PSM1700 - \textit{PsimetriQ}
PSM1735 - \textit{NumetriQ}

Communications Manual

PSM1700 firmware v1.67
PSM1735 firmware v1.61

Version 2.00

03\textsuperscript{rd} February 2020
“Do not be hasty when making measurements.”

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

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

- 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 must not be connected to signals greater than is indicated on the front panel.
- Keep the ventilation holes on the underneath and sides free from obstruction.
- Do not operate or store under conditions where condensation may occur or where conducting debris may enter the case.
- There are no user serviceable parts inside the instrument – do not attempt to open the instrument, refer service to the manufacturer or his appointed agent.

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

This manual gives details of the communication commands recognized by the PSM17xx series of instruments over RS232, LAN or GPIB. For more general operating instructions for the instrument refer to the specific user manual.

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

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

Revision 1.67 (PsimetriQ) 1.61 (NumetriQ)

This manual is copyright © 2004 - 2020 Newtons4th Ltd. and all rights are reserved. No part may be copied or reproduced in any form without prior written consent.

03rd February 2020
## 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 streaming .............................................. 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  
   *STB? ............................................................... 2-9  
   *TRG ................................................................. 2-10  
   *TST? ............................................................... 2-11  
   *WAI ............................................................... 2-12  
   ABORT ............................................................... 2-13  
   ACTRIM ............................................................. 2-14  
   ALARM ............................................................... 2-15  
   ALARM1 ............................................................. 2-17  
   ALARM2 ............................................................. 2-18  
   ALARME ............................................................. 2-19  
   AMPLIT ............................................................. 2-21  
   ANALOG ............................................................. 2-22  
   BANDWI ............................................................. 2-23  
   BEEP ................................................................. 2-24  
   BLANKI ............................................................. 2-25  
   CONFIG .............................................................. 2-26  
   COUPLI .............................................................. 2-28  
   CYCLES .............................................................. 2-29  
   DATALO .............................................................. 2-30  
   DAV? ................................................................. 2-31  
   DAVER ............................................................... 2-32  
   DELAY ............................................................... 2-33  
   FILTER .............................................................. 2-35  
   FRA ................................................................. 2-36  
   FREQUE ............................................................ 2-37  
   FSweep .............................................................. 2-38  
   FSweep .............................................................. 2-39  
   FSweep .............................................................. 2-40
1 Using remote control

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

<table>
<thead>
<tr>
<th></th>
<th>Rx expects</th>
<th>Tx sends</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS232</td>
<td>carriage return</td>
<td>carriage return (line feed ignored) and line feed</td>
</tr>
<tr>
<td>LAN</td>
<td>carriage return (line feed ignored)</td>
<td>carriage return (line feed ignored) and line feed</td>
</tr>
<tr>
<td>IEEE488</td>
<td>carriage return or line feed or EOI</td>
<td>carriage return with EOI</td>
</tr>
</tbody>
</table>

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

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

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

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

```
AMPLIT,1.5;OUTPUT,ON
```

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

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

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

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

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

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

<table>
<thead>
<tr>
<th>PON</th>
<th>CME</th>
<th>EXE</th>
<th>DDE</th>
<th>QYE</th>
<th>OPC</th>
</tr>
</thead>
</table>

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

<table>
<thead>
<tr>
<th></th>
<th>ESB</th>
<th>MAV</th>
<th>ALM</th>
<th>FDV</th>
<th>SDV</th>
<th>RDV</th>
</tr>
</thead>
</table>

- **bit 0** RDV (result data available)  
  set when results are available to be read as enabled by DAVER

- **bit 1** SDV (sweep data available)  
  set when sweep results are available to be read as enabled by DAVER

- **bit 2** FDV (fast data available (streaming))  
  set when data streaming 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).

<table>
<thead>
<tr>
<th>Pin</th>
<th>Function</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DCD</td>
<td>in (+ weak pull up)</td>
</tr>
<tr>
<td>2</td>
<td>RX data</td>
<td>in</td>
</tr>
<tr>
<td>3</td>
<td>TX data</td>
<td>out</td>
</tr>
<tr>
<td>4</td>
<td>DTR</td>
<td>out</td>
</tr>
<tr>
<td>5</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>DSR</td>
<td>not used</td>
</tr>
<tr>
<td>7</td>
<td>RTS</td>
<td>out</td>
</tr>
<tr>
<td>8</td>
<td>CTS</td>
<td>in</td>
</tr>
<tr>
<td>9</td>
<td>RI</td>
<td>not used</td>
</tr>
</tbody>
</table>

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 streaming

The phase angle voltmeter and power meter modes have the option of high speed data streaming. In this operation, the window width for the measurement may be specified from 660us to 100ms and the data for each measurement window is transmitted over the communications in a continuous stream. The window is adjusted to synchronise to the measured frequency.

The instrument buffers the data and transmits at the fastest rate that is possible. The buffer depth is over 8000 data values so more than 5 seconds of data can be captured at the fastest rate of 1500 readings per second even if the data is not read at all. If the window size is such that the data can be read out in real time then data streaming can continue indefinitely.

Once the data streaming window has been setup but before the streaming has been started, the display periodically shows the measured value. Once streaming has been started, the display is blanked to minimise processing overheads. Streaming can be stopped either immediately (ABORT) or may be stopped but remaining data continues to be transmitted until the buffer is empty (STOP).

STREAM,ENABLE,0.01
START
read data
STOP
continue to read stored data
2 Communication commands

*CLS

Function: Clear status
Description: Clears the standard event status register.
Format: *CLS
Arguments: none
Reply: none
Example: *CLS
*ESR?
0

Notes:
**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?**

**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.
**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,PSIMETRIQ,01234,1.00

**Notes:**
*OPC?  *OPC?

Function: Test for operation complete
Description: Returns 1 if previous operation is completed, 0 if not.
Format: *OPC?
Arguments: none
Reply: 0 or 1
Example: START
  *OPC?
  0
  *OPC?
  0
  *OPC?
  1

Notes: *OPC? can be used to indicate when data is available or when a frequency sweep has completed.
**RST**

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.

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

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?**

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?**

Function: Read serial poll status byte

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

Format: *STB?

Arguments: none

Reply: decimal value of the serial poll status byte

Example: *STB?
1

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

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

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

Notes:
*TST?  *TST?

Function: Self test query
Description: Returns the results of self test
Format: *TST?
Arguments: none
Reply: single integer
  bit 0 – set if uncalibrated
  bit 1 – set if error with analogue zero
  > 15 – major system error

Example: *TST?
  0

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

**Function:** Abort sweep

**Description:** Abort an active sweep, or data streaming.

**Format:** ABORT

**Arguments:** none

**Reply:** none

**Example:**
```
MODE,PHASE,STREAM,0.01
START
read data values as required
ABORT
```

**Notes:** Any remaining values held in the buffer will be discarded.
**ACTRIM**

**Function:** Set ac control parameters

**Description:** Sets the specified signal level, tolerance and input channel for the ac control (amplitude compression).

**Format:** ACTRIM, channel, level, tolerance

**Arguments:**
- channel:
  - DISABL
  - CH1
  - CH2
- level:
  - required ac level in V or A or dBm
- tolerance:
  - required accuracy in percent

**Reply:** none

**Example:** ACTRIM, CH1, 1.0, 5 (1.0V, 5%)

**Notes:**
- The level should be set in dBm if dBm mode is selected (OUTPUT, DBM)
- It is not necessary to send all the arguments but those that are sent must be in the correct sequence.
Function: Set common controls for alarm1 and alarm2.

Description: Set the alarm latch and sounder control.

Format: \texttt{ALARM,latch,sounder}

Arguments:

\begin{itemize}
  \item [latch:] \texttt{ON}
  \item [ON]
  \item [OFF]
  \item [sounder:] \texttt{ENABLED}
  \item [ENABLED]
  \item [DISABLED]
\end{itemize}

Reply: none

Example: \texttt{ALARM,ON,DISABLED}

Notes:
**ALARM?**

<table>
<thead>
<tr>
<th>Function:</th>
<th>Read alarm status.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description:</td>
<td>Reads the status of the measurements and 2 alarms.</td>
</tr>
<tr>
<td>Format:</td>
<td>ALARM?</td>
</tr>
<tr>
<td>Arguments:</td>
<td>none</td>
</tr>
<tr>
<td>Reply:</td>
<td>single integer</td>
</tr>
<tr>
<td></td>
<td>bit 0  data available</td>
</tr>
<tr>
<td></td>
<td>bit 1  data error</td>
</tr>
<tr>
<td></td>
<td>bit 2  alarm 1</td>
</tr>
<tr>
<td></td>
<td>bit 3  alarm 2</td>
</tr>
<tr>
<td>Example:</td>
<td>ALARM?</td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Notes:</td>
<td>An alarm is present if bit 0 is high (data is available) and either alarm 1 or alarm 2 bits are high.</td>
</tr>
</tbody>
</table>
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

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**

**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?

<table>
<thead>
<tr>
<th>Function:</th>
<th>Read alarm status enable register</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description:</td>
<td>Read back present bits in the alarm status enable register which controls the alarm active bit in the status byte.</td>
</tr>
<tr>
<td>Format:</td>
<td>ALARME?</td>
</tr>
<tr>
<td>Arguments:</td>
<td>none</td>
</tr>
<tr>
<td>Reply:</td>
<td>decimal equivalent of alarm bits</td>
</tr>
<tr>
<td>Example:</td>
<td>ALARME?</td>
</tr>
<tr>
<td></td>
<td>12</td>
</tr>
<tr>
<td>Notes:</td>
<td></td>
</tr>
</tbody>
</table>
### AMPLIT

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

Function: Set up analogue output

Description: Sets the scaling of the analogue output or a constant value.

Format: 
- ANALOG,MONITOR,zero,scale
- ANALOG,MANUAL,value

Arguments: 
- zero level for monitored value
- full scale for monitored value
- value between 0 and 1 for manual

Reply: none

Example: ANALOG,MONITOR,

Notes: Analog output given by:
\[
\text{fraction} = \frac{\text{value} - \text{zero}}{\text{full scale}}
\]
**PSM17xx communications manual**

**BANDWI**

PSM1735 NumetriQ only

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

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

**Format:** BANDWI,type

**Arguments:**

- type:
  - AUTO
  - WIDE
  - LOW

**Reply:** none

**Example:** BANDWI,WIDE

**Notes:** In wide bandwidth mode the frequency range is limited to 1MHz. In low bandwidth mode, the frequency is restricted to 30kHz.
Function: Sound the buzzer
Description: Makes a “beep” from the instrument.
Format: BEEP
Arguments: none
Reply: none
Example: BEEP
Notes:
**Function:** Select blanking

**Description:** Enable or disable low value blanking.

**Format:** `BLANKI,value,threshold`

**Arguments:**
- **value:**
  - ON
  - OFF
- **threshold:**
  - threshold in appropriate units

**Reply:** none

**Example:**
- `BLANKI,OFF`
- `BLANKI,ON,-35`

**Notes:** It is not necessary to send the threshold. Not all functions have a threshold option, it will only be used if in an appropriate mode.
<table>
<thead>
<tr>
<th><strong>CONFIG</strong></th>
<th><strong>CONFIG</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Function:</strong></td>
<td>Direct access of configuration parameters</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
<td>Sets configuration parameter for which there may not be a direct command.</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td>CONFIG,&lt;index&gt;,&lt;data&gt;</td>
</tr>
<tr>
<td><strong>Arguments:</strong></td>
<td>index is the number of the parameter data is the data for that parameter</td>
</tr>
<tr>
<td><strong>Reply:</strong></td>
<td>none</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>CONFIG,6,1 (set phase convention)</td>
</tr>
<tr>
<td><strong>Notes:</strong></td>
<td>The list of configurable parameters is given in the appendix. CONFIG goes through the same limit checking as when entering data from the menus.</td>
</tr>
</tbody>
</table>
**CONFIG?**

**Function:** Configurable parameter query

**Description:** Reads the present value of a single parameter.

**Format:**
- CONFIG, \textit{index}?  
- or:
- CONFIG? \textit{index}

**Arguments:** \textit{index} is the parameter number

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

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

**Notes:** The list of configurable parameters is given in the appendix.
Function: Set ac or dc coupling.
Description: Selects the input coupling for a given input channel.
Format: COUPLI,channel,coupling
Arguments: channel:
  CH1
  CH2
coupling:
  AC+DC
  ACONLY
Reply: none
Example: COUPLI,CH2,AC+DC
Notes:
**CYCLES**

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

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

**Format:** CYCLES, cycles

**Arguments:** minimum number of cycles

**Reply:** none

**Example:** CYCLES, 4

**Notes:**
Function: Set up datalog
Description: Sets datalog parameters or accesses datalog non-volatile store.
Format: DATALO, function, interval
Arguments:
  function:
    DISABLE
    RAM
    NONVOL
    RECALL
    DELETE
  interval:
    datalog interval in seconds
Reply: none
Example: DATALOG, NONVOL, 10
Notes:
**DATALO?**

Function: Read back datalog results

Description: Return datalog values, one record per line

Format: DATALO,start,records

Arguments:

- **start:**
  - first record to return
- **records:**
  - number of records to return

Reply: 3 to 6 data values depending on settings:

- index 1-n
- elapsed time in hours
- data1
- data2 (if stored)
- data3 (if stored)
- data4 (if stored)

one record per line

Example:

DATALOG,NONVOL,36
START
wait for datalog
STOP
DATALOG,20,4?
20,1.9000E-1,1.2345E0
21,2.0000E-1,1.2345E0
22,2.1000E-1,5.6789E3
23,2.2000E-1,1.2345E0

Notes: if no arguments are sent then DATALOG? returns all data in the same format
DAV?

Function: Data available query
Description: Returns data availability status.
Format: DAV?
Arguments: none
Reply: Decimal equivalent of data available bits:
   bit0  new data available
   bit1  data available
   bit2  new full sweep data available
   bit3  sweep data available
   bit4  streaming data available
   bit5  more streaming data to come
   bit6  integration data available
   bit7  datalog data available
Example: START (trigger sweep)
          DAV?
          0
          DAV?
          11 (first data available)
          DAV?
          11
          DAV?
          11
          DAV?
          15 (full sweep data available)
Notes: DAV? does not modify the status bits.
Function: Set data available enable register

Description: Sets bits in the data available enable register to control which status bits set the data available bits in the status byte.

Format: DAVER, value

Arguments: decimal equivalent of data available bits
- bit0 set bit 0 of status byte when new data available
- bit1 set bit 0 of status byte when data available
- bit2 set bit 1 of status byte when new full sweep data available
- bit3 set bit 1 of status byte when sweep data available
- bit4 set bit 2 of status byte when streaming data available
- bit5 set bit 2 of status byte if more streaming data is to come

Reply: none

Example: DAVER, 4
set bit 1 in status byte only when full sweep data is ready

Notes: default value is 6:
- bit 0 of status byte is set whenever data is available
- bit 1 of status byte is set when full sweep data is available.
DAVER?

Function: Read data available enable register
Description: Read back present setting of the data available enable register, which controls the status bits that set the data available bits in the status byte.
Format: DAVER?
Arguments: none
Reply: decimal equivalent of bits
Example: DAVER?
4
Notes:
DELAY

Function: Set a delay time between frequency points

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

Format: DELAY, time

Arguments: delay time in seconds

Reply: none

Example: DELAY, 1

Notes:
FILTER

Function: Select the filtering

Description: Sets the filter time constant and dynamic response.

Format: FILTER,type,dynamics

Arguments:

- type:
  - NONE
  - NORMAL
  - SLOW

- dynamics:
  - AUTO
  - FIXED

Reply: none

Example:

- FILTER,NORMAL,FIXED
- FILTER,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.
Function: Set frequency response analyser mode.
Description: Set frequency response analyser mode.
Format: FRA
Arguments: 
Reply: none
Example: FRA
Notes: This command has the same effect as MODE,GAINPH. FRA, GAINPH, TFA are aliases for the same command.
**FRA?**

**Function:** frequency response analyser query

**Description:**

**Format:**
FRA?

or:
FRA?SWEEP

or:
FRA,SWEEP?

**Arguments:**
none, or SWEEP

**Reply:**
6 data values separated by commas
freq,mag1,mag2,db,phase, delay
one line per result for sweep data

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

**Notes:**
FRA? waits for next unread data.
FRA?SWEEP does not wait for new data – data can be read multiple times.
FRA, GAINPH, TFA are aliases for the same command
Function: Set the output frequency
Description: Sets the generator output frequency in Hz.
Format: FREQUE,frequency
Arguments: frequency in Hz
Reply: none
Example: FREQUE,5e4 (set frequency to 50kHz)
Notes:
FSWEEP

Function: Set the frequency sweep parameters

Description: Sets the start frequency in Hz, the end frequency, the number of steps and log/linear for the selected function.

Format: FSWEEP,steps,start,end,type

Arguments:  
steps: number of steps
start: start frequency in Hz
end: end frequency in Hz
type: LOGARI, LINEAR

Reply: none

Example: MODE,GAINPH
          FSWEEP,50,1000,1e6
          (set 50 steps between 1kHz and 1MHz)

Notes: It is not necessary to send all the arguments, but if they must be in the specified order. The action at the end of the sweep is specified in the OUTPUT command.
GAINPH

Function: Set gain/phase analyser mode.
Description: Set gain/phase analyser mode.
Format: GAINPH
Arguments: 
Reply: none
Example: GAINPH
Notes: This command has the same effect as MODE,GAINPH.
FRA, GAINPH, TFA are aliases for the same command.
**Function:** Gain/phase query

**Description:**
Read gain/phase analyser results.
Sets gain/phase analyser mode if not already set.
Waits for next unread data if necessary.
Clears new data available bit read by DAV?

**Format:**
GAINPH?

**or:**
GAINPH?SWEEP

**or:**
GAINPH,SWEEP?

**Arguments:**
none, or SWEEP

**Reply:**
6 data values separated by commas
freq,mag1,mag2,db,phase,delay
one line per result for sweep data

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

**Notes:**
GAINPH? waits for next unread data.
GAINPH?SWEEP does not wait for new data – data can be read multiple times.
**Function:** Set harmonic analyser mode.

**Description:** Set harmonic analyser mode and parameters.

**Format:**

```
HARMON, scan, parameter, harmonic, max
```

**Arguments:**

- **scan:**
  - SINGLE
  - THDD
  - THDS

- **parameter:**
  - PERCEN
  - DB

- **harmonic:**
  - single harmonic 2-50 for display

- **max:**
  - harmonic series 2-50 for series thd

**Reply:** none

**Example:**

```
HARMON, SINGLE, PERCEN, 3
```

**Notes:** It is not necessary to send any arguments, but if any are sent they must be in the specified order.
<table>
<thead>
<tr>
<th>Function:</th>
<th>Harmonic analyser query</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description:</td>
<td>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?</td>
</tr>
<tr>
<td>Format:</td>
<td>HARMON?</td>
</tr>
<tr>
<td>or:</td>
<td>HARMON,SWEEP?</td>
</tr>
<tr>
<td>or:</td>
<td>HARMON,SERIES?</td>
</tr>
<tr>
<td>Arguments:</td>
<td>none, or SWEEP, or SERIES</td>
</tr>
</tbody>
</table>
| Reply:    | 7 data values separated by commas:  
|           | single: freq,mag1,mag2,h1,h2,harm1,harm2  
|           | thd: freq,mag1,mag2,thd1,thd2,harm1,harm2  
|           | 1 line per result for sweep data  
|           | series: 6 data values separated by commas:  
|           | mag1,%1,phase1,mag2,%2,phase2 |
| Example:  | HARMON?  
|           | data returned |
| Notes:   | HARMON? waits for next unread data. HARMON?SWEEP does not wait for new data – data can be read multiple times. |
HOLD

Function: Set/clear HOLD mode
Description: HOLD mode stops the instrument from updating the measured values
Format: HOLD,value
Arguments: value:
ON
OFF
Reply: none
Example: HOLD,ON
Notes:
INPUT

Function: Set input mode
Description: Selects the input type of the instrument
Format: INPUT,channel,type
Arguments: channel:
  CH1
  CH2
type:
  DISABLE
  VOLTAGE
  SHUNT
Reply: none
Example: INPUT,CH1,SHUNT
Notes:
**INTYPE**

**PSM1735 NumetriQ only**

- **Function:** Set input connection
- **Description:** Selects the input type for each channel to be non-inverting, inverting, or differential.
- **Format:** `INPUT,channel,type`
- **Arguments:**
  - channel:
    - CH1
    - CH2
  - type:
    - MAIN
    - SECOND
    - DIFFER
- **Reply:** none
- **Example:** `INPUT,CH1,DIFFERENTIAL`
- **Notes:**

---

2-47
<table>
<thead>
<tr>
<th>KEYBOA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Function:</strong></td>
</tr>
<tr>
<td><strong>Description:</strong></td>
</tr>
<tr>
<td><strong>Format:</strong></td>
</tr>
<tr>
<td><strong>Arguments:</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Reply:</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td><strong>Notes:</strong></td>
</tr>
</tbody>
</table>
LCR

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

Function: LCR meter query

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

Format: LCR?
or: LCR?SWEEP
or: LCR,SWEEP?

Arguments: none, or SWEEP

Reply: 14 data values separated by commas:
freq, mag1, mag2, impedance,
phase, series R, series L, series C,
//R, //L, //C, tanδ, Q, reactance
or 11 data values separated by commas:
freq, mag1, mag2, impedance,
phase, resistance, reactance,
admittance, phase, conductance,
susceptance

sweep reply: 8 data values per line per sweep result:
freq,Q,tanδ,impedance,phase,L,C,R

Example: OUTPUT,ON
LCR?
data returned

Notes: LCR? waits for next unread data. LCR?SWEEP does not wait for new data – data can be read multiple times.
## LOWFRE

**Function:** Set low frequency mode  
**Description:** Sets the low frequency option for external frequency measurement.  
**Format:** LOWFRE,\textit{value}  
**Arguments:** \textit{value}:  
\begin{itemize}  
\item ON  
\item OFF  
\end{itemize}  
**Reply:** none  
**Example:** LOWFRE,ON  
**Notes:** LOWFRE is mainly used for measuring low frequencies when not using the instrument generator for the frequency reference. However, as it applies digital filtering, it may also be useful when analysing any signals below a few hundred Hertz.
**MARKER**

Function: Set frequency marker

Description: Enable or disable frequency marker.

Format: `MARKER,value,frequency`

Arguments:
- value: `ON` or `OFF`
- frequency: marker frequency in Hz

Reply: none

Example:
- `MARKER,OFF`
- `MARKER,ON,25e3`

Notes: It is not necessary to send the frequency when enabling the marker if it has already been set.
MODE

Function: Set mode
Description: Sets the fundamental operating mode of the instrument.
Format: MODE,type
Arguments: type:
- SIGGEN (signal generator only)
- VRMS (rms voltmeter)
- GAINPH (gain/phase analyser)
- VECTOR (vector voltmeter)
- POWER (power meter)
- LCR (LCR meter)
- HARMON (harmonic analyser)
- TXA (transformer analyser)
Reply: none
Example: MODE,GAINPH
Notes: MODE sets the measurement mode of the instrument
OFFSET

Function: Set the output offset
Description: Sets the output generator offset in Volts.
Format: OFFSET,offset
Arguments: offset in Volts
Reply: none
Example: OFFSET,5e-3 (set offset to 5mV)
Notes:
OUTPUT

Function: Set output

Description: Turns the output on or off, or sets the level mode to dBm or voltage. Also specifies the action at the end of a sweep.

Format: OUTPUT,command,sweep,phase

Arguments: command:
- OFF
- ON
- VOLT
- DBM
- DCONLY (PsimetriQ only)
- PHRESE (PsimetriQ only)

sweep:
- OFF
- ON
- DCONLY
- PHRESE (PsimetriQ only)

phase:
- phase to switch off (PsimetriQ only)

Reply: none

Example: OUTPUT,ON

Notes: For safety, the output defaults to off and must be turned on explicitly. It is not necessary to send all the arguments, but if they are sent they must be in the specified order.
PAV

Function: Set phase angle voltmeter mode.

Description: Set phase angle voltmeter mode and parameter.

Format: PAV, \textit{parameter}, lvdt scale

Arguments: parameter:
\begin{itemize}
  \item INPHAS
  \item QUADR
  \item TANPHI
  \item MAGNIT
  \item POLAR
  \item A2/1
  \item RMS2
  \item RMS2/1
  \item LVDT-D
  \item LVDT-R
\end{itemize}

lvdt scale: scale factor in m for lvdt applications

Reply: none

Example: PAV, LVDT-D, 0.1

Notes: It is not necessary to send any arguments, but those that are sent must be in the specified order. PAV and VECTOR are aliases for the same command.
PAV?

Function: Phase angle voltmeter query

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

Format: PAV?
or: PAV?SWEEP
or: PAV,SWEEP?

Arguments: none, or SWEEP

Reply: 7 data values separated by commas: freq,mag1,mag2,parameter,phase,a,b
1 line per result for sweep data

Example: FREQ,3300 OUTPUT,ON PAV?LVDT_D,0.1 data returned

Notes: PAV? waits for next unread data. PAV?SWEEP does not wait for new data – data can be read multiple times. PAV and VECTOR are aliases for the same command.
<table>
<thead>
<tr>
<th>Function:</th>
<th>Set phase meter mode.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description:</td>
<td>Select phase meter mode.</td>
</tr>
<tr>
<td>Format:</td>
<td>PHASE, <em>mode</em>, <em>window size</em></td>
</tr>
<tr>
<td>Arguments:</td>
<td>mode:</td>
</tr>
<tr>
<td></td>
<td>NORMAL</td>
</tr>
<tr>
<td></td>
<td>STREAM</td>
</tr>
<tr>
<td></td>
<td>window size</td>
</tr>
<tr>
<td></td>
<td>streaming window size in seconds</td>
</tr>
<tr>
<td>Reply:</td>
<td>none</td>
</tr>
<tr>
<td>Example:</td>
<td>PHASE</td>
</tr>
<tr>
<td>Notes:</td>
<td>It is not necessary to send any arguments</td>
</tr>
<tr>
<td></td>
<td>but if they are sent they must be in the</td>
</tr>
<tr>
<td></td>
<td>specified order.</td>
</tr>
</tbody>
</table>
**PHASE?**

**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:** PHASE?

**Arguments:** none

**Reply:** 2 data values separated by commas freq,phase

**Example:** PHASE?
1.8396E2,5.0342E-2,

**Notes:**
The phase convention can be set to 0° to -360°, 0° to +360°, or +180° to -180° in the SYSTEM menu or using CONFIG,6,0-2 (see appendix).
**Function:** Set phase convention

**Description:** Set phase convention

**Format:** PHCONV, *convention*

**Arguments:**

*convention*:

- 180: -180 to +180
- -360: 0 to -360
- +360: 0 to +360

**Reply:** none

**Example:** PHCONV, -360

**Notes:**
PHREF

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

Function: Set up power meter mode.

Description: Configure power meter with integration type

Format: `POWER,integration type`

Arguments: integration type:
- MAGNITUDE
- SIGNED

Reply: none

Examples: `POWER,SIGNED`

POWER

Notes: It is not necessary to send the integration type argument.
**POWER?**

**Function:** Read power meter results

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

**Format:**
- POWER?
- POWER? *results*
- POWER, *results*?

**Arguments:**
- results:
  - WATTS
  - RMS
  - INTEGR

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

**Example:** POWER?WATTS

**Notes:**
**PPORT**

**Function:** Set the parallel port  
**Description:** Force the logic level on the parallel port data lines  
**Format:** `PPORT, value`  
**Arguments:** decimal value to be written to the port  
**Reply:** None  
**Example:** `PPORT, 64`  
**Notes:** The parallel port may be used as an 8 bit logic level output port and a 4 bit logic level input port.
<table>
<thead>
<tr>
<th><strong>Function:</strong></th>
<th>Read the parallel port</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description:</strong></td>
<td>Read the logic level on the parallel port control input lines</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td>PPORT?</td>
</tr>
<tr>
<td><strong>Arguments:</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>Reply:</strong></td>
<td>Single integer data value</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>PPORT? 12</td>
</tr>
<tr>
<td><strong>Notes:</strong></td>
<td>The parallel port may be used as a 4 bit logic level input port and an 8 bit logic level output port.</td>
</tr>
</tbody>
</table>
**PROGRA**

**Function:** Access non volatile program stores.

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

**Format:** PROGRA,<i>function</i>,<i>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.
<table>
<thead>
<tr>
<th>Function</th>
<th>Identify current program.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Reads the name of the last program to be loaded or recalled.</td>
</tr>
<tr>
<td>Format</td>
<td>PROGRA?</td>
</tr>
<tr>
<td>Arguments</td>
<td>none</td>
</tr>
<tr>
<td>Reply</td>
<td>text string</td>
</tr>
<tr>
<td>Example</td>
<td>PROGRA?</td>
</tr>
<tr>
<td></td>
<td>factory default</td>
</tr>
<tr>
<td>Notes:</td>
<td></td>
</tr>
</tbody>
</table>
### RANGE

<table>
<thead>
<tr>
<th>Function:</th>
<th>Set channel ranging.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description:</td>
<td>Select minimum range and range control for a given input channel.</td>
</tr>
<tr>
<td>Format:</td>
<td>RANGE,channel,ranging,range</td>
</tr>
<tr>
<td>Arguments:</td>
<td>channel:</td>
</tr>
<tr>
<td></td>
<td>CH1</td>
</tr>
<tr>
<td></td>
<td>CH2</td>
</tr>
<tr>
<td></td>
<td>ranging:</td>
</tr>
<tr>
<td></td>
<td>AUTO</td>
</tr>
<tr>
<td></td>
<td>UPAUTO</td>
</tr>
<tr>
<td></td>
<td>MANUAL</td>
</tr>
<tr>
<td></td>
<td>range:</td>
</tr>
<tr>
<td></td>
<td>nominal range value</td>
</tr>
<tr>
<td>Reply:</td>
<td>none</td>
</tr>
<tr>
<td>Example:</td>
<td>RANGE,CH2,MANUAL,3V</td>
</tr>
<tr>
<td>Notes:</td>
<td></td>
</tr>
</tbody>
</table>
**RESOLU**

Function: Set the data resolution

Description: Data is returned in scientific format with exponent and mantissa. The resolution of the mantissa may be selected to be 5 digit (NORMAL) or 6 digit (HIGH).

Format: RESOLU,format

Arguments: format:

- NORMAL (5 digit mantissa)
- HIGH (6 digit mantissa)
- BINARY (raw binary format)

Reply: none

Example: RESOLU,HIGH

Notes: The resolution only changes the real number replies.

Data format for NORMAL is:

-1.2345E-00

Data format for HIGH is:

-1.23456E-00

The signs of the mantissa and exponent, shown as [-] in the above examples, are only sent if they are negative.

Data format for BINARY is a proprietary floating point format which returns raw data in a minimum number of data bytes.
<table>
<thead>
<tr>
<th>Function</th>
<th>Rezero front end</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Request the DSP to re-compensate for dc offset and compute a new autozero</td>
</tr>
<tr>
<td>Format</td>
<td>REZERO</td>
</tr>
<tr>
<td>Arguments</td>
<td>none</td>
</tr>
<tr>
<td>Reply</td>
<td>none</td>
</tr>
<tr>
<td>Example</td>
<td>REZERO</td>
</tr>
<tr>
<td>Notes</td>
<td></td>
</tr>
<tr>
<td>RUN?</td>
<td>RUN?</td>
</tr>
<tr>
<td>------------</td>
<td>------</td>
</tr>
<tr>
<td>Function:</td>
<td>Returns status of various internal processes</td>
</tr>
<tr>
<td>Description:</td>
<td>Returns status of various internal processes</td>
</tr>
<tr>
<td>Format:</td>
<td>RUN?</td>
</tr>
<tr>
<td>Arguments:</td>
<td>none</td>
</tr>
</tbody>
</table>
| Reply:     | Bit 0 : Sweep running  
|            | Bit 1 : Fast sweep running  
|            | Bit 2 : Integrator running  
|            | Bit 3 : Datalog running  
|            | Bit 4 : Fast datalog running  
|            | Bit 5 : Fast analog output running  
|            | Bit 6 : Not used  
|            | Bit 7 : Generator active |
| Example:   | RUN?  
|           | 9 |
| Notes:     | |
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:
**SHUNT**

Function:  Set channel shunt value

Description:  Set the resistance factor of a current shunt to be divided into the measured voltage for a given input channel.

Format:  \texttt{SHUNT,channel,resistance}

Arguments:  
- \texttt{channel}:
  - CH1
  - CH2
- \texttt{resistance}:
  - shunt resistance in Ohms

Reply:  none

Example:  \texttt{SHUNT,CH1,10}

Notes:  The SHUNT command is still accepted if the channel has not been configured for current. The value stored will be used when the channel is configured for current.
Function: Set single measurement mode

Description: Selects the measurement mode to be single measurements instead of normal continuous measurements.

Format: \texttt{SINGLE,}\textit{value}

Arguments: value:
- \texttt{ON}
- \texttt{OFF}

Reply: none

Example:
\texttt{SINGLE,ON}
\texttt{SPEED,WINDOW,0.8}
\texttt{OUTPUT,ON,OFF}
\texttt{*TRG; FRA?}
data returned
\texttt{*TRG; FRA?}
data returned

Notes: Single measurement mode allows the output to be turned on for only one measurement then turned off again.
**SPEED**

Function: Sets the measurement speed

Description: Sets the minimum window size for the measurement.

Format: SPEED,value
        SPEED,WINDOW,time

Arguments: value:
          FAST
          MEDIUM
          SLOW
          VSLOW
          WINDOW

Reply: none

Example: SPEED,SLOW
        SPEED,WINDOW,0.1

Notes:
<table>
<thead>
<tr>
<th>SSWEEP</th>
<th>SSWEEP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Function:</strong></td>
<td>Access non volatile sweep results stores.</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
<td>Recall, store or delete non-volatile sweep results store.</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td><code>SSWEEP, function, number</code></td>
</tr>
</tbody>
</table>
| **Arguments:** | function:  
RECALL  
STORE  
DELETE  
number  
1-30  |
| **Reply:** | none |
| **Example:** | `SSWEEP,RECALL,13` |
| **Notes:** | The sweep data can be read back using the sweep query command for each mode eg. FRA,SWEEP? for an FRA sweep. |
**START**

Function: Start sweep
Description: Initiate sweep in those functions that have a sweep or resets filtering in others.
Format: START
Arguments: none
Reply: none
Example: MODE,GAINPH (set gain phase analyser) START
Notes:
**STATUS?**

Function: Read back channel ranging status.

Description: Read back condition of selected channel:
- range number (1-16)
- range text
- overflow/underflow status

Format: **STATUS,channel?**
or: **STATUS?channel**

Arguments: channel:
- CH1
- CH2

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?**
- 6,3V,OK

Notes:
Function: Stop sweep
Description: Stop an active sweep, or data streaming.
Format: STOP
Arguments: none
Reply: none
Example: MODE,PHASE,STREAM,0.01
START
read data values as required
STOP
read remaining data values

Notes:
STREAM

Function: Set data streaming mode
Description: Set instrument ready for data streaming with specified window
Format: STREAM, control, window
Arguments:
  control
    ENABLE
    DISABL
  window
    streaming window size in seconds
Reply: none
Example:
  PAV, TANPHI
  STREAM, ENABLE, 0.01
  START
  read data values as required
  STOP
  read remaining data values

Notes: Data streaming is valid for phase meter and phase angle voltmeter (vector voltmeter) modes.
STREAM?

Function: Start to read streaming data
Description: Start to read streaming data
Format: STREAM?
STREAM,max?
STREAM?max
Arguments: max
maximum number of values
none
return all data
Reply: data stream separated by commas
Example: PAV,TANPHI
STREAM,ENABLE,0.01
START
STREAM?
data,data,data,data,data, ......
Notes: This command is only used with IEEE488 (GPIB) - streaming starts immediately with RS232.
### SUSPEND

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

**Notes:**
<table>
<thead>
<tr>
<th>Function:</th>
<th>Set transfer function analyser mode.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description:</td>
<td>Set transfer function analyser mode.</td>
</tr>
<tr>
<td>Format:</td>
<td>TFA</td>
</tr>
<tr>
<td>Arguments:</td>
<td></td>
</tr>
<tr>
<td>Reply:</td>
<td>none</td>
</tr>
<tr>
<td>Example:</td>
<td>TFA</td>
</tr>
<tr>
<td>Notes:</td>
<td>This command has the same effect as MODE,GAINPH. FRA, GAINPH, TFA are aliases for the same command.</td>
</tr>
</tbody>
</table>
Function: transfer function analyser query

Description: Read transfer function analyser results. Sets transfer function analyser mode if not already set. Waits for next unread data if necessary. Clears new data available bit read by DAV?

Format: TFA?

or: TFA?SWEEP

or: TFA,SWEEP?

Arguments: none, or SWEEP

Reply: 6 data values separated by commas

freq,mag1,mag2,db,phase,delay

one line per result for sweep data

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

Notes: TFA? waits for next unread data. TFA?SWEEP does not wait for new data – data can be read multiple times. FRA, GAINPH, TFA are aliases for the same command
**TXA**

**Function:** Set transformer analyser mode.

**Description:** Set transformer analyser mode, test, fixture, and resistances

**Format:** TXA, *test*, *fixture*, source, load

**Arguments:**
- **test:**
  - TXTR
  - TXL
  - TXLL
  - TXACR
  - TXDCR
  - TXIWC
  - TXMAGI
  - TXRLOS
  - TXILOS
  - TXHARM
  - TXTHD
  - TXLBAL
- **fixture:**
  - NONE
  - TAF01
  - TAF02

**source resistance**

**load resistance**

**Reply:** none

**Example:** TXA, TXILOS, TAF01, 135, 25.4

**Notes:** It is not necessary to send any arguments, but they must be in the specified order.
**TXA?**

**Function:** Transformer analyser query

**Description:** Read transformer analyser results. Sets transformer analyser mode if not already set. Waits for next unread data if necessary. Clears new data available bit read by DAV?

**Format:**
- TXA?
- TXA?SWEEP
- TXA,SWEEP?

**Arguments:** none, or SWEEP

**Reply:** 3 or 4 data values separated by commas:
- freq,mag1,mag2,parameter
- txdcr dc1,dc2,resistance
- txmagi freq,rms1,rms2,watts

sweep reply: Data as above, one line per sweep result.

**Example:** TXA?
data returned

**Notes:** TXA? waits for next unread data. TXA?SWEEP does not wait for new data – data can be read multiple times.
**TXTEST**

<table>
<thead>
<tr>
<th>Function:</th>
<th>Set transformer analyser test.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description:</td>
<td>Set transformer analyser test and winding(s).</td>
</tr>
<tr>
<td>Format:</td>
<td>TXTEST,test,winding1,winding2</td>
</tr>
<tr>
<td>Arguments:</td>
<td>test:</td>
</tr>
<tr>
<td></td>
<td>as TXA command</td>
</tr>
<tr>
<td></td>
<td>winding1:</td>
</tr>
<tr>
<td></td>
<td>W1</td>
</tr>
<tr>
<td></td>
<td>W2</td>
</tr>
<tr>
<td></td>
<td>W3</td>
</tr>
<tr>
<td></td>
<td>W4</td>
</tr>
<tr>
<td></td>
<td>(TAF02 only)</td>
</tr>
<tr>
<td></td>
<td>W2+3</td>
</tr>
<tr>
<td></td>
<td>(TAF01 only)</td>
</tr>
<tr>
<td></td>
<td>winding2:</td>
</tr>
<tr>
<td></td>
<td>(turns ratio test only)</td>
</tr>
<tr>
<td></td>
<td>W1:W2</td>
</tr>
<tr>
<td></td>
<td>W2:W1</td>
</tr>
<tr>
<td></td>
<td>W1:W3</td>
</tr>
<tr>
<td></td>
<td>W3:W1</td>
</tr>
<tr>
<td></td>
<td>W2:W3</td>
</tr>
<tr>
<td></td>
<td>W3:W2</td>
</tr>
<tr>
<td></td>
<td>W1:W4</td>
</tr>
<tr>
<td></td>
<td>(TAF02 only)</td>
</tr>
<tr>
<td></td>
<td>W4:W1</td>
</tr>
<tr>
<td></td>
<td>(TAF02 only)</td>
</tr>
<tr>
<td></td>
<td>W1:2+3</td>
</tr>
<tr>
<td></td>
<td>(TAF01 only)</td>
</tr>
<tr>
<td></td>
<td>W2+3:1</td>
</tr>
<tr>
<td></td>
<td>(TAF01 only)</td>
</tr>
<tr>
<td>Reply:</td>
<td>none</td>
</tr>
<tr>
<td>Example:</td>
<td>TXTEST,TXTR,W1,W2:W3</td>
</tr>
<tr>
<td>Notes:</td>
<td>It is not necessary to send all the arguments, but they must be in the specified order.</td>
</tr>
</tbody>
</table>
**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?
Newton4th Ltd
R&D department
PsimetriQ #4
```

**Notes:**
VECTOR

Function: Set vector voltmeter mode.

Description: Set vector voltmeter mode and parameter.

Format: VECTOR, parameter, lvdt scale

Arguments: parameter:
NONE
INPHAS
QUADR
TANPHI
POLAR
A2/A1
RMS2/1
LVDT-D
LVDT-R

lvdt scale:
scale factor in m for lvdt applications

Reply: none

Example: VECTOR,LVDT-D,0.1

Notes: It is not necessary to send any arguments, but those that are sent must be in the specified order. PAV and VECTOR are aliases for the same command.
VECTOR?

Function: Vector voltmeter query
Description: Read vector voltmeter results.
Sets vector voltmeter mode if not already set.
Waits for next unread data if necessary.
Clears new data available bit read by DAV?

Format: VECTOR?
or: VECTOR?SWEEP
or: VECTOR,SWEEP?
Arguments: none, or SWEEP
Reply: 7 data values separated by commas:
freq,mag1,mag2,parameter,phase,a,b
1 line per result for sweep data
Example: FREQ,3300
OUTPUT,ON
VECTOR?LVDT_D,0.1
data returned

Notes: VECTOR? waits for next unread data.
VECTOR?SWEEP does not wait for new data – data can be read multiple times.
PAV and VECTOR are aliases for the same command.
<table>
<thead>
<tr>
<th><strong>VERSIO?</strong></th>
<th><strong>VERSIO?</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Function:</strong></td>
<td>Read the instrument code versions.</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
<td>Returns an ASCII string with the details of the various parts of the instrument firmware.</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td>VERSIO?</td>
</tr>
<tr>
<td><strong>Arguments:</strong></td>
<td>none</td>
</tr>
<tr>
<td><strong>Reply:</strong></td>
<td>date code, type, cpu, dsp, fpga, boot</td>
</tr>
<tr>
<td><strong>Examples:</strong></td>
<td>VERSION?</td>
</tr>
<tr>
<td></td>
<td>PQ3504,1,1.21,1.21,1.21,1.02</td>
</tr>
<tr>
<td><strong>Notes:</strong></td>
<td>This data can be displayed on the screen by pressing SYSTEM then BACK</td>
</tr>
</tbody>
</table>
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?

Function: Read true rms voltmeter results
Description: Reads back latest voltmeter results. Sets voltmeter mode if not already set. Waits for next unread data if necessary. Clears new data available bit read by DAV?

Format:
- VRMS?
- VRMS,results?
- VRMS?results

Arguments:
- results:
  - RMS
  - SURGE

Reply:
- RMS:
  - 8 data values separated by commas
    - rms1,2,dc1,2,ac1,2,dbm1,2
- SURGE:
  - 6 data values separated by commas
    - pk1,2,cf1,2,surge1,2
- no argument:
  - 14 data values separated by commas
    - RMS results then SURGE

Example: VRMS?RMS

Notes: As VRMS? does not send the same data twice but waits instead for the next result, it is not necessary to check the data available bits before sending the VRMS? command.
WAVEFO

Function: Set the output waveform

Description: Selects the output waveform for the signal generator.

Format: WAVEFO,type

Arguments: type:
SINEWA (sine wave)
TRIANG (triangle wave)
SQUARE (square wave)
LEADIN (leading sawtooth)
TRAILI (trailing sawtooth)

Reply: None

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

Notes: PSM1735 NumetriQ only has SINEWA or SQUARE option
**ZERO**

Function: Apply or remove the zero

Description: Applies or removes a zero function depending on the measurement mode (same as pressing ZERO key). Performs lead compensation in LCR mode.

Format:

- `ZERO`
- `ZERO,DELETE`
- `ZERO,DB,offset`
- `ZERO,PHASE,offset`
- `ZERO,SINGLE`
- `ZERO,SWEEP,steps,start,finish`
- `ZERO,OPEN`
- `ZERO,SHORT`
- `ZERO,STORE`
- `ZERO,RECALL`

Arguments:

- `offset`: offset value
- `steps`: LCR sweep compensation steps
- `start`: LCR compensation start frequency
- `stop`: LCR compensation stop frequency

Reply: none

Example:

- `ZERO,SWEEP,100,1e3,1e6`
- `ZERO,OPEN`

  *performs open circuit compensation*

Notes:
Function: Sets the display zoom parameters.

Description: Sets the zoom level and data.

Format: ZOOM, level, data1, data2, data3, data4

Arguments:
- level:
  - 0 – normal
  - 1 – 2 line display (zoom level 1)
  - 2 – single line display (zoom level 2)

- data1:
  - first data (zoom level 1)
  - or data for single line (zoom level 2)

- data2-4:
  - other data (zoom level 1)

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

Reply: None

Example: VRMS
  ZOOM, 1, 1, 12 (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?**

**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)

- **data1-4:**
  - zoom data

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

**Example:** ZOOM?

1,1,129,0,0  (level 1, ch1 rms, ch2 rms)

**Notes:**
Appendices

COMMAND SUMMARY

CONFIGURABLE PARAMETERS
command format

*CLS
*ESE,value
*ESE?
*ESR?
*IDN?
*OPC?
*RST
*SRE,value
*SRE?
*STB?
*TRG
*TST?
*WAI

reply format

single integer data value

single integer data value

company, product, serial no, version

0 or 1

single integer data value

single integer data value

single integer data value

single integer data value

single integer data value

single integer data value

single integer data value

single integer or real data value

single integer data value

single integer data value

index, time, data… one record per line

single integer data value

single integer data value

f.req, mag1, mag2, dB, phase, delay

n lines of FRA? data

ABORT

ACTRIM, channel, level, tol

ALARM, latch, sounder

ALARM?

ALARME, value

ALARME?

ALARM1, type, data, high, low

ALARM2, type, data, high, low

AMPLIT, amplitude

ANALOG, type, value 1, value 2

BANDWI, type

BEEP

BLANKI, on/off, threshold

CONFIG, parameter, data

CONFIG, parameter?

COUPLI, channel, coupling

COUPLI, channel?

CYCLES, cycles

DATALO, function, interval

DATALO, start, records?

DAV?

DAVER, value

DAVER?

DELAY, time

FILTER, type, dynamics

FRA

FRA?

FRA, SWEEP?
PSM17xx communications manual

FREQUENCY,frequency
FSWEEP,steps,start,end,log
GAINPH
GAINPH?, freq,mag1,mag2,dB,phase,delay
GAINPH-SWEEP?, n lines of GAINPH? data
HARMON,scan,para,h,hmax
HARMON?, freq,mag1,mag2,hmag1,hmag2,h1,h2
  or
HARMON-SERIES?, freq,mag1,mag2,thd1,thd2,h1,h2
HARMON-SWEEP?, n lines of HARMON? data
HOLD, on/off
INPUT,channel,type
INPUT,channel?, single integer data value
INTYPE,channel,type
KEYBOA,value
LCR,conditions,param,head
LCR?, freq, mag1, mag2, impedance, phase, R, L, C (series), R, L, C (parallel), tanδ, Q, reactance
  or
LCR-SWEEP?, freq, mag1, mag2, impedance, phase, resistance, reactance, admittance, phase, conductance, susceptance
  n lines of data:
  freq,QF,tanδ,impedance,phase,L,C,R
  freq,QF,tanδ,admittance,phase,L,C,R
LOWFRE, on/off
MARKER, on/off, frequency
MODE,type
OFFSET, offset
OUTPUT, type, sweep, phase
PAV, parameter, scaling
PAV?, freq,mag1,mag2,parameter,phase,a,b
PAV-SWEEP?, n lines of VECTOR? data
PHASE
PHASE?, freq,phase
PHASE-STREAM,window
PHCONV, convention
PHREF, channel
POWER, integration type
POWER-WATTS?, W,W.f,VA,VA.f,pf,pf.f,Wdc,W.h,freq
POWER-RMS?, rms1,rms2,dc1,dc2,fnd1,fnd2,φ1,φ2
POWER-INTEGR?, Wh,Wh.f,VAh,VAh.f,pf,pf.f,Ah,Ah.f,t
PPORT, value
PPORT?, single integer data value
PROGRAM, function, number
PROGRAM?
CR terminated text string
RANGE, ch, ranging, range
RESOLU, format
REZERO
SCALE, channel, factor
SCALE, channel?
single real data value
SHUNT, channel, resistance
SHUNT, channel?
single real data value
SINGLE, on/off
SPEED, speed
SSWEEP, function, number
START
STATUS, channel?
range number, range text, over/low/ok
STOP
STREAM, enable, window
STREAM, disable
STREAM?
data, data, data, data, data, ...
SUSPEND, on/off
TFA
TFA?
freq, mag1, mag2, dB, phase, delay
TFA, SWEEP?
n lines of TFA? data
TXA, test, fixture, load, source
TXA?
freq, mag1, mag2, parameter
TXA, SWEEP?
n lines of TXA? data
TXTEST, test, wind1, wind2
USER?
VECTOR, parameter, scaling
VECTOR?
freq, mag1, mag2, parameter, phase, a, b
VECTOR, SWEEP?
n lines of VECTOR? data
VERSION?
datecode, type, cpu, dsp, fpga, boot
VRMS
VRMS?
RMS? data followed by SURGE?
VRMS, RMS?
rms1, rms2, dc1, dc2, ac1, ac2, db1, db2
VRMS, SURGE?
.pk1, pk2, cf1, cf2, surge1, surge2
WAVEFO, type
ZERO
ZERO, DELETE
ZOOM, level, d1, d2, d3, d4
ZOOM?
level, d1, d2, d3, d4
calibration commands

CALAPP
CALCOM,freq
CALDCO,value
CALFIL,index,value
CALFIL? six real data values
CALFRQ,index,freq
CALFRQ? seven real data values
CALHF,index,value
CALIBR,index,value
CALIBR? single integer data value
CALIDS,string
CALIDS? string
CALOUT,index,value
CALPHA,index
CALRES
CALSAV,password
CALSNO,serial number
CALSTR,string
CALSTR? string
Appendix B – Configurable parameters

All parameters can be accessed using the CONFIG command:

CONFIG,number,parameter?
CONFIG,parameter,data

<table>
<thead>
<tr>
<th>Number</th>
<th>Function</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>System parameters</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| 1 | Operating mode, (Sets main mode) | 0=RMS Voltmeter  
1=Frequency Response analyser  
2=Power Meter  
3=LCR Meter  
4= Vector Voltmeter  
5=Harmonic Analyser  
6=Transformer Analyser |
| 3 | Bandwidth, (Acquisition Control) (Numetrix series only) | 0=Auto  
1=Wide |
| 4 | Autozero, (System Options) | 0=Auto  
1=Manual |
| 5 | Low blanking, (System Options & RMS Voltmeter) | 0=Off  
1=On |
| 6 | phase convention, (System Options) | 0=-180° to +180°  
1=0° to -360°  
2=0° to +360° |
| 7 | Generator output, (Output Options) | 1=On  
0=Off |
| 8 | Graph, (System Options) | 0=Dots  
1=Lines |
9  Keyboard beep, (System Options)
   0=Off
   1=On

11  Low frequency mode, (Acquisition Control)
    0=Off
    1=On

12  Speed “window size”, (Acquisition Control, Enter figures)

13  Speed, (Acquisition Control)
    0=Very slow
    1=Slow
    2=Medium
    3=Fast
    4=Window

14  Filter, (Acquisition control)
    0=Normal
    1=Slow
    2=None

15  Filter dynamics, (Acquisition Control, “Filter normal/slow”)
    0=Auto reset
    1=Fixed time

16  Baud rate, (Comms-Remote Options, RS232)
    0=19200
    1=9600
    2=4800
    3=2400
    4=1200

18  Sweep steps, (Sweep Control-Enter step number figures)

19  Sweep start frequency, (Sweep Control-Enter figures)

20  Sweep end frequency, (Sweep Control-Enter figures)

21  Sweep-type, (Sweep Control)
    0=Single
    1=Repeat
Conditions, (LCR Meter)
0=Auto frequency
1=Manual
2=Auto shunt

Shunt, (System Options)
0=Default
1=Manual

**Input parameters**

Input 1 (CH1), (CH1-Input 1)
0=Voltage input
2=External shunt

Input 2 (CH2), (CH2-Input 2)
0=Voltage input
2=External shunt

Minimum range (CH1), (CH1-Input 1)
0=1mv
1=3mv
2=10mv
3=30mv
4=100mv
5=300mv
6=1v
7=3v
8=10v

Minimum range (CH2), (CH2-Input 2)
0=1mv
1=3mv
2=10mv
3=30mv
4=100mv
5=300mv
6=1v
7=3v
8=10v

Autoranging (CH1), (CH1-Input 1)
0=Full Autorange
1=Autorange up
2=Manual
29 Autoranging (CH2), (CH2-Input 2)
   0=Full Autorange
   1=Autorange up
   2=Manual

30 Coupling (CH1), (CH1-Input 1)
   0=ac+dc
   1=ac

31 Coupling (CH2), (CH2-Input 2)
   0=ac+dc
   1=ac

32 Scale (CH1), (CH1-Input, Enter figures)

33 Scale (CH2), (CH2-Input, Enter figures)

34 External shunt (CH1), (CH1-Input, Enter figures)

35 External shunt (CH2), (CH2-Input, Enter figures)

36 Connection (CH1), (CH1-Input, (Numetriq series only)
   0=Main right
   1=secondary left
   2=Differential (both)

37 Connection (CH2), (CH2-Input, (Numetriq series only)
   0=Main right
   1=secondary left
   2=Differential (both)

**General parameters**

38 Resolution, Comms-Remote Options)
   0=Normal
   1=High
   2=Binary

39 Phase reference, (Aquisition Control)
   0=ch1
   1=ch2
PSM17xx communications manual

**Display parameters**

42 Zoom level, (Main Display)
   0=Zoom -
   1=Zoom +
   2=Second zoom +

43 Display zoom characters on line 1
44 Display zoom characters on line 2
45 Display zoom characters on line 3
46 Display zoom characters on line 4

47 Display type, (Main display-datalog or sweep display mode)
   0=Real Time
   1=Table
   2=Graph

**Signal generator parameters**

48 Generator frequency, (Output Options-Enter figures)
49 Generator amplitude, (Output Options-Enter figures)
50 Generator offset, (Output Options-Enter figures)
51 Generator waveform, (Output Options)
   0=Sinewave
   1=Triangle
   3=Square wave
   4=Leading sawtooth
   5=Trailing sawtooth

52 Frequency step, (Output options-Enter figures)
53 Amplitude step, (Output options-Enter figures)
54 Amplitude dBm (Output options-[116 system control]-Enter figures)
55 Generator after sweep, (Sweep Control)
   0=Off
   1=On
Datalog parameters

58  Datalog, (Acquisition Control-memory type)
    0=Disabled
    1=RAM
    2=Non volatile

59  Interval, (Acquisition Control-RAM/Non volatile-Enter time figures)

General parameters

64  Frequency marker, (Sweep Control)
    0=Off
    1=On

65  Marker frequency, (Enter frequency-Graph display-After sweep, alters marker position)

66  Program 1-6 direct load, (System Options)
    0=Disabled
    1=Enabled

67  Parallel port, (Alarm-monitor options)
    0=Disabled
    1=Enabled

Power meter parameters

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

Streaming parameters

95  Data streaming, (Acquisition control-Mode)
    0=Normal
    1=Streaming

96  Window, (Acquisition control-Mode-streaming-Enter figures)
Harmonic analyser parameters

99 Scan, (Harmonic analyser)
   0=Single
   1=Difference thd
   2=Series thd

100 Harmonic, (Harmonic analyser)
   0, 1 & 2=2
   3=3
   4=4
   5=5
   etc up to 64

101 Harmonics (Max), (Harmonic analyser-scan-series thd)
   0, 1 & 2=2
   3=3
   4=4
   5=5
   etc up to 64

102 Parameter, (Harmonic analyser)
   0=%
   1=dB

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

LCR sweep zero parameters

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

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

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

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

System parameters

116 Control, (System options)
   0=Volts
   1=dBm
Step message, (System options)  
0=Enabled  
1=Disabled

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

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

Transformer analyser parameters

Parameter, (Mode-Transformer analyser)  
0=Turns ratio  
1=Inductance  
2=Leakage inductance  
3=AC resistance  
4=DC resistance  
5=Interwinding capacitance  
6=Magnetising current  
7=Return loss  
8=Insertion loss  
9=Single harmonic  
10=thd  
11=Longitudinal balance

Fixture, (Mode-Transformer analyser-Auxiliary control)  
0=None  
1=LCR active head  
2=TAF01  
3=TAF02  
4=Impedance analyser interface  
6=High Z interface

Winding, (Mode-Transformer analyser-Aux control-TAF01/2)  
TAF01 TAF02  
0=W1 0=W1  
1=W2+3 1=W2  
2=W2 2=W3
124 **Turns Ratio**, (Mode-Transformer analyser-Aux control-TAF01/2)

<table>
<thead>
<tr>
<th>TAF01</th>
<th>TAF02</th>
</tr>
</thead>
<tbody>
<tr>
<td>0=W1:W2+3</td>
<td>0=W1:W2</td>
</tr>
<tr>
<td>1=W2+3:W1</td>
<td>1=W2:W1</td>
</tr>
<tr>
<td>2=W1:W2</td>
<td>2=W1:W3</td>
</tr>
<tr>
<td>3=W2:W1</td>
<td>3=W2:W3</td>
</tr>
<tr>
<td>4=W1:W3</td>
<td>4=W1:W4</td>
</tr>
<tr>
<td>5=W3:W1</td>
<td>5=W2:W4</td>
</tr>
<tr>
<td>6=W2:W3</td>
<td>7=W3:W2</td>
</tr>
</tbody>
</table>

125 **Source** (Various parameter options), (Mode-Transformer analyser-Auxiliary control-Enter figures)

126 **Load**, (Insertion loss), (Mode-Transformer analyser-Auxiliary control-Enter figures)

127 **Nominal** (Turns ratio), (Mode-Transformer analyser-Auxiliary control-Enter figures)

**TAF control parameters (TAF01)**

130 **Source**, (Auxiliary control)

| 0=W1 |
| 1=W2+3 |
| 2=W2 |
| 3=W3 |

131 **Load**, (Auxiliary control)

| 0=W1 |
| 1=W2+3 |
| 2=W2 |
| 3=W3 |

132 **Input 1**, (Auxiliary control)

| 0=W1 |
| 1=W2+3 |
| 2=W2 |
| 3=W3 |
133 Input 2, (Auxiliary control)
   0=W1
   1=W2+3
   2=W2
   3=W3

**LCR meter parameters**

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

138 Sweep, (LCR Meter)
   0=Series
   1=Parallel

139 Graph, (LCR Meter)
   0=Single
   1=\(\tan\delta/QF\)
   2=Resistance

140 LCR head shunt, (Auxiliary control-fixture-LCR active head)
   0=Low
   1=Normal
   2=High
   3=Very high

141 Graph, (LCR meter-impedance)
   0=Linear
   1=Log

142 Phase reference, (Mode-LCR-Zero-LCR Compensation-Enter figures)

143 Reference (Value), (Mode-LCR-Zero-LCR Compensation-Enter figures)

144 Reference, (Mode-LCR-Zero-LCR Compensation)
   0=Capacitance
   1=Resistance

B-10
Connection, (LCR Meter)
0=Shunt
1=Divider Zx low
2=Divider Zx high

Gain/Phase analyser parameters
Graph (time selection), (FRA)
0=Phase
1=Delay

dB offset, (FRA-Enter figures)

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

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

System parameters
Minimum cycles, (Acquisition control-Enter figures)

Delay time, (Acquisition control-Enter figures)

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

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

Alarm functions (Monitor 1)
Monitor 1 data, (Alarm-monitor options)
0=Zoom1
1=Zoom2
2=Zoom3
3=Zoom4
PSM17xx communications manual

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

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

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

160  Alarm latch (Alarm type), (Alarm-monitor options)
     0=Off
     1=On

161  Alarm sounder (Alarm type), (Alarm-monitor options)
     0=Enabled
     1=Disabled

162  Analogue output, (Alarm-monitor options)
     0=Disabled
     1=Monitor
     2=Manual

164  Analogue zero (Analog output), (Alarm-Monitor options-Enter figures)

165  Analogue scale (Analog output), (Alarm-Monitor options-Enter figures)

**Alarm functions (Monitor 2)**

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

B-12
168 Alarm 2 type, (Alarm-monitor options)
   0=Disabled
   1=Linear
   2=Alarm if high
   3=Alarm if low
   4=Outside window
   5=Inside window

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

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

**Graph functions**

173 Graph 2 scaling, (Sweep control)
   0=Auto
   1=Manual

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

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

**Phase angle voltmeter parameters**

177 Parameter, (Vector voltmeter)
   0=In-phase
   1=Quadrature
   2=Tanδ
   3=Magnitude
   4=Phase
   5=In-phase ratio
   6=rms
   7=rms2/rms1
   8=LVDT diff
   9=LVDT ratio
   10=User interface

178 Scale factor (LVDT), (Vector voltmeter-Enter figures)
179 Null meter, (Vector voltmeter)
   0=Off
   1=Auto
   2=Manual

180 Upper limit (Null meter), (Vector voltmeter-Enter figures)

181 Offset (Parameter), (Vector voltmeter-Enter figures)

**Trim parameters**

186 ac trim data, (Trim control)
   0=Disabled
   1=CH1
   2=CH2

188 ac level (Trim data), (Vector voltmeter-Enter figures)

190 Trim tolerance (Trim data), (Vector voltmeter-Enter figures)

**Other parameters**

192 Steps, (Sweep control)
   0=Log
   1=Linear

193 Graph 1 scaling, (Sweep control)
   0=Auto
   1=Manual

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

195 Lower limit (Graph 1 scaling), (Sweep control-Enter figures)
Newtons4th Ltd. contact details

Please direct all queries or comments regarding the PSM17xx instruments or this manual to:

Newtons4th 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 Newtons4th 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.