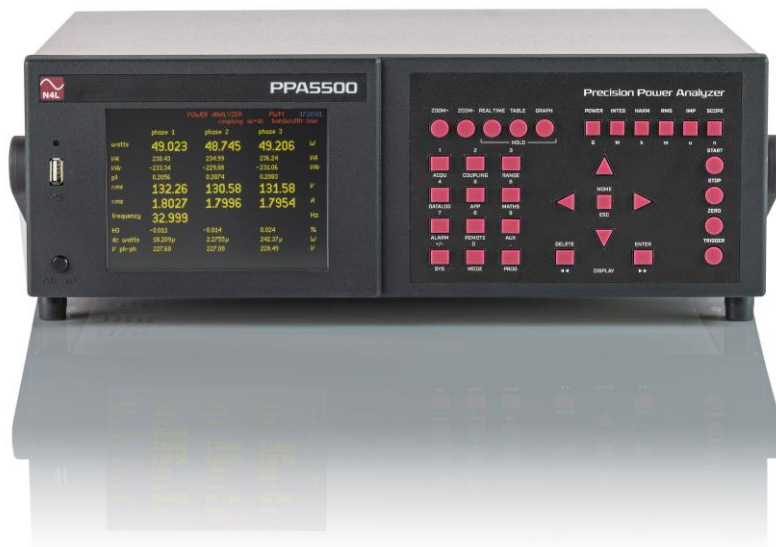




**N4L Newtons4th Ltd**

# PPA5500

## COMMUNICATIONS MANUAL



**Firmware v2.178**

**1<sup>st</sup> August 2019**

## **IMPORTANT SAFETY INSTRUCTIONS**

This equipment is designed to comply with BSEN 61010-1 (2001) (Safety requirements for electrical equipment for measurement, control, and laboratory use) – observe the following precautions:

- Ensure that the supply voltage agrees with the rating of the instrument printed on the back panel **before** connecting the mains cord to the supply.
- This appliance **must** be earthed. Ensure that the instrument is powered from a properly grounded supply.
- The inputs are rated at 1kV rms or dc cat II; 600V rms or dc cat III. **Do not exceed the rated input.**
- Keep the ventilation holes on the underneath and rear 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 PPA45xx series of instruments 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.

Firmware Revision 2.178

This manual is copyright © 2006-2019 Newtons4th Ltd. and all rights are reserved. No part may be copied or reproduced in any form without prior written consent.

1<sup>st</sup> August 2019

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
1.4	Data format.....	1-6
2	Communication commands.....	2-1
	*CLS .....	2-1
	*ESE .....	2-2
	*ESR?.....	2-3
	*IDN?.....	2-4
	*OPC? .....	2-5
	*RST .....	2-6
	*SRE .....	2-7
	*SRE?.....	2-8
	*STB?.....	2-9
	*TRG .....	2-10
	*TST?.....	2-11
	*WAI .....	2-12
	ABORT .....	2-13
	ALARM .....	2-14
	ALARM? .....	2-15
	ALARM1 .....	2-16
	ALARM2 .....	2-17
	ALARME .....	2-18
	ALARME?.....	2-19
	APPLIC.....	2-20
	BANDWI.....	2-21
	BANDWI? .....	2-22
	BEEP .....	2-23
	BLANKI .....	2-24
	CALVER.....	2-25
	CALVER? .....	2-26
	CONFIG .....	2-27
	CONFIG? .....	2-28
	COUPLI .....	2-29
	COUPLI? .....	2-30
	DATALO .....	2-31
	DATALO?.....	2-32
	DAV? .....	2-33

## PPA45xx communications manual

DAVER.....	2-34
DAVER? .....	2-35
DISPLAY .....	2-36
DISPLAY? .....	2-37
EFFICI .....	2-38
EFFICI? .....	2-39
FAST .....	2-40
FQLOCK .....	2-41
FQREF .....	2-42
FREQFI .....	2-43
FREQUE .....	2-44
FSD? .....	2-45
Function: .....	2-45
HARMON .....	2-46
HARMON? .....	2-48
HOLD.....	2-49
INPUT .....	2-50
INTEGR.....	2-51
INTEGR? .....	2-52
KEYBOA .....	2-53
LCR .....	2-54
LCR?.....	2-55
LOWFRE.....	2-56
MODE .....	2-57
MSLAVE .....	2-58
MULTIL .....	2-59
MULTIL?.....	2-60
NEWLOC .....	2-61
NOISEF .....	2-62
NOOVER .....	2-63
NORMAL .....	2-64
PFCNV .....	2-65
PHANGREF .....	2-66
PHANGREF .....	2-66
PHASEM.....	2-67
PHASEM? .....	2-68
PHCONV.....	2-69
POWER .....	2-70
POWER?.....	2-71
PROGRA.....	2-73
PROGRA? .....	2-74
RANGE.....	2-75
RESOLU .....	2-76
RESULT.....	2-77

# PPA45xx communications manual

RESULT? .....	2-78
REZERO .....	2-79
SCALE .....	2-80
SCOPE?.....	2-81
SCREEN?.....	2-82
SETUP .....	2-83
SETUP? .....	2-84
SHUNT.....	2-85
SMOOTH .....	2-86
SPEED .....	2-87
START .....	2-88
STATUS?.....	2-89
STOP .....	2-90
SUSPEN .....	2-91
TAGREP .....	2-92
TEMPER .....	2-93
EMPER? .....	2-94
TORQSP.....	2-95
TORQSP? .....	2-96
USER? .....	2-97
VARCON.....	2-98
VERSIO? .....	2-99
VRMS .....	2-100
VRMS? .....	2-101
WIRING .....	2-102
ZERO.....	2-103
ZOOM.....	2-104
ZOOM? .....	2-105
Multilog Application Guide.....	2-105

Appendix A – command summary

Appendix B – configurable parameters

Appendix C – MULTILOG parameters

## 1 Using remote control

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

	Rx expects	Tx sends
RS232 USB, LAN	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 any interface, and results read back. When the IEEE488 interface is set to 'remote' the other ports are 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.

FQREF,CURRENT;POWER?

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

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 2).

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 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 1.3); baud rate is selectable via the MONITOR 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	ALA			RDV
--	--	-----	-----	-----	--	--	-----

- bit 0 RDV (result data available)  
set when 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

### 1.4 Data format

Non integer results are sent as ASCII characters in a scientific format consisting of 5 or 6 digit mantissa plus exponent:

```
+1.2345+E00
+1.23456+E00
```

For higher speed transfer a proprietary binary format can be selected which compresses the data into 4 bytes, each of which is sent with the msb set to distinguish them from ASCII control characters. The data is sent as a 7 bit signed exponent, a mantissa sign, and a 20 bit mantissa:

byte	data
1	7 bit signed exponent +63 to -64
2	bit 6 = mantissa sign bit 5:0 = mantissa bit 19:14
3	mantissa bit 13:7
4	mantissa bit 6:0

The value is coded as a binary fraction between 0.5 and 0.9999..., a multiplier of  $2^n$  and a sign ie:

$$\text{Value} = (\text{mantissa} / 2^{20}) \times 2^{\text{exponent}} \times -1^{\text{sign}}$$

value	equivalent	hex data transmitted
3.0	$0.75 \times 2^2$	0x82,0xB0,0x80,0x80
0.1	$0.8 \times 2^{-3}$	0xFD,0xB3,0x99,0xCD
-320	$-0.625 \times 2^9$	0x89,0xE8,0x80,0x80

Any valid number would have the msb of the mantissa set; any number without the msb of the mantissa set is zero.

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 *standard event status register* and clears it.

Format: \*ESR?

Arguments: none

Reply: decimal equivalent of bits in standard event status register

Example: \*ESR?  
33

Notes: The following bits in the standard event status 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, if a command is sent incorrectly and is not recognised, the CME bit will be set and the value of 33 will be returned.

**\*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,PPA4530,165-05582,2.178

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 *standard event status register*.

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.

**\*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 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 range and smoothing.

Format: \*TRG

Arguments: none

Reply: none

Example: MODE,VRMS  
\*TRG  
VRMS,SURG?

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 DSP zero error  
bit 2 – set if DSP run error  
bit 3 – not used  
bit 4 – System error, FPA initialisation  
bit 5 – System error, DSP RAM  
bit 6 – System error, DSP run  
bit 7 – System error, external RAM  
bits 8 – 14 not used  
> 15 – major system error  
bit 0 – set

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: \*TRG  
\*WAI  
POWER,PHASE1?  
Notes:



**ABORT**

**ABORT**

Function: Abort datalog  
Description: Abort datalog data acquisition.  
Format: ABORT  
Arguments: none  
Reply: none  
Example: DATALOG, RAM, 0.02  
START  
*wait for data values*  
ABORT

Notes:

**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,type,data,high,low*

Arguments: type:  
          DISABLED  
          HIGH  
          LOW  
          INSIDE  
          OUTSIDE  
          LINEAR  
data  
          1-4  
high:  
          high threshold  
low:  
          low threshold

Reply: none

Example: *ALARM1,HIGH,1,2,0*

Notes: 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,type,data,high,low*

Arguments: type:  
          DISABLED  
          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: 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,value*

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:

**APPLIC**

**APPLIC**

Function: Select application mode.

Description: Some applications require special settings within the instrument for optimum measurement

Format: *APPLIC,type,setting*

Arguments: type:

NORMAL	
PWM	(PWM Motor Drive)
BALLAST	(Lighting ballast)
INRUSH	(Inrush Current)
POWERT	(Transformer mode)
STANDB	(Standby power)
CALIBR	(Calibration)

setting:

filter 0-2 (PWM only)
0: 4kHz
1: 1kHz
2: 250Hz
speed 0-3 (ballast only)
0: fixed time
1: fast
2: medium
3: slow

Reply: none

Example: *APPLIC,POWERT*  
*APPLIC,PWM,1*

Notes:



**BANDWI****BANDWI**

Function: Select bandwidth.

Description: The analogue bandwidth of the instrument can be selected as "wide" (to 3MHz). For low noise measurements at low frequency the bandwidth can be restricted to "low" (to 40kHz). For measurements of dc in the presence of large ac signal, the bandwidth can be further restricted to "dc only" (to 10Hz).

Format: BANDWI,*phase,type*

Arguments: phase:  
                   PHASE1  
                   PHASE2  
                   PHASE3  
 type:  
                   WIDE  
                   LOW  
                   DCONLY

Reply: none

Example: BANDWI,WIDE

Notes: Only use DCONLY to improve accuracy of measurement of small dc in the presence of a large ac signal. For normal dc measurements use bandwidth = LOW.

**BANDWI?**

**BANDWI?**

Function: Read bandwidth setting.

Description: Returns a numerical value for the bandwidth setting.

Format: BANDWI,*phase?*

Arguments: phase:  
            PHASE1  
            PHASE2  
            PHASE3

Reply: 0 = WIDE  
       1 = LOW  
       2 = DCONLY

Example: BANDWI,PHASE3,LOW  
          BANDWI,PHASE3?  
          1

Notes: If independent input control has not been enabled then the setting for phase 1 is used for all phases.

**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:

**CALVER****CALVER**

- Function: Load a calibration verification string.
- Description: When calibrated, the instrument stores a text string which can be read on the front panel (press SYS and LEFT). This shows the date of calibration. Users who subsequently verify the accuracy in their own calibration facilities can enter an alternative string with the new date. The original string is not overwritten but the alternative string is displayed instead.
- Format: *CALVER,string*
- Arguments: *string* is any sequence of printable alpha numeric characters. Use the underscore character to add a space between words. CALVER without a string argument clears the previously stored string.
- Reply: none
- Example: CALVER,12\_DEC\_2008\_AMW
- Notes: As all white space is stripped from any communications string, the underscore character (ASCII 95 or 0x5F) must be used to space out the words. Underscore is shown as a space on the screen.

**CALVER?**

**CALVER?**

Function: Read back the calibration verification string.

Description: When calibrated, the instrument stores a text string which can be read on the front panel (press SYS and LEFT). This shows the date of calibration. Users who subsequently verify the accuracy in their own calibration facilities can enter an alternative string with the new date. The original string is not overwritten but the alternative string is displayed instead.

Format: CALVER?

Arguments: none.

Reply: alphanumeric string

Example: CALVER?  
12\_DEC\_2008\_AMW

Notes:

**CONFIG**

**CONFIG**

Function: Direct access of configuration parameters

Description: Sets configuration parameter for which there may not be a direct command.

Format: *CONFIG,index,data*

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,index?*  
or: *CONFIG?index*

Arguments: index is the parameter number

Reply: Value of parameter, real or integer as appropriate.

Example: *CONFIG,6?* (read phase convention)  
0  
*CONFIG,6,1* (set phase convention)  
*CONFIG,6?*  
1

Notes: The list of configurable parameters is given in the appendix.



**COUPLI**

**COUPLI**

Function: Set ac or ac+dc coupling.

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

Format: *COUPLI,phase,coupling*

Arguments: phase:  
            PHASE1  
            PHASE2  
            PHASE3  
            coupling:  
            AC+DC  
            ACONLY  
            DCONLY

Reply: none

Example: COUPLI,PHASE2,AC+DC

Notes: In multi phase applications, the coupling on phase 1 is applied to other phases unless "independent input control" is enabled.

**COUPLI?**

**COUPLI?**

Function: Read ac/dc coupling setting.

Description: Returns a numerical value for the coupling setting.

Format: *COUPLI,phase,coupling*

Arguments: phase:  
          PHASE1  
          PHASE2  
          PHASE3

Reply: 0 = AC+DC  
       1 = AONLY  
       2 = DONLY

Example: COUPLI,PHASE2,AC+DC  
          COUPLI,PHASE2?  
          0

Notes: In multi phase applications, the coupling on phase 1 is applied to other phases unless "independent input control" is enabled.

**DATALO**

**DATALO**

Function: Set up datalog

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

Format: *DATALO,function,interval,speed*

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

Reply: none

Example: DATALOG,NONVOL,10  
          DATALOG,RAM,0,HIGH

Notes: set interval to 0 to record every measurement as fast as possible. Set HIGH to select high speed mode for any combination of W, VA, VAr, pf, Vrms, Arms, and frequency. If HIGH is not sent then high speed mode is reset.

**DATALO?**

**DATALO?**

Function: Read back datalog results

Description: Return datalog values, one record per line, or the number of lines available

Format: DATALO,*start,records?*  
 DATALO,0?  
 DATALO,LINES?

Arguments: start:  
           first record to return  
 records:  
           number of records to return  
 0:  
           return all new records since last read

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,10  
 START  
 wait for datalog  
 STOP  
 DATALOG,LINES?  
 30  
 DATALOG,21,3?  
 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 the available lines of data

**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 harmonic series data available  
bit6 integration data available  
bit7 datalog data available

Example: SPEED,SLOW  
\*TRG  
DAV?  
0  
DAV?  
0  
DAV?  
0  
DAV?  
3 (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

Reply: none

Example: *DAVER, 1*  
set bit 0 in status byte when new data is available

Notes: default value is 2:  
bit 0 of status byte is set whenever 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:

## DISPLAY

## DISPLAY

- Function: Set the display page
- Description: Selects the page on the display so that the zoom data can be used for alarms.
- Format: DISPLAY,*page*
- Arguments: page:  
PHASE1  
PHASE2  
PHASE3  
SUM  
NEUTRAL  
TOTAL  
FUNDAMENTAL  
VOLTAGE  
CURRENT
- Reply: None
- Example: DISPLAY,FUNDAMENTAL
- Notes: VOLTAGE is the same as TOTAL;  
CURRENT is the same as FUNDAMENTAL.  
They refer to the multiphase display modes.



**DISPLAY?**

**DISPLAY?**

Function: Read the displayed data

Description: Returns all the values presently on the screen.

Format: DISPLAY?

Arguments: none

Reply: Multiple floating point values separated by commas

Example: DISPLAY?

Notes:

**EFFICI**

**EFFICI**

Function: Set efficiency calculation

Description: Selects the data to be used for the efficiency calculation.

Format: *EFFICI,formula*

Arguments: formula:  
0 – disabled  
1 – phase 1 / phase 2  
2 – phase 2 / phase 1  
3 – slave / master  
4 – master /slave  
5 – mechanical sum  
6 – sum / mechanical  
7 – phase 3 / sum  
8 – sum /phase

Reply: none

Example: EFFICIENCY,2

Notes:

**EFFICI?**

**EFFICI?**

Function: Read efficiency result

Description: Reads back the total and fundamental efficiency results.

Format: EFFICI?

Arguments: none

Reply: 2 data values separated by commas:  
total, fundamental

Example: EFFICI?  
data returned

Notes:

**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.

**FQLOCK**

**FQLOCK**

Function: Lock frequency.

Description: Set the technique for determining the frequency for analysis.

Format: *FQLOCK,value,frequency*

Arguments: value:  
ON  
OFF  
NORMAL  
CONSTANT  
DYNAMIC  
  
frequency (optional)  
CONSTANT -enter frequency  
DYNAMIC -enter minimum frequency

Reply: none

Example: FQLOCK,ON  
FQLOCK,Dynamic,100

Notes: FQLOCK,CONSTANT  
Without an argument locks the frequency to the present value.

ON is the same as CONSTANT  
OFF is the same as NORMAL

When Dynamic is selected the minimum frequency can be set between 0.010Hz (10mHz) and 500Hz.

**FQREF**

**FQREF**

Function: Set frequency reference.

Description: Select the channel to be used for measuring the frequency.

Format: *FQREF,phase*  
*FQREF,channel*  
*FQREF,phase,channel*

Arguments: channel:  
                 voltage  
                 current  
                 Speed  
                 Ac Line

                phase:  
                         PHASE1  
                         PHASE2  
                         PHASE3

Reply: none

Example: FQREF,CURRENT

Notes: Measured phase is always referred to phase 1 voltage no matter what channel is selected to measure the frequency, unless phase 1 is not active (eg phase 2 only mode).

**FREQFI**

**FREQFI**

Function: Set the frequency filter

Description: Selects a filter to be applied to the data used for frequency measurement to help synchronise in noisy environments.

Format: FREQFI,*value*

Arguments: value:  
ON  
OFF

Reply: none

Example: FREQFI,ON

Notes: The filter is applied only to the data used for frequency measurement and does not change the data used for the measurements.

**FREQUE**

**FREQUE**

Function: Set the analysis frequency

Description: Sets the analysis frequency in Hz for frequency lock mode.

Format: *FREQUE,frequency*

Arguments: frequency in Hz

Reply: none

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

Notes: Lock the frequency with FQLOCK,ON before sending the desired frequency with the FREQUE command.



**FSD?**

**FSD?**

Function: Read the full scale of all input channels at once or that of an individually selected input channel.

Description: Returns the full scale value for all channels or that of a single selected channel.

Format: FSD?  
FSD,CH?

Arguments: None  
CH1, CH2, CH3, CH4, CH5, CH6

Reply: Up to six data values separated by commas

Example 1: FSD?  
Data returned, data returned, data returned, data returned, data returned

Example 2: FSD,CH1?  
Data returned

Notes: Number of channels that can be read and the number of data values returned is dependent on the number of phases selected in the instruments settings.  
CH1 = PH1: Voltage Input  
CH2 = PH1: Current Input  
CH3 = PH2: Voltage Input  
CH4 = PH2: Current Input  
CH5 = PH3: Voltage Input  
CH6 = PH3: Current Input

**HARMON****HARMON**

Function: Set harmonic analyser mode.

Description: Set harmonic analyser mode and parameters.

Format: *HARMON,para,harmonic,max,stepsize*

Arguments: para:

THDD	difference formula THD
THDS	harmonic series THD
TIF	Telephone Influence Factor
THF	Telephone Harmonic Factor
TDD	Total Demand Distortion
TRD	Total Rated Distortion
HPHASE	harmonic phase
INTERH	Interharmonic sweep
HRMS	Harmonic RMS
PH-PH	Phase to Phase harmonics

harmonic:  
individual harmonic for display

max:  
length of harmonic series

stepsize:  
frequency step size (0.5Hz – 100Hz)

Reply: none

Example: HARMON,TRD  
HARMON,THDS,3,50  
HARMON,INTERH,3,7500,20

Notes: It is not necessary to send any arguments, but if any are sent they must be in the specified order. PH-PH command is reset by any THD command.

## PPA45xx communications manual

The maximum value for length of harmonic series is as follows:

100 for harmonic factor, harmonic RMS, TIF, THF, TDD and TRD.

125 for Harmonic Series and Series Harmonic Phase.

9999 for Interharmonic sweep.

The stepsize argument only applies to Interharmonic Sweep.

**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,*phase*?  
or: HARMON,SERIES?  
or: HARMON,*phase*,SERIES?

Arguments: phase:  
          PHASE1  
          PHASE2  
          PHASE3  
          NEUTRAL  
          PHASES

Reply: 11 data values separated by commas:  
freq,mag1,mag2,hmag1,hmag2,h%1,  
h%2,thd%1,thd%2,hphase1,hphase2  
or: magnitude and percentage for each harmonic, one channel per line  
or: magnitude and phase for each harmonic, one channel per line

Example: HARMON,PHASE2?  
data returned

Notes: HARMON? waits for next unread data.

**HOLD**

**HOLD**

Function: Set data hold

Description: Turns data hold on or off. Useful for reading data from different phases without it being changed between reads.

Format: HOLD,state

Arguments: State:  
ON  
OFF

Reply: none

Example: HOLD,ON  
POWER,PHASE1,WATTS?  
POWER,PHASE2,WATTS?  
POWER,PHASE3,WATTS?  
HOLD,OFF

Notes:

**INPUT**

**INPUT**

Function: Set input mode

Description: Selects the input type of the instrument

Format: *INPUT,channel,type*

Arguments: channel:  
            CH1  
            CH2  
          type:  
            INTERN  
            EXTATT  
            EXTSHU

Reply: none

Example: INPUT,CH1,EXTSHU

Notes: CH1 applies to all voltage channels  
       CH2 applies to all current channels

**INTEGR**

**INTEGR**

Function: Set integrated power mode.

Description: Set integrated power mode, whether the integration for Watts and current use signed or unsigned values, and whether accumulated or averaged values are computed.  
Also sets up run time for integration over a specific interval.

Format: *INTEGR,type,display*  
*INTEGR,RUNTIM,hours,minutes*

Arguments: type:  
          SIGNED  
          MAGNITUDE  
display:  
          TOTAL  
          AVERAGE  
hours:  
          integer  
minutes:  
          integer

Reply: none

Example: INTEGR,MAGNITUDE,TOTAL

Notes:

**INTEGR?**

**INTEGR?**

Function: Read integrated power mode.

Description: Read integrated power mode for the selected phase.

Format: `INTEGR,phase?`

Arguments: phase:  
          PHASE1  
          PHASE2  
          PHASE3  
          PHASES  
          SUM

Reply: 13 values separated by commas  
          time,Wh,WH.f,VAh,VAh.f,VArh,Varh.f  
          pf,pf.f,V,V.f,Ah,Ah.f

Example: START  
          wait for integration time  
          INTEGR,PHASE1?  
          data returned

Notes: INTEGR? without specifying the phase returns the appropriate single phase data.



**KEYBOA**

**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.

**LCR**

**LCR**

Function: Set LCR meter mode.  
Description: Set LCR mode and conditions.  
Format: *LCR,parameter*  
Arguments: parameter:  
            AUTO  
            CAPACITANCE  
            INDUCTANCE  
            IMPEDANCE  
Reply: none  
Example: LCR,IMPEDA  
Notes:

**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,*phase?*

Arguments: phase:  
PHASE1  
PHASE2  
PHASE3  
PHASES

Reply: 11 data values separated by commas:  
freq, Vmag, Amag, impedance,  
phase, R, C, L,  $\tan\delta$ , Qf, reactance

Example: LCR,IMPEDA  
LCR,PHASES?  
data returned

Notes: LCR? waits for next unread data.  
LCR? without specifying the phase returns  
the appropriate single phase data.

**LOWFRE**

**LOWFRE**

Function: Set low frequency mode

Description: Sets the low frequency option for extending the measurement window.

Format: LOWFRE,*value*

Arguments: value:  
            ON  
            OFF

Reply: none

Example: LOWFRE,ON

Notes: LOWFRE is mainly used for measuring low frequencies (<5 Hz). However, as it applies digital filtering, it may also be useful when analysing any signals below a few hundred Hertz.

**MODE**

**MODE**

Function: Set mode

Description: Sets the basic operating mode of the instrument.

Format: MODE,*type*

Arguments: type:  
POWER (power meter)  
INTEGR (integrator)  
HARMON (harmonic analyser)  
RMS (rms voltmeter)  
LCR (LCR meter)  
SCOPE (oscilloscope)  
PHASEM (phase meter)

Reply: none

Example: MODE,LCR

Notes:

**MSLAVE**

**MSLAVE**

Function: Set master/slave mode

Description: Enables the instrument to synchronise with a second instrument to simultaneously measure up to 6 phases.

Format: MSLAVE,*type*

Arguments: type:  
          DISABLE  
          MASTER  
          SLAVE

Reply: none

Example: MSLAVE,MASTER

Notes:

**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 using the MULTIL? command.

Format: *MULTILOG,index,phase,function*

Arguments: index:  
           0          clear all  
           1-64      select data 1-64  
 phase:  
           1-3      phase 1-3  
           4         sum  
           5         neutral  
 function:  
           1-99      see appendix C

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:

For further information and assistance with the Multilog application please go to page 2-105 where you will find an application guide to assist with this function.

**MULTIL?**

**MULTIL?**

Function: Reads multi string reply

Description: Waits for data to be available (if required) then returns selected results. Either a single string or multiple string replies can be selected.

Format: MULTIL?  
MULTIL,number?

Arguments: number: The required number of data string replies

Reply: A single reply string containing up to 64 data values as selected by the MULTIL command.

Multiple reply strings each containing the same number of data values (maximum of 64) as selected by the MULTIL command.

Example: MULTIL,0  
MULTIL,1,1,2 (phase 1 Watts)  
MULTIL,2,2,2 (phase 2 Watts)  
MULTIL,3,4,3 (sum VA)

MULTIL?  
In the above example a single string reply containing 3 data values is returned.

MULTIL,10?  
In the above example 10 data strings are returned, each string containing 3 data values.

Notes:  
For further information and assistance with the Multilog application please go to page 2-105 where you will find an application guide to assist with this function.



**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:              Data as per returned parameter query. ie from power, harmonics etc.

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

**NOISEF**

**NOISEF**

Function: Sets the noise filter.  
Description: Sets noise filter to value sent in string between 1KHz and 250KHz.  
Format: NOISEF,[PHASEx],value,frequency

Arguments:

[PHASEx]:  
Phase1  
Phase2  
Phase3

Value:  
ON  
OFF

frequency:  
Between: 1000 – 250000

Reply: none

Example: NOISEF,PHASE1,ON,1500

Notes: Applies a digital filter for use in high noise environments. When in independent mode use [PHASEx] command to set noise filter on individual phases. [PHASESx] command is not required in any other wiring mode.

**NOOVER**

**NOOVER**

Function: Disable overranging

Description: Prevents an overrange error from blanking out results in manual ranging.

Format: NOOVER,*value*

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.

**NORMAL**

**NORMAL**

Function: Sets the Normalise reference to Current or Voltage.

Description: Sets the Reference for the NORMALISE function. Press ZERO on the instrument to action the function.

Format: NORMAL,reference

Arguments: Reference:  
CURRENT  
VOLTAGE

Reply: none

Example: NORMAL,VOLTAGE  
NORMAL,CURRENT  
NORMALISE,VOLTAGE  
NORMALISE,CURRENT

Notes: The "normalise" function adjusts the scale factors on each current channel so that they read the same as phase 1. The reference can be either the current measured on phase 1 or if there is a reference CT it can be connected to the external input of phase 1 voltage and used as a reference.

**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,*type*

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

**PHANGREF**

**PHANGREF**

Function: Set phase angle reference.

Description: Select phase angle reference to current or voltage.

Format: PHANGREF,*reference*

Arguments: reference:  
              Current  
              Voltage

Reply: none

Example: PHANGREF,current  
          PHANGREF,voltage

Notes:

**PHASEM**

**PHASEM**

Function: Set phase meter mode.

Description: Select phase meter mode and reference.

Format: PHASE,*reference*

Arguments: reference:  
          CH1 ratio = ch2/ch1  
          CH2 ratio = ch1/ch2

Reply: none

Example: PHASEM,CH2

Notes:

**PHASEM?**

**PHASEM?**

Function: Phase meter query

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

Format: PHASEM?  
PHASEM,*phase?*

Arguments: phase:  
PHASE1  
PHASE2  
PHASE3  
PHASES?

Reply: 5 data values separated by commas  
freq,mag1,mag2,dB,phase

Example: PHASEM,CH1  
PHASEM,PHASE1?  
data returned

Notes: The phase convention can be set to 0° to -360°, 0° to +360°, or +180° to -180° in the SYSTEM menu or using PHCONV command.  
PHASEM? without specifying the phase returns the appropriate single phase data.



**PHCONV****PHCONV**

Function: Set phase convention and the harmonic angle.

Description: Set phase convention and optionally the harmonic angle.

Format: PHCONV,*convention,angle*

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

Angle:  
           Cosine  
           Sine

Reply: none

Example: PHCONV, -360  
           PHCONV,180  
           PHCONV,180,cosine

Notes: 0 to -360 degrees is usually used for power analysis applications.

The Harmonic Angle argument is optional so does not have to be specified. However, to update the Harmonic phase angle argument the phase convention must be included in the command. See examples above. The default setting in the SYS menu is Cosine.

**POWER**

**POWER**

Function: Set up power analyser mode.

Description: Configure power analyser with sum current display type

Format: *POWER,sum type*

Arguments: sum type:  
TOTAL  
AVERAGE

Reply: none

Examples: POWER,TOTAL

Notes:

**POWER?**

**POWER?**

Function: Read power analyser results

Description: Reads back latest power analyser results.  
Sets power analyser mode.  
Waits for next unread data if necessary.  
Clears new data available status bit.

Format: *POWER,phase,results?*

Arguments: phase:  
           PHASE1  
           PHASE2  
           PHASE3  
           PHASES  
           SUM  
           NEUTRAL (current only)  
 results:  
           WATTS  
           VOLTAGE  
           CURRENT  
           VECTORS  
           RMS  
           WVA  
           PH-PH

Reply: WATTS:  
           freq,W,W.f,VA,VA.f,VAR,VAR.f,pf,pf.f,  
           Wdc,W.h  
 VOLTAGE or CURRENT:  
           freq,rms,mag,dc,phase,pk,cf,mean,  
           form factor,harm  
 VECTORS:  
           freq,vmag1,vlag1,amag1,alag1.....  
 RMS:  
           freq,vrms1,vdc1,arms1,adc1.....  
 WVA:  
           freq,w1,vrms1,arms1,w2.....

## PPA45xx communications manual

PH-PH:

freq,rms1,mag1,lag1,rms2...

Example:

POWER,VECTORS?

data returned

Notes:

POWER? without specifying the phase returns the appropriate single phase data. PHASES returns the data for all valid phases 1-3.

**PROGRA**

**PROGRA**

Function: Access non volatile program stores.

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

Format: *PROGRA,function,number*

Arguments: function:  
              RECALL  
              STORE  
              DELETE  
              number  
              0-100

Reply: none

Example: *PROGRA,RECALL,13*

Notes: Number 0 represents factory default, which can only be recalled.

**PROGRA?**

**PROGRA?**

Function: Identify current program or list all stored programs.

Description: Sending the argument FILES? - Lists all stored programs. The reply includes the location, file name and date saved for each program.  
Sending the argument NAME? - Displays the name of the last program to be loaded or recalled.

Format: PROGRA

Arguments: FILES?  
NAME?

Reply: text string

Example: PROGRA,FILES?  
2,PCIS,21/11/2017  
3,,21/11/2017  
10,remote program,11/01/18

PROGRA,NAME?  
factory default

PROGRAM,NAME?  
Remote program

Notes: If a program is stored but not given a name the return string will display no data for the name. See example above.

Only the first six digits of the command are required so PROGRA and PROGRAM are both valid, both return the same data.

**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:  
          range number 1-9

Reply: none

Example: RANGE,CH2,MANUAL,4

Notes: CH1 sets the voltage range  
CH2 sets the current range  
Refer to the user manual for the range corresponding to each range number

**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) or 20 bit (BINARY).

Format: RESOLU,*format*

Arguments: format:  
                   NORMAL          (5 digit mantissa)  
                   HIGH             (6 digit mantissa)  
                   BINARY          (compressed format)

Reply: none

Example: RESOLU,HIGH

Notes: Data format for NORMAL is:  
           [-]1.2345E[-]00  
       Data format for HIGH is:  
           [-]1.23456E[-]00  
       The sign of the mantissa and exponent are only sent if negative shown as [-] in the above examples  
       BINARY format encodes each non-integer value in a proprietary 4 byte format for higher speed data transfer.

[Further notes on data format are included in section 1.4]



**RESULT**

**RESULT**

Function: Access non volatile results stores.  
Description: Recall, store or delete non-volatile results.  
Format: *RESULT,function,number*  
Arguments: function:  
            RECALL  
            STORE  
            DELETE  
          number  
            1-20  
Reply: none  
Example: RESULT,RECALL,13  
Notes: There are 3 types of result: normal, harmonic and scope. Harmonic and scope results occupy 3 locations each.

**RESULT?**

**RESULT?**

Function: Identify used result stores.

Description: Reads a directory of the 20 non-volatile result locations.

Format: RESULT?

Arguments: none

Reply: 20 integers separated by commas

Example: RESULT?  
0,0,1,3,-1,-1,0,2,-1,-1,0,0,0,0,0,0,0,0,0,0

Notes: 0 = empty  
1 = normal result  
2 = harmonic result  
3 = scope result  
-1 = continuation of previous

**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  
            factor:  
            multiplying scale factor

Reply: none

Example: SCALE,CH2,10

Notes: CH1 sets the scale for all voltage channels  
        CH2 sets the scale for all current channels

**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 an external current shunt to be divided into the measured voltage for a given input channel.

Format: SHUNT,*channel,resistance*

Arguments: channel:  
            CH1  
            CH2  
            resistance:  
            shunt resistance in Ohms

Reply: none

Example: SHUNT,CH1,10

Notes: The shunt value is set for all current channels

**SMOOTH**

**SMOOTH**

Function: Select the smoothing

Description: Sets the filter time constant and dynamic response.

Format: *SMOOTH,type,dynamics*

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,value,window*

Arguments: value:  
VFAST  
FAST  
MEDIUM  
SLOW  
VSLOW  
WINDOW

Reply: none

Example: SPEED,SLOW  
SPEED,WINDOW,0.1

Notes: The window size argument is only needed for the WINDOW option

**START**

**START**

Function: Start datalog  
Description: Initiate datalog data acquisition.  
Format: START  
Arguments: none  
Reply: none  
Example: DATALOG, RAM, 0.02  
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,*channel?*  
STATUS?*channel*

Arguments: channel:  
CH1  
.  
.  
CH6

Reply: range number,range text,over/under/ok  
1-16  
range as per RANGE command  
OVER if overflow  
LOW if underflow  
OK if in range

Example: STATUS,CH1?  
6,300V,OK  
STATUS?  
OK

Notes:

**STOP**

**STOP**

Function: Stop datalog  
Description: Stop datalog data acquisition.  
Format: STOP  
Arguments: none  
Reply: none  
Example: DATALOG, RAM, 0.02  
START  
*wait for data values*  
STOP  
*read data values*

Notes:

**SUSPEN**

**SUSPEN**

Function: Suspend data acquisition.

Description: Disable the data acquisition to maximise the communication speed.

Format: SUSPEN,*value*

Arguments: value:  
            ON  
            OFF

Reply: none

Example: FAST,ON  
          SUSPEN,ON  
          MULTILOG?  
          SUSPEN,OFF  
          FAST,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,*on/off*

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



**TEMPER**

**TEMPER**

Function: Set up temperature measurement

Description: Set scaling and offset for a temperature sensor connected to the torque input (power transformer application mode)

Format: *TEMPER,type,scalefactor,offset*

Arguments: type:  
                  DISABLED  
                  CENTIG  
                  FARHEN  
          scale:  
                  multiplying factor in degrees/Volt  
          offset:  
                  additive zero in Volts

Reply: none

Example: TEMPER,CENTIG,5,-2  
          sensor scaling = 5°C/V  
          0V = 10°C

Notes:

**EMPER?**

**TEMPER?**

Function: Read the temperature

Description: Returns the measured temperature from a sensor connected to the torque input

Format: TEMPER?

Arguments: none

Reply: single data value

Example: TEMPER?  
data returned

Notes:

**TORQSP****TORQSP**

Function:	Set up torque and speed measurement
Description:	Set scaling and offset for torque and speed measurements. Pulsed input has a value for the number of pulses per revolution.
Format:	TORQSP, <i>type,scale1,scale2</i> TORQSP,OFFSET, <i>offset1,offset2</i>
Arguments:	type: DISABLED ANALOG PULSED (SPEED) OFFSET scale1 and scale 2 multiplying factor in Nm/V or rpm/V pulses/rev offset1 and offset2 zero level in V
Reply:	none
Examples:	TORQSP,PULSED,10,50 speed measured by pulse torque scaling = 10Nm/V 50 pulses/revolution  TORQSP,ANALOG,10,1
Notes:	If type = ANALOG then speed scaling is in rpm/V, if type = PULSED then speed scaling is pulses/rev Torque scaling is always Nm/V

**TORQSP?**

**TORQSP?**

Function: Read the mechanical power, torque and speed

Description: Returns measured mechanical power value along with the torque and speed values

Format: TORQSP?

Arguments: none

Reply: 3 data values separated by commas:  
power, torque, speed

Example: TORQSP?  
data returned

Notes: Mechanical power displayed in Watts  
Torque displayed in Nm  
Speed displayed in rpm

**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  
          PPA4530 #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,*type*

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

**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 type:  
0 – normal (30A)  
2 – low current version (10A)  
4 – high current version (50A)

Examples: VERSION?  
KQ1306,0,1.10,1.10,1.10,1.01

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.  
 Waits for next unread data if necessary.  
 Clears new data available status bit.

Format: *VRMS,phase,results?*

Arguments: results:  
           RMS  
           MEAN  
           SURGE  
 phase:  
           PHASE1  
           PHASE2  
           PHASE3  
           PHASES

Reply: RMS:  
           6 data values separated by commas  
           Vrms,Arms,Vdc,Adc,Vac,Aac  
 MEAN:  
           6 data values separated by commas  
           Vrms,Arms,Vmean,Amean,Vff,Aff  
 SURGE:  
           8 data values separated by commas  
           Vrms,Arms,Vpk,Apk,Vcf,Acf,  
           Vsurge1,Asurge

Example: VRMS,PHASE1,RMS?

Notes: VRMS? without specifying the phase  
 returns the appropriate single phase data.

**WIRING**

**WIRING**

Function: Select wiring mode.

Description: Set wiring mode for computation of SUM and neutral data.

Format: WIRING,*type*

Arguments: type:  
 SINGLE (single ph 1)  
 2PHASE (2 ph 2 wattmeter)  
 3PH2WA (3 ph 2 wattmeter)  
 3PH3WA (3 ph 3 wattmeter)  
 INDPH3 (3 ph 2 wattmeter + ph3)  
 PHASE1 (single ph 1)  
 PHASE2 (single ph 2)  
 PHASE3 (single ph 3)  
 INDEP (independent)  
 3PH3WA,DELTAS (Delta – Star)  
 3PH3WA,PPRMS (PH-PH RMS)  
 3PH3WA,PPMEAN (Rectified mean)  
 3PH3WA,STARDE (Star – Delta)

Reply: none

Examples: WIRING,PHASE2

Notes: WIRING,SINGLE is the same as WIRING,PHASE1

**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). Resets the integration data and timer if in power integration mode.

Format: ZERO  
ZERO,DELETE

Arguments: none

Reply: none

Example: ZERO

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 – normal</li> <li>1 – 2 line display (zoom level 1)</li> <li>2 – single line display (zoom level 2)</li> <li>3 – single line display (zoom level 3)</li> </ul> <p>data1:</p> <ul style="list-style-type: none"> <li>first data (zoom level 1)</li> <li>or data for single line (zoom level 2)</li> </ul> <p>data2-4:</p> <ul style="list-style-type: none"> <li>other data (zoom level 1)</li> </ul> <p>data consists of line number for channel 1 or line number + 64 for channel 2</p>
Reply:	None
Example:	<p>VRMS</p> <p><i>ZOOM,1,1,65</i> (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:

0 – normal

1 – 2-4 value display (zoom level 1)

2 – single line display (zoom level 2)

3 – single line display (zoom level 3)

data1-4:

zoom data

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

Example:

ZOOM?

1,1,65,0,0 (level 1, ch1 rms, ch2 rms)

Notes:

## **Multilog Application Guide** **Configuring the N4L PPA Power Analyzer for** **Data logging**

The Multilog (MULTIL) command provides an excellent method for data logging up to 64 parameters of information via one query command - MULTIL?

The instrument will return a comma-separated string which relates to the MULTIL,X,X,X setup commands previously entered by the relevant communication method. This enables the system to send one query and return up to 64 different parameters, from different phases in one response.

### **Step 1.**

Reset "MULTILOG" using the **MULTIL,0** command

This will clear any previously entered Multilog parameters and ensure the instrument does not return unwanted results.

### **Step 2.**

Set up the Multilog parameters

The format of the Multilog command is as follows

MULTILOG, Index, Phase, function

Index is the order in which the value is returned (Effectively allocating a "slot" for the parameter in the returned string)

Phase is the phase (PH1,PH2,PH3 etc) from which the result should be acquired.

Function is the parameter type (eg. Watts, VAr, Frequency etc) of the return.

## PPA45xx communications manual

The Function ID is chosen from Appendix C which is a continually growing list due to firmware upgrades of the power analyzers at N4L, at present the PPA4500 has 93 possible functions:

Function	Measurement	Notes
1	frequency	
2	watts	
3	VA	
4	VAr	
5	power factor	
6	fundamental watts	
7	fundamental VA	
8	fundamental VAr	
9	fundamental PF	
10	harmonic watts	
11	harmonic watts %	
12	impedance	
13	resistance	

Example extract from the Multilog function list

**Required Parameters**

Order parameter to be returned within string	Phase (channel) of data returned	Parameter required
1	1	Frequency
2	1	Watts Phase 1
3	2	Watts Phase 2
4	3	Watts Phase 3
5	1	RMS Voltage Phase 1
6	2	RMS Voltage Phase 1
7	3	RMS Voltage Phase 1

**MULTILOG Pattern**

Command	Index	Phase	Function
MULTIL,	1	1	1
MULTIL,	2	1	2
MULTIL,	3	2	2
MULTIL,	4	3	2
MULTIL,	5	1	50
MULTIL,	6	2	50
MULTIL,	7	3	50

Command strings to sent, reference the above Multilog pattern;

MULTIL,0 // clears Multilog

MULTIL,1,1,1 // set Frequency as parameter 1

MULTIL,2,1,2 // set Phase 1 Watts as parameter 2

MULTIL,3,2,2 // set Phase 2 Watts as parameter 3

MULTIL,4,3,2 // set Phase 3 Watts as parameter 4

MULTIL,5,1,50 // set Phase 1 RMS Voltage as parameter 5

MULTIL,6,2,50 // set Phase 2 RMS Voltage as parameter 6

MULTIL,7,3,50 // set Phase 3 RMS Voltage as parameter 7



**Step 3.**

Send Multil query and read return string.

MULTIL? // returns a comma separated string as

Example return string:

5.0000E1,	2.4500E2,	2.4320E2,	2.5421E2,	1.0232E3,	1.0152E3,	1.0546E3
↑	↑	↑	↑	↑	↑	↑
Frequency	PH1 Watts	PH2 Watts	PH3 Watts	PH1 RMS Volt	PH2 RMS Volt	PH3 RMS Volt

Appendix – command summary

COMMAND SUMMARY

## PPA45xx communications manual

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	
*SRE?	single integer data value
*STB?	single integer data value
*TRG	
*TST?	single integer data value
*WAI	
ABORT	
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	
APPLIC,type,setting	
BANDWI,phase,type	
BEEP	
BLANKI,on/off,threshold	
CALVER,string	
CALVER?	string
CONFIG,parameter,data	
CONFIG,parameter?	single integer or real data value
COUPLI,phase,coupling	
DATALO,func,interval,speed	
DATALO,LINES?	single integer
DATALO,0?	index,time,data... one record per line
DATALO,start,records?	index,time,data... one record per line
DAV?	single integer data value
DAVER,value	
DAVER?	single integer data value
DISPLAY,page	
DISPLAY?	multiple real data values
EFFICI,type	
EFFICI?	total efficiency, fundamental efficiency
FAST,on/off	
FQLOCK,on/off	

## PPA45xx communications manual

FQREF,phase,channel	
FREQFI,on/off,filter	
FREQUE,frequency	
FSD?	Single or multiple real data values
HARMON,para,h,hmax	
HARMON,phase?	freq,mag1,mag2,hmag1,hmag2,h1,h2, thd1,thd2,hphase1,hphase2
Or	
HARMON,phase,SERIES?	mag,%, x n harmonics
Or	mag,phase, x n harmonics
HOLD,on/off	
INPUT,channel,type	
INTEGR,type,display	
INTEGR,RUNTIM,hours,mins	
INTEGR,phase?	Time,Wh,Wh.f, Varh,Varh.f,Vah,Vah.f, pf,pf.f,Vav,Vav.fAh,Ah.f
KEYBOA,value	
LCR,conditions,param,head	
LCR,phase?	Freq,mag1,mag2,impedance,phase,R, L,C (series),R,L,C (parallel),tan $\delta$ ,Q
LOWFRE,on/off	
MODE,type	
MSLAVE,type	
MULTILOG,index,phase,func	
MULTILOG?	1-30 floats as selected
PFCNV,convention	
PHASEM,ratio	
PHASEM,phase?	Freq,mag1,mag2,dB,phase
PHCONV,convention	
POWER,sum A	
POWER,PHASE,WATTS?	Freq,W,W.f, VA,VA.f,Var,Var.f,pf,pf.f, Wdc,W.h
POWER,PHASE,VOLTAGE?	Freq,rms,mag,dc, $\phi$ ,peak,cf,mean,ff, harmonic
POWER,PHASE,CURRENT?	Freq,rms,mag,dc, $\phi$ ,peak,cf,mean,ff, harmonic
POWER,PH-PH?	Freq,rms1,mag1, $\phi$ 1,rms2,mag2, $\phi$ 2, rms3,mag3, $\phi$ 3
POWER,RMS?	Freq,vrms1,vdc1,arms1,adc1,vrms2, vdc2,arms2,adc2,vrms3,vdc3, arms3, adc3
POWER,VECTORS?	Freq,mag1, $\phi$ 1,mag2, $\phi$ 2,mag3, $\phi$ 3, mag4, $\phi$ 4,mag5, $\phi$ 5,mag6, $\phi$ 6

## PPA45xx communications manual

POWER,WVA?	Freq,w1,vrms1,arms1,w2,vrms2,arms2,w3,vrms3,arms3
PROGRAM,function,number PROGRAM?	CR terminated text string
RANGE,ch,ranging,range RESOLU.format RESULT,function,number RESULT REZERO	multiple integers
SCALE,channel,factor SCALE,channel? SCOPE,PHASE,v/a?	Single real data value Range, trigger, 250 signed integer values
SHUNT,channel,resistance SHUNT,channel?	Single real data value
SMOOTH,type,dynamics SPEED,value>window START	
STATUS,channel? STOP	Range number,range text,over/low/ok
STREAM,enable>window STREAM,disable STREAM?	Data, data, data, data, data, .....
SUSPEN,on/off TAGREP,on/off TEMPER,type,scale,offset TEMPER?	single real data value
TORQSP,type,tscale,sscale TORQSP,OFFSET,toff,soff TORQSP?	mechanical power, torque, speed
USER? VARCON,convention VERSION?	3 CR terminated text strings datecode,cpu,dsp,fpga,boot
VRMS VRMS,PHASE,RMS? VRMS,PHASE,MEAN? VRMS,PHASE,SURGE?	rms1,rms2,dc1,dc2,ac1,ac2 rms1,rms2,mean1,mean2,ff1,ff2 pk1,pk2,cf1,cf2,surge1,surge2
WIRING,configuration ZERO ZERO,DELETE ZOOM,level,d1,d2,d3,d4 ZOOM?	level,d1,d2,d3,d4

# PPA45xx communications manual

## calibration commands

CALAPP	
CALCOM,freq	
CALFIL,index,value	
CALFIL?	Six real data values
CALFRQ,index,freq	
CALFRQ?	Seven real data values
CALIBR,index,value,inputs	
CALIBR?	Single integer data value
CALIDS,string	
CALIDS?	String
CALJIG,value	
CALMOD,value	
CALPHA,index,inputs	
CALRES	
CALSAV,password	
CALSYS,index,value,inputs	
CALSNO,serial number	
CALSTR,string	
CALSTR?	String
CALTQS,index,value	
CALTQS?	Four real data values
CALVER,string	
CALVER?	String

# PPA45xx communications manual

## Appendix B – Configurable parameters

All parameters can be accessed using the CONFIG command:

<i>CONFIG,number,parameter</i>		
<i>number</i>	Function	<i>parameter</i>
1	Operating mode, (sets Main Mode)	0=RMS Voltmeter 1=Phase Meter 2=Power Analyser 3=Impedance Analyser 4=Power Integrator 5=Harmonic Analyser 7=Oscilloscope
2	Resolution, (remote options – digit resolution)	0=Normal 1=High 2=Binary
3	Master/slave, (Aux control)	0=Disabled 1=Master 2=Slave
4	Autozero manual or auto, (System options)	0=Auto 1=Manual
6	Phase convention, (System options)	0=-180° to +180° 1=0° to -360° 2=0° to +360°
7	Frequency lock on/off, (Acquisition advance options)	0=Off 1=On 2=Dynamic
8	Graph, (System options)	0=Dots 1=Lines

## PPA45xx communications manual

- 9           Keyboard beep on/off, (System options)  
              0=Off  
              1=On
- 10          Ignore overload, (Acquisition advance options)  
              0=Off  
              1=On
- 11          Low frequency mode, (Acquisition control)  
              0=Off  
              1=On
- 12          Window size, (Acquisition control, speed-window)  
              0=mS  
              1=Sec's
- 13          Speed, (Acquisition control or Phase meter)  
              0=Very Slow  
              1=Slow  
              2=Medium  
              3=Fast  
              4=Very Fast  
              5=Window
- 14          Smoothing (Acquisition Control or Phase Meter)  
              0=Normal  
              1=Slow  
              2=None
- 15          Smoothing Response (Acquisition Control or Phase meter)  
              0=Auto reset  
              1=Fixed time
- 16          Baud rate, (Remote options , RS232)  
              0=38400  
              1=19200  
              2=9600  
              3=1200
- 18          LAN IP address nibble 3, (Remote options - LAN - enter figure as required)
- 19          LAN IP address nibble 2, (Remote options - LAN - enter figure as required)
- 20          LAN IP address nibble 1, (Remote options - LAN - enter figure as required)
- 21          LAN IP address nibble 0, (Remote options - LAN - enter figure as required)



## PPA45xx communications manual

- 22      Independent ranging, (System options)  
          0=Disabled  
          1=Enabled
- 24      Enable channel 1, (Range - voltage input)  
          1=Internal  
          3=External Attenuator
- 25      Enable channel 2, (Range - current input)  
          1=Internal  
          2=External Shunt
- 26      Input range channel 1, (Range - minimum range voltage)  
          0=300mV  
          1=1V  
          2=3V  
          3=10V  
          4=30V  
          5=100V  
          6=300V  
          7=1kV  
          8=3KV
- 27      Input range channel 2, (Range - minimum range current)  
          0=30mA  
          1=100A  
          2=300mA  
          3=1A  
          4=3A  
          5=10A  
          6=30A  
          7=100A  
          8=300A
- 28      Input ranging channel 1, (Range - autoranging voltage)  
          0=Full Autorange  
          1=Range up only  
          2=Manual
- 29      Input ranging channel 2, (Range - autoranging current)  
          0= Full Autorange  
          1=Range up only  
          2=Manual



## PPA45xx communications manual

- 42        Zoom level, (Main display)  
          0=Zoom -  
          1=Zoom +  
          2=second Zoom +  
          3=Third Zoom +
- 43        Function zoomed on 1, (Main display)  
          0=Voltage, Current & Frequency  
          1=Watts, Current, Voltage & Frequency  
          2= VA, Current, Voltage & Frequency  
          3= VAr, Current, Voltage & Frequency  
          4= pf, Current, Voltage & Frequency
- 44        Function zoomed on 2, (Main display)  
          0=Current & Frequency  
          1= Watts, Current & Frequency  
          2= VA, Current & Frequency  
          3= VAr, Current & Frequency  
          4= pf, Current & Frequency  
          5= Current, Voltage & Frequency
- 45        Function zoomed on 3, (Main display)  
          0= Watts & Frequency  
          2= Watts, VA & Frequency  
          3= Watts, VAr & Frequency  
          4= Watts, pf & Frequency  
          5= Watts, Voltage & Frequency  
          6= Watts, Current & Frequency
- 46        Function zoomed on 4, (Main display)  
          0= Watts & VA  
          3= Watts, VA & VAr  
          4= Watts, VA & pf  
          5= Watts, VA & Voltage  
          6= Watts, VA & Current  
          7= Watts, VA & Frequency  
          8= Watts, VA & Harmonic  
          9= Watts, VA & dc watts  
          10= Watts, VA & V Ph-Ph

## PPA45xx communications manual

- 47      Datalog display type, (Datalog display information mode)  
          0=Real Time  
          1=Table  
          2=Graph
- 48      Manual frequency, (Acquisition advance options – frequency lock on)  
          0=Frequency in  $\mu$ Hz  
          1=Frequency in Hz
- 49      DFT selectivity, (Acquisition advance options)  
          0=Normal  
          1=Narrow
- 50      Program 1-6 direct load, (System options)  
          0=Disabled  
          1=Enabled
- 51      Language, (System options)  
          0=English  
          1=Other language if installed
- 52      Frequency filter, (Acquisition control)  
          0=Disabled  
          1=Enabled
- 53      Phase reference, (Acquisition control)  
          0=Voltage  
          1=Current
- 54      Datalog Zoom1, (Datalog-RAM)  
          0=Enabled  
          1=Disabled
- 55      Datalog Zoom2, (Datalog-RAM)  
          0=Enabled  
          1=Disabled
- 56      Datalog Zoom3, (Datalog-RAM)  
          0=Enabled  
          1=Disabled

## PPA45xx communications manual

- 57      Datalog Zoom4, (Datalog-RAM)  
          0=Enabled  
          1=Disabled
- 58      Datalog memory type, (Datalog)  
          0=Disabled  
          1=RAM  
          2=Internal Flash  
          3=USB Memory stick
- 59      Datalog Interval, (Datalog) (Enter interval time figure in seconds)
- 60      Datalog graph, (Datalog-RAM)  
          0=Together  
          1=Seperate
- 61      Formula, (Maths)  
          0=Disabled  
          1=(term1 + term2/term3 + term4)  
          2=(term1 + term2) x term3/term4  
          3=term1 x term2/(term3 + term4)
- 62      Argument term 1  
          0=Disabled  
          1=Constant  
          2=Voltage  
          3=Current  
          4=Torque  
          5=Speed
- 63      Sub argument term 1, (For voltage and current arguments only)  
          0=rms  
          1=dc  
          2=ac  
          3=Fundamental  
          4=Peak  
          5=Mean  
          6= Ph-Ph rms  
          7=Ph-Ph mag

## PPA45xx communications manual

- 64 Term 1 coefficient, (Enter value)
- 65 Argument term 2,  
0=Disabled  
1=Constant  
2=Voltage  
3=Current  
4=Torque  
5=Speed
- 66 Sub argument term 2, (For voltage and current arguments only)  
0=rms  
1=dc  
2=ac  
3=Fundamental  
4=Peak  
5=Mean  
6= Ph-Ph rms  
7=Ph-Ph mag
- 67 Term 2 coefficient, (Enter value)
- 69 Frequency lock, minimum freq, (ACQU, advanced options)  
  
Enter value (0.010 to 500)
- 70 application mode,  
0=Normal  
1=PWM motor Drive  
2=Lighting ballast  
3=Inrush current  
4=Transformer mode  
5=Standby power  
6=Calibration  
7=Not used  
8=Not used

## PPA45xx communications manual

- 71      Frequency filter, (Application options mode - PWM Motor Drive)  
          0=4KHz  
          1=1KHz  
          2=250Hz
- 72      Frequency tracking speed, (Application options mode - Lighting Ballast)  
          0=Fixed time  
          1=Fast  
          2=Medium  
          3=Slow
- 73      Low frequency, (Application options mode - PWM Motor Drive)  
          0=Off  
          1=On
- 74      Argument term 3  
          0=Disabled  
          1=Constant  
          2=Voltage  
          3=Current  
          4=Torque  
          5=Speed
- 75      Sub argument term 3, (For voltage and current arguments only)  
          0=rms  
          1=dc  
          2=ac  
          3=Fundamental  
          4=Peak  
          5=Mean  
          6= Ph-Ph rms  
          7=Ph-Ph mag
- 76      Term 3 coefficient, (Enter value)
- 77      Argument term 4  
          0=Disabled  
          1=Constant  
          2=Voltage  
          3=Current  
          4=Torque  
          5=Speed

## PPA45xx communications manual

- 78 Sub argument term 4, (For voltage and current arguments only)  
0=rms  
1=dc  
2=ac  
3=Fundamental  
4=Peak  
5=Mean  
6= Ph-Ph rms  
7=Ph-Ph mag
- 79 Term 4 coefficient, (Enter value)
- 80 Temperature, (Application-Transformer mode)  
0=Disabled  
1=Enabled °C  
2=Enabled °F
- 81 Sum watts, (Auxiliary-Master)  
0=Master  
1=Master + Slave
- 82 Wiring configuration, (Acquisition control)  
0=Single phase 1  
1=2 phase 2 wattmeter  
2=3 phase 2 wattmeter  
3=3 phase 3 wattmeter  
4=Single phase 2  
5=Single phase 3  
6=3 phase 2 wattmeter + PH3  
7=Independent
- 83 Integration, (Power analyzer - Power integrator)  
0=Signed  
1=Magnitude
- 84 Torque + speed, (Application options - PWM motor drive)  
0=Disabled  
1=Analogue speed  
2=Pulsed speed
- 85 Torque scaling Nm/V, (Applications - PWM motor drive) (Also transformer scale factor Deg/v)(Enter Nm/v value)
- 86 Speed scaling Hz/V,(Applications - PWM motor drive)(Enter rpm/v value)



## PPA45xx communications manual

- 87 Pulses per revolution, (Applications-PWM motor drive)(Enter pulses/rev value)
- 88 Integration display, (Mode - Power integrator)  
0=Total  
1=Average
- 89 Sum current average, (Power analyzer)  
0=Total  
1=Average
- 90 Phase 3 reference, (Acquisition control - 3 phase 2 wattmeter + PH3)  
0=Voltage  
1=Current  
2=ac line  
3=Phase 1 & 2
- 91 Power factor sign, (Power analyzer)  
0=Negative lagging  
1=Negative leading
- 92 VAR sign, (Power analyzer)  
0= Negative lagging  
1=Negative leading
- 93 Efficiency computation, (Power analyzer)  
0=Disabled  
1=Phase 1 / Phase 2  
2=Phase 2 / Phase 1  
3=Slave/Master  
4=Master/Slave  
5=Mechanical/Sum  
6=Sum/Mechanical  
7=Phase 3/Sum  
8=Sum/Phase 3
- 94 Range lock across phases, (Range - when acquisition is using 3 phases)  
0=Disabled  
1=Enabled
- 95 Torque offset, (Applications-PWM motor drive)(Also transformer mode)(Enter Nm offset value)
- 96 Speed offset, (Application options mode - PWM motor drive - rpm offset value)

## PPA45xx communications manual

- 99 Computation mode, (Harmonic analyzer)  
0=Difference formula  
1=Harmonic series  
2=TIF  
3=THF  
4=TRD  
5=TDD  
6=Series harmonic phase  
7=Interharmonic sweep  
8=Harmonic RMS  
9=Harmonic factor
- 100 Selected harmonic, (Harmonic analyzer - figure = harmonic required)
- 101 Harmonic series up to, (Harmonic analyzer - figure = harmonic max)
- 102 Voltage bargraph scale, (Harmonic analyzer - figure = % required)
- 103 Current rating (TRD), (Harmonic analyzer - TRD mode - enter figure)
- 104 Current bargraph scale, (Harmonic analyzer - figure = % required)
- 106 Timebase, (Scope - Enter figure/div)
- 107 trigger level, (Scope - Enter figure/div)
- 108 Pretrigger, (Scope)  
0=None  
1=25%  
2=50%  
3=75%
- 109 trigger polarity, (Scope)  
0=Rising edge  
1=Falling edge
- 110 trigger Mode, (Scope)  
0=Auto  
1=Normal  
2=Single shot

## PPA45xx communications manual

- 111 trigger reference, (Scope)  
0=Voltage  
1=Current
- 112 trigger phase, (Scope)  
0=Phase 1  
1=Phase 2  
2=Phase 3
- 113 cursors enable, (Scope)  
0=Off  
1=On
- 114 trigger HF reject, (Scope)  
0=Off  
1=On
- 115 Trace, (Scope)  
0=Dual  
1=Voltage  
2=Current
- 119 zoom 2 high resolution, (System)  
0=Disabled  
1=Enabled
- 120 Brightness, (System)  
0=Low  
1=High
- 121 Display, (System)  
0=Colour  
1=White on black  
2=Black on white
- 122 Auxiliary device, (Aux control)  
0=None  
6=PCIS



## PPA45xx communications manual

- 138 Measurement, (Impedance analyzer)  
0=Series  
1=Parallel
- 139 Phase offset, (Impedance analyzer - Enter figures)
- 140 Voltage peak, (rms voltmeter)  
0=Signed  
1=Separate  
2=Unfiltered
- 143 Sampling Rate / compensation (ACQU - sampling)  
0=Auto  
1=Fast  
2=Medium  
3=Slow  
4=19.2uS compensation  
5=3.857uS compensation
- 144 Rectified mean, (rms voltmeter)  
0=Absolute  
1=Normalised
- 148 dB offset, (Phase meter - Enter figures)
- 150 Computation, (Phase meter)  
0=ch2/ch1  
1=ch1/ch2
- 152 RS232 printer enable, (Remote options)  
0=Disabled  
1=Enabled
- 153 IEEE address, (Remote options - GPIB mode - enter address figures)
- 154 Interface, (Remote options)  
0=RS232  
1=USB  
2=LAN  
3=GPIB
- 155 Recall with program, (Remote options)  
0=Off  
1=On

**Alarm functions**

- 156 Alarm 1 data, (Alarm options)  
0=Zoom1  
1=Zoom 2  
2=Zoom3  
3=Zoom 4
- 157 Alarm 1 type, (Alarm options)  
0=Disabled  
1=Linear  
2=Alarm if high  
3=Alarm if low  
4=Outside window  
5=Inside window
- 158 Alarm 1 high threshold, (Alarm options – alarm if high – enter figure)
- 159 Alarm 1 low threshold, (Alarm options – alarm if low – enter figure)
- 160 Alarm latch, (Alarm options – alarm if high)  
0=Off  
1=On
- 161 Alarm sounder, (Alarm options – alarm if high)  
0=Enabled  
1=Disabled
- 162 Analog output, (Alarm options – alarm if high)  
0=Disabled  
1=Zoom 1  
2=Zoom 2  
3=Zoom 3  
4=Zoom 4  
5=Manual
- 164 Analog zero, (Alarm options – enter figure)
- 165 Analog full scale, (Alarm options – enter figure)

## PPA45xx communications manual

- 167 Alarm 2 data, (Alarm options)  
0=Zoom1  
1=Zoom 2  
2=Zoom 3  
3=Zoom 4
- 168 Alarm 2 type, (Alarm options)  
0=Disabled  
1=Linear  
2=Alarm if high  
3=Alarm if low  
4=Outside window  
5=Inside window
- 169 Alarm 2 high threshold, (Alarm options – alarm if high – enter figure)
- 170 Alarm 2 low threshold, (Alarm options – alarm if low – enter figure)
- 171 Sync on alarm, (Alarm options – alarm if high)  
0=Disabled  
3=Enabled
- 176 Enable channel 3, (Range–voltage input)(Sys - independent ranging enabled)  
1=Internal  
3=External attenuator
- 177 Enable channel 4, (Range – current input)(Sys independent ranging enabled)  
1=Internal  
2=External shunt
- 178 Input range channel 3, (Range – minimum range voltage) (Sys independent ranging enabled)  
0=300mV  
1=1V  
2=3V  
3=10V  
4=30V  
5=100V  
6=300V  
7=1kV  
8=3KV

## PPA45xx communications manual

- 179 Input range channel 4, (Range – minimum range current) (Sys independent ranging enabled)  
0=30mA  
1=100mA  
2=300mA  
3=1A  
4=3A  
5=10A  
6=30A  
7=100A  
8=300A
- 180 Input ranging channel 3, (Range – autoranging voltage) (Sys independent ranging enabled)  
0=Full Autorange  
1=Range up only  
2=Manual
- 181 Input ranging channel 4, (Range – autoranging current) (Sys independent ranging enabled)  
0= Full Autorange  
1=Range up only  
2=Manual
- 182 Coupling phase 2, (Coupling) (Sys independent ranging enabled)  
0=ac +dc  
1=ac  
2=dc
- 183 Bandwidth phase 2, (Coupling - bandwidth) (Sys independent ranging enabled)  
0=Wide (dc–2MHz)  
1=Low (dc-200KHz)  
2=dc (dc-5Hz)
- 184 Scale factor channel 3 voltage, (Ranging - Enter figures as required)(Sys independent ranging enabled)
- 185 Scale factor channel 4 current, (Ranging - Enter figures as required) (Sys independent ranging enabled)
- 186 External attenuator channel 3, (Ranging – voltage input - attenuator ratio - Enter figures as required)(Sys independent ranging enabled)



## PPA45xx communications manual

- 187 External shunt channel 4, (Ranging – current input – resistance value Enter figures as required) (Sys independent ranging enabled)
- 200 Enable channel 5, (Range – voltage input) (Sys independent ranging enabled)  
1=Internal  
3=External attenuator
- 201 Enable channel 6, (Range – current input) (Sys independent ranging enabled)  
1=Internal  
2=External shunt
- 202 Input range channel 5, (Range – minimum range voltage)  
0=300mV  
1=1V  
2=3V  
3=10V  
4=30V  
5=100V  
6=300V  
7=1kV  
8=3KV
- 203 Input range channel 6, (Range – minimum range current) (Sys independent ranging enabled)  
0=30mA  
1=100mA  
2=300mA  
3=1A  
4=3A  
5=10A  
6=30A  
7=100A  
8=300A
- 204 Input ranging channel 5, (Range – autoranging voltage) (Sys independent ranging enabled)  
0=Full Autorange  
1=Range up only  
2=Manual



## PPA45xx communications manual

- 226 Set clock hours, (System – Enter figures as required)
- 227 Set clock minutes, (System – Enter figures as required)
- 228 Set clock Seconds, (System – Enter figures as required)
- 229 Set date day, (System – Enter figures as required)
- 230 Set date month, (System – Enter figures as required)
- 231 Set date year, (System – Enter figures as required)

## PPA45xx communications manual

### Appendix C – MULTILog parameters

function	measurement	notes
1	frequency	
2	watts	
3	VA	
4	VAr	
5	power factor	
6	fundamental watts	
7	fundamental VA	
8	fundamental VAr	
9	fundamental PF	
10	harmonic watts	
11	harmonic watts %	
12	impedance	Imp meter mode
13	resistance	Imp meter mode
14	reactance	Imp meter mode
15	impedance phase	Imp meter mode
16	efficiency	
17	fundamental efficiency	
18	maths	
19	integrated watts	integrator mode
20	integrated VA	integrator mode
21	integrated VAr	integrator mode
22	integrated rms current	integrator mode
23	average power factor	integrator mode
24	integrated fundamental watts	integrator mode
25	integrated fundamental VA	integrator mode
26	integrated fundamental VAr	integrator mode
27	integrated fundamental current	integrator mode
28	average fundamental power factor	integrator mode
29	average integrated watts	integrator mode
30	average integrated VA	integrator mode
31	average integrated VAr	integrator mode
32	average integrated fundamental watts	integrator mode
33	average integrated fundamental VA	integrator mode
34	average integrated fundamental VAr	integrator mode
35	average rms voltage	integrator mode
36	average fundamental voltage	integrator mode
37	Standby mode frequency	
38	DC watts	

## PPA45xx communications manual

39	average rms current	integrator mode
40	average fundamental current	integrator mode
41	delta watts	
42	fundamental delta watts	
43	elapsed time	integrator mode
44	resistance	Imp meter mode
45	inductance	Imp meter mode
46	capacitance	Imp meter mode
47	tan delta	Imp meter mode
48	Q factor – see notes	Imp meter mode
48	k-factor – see notes	Transformer mode
49	corrected power	Transformer mode
50	rms voltage	
51	rms current	
52	fundamental voltage	
53	fundamental current	
54	voltage phase	
55	current phase	
56	harmonic voltage	
57	harmonic current	
58	dc voltage	
59	dc current	
60	ac voltage	
61	ac current	
62	peak voltage	
63	peak current	
64	voltage crest factor	
65	current crest factor	
66	rectified mean voltage	
67	rectified mean current	
68	voltage form factor	
69	current form factor	
70	voltage harmonic	harmonic mode
71	current harmonic	harmonic mode
72	voltage harmonic percentage	harmonic mode
73	current harmonic percentage	harmonic mode
74	voltage thd	harmonic mode
75	current thd	harmonic mode
76	voltage tif	harmonic mode
77	current tif	harmonic mode
78	phase to phase rms voltage	
79	phase to phase fundamental voltage	
80	phase to phase voltage phase angle	

## PPA45xx communications manual

81	phase to phase rms voltage	
82	voltage surge	
83	current surge	
84	voltage rms deviation	transformer mode
85	voltage fundamental deviation	transformer mode
86	voltage phase deviation	transformer mode
87	voltage positive peak	
88	current positive peak	
89	voltage negative peak	
90	current negative peak	
91	voltage positive peak unfiltered	
92	current positive peak unfiltered	
93	voltage negative peak unfiltered	
94	current negative peak unfiltered	
95-99	reserved for future expansion	

### Notes:

Function 48 is used to measure Q-factor in Imp meter mode AND to measure corrected power in Transformer mode.

Functions 78 and 81 are the same.

### Phase selection:

- 1 = phase 1
- 2 = phase 2
- 3 = phase 3
- 4 = sum
- 5 = neutral

## PPA45xx communications manual

There are some special functions:

Measurement (function)	phase	Previous function
mechanical speed in Hz	neutral	dc voltage (function 58)
mechanical speed in rpm	neutral	ac voltage (function 60)
torque in Nm	neutral	rms voltage (function 50)
mechanical power	neutral	Watts (function 2)

Notes:

These special functions must use the Neutral Phase (Phase 5)

Due to the limited number of function numbers available these Special functions re-use function numbers that apply to other measurements for Phases 1 to 3.

Examples for setting up each measurement:

- > multil,0                      Setting to clear any previous data
- > multil,1,5,58                Setting for Mechanical speed in Hz
- > multil,1,5,60                Setting for Mechanical speed in rpm
- > multil,1,5,50                Setting for Torque in Nm
- > multil,1,5,2                 Setting for Mechanical Power in nW
- > multil?                        Setting to read back and display data

Example script to return results for Mechanical Power, Torque & Speed (in rpm):

```
>
>
> multil,0
> multil,1,5,2
> multil,2,5,50
> multil,3,5,60
> multil?
-1.8846E-7,-2.0984E-3,8.5765E-4
```

## Newtonson4th Ltd. contact details

Please direct all queries or comments regarding the PPA45xx instruments or this manual to:

Newtonson4th Ltd.  
1 Bede Island Road  
Leicester  
LE2 7EA  
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.