DMM7510 EPICS IOC User Guide
Version 1.0
June/2017

Beam Diagnostics Group (DIG)
Brazilian Synchrotron Light Laboratory (LNLS)
Brazilian Center for Research in Energy and Materials (CNPEM)

Contents

1 DMM7510 IOC

The DMM7510 IOC provides most of the multimeter parameters as EPICS PVs. Its goal is to facilitate the process of building application-specific IOCs which could make use of a DMM7510 general IOC.

2 Document Overview

This document lists the IOC PVs along with their data type, limits, units, description, and related TSP command. In most cases, a PV is a direct mapping of a multimeter parameter, and its description is the same provided for the parameter in the *Model DMM7510 Reference Manual*. The multimeter reference manual provides all the information about the multimeter features and options. After a multimeter function or parameter is well understood, it should be easy to locate the associated PVs in this document.

3 PVs Suffixes

The records in this IOC fall into different categories depending on their input data types. The categories are indicated by the following set of suffixes.

Table 1: PVs Suffixes

Mnemonic	**Name**	**Description**
-SP	Set Point	A non-enumerated value (real number or string). It sets a system parameter.
-RB	Read Back	A non-enumerated value (real number or string). Read-only. It displays the read back value of a parameter, providing confirmation to changes.
-Sel	Selection	Enumerated value. Sets a system parameter.
-Sts	Status	Enumerated value. Read-only. It displays the read back value of an enumerated parameter, providing confirmation to changes.
-Cmd	Command	Binary command. It causes a given action to be executed.

4 PV List

4.1 Measure/Digitize Function

MeasFnc-Sel

Measure Function Selection

enum {

DC_VOLTAGE
AC_VOLTAGE
DC_CURRENT
AC_CURRENT
RESISTANCE
4W_RESISTANCE
DIODE
CAPACITANCE
TEMPERATURE
CONTINUITY
ACV_FREQUENCY
ACV_PERIOD
DCV_RATIO

}

Description: This PV selects the active measure function. When you select a function, settings for other commands that are related to the function become active.

TSP command: dmm.measure.func

MeasFnc-Sts

Measure Function Status

enum {

NONE
DC_VOLTAGE
AC_VOLTAGE
DC_CURRENT
AC_CURRENT
RESISTANCE
4W_RESISTANCE
DIODE
CAPACITANCE
TEMPERATURE
CONTINUITY
ACV_FREQUENCY
ACV_PERIOD
DCV_RATIO

}

Description: This PV shows the active measure function. If a digitize measurement function is active, this PV indicates *NONE*.

TSP command: dmm.measure.func

DigtzeFnc-Sel

Digitize Function Selection

enum {

DIGITIZE_VOLTAGE
DIGITIZE_CURRENT

}

Description: This PV determines which digitize function is active.

TSP command: dmm.digitize.func

DigtzeFnc-Sts

Digitize Function Status

enum {

NONE
DIGITIZE_VOLTAGE
DIGITIZE_CURRENT

}

Description: This PV shows which digitize function is active. If a basic (non-digitize) measurement function is selected, this PV indicates *NONE*.

TSP command: dmm.digitize.func

Function-Sts

Function Status

enum {

DC_VOLTAGE
AC_VOLTAGE
DC_CURRENT
AC_CURRENT
RESISTANCE
4W_RESISTANCE
DIODE
CAPACITANCE
TEMPERATURE
CONTINUITY
ACV_FREQUENCY
ACV_PERIOD
DCV_RATIO
DIGITIZE_VOLTAGE
DIGITIZE_CURRENT

}

Description: This PV shows which measurement or digitize function is active.

TSP command: No command

4.2 Measurement settings

MeasApert-SP

Measure Aperture Set Point

float

unit: seconds

Description: This PV determines the aperture setting for the selected measurement function. The aperture sets the amount of time the ADC takes when making a measurement, which is the integration period for the selected measurement function.

TSP command: dmm.measure.aperture

MeasApert-RB

Measure Aperture Read Back

float

unit: second

Description: This PV shows the aperture setting for the selected measurement function.

TSP command: dmm.measure.aperture

MeasNPLC-SP

Measure Number of Power Line Cycles Set Point

float

Description: This PV sets the time that the input signal is measured for the selected function. The amount of time is specified as the number of power line cycles (NPLCs). Each PLC for 60 Hz is 16.67 ms (1/60) and each PLC for 50 Hz is 20 ms (1/50).

TSP command: dmm.measure.nplc

MeasNPLC-RB

Measure Number of Power Line Cycles Read Back

float

Description: This PV shows the time that the input signal is measured for the selected function, as the number of power line cycles.

TSP command: dmm.measure.nplc

MeasCount-SP

Measure Count Set Point

long

Min=1

Max=1000000

Description: This PV sets the number of measurements to make when a measurement is requested. This PV sets the count for all measure functions.

TSP command: dmm.measure.count

MeasCount-RB

Measure Count Read Back

long

Description: This PV shows the number of measurements to make when a measurement is requested.

TSP command: dmm.measure.count

MRange-SP

Measure Range Set Point

float

Description: This PV determines the positive full-scale measure range. The instrument selects the closest fixed range that is large enough to measure the entered number.

TSP command: dmm.measure.range

MRange-RB

Measure Range Read Back

float

Description: This PV shows the selected positive full-scale measure range.

TSP command: dmm.measure.range

MAutoRange-Sel

Measure Auto Range Selection

bool{

OFF
ON

}

Description: This PV determines if the measurement range is set manually or automatically for the selected function. When auto range is enabled, the range increases at 120 percent of range and decreases occurs when the reading is <10 percent of nominal range.

TSP command: dmm.measure.autorange

MAutoRange-Sts

Measure Auto Range Status

bool{

OFF
ON

}

Description: This PV shows if the measurement range is set manually or automatically for the selected function.

TSP command: dmm.measure.autorange

AutoZero-Sel

Auto Zero Selection

bool{

OFF
ON

}

Description: This PV enables or disables automatic updates to the internal reference measurements (autozero) of the instrument. To ensure the accuracy of readings, the instrument must periodically get new measurements of its internal ground and voltage reference. The time interval between updates to these reference measurements is determined by the integration aperture that is being used for measurements. The time to make the reference measurements is in addition to the normal measurement time. If timing is critical, you can disable autozero to avoid this time penalty. When autozero is set to off, the instrument may gradually drift out of specification. For AC voltage and AC current measurements where the detector bandwidth is set to 3 Hz or 30 Hz, autozero is set on and cannot be changed.

TSP command: dmm.measure.autozero.enable

AutoZero-Sts

Auto Zero Status

bool{

OFF
ON

}

Description: This PV shows if automatic updates to the internal reference measurements (autozero) of the instrument are enabled.

TSP command: dmm.measure.autozero.enable

AZeroOnce-Cmd

Auto Zero Once Command

bool{

OFF
ON

}

Description: Sending 1 or *ON* causes the instrument to refresh the reference and zero measurements once. If the NPLC setting is less than 0.2 PLC, sending autozero once can result in delay of more than a second.

TSP command: dmm.measure.autozero.once()

MeasAutoDly-Sel

Measure Auto Delay Selection

bool{

OFF
ON

}

Description: This PV enables or disables the automatic delay that occurs before each measurement.

TSP command: dmm.measure.autodelay

MeasAutoDly-Sts

Measure Auto Delay Status

bool{

OFF
ON

}

Description: This PV shows if the automatic delay that occurs before each measurement is enabled.

TSP command: dmm.measure.autodelay

MeasImpedance-Sel

Measure Input Impedance Selection

bool{

AUTO
10MOhm

}

Description: This PV determines when the 10 MΩ input divider is enabled for the seslected measure function. Choosing automatic input impedance is a balance between achieving low DC voltage noise on the 100 mV and 1 V ranges and optimizing measurement noise due to charge injection. The Model DMM7510 is optimized for low noise and charge injection when the DUT has less than 100 kΩ input resistance. When the DUT input impedance is more than 100 kΩ, selecting an input impedance of 10 MΩ optimizes the measurement for lowest noise on the 100 mV and 1 V ranges. For the 10 V to 1000 V ranges, both input impedance settings achieve low charge injection.

TSP command: dmm.measure.inputimpedance

MeasImpedance-Sts

Measure Input Impedance Status

bool{

AUTO
10MOhm

}

Description: This PV shows when the 10 MΩ input divider is enabled for the selected measure function.

TSP command: dmm.measure.inputimpedance

MeasLineSync-Sel

Measure Line Synchronization Selection

bool{

OFF
ON

}

Description: This PV determines if line synchronization is used during the measurement. When line synchronization is enabled, measurements are initiated at the first positive-going zero crossing of the power line cycle after the trigger.

TSP command: dmm.measure.linesync

MeasLineSync-Sts

Measure Line Synchronization Status

bool{

OFF
ON

}

Description: This PV shows if line synchronization is used during the measurement.

TSP command: dmm.measure.linesync

MeasStim-Sel

Measure Stimulus Selection

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<n>
$(1 \leq n \leq 8)$
EVENT_COMMAND
EVENT_DIGIO<n>
$(1 \leq n \leq 6)$
EVENT_TSPLINK<n>
$(1 \leq n \leq 3)$
EVENT_LAN<n>
$(1 \leq n \leq 8)$
EVENT_BLENDER<n>
$(1 \leq n \leq 2)$
EVENT_TIMER<n>
$(1 \leq n \leq 4)$
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

Description: This PV sets the instrument to make a measurement when it detects the specified trigger event. A measure function must be active before setting this PV. The measurement is made for the active measure function. If a digitize function is active, an error is generated. If the count is set to more than 1, the first reading is initialized by this trigger. Subsequent readings occur as rapidly as the instrument can make them. If a trigger occurs during the group measurement, the trigger is latched and another group of measurements with the same count will be triggered after the current group completes.

TSP command: dmm.trigger.measure.stimulus

MeasStim-Sts

Measure Stimulus Status

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<n>
($1 \leq n \leq 8$)
EVENT_COMMAND
EVENT_DIGIO<n>
($1 \leq n \leq 6$)
EVENT_TSPLINK<n>
($1 \leq n \leq 3$)
EVENT_LAN<n>
($1 \leq n \leq 8$)
EVENT_BLENDER<n>
($1 \leq n \leq 2$)
EVENT_TIMER<n>
($1 \leq n \leq 4$)
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

Description: This PV shows the instrument configured measurement trigger event for the selected measure function.

TSP command: dmm.trigger.measure.stimulus

MATrMode-Sel

Measure Analog Trigger Mode Selection

enum{

OFF
Edge
Pulse
Window

}

Description: This PV configures the type of signal behavior that can generate an analog trigger event. When edge is selected, the analog trigger occurs when the signal crosses a certain level. You also specify if the analog trigger occurs on the rising or falling edge of the signal. When pulse is selected, the analog trigger occurs when a pulse passes through the specified level and meets the constraint that you set on its width. You also specify the polarity of the signal (above or below the trigger level). When window is selected, the analog trigger occurs when the signal enters or exits the window defined by the low and high signal levels.

TSP command: dmm.digitize.analogtrigger.mode

MATrMode-Sts

Measure Analog Trigger Mode Status

enum{

OFF
Edge
Pulse
Window

}

Description: This PV shows the configured type of signal behavior that can generate an analog trigger event.

TSP command: dmm.digitize.analogtrigger.mode

MATrEdgeSlp-Sel

Measure Analog Trigger Edge Slope Selection

bool{

Rising
Falling

}

Description: This PV defines the slope of the analog trigger edge. This is only available when the analog trigger mode is set to edge. Rising causes an analog trigger event when the analog signal trends from below the analog signal level to above the level. Falling causes an analog trigger event when the signal trends from above to below the level.

TSP command: dmm.digitize.analogtrigger.edge.slope

MATrEdgeSlp-Sts

Measure Analog Trigger Edge Slope Status

bool{

Rising
Falling

}

Description: This PV shows the slope of the analog trigger edge.

TSP command: dmm.digitize.analogtrigger.edge.slope

MATrEdgeLvl-SP

Measure Analog Trigger Edge Level Set Point

float

Description: This PV defines the signal level that generates the analog trigger event for the edge trigger mode. This attribute is only available when the analog trigger mode is set to edge. The edge level can be set to any value in the active measurement range. See the Model DMM7510 specifications for more information on the resolution and accuracy of the analog trigger. To use the analog trigger with the measure functions, a range must be set (you cannot use autorange) and autozero must be disabled.

TSP command: dmm.digitize.analogtrigger.edge.level

MATrEdgeLvl-RB

Measure Analog Trigger Edge Level Read Back

float

Description: This PV shows the signal level that generates the analog trigger event for the edge trigger mode.

TSP command: dmm.digitize.analogtrigger.edge.level

MATrHFR-Sel

Measure Analog Trigger High Frequency Rejection Selection

bool{

OFF
ON

}

Description: This PV enables or disables high frequency rejection on analog trigger events. High frequency rejection avoids the false triggers by the requiring the trigger event to be sustained for at least 64 µs. This behavior is similar to a low pass filter effect with a 4 kHz 3 dB bandwidth.

TSP command: dmm.measure.analogtrigger.highfreqreject

MATrHFR-Sts

Measure Analog Trigger High Frequency Rejection Status

bool{

OFF
ON

}

Description: This PV shows if high frequency rejection on analog trigger events is enabled.

TSP command: dmm.measure.analogtrigger.highfreqreject

MATrPulCond-Sel

Measure Analog Trigger Pulse Condition Selection

bool{

Greater
Less

}

Description: This PV defines if the pulse must be greater than or less than the pulse width before an analog trigger is generated. Only available when the analog trigger mode is set to pulse.

TSP command: dmm.measure.analogtrigger.pulse.condition

MATrPulCond-Sts

Measure Analog Trigger Pulse Condition Status

bool{

Greater
Less

}

Description: This PV shows if the pulse must be greater than or less than the pulse width before an analog trigger is generated.

TSP command: dmm.measure.analogtrigger.pulse.condition

MATrPulPol-Sel

Measure Analog Trigger Pulse Polarity Selection

bool{

Above
Below

}

Description: This PV defines the polarity of the pulse that generates an analog trigger event. Only used when analog trigger mode is pulse. Determines if the analog trigger occurs when the pulse is above the defined signal level or below the defined signal level.

TSP command: dmm.measure.analogtrigger.pulse.polarity

MATrPulPol-Sts

Measure Analog Trigger Pulse Polarity Status

bool{

Above
Below

}

Description: This PV shows the polarity of the pulse that generates an analog trigger event.

TSP command: dmm.measure.analogtrigger.pulse.polarity

MATrPulLvl-SP

Measure Analog Trigger Pulse Level Set Point

float

Description: This PV defines the pulse level that generates an analog trigger event. Only available when the analog trigger mode is set to pulse.

TSP command: dmm.measure.analogtrigger.pulse.level

MATrPulLvl-RB

Measure Analog Trigger Pulse Level Read Back

float

Description: This PV shows the pulse level that generates an analog trigger event.

TSP command: dmm.measure.analogtrigger.pulse.level

MATrPulWidth-SP

Measure Analog Trigger Pulse Width Set Point

float

Min=0.000001

Max=0.04

unit: second

Description: This PV defines the threshold value for the pulse width. This option is only available when the analog trigger mode is set to pulse. This option sets either the minimum or maximum pulse width that generates an analog trigger event. The value of pulse condition determines whether this value is interpreted as the minimum or maximum pulse width.

TSP command: dmm.measure.analogtrigger.pulse.width

MATrPulWidth-RB

Measure Analog Trigger Pulse Width Read Back

float

unit: second

Description: This PV defines the threshold value for the pulse width.

TSP command: dmm.measure.analogtrigger.pulse.width

MATrWindHigh-SP

Measure Analog Trigger Window High Level Set Point

float

Description: This PV defines the upper boundary of the analog trigger window. Only available when the analog trigger mode is set to window. The high level must be greater than the low level.

TSP command: dmm.measure.analogtrigger.window.levelhigh

MATrWindHigh-RB

Measure Analog Trigger Window High Level Read Back

float

Description: This PV shows the upper boundary of the analog trigger window.

TSP command: dmm.measure.analogtrigger.window.levelhigh

MATrWindLow-SP

Measure Analog Trigger Window Low Level Set Point

float

Description: This PV defines the lower boundary of the analog trigger window. Only available when the analog trigger mode is set to window. The low level must be less than the high level.

TSP command: dmm.measure.analogtrigger.window.levellow

MATrWindLow-RB

Measure Analog Trigger Window Low Level Read Back

float

Description: This PV shows the lower boundary of the analog trigger window.

TSP command: dmm.measure.analogtrigger.window.levellow

MATrWindDir-Sel

Measure Analog Trigger Window Direction Selection

bool{

Enter
Leave

}

Description: This PV defines if the analog trigger occurs when the signal enters or leaves the defined upper and lower analog signal level boundaries. This is only available when the analog trigger mode is set to window.

TSP command: dmm.measure.analogtrigger.window.direction

MATrWindDir-Sts

Measure Analog Trigger Window Direction Status

bool{

Enter
Leave

}

Description: This PV shows if the analog trigger occurs when the signal enters or leaves the defined upper and lower analog signal level boundaries.

TSP command: dmm.measure.analogtrigger.window.direction

MRelOffEnbl-Sel

Measure Relative Offset Enable Selection

bool{

OFF
ON

}

Description: This PV enables or disables the application of a relative offset value to the measurement for the selected measure function.

TSP command: dmm.measure.rel.enable

MRelOffEnbl-Sts

Measure Relative Offset Enable Status

bool{

OFF
ON

}

Description: This PV enables or disables the application of a relative offset value to the measurement for the selected measure function. When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value. You can enter a relative offset value or have the instrument acquire a relative offset value. Each returned measured relative reading is the result of the following calculation:

$$\text{Displayed reading} = \textit{Actual measured reading} - \textit{Relative offset value}$$

TSP command: dmm.measure.rel.enable

MRelOffAcq-Cmd

Measure Relative Offset Acquire Command

bool{

OFF
ON

}

Description: When set to 1 or *ON*, this function acquires a measurement and stores it as the relative offset value. When the relative offset is acquired, the instrument does not apply any math, limit test, or filter settings to the measurement, even if they are set. You must change to the function for which you want to acquire a value before sending the command. The instrument must have relative offset enabled to use the acquired relative offset value.

TSP command: dmm.measure.rel.acquire()

MRelOff-SP

Measure Relative Offset Level Set Point

float

Description: This PV sets the relative offset value for the selected measure function. When relative offset is enabled, all subsequent measured readings are offset by the value that is set for this PV. You can set this value, or have the instrument acquire a value.

TSP command: dmm.measure.rel.level

MRelOff-RB

Measure Relative Offset Level Read Back

float

Description: This PV shows the relative offset value for the selected measure function.

TSP command: dmm.measure.rel.level

MMathEnbl-Sel

Measure Math Enable Selection

bool{

OFF
ON

}

Description: This PV enables or disables math operations on measurements for the selected measurement function. When this PV is set to *ON*, the math operation specified is performed before completing a measurement.

TSP command: dmm.measure.math.enable

MMathEnbl-Sts

Measure Math Enable Status

bool{

OFF
ON

}

Description: This PV shows if math operations on measurements for the selected measurement function are enabled.

TSP command: dmm.measure.math.enable

MMathOp-Sel

Measure Math Operation Selection

enum{

y=mx+b
Percent
Reciprocal

}

Description: This PV specifies which math operation is performed on measurements when math operations are enabled. You can choose one of the following math operations:

- y = mx+b: Manipulate normal display readings by adjusting the m and b factors.
- Percent: Displays measurements as the percentage of deviation from a specified reference constant.
- Reciprocal: The reciprocal math operation displays measurement values as reciprocals. The displayed value is 1/X, where X is the measurement value (if relative offset is being used, this is the measured value with relative offset applied).

Math calculations are applied to the input signal after relative offset.

TSP command: dmm.measure.math.format

MMathOp-Sts

Measure Math Operation Status

enum{

y=mx+b
Percent
Reciprocal

}

Description: This PV shows which math operation is performed on measurements when math operations are enabled.

TSP command: dmm.measure.math.format

MMathBFactor-SP

Measure Math B Factor Set Point

float

Min=-1000000000000

Max=1000000000000

Description: This PV specifies the offset, b, for the y = mx + b operation. The mx + b math operation lets you manipulate normal display readings (x) mathematically according to the following calculation:

$$y = mx + b$$

Where:

- y is the displayed result

  m is a user-defined constant for the scale factor

- x is the measurement reading (if you are using a relative offset, this is the measurement with relative offset applied)
- b is the user-defined constant the offset for factor.

TSP command: dmm.measure.math.mxb.bfactor

MMathBFactor-RB

Measure Math B Factor Read Back

float

Description: This PV shows the offset, b, for the y = mx + b operation.

TSP command: dmm.measure.math.mxb.bfactor

MMathMFactor-SP

Measure Math M Factor Set Point

float

Min=-1000000000000

Max=1000000000000

Description: This PV specifies the scale factor, m, for the y = mx + b math operation. The mx + b math operation lets you manipulate normal display readings (x) mathematically according to the following calculation:

$$y = mx + b$$

Where:

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

TSP command: dmm.measure.math.mxb.mfactor

MMathMFactor-RB

Measure Math M Factor Read Back

float

Description: This PV shows the scale factor, m, for the y = mx + b math operation.

TSP command: dmm.measure.math.mxb.mfactor

MMathPercRef-SP

Measure Math Percent Reference Set Point

float

Min=-1000000000000

Max=1000000000000

Description: This PV specifies the reference constant that is used when math operations are set to percent. The percent math function displays measurements as percent deviation from a specified reference constant. The percent calculation is:

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

Where:

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

TSP command: dmm.measure.math.percent

MMathPercRef-RB

Measure Math Percent Reference Read Back

float

Description: This PV shows the reference constant that is used when math operations are set to percent.

TSP command: dmm.measure.math.percent

FilterEnbl-Sel

Filter Enable Selection

bool{

OFF
ON

}

Description: This PV enables or disables the averaging filter for measurements of the selected function. When this is enabled, the reading returned by the instrument is an averaged value, taken from multiple measurements. The settings of the filter count and filter type for the selected measure function determines how the reading is averaged.

TSP command: dmm.measure.filter.enable

FilterEnbl-Sts

Filter Enable Status

bool{

OFF
ON

}

Description: This PV shows if the averaging filter for measurements of the selected function is enabled.

TSP command: dmm.measure.filter.enable

FilterCount-SP

Filter Count Set Point

long

Min=1

Max=100

Description: This PV sets the number of measurements that are averaged when filtering is enabled. The filter count is the number of readings that are acquired and stored in the filter stack for the averaging calculation. When the filter count is larger, more filtering is done and the data is less noisy.

TSP command: dmm.measure.filter.count

FilterCount-RB

Filter Count Read Back

long

Description: This PV shows the number of measurements that are averaged when filtering is enabled.

TSP command: dmm.measure.filter.count

FilterTyp-Sel

Filter Type Selection

bool{

Repeat
Moving

}

Description: This PV defines the type of averaging filter that is used for the selected function when the filter is enabled. When the repeating average filter is selected, a set of measurements are made. These measurements are stored in a measurement stack and averaged together to produce the averaged sample. Once the averaged sample is produced, the stack is flushed and the next set of data is used to produce the next averaged sample. When the moving average filter is selected, the measurements are added to the stack continuously on a first-in, first-out basis. As each measurement is made, the oldest measurement is removed from the stack. A new averaged sample is produced using the new measurement and the data that is now in the stack.

TSP command: dmm.measure.filter.type

FilterTyp-Sts

Filter Type Status

bool{

Repeat
Moving

}

Description: This PV shows the type of averaging filter that is used for the selected function when the filter is enabled.

TSP command: dmm.measure.filter.type

FilterWind-SP

Filter Window Set Point

long

Min=0

Max=10

Description: This PV sets the window for the averaging filter that is used for measurements for the selected function. The noise window allows a faster response time to large signal step changes. A reading that falls outside the plus or minus noise window fills the filter stack immediately. If the noise does not exceed the selected percentage of range, the reading is based on an average of reading conversions the normal averaging filter. If the noise does exceed the selected percentage, the reading is a single reading conversion, and new averaging starts from this point.

TSP command: dmm.measure.filter.window

FilterWind-RB

Filter Window Read Back

long

Description: This PV shows the window for the averaging filter that is used for measurements for the selected function.

TSP command: dmm.measure.filter.window

4.3 Digitize settings

DigtzeApert-SP

Digitize Aperture Set Point

long

Min=1

Max=1000

unit: µs

Description: This PV determines the aperture setting for the selected function. The aperture is the actual acquisition time of the instrument on the signal. It must be less than the set sample rate. The minimum aperture is 1 s when the maximum sampling rate is 1,000,000 samples per second. If you set a value that is longer than the sample rate interval, the instrument generates an error event. Set the sample rate before changing the aperture.

TSP command: dmm.digitize.aperture

DigtzeApert-RB

Digitize Aperture Read Back

long

unit: µs

Description: This PV shows the aperture setting for the selected function.

TSP command: dmm.digitize.aperture

DigtzeApertAuto-Cmd

Digitize Auto Aperture Command

bool{

OFF
ON

}

Description: When set to 1 or *ON*, this PV sets the aperture setting to *AUTO* for the selected function. When the aperture is set to automatic, the aperture is equivalent to the sample rate interval, which is the maximum value possible for the selected sample rate.

TSP command: dmm.digitize.aperture

DigtzeCount-SP

Digitize Count Set Point

long

Min=1

Max=55000000

Description: This PV sets the number of measurements to digitize when a measurement is requested.

TSP command: dmm.digitize.count

DigtzeCount-RB

Digitize Count Read Back

long

Description: This PV shows the number of measurements to digitize when a measurement is requested.

TSP command: dmm.digitize.count

DigtzeSR-SP

Digitize Sample Rate Set Point

long

Min=1000

Max=1000000

Description: This PV defines the precise acquisition rate at which the digitizing measurements are made. Set the sample rate before setting the aperture. If the aperture setting is too high for the selected sample rate, it is automatically adjusted to the highest aperture that can be used with the sample rate.

TSP command: dmm.digitize.samplerate

DigtzeSR-RB

Digitize Sample Rate Read Back

long

Description: This PV shows the precise acquisition rate at which the digitizing measurements are made.

TSP command: dmm.digitize.samplerate

DRange-SP

Digitize Range Set Point

long

Min=0

Description: This PV determines the positive full-scale measure range for digitizer.

TSP command: dmm.digitize.range

DRange-RB

Digitize Range Read Back

long

Description: This PV shows the positive full-scale measure range for digitizer.

TSP command: dmm.digitize.range

DigtzeCoup-Sel

Digitize Coupling Selection

bool{

AC
DC

}

Description: This PV determines if AC or DC signal coupling is used. When DC is selected, the instrument measures AC and DC components of the signal. When AC is selected, the instrument only measures the AC components of the signal.

TSP command: dmm.digitize.coupling.type

DigtzeCoup-Sts

Digitize Coupling Status

bool{

AC
DC

}

Description: This PV shows if AC or DC signal coupling is used.

TSP command: dmm.digitize.coupling.type

DigtzeACFilter-Sel

Digitize Coupling AC Filter Selection

bool{

Slow
Fast

}

Description: This PV selects the instrument settling time when coupling is set to AC. This option is only used when digitize signal coupling is set to AC. When the signal coupling is set to AC, there may still be some DC signal content that comes in with the AC signal. To allow this signal to settle out, you can set AC coupling filter to slow. When the filter is set to slow, the instrument adds an 800 ms delay before making measurements. When the AC coupling filter is set to fast, the instrument adds an 80 ms delay before making measurements.

TSP command: dmm.digitize.coupling.acfilter

DigtzeACFilter-Sts

Digitize Coupling AC Filter Status

bool{

Slow
Fast

}

Description: This PV shows the instrument settling time when coupling is set to AC.

TSP command: dmm.digitize.coupling.acfilter

DigtzeACFreq-SP

Digitize Coupling AC Frequency Set Point

float

Min=3

Max=1000000

unit: Hz

Description: This PV allows you to optimize the amplitude to compensate for signal loss across the coupling capacitor when AC coupling is selected. This attribute is only used when the digitize coupling type is set to AC. For example, if you are measuring a 50 Hz signal, you could set this to 50 Hz to compensate for voltage drop across the coupling capacitor.

TSP command: dmm.digitize.coupling.acfrequency

DigtzeACFreq-RB

Digitize Coupling AC Frequency Read Back

float

unit: Hz

Description: This PV shows the AC frequency setting used to optimize the amplitude to compensate for signal loss across the coupling capacitor when AC coupling is selected.

TSP command: dmm.digitize.coupling.acfrequency

DigtzeImpedance-Sel

Digitize Input Impedance Selection

bool{

AUTO
10MOhm

}

Description: This PV determines when the 10 MΩ input divider is enabled. Choosing automatic input impedance is a balance between achieving low DC voltage noise on the 100 mV and 1 V ranges and optimizing measurement noise due to charge injection. The Model DMM7510 is optimized for low noise and charge injection when the DUT has less than 100 kΩ input resistance. When the DUT input impedance is more than 100 kΩ, selecting an input impedance of 10 MΩ optimizes the measurement for lowest noise on the 100 mV and 1 V ranges. For the 10 V to 1000 V ranges, both input impedance settings achieve low charge injection. The input impedance setting is only available when coupling is set to DC.

TSP command: dmm.digitize.inputimpedance

DigtzeImpedance-Sts

Digitize Input Impedance Status

bool{

AUTO
10MOhm

}

Description: This PV shows when the 10 MΩ input divider is enabled.

TSP command: dmm.digitize.inputimpedance

DigtzeStim-Sel

Digitize Stimulus Selection

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<n>
($1 \leq n \leq 8$)
EVENT_COMMAND
EVENT_DIGIO<n>
($1 \leq n \leq 6$)
EVENT_TSPLINK<n>
($1 \leq n \leq 3$)
EVENT_LAN<n>
($1 \leq n \leq 8$)
EVENT_BLENDER<n>
($1 \leq n \leq 2$)
EVENT_TIMER<n>
($1 \leq n \leq 4$)
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

Description: This PV sets the instrument to digitize a measurement when it detects the specified trigger event. A digitize function must be active before setting this PV. If a measure function is active, an error is generated. If the count is set to more than 1, the first reading is initialized by this trigger. Subsequent readings occur as rapidly as the instrument can make them. If a trigger occurs during the group measurement, the trigger is latched and another group of measurements with the same count will be triggered after the current group completes.

TSP command: dmm.trigger.digitize.stimulus

DigtzeStim-Sts

Digitize Stimulus Status

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<n>
$(1 \leq n \leq 8)$
EVENT_COMMAND
EVENT_DIGIO<n>
$(1 \leq n \leq 6)$
EVENT_TSPLINK<n>
$(1 \leq n \leq 3)$
EVENT_LAN<n>
$(1 \leq n \leq 8)$
EVENT_BLENDER<n>
$(1 \leq n \leq 2)$
EVENT_TIMER<n>
$(1 \leq n \leq 4)$
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

Description: This PV shows the specified trigger event that causes a measurement digitize.

TSP command: dmm.trigger.digitize.stimulus

DATrMode-Sel

Digitize Analog Trigger Mode Selection

enum{

OFF
Edge
Pulse
Window

}

Description: This PV configures the type of signal behavior that can generate an analog trigger event. When edge is selected, the analog trigger occurs when the signal crosses a certain level. You also specify if the analog trigger occurs on the rising or falling edge of the signal. When pulse is selected, the analog trigger occurs when a pulse passes through the specified level and meets the constraint that you set on its width. You also specify the polarity of the signal (above or below the trigger level). When window is selected, the analog trigger occurs when the signal enters or exits the window defined by the low and high signal levels.

TSP command: dmm.digitize.analogtrigger.mode

DATrMode-Sts

Digitize Analog Trigger Mode Status

enum{

OFF
Edge
Pulse
Window

}

Description: This PV shows the type of signal behavior that can generate an analog trigger event.

TSP command: dmm.digitize.analogtrigger.mode

DATrEdgeSlp-Sel

Digitize Analog Trigger Edge Slope Selection

bool{

Rising
Falling

}

Description: This PV defines the slope of the analog trigger edge. This is only available when the analog trigger mode is set to edge. Rising causes an analog trigger event when the analog signal trends from below the analog signal level to above the level. Falling causes an analog trigger event when the signal trends from above to below the level.

TSP command: dmm.digitize.analogtrigger.edge.slope

DATrEdgeSlp-Sts

Digitize Analog Trigger Edge Slope Status

bool{

Rising
Falling

}

Description: This PV shows the slope of the analog trigger edge.

TSP command: dmm.digitize.analogtrigger.edge.slope

DATrEdgeLvl-SP

Digitize Analog Trigger Edge Level Set Point

float

Description: This PV defines the signal level that generates the analog trigger event for the edge trigger mode. This attribute is only available when the analog trigger mode is set to edge. The edge level can be set to any value in the active measurement range. See the Model DMM7510 specifications for more information on the resolution and accuracy of the analog trigger.

TSP command: dmm.digitize.analogtrigger.edge.level

DATrEdgeLvl-RB

Digitize Analog Trigger Edge Level Read Back

float

Description: This PV shows the signal level that generates the analog trigger event for the edge trigger mode.

TSP command: dmm.digitize.analogtrigger.edge.level

DATrHFR-Sel

Digitize Analog Trigger High Frequency Rejection Selection

bool{

OFF
ON

}

Description: This PV enables or disables high frequency rejection on analog trigger events. High frequency rejection avoids the false triggers by the requiring the trigger event to be sustained for at least 64 µs. This behavior is similar to a low pass filter effect with a 4 kHz 3 dB bandwidth.

TSP command: dmm.digitize.analogtrigger.highfreqreject

DATrHFR-Sts

Digitize Analog Trigger High Frequency Rejection Status

bool{

OFF
ON

}

Description: This PV shows if high frequency rejection on analog trigger events is enabled.

TSP command: dmm.digitize.analogtrigger.highfreqreject

DATrPulCond-Sel

Digitize Analog Trigger Pulse Condition Selection

bool{

Greater
Less

}

Description: This PV defines if the pulse must be greater than or less than the pulse width before an analog trigger is generated. Only available when the analog trigger mode is set to pulse.

TSP command: dmm.digitize.analogtrigger.pulse.condition

DATrPulCond-Sts

Digitize Analog Trigger Pulse Condition Status

bool{

Greater
Less

}

Description: This PV shows if the pulse must be greater than or less than the pulse width before an analog trigger is generated.

TSP command: dmm.digitize.analogtrigger.pulse.condition

DATrPulPol-Sel

Digitize Analog Trigger Pulse Polarity Selection

bool{

Above
Below

}

Description: This PV defines the polarity of the pulse that generates an analog trigger event. Only used when analog trigger mode is pulse. Determines if the analog trigger occurs when the pulse is above the defined signal level or below the defined signal level.

TSP command: dmm.digitize.analogtrigger.pulse.polarity

DATrPulPol-Sts

Digitize Analog Trigger Pulse Polarity Status

bool{

Above
Below

}

Description: This PV shows the polarity of the pulse that generates an analog trigger event.

TSP command: dmm.digitize.analogtrigger.pulse.polarity

DATrPulLvl-SP

Digitize Analog Trigger Pulse Level Set Point

float

Description: This PV defines the pulse level that generates an analog trigger event. Only available when the analog trigger mode is set to pulse.

TSP command: dmm.digitize.analogtrigger.pulse.level

DATrPulLvl-RB

Digitize Analog Trigger Pulse Level Read Back

float

Description: This PV shows the pulse level that generates an analog trigger event.

TSP command: dmm.digitize.analogtrigger.pulse.level

DATrPulWidth-SP

Digitize Analog Trigger Pulse Width Set Point

float

Min=0.000001

Max=0.04

unit: second

Description: This PV defines the threshold value for the pulse width. This option is only available when the analog trigger mode is set to pulse. This option sets either the minimum or maximum pulse width that generates an analog trigger event. The value of pulse condition determines whether this value is interpreted as the minimum or maximum pulse width.

TSP command: dmm.digitize.analogtrigger.pulse.width

DATrPulWidth-RB

Digitize Analog Trigger Pulse Width Read Back

float

unit: second

Description: This PV shows the threshold value for the pulse width.

TSP command: dmm.digitize.analogtrigger.pulse.width

DATrWindHigh-SP

Digitize Analog Trigger Window High Level Set Point

float

Description: This PV defines the upper boundary of the analog trigger window. Only available when the analog trigger mode is set to window. The high level must be greater than the low level.

TSP command: dmm.digitize.analogtrigger.window.levelhigh

DATrWindHigh-RB

Digitize Analog Trigger Window High Level Read Back

float

Description: This PV shows the upper boundary of the analog trigger window.

TSP command: dmm.digitize.analogtrigger.window.levelhigh

DATrWindLow-SP

Digitize Analog Trigger Window Low Level Set Point

float

Description: This PV defines the lower boundary of the analog trigger window. Only available when the analog trigger mode is set to window. The low level must be less than the high level.

TSP command: dmm.digitize.analogtrigger.window.levellow

DATrWindLow-RB

Digitize Analog Trigger Window Low Level Read Back

float

Description: This PV shows the lower boundary of the analog trigger window.

TSP command: dmm.digitize.analogtrigger.window.levellow

DATrWindDir-Sel

Digitize Analog Trigger Window Direction Selection

bool{

Enter
Leave

}

Description: This PV defines if the analog trigger occurs when the signal enters or leaves the defined upper and lower analog signal level boundaries. This is only available when the analog trigger mode is set to window.

TSP command: dmm.digitize.analogtrigger.window.direction

DATrWindDir-Sts

Digitize Analog Trigger Window Direction Status

bool{

Enter
Leave

}

Description: This PV shows if the analog trigger occurs when the signal enters or leaves the defined upper and lower analog signal level boundaries.

TSP command: dmm.digitize.analogtrigger.window.direction

DRelOffEnbl-Sel

Digitize Relative Offset Enable Selection

bool{

OFF
ON

}

Description: This PV enables or disables the application of a relative offset value to the measurement. When relative measurements are enabled, all subsequent measured readings are offset by the relative offset value that was calculated when you acquired the relative offset value. Each returned measured relative reading is the result of the following calculation:

$$\text{Displayed reading} = \text{Actual measured reading} - \text{Relative offset value}$$

TSP command: dmm.digitize.rel.enable

DRelOffEnbl-Sts

Digitize Relative Offset Enable Status

bool{

OFF
ON

}

Description: This PV shows if the application of a relative offset value to the measurement is enabled.

TSP command: dmm.digitize.rel.enable

DRelOffAcq-Cmd

Digitize Relative Offset Acquire Command

bool{

OFF
ON

}

Description: When set to 1 or *ON*, this function acquires a measurement and stores it as the relative offset value. When the relative offset is acquired, the instrument does not apply any math, limit test, or filter settings to the measurement, even if they are set. You must change to the function for which you want to acquire a value before sending this command. The instrument must have relative offset enabled to use the acquired relative offset value.

TSP command: dmm.digitize.rel.acquire()

DRelOff-SP

Digitize Relative Offset Level Set Point

float

Description: This PV sets the relative offset value. When relative offset is enabled, all subsequent measured readings are offset by the value that is set for this PV. You can set this value, or have the instrument acquire a value.

TSP command: dmm.digitize.rel.level

DRelOff-RB

Digitize Relative Offset Level Read Back

float

Description: This PV shows the relative offset value.

TSP command: dmm.digitize.rel.level

DMathEnbl-Sel

Digitize Math Enable Selection

bool{

OFF
ON

}

Description: This PV enables or disables math operations on measurements for the selected digitize function. When this PV is set to *ON*, the math operation specified is performed before completing a measurement.

TSP command: dmm.digitize.math.enable

DMathEnbl-Sts

Digitize Math Enable Status

bool{

OFF
ON

}

Description: This PV enables or disables math operations on measurements for the selected digitize function.

TSP command: dmm.digitize.math.enable

DMathOp-Sel

Digitize Math Operation Selection

enum{

y=mx+b
Percent
Reciprocal

}

Description: This PV specifies which math operation is performed for the selected measurement function when math operations are enabled. You can choose one of the following math operations:

- y = mx+b: Manipulate normal display readings by adjusting the m and b factors.
- Percent: Displays measurements as the percentage of deviation from a specified reference constant.
- Reciprocal: The reciprocal math operation displays measurement values as reciprocals. The displayed value is 1/X, where X is the measurement value (if relative offset is being used, this is the measured value with relative offset applied).

Math calculations are applied to the input signal after relative offset.

TSP command: dmm.digitize.math.format

DMathOp-Sts

Digitize Math Operation Status

bool{

y=mx+b
Percent
Reciprocal

}

Description: This PV shows which math operation is performed for the selected measurement function when math operations are enabled.

TSP command: dmm.digitize.math.format

DMathBFactor-SP

Digitize Math B Factor Set Point

float

Min=-1000000000000

Max=1000000000000

Description: This PV specifies the offset, b, for the y = mx + b operation. The mx + b math operation lets you manipulate normal display readings (x) mathematically according to the following calculation:

$$y = mx + b$$

Where:

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

TSP command: dmm.digitize.math.mxb.bfactor

DMathBFactor-RB

Digitize Math B Factor Read Back

float

Description: This PV shows the offset, b, for the y = mx + b operation.

TSP command: dmm.digitize.math.mxb.bfactor

DMathMFactor-SP

Digitize Math M Factor Set Point

float

Min=-1000000000000

Max=1000000000000

Description: This PV specifies the scale factor, m, for the y = mx + b math operation. The mx + b math operation lets you manipulate normal display readings (x) mathematically according to the following calculation:

$$y = mx + b$$

Where:

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

TSP command: dmm.digitize.math.mxb.mfactor

DMathMFactor-RB

Digitize Math M Factor Read Back

float

Description: This PV shows the scale factor, m, for the y = mx + b math operation.

TSP command: dmm.digitize.math.mxb.mfactor

DMathPercRef-SP

Digitize Math Percent Reference Set Point

float

Min=-1000000000000

Max=1000000000000

Description: This PV specifies the reference constant that is used when math operations are set to percent. The percent math function displays measurements as percent deviation from a specified reference constant. The percent calculation is:

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

Where:

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

TSP command: dmm.digitize.math.percent

DMathPercRef-RB

Digitize Math Percent Reference Read Back

float

Description: This PV shows the reference constant that is used when math operations are set to percent.

TSP command: dmm.digitize.math.percent

4.4 Buffer operations

StartRead<n>-SP

Buffer<n>Start Reading Position Set Point

long

Min=1

n: number of default buffer (1 or 2)

Description: This PV specifies the start index to start reading the buffer when a read buffer command is issued.

TSP command: No command

EndRead<n>-SP

Buffer<n>End Reading Position Set Point

long

Min=1

n: number of default buffer (1 or 2)

Description: This PV specifies the last buffer index to read from when a read buffer command is issued.

TSP command: No command

ReadBuff<n>-Cmd

Buffer<n>Read Command

bool{

OFF
ON

}

n: number of default buffer (1 or 2)

Description: When set to 1 or emphON, this PV causes ReadBuff<n>-Mon to process once to read the buffer section specified by *StartRead<n>-SP* and *EndRead<n>-SP*.

TSP command: No command

ReadBuff<n>-Mon

Buffer<n>Readings Monitor

double[1000]

n: number of default buffer (1 or 2)

Description: This record reads an array of readings from the corresponding buffer when processed. Set *ReadBuff<n>-Mon.PROC* to any value to retrieve buffer readings once. Set *ReadBuff<n>-Mon.SCAN* to a valide EPICS SCAN value in order to get data from the buffer periodically.

TSP command: printbuffer(StartRead<n>, EndRead<n>, defbuffer<n>.readings)

FetchBuff<n>-Mon

Buffer<n>Fetch Reading Monitor

double

n: number of default buffer (1 or 2)

Description: This record reads the latest measurement from the corresponding buffer when processed. Set *FetchBuff<n>-Mon.PROC* to any value in order to fetch a measurement once. Set *FetchBuff<n>-Mon.SCAN* to a valide EPICS SCAN value in order to fetch measurements periodically.

TSP command: printbuffer(defbuffer<n>.endindex, defbuffer<n>.endindex, defbuffer<n>.readings)

MeasBuff<n>-Cmd

Measure and Store in Buffer<n>Command

bool{

OFF
ON

}

n: number of default buffer (1 or 2)

Description: When set to 1 or *ON*, this PV causes a measurement to be taken (a measure function must be selected) and stored in the *FetchBuff<n>-Mon* PV.

TSP command: print(dmm.measure.read(defbuffer<n>))

DigtzBuff<n>-Cmd

Digitize and Store in Buffer<n>Command

bool{

OFF
ON

}

n: number of default buffer (1 or 2)

Description: When set to 1 or *ON*, this PV causes a digitize measurement to be taken (a digitize function must be selected) and stored in the *FetchBuff<n>-Mon* PV.

TSP command: print(dmm.digitize.read(defbuffer<n>))

StartBuff<n>-Mon

Buffer<n>Start Index Monitor

long

n: number of default buffer (1 or 2)

Description: This PV contains the corresponding buffer start index. Set *StartBuff<n>-Mon.PROC* to any value in order to read the buffer start index once. Set *StartBuff<n>-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the buffer start index, periodically.

TSP command: print(defbuffer<n>.startindex)

EndBuff<n>-Mon

Buffer<n>End Index Monitor

long

n: number of default buffer (1 or 2)

Description: This PV contains the corresponding buffer end index. Set *EndBuff<n>-Mon.PROC* to any value in order to read the buffer end index once. Set *EndBuff<n>-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the buffer end index, periodically.

TSP command: print(defbuffer<n>.endindex)

SizeBuff<n>-SP

Buffer<n>Size Set Point

long

Min=0

n: number of default buffer (1 or 2)

Description: This PV specifies the number of readings the buffer can store.

TSP command: defbuffer<n>.capacity

SizeBuff<n>-RB

Buffer<n>Size Read Back

long

n: number of default buffer (1 or 2)

Description: This PV shows the number of readings the buffer can store.

TSP command: defbuffer<n>.capacity

FillModeBuff<n>-Sel

Buffer<n>Fill Mode Selection

bool{

Once
Continuous

}

n: number of default buffer (1 or 2)

Description: This PV determines if a reading buffer is filled continuously or is filled once and stops.

TSP command: defbuffer<n>.fillmode

FillModeBuff<n>-Sts

Buffer<n>Fill Mode Status

bool{

Once
Continuous

}

n: number of default buffer (1 or 2)

Description: This PV shows if a reading buffer is filled continuously or is filled once and stops.

TSP command: defbuffer<n>.fillmode

CntBuff<n>-Mon

Buffer<n>Count Monitor

long

n: number of default buffer (1 or 2)

Description: This record gets the number of readings in the specified reading buffer when processed. Set *CntBuff<n>-Mon.PROC* to any value in order to read the buffer count once. Set *CntBuff<n>-Mon.SCAN* to a valide EPICS SCAN value in order to periodically monitor the buffer count.

TSP command: defbuffer<n>.n

ClrBuff<n>-Cmd

Clear Buffer<n>Command

bool{

OFF
ON

}

n: number of default buffer (1 or 2)

Description: When set to 1 or *ON*, this PV clears all readings and statistics from the buffer.

TSP command: defbuffer<n>.clear()

AvgBuff<n>-Mon

Buffer<n>Average Value Monitor

float

n: number of default buffer (1 or 2)

Description: This record gets the average of the corresponding buffer readings when processed. Set *AvgBuff<n>-Mon.PROC* to any value in order to read the buffer average value once. Set *AvgBuff<n>-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the buffer average periodically.

TSP command: statsVar = buffer.getstats(defbuffer<n>-Mon); print(statsVar.mean);

MaxBuff<n>-Mon

Buffer<n>Maximum Value Monitor

float

n: number of default buffer (1 or 2)

Description: This record gets the maximum value reading from the corresponding buffer when processed. Set *MaxBuff<n>-Mon.PROC* to any value in order to read the buffer maximum value once. Set *MaxBuff<n>-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the buffer maximum value periodically.

TSP command: statsVar = buffer.getstats(defbuffer<n>); print(statsVar.max.reading;

MinBuff<n>-Mon

Buffer<n>Minimum Value Monitor

float

n: number of default buffer (1 or 2)

Description: This record gets the minimum value reading from the corresponding buffer when processed. Set *MinBuff<n>-Mon.PROC* to any value in order to read the buffer maximum value once. Set *MinBuff<n>-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the buffer minimum value periodically.

TSP command: statsVar = buffer.getstats(defbuffer<n>); print(statsVar.min.reading);

StdDBuff<n>-Mon

Buffer<n>Standard Deviation Monitor

float

n: number of default buffer (1 or 2)

Description: This record gets the standard deviation of readings for the corresponding buffer when processed. Set *StdDBuff<n>-Mon.PROC* to any value in order to read the buffer standard devitation once. Set *StdDBuff<n>-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the buffer standard deviation periodically.

TSP command: statsVar = buffer.getstats(defbuffer<n>); print(statsVar.stddev);

ClrStatBuff<n>-Cmd

Clear Statistics of Buffer<n>Command

bool{

OFF
ON

}

n: number of default buffer (1 or 2)

Description: When set to 1 or *ON*, this PV clears the statistical information associated with the specified buffer.

TSP command: buffer.clearstats(defbuffer<n>)

4.5 External I/O

ExInEdge-Sel

External Input Edge Selection

enum{

FALLING
RISING
EITHER

}

Description: This PV sets the type of edge that is detected as an input on the external in line.

TSP command: trigger.extin.edge

ExInEdge-Sts

External Input Edge Status

bool{

FALLING
RISING
EITHER

}

Description: This PV shows the type of edge that is detected as an input on the external in line.

TSP command: trigger.extin.edge

ExInOver-Mon

External Input Overrun Monitor

bool{

No overrun
Overrun

}

Description: This PV shows the event detector overrun status.

TSP command: trigger.extin.overrun

ClearExInEv-Cmd

Clear External Input Event Command

bool{

OFF
ON

}

Description: When set to 1 or *ON*, this PV clears the trigger event on the external in line.

TSP command: trigger.extin.clear()

ExOutPol-Sel

External Output Polarity Selection

bool{

Positive
Negative

}

Description: This PV sets the output logic of the trigger event generator to positive or negative for the external out line.

TSP command: trigger.extout.logic

ExOutPol-Sts

External Output Polarity Status

bool{

Positive
Negative

}

Description: This PV shows the output logic of the trigger event generator to positive or negative for the external out line.

TSP command: trigger.extout.logic

ExOutStim-Sel

External Output Stimulus Selection

bool{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<n>
$(1 \leq n \leq 8)$
EVENT_COMMAND
EVENT_DIGIO<n>
$(1 \leq n \leq 6)$
EVENT_TSPLINK<n>
$(1 \leq n \leq 3)$
EVENT_LAN<n>
$(1 \leq n \leq 8)$
EVENT_BLENDER<n>
$(1 \leq n \leq 2)$
EVENT_TIMER<n>
$(1 \leq n \leq 4)$
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

Description: This PV selects the event that causes a trigger to be asserted on the external output line.

TSP command: trigger.extout.stimulus

ExOutStim-Sts

External Output Stimulus Status

bool{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<n>
($1 \leq n \leq 8$)
EVENT_COMMAND
EVENT_DIGIO<n>
($1 \leq n \leq 6$)
EVENT_TSPLINK<n>
($1 \leq n \leq 3$)
EVENT_LAN<n>
($1 \leq n \leq 8$)
EVENT_BLENDER<n>
($1 \leq n \leq 2$)
EVENT_TIMER<n>
($1 \leq n \leq 4$)
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

Description: This PV shows the event that causes a trigger to be asserted on the external output line.

TSP command: trigger.extout.stimulus

4.6 Digital I/O

DigWrite-SP

Digital Port Write Set Point

long

Min=0

Max=63

Description: This PV writes to all digital I/O lines. The binary representation of the value indicates the output pattern to be written to the I/O port. For example, a value of 63 has a binary equivalent of 111111 (all lines are set high), and a data value of 42 has a binary equivalent of 101010 (lines 2, 4, and 6 are set high, and the other 3 lines are set low). An instrument reset does not affect the present states of the digital I/O lines. All six lines must be configured as digital control lines. If not, this command generates an error.

TSP command: digio.writeport()

DigRead-Mon

Digital Port Read Monitor

long

Description: This record reads the digital I/O port (all lines) state when processed. The least significant bit (bit B1) of the binary number corresponds to digital I/O line 1; bit B6 corresponds to digital I/O line 6. For example, a returned value of 42 has a binary equivalent of 101010, which indicates that lines 2, 4, 6 are high (1), and the other lines are low (0). Set *DigRead-Mon.PROC* to any value in order to read the I/O port state once. Set *DigRead-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the port value periodically.

TSP command: digio.readport()

Dig<n>Mod-Sel

Digital Line <n>Mode Selection

enum{

DIGIN
DIGOUT
DIGOPEN
TRIGIN
TRIGOUT
TRIGOPEN
SYNCMASTER
SYNCACC

}

n: digital line number (1 to 6)

Description: This PV sets the mode of the digital I/O line to be a digital line, trigger line, or synchronous line and sets the line to be input, output, or open-drain. The following settings of line mode set the line for direct control as a digital line:

- DIGIN (Digital Input): The instrument automatically detects externally generated logic levels. You can read an input line, but you cannot write to it.
- DIGOUT (Digital Output): You can set the line as logic high (+5 V) or as logic low (0 V). The default level is logic low (0 V). When the instrument is in output mode, the line is actively driven high or low.
- DIGOPEN (Digital Open Drain): Configures the line to be an open-drain signal. The line can serve as an input, an output or both. When a digital I/O line is used as an input in open-drain mode, you must write a 1 to it.

The following settings of line mode set the line as a trigger line:

- TRIGIN(Trigger Input): The line automatically responds to and detects externally generated triggers. It detects falling-edge, rising-edge, or either-edge triggers as input depending on the configuration of the *Dig<n>Edge-Sel* PV.
- TRIGOUT (Trigger Output): The line is automatically set high or low depending on the output logic setting. Use the negative logic setting when you want to generate a falling edge trigger and use the positive logic setting when you want to generate a rising edge trigger.
- TRIGOPEN (Trigger Open Drain): Configures the line to be an open-drain signal. You can use the line to detect input triggers or generate output triggers. This line state uses the edge setting specified by the *Dig<n>Edge-Sel* PV.

When the line is set as a synchronous acceptor (SYNCACC), the line detects the falling-edge input triggers and automatically latches and drives the trigger line low. Asserting an output trigger releases the latched line. When the line is set as a synchronous master (SYNCMASTER), the line detects rising-edge triggers as input. For output, the line asserts a TTL-low pulse.

TSP command: digio.line[n].mode

Dig<n>Mod-Sts

Digital Line <n>Mode Status

enum{

DIGIN
DIGOUT
DIGOPEN
TRIGIN
TRIGOUT
TRIGOPEN
SYNCMASTER
SYNCACC

}

n: digital line number (1 to 6)

Description: This PV shows the mode of the digital I/O line: digital line, trigger line, or synchronous line; and I/O configuration: input, output, or open-drain.

TSP command: digio.line[n].mode

Dig<n>State-Sel

Digital Line <n>State Selection

bool{

LOW
HIGH

}

n: digital line number (1 to 6)

Description: This PV sets the corresponding digital I/O line high or low when the line is set for digital control.

TSP command: digio.line[n].state

Dig<n>State-Mon

Digital Line <n>State Monitor

bool{

LOW
HIGH

}

n: digital line number (1 to 6)

Description: When processed, this record reads the state of the corresponding digital I/O line. Set *Dig<n>State-Mon.PROC* to any value in order to read the I/O line state once. Set *Dig<n>State-Mon.SCAN* to some valide EPICS SCAN value in order to monitor the I/O line state periodically. When a reset occurs, the digital line state can be read as high because the digital line is reset to a digital input. A digital input floats high if nothing is connected to the digital line.

TSP command: digio.line[n].state

Dig<n>ClrEv-Cmd

Digital Line <n>Clear Event Command

bool{

OFF
ON

}

n: digital line number (1 to 6)

Description: When set to 1 or *ON*, this PV clears the trigger event on the corresponding digital input line.

TSP command: trigger.digin[n].clear()

Dig<n>Edge-Sel

Digital Line <n>Edge Selection

enum{

Falling
Rising
Either

}

n: digital line number (1 to 6)

Description: This PV sets the edge used by the trigger event detector on the given trigger line.

TSP command: trigger.digin[n].edge

Dig<n>Edge-Sts

Digital Line <n>Edge Status

enum{

Falling
Rising
Either

}

n: digital line number (1 to 6)

Description: This PV shows the edge used by the trigger event detector on the given trigger line.

TSP command: trigger.digin[n].edge

Dig<n>Over-Mon

Digital Line <n>Overrun Monitor

bool{

No overrun
Overrun

}

n: digital line number (1 to 6)

Description: This PV shows the event detector overrun status.

TSP command: trigger.digin[n].overrun

Dig<n>Pol-Sel

Digital Line <n>Polarity Selection

bool{

Positive
Negative

}

n: digital line number (1 to 6)

Description: This PV sets the output logic of the trigger event generator to positive or negative for the corresponding line.

TSP command: trigger.digout[n].logic

Dig<n>Pol-Sts

Digital Line <n>Polarity Status

bool{

Positive
Negative

}

n: digital line number (1 to 6)

Description: This PV shows the output logic of the trigger event generator for the corresponding line.

TSP command: trigger.digout[n].logic

Dig<n>Width-SP

Digital Line <n>Width Set Point

float

Min=0

Max=100000

n: digital line number (1 to 6)

Description: This PV sets the length of time that the trigger line is asserted for output triggers. Setting the pulse width to zero (0) seconds asserts the trigger indefinitely. To release the trigger line, use *Dig<n>Release-Cmd*.

TSP command: trigger.digout[n].pulsewidth

Dig<n>Width-RB

Digital Line <n>Width Read Back

float

n: digital line number (1 to 6)

Description: This PV shows the length of time that the trigger line is asserted for output triggers.

TSP command: trigger.digout[n].pulsewidth

Dig<n>Stim-Sel

Digital Line <n>Stimulus Selection

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<m>
$(1 \leq m \leq 8)$
EVENT_COMMAND
EVENT_DIGIO<m>
$(1 \leq m \leq 6)$
EVENT_TSPLINK<m>
$(1 \leq m \leq 3)$
EVENT_LAN<m>
$(1 \leq m \leq 8)$
EVENT_BLENDER<m>
$(1 \leq m \leq 2)$
EVENT_TIMER<m>
$(1 \leq m \leq 4)$
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

n: digital line number (1 to 6)

Description: This PV selects the event that causes a trigger to be asserted on the corresponding digital output line.

TSP command: trigger.digout[n].stimulus

Dig<n>Stim-Sts

Digital Line <n>Stimulus Status

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<m>
($1 \leq m \leq 8$)
EVENT_COMMAND
EVENT_DIGIO<m>
($1 \leq m \leq 6$)
EVENT_TSPLINK<m>
($1 \leq m \leq 3$)
EVENT_LAN<m>
($1 \leq m \leq 8$)
EVENT_BLENDER<m>
($1 \leq m \leq 2$)
EVENT_TIMER<m>
($1 \leq m \leq 4$)
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

n: digital line number (1 to 6)

Description: This PV shows the event that causes a trigger to be asserted on the corresponding digital output line.

TSP command: trigger.digout[n].stimulus

Dig<n>Assert-Cmd

Digital Line <n>Assert Command

bool{

OFF
ON

}

n: digital line number (1 to 6)

Description: When set to 1 or *ON*, this PV asserts a trigger pulse on the corresponding digital I/O line.

TSP command: trigger.digout[n].assert()

Dig<n>Release-Cmd

Digital Line <n>Release Command

bool{

OFF
ON

}

n: digital line number (1 to 6)

Description: When set to 1 or *ON*, this PV releases an indefinite length or latched trigger.

TSP command: trigger.digout[n].release()

4.7 Timer

Timer<n>Enbl-Sel

Timer <n>Enable Selection

bool{

OFF
ON

}

n: timer number (1 to 4)

Description: This PV enables the trigger timer. You must enable a timer before it can use the delay settings or the alarm configuration. For expected results from the timer, it is best to disable the timer before changing a timer setting, such as delay or start seconds. To use the timer as a simple delay or pulse generator, make sure the timer start time in seconds and fractional seconds is configured for a time in the past. To use the timer as an alarm, configure the timer start time in seconds and fractional seconds for the desired alarm time.

TSP command: trigger.timer[n].enable

Timer<n>Enbl-Sts

Timer <n>Enable Status

bool{

OFF
ON

}

n: timer number (1 to 4)

Description: This PV shows if the trigger timer is enabled.

TSP command: trigger.timer[n].enable

Timer<n>Dly-SP

Timer <n>Delay Set Point

float

Min=0.000008

Max=100000

n: timer number (1 to 4)

Description: This PV sets the timer delay. Once the timer is enabled, each time the timer is triggered, it uses this delay period.

TSP command: trigger.timer[n].delay

Timer<n>Dly-RB

Timer <n>Delay Read Back

float

n: timer number (1 to 4)

Description: This PV shows the timer delay.

TSP command: trigger.timer[n].delay

Timer<n>Count-SP

Timer <n>Count Set Point

float

Min=0

Max=1048575 n: timer number (1 to 4)

Description: This PV sets the number of events to generate each time the timer generates a trigger event or is enabled as a timer or alarm. If count is set to a number greater than 1, the timer automatically starts the next trigger timer delay at the expiration of the previous delay. Set count to zero (0) to cause the timer to generate trigger events indefinitely.

TSP command: trigger.timer[n].count

Timer<n>Count-RB

Timer <n>Count Read Back

float

n: timer number (1 to 4)

Description: This PV shows the number of events that are generated each time the timer triggers an event or is enabled as a timer or alarm.

TSP command: trigger.timer[n].count

Timer<n>Gen-Sel

Timer <n>Event Generation Selection

bool{

Elapse
Start and elapse

}

n: timer number (1 to 4)

Description: This PV specifies when timer events are generated. When this PV is set to *Start and elapse*, a trigger event is generated immediately when the timer is triggered and when it elapses. When it is set to *Elapse*, a trigger event is generated only when the timer elapses.

TSP command: trigger.timer[n].start.generate

Timer<n>Gen-Sts

Timer <n>Event Generation Status

bool{

Elapse
Start and elapse

}

n: timer number (1 to 4)

Description: This PV shows when timer events are generated.

TSP command: trigger.timer[n].start.generate

Timer<n>Sec-SP

Timer <n>Start Second Set Point

long

Min=0

Max=2147483647

n: timer number (1 to 4)

Description: This PV configures the seconds of an alarm or a time in the future when the timer will start. When the timer is enabled, the timer starts immediately if the timer is configured for a start time that has passed.

TSP command: trigger.timer[n].start.seconds

Timer<n>Sec-RB

Timer <n>Start Second Read Back

long

n: timer number (1 to 4)

Description: This PV shows, in seconds, the time of an alarm or a time in the future when the timer will start.

TSP command: trigger.timer[n].start.seconds

Timer<n>Frac-SP

Timer <n>Start Fractional Second Set Point

float

Min=0

Max=1

n: timer number (1 to 4)

Description: This PV configures the fractional seconds of an alarm or a time in the future when the timer will start.

TSP command: trigger.timer[n].start.fractionalseconds

Timer<n>Frac-RB

Timer <n>Start Fractional Second Read Back

float

n: timer number (1 to 4)

Description: This PV shows the fractional seconds of an alarm or a time in the future when the timer will start.

TSP command: trigger.timer[n].start.fractionalseconds

Timer<n>Stim-Sel

Timer <n>Stimulus Selection

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<m>
$(1 \leq m \leq 8)$
EVENT_COMMAND
EVENT_DIGIO<m>
$(1 \leq m \leq 6)$
EVENT_TSPLINK<m>
$(1 \leq m \leq 3)$
EVENT_LAN<m>
$(1 \leq m \leq 8)$
EVENT_BLENDER<m>
$(1 \leq m \leq 2)$
EVENT_TIMER<m>
$(1 \leq m \leq 4)$
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

n: timer number (1 to 4)

Description: This PV sets the event that starts the trigger timer. Set this attribute to zero (0) to disable event processing and use the timer as a timer or alarm based on the start time.

TSP command: trigger.timer[n].start.stimulus

Timer<n>Stim-Sts

Timer <n>Stimulus Status

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<m>
($1 \leq m \leq 8$)
EVENT_COMMAND
EVENT_DIGIO<m>
($1 \leq m \leq 6$)
EVENT_TSPLINK<m>
($1 \leq m \leq 3$)
EVENT_LAN<m>
($1 \leq m \leq 8$)
EVENT_BLENDER<m>
($1 \leq m \leq 2$)
EVENT_TIMER<m>
($1 \leq m \leq 4$)
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

n: timer number (1 to 4)

Description: This PV shows the event that starts the trigger timer.

TSP command: trigger.timer[n].start.stimulus

Timer<n>Over-Mon

Timer <n>Overrun Monitor

bool{

No overrun
Overrun

}

n: timer number (1 to 4)

Description: This PV indicates if an event was ignored because of the event detector state.

TSP command: trigger.timer[n].start.overrun

Timer<n>Clr-Cmd

Timer <n>Clear Event Command

bool{

OFF
ON

}

n: timer number (1 to 4)

Description: This function clears the timer event detector and overrun indicator for the corresponding trigger timer number.

TSP command: trigger.timer[n].clear()

4.8 Blender

Blend<n>Op-Sel

Blender <n>Operation Selection

bool{

OR
AND

}

n: blender number (1 or 2)

Description: This PV selects whether the blender performs OR operations or AND operations.

TSP command: trigger.blender[n].orenable

Blend<n>Op-Sts

Blender <n>Operation Status

bool{

OR
AND

}

n: blender number (1 or 2)

Description: This PV shows the blender operation type.

TSP command: trigger.blender[n].orenable

Blend<n>Stim<m>-Sel

Blender <n>Stimulus <m>Selection

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<i>
$(1 \leq i \leq 8)$
EVENT_COMMAND
EVENT_DIGIO<i>
$(1 \leq i \leq 6)$
EVENT_TSPLINK<i>
$(1 \leq i \leq 3)$
EVENT_LAN<i>
$(1 \leq i \leq 8)$
EVENT_BLENDER<i>
$(1 \leq i \leq 2)$
EVENT_TIMER<i>
$(1 \leq i \leq 4)$
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

n: blender number (1 or 2)

m: stimulus number (1 to 4)

Description: This PV specifies the events that trigger the blender. There are four stimulus inputs that can each select a different event. Use zero to disable the blender input.

TSP command: trigger.blender[n].stimulus[m]

Blend<n>Stim<m>-Sts

Blender <n>Stimulus <m>Status

enum{

EVENT_NONE
EVENT_DISPLAY
EVENT_NOTIFY<i>
$(1 \leq i \leq 8)$
EVENT_COMMAND
EVENT_DIGIO<i>
$(1 \leq i \leq 6)$
EVENT_TSPLINK<i>
$(1 \leq i \leq 3)$
EVENT_LAN<i>
$(1 \leq i \leq 8)$
EVENT_BLENDER<i>
$(1 \leq i \leq 2)$
EVENT_TIMER<i>
$(1 \leq i \leq 4)$
EVENT_ANALOGTRIGGER
EVENT_EXTERNAL

}

n: blender number (1 or 2)

m: stimulus number (1 to 4)

Description: This PV shows the events that trigger the blender.

TSP command: trigger.blender[n].stimulus[m]

Blend<n>Over-Mon

Blender <n>Overrun Monitor

bool{

No overrun
Overrun

}

n: blender number (1 or 2)

Description: This PV indicates whether or not an event was ignored because of the event detector state.

TSP command: trigger.blender[n].overrun

Blend<n >Clr-Cmd

Blender <n>Clear Event Command

bool{

OFF
ON

}

n: blender number (1 or 2)

Description: This function clears the blender event detector and resets the overrun indicator of blender<n >.

TSP command: trigger.blender[n].clear()

4.9 Autocalibration

ACalStart-Cmd

Autocalibration Start Command

bool{

OFF
ON

}

When set to 1 or *ON*, this PV causes the instrument to immediately run auto calibration and stores the constants. During auto calibration, a progress message is displayed on the front panel. At completion, an event message is generated. If you have set up auto calibration to run at a scheduled interval, when you send the run command, the instrument adjusts the next scheduled auto calibration to be the next interval. For example, if auto calibration is scheduled to run every 7 days, but you run auto calibration on day 3, the next auto calibration will run 7 days after day 3.

TSP command: acal.run()

ACalRev-Cmd

Autocalibration Revert Command

bool{

OFF
ON

}

When set to 1 or *ON*, this PV causes the instrument to return auto calibration constants to the previous constants. The last run time and internal temperature are reverted to the previous values. The auto calibration count is not changed.

TSP command: acal.revert()

ACalLast-Mon

Last Autocalibration Monitor

string

This record returns, when processed, the date and time when auto calibration was last run. Set *ACalLast-Mon.PROC* to any value in order to get the last calibration time once. Set *ACalLast-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the last calibration time periodically. The date and time is returned in the format:

MM/DD/YYYY HH:MM:SS.NNNNNNNNN

Where:

- MM/DD/YYYY is the month, date, and year
- HH:MM:SS.NNNNNNNNN is the hour, minute, second, and fractional second

TSP command: acal.lastrun.time

ACalCount-Mon

Autocalibration Count Monitor

long

This PV returns, when processed, the number of times automatic calibration has been run since the last factory calibration. Set *ACalCount-Mon.PROC* to any value in order to get the automatic calibration count once. Set *ACalCount-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the automatic calibration count periodically.

TSP command: acal.count

ACalSchAct-Sel

Autocalibration Schedule Action Selection

enum{

NONE
Run
Notify

}

This PV sets the autocalibration action to be executed at the scheduled time.

TSP command: acal.schedule()

ACalSchAct-Sts

Autocalibration Schedule Action Status

enum{

NONE
Run
Notify

}

This PV shows the autocalibration action configured to be executed at the scheduled time.

TSP command: acal.schedule()

ACalSchInt-Sel

Autocalibration Schedule Interval Selection

bool{

8 hours
16 hours
1 day
7 days
14 days
30 days
90 days

}

This PV determines how often the auto calibration action should be executed.

TSP command: acal.schedule()

ACalSchInt-Sts

Autocalibration Schedule Interval Status

bool{

8 hours
16 hours
1 day
7 days
14 days
30 days
90 days

}

This PV shows how often the auto calibration action is configured to be executed.

TSP command: acal.schedule()

ACalSchHr-SP

Autocalibration Schedule Hour Set Point

long

Min=0

Max=23

Specify, in 24-hour time format, when the auto calibration action should occur.

TSP command: acal.schedule()

ACalSchHr-RB

Autocalibration Schedule Hour Read Back

long

Shows, in 24-hour format, the configured time for the auto calibration action.

TSP command: acal.schedule()

ACalNext-Mon

Next Autocalibration Monitor

string

This PV returns, when processed, the date and time when the next auto calibration is scheduled to be run. Set *ACalNext-Mon.PROC* to any value in order to get the next auto calibration date and time once. Set *ACalNext-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the next auto calibration date and time periodically.

TSP command: acal.nextrun.time

ACalDiff-Mon

Autocalibration Temperature Difference Monitor

float

unit: °C

When processed, this PV returns the difference between the internal temperature and the temperature when auto calibration was last run. Set *ACalDiff-Mon.PROC* to any value in order to read the temperature difference once. Set *ACalDiff-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the difference between the current internal temperature and the temperature when auto calibration was last run, periodically.

TSP command: acal.lastrun.tempdiff

ACalLim-SP

Autocalibration Temperature Difference Limit Set Point

float

unit: °C

This sets the maximum accepted instrument internal temperature variation. When the variation exceeds the specified value, the *ACalWarn-Mon* PV is set to 1.

TSP command: No command

ACalWarn-Mon

Autocalibration Temperature Difference Warning Monitor

float

This PV indicates when the instrument internal temperature variation has exceeded the value specified by *ACalLim-SP*. When the limit is exceeded, the PV is set to 1 until a *warning reset* is performed (*ACalRst-Cmd*).

TSP command: No command

ACalRst-Cmd

Autocalibration Reset Warning Command

bool{

OFF
ON

}

When set to 1 or *ON*, this PV resets the autocalibration warning, i.e., the *ACalWarn-Mon* PV is set to 0.

TSP command: No command

4.10 Trigger Model

TMStart-Cmd

Trigger Model Start Command

bool{

OFF
ON

}

Description: When set to 1 or *ON*, this PV starts the trigger model.

TSP command: trigger.model.initiate()

TMAbort-Cmd

Trigger Model Abort Command

bool{

OFF
ON

}

Description: When set to 1 or *ON*, this PV stops all trigger model commands on the instrument.

TSP command: trigger.model.abort()

TMClear-Cmd

Trigger Model Clear Command

bool{

OFF
ON

}

Description: When set to 1 or *ON*, this PV clears the trigger model.

TSP command: trigger.model.load("Empty")

TM-Mon

Trigger Model State Monitor

enum{

Idle
Running
Waiting
Empty
Building
Failed
Aborting
Aborted

}

Description: When processed, this record reads the present state of the trigger model. The trigger model states are:

- Idle: The trigger model is stopped.
- Running: The trigger model is running.
- Waiting: The trigger model has been in the same wait block for more than 100 ms.
- Empty: The trigger model is selected, but no blocks are defined.
- Building: Blocks have been added.
- Failed: The trigger model is stopped because of an error.
- Aborting: The trigger model is stopping because of a user request.
- Aborted: The trigger model is stopped because of a user request.

Set *TM-Mon.PROC* to any value in order to read the trigger model state once. Set *TM-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the trigger model state periodically.

TSP command: trigger.model.state()

TMBlockList-Mon

Trigger Model Block List Monitor

char[2500]

Description: When processed, this record reads the settings for all trigger model blocks. Set *TMBlockList-Mon.PROC* to any value in order to read the trigger model settings once. Set *TMBlockList-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the trigger model settings periodically.

TSP command: trigger.model.getblocklist()

4.11 General

Reset-Cmd

Reset Command

bool{

}

Description: When set to 1 or *ON*, this PV resets parameters to their default settings and clears the buffers.

TSP command: reset()

Access-Sel

User Access Selection

bool{

FULL
EXCLUSIVE
PROTECTED
LOCKOUT

}

Description: This PV defines the type of access users have to the instrument through different interfaces. When access is set to full, the instrument accepts commands from any interface with no login or password. When access is set to exclusive, you must log out of one remote interface and log into another one to change interfaces. You do not need a password with this access. Protected access is similar to exclusive access, except that you must enter a password when logging in. When the access is set to locked out, a password is required to change interfaces, including the front-panel interface. Under any access type, if a script is running on one remote interface when a command comes in from another remote interface, the command is ignored and the message "FAILURE: A script is running, use ABORT to stop it" is generated.

TSP command: localnode.access

Access-Sts

User Access Status

bool{

FULL
EXCLUSIVE
PROTECTED
LOCKOUT

}

Description: This PV shows the type of access users have to the instrument through different interfaces.

TSP command: localnode.access

Login-SP

Login Set Point

string

Description: This PV sends a login command to the instrument using the password entered.

TSP command: login

Logout-Cmd

Logout Command

bool{

OFF
ON

}

Description: When set to 1 or *ON*, this PV sends a logout command to the instrument.

TSP command: logout

PassNew-SP

New Password Set Point

string

Description: This PV sets the instrument password. When the access to the instrument is set to protected or lockout, this is the password that is used to gain access. If you forget the password, you can reset the password to the default:

1. On the front panel, press MENU.
2. Under System, select Info/Manage.
3. Select Password Reset.

You can also reset the password and the LAN settings from the rear panel by inserting a straightened paper clip into hole below LAN RESET.

TSP command: localnode.password

Time-SP

Instrument Date and Time Set Point

string

Format: <year>, <month>, <day>, <hour>, <minute>, <second>

Description: This PV sets the date and time of the instrument.

TSP command: localnode.settime()

Time-Mon

Instrument Date and Time Monitor

string

Format: <day of the week>, <month>, <day>, <hour>, <minute>, <second>, <year>

Description: When processed, this record reads the date and time of the instrument. Set *Time-Mon.PROC* to any value in order to read the instrument date and time once. Set *Time-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the instrument date and time periodically.

TSP command: print(os.date('%c', gettime()))

EvLogCount-Mon

Event Log Count Monitor

long

Description: When processed, this record reads the number of unread events in the event log. Set *EvLogCount-Mon.PROC* to any value in order to read the number of unread events once. Set *EvLogCount-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the number of unread events periodically.

TSP command: print(eventlog.getcount())

EvLogNext-Mon

Event Log Next Event Message Monitor

char[250]

Description: When processed, this record reads the oldest unread event message from the event log. Set *EvLogNext-Mon.PROC* to any value in order to read the oldest unread event message once. Set *EvLogNext-Mon.SCAN* to a valide EPICS SCAN value in order to fetch events periodically.

TSP command: eventlog.next()

ClearEvLog-Cmd

Clear Event Log Command

bool{

    OFF
    ON

}

Description: When set to 1 or *ON*, this PV clears the event log.

TSP command: eventlog.clear()

LineFreq-Mon

Line Frequency Monitor

long

Description: When processed, this record reads the power line frequency setting that is used for NPLC calculations. The instrument automatically detects the power line frequency (either 50 Hz or 60 Hz) when the instrument is powered on. Set *LineFreq-Mon.PROC* to any value in order to read the power line frequency setting once.

TSP command: print(localnode.linefreq)

Temp-Mon

Internal Temperature Monitor

float

unit: °C

Description: When processed, this record reads the internal temperature of the instrument. The instrument checks internal temperature when it updates references when autozero is on. Internal temperature is not checked if autozero is set to off. If the temperature changes more than ±5 °C, the instrument logs an event and displays a message on the front panel that recommends that you perform auto calibration. Set *Temp-Mon.PROC* to any value in order to read the instrument internal temperature once. Set *Temp-Mon.SCAN* to a valide EPICS SCAN value in order to monitor the instrument internal temperature periodically.

TSP command: print(localnode.internaltemp)

TimeSec-Mon

Time Seconds Monitor

long

Description: This PV monitors the instrument time in order to periodically check the connection status. The connection status is display by the *Network-Mon* PV. In order to disable connection monitoring, set *TimeSec-Mon.SCAN* to *Passive*.

TSP command: print(localnode.gettime())

Network-Mon

Network Connection Monitor

bool{

OFF
ON

}

Description: This PV displays the status of the connection between the instrument and the PC running the IOC. In order to disable connection monitoring, set *TimeSec-Mon.SCAN* to *Passive*.

TSP command: No command

Upload-Cmd

Upload Command

bool{

OFF
ON

}

Description: When set to 1 or *ON*, this PV updates all readback and status PVs (-RB and -Sts) of the IOC. An update happens automatically when the IOC is started up, or when the network connection is lost and reconnects. The later requires *TimeSec-Mon.SCAN* to be different from *Passive*.

TSP command: No command

Custom-SP

Custom Command Set Point

char[250]

Description: This PV is an array that can send any string as a command to the instrument, provided that the string length does not exceed the array length.

TSP command: Any command passed as a string