	Read the instrument code versions.
Description:	Returns an ASCII string with the details of the various parts of the instrument firmware.
Format:	VERSIO?
Arguments:	none
Reply:	date code, type, cpu, dsp, fpga, boot
Examples:	VERSION? PQ3504,1,1.12,1.12,1.01,2.01
Notes:	This data can be displayed on the screen by pressing SYSTEM then BACK

## **VRMS**

## **VRMS**

Function: Set up rms voltmeter.

Description: Set mode to rms voltmeter.

Format: VRMS

Arguments: none

Reply: none

Examples: VRMS

Notes: This has the same effect as MODE,VRMS

## VRMS?

## VRMS?

Function:	Read true rms voltmeter results
Description:	Reads back latest voltmeter results. Sets voltmeter mode if not already set. Waits for next unread data if necessary. Clears new data available bit read by DAV?
Format: or:	VRMS? VRMS, <i>results?</i>
Arguments:	results: RMS SURGE
Reply:	RMS: 8 data values separated by commas rms1,2,dc1,2,ac1,2,dbm1,2 SURGE: 6 data values separated by commas pk1,2,cf1,2,surge1,2 no argument: 14 data values separated by commas RMS results then SURGE
Example:	VRMS?RMS
Notes:	As VRMS? does not send the same data twice but waits instead for the next result, it is not necessary to check the data available bits before sending the VRMS? command.

## WAVEFO

## WAVEFO

Function: Set the output waveform

Description: Selects the output waveform for the signal generator.

Format: WAVEFO, *type*

Arguments: type:  
               SINEWAVE  
               SQUARE  
               TRIANGLE  
               SAWTOOTH  
               PULSE  
               WHITENOISE

Reply: None

Example: FREQUE,500  
           WAVEFO,TRIANG (triangle wave)  
           OUTPUT,ON

Notes:



## WIRING

## WIRING

Function: Set the wiring configuration

Description: Selects 2 channel or 3 channel operation.

Format: WIRING, *type*

Arguments: type:  
              CH2  
              CH3

Reply: None

Example: WIRING,CH3

Notes:

**ZERO****ZERO**

Function:	Apply or remove the zero
Description:	Applies or removes a zero function depending on the measurement mode (same as pressing ZERO key). Performs lead compensation in LCR mode.
Format:	ZERO ZERO,DELETE ZERO,DB, <i>offset</i> ZERO,PHASE, <i>offset</i>
LCR compensation	ZERO,SINGLE ZERO,SWEEP, <i>steps,start,finish</i> ZERO,OPEN ZERO,SHORT ZERO,STORE ZERO,RECALL
Arguments:	offset: offset value steps: LCR sweep compensation steps start: LCR compensation start frequency stop: LCR compensation stop frequency
Reply:	none
Example:	ZERO,SWEEP,100,1e3,1e6 ZERO,OPEN <i>performs open circuit compensation</i>
Notes:	

## ZOOM

## ZOOM

Function:	Sets the display zoom parameters.
Description:	Sets the zoom level and data.
Format:	<i>ZOOM,level,data1,data2,data3,data4</i>
Arguments:	<p>level:</p> <ul style="list-style-type: none"> <li>0 – no zoom</li> <li>1 – normal</li> <li>2 – 4 line display</li> <li>3 – 3 line display</li> </ul> <p>data1-4:</p> <p>function data for zoom</p> <p>data consists of line number for channel 1 or line number + 64 for channel 2 or line number + 128 for channel 3</p>
Reply:	None
Example:	<p>VRMS</p> <p>ZOOM,1,1,12 (level 1, ch1 rms, ch2 rms)</p>
Notes:	It is not necessary to send all the parameters, but whatever parameters are sent must be in the correct order.

## **ZOOM?**

## **ZOOM?**

Function: Read the display zoom parameters.

Description: Reads the zoom level and data.

Format: ZOOM?

Arguments:

Reply: 5 integers separated by commas:  
Level,data1,data2,data3,data4

data consists of line number for channel 1  
or line number + 64 for channel 2  
or line number + 128 for channel 3

Example: ZOOM?  
1,1,129,0,0 (level 1, ch1 rms, ch2 rms)

Notes:



Appendices

COMMAND SUMMARY

CONFIGURABLE PARAMETERS

## COMMAND SUMMARY

command format	reply format
*CLS	
*ESE,value	
*ESE?	single integer data value
*ESR?	single integer data value
*IDN?	company,product,serial no,version
*OPC?	0 or 1
*RST	
*SRE,value	single integer data value
*SRE?	
*STB?	single integer data value
*TRG	
*TST?	single integer data value
*WAI	
ABORT	
ACTRIM,channel,level,tol	
ALARM,latch,sounder	
ALARM?	single integer data value
ALARME,value	
ALARME?	single integer data value
ALARM1,type,data,high,low	
ALARM2,type,data,high,low	
AMPLIT,amplitude	
AUXILI,device,value	
BANDWI,type	
BEEP	
BLANKI,on/off	
CONFIG,parameter,data	
CONFIG,parameter?	single integer or real data value
COUPLI,channel,coupling	
COUPLI,channel?	single integer data value
CYCLES,cycles	
DATALO,function,interval	
DATALO,start,records?	index,time,data... one record per line
DAV?	single integer data value
DAVER,value	
DAVER?	single integer data value
DELAY,time	
FAST,on/off	

## PSM3750 communications manual

FILTER,type,dynamics	
FRA	
FRA?	freq,mag1,mag2,dB,phase,delay
FRA,SWEEP?	n lines of FRA? data
FREQUE,frequency	
FSWEEP,steps,start,end,log	
GAINPH	
GAINPH?	freq,mag1,mag2,dB,phase,delay
GAINPH,SWEEP?	n lines of GAINPH? data
HARMON,scan,para,h,hmax	
HARMON?	freq,mag1,mag2,hmag1,hmag2,h1,h2
or	freq,mag1,mag2,thd1,thd2,h1,h2
HARMON,SERIES?	mag1,%1, $\phi$ 1,mag2,%2, $\phi$ 2
HARMON,SWEEP?	n lines of HARMON? data
HOLD,on/off	
INPUT,channel,type	
INPUT,channel?	single integer data value
KEYBOA,value	
LCR,conditions,param,head	
LCR?	freq, mag1, mag2, impedance, phase, R, L, C (series), R, L, C (parallel), tan $\delta$ , Q, reactance
or	freq, mag1, mag2, impedance, phase, resistance, reactance, admittance, phase, conductance, susceptance
LCR,SWEEP?	n lines of data: freq,QF,tan $\delta$ ,impedance,phase,L,C,R freq,QF,tan $\delta$ ,admittance,phase,L,C,R
or	
LOWFRE,on/off	
MARKER,on/off,frequency	
MODE,type	
MULTIL,index,ch,func	
MULTIL?	up to 60 floating point values
NEWLOC	
NOOVER,on/off	
OFFSET,offset	
OUTPUT,type,sweep,phase	
PAV,parameter,scaling	
PAV?	freq,mag1,mag2,parameter,phase,a,b
PAV,SWEEP?	n lines of VECTOR? data
PFCNV,type	
PHASEM	
PHASEM?	same as FRA
PHCONV,convention	



## PSM3750 communications manual

PHREF,channel	
POWER,integration type	
POWER,WATTS?	W,W.f,VA,VA.f,pf,pf.f,Wdc,W.h,freq
POWER,RMS?	rms1,rms2,dc1,dc2,fnd1,fnd2, $\phi$ 1, $\phi$ 2
POWER,INTEGR?	Wh,Wh.f,VAh,VAh.f,pf,pf.f,Ah,Ah.f,t
PROGRAM,function,number	
PROGRAM?	CR terminated text string
RANGE,ch,ranging,range	
RESOLU.format	
RESULT,function,number	
RESULT?	CR terminated text string
REZERO	
SCALE,channel,factor	
SCALE,channel?	single real data value
SCOPE,channel?	252 signed integers
SCREEN?	240 lines of 40 bytes
SETUP,index,data	
SETUP?	16 lines of ASCII data
SHUNT,channel,resistance	
SHUNT,channel?	single real data value
SMOOTH,type,dynamics	
SPEED,speed	
START	
STATUS,channel?	range number,range text,over/low/ok
STOP	
SUSPEND,on/off	
TAGREP,on/off	
TFA	
TFA?	freq,mag1,mag2,dB,phase,delay
TFA,SWEEP?	n lines of TFA? data
USER?	3 CR terminated text strings
VARCON,type	
VECTOR,parameter,scaling	
VECTOR?	freq,mag1,mag2,parameter,phase,a,b
VECTOR,SWEEP?	n lines of VECTOR? data
VERSION?	datecode,type,cpu,dsp,fpga,boot
VRMS	
VRMS?	RMS? data followed by SURGE?
VRMS,RMS?	rms1,rms2,dc1,dc2,ac1,ac2,db1,db2
VRMS,SURGE?	pk1,pk2,cf1,cf2,surge1,surge2
WAVEFO,type	
WIRING,type	
ZERO	
ZERO,DELETE	

## PSM3750 communications manual

ZOOM,level,d1,d2,d3,d4

ZOOM?

level,d1,d2,d3,d4

calibration commands

CALAPP	
CAL AUX,string	
CAL COM,freq	
CAL DCO,value	
CAL FIL,index,value	
CAL FIL?	six real data values
CAL FRQ,index,freq	
CAL FRQ?	seven real data values
CAL IAI,freq,res, reac	
CAL IBR,index,value	
CAL IBR?	single integer data value
CAL IDS,string	
CAL IDS?	string
CAL OUT ,index,value	
CAL PHA,index	
CAL RES	
CAL SAV,password	
CAL SNO,serial number	
CAL STR,string	
CAL STR?	string

## Appendix B – Configurable parameters

All parameters can be accessed using the CONFIG command:

CONFIG,number,parameter?

CONFIG,parameter,data

<i>Number</i>	<i>Function</i>	<i>Parameter</i>
<b>System parameters</b>		
1	Operating mode, (Sets main mode)	0=RMS Voltmeter 1=Frequency Response analyser 2=Power Meter 3=LCR Meter 4=Harmonic Analyser 5= Vector Voltmeter 6=Oscilloscope
2	Language, (System Options if installed)	0=English 1=Other (if installed)
3	Bandwidth, (Acquisition Control)	0=Auto 1=Wide
4	Autozero, (System Options)	0=Auto 1=Manual
5	Low blanking, (System Options & RMS Voltmeter)	0=Off 1=On
6	Phase convention, (System Options)	0=-180° to +180° 1=0° to -360° 2=0° to +360°
7	Generator output, (Output Options)	0=Off 1=On

## PSM3750 communications manual

2=dc only

- 8            Graph, (System Options)
  - 0=Dots
  - 1=Lines
- 9            Keyboard beep, (System Options)
  - 0=Off
  - 1=On
- 11          Low frequency mode, (Acquisition Control)
  - 0=Off
  - 1=On
- 12          Speed "window size", (Acquisition Control, Enter figures)
- 13          Speed, (Acquisition Control)
  - 0=Very slow
  - 1=Slow
  - 2=Medium
  - 3=Fast
  - 4=Very fast
  - 5=Window
- 14          Filter, (Acquisition control)
  - 0=Normal
  - 1=Slow
  - 2=None
- 15          Filter dynamics, (Acquisition Control, "Filter normal/slow")
  - 0=Auto reset
  - 1=Fixed time
- 16          Baud rate, (Comms-Remote Options, RS232)
  - 0=19200
  - 1=9600
  - 2=4800
  - 3=2400
  - 4=1200
- 18          Sweep steps, (Sweep Control-Enter step number figures)
- 19          Sweep start frequency, (Sweep Control-Enter figures)

## PSM3750 communications manual

- 20 Sweep end frequency, (Sweep Control-Enter figures)
- 21 Sweep-type, (Sweep Control)  
0=Single  
1=Repeat
- 22 Conditions, (LCR Meter)  
0=Auto frequency  
1=Manual  
2=Auto shunt
- 23 Shunt, (System Options)  
0=Default  
1=Manual
- Input parameters**
- 24 Input 1 (CH1), (CH1-Input 1)  
0=Direct  
1=External shunt  
2=External attenuator
- 25 Input 2 (CH2), (CH2-Input 2)  
As Ch1
- 26 Input 3 (CH3), (CH3-Input 3)  
As Ch1
- 27 Minimum range (CH1), (CH1-Input 1)  
0=1mv  
1=3mv  
2=10mv  
3=30mv  
4=100mv  
5=300mv  
6=1v  
7=3v  
8=10v
- 28 Minimum range (CH2), (CH2-Input 2)  
As CH1
- 29 Minimum range (CH3), (CH3-Input 3)  
As CH1

- 30 Autoranging (CH1), (CH1-Input 1)  
As Ch1
- 31 Autoranging (CH2), (CH2-Input 2)  
As Ch1
- 32 Autoranging (CH3), (CH3-Input 3)  
0=Full Autorange  
1=Autorange up  
2=Manual
- 33 Coupling (CH1), (CH1-Input 1)  
0=ac+dc  
1=ac
- 34 Coupling (CH2), (CH2-Input 2)  
As Ch1
- 35 Coupling (CH3), (CH23-Input 3)  
As Ch1
- 36 Scale (CH1), (CH1-Input, Enter figures)
- 37 Scale (CH2), (CH2-Input, Enter figures)
- 38 Scale (CH3), (CH3-Input, Enter figures)
- 39 External shunt (CH1), (CH1-Input, Enter figures)
- 40 External shunt (CH2), (CH2-Input, Enter figures)
- 41 External shunt (CH3), (CH3-Input, Enter figures)

**Display parameters**

- 42 Zoom level, (Main Display)  
0=Zoom -  
1=Zoom +  
2=Second zoom +
- 43 Display zoom characters on line 1
- 44 Display zoom characters on line 2
- 45 Display zoom characters on line 3

- 46 Display zoom characters on line 4
- 47 Display type, (Main display-datalog or sweep display mode)  
0=Real Time  
1=Table  
2=Graph

**Signal generator parameters**

- 48 Generator frequency, (Output Options-Enter figures)
- 49 Generator amplitude, (Output Options-Enter figures)
- 50 Generator offset, (Output Options-Enter figures)
- 51 Generator waveform, (Output Options)  
0=Sinewave  
1=Triangle  
3=Square wave  
4=sawtooth  
5=pulse  
6=white noise
- 52 Frequency step, (Output options-Enter figures)
- 53 Amplitude step, (Output options-Enter figures)
- 54 Amplitude dBm (Output options-[116 system control]-Enter figures)
- 55 Generator after sweep, (Sweep Control)  
0=Off  
1=On

**Datalog parameters**

- 58 Datalog, (Acquisition Control-memory type)  
0=Disabled  
1=RAM  
2=Non volatile
- 59 Interval, (Acquisition Control-RAM/Non volatile-Enter time figures)



**General parameters**

- 64 Frequency marker, (Sweep Control)  
0=Off  
1=On
- 65 Marker frequency, (Enter frequency-Graph display-After sweep, alters marker position)
- 66 Program 1-6 direct load, (System Options)  
0=Disabled  
1=Enabled

**Power meter parameters**

- 83 Integration type, (Power meter)  
0=Signed  
1=Magnitude

**Harmonic analyser parameters**

- 99 Scan, (Harmonic analyser)  
0=Single  
1=Difference thd  
2=Series thd
- 100 Harmonic, (Harmonic analyser)  
0, 1 & 2=2  
3=3  
4=4  
5=5  
etc up to 64
- 101 Harmonics (Max), (Harmonic analyser-scan-series thd)  
0, 1 & 2=2  
3=3  
4=4  
5=5  
etc up to 64
- 102 Parameter, (Harmonic analyser)  
0=%

1=dB

103 Bargraph Scale, (Harmonic analyser-scan-series thd-Enter figure)

### **LCR sweep zero parameters**

106 Frequency, (LCR Mode-Zero)  
0=Single  
1=Sweep

107 Sweep start (frequency), (LCR Mode-Zero-Enter figures)

108 Sweep end (frequency), (LCR Mode-Zero-Enter figures)

109 Steps, (LCR Mode-Zero-Enter figures)

### **System parameters**

116 Control, (System options)  
0=Volts  
1=dBm

117 Step message, (System options)  
0=Enabled  
1=Disabled

118 Display sequence, (Graph display- After sweep alters screen display)  
0=Primary Parameter  
1=Secondary Parameter  
2=Both Parameters

119 Length units, (System options)  
0=Metres  
1=Inch

### **LCR meter parameters**

137 Parameter, (LCR Meter)  
0=Auto  
1=Capacitance  
2=Inductance  
3=Impedance

4=Admittance

- 138 Sweep, (LCR Meter)
  - 0=Series
  - 1=Parallel
- 139 Graph, (LCR Meter)
  - 0=Single
  - 1=Tan $\delta$ /QF
  - 2=Resistance
- 140 LCR head shunt, (Auxiliary control-fixture-LCR active head)
  - 0=Low
  - 1=Normal
  - 2=High
  - 3=Very high
- 141 Graph, (LCR meter-impedance)
  - 0=Linear
  - 1=Log
- 142 Phase reference, (Mode-LCR-Zero-LCR Compensation-Enter figures)
- 143 Reference (Value), (Mode-LCR-Zero-LCR Compensation-Enter figures)
- 144 Reference, (Mode-LCR-Zero-LCR Compensation)
  - 0=Capacitance
  - 1=Resistance
  - 2=Inductance
- 145 Connection, (LCR Meter)
  - 0=Shunt
  - 1=Divider Zx low
  - 2=Divider Zx high

**Gain/Phase analyser parameters**

- 147 Graph (time selection), (FRA)
  - 0=Phase
  - 1=Delay
- 148 dB offset, (FRA-Enter figures)

149 Gain/Phase margins, (FRA)  
                                   0=Disabled  
                                   1=Enabled

150 Ratio, (FRA)  
                                   0=ch2/ch1  
                                   1=ch1/ch2

### **System parameters**

151 Minimum cycles, (Acquisition control-Enter figures)

152 Delay time, (Acquisition control-Enter figures)

153 IEEE address, (Comms-Remote options-interface-GPIB-Enter figures)

154 Interface, (Comms-Remote options)  
                                   0=RS232  
                                   1=LAN  
                                   2=GPIB

### **Alarm functions (Monitor 1)**

156 Monitor 1 data, (Alarm-monitor options)  
                                   0=Zoom1  
                                   1=Zoom2  
                                   2=Zoom3  
                                   3=Zoom4

157 Alarm type, (Alarm-monitor options)  
                                   0=Disabled  
                                   1=Linear  
                                   2=Alarm if high  
                                   3=Alarm if low  
                                   4=Outside window  
                                   5=Inside window

158 High threshold (Alarm type), (Alarm-monitor options-Enter figures)

159 low threshold (Alarm type), (Alarm-monitor options-Enter

figures)

160 Alarm latch (Alarm type), (Alarm-monitor options)

0=Off

1=On

161 Alarm sounder (Alarm type), (Alarm-monitor options)

0=Enabled

1=Disabled

### **Alarm functions (Monitor 2)**

167 Monitor 2 data, (Alarm-monitor options)

0=Zoom1

1=Zoom2

2=Zoom3

3=Zoom4

168 Alarm 2 type, (Alarm-monitor options)

0=Disabled

1=Linear

2=Alarm if high

3=Alarm if low

4=Outside window

5=Inside window

169 High threshold (Alarm type), (Alarm-monitor options-Enter figures)

170 Low threshold, (Alarm type), (Alarm-monitor options-Enter figures)

### **Graph functions**

173 Graph 2 scaling, (Sweep control)

0=Auto

1=Manual

174 Upper limit (Graph 2 scaling), (Sweep control-Enter figures)

175 Lower limit (Graph 2 scaling), (Sweep control-Enter figures)

**Phase angle voltmeter parameters**

- 177      Parameter, (Vector voltmeter)  
            0=In-phase  
            1=Quadrature  
            2=Tan $\delta$   
            3=Magnitude  
            4=Phase  
            5=In-phase ratio  
            6=rms  
            7=rms2/rms1  
            8=LVDt diff  
            9=LVDt ratio  
            10=User interface
- 178      Scale factor (LVDt), (Vector voltmeter-Enter figures)
- 179      Null meter, (Vector voltmeter)  
            0=Off  
            1=Auto  
            2=Manual
- 180      Upper limit (Null meter), (Vector voltmeter-Enter figures)
- 181      Offset (Parameter), (Vector voltmeter-Enter figures)
- Trim parameters**
- 186      ac trim data, (Trim control)  
            0=Disabled  
            1=CH1  
            2=CH2
- 188      ac level (Trim data), (Vector voltmeter-Enter figures)
- 190      Trim tolerance (Trim data), (Vector voltmeter-Enter figures)
- Other parameters**
- 192      Steps, (Sweep control)

## PSM3750 communications manual

0=Log  
1=Linear

193 Graph 1 scaling, (Sweep control)

0=Auto  
1=Manual

194 Upper limit (Graph 1 scaling), (Sweep control-Enter figures)

195 Upper limit (Graph 2 scaling), (Sweep control-Enter figures)

198 Resolution, Comms-Remote Options)

0=Normal  
1=High  
2=Binary

## Newtonson4th Ltd. contact details

Please direct all queries or comments regarding the PSM3750 instrument or this manual to:

Newtonson4th Ltd.  
30 Loughborough Rd.  
Mountsorrel  
LOUGHBOROUGH  
LE12 7AT  
United Kingdom

Tel: (0116) 230 1066      international    +44 116 230 1066  
Fax: (0116) 230 1061      international    +44 116 230 1061

E-mail address:      [sales@newtonson4th.com](mailto:sales@newtonson4th.com)  
                                 [office@newtonson4th.com](mailto:office@newtonson4th.com)

web site:                      [www.newtonson4th.com](http://www.newtonson4th.com)

At Newtonson4th Ltd. we have a policy of continuous product improvement and are always keen to hear comments, whether favourable or unfavourable, from users of our products. Please telephone, fax, write or e-mail with your comments.