



Newton4th Ltd

PPA3500

Communications Manual



Version v3.00

7th October 2019

IMPORTANT SAFETY INSTRUCTIONS

This equipment is designed to comply with BSEN 61010-1 (2010) (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 slots in the top and sides of the cover 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 PPA35xx series of instruments over RS232, USB, or LAN. 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. 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 v3.00

This manual is copyright © 2015-2019 Newtons4th Ltd. and all rights are reserved. No part may be copied or reproduced in any form without prior written consent.

4th January 2019

CONTENTS

1	Using remote control	1
1.1	Standard event status register.....	3
1.2	Serial Poll status byte	4
1.3	RS232 connections.....	5
1.4	Data format.....	6
2	Communication commands.....	7
	*CLS	7
	*ESE	8
	*ESR?.....	9
	*IDN?.....	10
	*OPC?	11
	*RST	12
	*SRE	13
	*SRE?.....	14
	*STB?.....	15
	*TRG.....	16
	*TST?.....	17
	*WAI	18
	ABORT	19
	ADIMAP	20
	ALARM	21
	ALARM?	22
	ALARM1	23
	ALARM2	24
	ALARME	25
	ALARME?.....	26
	ANALOG.....	27
	ANALOG?	28
	APPLIC.....	29
	BANDWI.....	30
	BEEP	31
	BLANKI	32
	CALSTR.....	33
	CALSTR?	34
	CONFIG	35
	CONFIG?	36
	COUPLI	37
	COUPLI?	38
	DATALO	39

PPA35xx communications manual

DATALO?	40
DAV?	41
DAVER	42
DAVER?	43
DISPLAY	44
DISPLAY?	45
EFFICI	46
EFFICI?	47
FAST	48
FQLOCK	49
FQREF	50
FREQUE	51
FSD?	52
GROUP	53
GROUP1	54
HARMON	55
HARMON?	56
HOLD	57
HPOWER	58
HPOWER?	59
INPUT	60
INTEGR	61
INTEGR?	62
KEYBOA	63
LCR	64
LCR?	65
LOWFRE	66
MODE	67
MULTIL	68
MULTIL?	69
NEWLOC	70
NOISEF	71
NOOVER	72
NORMAL	73
PFCONV	74
PHASEM	75
PHANGR	76
PHASEM?	77
PHCONV	78
POWER	79
POWER?	80
PRIMAR	82
PROGRA	83
PROGRA?	84

PPA35xx communications manual

RANGE.....	85
RESOLU	86
RESULT.....	87
RESULT?	88
REZERO	89
SCALE	90
SCOPE?.....	91
SCREEN?.....	92
SETUP	93
SETUP?	94
SHUNT.....	95
SMOOTH	96
SPEED	97
START	98
STATUS?.....	99
STOP	100
SUSPEN	101
TAGREP	102
TORQSP.....	103
TORQSP?	104
USER?	105
VARCON.....	106
VERSIO?	107
VRMS	108
VRMS?.....	109
WIRING	110
ZERO.....	111
ZOOM.....	112
ZOOM?	113
3 Multilog Application Guide	114

Appendix A – configurable parameters

Appendix B – MULTIL parameters

1 Using remote control

The instrument is fitted with an RS232 serial communications port and USB interface as standard, and may have a LAN interface fitted as an option. All the interfaces use the same ASCII protocol:

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

All the functions of the instrument can be programmed via any interface, and results read back.

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. E.g.

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 LAN (see section 2).

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 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 i.e.:

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

value	equivalent	hex data transmitted
3.0	0.75×2^2	0x82,0xB0,0x80,0x80
0.1	0.8×2^{-3}	0xFD,0xB3,0x99,0xCD
-320	-0.625×2^9	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,PPA3560,196-04676,2.48
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.

***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, FPGA 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
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:

ADIMAP**ADIMAP**

Function:	Map multilog parameters to outputs
Description:	Applies offset and scaling to a multilog value and maps value to chosen ADI output
Format:	ADIMAP,output,multilog,offset,scale
Arguments:	output: 1-20 multilog: 1-64 offset: Float scale: Float
Reply:	None
Example:	MULTIL,0 MULTIL,2,1,1 (PH1 Frequency) ADIMAP,1,2,0.2,0.5 Output 1 = $0.5 * (\text{PH1 frequency} - 0.2)$
Notes:	Offset is subtracted from multilog value, then scale is applied within the limits of +/- 10

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:	<i>ALARM2,type,data,high,low</i>
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, <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:

ANALOG**ANALOG**

Function:	Interface with ADI40
Description:	Write to individual ADI40 outputs
Format:	<i>ANALOG,channel,value</i>
Arguments:	Channel: 1-20 Value: -10.00 to +10.00
Reply:	None
Example:	ANALOG,5,-3.14
Notes:	Up to 9 outputs can be written to with one CommView transfer, by separating each instance with a ";".

ANALOG?

ANALOG?

Function: Interface with ADI40

Description: Read from individual ADI40 inputs

Format: ANALOG,channel?

Arguments: Channel:
1-20

Reply: ASCII characters in scientific format:
1 - 16 in Volts
17- 20 in °C

Example: ANALOG,12?

Notes: Up to 9 inputs can be read back at once with this command by separating each instance with a ";".

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
 BALLAST
 INRUSH
 POWERT
 STANDB
 setting:
 filter 0-6 (PWM only)
 speed 0-3 (ballast only)
 0: fixed time
 1: fast
 2: medium
 3: slow

Reply: none

Example: *APPLIC,POWERT*
APPLIC,BALLAST,1

Notes:

BANDWI

BANDWI

Function:	Selects the hardware bandwidth filter
Description:	The bandwidth may be set to "wide" or "low" to minimise high frequency noise in noisy environments.
Format:	BANDWI , <i>setting</i>
Arguments:	Setting: LOW WIDE HIGH
Reply:	None
Example:	BANDWI ,LOW
Notes:	WIDE and HIGH are the same.

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:

CALSTR**CALSTR**

Function:	Load a calibration 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:	CALSTR, <i>string</i>
Arguments:	<i>string</i> is any sequence of printable alpha numeric characters. Use the underscore character to add a space between words. CALSTR without a string argument clears the previously stored string.
Reply:	none
Example:	CALSTR,12_DEC_2015_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.

CALSTR?

CALSTR?

Function:	Read back the calibration 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:	CALSTR?
Arguments:	none.
Reply:	alphanumeric string
Example:	CALSTR? 12_DEC_2015_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, <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: or:	CONFIG, <i>index</i> ? 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 (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, <i>phase,coupling</i>
Arguments:	phase: PHASE1 PHASE2 PHASE3 PHASE4 PHASE5 PHASE6 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. PHASE4-6 may also be set by using PHASE1-3 when group 2 is the active group.

COUPLI?**COUPLI?**

Function:	Read ac/dc coupling setting.
Description:	Returns a numerical value for the coupling setting.
Format: or:	COUPLI, <i>phase,coupling?</i> COUPLI? <i>phase,coupling</i>
Arguments:	phase: PHASE1 PHASE2 PHASE3 PHASE4 PHASE5 PHASE6
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.
Format:	<i>DATALO,function,interval,speed</i>
Arguments:	function: DISABLE RAM NONVOL RECALL DELETE interval: datalog interval in seconds speed: HIGH
Reply:	none
Example:	DATALO,RAM,10 DATALO,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, <i>start,records?</i> 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:	DATALO,RAM,10 START wait for datalog STOP DATALO,LINES? 30 DATALO,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 DATALO? 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, <i>value</i>
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 for the active group.
Format:	DISPLAY, <i>page</i>
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 for the active group.

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 for the active group.
Format:	EFFICI, <i>formula</i>
Arguments:	formula: 0 – disabled 1 – phase / next phase 2 – next phase / phase 3 – group 2 / group 1 4 – group 1 / group 2 5 – mechanical / sum 6 – sum / mechanical 7 – phase 3 / sum 8 – sum /phase 3
Reply:	none
Example:	EFFICIENCY,3
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 or 6 data values
Example:	EFFICI? data returned
Notes:	6 data values returned if efficiency option is phase / next phase and 3 phase wiring is configured

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: Fix the frequency for analysis to the present value.

Format: FQLOCK,*mode,frequency*

Arguments: value:
ON
OFF
NORMAL
CONSTANT [,frequency]
DYNAMIC

Reply: none

Example: FQLOCK,ON

Notes: OFF is the same as NORMAL
To fix the analysis to a specified frequency, either first lock the frequency with FQLOCK,ON and send the desired frequency with the FREQUE command or send CONSTANT followed by the frequency.

FQREF**FQREF**

Function:	Set frequency reference.
Description:	Select the channel to be used for measuring the frequency on the active group.
Format:	<i>FQREF,phase</i> <i>FQREF,channel</i> <i>FQREF,phase,channel</i>
Arguments:	channel: voltage current 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).

FREQUE**FREQUE**

Function:	Set the analysis frequency
Description:	Sets the analysis frequency in Hz for frequency lock mode.
Format:	FREQUE, <i>frequency</i>
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,*channel*

Arguments: Channel:
CH1
CH2
CH3
CH4
CH5
CH6

Reply: Up to six data values separated by commas

Example: 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

GROUP

GROUP

Function: Set the active group.

Description: Select the group to which all future commands will be directed.

Format: GROUP,*group*

Arguments: group:
 1 or 2

Reply: none

Example: GROUP,2

Notes:

GROUP1**GROUP1**

Function:	Set the number of phases in group 1.
Description:	If there are more than three phases in group 1 then the instrument will operate in single group mode, otherwise the instrument will operate with two independent groups. A six phase unit always has at least three phases in group 1; a four phase unit could have one to four phases in group1
Format:	GROUP1, <i>phases</i>
Arguments:	phases: 1 to 6
Reply:	none
Example:	GROUP1,6
Notes:	For a four phase unit, set group 1 to 1 for 1 : 2+3+4 2 for 1+2 : 3+4 3 for 1+2+3 : 4 4 for 1+2+3+4 (single group)

HARMON**HARMON**

Function: Set harmonic analyser mode.

Description: Set harmonic analyser mode and parameters.

Format: *HARMON,para,harmonic,max*

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
HRMS	harmonic rms
HFACTO	harmonic factor
PH-PH	phase to phase

harmonic:

individual harmonic for display

max:

length of harmonic series (to 50)

Reply: none

Example: *HARMON,THDS,3,50*

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, <i>phase</i> ?
or:	HARMON,SERIES?
or:	HARMON, <i>phase</i> ,SERIES?
Arguments:	phase: PHASE1 PHASE2 PHASE3 PHASE4 PHASE5 PHASE6 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:

HPOWER

HPOWER

Function:	Set harmonic power parameters
Description:	Sets power parameters but does not change to power mode so that power can be measured in harmonic series mode.
Format:	See POWER
Arguments:	See POWER
Reply:	none
Example:	See POWER
Notes:	

HPOWER?

HPOWER?

Function:	Read harmonic power results
Description:	Reads power results but does not change to power mode so that power can be read in harmonic series mode.
Format:	See POWER?
Arguments:	See POWER?
Reply:	None
Example:	See POWER?
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 CH4 CH5 CH6 type: INTERN EXTATT EXTSHU INTX10
Reply:	None
Example:	INPUT,CH1,EXTSHU
Notes:	CH1 applies to all voltage channels (unless in single phase 2 or 3 wiring) CH2 applies to all current channels (unless in single phase 2 or 3 wiring) CH3 and 4 apply to phase 2 voltage and current when in single phase 2 wiring CH5 and 6 apply to phase 3 voltage and current when in single phase 3 wiring

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:	<i>INTEGR,type,display</i> <i>INTEGR,RUNTIM,hours,minutes</i>
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, <i>phase?</i>
Arguments:	phase: PHASE1 PHASE2 PHASE3 PHASE4 PHASE5 PHASE6 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, <i>value</i>
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, <i>phase?</i>
Arguments:	phase: PHASE1 PHASE2 PHASE3 PHASE4 PHASE5 PHASE6 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, <i>value</i>
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:

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: *MULTIL,index,phase,function*

Arguments: index:
 0 clear all
 1-64 select data 1-64
 phase:
 1-3 phase 1-3
 4 sum
 5 neutral
 6 ADI40
 7-9 phase 4-6
 10 sum 2
 11 neutral 2
 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:	MULTIL?
Arguments:	none:
Reply:	Up to 64 data values as selected by the MULTIL command in a single reply string
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 2-93 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 from the same data.
Notes:	After the command the data will still be held so to release the lock send SUSPEND,OFF

NOISEF**NOISEF**

Function:	Set the digital noise filter
Description:	Select the noise filter and set the filter frequency to reduce the presence of high frequency noise.
Format:	<i>NOISEF,setting,frequency</i>
Arguments:	setting: ON OFF Frequency: corner frequency in Hz
Reply:	none
Example:	NOISEF,ON,150E3
Notes:	Minimum filter frequency is 1kHz. Because the same digital filter is applied to voltage and current there is no introduced phase error.

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.

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.

PFCNV

PFCNV

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:	PFCNV, <i>type</i>
Arguments:	type: NEGLAG NEGLEA
Reply:	none
Example:	PFCNV,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

PHASEM**PHASEM**

Function: Set phase meter mode.

Description: Select phase meter mode and reference.

Format: PHASEM,*reference*

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

Reply: none

Example: PHASEM,CH2

Notes:

PHANGR**PHANGR**

Function:	Set phase angle reference
Description:	Fourier transform analysis can be phase referred to current or voltage fundamental
Format:	PHANGR, <i>reference</i>
Arguments:	reference: VOLTAGE CURRENT
Reply:	none
Example:	PHANGR,CURRENT
Notes:	If measuring current without any voltage present it is important to set the phase angle reference to CURRENT for the fundamental to be accurate.

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
PHASE4
PHASE5
PHASE6
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
Description:	Set phase convention
Format:	PHCONV, <i>convention</i>
Arguments:	convention: 180: -180 to +180 -360: 0 to -360 +360: 0 to +360
Reply:	none
Example:	PHCONV, -360
Notes:	0 to -360 degrees is usually used for power analysis applications

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
 PHASE4
 PHASE5
 PHASE6
 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.....
 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.

PRIMAR**PRIMAR**

Function:	Select only primary functions.
Description:	Sets the instrument to only compute total functions not fundamentals, in order to allow shorter measurement windows..
Format:	PRIMAR, <i>value</i>
Arguments:	value: ON OFF
Reply:	none
Example:	PRIMAR,ON
Notes:	When primary is on, fundamental values will be displayed as zero.

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 0-100
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 1-8

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, <i>format</i>		
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

Function: Access results stores.

Description: Recall, store or delete results.

Format: *RESULT,function,number*

Arguments: function:
 RECALL
 STORE
 DELETE
 number
 1-99

Reply: none

Example: RESULT,RECALL,13

Notes:

RESULT

RESULT?

RESULT?

Function: Identify used result stores.

Description: Reads a directory of the result locations.

Format: RESULT,NAME?
RESULT,FILES?

Arguments: none

Reply: Name of last recalled result or program
List of names of all stored results

Example: RESULT,FILES?
10,AMW1,18/01/2016
11,AMW2,21/01/2016
RESULT,RECALL,11
RESULT,NAME?
AMW2

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:	<i>SCALE,channel,factor</i>
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
 PHASE4
 PHASE5
 PHASE6
 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? SCREEN,COLOUR? SCREEN, <i>group</i> SCREEN,COLOUR, <i>group</i> ?
Arguments:	Group GROUP1 GROUP2
Reply:	Multiple lines of data bit values
Example:	SCREEN? data returned
Notes:	SCREEN? response: 272 lines of 60 bytes in ASCII coded Hex (2 characters for each byte) 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 SCREEN,COLOUR? response: 1088 lines of 120 bytes preceded by #C Each line of the display is sent as four lines of data from left to right Each byte represents a single RGB dot in binary format: 0 1 r1 r0 g1 g0 b1 b0

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: 0-31 data: ASCII hex as returned by SETUP?
Reply:	None
Example:	SETUP? Read 31 lines of data SETUP,00,data00 SETUP,01,data01 . . SETUP,31,data31
Notes:	The settings are only updated when the 32 nd 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:	32 lines of ASCII data
Example:	SETUP? Read 32 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:	<i>SMOOTH,type,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 target window size for the measurement.
Format:	<i>SPEED,value>window</i>
Arguments:	value: VERY FAST 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 The actual window size used depends on the frequency of the signal.

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, <i>channel</i> ?
Arguments:	channel: CH1 . . CH6
Reply:	range number,range text,over/under/ok 1-8 range as per RANGE command OVER if overflow LOW if underflow OK if in range
Example:	STATUS,CH1? 6,300V,OK STATUS? OK
Notes:	STATUS? gives a summary value reporting OK only if all channels are not over range or under range

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 the comms reply tag
Description:	When TAGREP is enabled any reply string is preceded by an identifier string in order to identify a message from a given instrument connected to a network.
Format:	TAGREP, <i>setting</i>
Arguments:	setting: DISABLED or OFF ENABLED or ON
Reply:	none
Example:	*ESR? 1 TAGREP,ON *ESR? PPA3560:04656:1
Notes:	All reply strings will be preceded with: Instrument type:serial number:

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,10,1none 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
 PPA3560 #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

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 – 20A 4 – 30A
Examples:	VERSIO? KQ4715,4,2.11,2.11,2.02,2.01
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. Waits for next unread data if necessary. Clears new data available status bit.
Format:	VRMS, <i>phase,results?</i>
Arguments:	results: RMS MEAN SURGE phase: PHASE1 PHASE2 PHASE3 PHASE4 PHASE5 PHASE6 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, Vsurge,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, <i>type</i>
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)
Reply:	none
Examples:	WIRING,PHASE2
Notes:	WIRING,SINGLE is the same as WIRING,PHASE1 Use GROUP1 command to set a single group with four or more phases.

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:	level: 0 – normal 1 – zoom data larger font (zoom level 1) 2 – zoom data only (zoom level 2) 3 – first three zoom data only (zoom level 3) data1-4: zoom data data consists of line number for channel 1 or line number + 64 for channel 2
Reply:	None
Example:	VRMS ZOOM,1,1,65 (level 1, ch1 rms, ch2 rms)
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 – zoom data larger font (zoom level 1)

2 – zoom data only (zoom level 2)

3 – first three zoom data only (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:

3 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 (e.g. Watts, VAr, Frequency etc) of the return.

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 PPA3500 has 99 possible functions:

Function	Measurement	Notes
1	frequency	
2	watts	
3	VA	
4	VA _r	
5	power factor	
6	fundamental watts	
7	fundamental VA	
8	fundamental VA _r	
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 A – Configurable parameters

All parameters can be accessed using the CONFIG command:

CONFIG,*number,parameter*

Send GROUP,GROUP2 to access the parameters for group 2. All subsequent commands will be applied to group 2 until GROUP,GROUP1 is sent.

Note that not all parameters have a corresponding value for group 2. For example the selection of COM port is common to both groups.

<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
4	Autozero manual or auto,	(System options) 0=Auto 1=Manual

PPA35xx communications 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
- 8 Graph, (System options)
 0=Dots
 1=Lines
- 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

PPA35xx communications manual

- 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)
- 22 Independent ranging, (System options)
 0=Disabled
 1=Enabled
- 23 Grouping, (number of phases in group 1)
- 24 Enable channel 1, (Range - voltage input)
 1=Internal
 3=External Attenuator
 4=Internal x 10
- 25 Enable channel 2, (Range - current input)
 1=Internal
 2=External Shunt
 4=Internal x10
- 26 Input range channel 1, (Range - minimum range voltage)
 0=1V
 1=3V
 2=10V
 3=30V
 4=100V
 5=300V
 6=1kV
 7=3kV

PPA35xx communications manual

- 27 Input range channel 2, (Range – minimum range current)
0=300mA
1=1A
2=3A
3=10A
4=30A
5=100A
6=300A
7=1kA

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

30 Coupling, (Coupling)
0=ac+dc
1=ac
2=dc

32 Scale factor channel 1 voltage, (Ranging - Enter figures as required)

33 Scale factor channel 2 current, (Ranging - Enter figures as required)

34 External attenuator channel 1, (Ranging – voltage input - attenuator ratio
– Enter figures as required)

35 External shunt channel 2, (Ranging – current input - resistance value- Enter
figures as required)

36 Phase 1 noise filter, (Acqu
0 = Off
1 = On

37 Phase 1 noise filter frequency, (Acqu corner frequency in Hertz)
0 = Off
1 = On

38 Frequency reference voltage/current, (Acquisition control)
0=Voltage
1=Current

- 40 Frequency reference phase, (Acquisition control)
 0=Phase 1
 1=Phase 2
 2=Phase 3
- 41 Display page, (Main display)
 0=Phase 1 page
 1=Phase 2 page
 2=Phase 3 page
 3=Sum page
 4=Phase 1,2 & 3 page
 5=Phase 1,2 & 3 fundamentals page
 6=NEU page
- 42 Zoom level, (Main display)
 0=Zoom level 0
 1=Zoom level 1
 2=Zoom level 2 – 4 figures
 3=Zoom level 3 – 3 figures
- 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

PPA35xx communications manual

- 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
- 47 Datalog display type, (Datalog display information mode)
 0=Real Time
 1=Table
 2=Graph
- 48 Manual frequency, (Acqu advanced options – frequency in Hertz)
- 49 DFT selectivity, (Acqu advance options)
 0=Normal
 1=Narrow
- 50 Program 1-4 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, fundamental>1kHz
 2=Enabled, fundamental<1kHz
- 53 Phase reference, (Acquisition control)
 0=Voltage
 1=Current
- 54 Datalog Zoom1, (Datalog-RAM)
 0=Enabled
 1=Disabled

PPA35xx communications manual

- 55 Datalog Zoom2, (Datalog-RAM)
 0=Enabled
 1=Disabled
- 56 Datalog Zoom3, (Datalog-RAM)
 0=Enabled
 1=Disabled
- 57 Datalog Zoom4, (Datalog-RAM)
 0=Enabled
 1=Disabled
- 58 Datalog memory type, (Datalog)
 0=Disabled
 1=RAM
- 59 Datalog Interval, (Datalog) (Enter interval time figure in seconds)
- 60 Datalog graph, (Datalog-RAM)
 0=Together
 1=Separate
- 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
- 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 Low frequency mode minimum frequency, (Enter value)
- 70 Application mode,
 0=Normal
 1=PWM
 2=Lighting ballast
 3=Inrush current
 4=Transformer mode
 5=Standby power
 6=Calibration mode

PPA35xx communications manual

- 72 Frequency tracking speed, (Application options mode - Lighting Ballast)
 0=Fixed time
 1=Fast
 2=Medium
 3=Slow
- 73 PWM and ballast low frequency, (Application options mode)
 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

PPA35xx 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)
- 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=group2 uses group1 data
- 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)
- 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

PPA35xx communications manual

- | | |
|-----|--|
| 90 | Input compensation, (Mode)
0=Disabled
1=Enabled |
| 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 / next
2=Next / Phase
3=group2/group1
4=group1/group2
5=mechanical/sum
6=sum/mechanical
7=Phase 3/Sum
8=Sum/Phase 3 |
| 94 | Torque offset, (Mode – value in Volts) |
| 95 | Speed offset, (Mode – value in Volts) |
| 96 | Voltage rating for HVF, (Harmonic mode – value in Volts) |
| 99 | Computation mode, (Harmonic analyzer)
0=Difference formula
1=Harmonic series
2=TIF
3=THF
4=TRD
5=TDD
6=Series harmonic phase
7=Harmonic rms
8=Harmonic factor |
| 100 | Selected harmonic, (Harmonic analyzer - figure = harmonic required) |
| 101 | Harmonic series up to, (Harmonic analyzer - figure = harmonic max) |

PPA35xx communications manual

- 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
- 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

PPA35xx communications manual

- | | | |
|-----|---|--|
| 115 | Trace, (Scope) | 0=Dual
1=Voltage
2=Current |
| 117 | DFT phase angle ref, (System) | 0=Cosine
1=Sine |
| 119 | Zoom 2 high resolution, (System) | 0=Disabled
1=Enabled |
| 120 | Brightness, (System) | 0=Low
1=High |
| 122 | Auxiliary device, (Aux control) | 0=None
6=PCIS
9-12=ADI40 |
| 128 | Switch phase offset, (Aux control – PCIS device) | 0=0°
1=45°
2=90°
3=135°
4=180°
5=225°
6=270°
7=315° |
| 129 | Switch on cycles, (Aux control – PCIS device) | 0=Single cycle
1=Continuous
2=Half cycle |
| 131 | 2 Wattmeter sum computation, (Power Analyser) (select in acquisition wiring-2 phase 2 wattmeter) | 0=Low distortion
1=High Distortion |
| 132 | Integrator-run time (Hours), (Mode – Power integrator - enter figure) | |

PPA35xx communications manual

- 133 Integrator-Run time (mins), (Mode - Power integrator – enter figure)
- 134 Ph – Ph Measurement, (Power analyser)
0=rms
1=Mean
2=star-delta
3=delta-star
- 135 Difference THD, (Power analyser – penultimate line - Vthd)
0=Disabled
1=Enabled including dc
2=Enabled excluding dc
- 137 Parameter, (Impedance analyzer)
0=Auto
1=Capacitance
2=Inductance
3=Impedance
- 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
- 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

PPA35xx communications manual

- 154 Interface, (Remote options)
 0=RS232
 1=USB
 2=LAN
 3=GPIB
- 155 Recall with program, (Remote options)
 0=Off
 1=On
- 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 window5=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
- 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)
- 176 Enable channel 3, (Range–voltage input)(Sys - independent ranging enabled)
1=Internal
3=External Attenuator
4=Internal x10
- 177 Enable channel 4, (Range – current input)(Sys independent ranging enabled)
1=Internal
2=External Shunt
4=Internal x10
- 178 Input range channel 3, (Range – minimum range voltage) (Sys independent ranging enabled)
0=1V
1=3V
2=10V
3=30V
4=100V
5=300V
6=1kV
7=3kV
- 179 Input range channel 4, (Range – minimum range current) (Sys independent ranging enabled)
0=300mA
1=1A
2=3A
3=10A
4=30A
5=100A
6=300A
7=1kA

PPA35xx communications manual

- 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

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)

187 External shunt channel 4, (Ranging – current input – resistance value Enter
figures as required) (Sys independent ranging
enabled)

188 Phase 2 noise filter, (Acqu)
 0 = Off
 1 = On

189 Phase 2 noise filter frequency, (Acqu corner frequency in Hertz)
 0 = Off
 1 = On

196 ID tag prepends comms replies
 0 = Off
 1 = On

197 High speed mode
 0 = Off
 1 = On

PPA35xx communications manual

- 200 Enable channel 5, (Range – voltage input) (Sys independent ranging enabled)
 1=Internal
 3=External Attenuator
 4=Internal x10
- 201 Enable channel 6, (Range – current input) (Sys independent ranging enabled)
 1=Internal
 2=External Shunt
 4=Internal x10
- 202 Input range channel 5, (Range – minimum range voltage)
 0=1V
 1=3V
 2=10V
 3=30V
 4=100V
 5=300V
 6=1kV
 7=3kV
- 203 Input range channel 6, (Range – minimum range current) (Sys independent ranging enabled)
 0=300mA
 1=1A
 2=3A
 3=10A
 4=30A
 5=100A
 6=300A
 7=1kA
- 204 Input ranging channel 5, (Range – autoranging voltage) (Sys independent ranging enabled)
 0=Full Autorange
 1=Range up only
 2=Manual
- 205 Input ranging channel 6, (Range – autoranging current) (Sys independent ranging enabled)
 0= Full Autorange
 1=Range up only
 2=Manual

PPA35xx communications manual

- | | |
|-----|--|
| 206 | Coupling phase 3,
0=ac +dc
1=ac
2=dc

 |
| 208 | Scale factor channel 5 voltage, |
| 209 | Scale factor channel 6 current, |
| 210 | External attenuator channel 5, |
| 211 | External shunt channel 6, |
| 212 | Phase 3 noise filter,
0 = Off
1 = On

 |
| 213 | Phase 3 noise filter frequency,
0 = Off
1 = On

 |
| 231 | Memory,
0=Internal
1=USB Memory stick

 |
| 232 | Data,
0=Program
1=Results
2=Datalog

 |
| 233 | Action,
0=Recall
1=Store
2>Delete

 |
| 234 | Location, |

PPA35xx communications manual

Appendix B – MULTIL 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	
13	resistance	
14	reactance	
15	impedance phase	
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	

PPA35xx 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	LCR resistance	
45	LCR inductance	
46	LCR capacitance	
47	LCR tan delta	
47	LCR Q factor	
49	reserved for future expansion	
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	
71	current harmonic	
72	voltage harmonic percentage	
73	current harmonic percentage	
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	
81	phase to phase rms voltage	

PPA35xx communications manual

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	In-phase component of voltage	
96	Quadrature component of voltage	
97	In-phase component of current	
98	Quadrature component of current	
99	reserved for future expansion	

Phase selection:

- 1 = phase 1
- 2 = phase 2
- 3 = phase 3
- 4 = sum
- 5 = neutral
- 6 = ADI40
- 7 = Phase 4
- 8 = Phase 5
- 9 = Phase 6
- 10 = Sum 2
- 11 = Neutral 2

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)

PPA35xx communications manual

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 individual special function measurements:

> 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
```

Newton4th Ltd. contact details

Please direct all queries or comments regarding the PPA35xx instruments or this manual to:

Newton4th 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@newtons4th.com
 office@newtons4th.com

web site: www.newtons4th.com

At Newton4th 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.