Tektronix

PAMJET
Transmitter Analysis Solution
Application Help

077-1207-11

PAMJET
Transmitter Analysis Solution

Application Help

Register now!
Click the following link to protect your product.
www.tek.com/register

077-1207-11

Tektronix, Inc.

14150 SW Karl Braun Drive

P.O. Box 500

Beaverton, OR 97077

USA

For product information, sales, service, and technical support:

- In North America, call 1-800-833-9200.
- Worldwide, visit *www.tek.com* to find contacts in your area.

Table of Contents

List of Figures

List of Tables

Welcome

The PAMJET Transmitter Analysis software application enhances the capabilities of the DPO/DPS/MSO70000DX/SX series oscilloscopes (with 13 GHz or greater bandwidth), adding transmitter and channel testing for four-level Pulse Amplitude Modulation (PAM4) devices and interfaces for both electrical and optical physical domains.

The PAMJET application can run on the following Tektronix products:

DPO77002SX Oscilloscope

DPS77004SX Oscilloscope

DPO75902SX Oscilloscope

DPS75904SX Oscilloscope

DPO75002SX Oscilloscope

DPS75004SX Oscilloscope

DPO73304SX Oscilloscope

DPS73308SX Oscilloscope

DPO72504SX Oscilloscope

DPO72304SX Oscilloscope

DPO72004SX Oscilloscope

DPO71604SX Oscilloscope

DPO71304SX Oscilloscope

DPO73304DX Oscilloscope

DPO72504DX Oscilloscope

DPO72304DX Oscilloscope

DPO72004DX Oscilloscope

DPO71604DX Oscilloscope

MSO73304DX Oscilloscope

MSO72504DX Oscilloscope

MSO72304DX Oscilloscope

MSO72004DX Oscilloscope

MSO71604DX Oscilloscope

Getting started

Getting started with PAMJET Transmitter Analysis solution

The PAMJET Transmitter Analysis software application runs on the Tektronix performance oscilloscopes listed in the Preface. The application enables transmitter and channel testing for four-level Pulse Amplitude Modulation (PAM4) devices and interfaces, for both electrical and optical physical domains.

Optical testing is typically performed using the Tektronix DPO7OE1 or DPO7OE2 optical probe, although 3rd-party O/E converters may also be used. The electrical and optical capabilities are separately licensed, so some features described in this manual may be disabled depending on which licenses you have available. Trial licenses are separately available for the electrical and optical features.

Using filter files produced by the Serial Data Link Analysis (SDLA) application, the PAMJET Transmitter Analysis application allows you to de-embed probes or fixtures and embed a channel model. PAMJET Transmitter Analysis also allows you to apply several types of equalization from directly within the PAMJET application.

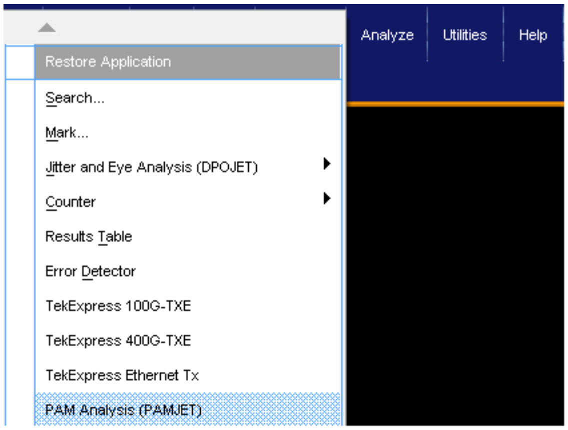
After any waveform processing is complete, PAMJET Transmitter Analysis uses the DPOJET application and additional resources to perform voltage and timing measurements and render eye diagrams on the target waveform. If the waveform contains a repeating pattern, the application additionally calculates the Correlated Waveform (which removes uncorrelated behavior such as random and periodic jitter/noise) and provides additional measurements on this waveform.

The key features of the PAMJET application are:

- Channel embedding and fixture de-embedding by way of .flt files
- Equalization (CTLE, FFE, and DFE, either individually or together)
- Configurable clock recovery
- Jitter measurements per eye for repeating patterns
- Eye diagrams for performance visualization
- BER eye contours for each eye along with IEEE and OIF-CEI-compliant eye measurements such as EW and EH
- IEEE and OIF-CEI-compliant jitter measurements such as J_{4u}, J_{rms}, UUGJ, UBHPJ and even-odd jitter
- PCIe Gen 6 compliant jitter measurements such as $T_{TX\text{-}uTJ}$, $T_{TX\text{-}RJ}$, $T_{TX\text{-}uDJdd}$, $T_{TX\text{-}UPW\text{-}TJ}$, $T_{TX\text{-}UPW\text{-}DJDD}$, JNu(PCIe), and Jrms(PCIe)
- PCIe Gen 6 compliant transmitter voltage parameters such as Signal to Noise and Distortion Ratio (SNDR), Level Separation Mismatch Ratio (R_{LM}), $V_{TX\text{-}DIFF\text{-}PP}$, and $V_{TX\text{-}EIEOS}$
- Symbol error detection and navigation plus SER and BER measurements
- Transmitter characterization measurements: such as Signal to Noise and Distortion Ratio (SNDR) and TDECQ measurements
- Correlated waveform computation
- HTML-style report generation for presentation, sharing or archiving
- Programmable interface for test automation
- Automatic or manual voltage reference thresholds
- Measurement logging to .csv files
- Ck measurements like $V_{di(pk\text{-}pk)}$, VMA, dv_f, dR_{peak}, Gaussian window, Compute eye opening, Receive Filter, ISI_{RES} .

Starting-Closing the application

After you install the PAMJET software on your oscilloscope, start your oscilloscope application and wait for it to initialize. To launch the PAMJET application, select **Analyze > PAM Analysis (PAMJET)** from the oscilloscope menu, as shown in the following figure.



The first time that the application is started after installation, it may take up to two minutes to appear. This is normal, and is due to one-time initialization.

To close the PAMJET application, click the red square in the upper right corner of the application window and then click "Yes" in the verification dialog.

Using the software

This section describes the basic operation of the PAMJET Transmitter Analysis application.

Main controls

The following controls are on the main application panel and are accessible regardless of which tab is currently selected.

Config: Save

Select this button to display a dialog that allows you to save the current PAMJET Transmitter Analysis configuration in a proprietary binary file with the .psf extension.

Config: Recall

Select this button to recall a saved configuration file.

Config: Default

Select this button to restore all configuration settings to the values present when the application is first launched.

Clear

Selecting this button clears all accumulated measurement results. The results are also cleared when you change any configuration parameter.

Recalc

Selecting this button causes the application to perform a processing cycle on the waveform already present in scope memory. This allows you to change configuration parameters such as clock recovery or threshold and then re-run the analysis on the same waveform. For reference waveforms, Single and Recalc behave the same way.

Some measurements cannot produce a result until multiple waveforms have been processed, due to the sample size requirements of the measurement. Since Recalc can only re-analyze the waveform already in the oscilloscope memory, it cannot be used to re-compute these population-intensive measurements.

Single

Selecting this button causes the application to acquire a single waveform and perform a processing cycle, consisting of waveform filtering, equalization, clock recovery, and computation of measurements and eye diagrams. If the scope was in continuous acquisition (Run) mode, it is stopped so that the waveform that was analyzed remains on the screen and in acquisition memory.

Some measurements require a higher population of symbols than a single analysis cycle can obtain. In these cases, you can either select Single again or use Run mode.

Run

Selecting this button causes the PAMJET Application to repeatedly acquire waveforms and perform processing cycles as described for Single. The process continues until you press the Stop button or until a configured population limit has been met. For some measurements (e.g. EW, EH, J_{4u}, J_{rms}, UUGJ and UBHPJ) the results must be accumulated across multiple cycles to build the population required by the measurements. For other measurements, results are re-computed for each new processing cycle.

Stop

The Stop button replaces the Single or Run button when processing is in progress. Selecting the Stop button interrupts processing.

Show Plots

The Show Plots button is only visible when auxiliary plot windows are present, and is used to bring those windows to the foreground in case they have been minimized or are behind the main application window.

Status Bar

The status bar is at the bottom of the screen. During analysis, the status bar reports the progress of the processing.

Help

Selecting the question mark button opens the PAMJET Transmitter Analysis Software User Manual (this document) in a separate window. The manual opens to the section that corresponds to the panel (tab) currently displayed.

Version

Click on About text in the status bar to display the software version of the application.

This is the same information that appears briefly when the application launches.

The version also appears in any HTML-style reports that are generated.

Tabs

Setup

Selecting the Setup tab opens the Setup screen, which allows you to set up the most commonly used analysis parameters, including which type of equalizer to use. See *Setting up analysis parameters* on page 17.

Select

Clicking the Select tab opens a screen where you can choose the measurement category (Electrical vs. Optical). If the electrical category is chosen, then you can choose "PCIe Gen 6" or "400GCK" technologies and then select specific measurements. It also allows you to configure the measurements.

See *Selecting and configuring measurements* on page 24.

Full waveform

Selecting the Full Wfm tab opens the Results for Full Waveform screen. This screen displays the measurement results for the full waveform. See *Viewing the results* on page 29.

Averaged waveform

Selecting the Avg. Wfm tab opens the Results for Correlated Waveform screen. This screen displays the measurement results for the correlated waveform. Results of averaged waveform are displayed only if *Correlated Waveform Analysis* measurement is chosen in the test tree. See *Viewing the results* on page 29.

Rise fall

Selecting the Rise/Fall tab opens the Rise Times and Fall Times screen. The maximum, minimum, and mean rise and fall times are displayed for all transitions in the correlated waveform, along with the number of times each transition type occurs. Results of Rise and fall measurements are displayed only if Rise/Fall measurement is chosen in the test tree. See *Viewing the results* on page 29.

Reports

Selecting the Reports tab opens the Reports screen. This screen allows you to select a file name and save the analysis report.

Log

Selecting the Log tab provides a view of the activity log, which shows information such as when the analysis started and stopped, how many UI have been accumulated, and when data was saved. The log also captures warnings, error messages and other auxiliary information. See *Using the event log* on page 44.

Prefs

Selecting the Prefs tab allows you to adjust user preferences and some infrequently-used configuration items. See *Preferences* on page 45.

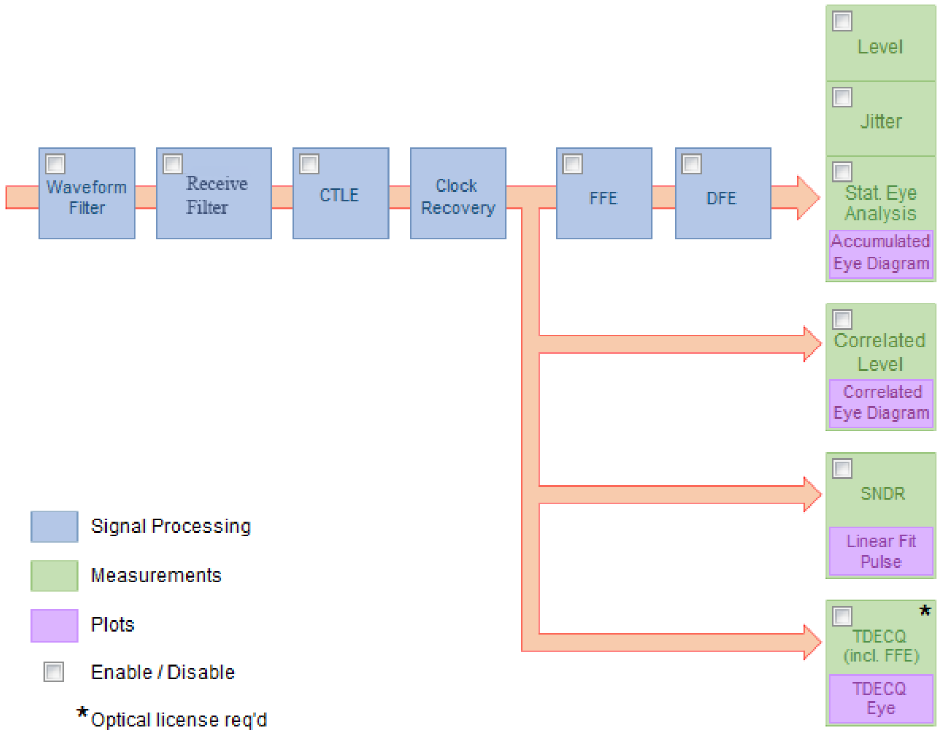
Setting up analysis parameters

Use the Setup screen to set the most common analysis parameters.

Note: Configuration parameters should only be changed when the application is in the idle state.

Processing flow

In broad terms, the individual boxes on the Setup Panel represent processing blocks through which a chosen input signal flows prior to being measured. That processing flow can be better understood by reference to the following figure:



The blue blocks in the flow diagram correspond to the blue boxes on the Setup panel, and all except the Clock Recovery block may be enabled or disabled. When you first launch the application, all the processing blocks are disabled so that the input signal, after clock recovery, goes directly to the measurements and plots.

The green blocks in the flow diagram correspond to measurements. Like the processing blocks, most of them can be enabled or disabled. The plots, represented by purple blocks, cannot be individually enabled but are tied to the measurements. (Some special-purpose plots are not shown in the diagram.)

Note: The Correlated Eye and associated measurements, as well as the SNDR measurements, are always done prior to the FFE and DFE equalizers. For this reason, the Accumulated and Correlated eye diagrams may have a different scale or visual character when FFE or DFE equalization is enabled. Similarly, the TDECQ measurement includes its own FFE, so the TDECQ eye may have a different visual appearance.

Main setup

Choose the PAMJET waveform source that you wish to analyze. The available sources are Ch1 - Ch4, Math1 and Ref1. Some or all of the remaining Math and Ref sources may be used by the PAMJET application (depending on which features and options are being used), and may be overwritten by the PAMJET application without warning.

There are two modes for controlling data rate and pattern, Auto Detect and User-guided. In the default Auto Detect mode, the PAMJET application will automatically detect the data rate during analysis, in most cases. The detected rate always appears in the measurement results on the Full Wfm panel but it also appears in the Main Setup panel when Auto Detect is enabled, after clock recovery has finished.

In the User-guided mode, you enter an approximate data rate to guide the detection process in either Giga-symbols per second (GBaud) or GBits per second. Remember that since PAM4 signaling carries 2 bits per symbol, the symbol rate or baud rate is one half of the bit rate. For example, a bit rate of 51.2 Gbps corresponds to a symbol rate of 25.6 GBaud.

User-guided mode is particularly useful when the input waveform lacks cues about its true rate. For example, one PAM4 Linearity pattern consists of 16 consecutive repeats of each of 10 symbols. If it is running at 25 GBd, the waveform only has transitions every 640 ps even though one unit interval is 40 ps. This makes the signal appear to be a 10-symbol pattern at a symbol rate of 25 GBd/16, or 1.5625 GBd. Using User-guided mode and entering a symbol rate of 25 GBd will cause it to be correctly interpreted as a 160-symbol pattern. If the entered rate is within about 5% of the actual rate, the clock will be recovered and the actual rate will be displayed on the Full Wfm panel.

Like the symbol rate, the data pattern can either be auto-detected (default) or user-directed. When it is user-directed the known symbol stream is imported by means of an ASCII text file. The file may contain symbols from any set of four numerical values that are equally spaced. Some common examples of symbol sets are:

- {0, 1, 2, 3}
- {1, 2, 3, 4}
- {-1, -0.33, +0.33, +1}

If only two numerical values are represented in the text file, they will taken to match the two outer PAMJET symbols. A PRBS13Q text file is installed by default and you can use this as a model to create other pattern files as needed.

If user-guided symbol rate is used, the application will still perform automatic symbol detection. It will then auto-correlate the recovered symbols with the specified ones to determine the most likely pattern phase (timing offset). During this process it will also check whether the algebraic inverse of the given pattern provides a better fit to the waveform. If the inverted pattern is a better fit, this is noted in the message history on the Log tab.

In addition to importing a pattern to control analysis, you can export the automatically-detected pattern as an ASCII text file, either for every processing cycle or for the current waveform. Refer *Preferences* on page 45 for more details.

You can also export an ASCII file that shows all deviations from the expected pattern (whether that pattern was auto-detected or user-provided). See the **export errors** command in the GPIB Command Reference portion of this manual for more details.

Clock recovery

Clock recovery is performed using a noise-tolerant software model of a phase locked loop (PLL). You can choose a Type 1 or Type 2 loop (which in most contexts are equivalent to 1st and 2nd order loops, respectively). You can enter the bandwidth in Megahertz, and for Type 2 loops, the loop damping.

The bandwidth entered is the Jitter Transfer Function (JTF) bandwidth, which is the bandwidth of the high-pass filter function defining which jitter is NOT tracked by the clock recovery. For Type 1 loops, the JTF bandwidth is the same as the loop bandwidth. For Type 2 loops, the JTF and loop bandwidth are slightly different (by a factor that depends on the damping factor). For most standards, it is the JTF bandwidth that is specified.

You can also switch between the configurable PLL and fixed behavioral CDR methods. The behavioral clock recovery contains CDR models of transfer functions based on the standard requirements.

The collections available at this time are:

- PCIe Gen6 64G 0p7 (Nov 2021)

If you have enabled one or both equalizers, the clock recovery is applied after the CTLE and jointly with the FFE/DFE processing.

Threshold setup

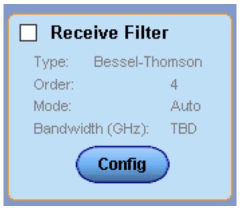
The three thresholds nominally correspond to the centers, in the vertical dimension, of the three eyes in the PAMJET eye diagram. By default (Auto Detect), appropriate thresholds are picked based on analysis of symbol levels in the source waveform, and displayed both on the Config tab and on the Full Wfm results screen. To designate specific thresholds, unselect the Auto Detect checkbox and directly enter the desired thresholds using units of millivolts (electrical) or microwatts (optical).

Waveform filter

The Waveform Filter feature allows you to modify the source waveform by applying a FIR filter, using the oscilloscope Math system. Click the Waveform Filter checkbox to enable the feature and use the Browse button to select the desired filter file, which will have a .flt extension. See your oscilloscope documentation for more information on filter files.

One common use for waveform filters is to de-embed a probe or fixture, or to embed a reference channel. The SDLA application can be used to create or combine filters that can be applied using this feature.

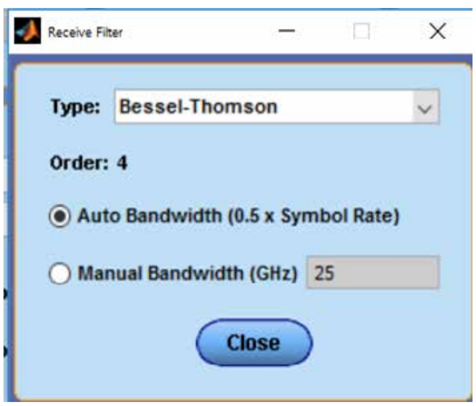
Receive filter



When the Receive filter checkbox is enabled, a filter based on the choices is applied to the source waveform after any waveform filter and before CTLE. This filter is required for several compliance measurements.

Note: Be careful not to apply the BT filter twice. For example, when using an optical probe, the probe often includes an Optical Reference Receiver (ORR), which has a Bessel-Thomson response. Be sure to disable the ORR in the probe. When working with Tektronix optical probes (e.g. DPO7OE1), the PAMJET application automatically sets the probe response to "Flat to maximum bandwidth" if the BT filter is enabled and the measurement category is set to "Optical".

The type of receive filter, order, filter's selection mode(auto/manual), and -3 dB bandwidth are displayed in the filter's setup panel when the filter configuration menu is closed. To adjust these parameters, press the **Config** button under the bandwidth to open the Receive Filter configuration window.



The receive filter can be configured to either “Bessel-Thomson” or “Butterworth”. The order of the filter is shown based on the choice made. In Auto Bandwidth mode, the PAMJET application automatically sets the filter's -3 dB bandwidth to

- 0.5 of the actual symbol rate for Bessel-Thomson filter.
- 0.75 of the actual symbol rate for Butterworth filter.

In Manual Bandwidth mode, you can specify the precise -3 dB bandwidth. For every processing cycle during which the Receive Filter is enabled, the PAMJET application automatically exports the created filter in the format of a .flt file ("ReceiveFilter.flt") to `C:\Users\<username>\Tektronix\TekApplications\PAMJET\Filters`. One common use for this feature is to apply the filter file in scope MATH or other analysis packages to emulate the Receive filter.

The filter's selection mode and -3 dB bandwidth are displayed in the filter's setup panel when the filter configuration menu is closed. To adjust these parameters, press the Config button under the bandwidth to open the Bessel-Thomson Filter configuration window.

CTLE, FFE and DFE

Depending on the characteristics of the transmitter and signal channel, the PAMJET eye might be closed at the channel output. An equalizer can be used to open the eye back up. By checking the associated boxes in the Setup screen, one can model one or multiple of the following types of equalizers:

- Continuous-time linear equalizer (CTLE)
- Feed-forward equalizer (FFE)
- Decision-feedback equalizer (DFE)

Set the equalizer parameters as desired for modeling the effects of the equalizer on your signal. Change the parameters and run the analysis again to see the effects of various settings.

The CTLE model offers two types of operation. The first is Custom design, in which you can directly control individual poles and zeros. You can choose between a design with one zero and two poles (typical of first-generation standards) or a design with two zeros and three poles. Once the design is chosen, you can directly adjust each available parameter.

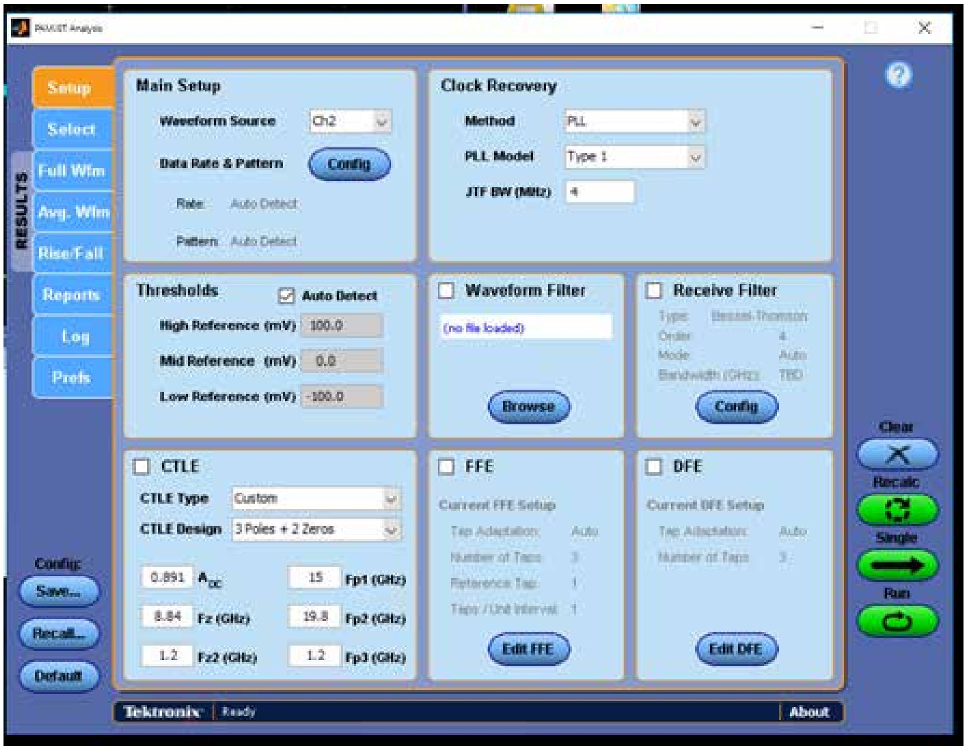
The other type of CTLE is one based on selectable aggregations of parameters, called Presets, which are usually specified by the standards bodies, and usually identified by the amount of de-emphasis such as '1 dB' or '3.5 dB'. For a given preset, you cannot change individual CTLE parameters but you can view the pole and zero frequencies that are in effect. First use the CTLE Type control to choose a collection of presets, and then use the Preset Name control to select a specific preset. The collections available at this time are:

- OIF-CEI Gen I: 2-pole/1-zero design per oif2014.230.06.pdf (Feb 2015)
- OIF-CEI Gen II: 3-pole/2-zero design per oif2014.230.10.pdf (Feb 2017)
- IEEE 802.3bs Draft 3.3: 3-pole/2-zero design with 17 presets (Jul 2017)
- IEEE 802.3bs Draft 3.5 (final): 3-pole/2-zero design with 17 presets (Oct 2017)
- PCIe Gen6 64G 0p7: 6-pole/3-zero design with 11 presets (Nov 2021)
- IEEE 802.3ck TP1a: 3-pole/2-zero design with 54 presets (June 2022)
- IEEE 802.3ck TP4 Far-end: 3-pole/2-zero design with 40 presets (June 2022)
- IEEE 802.3ck TP4 Near-end: 3-pole/2-zero design with 25 presets(June 2022)
- USB-Gen4 v2 Captive device: 1-pole 1-zero design with 10 presets (Oct 2021)

Don't forget to enable the checkbox at the top of the CTLE block; otherwise the CTLE model will not take effect. Similar to Bessel-Thomson filter, for every processing cycle during which the CTLE is enabled, the PAMJET application automatically exports the created filter. The filter is exported in the format of a .flt file ("CTLE.flt") to C:\Users\<username>\Tektronix\TekApplications\PAMJET\Filters.

Note: Although the CTLE poles are designated using numerical suffixes (e.g. Fp1 and Fp2), the order is not important (and is not consistent across the various standards that use them). The equalizer operates exactly the same way regardless of the order in which the pole frequencies appear. Likewise, a CTLE with two zeros will behave exactly the same way independent of the order in which the two zero frequencies appear.

As with CTLE, FFE and DFE can be individually enabled or disabled by using their respective checkboxes.

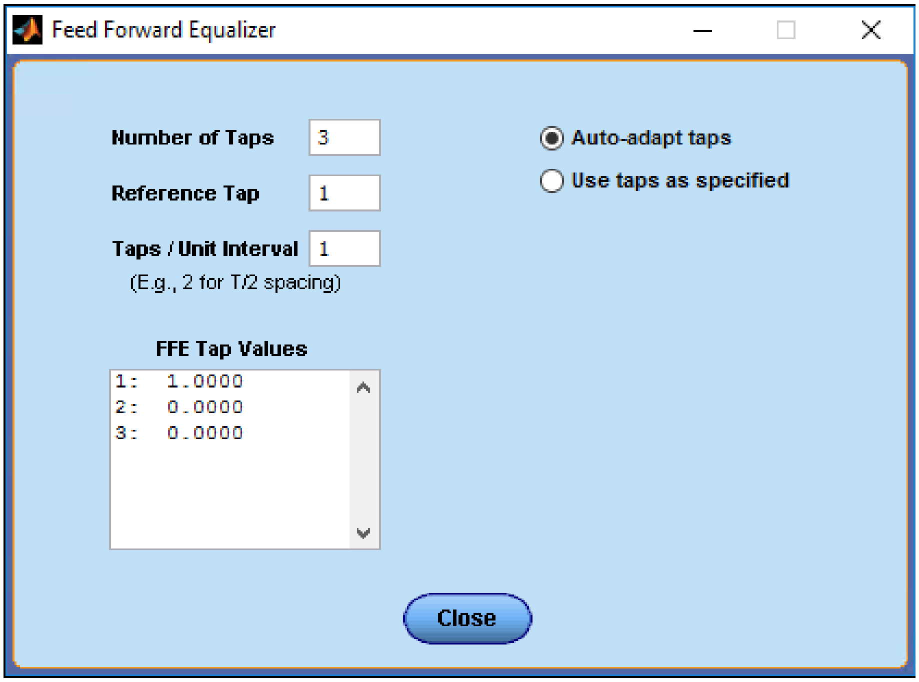


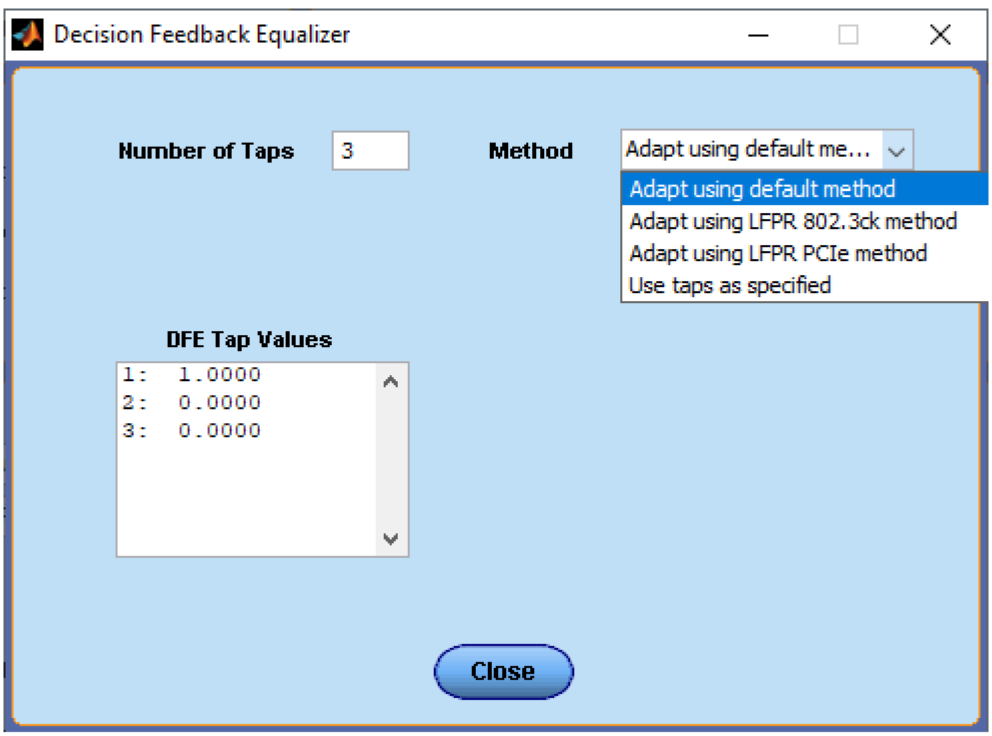
FFE and DFE can be individually configured by clicking on “Edit FFE” and “Edit DFE”, which brings up a separate window for configuring each equalizer.

The following options are available for FFE/DFE tap adaptation:

- Auto-adapt taps: The adaptation routine starts by identifying initial tap settings and adjusts them to optimize recovery of the data and clock.
- Use taps as specified: The equalizer uses the current tap values as input on the graphical user interface. Use the entered values without changes and adaptation.
- Adapt using LFPR 802.3ck method [License Dependent]: Calculates the tap values using the Linear Fit Pulse Response (LFPR) as defined by IEEE 802.3ck.
- Adapt using LFPR PCIe method [License Dependent]: Calculates the tap values using the Linear Fit Pulse Response (LFPR) as defined by PCI Express Base Spec Rev 6.0 .

Note: The LFPR DFE methods automatically set the number of DFE taps and apply constraints to be spec compliant. However, you can subsequently adjust the number of taps to be different from the spec definition which may violate the constraints. Using any LFPR based DFE requires the pattern length to be at least 30.


Feed Forward Equalizer
Number of Taps
3
Reference Tap
1
Taps / Unit Interval
1
(E.g., 2 for T/2 spacing)
Auto-adapt taps
Use taps as specified
FFE Tap Values
1: 1.0000
2: 0.0000
3: 0.0000
Close


Decision Feedback Equalizer
Number of Taps
3
Method
Adapt using default me...
Adapt using default method
Adapt using LFPR 802.3ck method
Adapt using LFPR PCIe method
Use taps as specified
DFE Tap Values
1: 1.0000
2: 0.0000
3: 0.0000
Close

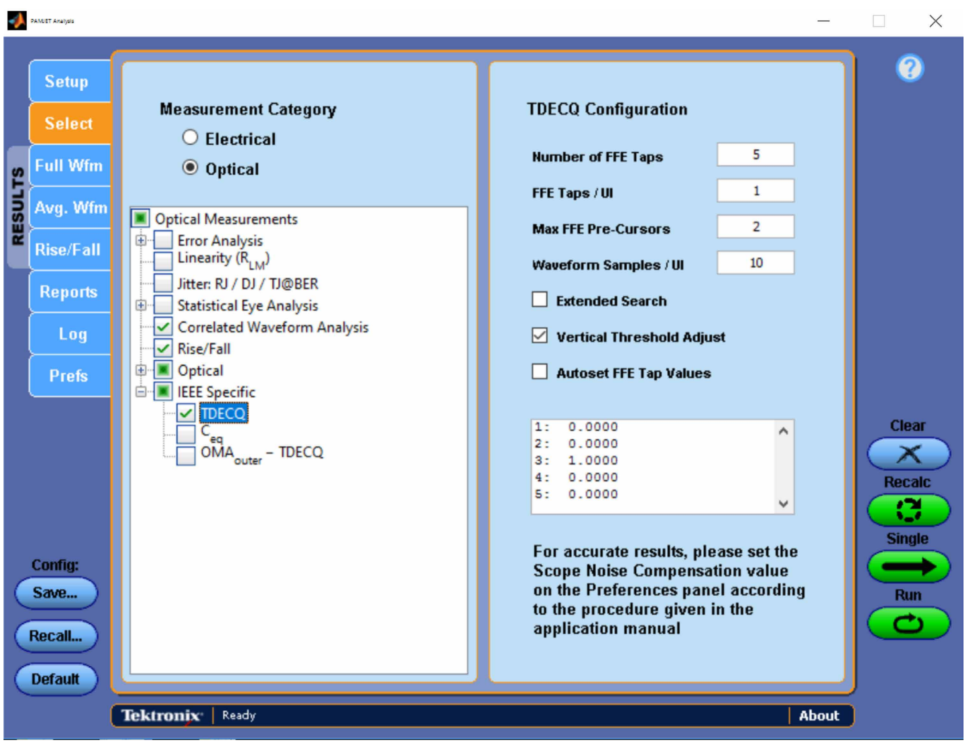
Parameter name		Parameter value limit
FFE	Number of taps	[0, 25]
	Reference tap position	1. Cannot exceed number of taps 2. (Ref tap - 1) /Taps per UI must be an integer
	Number of taps per UI	1. [1, 10] 2. Cannot exceed number of taps
DFE	Number of taps	[0, 16]

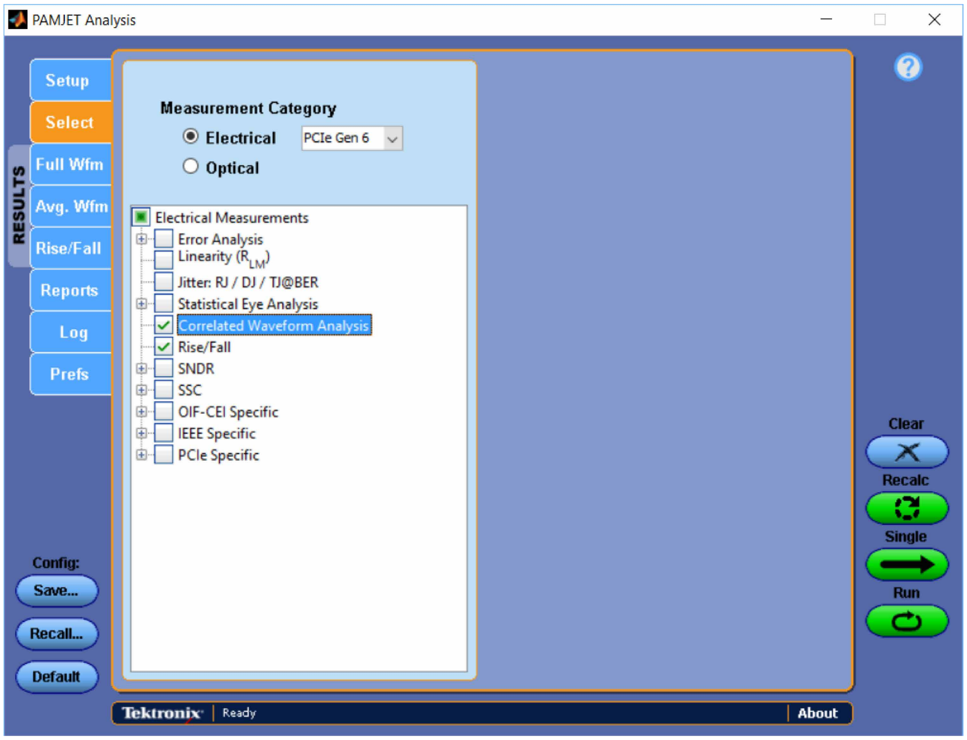
Selecting and configuring measurements

The Select screen allows you to:

- Choose either the Electrical or Optical domain
- Select different Electrical technologies
- Select which specific measurements to perform
- Configure any necessary measurement options

The Measurement Category radio buttons let you choose whether to perform electrical or optical measurements. If technology licenses are available, a drop-down menu will appear in front of the Electrical radio button and will populate the measurements accordingly. These categories are separately licensed, so you may find that one category has been pre-selected. If only one technology license is available, it will always remain selected.





Below the radio buttons, a hierarchical selection tree allows you to choose measurements either individually or by groups. Selecting only the measurements you need can speed processing, and can also reduce clutter in the results screens and reports that you save.

To select an individual measurement, click the checkbox to the left of the measurement name. To select an entire group, click the corresponding group's box. If the group already contains a mix of selected and unselected measurements, a first click will de-select all, and the second will select all.

When you select or de-select a measurement or click directly on the measurement name, a new panel may appear to the right of the selection tree showing any configuration parameters for the highlighted measurement. If that measurement has no specific parameters, the right side of the screen will be blank. In many cases a configuration panel will be applicable to an entire group, in which case it only needs to be set for one measurement and will apply across the group.

For descriptions of the individual measurements, see *Measurements* on page 55. For descriptions of configuration panels that apply to specific measurement groups, see *Statistical eye analysis* on page 25, *BER targets for jitter analysis* on page 27, and *Measurement of SNDR (Signal to Noise and Distortion Ratio)* on page 27.

Statistical eye analysis

Several PAM standards employ eye analysis that monitors both horizontal and vertical eye closure on a prescribed population of symbols. Since PAM interfaces typically have low noise margin and use coding (such as Forward Error Correction or FEC) to attain higher reliability, the performance goal for the physical layer bit error rate (BER) is usually 1e-5 to 1e-6. The applicable standards rely on direct acquisition of sufficient population for direct measurement rather than extrapolating eye closure from a smaller population. Example measurements based on this technique are EW6, EH6 and VEC, and the standards also define intermediate values (Hupp, Vupp, Hmid, Vmid, Hlow, Vlow) that are used to calculate the prescribed measurements. An analytical benefit of this approach is that BER eye contours are available that can give insight into eye closure at all timing phases and reference levels simultaneously.

Enable Statistical Eye Analysis : This checkbox turns the Statistical eye analysis feature on (default) or off. The measurement system behaves slightly differently depending on whether the feature is on, as defined here:

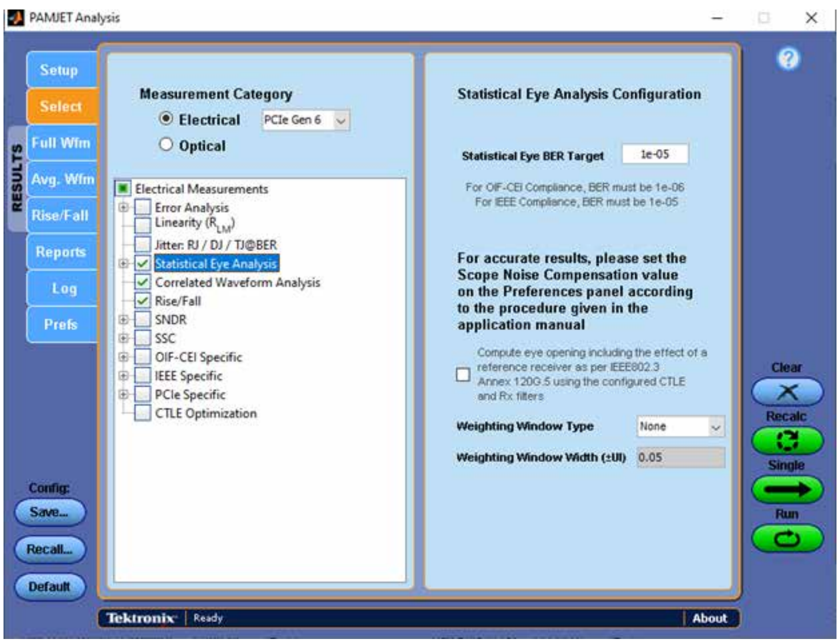
- Statistical Eye Analysis Off: All eye diagrams are rendered by the DPOJET application and automatically imported into the PAMJET Transmitter Analysis application. On the Full Wfm results panel, the last two columns in the eye measurement table display Eye Width and Eye Height. The EW6, EH6 and VEC measurements are displayed as N/A.

- Statistical Eye Analysis On: A second plot window, titled PAMJET Plots, is created as analysis proceeds, and is used to accumulate annotated eye diagrams that show the width and height measurement locations. In this mode, the eye diagrams are rendered directly by the PAMJET application rather than being imported from DPOJET. On the Full Wfm results panel, the last two columns in the eye measurement table change to H_eye and V_eye. The EW6, EH6 and VEC measurements as well as the H_eye and V_eye measurements are only shown if an adequate symbol population has been accumulated; otherwise a footnote to the tables indicates that more population is needed. The table below shows how the H_eye and V_eye results shown in the eye measurement table map to the values {Hupp - Vlow} prescribed by CEI-56G-VSR-PAMJET (oif2014.230 draft 7, dated 10-June-2016):

Row	H_eye	V_eye
Upper	Hupp	Vupp
Middle	Hmid	Vmid
Lower	Hlow	Vlow

Statistical Eye Analysis BER Target: This statistical level determines what BER contour line will be drawn in the three PAM4 eyes. The contour lines are, in turn, used to determine the EW, EH and VEC measurement values. For OIF-CEI compliance, the target must be 1e-6. If it is changed to a different value (say, 1e-5) then the EW6 and EH6 measurements on the Full Wfm panel change correspondingly (in this example, to EW5 and EH5). Since these measurements rely on direct measurement rather than extrapolation, changing the BER target will also change the population of symbols that must be acquired in order to make the measurement. The required population is 4 * 1/(BER Target), so for the default target of 1e-6, a population of 4e6 (or 1e6 in each of the four symbol levels) is needed. (The application will actually make the measurement if at least 95% of this population is acquired.) Eye diagrams retain their information across processing cycles unless cleared, so if the number of symbols is insufficient, you can click "Single" again to build greater population (assuming you have a live signal).

Additional configurations



Option to Enable Reference Receiver: When this checkbox is enabled, the noise due to reference receiver as defined in the IEEE 802.3 Annex 120G.5 is considered. The computation is dependent on the CTLE and Receive Filter configurations in the Setup menu <add the reference>.

Weighting Window Type:

This determines the type of windowing technique to be applied to the vertical probability density functions (PDF) around the eye center.

- **None**: Consider only the vertical PDF at the eye center. No weighting.

- **Rectangular:** The vertical PDFs around the eye center are weighted uniformly.
- **Gaussian**: The vertical PDFs around the eye center are weighted using a gaussian function as defined in the IEEE 802.3 Annex 120G.5.

Weighting Window Width (±UI):

This determines the number of vertical PDFs to be considered around the eye center. The number is the amount of Unit Interval (UI) that needs to be considered. For example, a value of "0.05" conveys that about 0.05 (or 5%) of the Unit Interval from the left of eye center and to the right of the eye center should be considered for the weighted window. The value is configurable from 0.01 (1%) till 0.1 (10%) of the Unit Interval (UI) in steps of 0.005 (0.5%).

BER targets for jitter analysis

These three BER Target values correspond to the three TJ@BER columns in the central results table on the Full Wfm panel. For example, when a value of -12 is entered on the Prefs panel (corresponding to a BER target value of 1e-12 or one expected error for every 1 x 10^{12} bits), the corresponding column heading in the results table changes to TJ@-12. This shorthand for the more conventional TJ@1E-12 is due to space constraints.

These targets are used for conventional jitter analysis and eye width extrapolation, and should not be confused with the Statistical Eye Analysis BER Target.

Measurement of SNDR (Signal to Noise and Distortion Ratio)

Signal to Noise and Distortion Ratio is defined as the ratio in dB between the square of the linear fit pulse peak and the sum of squares of linear fit error (σ_e) and vertical noise (σ_n).

$$SNDR = 10\log_{10}\left(\frac{p_{max}^2}{\sigma_e^2 + \sigma_n^2}\right) dB$$

For standards compliance it should be measured on the pattern prescribed by the applicable standard, although other repeating patterns are acceptable if the pattern length is no longer than 10000.

In addition to reporting the SNDR value in dB, the component values p_{max}, σ_e and as well as the σ_n value for each individual PAM level are provided. All measurement values appear in the upper left (scrollable) table on the Full Wfm results tab.

Samples per Symbol (M): This determines the granularity with which the individual symbols in the waveform are analyzed. It has no connection with the number of oscilloscope samples per symbol. The valid range is 32 ≤ M ≤ 200, with a default of 32.

Linear Pulse Length (Np): This determines the number of symbols included in the linear fit analysis, and should be long enough to capture most of the energy in the pulse response. The valid range is 2 ≤ Np ≤ 10000, with a default value of 14.

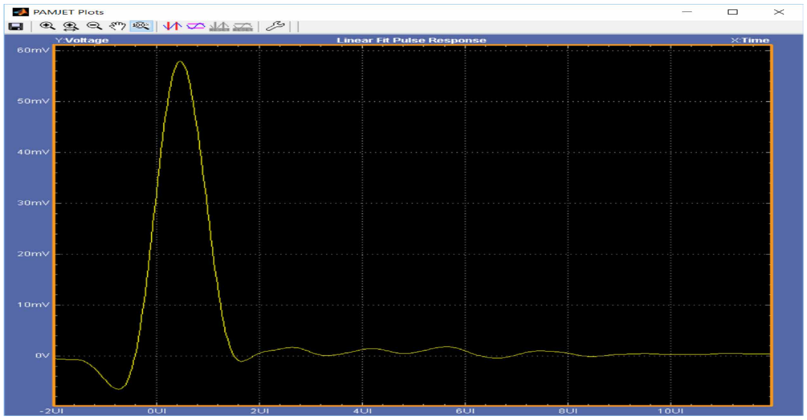
Linear Pulse Delay (Dp): This determines the duration, in symbols, of the part of the pulse response that is assumed to occur prior to the main pulse. Its typical value is 2, to allow for some pre-shoot, but the valid range is 2 ≤ Dp ≤ Np-2.

Show Detail Plots: When this checkbox is enabled, the following auxiliary plots are created in a separate plot window, for additional insight into non-essential aspects of the measurement:

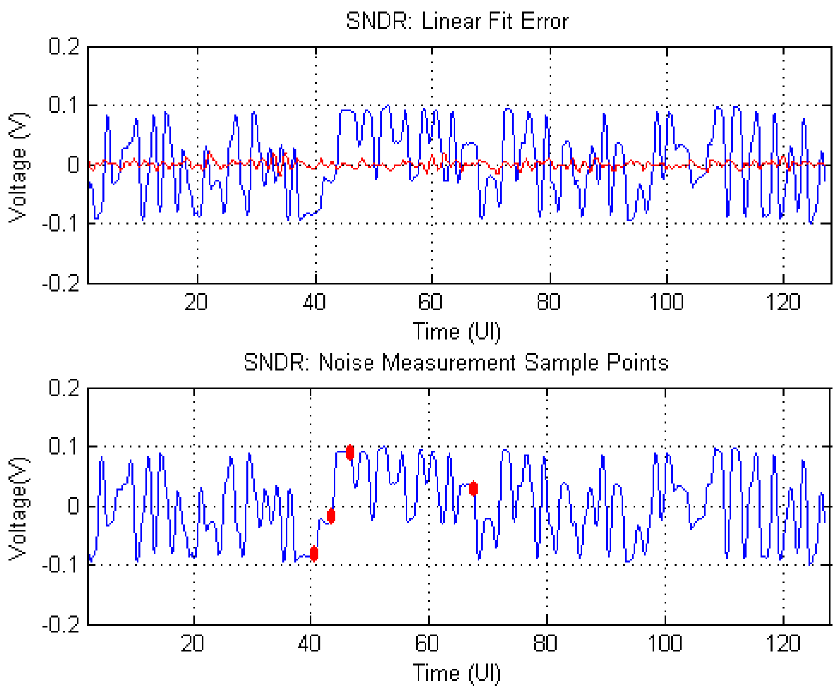
- Linear fit error, overlaid with one cycle of the averaged waveform.
- High and low symbol positions at which the noise measurements are taken within the averaged waveform.

(These plots are not saved in Report files.)

Whenever the SNDR measurement or any of its subcomponents are enabled, the linear fit pulse response is shown graphically in the main PAMJET Plots window, similar to the following image:



The optional detail plots, when enabled, typically appear as follows:



Note: SNDR(PCIe) is majorly identical to the regular SNDR measurement and will report the same values. It only differs by the configurable parameters (M, Np, Dp) and in the computation of SigmaN. The default values of M, Np, and Dp for SNDR(PCIe) are 32, 600, and 4 respectively.

Running an analysis

1. Set the signal path parameters on the Setup screen and verify that the options on the Prefs screen are as desired.
2. Click the Run, Single, or Recalc button to perform the analysis.
3. If the mode is Run or Single and the source directly or indirectly uses live channels (Ch1 - Ch4), a new waveform is acquired into the oscilloscope waveform memory. Initially the application uses the trigger settings that have been configured on the scope, but if no trigger occurs within about 10 seconds, the application changes to an edge trigger on the chosen source waveform.

   If the waveform filter is enabled, the PAMJET application applies the filter and places the result in Math 4 on the oscilloscope. If one or more of the three forms of equalization is enabled, the output of the equalizer(s) is placed in Ref4. A synthesized clock waveform representing the recovered clock is placed in Ref3. If any measurement under the Statistical Eye Analysis category is enabled, the PAMJET application renders an eye diagram of the complete waveform at high resolution, accumulating the diagram with prior acquisitions to build population. If the waveform contains at least three full cycles of a repeating pattern, the PAMJET application calculates one repeat of the correlated waveform and places it in Ref2. Finally, the PAMJET application sets up the appropriate measurements in DPOJET using either Math4, Ref4 or the original source waveform as the measurement source.

4. If correlated waveform analysis is possible, the PAMJET application creates the correlated waveform eye diagram and measurement values.
5. The PAMJET application finishes the current cycle by performing selected measurements based on the Select tab. For example, some jitter (RJ/DJ/TJ) measurements require DPOJET sequencing. When that is the case, the measurements and plots are imported from DPOJET for display in the PAMJET user interface. If no Statistical Eye Analysis measurements are enabled, DPOJET is also used to render the eye diagram.
6. If the mode is Run, steps 3 through 5 are repeated, until any population limits have been satisfied. A closed center eye may optionally be configured to halt Run mode.
7. The PAMJET application turns off the recovered clock waveform (Ref3) when processing completes, to avoid obscuring other waveforms on the oscilloscope screen. You can turn it back on from the oscilloscope File menu, if you want to view the recovered clock.

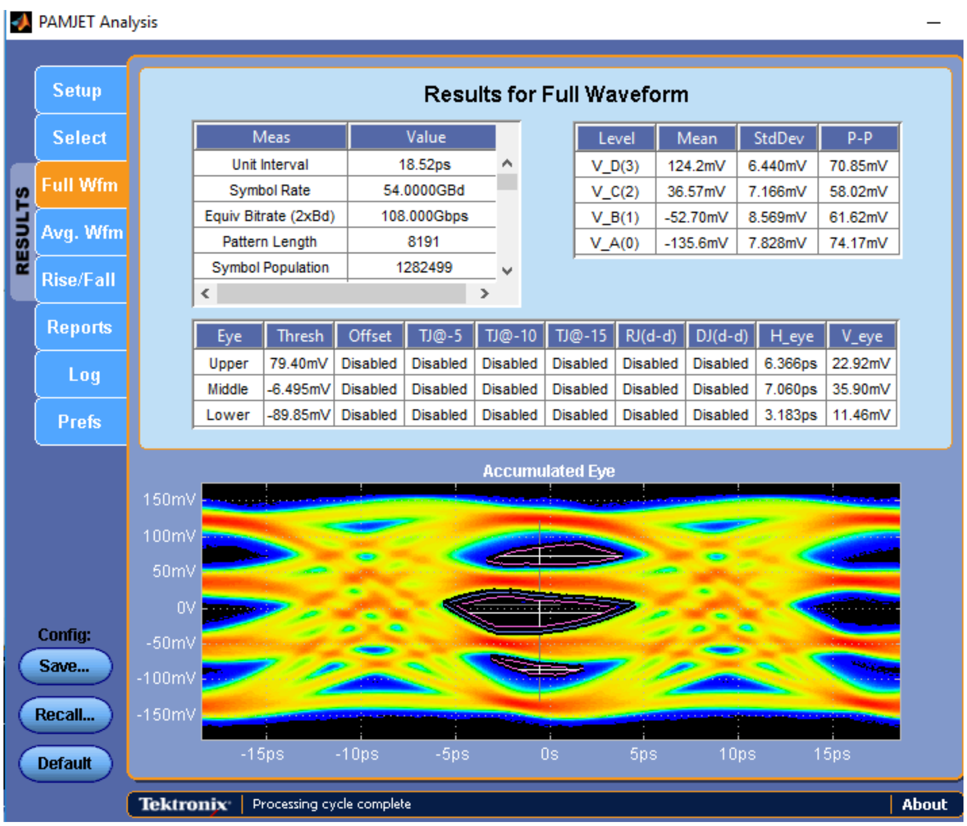
Viewing the results

To view measurement results open the following tabs.

Full waveform

Selecting the **Full Wfm** tab displays the Results for Full Waveform screen, which includes the Accumulated Eye diagram. Values for selected measurements are displayed above the eye diagram.

Note: Many measurements appear in the upper left table, and can be viewed by using that table's scroll bar.

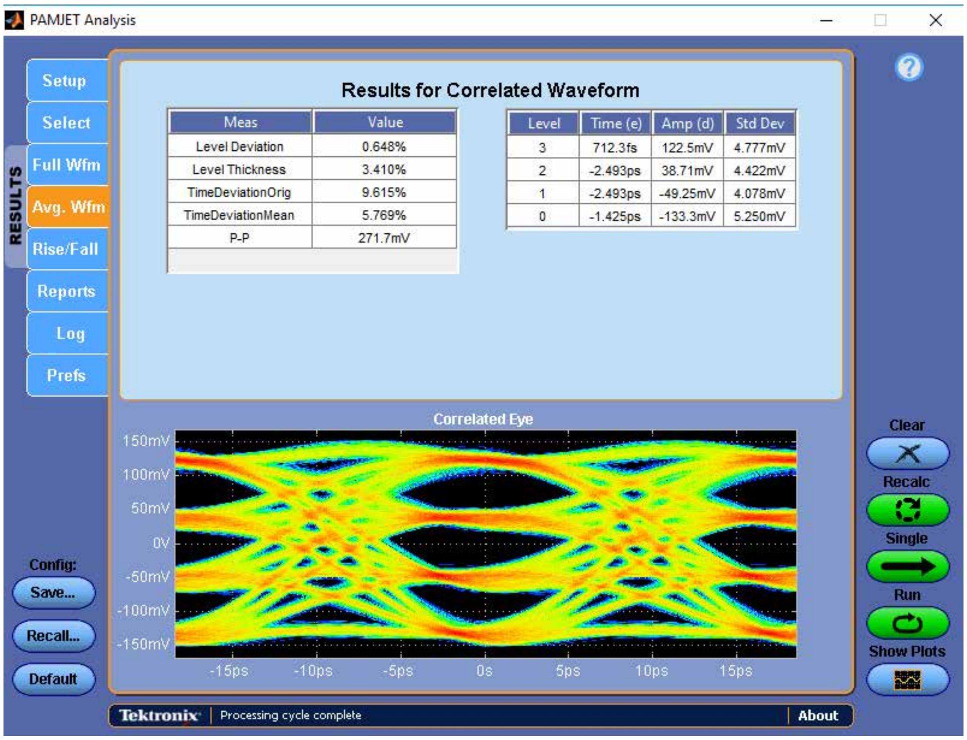


The eye diagram is updated at the completion of each acquisition cycle.

Correlated waveform

Selecting the **Avg. Wfm** tab displays the Results for Correlated Waveform screen, which includes an eye diagram from which uncorrelated jitter and noise have been removed. Values for time, amplitude, standard deviation, level, and time deviation are displayed above the eye diagram.

These results are displayed only if **Correlated Waveform Analysis** measurement is selected in the measurement tree.



Rise and fall times

Selecting the **Rise/Fall** tab displays the Rise Times and Fall Times screen. Selecting the tab displays the maximum, minimum, and mean rise times and fall times for each of the six transition types in the PAMJET eye diagram, as well as the population (count) for each transition type.

The graphics beside each group of results help explain which transition is represented by that group, and the legend in the lower right corner of the screen explains which results fields correspond to the max (maximum), mean, min (minimum) and count.

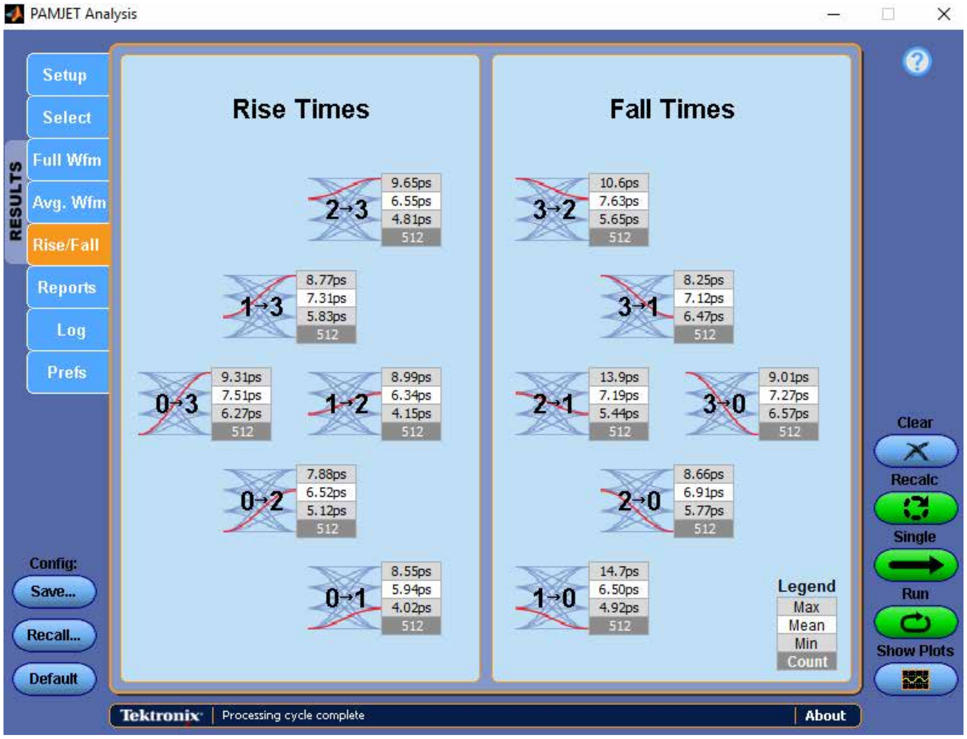
The rise and fall time measurements are made on the correlated waveform, because the deterministic waveform behavior (ISI) is usually of more interest than effects that may be due to instantaneous noise or jitter.

Rise and fall time measurement results are displayed only if **Rise/Fall** measurement is selected in the measurement tree.

The rise and fall times are measured between the 20% and 80% levels for the corresponding transition.

- If one or more transitions in the correlated waveform fail to reach one of the thresholds within the symbol period, then both the Mean and Max rise or fall time will remain blank, and only the Min time (corresponding to the fastest edge) will be displayed.
- If there are no transitions between a specific pair of symbol levels (which can happen depending on the symbol pattern), then all three statistics for that transition will remain blank and the corresponding count will be zero.

The sum of the populations for the 12 transition types may be equal to the waveform's pattern length, but it will usually be less. This is because no transition occurs when two adjacent symbols are alike (e.g. 0⟶0, 1⟶1, 2⟶2 or 3⟶3).

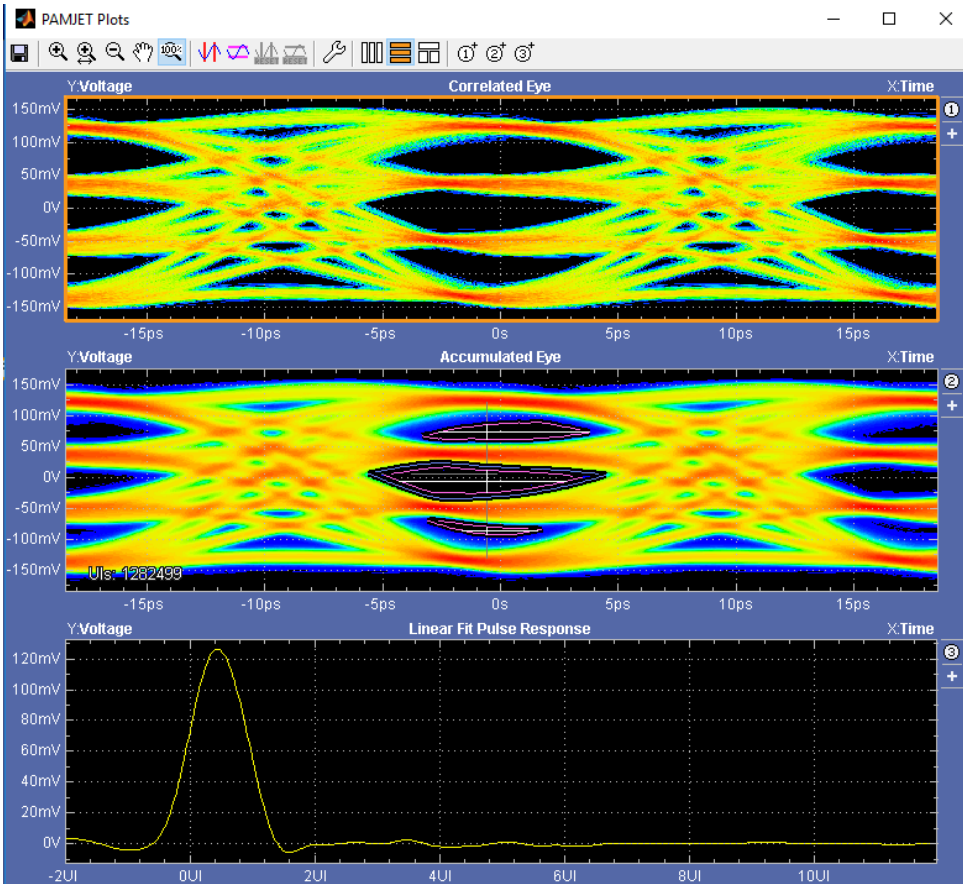


PAMJET Plots window

If any measurements from the Jitter, Statistical Eye Analysis, SNDR or IEEE-specific (Optical option only) groups are enabled, a dedicated PAMJET Plots window will appear whenever measurement results are available in the PAMJET Transmitter Analysis application. If the Jitter or Statistical Eye Analysis measurements are enabled, the window will include the accumulated eye and correlated eye. If SNDR measurements are enabled, the window will include the linear fit pulse response. If the TDECQ measurement is enabled, the TDECQ Eye will be displayed.

The window offers plot selection, resizing, additional zoom controls, horizontal and vertical cursors and plot image or data export. If the PAMJET Plots window is minimized or is behind another window, it can be brought to the foreground by selecting the Show Plots button on the main user interface window. Below is an example of the PAMJET Plots window:

Note: The Show Plots button will be absent if there are no measurement results, for example after you click the Clear button.



The toolbar functions in the PAMJET Plots window are similar to those used in the DPOJET application, and can be easily learned by direct exploration or by referring to the Toolbar functions in plot windows topic in the DPOJET manual. Note that before using some of the toolbar buttons, you must select one of the plots by clicking on it. A plot that has been selected will have an orange border.

The following specific plots may appear in the window:

Accumulated Eye: This is a conventional eye diagram, formed by overlaying all unit intervals in the current waveform along with those from prior waveforms. Clearing results or changing a configuration parameter causes this accumulation to be reset.

Correlated Eye: This is an eye diagram showing only the deterministic, pattern-related behavior (DDJ, ISI) of the waveform, and excluding the effects of random jitter/noise, crosstalk or periodic noise not correlated with the pattern. It is only available for signals that carry a repeating pattern. If the pattern length is N, there will be exactly N trajectories through the eye. In Run mode, this eye is only calculated on the first acquisition since the signal's statistics are presumed to be stationary.

Linear Fit Pulse Response: This shows the best-fit pulse that is used as a basis for finding pulse amplitude (p_{max}) and calculating noise (σ_n) and distortion (σ_e), components of the SNDR measurement, as well as certain compliant measurements such as SNR_{ISI}.

TDECQ Eye: This is an eye diagram that reflects the processing done within the TDECQ measurement algorithm. The TDECQ algorithm requires designing a feed-forward equalizer that is optimum in a particular sense, and the eye shows the effect of the chosen equalizer. It also shows the two vertical slices where histogram analysis is performed and the three eye thresholds used. It only renders the first 30,000 to 100,000 symbols rather than the full waveform into the eye.

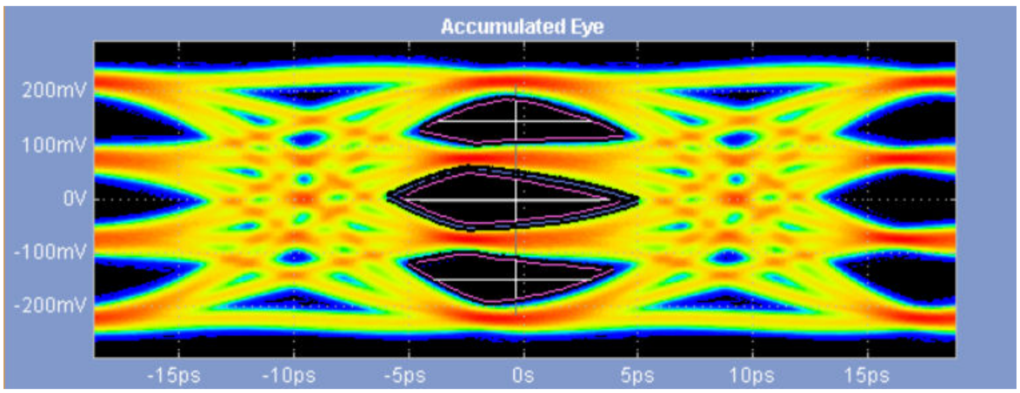
Accumulated eye plot annotations

To support the statistical eye measurements such as EH and EW, graphical annotations are placed on the Accumulated Eye plot when sufficient eye population has been collected.

In the lower left corner of the accumulated eye plot, a text annotation shows the number of Unit Intervals (UIs, also known as Symbols) that are represented in the eye. If the number of UIs is not yet sufficient to allow the BER-related measurements to be performed, the annotation also shows the percent complete.

If the number of UIs is not great enough for the measurements to complete but is greater than 4000, an eye contour is drawn in gray to show the 1e-3 BER level in the middle eye of the accumulated eye diagram. A horizontal annotation line is drawn at the widest point of this contour, and a vertical line is drawn at the horizontal midpoint of the widest-point line. The vertical line signifies the point at which the EW and EH measurements will be taken when sufficient overall population has been accumulated. These annotations can be seen in the Accumulated Eye plot above.

Once the eye population has reached an acceptable level, BER contours corresponding to the measurement's target BER are drawn in all three eyes as shown in the figure below. The required population may be calculated as 4 / (BER target), so for a BER target of 1e-6, a population of 4×10^6 is needed. This is based on a goal to have at least 10^6 waveform segments in each of the four symbol levels of the eye, for example:



In addition to the three contours, white horizontal and vertical annotation lines show the measurement points for H_{upp} and V_{upp} in the upper eye, H_{mid} and V_{mid} in the middle eye, and H_{low} and V_{low} in the lower eye. These measurement points are defined both in the IEEE 802.3bs specification and in OIF-CEI specifications such as 56G VSR.

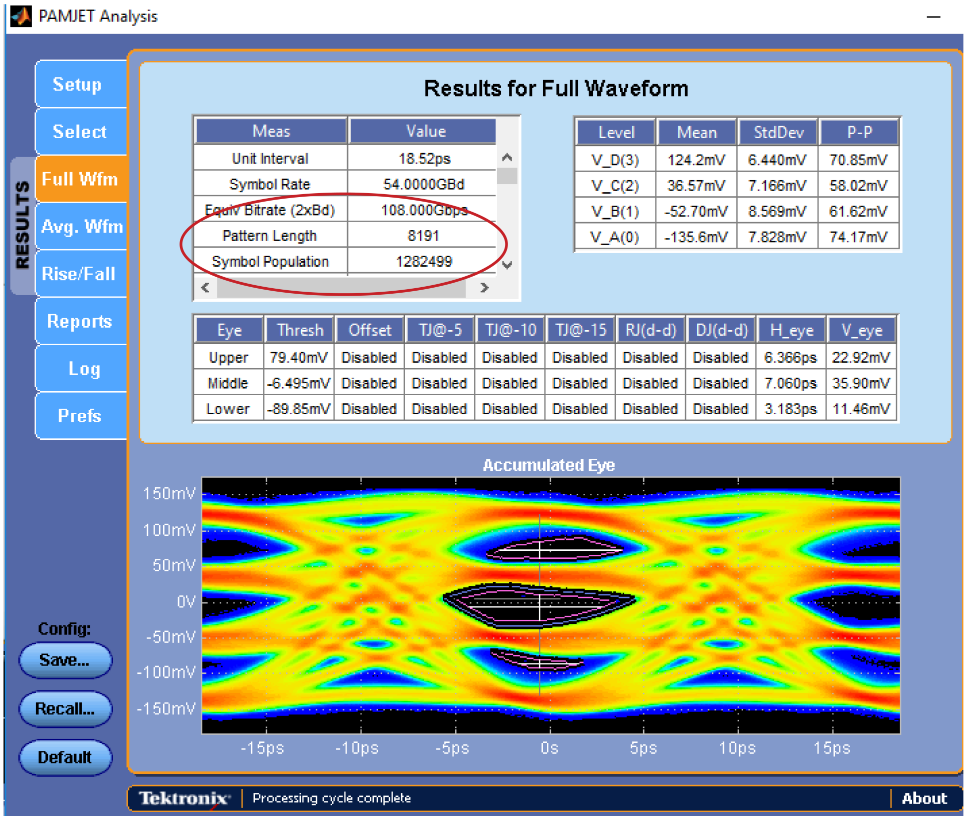
Error Navigator window

If the Error Navigator window is enabled (via a checkbox on the Preferences screen), it automatically appears when each Single processing cycle completes, assuming that at least one symbol error is detected in the corresponding waveform. The window does not appear in the following circumstances:

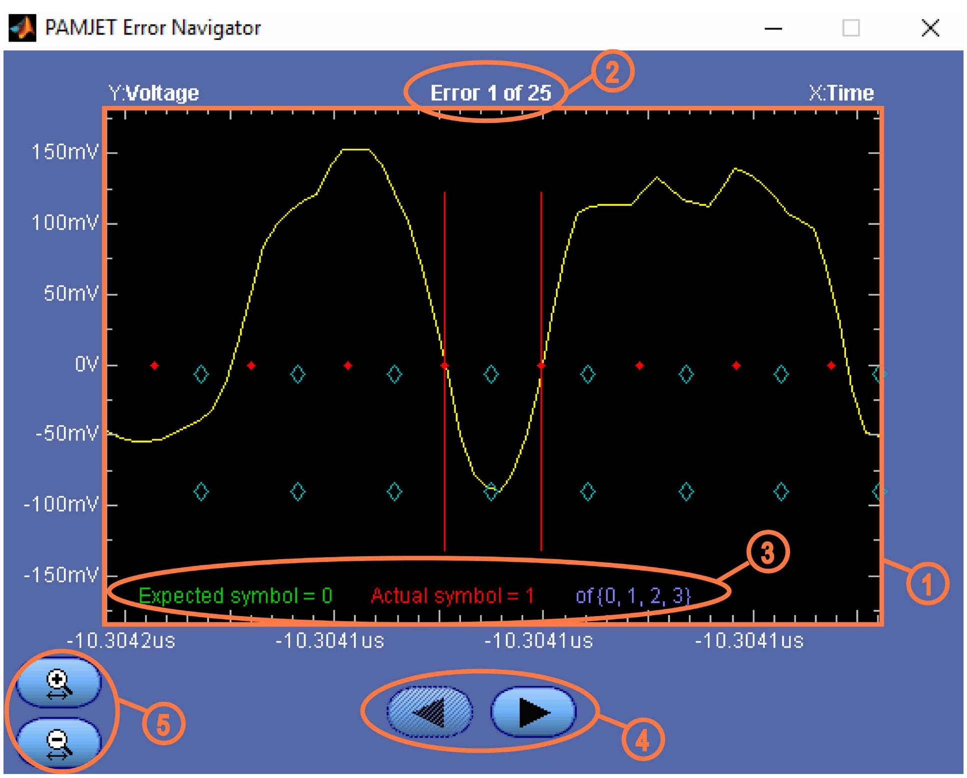
- There are zero pattern errors in the current waveform record
- The current waveform record does not contain a repeating pattern

If you have explicitly provided the expected symbol pattern (via the Data Rate & Pattern configuration on the Setup tab), the error detection uses that pattern as a reference. Otherwise, the error detection and navigation algorithm determines the reference pattern (that is, the sequence of symbols considered to be correct) based on what repeating sequence of symbols occurs most frequently in the waveform.

For the PAMJET waveform shown below, the center eye is nearly closed. A repeating pattern of 127 symbols has been detected, but there are 20 symbol errors among the 107,668 symbols, based on the chosen high, mid and low reference thresholds:



The Error Navigator, if enabled, would appear as shown in the following image. The components of the window are described below the image.



Error Navigator visual components:

1. The display grid initially shows eight unit intervals of the source waveform in yellow, with the location of one symbol error placed at the center of the screen. The sequence of red dots represents the recovered clock, which is nominally when the waveform switches between symbol levels. The two vertical red lines identify the two recovered clock times that bound the symbol error. The blue diamonds are placed vertically at the high, mid and low reference thresholds, and are placed horizontally at the timing phase corresponding to the eye center.
2. The text above the grid shows which symbol error is currently centered in the display, and the total number of symbol errors found for the current waveform. (If the error count exceeds 10,000, only the first 10,000 are shown.)
3. The text at the bottom of the grid shows what symbol was expected at the center of the screen (green text), what symbol actually occurred (red text), and what the set of valid symbols is (blue text).
4. The left and right arrow buttons allow navigation to prior or subsequent error locations.
5. The magnifying glass buttons allow the horizontal scale to be adjusted, so that more or less of the waveform context surrounding the symbol error can be viewed. The initial scale shows 8 unit intervals.

OIF-CEI jitter measurements: UUGJ, UBHPJ and Even-Odd Jitter

PAMJET Transmitter Analysis supports 3 transmitter output jitter measurements defined by the Optical Internetworking Forum (OIF) for Common Electrical Interface (CEI): UUGJ (uncorrelated unbounded Gaussian jitter), UBHPJ (uncorrelated bounded high probability jitter), and even-odd jitter. These measurements were defined in the draft OIF-CEI 56G MR specification as of June 2016 and was subsequently removed from the specification. They are kept in the current release, but are subject to removal in the future. **Users should use J_{rms}, J_{4u}, and even-odd jitter under IEEE Specific for the latest OIF-CEI jitter measurements.**

UUGJ and UBHPJ are calculated as follows:

1. Measure the jitter series at zero-crossing;
2. Create a CDF (cumulative distribution function) of the jitter series;
3. Based on the CDF, calculate J_5 as the time difference between (1-0.5×10^{-5}) and 0.5×10^{-5} probabilities, and J_6 as the time difference between (1-0.5×10^{-6}) and 0.5×10^{-6} probabilities;
4. Calculate UUGJ and UBHPJ as:

$$\begin{bmatrix} UUGJ \\ UBHPJ \end{bmatrix} = \begin{bmatrix} 1.0538 & -1.0538 \\ -9.3098 & 10.3098 \end{bmatrix} \begin{bmatrix} J_6 \\ J_5 \end{bmatrix}$$

Even-odd jitter is calculated as follows:

1. The time of a transition from 0 to 3, 3 to 0, 1 to 2, or 2 to 1 is the time at which the signal crosses the mid-point of V_{mid};
2. The time of a transition from 0 to 1 or 1 to 0 is the time at which the signal crosses the mid-point of V_{low};
3. The time of a transition from 2 to 3 or 3 to 2 is the time at which the signal crosses the mid-point of V_{upp};
4. The time of transitions from 0 to 2, or 2 to 0, is the time at which the signal crosses the mean value of the Level 1 signal in the central 0.05 UI of the eye;
5. The time of transitions from 1 to 3, or 3 to 1, is the time at which the signal crosses the mean value of the Level 2 signal in the central 0.05 UI of the eye;
6. Even-odd jitter is defined as the magnitude of the difference between the average deviation of all even-numbered transitions and the average deviation of all odd-numbered transitions, where determining if a transition is even or odd is based on possible transitions but only actual transitions are measured and averaged.

UUGJ and UBHPJ require a minimum symbol population of 10^7.

Even-odd jitter requires at least 2 pattern repetitions, and the availability of V_{mid}/V_{low}/V_{upp} results from Statistical Eye Analysis.

UUGJ, UBHPJ and even-odd jitter are included in the full waveform measurements table.

IEEE electrical measurements: J_{rms}, J_{4u}, J_{3u}, J_{3u03}, J_{4u04}, Even-Odd Jitter, Rise/ Fall Time and SNR_{ISI}

PAMJET Transmitter Analysis supports five transmitter electrical characterization measurements defined by IEEE 802.3bs: J_{rms}, J_{4u}, J_{3u}, even-odd jitter, rise/fall time and SNR_{ISI}. As of August 2017, J_{rms}, J_{4u}, and even-odd jitter are also defined in OIF-CEI 56G spec, replacing the removed UUGJ, UBHPJ and even-odd jitter definitions. Definitions of J_{rms}, J_{4u}, even-odd jitter, rise/fall time and SNR_{ISI} can be found in Annex 120D of IEEE 802.3bs spec, Draft 3.3 (as of July 2017). Due to their complexity, users are encouraged to refer to the latest spec draft for measurement definitions. Two additional jitter measurements, J_{3u03} and J_{4u03}, are defined in IEEE 802.3ck specification, Draft 3.2 (as of May 2022)

IEEE even-odd jitter measurements are only available on PRBS13Q and PRBS9Q pattern per spec definition. They are not computed on non-compliant patterns. The other measurements in this category may be computed on many non-compliant patterns, although PRBS13Q or PRBS9Q should be used for formal compliance testing purposes.

J_{3u} and J_{4u} are computed with a high-efficiency algorithm that uses jitter information in most of the pattern edges, while properly accounting for variations in DDJ, slew rate and scope noise conversion to jitter. J_{3u03} and J_{4u03} are calculated in the same way as J_{3u} and J_{4u} except that the jitter calculation uses only transitions R03 and F30. It can produce good results from relatively short waveforms but a symbol population of greater than $1x10^6$ is recommended. Because of the significant effect of scope vertical noise on jitter measurements, accurate results require methodically measuring and entering the actual scope noise, as described in *Measuring scope noise for electrical signals* on page 49.

Even-odd jitter requires at least two pattern repetitions.

SNR_{ISI} requires SNDR analysis to compute, as it is directly derived from the linear fit pulse response.

If selected, measurement results for J_{rms}, J_{4u}, J_{3u}, J_{3u03}, J_{4u03}, Even-Odd Jitter, Rise/Fall Time and $SNR_{ISI,}$ are included in the full waveform measurements table.

For further reading:

On the Tektronix website, navigate to Support > Learning Center and search for the White paper titled: "*A Fast, Stable Algorithm for Measuring Jrms, J4u and J3u on Real Time Oscilloscopes*"

IEEE electrical measurements: $V_{di(pk-pk)}$, VMA, dv_f and dR_{peak}

$V_{di(pk-pk)}$ is computed as the peak-to-peak differential voltage on an averaged/correlated waveform.

VMA is computed using high and low bits obtained from a long sequence of ones and zeros, respectively.

dv_f and dR_{peak} are computed as per the IEEE 802.3ck specification. Fixture s4p file must be given as input to these measurements which gets used for computation of reference device and package model.

PCIe Gen6 electrical measurements: $T_{TX-uTJ,}$ $T_{TX-RJ,}$ $T_{TX-uDJdd,}$ $T_{TX-UPW-TJ,}$ $T_{TX-UPW-DJDD,}$ JNu(PCIe), Jrms(PCIe), SNDR(PCIe), R_{LM}(PCIe), $V_{TX-DIFF-PP,}$ $V_{TX-EIEOS}$

$T_{TX-UPW-TJ}$, $T_{TX-UPW-DJDD}$ and $T_{TX-UPW-RJ}$ are measured on a toggle pattern. $T_{TX-UPW-RJ}$ is an informative measurement and is defined similar to T_{TX-RJ}. SNDR(PCIe), R_{LM}(PCIe), $V_{TX-DIFF-PP,}$ and $V_{TX-EIEOS}$ are measured on the PCIe Gen6 compliance pattern. The PCIe Gen6 Jitter measurement pattern (52 UI) is used for $T_{TX-uTJ,}$ $T_{TX-RJ,}$ $T_{TX-uDJdd,}$ JNu(PCIe), and Jrms(PCIe). JNu @BER 1e-6 (implying J6u) is identical to T_{TX-uTJ}. JNu(PCIe) and Jrms(PCIe) are for informational purposes only and allows for drawing comparisons with the IEEE counterparts. Definitions for these measurements can be found in PCI Express Base Spec Rev 6.0.

The jitter measurements will not be computed on non-compliant patterns. These jitter measurements are computed by using the jitter information present on all individual 48 edges present in the 52UI pattern (both edges in case of the toggle pattern), while accounting for variations in DDJ, slew rate and scope noise conversion to jiter. Due to the effect of scope vertical noise on the jitter of each edge, enter the methodically measured actual oscilloscope noise for the accurate results.

Note: If the oscilloscope noise value is too high, noise compensated jitter measurements will show warnings in the log for each edge whenever the jitter on that edge is being over compensated. This may result in artificially reduced jitter measurements. You will have to reduce the oscilloscope noise value if this happens.

If selected, measurement results for $T_{TX\text{-}uTJ,}$ $T_{TX\text{-}RJ,}$ $T_{TX\text{-}uDJdd,}$ $T_{TX\text{-}UPW\text{-}TJ,}$ $T_{TX\text{-}UPW\text{-}DJDD,}$ JNu(PCIe), Jrms(PCIe), SNDR(PCIe), R_{LM}(PCIe), $V_{TX\text{-}DIFF\text{-}PP,}$ and $V_{TX\text{-}EIEOS}$ are included in the full waveform measurements table.

PCIe Gen6 electrical measurements: Tx Preset Equalization

Tx Preset Equalization is measured on the 8768-symbol compliance pattern, and is a multiple-step process that requires acquisition of multiple waveform and the ability to control the DUT. The goal is to determine whether the FFE tap coefficients for each or all of the defined PCIe presets Q1 through Q10 are correct, by using preset Q0 as a reference for comparison.

Use the following steps to test the presets:

1. In PAMJET, set up the input source, clock recovery, BT filter, reference thresholds, and other configuration parameters as desired.
2. Click the checkbox to enable the Tx Preset Equalization test, found in the PCIe Specific node in the selection tree. (It is possible to enable other tests at the same time, but it is typically more efficient to run this test with the other measurements disabled.)
3. In this test's configuration panel to the right of the selection tree their is a drop-down box for setting the current Tx Preset. (If the right side panel is blank, click the Tx Preset Equalization measurement in the left side panel.) Set the Tx Preset to Q0 initially.
4. Configure the DUT for Q0 (or load the waveform corresponding to Q0, if using Ref waveforms). If using live waveforms, check that the scope's vertical scaling is appropriate.
5. Hit "Single" in PAMJET and wait for the measurement cycle to complete. This will record the Linear Fit Pulse Response (LFPR) for Q0 as a baseline.
6. On the Tx Preset Equalization measurement's configuration panel, set the drop-down for another preset (For Example, Q1).
7. Configure the DUT for that preset and check the vertical gain (or load the appropriate saved waveform).
8. Hit "Single".
9. When the measurement cycle completes, the Full Wfm Results table will include a results line with three values, such as: "-1.4dB, 5.0dB, 0.1dB". These are the Preshoot 2, Preshoot 1, and De-emphasis values respectively.
10. There should now be a file named "PCIeTxPresetEqReport.csv" in C:\Users\<username>\Tektronix\TekApplications\PAMJET\Reports. It will contain information for Q0 and Q1 (in this example). After you have run some more cycles, it will contain all presets that have been successfully run.
11. You can also create an HTML report in PAMJET. As with the csv, it will include all presets that have been run so far. (Once you have run Q0, you can run the others in any order or omit some.) Unlike the csv version of the result table, the HTML version also has a Pass/Fail column for each parameter.
12. In the most common use case, you would repeat *steps 6 – 8* once for each of presets Q1 – Q10, and then either produce an HTML report or save the csv file with all presets.

    If you uncheck the Tx Preset test and run a PAMJET measurement cycle, the Q0 response will be erased and you will have to start over. This is to protect against accidently using a Q0 baseline from some prior device.

Spread spectrum clocking measurements

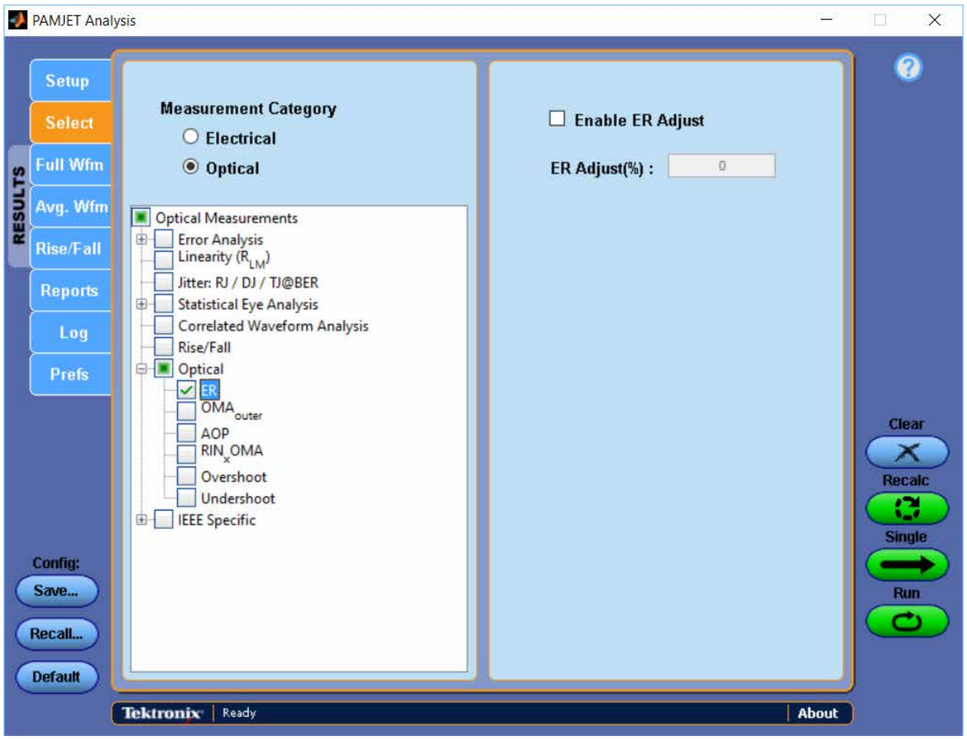
PAMJET application supports following spread spectrum clocking measurements on PAM4 modulated signal

- SSC Modulation Rate
- SSC Phase Deviation
- SSC min Frequency Deviation
- SSC Max Frequency Deviation

Results for these SSC measurements can be found in full measurement table.

SSC measurement results and SSC profile plot are available only when user selects SSC group of measurements in measurement tree and if recovered clock of the signal has SSC.

Optical measurements



PAMJET Transmitter Analysis supports the following generic optical measurements:

- (ER) (Extinction Ratio)
- OMA_{outer} (Outer Optical Modulation Amplitude)
- AOP (Average Optical Power)
- RINxOMA (Relative Intensity Noise)

In order to obtain the measurements, PAMJET Transmitter Analysis first computes mean power in the highest (P_3) and lowest (P_0) symbols of a PAMJET optical waveform, y:

$$ER = 10\log_{10}\frac{P_3}{P_0}(dB)$$

$$OMA_{outer} = P_3 - P_0(W)$$

$$AOP = 10\log_{10}\left(\frac{avg(y)}{0.001}\right)(dBm)$$

$$RIN_xOMA = \frac{\sigma_{ave}}{(P_3 - P_0)BW}$$

If selected, results for ER, OMA_{outer}, AOP, and RINxOMA are included in the full waveform measurements table.

ER adjust allows you to add or subtract a specified percentage from the measured ER value.

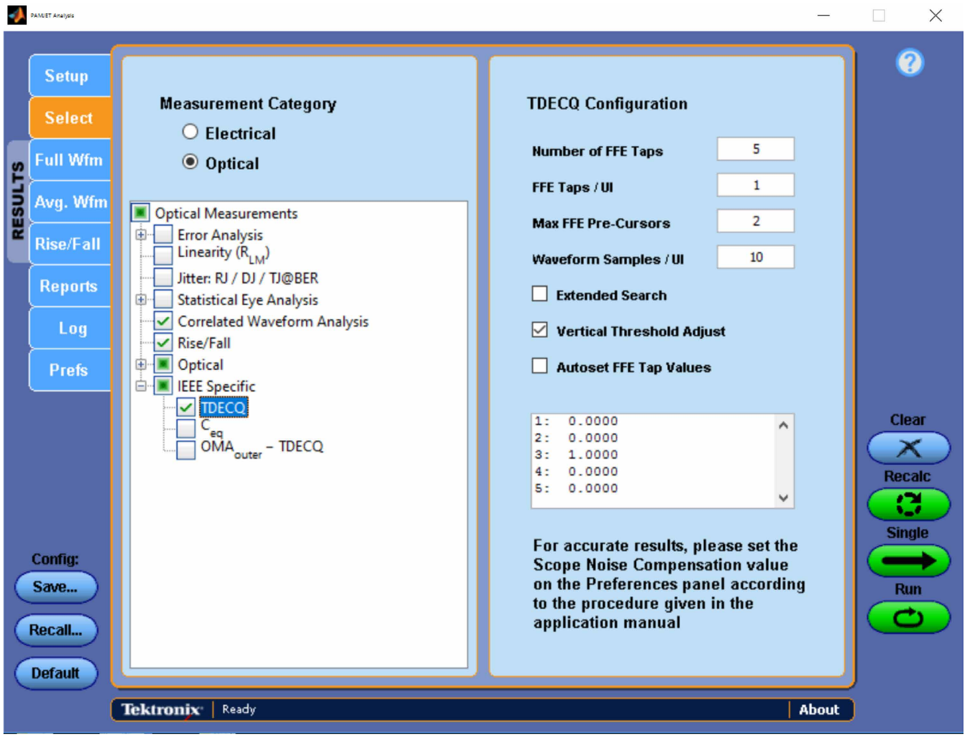
IEEE optical measurements

PAMJET Transmitter Analysis supports the following IEEE optical measurements:

Measurement	Description
TDECQ defined by IEEE 802.3bs and 802.3cd	The definition for TDECQ can be found in 121.8.5.3 of IEEE 802.3bs spec, Draft 3.5 (as of October 2017). TDECQ analysis finds the largest noise that can be combined with the signal by an ideal optical reference receiver when optimally equalized by a reference equalizer. The optimal equalizer tap coefficients are dependent on the amount of noise added to the signal, so finding the noise that can be added and the optical equalizer setting is an iterative process. $TDECQ = 10\log_{10}\left(\frac{OMA_{outer}}{6} \times \frac{1}{Q_t\sqrt{\sigma_G^2+\sigma_S^2}}\right)$
	TDECQ should be measured with an optical reference receiver (ORR) and reference equalizer as defined by the spec. As of 802.3bs Draft 3.5: • The ORR is a 4^{th}-order Bessel-Thomson filter with an optical bandwidth of 13.28125 GHz for 26.5625 GBd data rate. This can be achieved by enabling the Bessel-Thomson Filter on the Setup tab. Be careful not to duplicate ORR as optical probes may also come with their own ORR compensation. • The reference equalizer is a 5-tap, T-spaced, feed forward equalizer (FFE). In PAMJET Transmitter Analysis, the reference equalizer is included in TDECQ configuration and different from the stand-alone FFE on the Setup tab.
C_{eq}	Equalizer noise enhancement coefficient.
OMA_{OUTER}-TDECQ	This calculation uses the OMAouter measurement made after the TDECQ FFE.

Following configuration parameters require user input:

- Number of FFE Taps, FFE Taps/UI, and FFE Max Pre-Cursors: Reference equalizer parameters. For IEEE compliance, use a 5-tap, T-spaced FFE with a max pre-cursor of 2.
- Waveform Samples/UI: During TDECQ analysis, the input waveform is up-sampled to provide sufficient accuracy for the eye diagram. It is recommended to use 10 or above.
- Scope Noise (µWrms): σ_S, a standard deviation of the noise of the optical probe and oscilloscope combined. It should be measured with no optical input signal and with the same settings (vertical gain, filtering) as will be used to capture the optical signal.
- Extended Search: When enabled, the algorithms perform an extended search for the optimal FFE taps to minimize the TDECQ value. When disabled, the FFE adaptation will proceed more quickly and may be less optimum, which would result in a slightly higher TDECQ.
- Vertical Threshold Adjust: When enabled, the sub-eye threshold levels are allowed to adjust by ±1% of the OMAouter. IEEE 802.3cd allows this option to be selected for TDECQ measurements. When disabled, the sub-eye threshold levels are determined by the OMAouter and the average optical power.
- Autoset FFE Tap Values: When enabled, PAMJET Transmitter Analysis automatically determines the optimal FFE tap values and applies them to compute TDECQ. When disabled, the tap values entered on the graphical user interface are directly used to compute TDECQ.



If selected, the measurement is included in the full waveform measurements table.

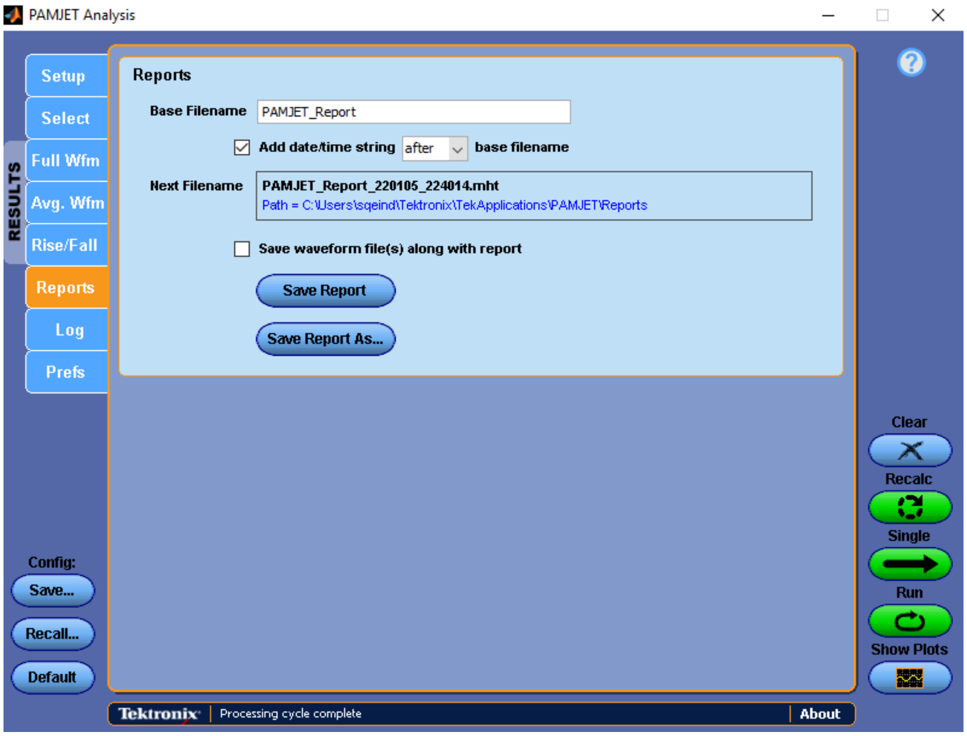
Saving results in a report

To save a report of the measurement results, click the Reports tab and use the Reports screen as follows:

1. Enter a Base Filename if desired. When the Save button is used along with the date/time option, this will allow a series of reports to be generated without supplying individual names.
2. To add the date and time to the base file name, check Add date/time string.
3. Select whether to add the date and time before or after the base file name.
4. Select **Save waveform file(s) along with report** to save the waveform along with the report. The waveform will be saved with the same name as the report but with channel name and number appended to it.

   Note: Waveform will be saved in the same folder where the report gets saved.

5. Click the Save Report button to save the report using the path and generated name shown in the Next Filename box.
6. If either the path or filename is unsuitable, click Save As... and use the conventional save dialog to change the path or name. The path selected will become the new default for subsequent Save operations.



The results are saved in a .mht file, which is an HTML format that combines both graphics and text markup into a single file for convenient sharing and archiving. The following figures show typical report content.

Note: The timestamp generated for the purposes of saving reports is the time when the analysis actually occurred rather than the time when the Save operation occurs.

PAM4 Analysis Report — Tektronix Enabling Innovation

C:\Users\Public\Tektronix\TekApplications\PAM4\PAM4_Report_171005_134624.mht 05-Oct-2017 13:47:23

▸ **Configuration**

Scope Configuration	
Scope Model	DPO77002SX
Scope Serial No.	B300221
Scope Version	10.6.0 Build 20
PAM4 Version	10.5.0.1
DPOJET Version	10.0.6.30

PAM4 Configuration	
Date Rate Mode	Auto
Pattern Control	Auto
Waveform Source	Ref1
Measurement Category	Electrical
Clock Recovery	
Method	PLL
PLL Model	Type 1
JTF BW	4.000MHz
Thresholds	Auto
High	234.2mV
Mid	-0.617mV
Low	-235.5mV
Waveform Filter	Disabled
Bessel-Thomson Filter	Enabled
Mode	Manual
Bandwidth (GHz)	33
CTLE	Disabled
FFE	Disabled
DFE	Disabled
Population Control	No Limit
Statistical Eye Analysis	Enabled
Target BER	1e-05
Halt on Closed Eye	Enabled
SNDR Measurement	Enabled
Samples/Sym. (M)	32
Length (Np)	200
Delay (Dp)	2
SNR_ISI	Enabled
Nb	10

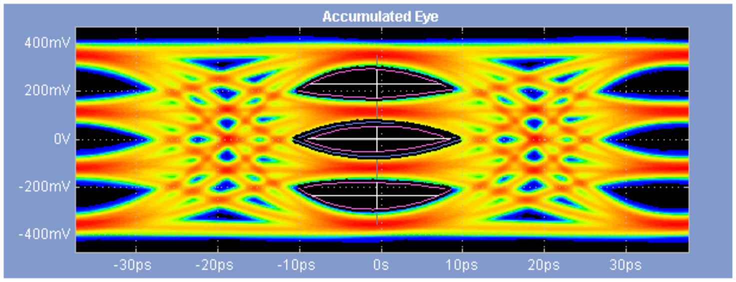
▸ **Full Waveform Results**

Measurement	Value
Unit Interval	---
Symbol Rate	---
Equiv Bitrate (2xBd)	---
Pattern Length	---
Symbol Population	---

Level	Mean	StdDev	P-P
V_D(3)	---	---	---
V_C(2)	---	---	---
V_B(1)	---	---	---
V_A(0)	---	---	---

Eye	Thresh	Offset	TJ@-5	TJ@-10	TJ@-15	RJ(d-d)	DJ(d-d)	Width	Height
Upper	---	---	---	---	---	---	---	---	---
Middle	---	---	---	---	---	---	---	---	---
Lower	---	---	---	---	---	---	---	---	---

- **Full Waveform Eye Diagram**

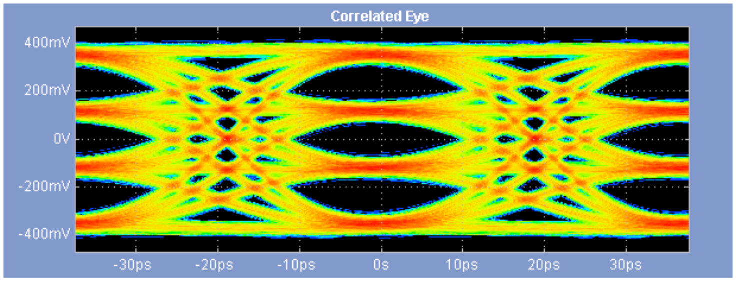


- **Correlated Waveform Results**

Measurement	Value
Level Deviation	1.007%
Level Thickness	4.359%
TimeDeviationOrig	2.679%
TimeDeviationMean	0.893%
P-P	779.7mV

Level	Time (e)	Amp (d)	Std Dev
3	-672.3fs	353.8mV	17.06mV
2	-1.345ps	116.8mV	17.00mV
1	-1.345ps	-118.1mV	16.96mV
0	-672.3fs	-355.3mV	16.95mV

- **Correlated Waveform Eye Diagram**

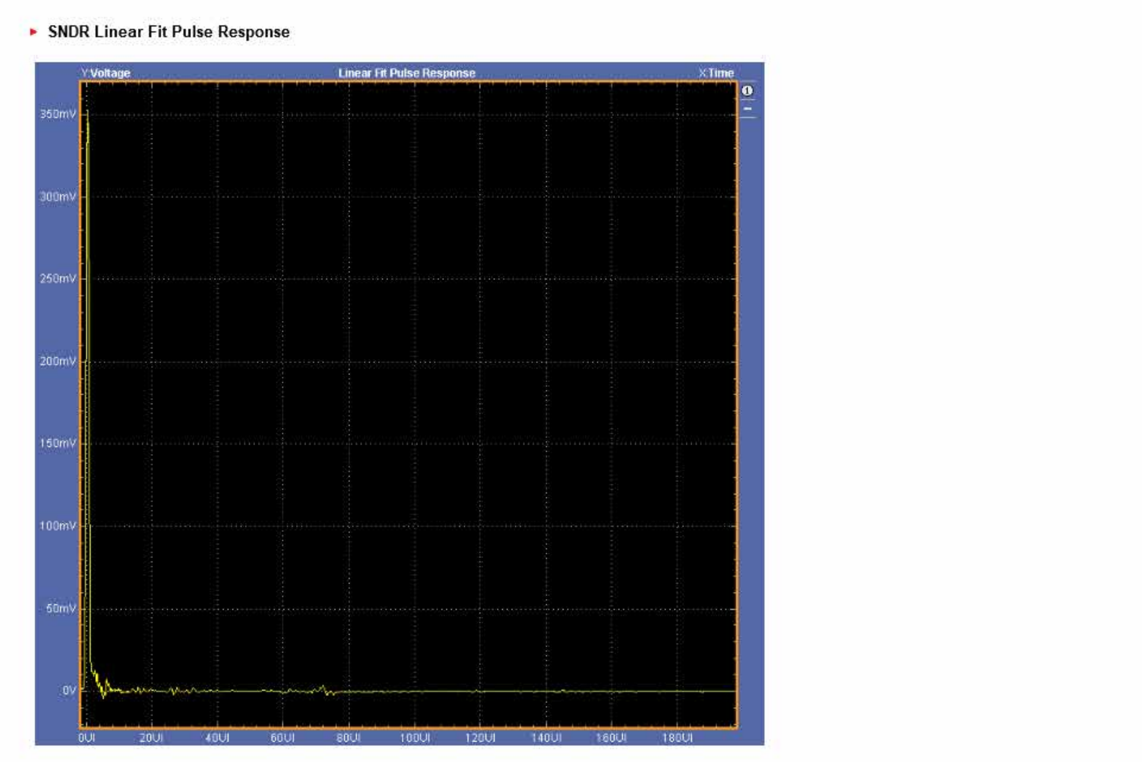


- **Rise Time Results**

Measurement	Mean	Max	Min	Count
Rise 0 ⇒ 1	---	---	11.1ps	512
Rise 1 ⇒ 2	14.1ps	19.4ps	10.6ps	512
Rise 2 ⇒ 3	---	---	11.3ps	512
Rise 0 ⇒ 2	14.4ps	16.6ps	12.5ps	512
Rise 1 ⇒ 3	14.6ps	17.0ps	12.7ps	512
Rise 0 ⇒ 3	14.2ps	15.8ps	13.0ps	512

- **Fall Time Results**

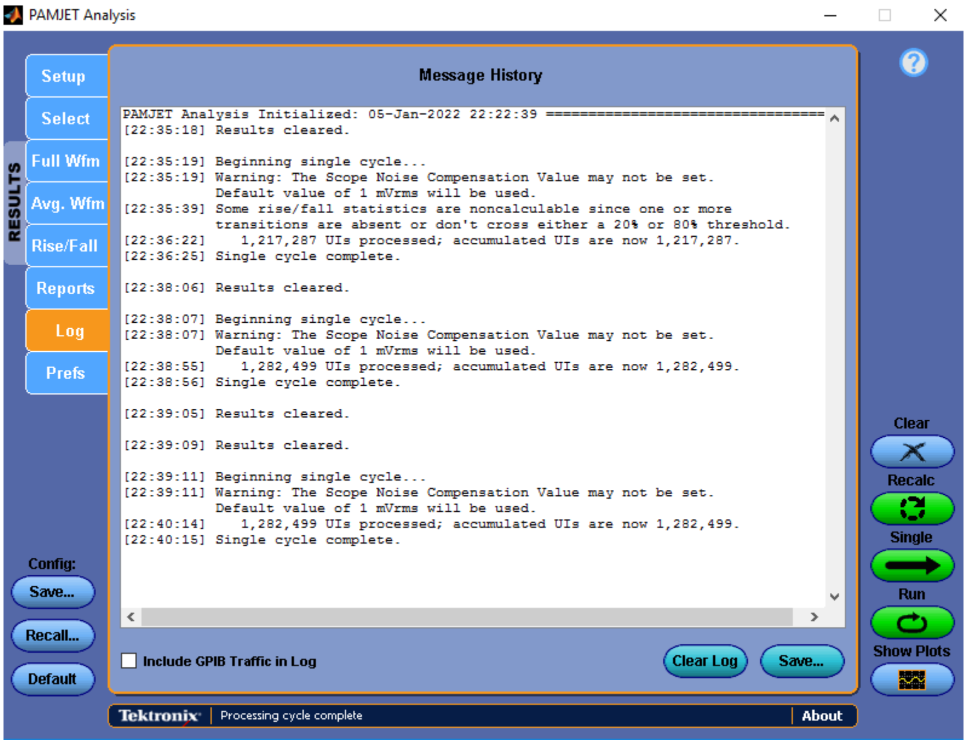
Measurement	Mean	Max	Min	Count
Fall 1 ⇒ 0	---	---	11.0ps	512
Fall 2 ⇒ 1	---	---	10.8ps	512
Fall 3 ⇒ 2	---	---	10.9ps	512
Fall 2 ⇒ 0	14.6ps	17.1ps	12.3ps	512
Fall 3 ⇒ 1	14.4ps	16.4ps	12.2ps	512
Fall 3 ⇒ 0	14.1ps	15.6ps	12.8ps	512



Note: When both SNDR and SNDR(PCIe) measurements are selected, only a single LFPR plot corresponding to the SNDR measurement will be displayed in the report. When either measurement is selected, the corresponding plot will be shown.

Using the event log

Selecting the Log tab displays information about events, such as when the results were last cleared, when the analysis started and stopped, how many unit intervals were accumulated, and any error or warning conditions. The event log should not be confused with the measurement log (streaming export of measurement results to a .csv file), which is controlled in the Prefs panel.



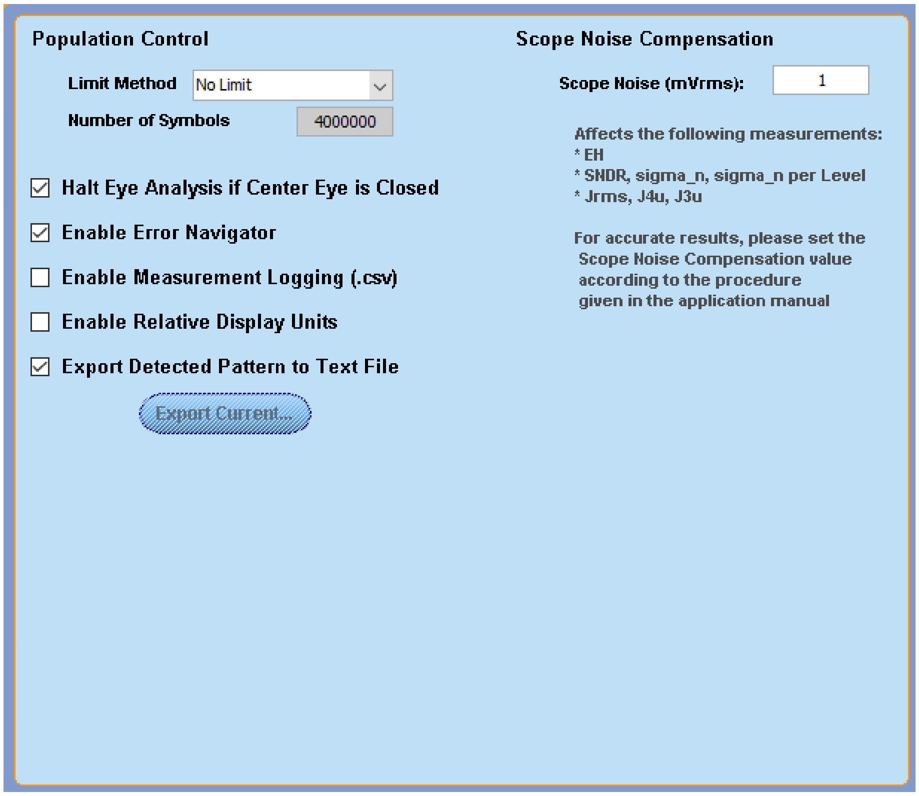
- Clear Log. Pressing this button clears the event log history.
- Save. Pressing this button allows you to save the event log history in a TXT file.
- Include GPIB Traffic in Log. When this box is selected, the event log records commands, queries, and responses that pass through the remote interface (GPIB). This may be helpful when you are developing remote-control scripts.

Preferences

Selecting the Prefs tab displays user preferences and some less-frequently-used configuration options, in seven categories:

- Population control
- Termination of Eye Analysis due to eye closure
- Error Navigator (for Symbol Error Analysis)
- Measurement logging
- Display units (Relative vs. Absolute)
- Export pattern to a text file
  - Export current
- Scope Noise Compensation (mVrms or µWrms)

Note: Setting the Scope Noise Compensation increases the accuracy of the PAMJET measurements are listed on the dialog screen. The measurements affected by noise compensation, and the units in which the noise is expressed, both depend on the measurement category selected in the **Select** tab. Refer *Measuring scope noise for electrical signals* on page 49 for a procedure to measure the noise value using the same configuration (vertical gain, de-embedding filter, Bessel-Thomson filter, etc) that will be used for the target signal.



Population control

The population control feature allows you to designate a target measurement population, either by the number of symbols or the number of waveform acquisitions (processing cycles). This is typically useful with the Run mode so that a population greater than is practical to obtain in a single cycle can be captured automatically. It is primarily designed to support some measurements defined by the OIF-CEI/IEEE standards that require a minimum symbol population. In many cases this is impractical to capture with a single waveform.

Limit Method: This control, in its default state of No Limit, disables the population limit feature and allows unconstrained sequencing. To enable population limits, select one of the other choices, either Number of Symbols or Number of Acquisitions.

Number of Symbols: When this method is selected via the Limit Method control, you can enter the target symbol population. If you subsequently select Run, the application will sequence until at least the designated population has been attained. (The application always acquires and processes full waveforms, so the total population when the application halts will generally be slightly greater than the target.)

Number of Acquisitions: When this method is selected via the Limit Method control, you can enter the number of waveform acquisitions (processing cycles) desired.

While the Population Control feature is mainly to support Run mode, you can also use Single to step toward the configured limit.

Halt on closed center eye

When the checkbox is selected, the Statistical Eye Analysis feature will cease trying to accumulate more population if the center eye is closed before the target BER has been achieved. The EW, EH and VEC measurements require that the eyes are still open at that population.

Error Navigator

The Error Navigator, when enabled, allows individual symbol errors in the current source waveform to be identified and viewed in a dedicated plot window. The Error Navigator window itself is described in *Error Navigator window* on page 33.

Whenever the PAMJET Transmitter Analysis application processes a waveform, it extracts the actual symbol sequence from the waveform as a part of the clock recovery process. The symbol sequence is analyzed to see if it contains a repeating pattern.

- **Error-Free Pattern**: In some cases the recovered symbol sequence will reflect a perfect repeating pattern, in which case the Error Navigator figure will not appear (since there are no errors to display).
- **Pattern with Errors**: In other cases, a few or many symbols will deviate from the pattern established by most of the waveform. In this case, the Error Navigator will appear when the processing cycle completes, allowing you to use forward and backward arrow buttons to browse through the located errors.
- **No Pattern Detected**: If no pattern can be discerned that would have fewer than 10% of the symbols in error (1e-1 SER), the waveform is identified as not having a pattern and the Error Navigator is not shown.
- **User-Determined Pattern**: If you use the Date Rate & Pattern control feature to specify an expected pattern via a text file, that pattern will take precedence over any auto-detection. Autocorrelation will be used to identify the most likely pattern starting point to match the specified pattern with the recovered symbols. Any deviations will be shown in the Error Navigator.

The Error Navigator figure is further described at *Error Navigator window* on page 33.

Measurement logging

Selecting the Enable Measurement Logging (.csv) checkbox causes all of the PAMJET measurement values from every processing cycle to be written to a file in comma-separated value (csv) format. This format can be opened by many spreadsheet applications such as Excel®, for example to allow post-processing. The output file is C:\Users\<username>\Tektronix\TekApplications\PAMJET\Log\PAMJET_Results.csv, and cannot be changed. The initial row in the file is a header row consisting of measurement names, and each processing cycle adds a row of values. Clearing results or changing any configuration parameter causes a blank line and a new header row to be written prior to the next measurement results row. The checkbox may be selected and de-selected during a measurement session to enable or disable logging, and the log file will remain after the application is closed. However, the log file is always cleared when the PAMJET Transmitter Analysis application is started, so be sure to copy or rename any results that you want to save before re-launching the application.

An example log, as viewed in a spreadsheet, is shown below. In this example, the application was cycled three times, cleared, and cycled once more.

Unit Inter	Symbol Ra	Equiv Bitr	Pattern Le	Symbol Pc	Symbol Er	Symbol Er	Bit Error R	<html>Lin	EW6	EH6	VEC	SND
31.25ps	32.00GBd	64.00Gbps	127	1253634	0	0	0	98.81%	insufficier	insufficier	insufficier	23.0
31.25ps	32.00GBd	64.00Gbps	127	2507268	0	0	0	98.81%	insufficier	insufficier	insufficier	23.0
31.25ps	32.00GBd	64.00Gbps	127	3760902	0	0	0	98.81%	insufficier	insufficier	insufficier	23.0
Unit Inter	Symbol Ra	Equiv Bitr	Pattern Le	Symbol Pc	Symbol Er	Symbol Er	Bit Error R	<html>Lin	EW6	EH6	VEC	SND
31.25ps	32.00GBd	64.00Gbps	127	1253634	0	0	0	98.81%	insufficier	insufficier	insufficier	23.0

Display units

For measurements of time parameters, the display unit can be switched between "Relative" and "Absolute". "Relative" displays time in unit intervals (UI), while "Absolute" displays time in seconds. Plots are not affected by this setting.

The default is "Absolute" display units.

Scope noise compensation

To ensure accurate measurement results, set the Scope Noise Compensation to the value measured in the procedure provided. See *Measuring scope noise for electrical signals* on page 49.

The noise compensation value is set in either mV or µW, depending on the Measurement Category selection (Electrical or Optical respectively, as set in the Select tab).

Export pattern to a text file

The **Export pattern to a text file** option allows you to export the detected pattern automatically during each processing cycle to a text file located at `C:\Users\<Current user>\Tektronix\TekApplications\PAMJET\Patterns\DetectedPattern.txt`

Note: In the subsequent processing cycle, the previously exported text file will be overwritten. The **Export current** option should be used to save the file with a different name or location.

If any SNDR measurement is enabled, the pattern used will be exported to a text file located at `C:\Users\<Current user>\Tektronix\TekApplications\PAMJET\Patterns\DetectedPatternPostEQ.txt`

Note: The pattern in DetectedPatternPostEQ.txt can be different from the one in DetectedPattern.txt when the eye is closed. This is because a silent attempt is made to open the eye in order to extract the most accurate pattern for computing the linear fit pulse response.

Export current

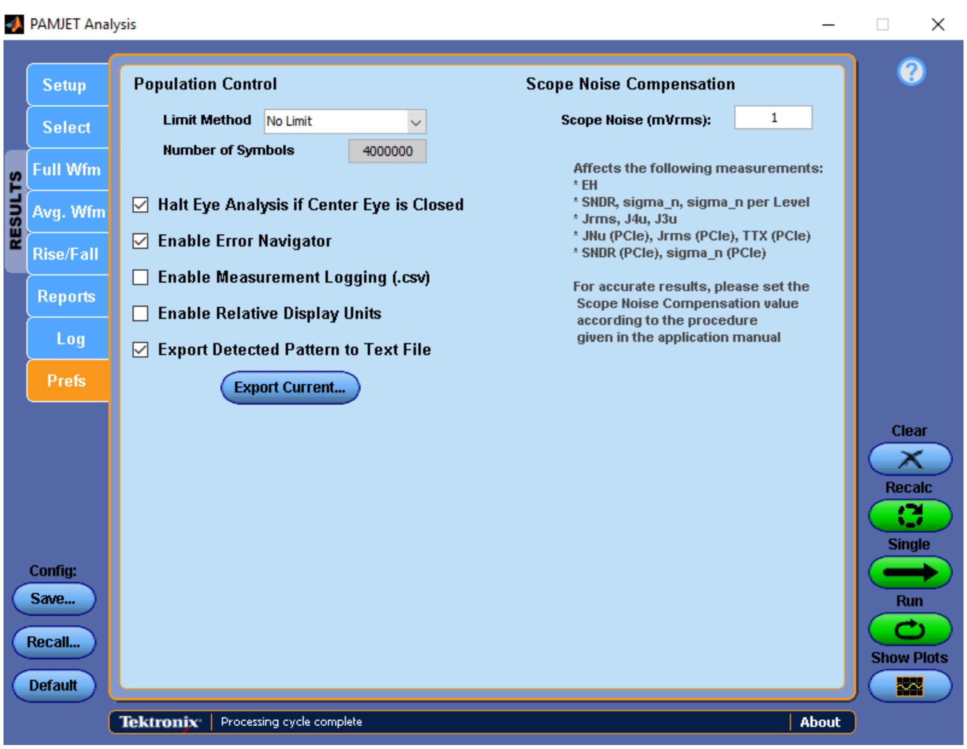
The **Export current** option allows you to export the presently detected pattern into a text file with the name and specified path.

Note: This option can only be used once the analysis is complete and pattern data is available to export.

Scope noise characterization

Scope noise, σ_S is defined as the standard deviation of the noise of the probe and oscilloscope combined. Many of the optical and electrical measurements require that the scope noise be measured and then entered into the Prefs tab.

This section provides instructions on how to characterize the scope noise for either electrical or optical signals. The measured value is then entered into the Scope Noise Compensation field.



Measuring scope noise for electrical signals

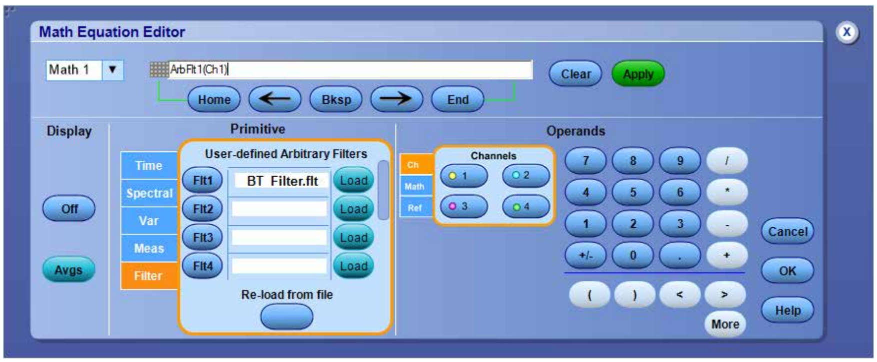
Some electrical measurements require the scope noise value (to be entered on the Prefs tab) for most accurate results.

The following procedure should be used to characterize the noise characterization steps:

1. Connect the live signal to one of the available channels.
2. Adjust the vertical scale such that the signal occupies at least 80% of vertical range, without clipping.
3. If the Bessel-Thomson filter available in the PAMJET Transmitter Analysis application is used, then this filter needs to be applied to the live signal using Math->Arbflt.

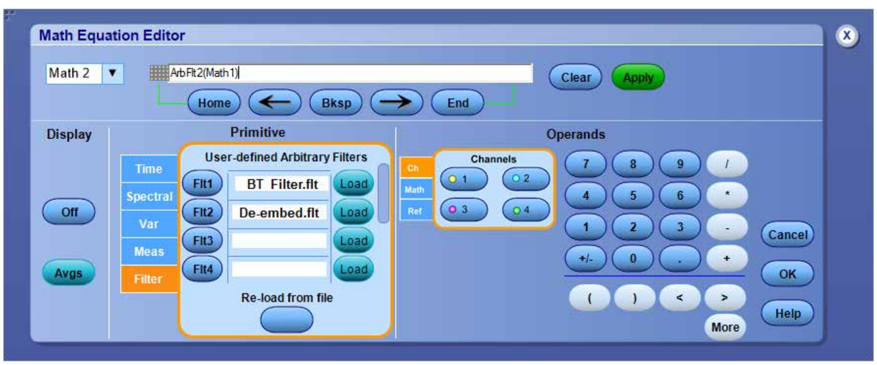
   Let the output of this filter be Math1.

   (If you have sequenced the application at least once in the current session with its Bessel-Thomson filter enabled, the created filter file can be found in the path specified. Refer to the *Receive filter* on page 20 section for more information.

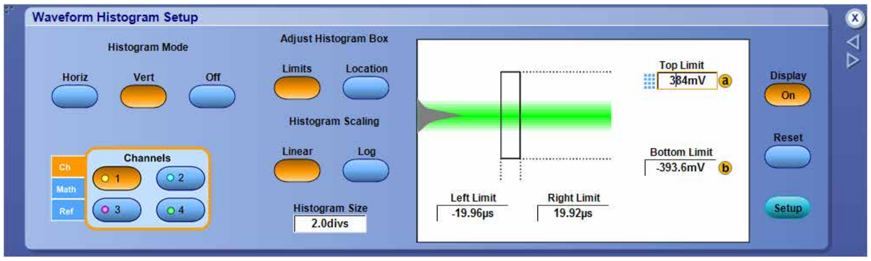


4. If cable embedding/de-embedding is used, this filter needs to be applied similarly as in Step 3. ”

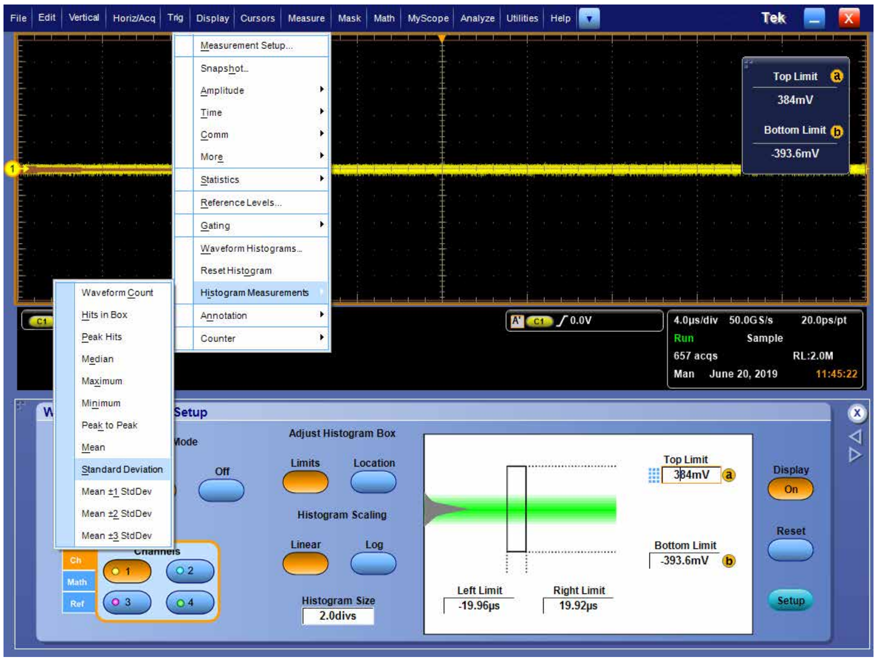
   Let the output of this filter be Math2.



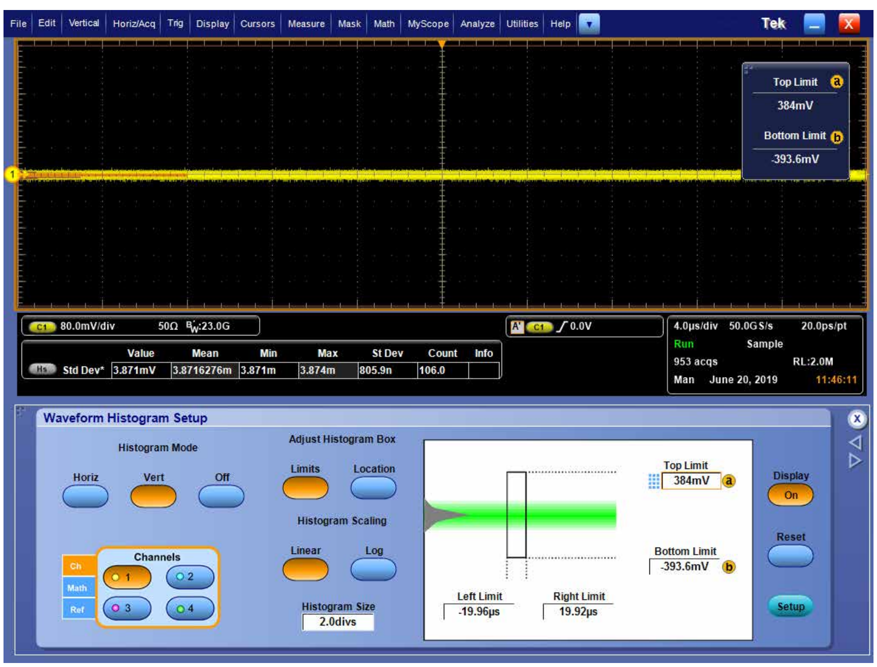
5. Disconnect the live signal from the scope.
6. Identify the source to be used for noise characterization according to the setup:
   a. When no cable embedding/de-embedding and Bessel-Thomson filter are used: Choose the channel connected to the intended live signal.
   b. When both cable embedding/de-embedding and Bessel-Thomson filter are used: Choose Math2.
   c. When only Bessel-Thomson filter is used: Choose Math1.
   d. When only cable embedding/de-embedding is used: Choose Math2.
7. On the TekScope menu, go to Measure>Waveform Histograms.
8. Set Histogram Mode to Vert. Set the source to the channel chosen in Step 6.
9. Set the Left, Right, Top and Bottom limits to cover the entire graticule.



10. On the TekScope menu, select Measure>Histogram Measurements>Standard Deviation.



11. “Hs Std Dev” is now be visible in the Measurement badge – The Mean value of this is the scope noise RMS value.



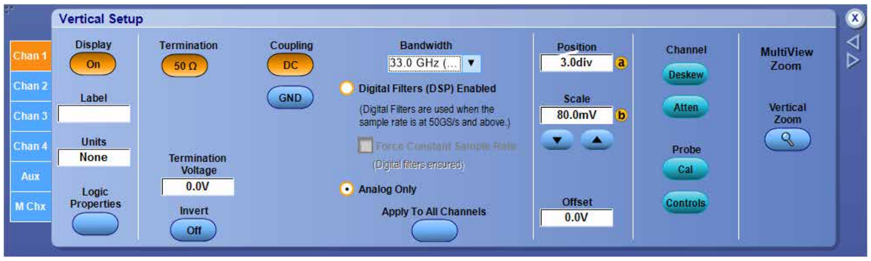
12. The scope noise RMS value should be measured for several different vertical positions using Steps 9 to 11. Calculate the average of these values to obtain the final scope noise RMS value.

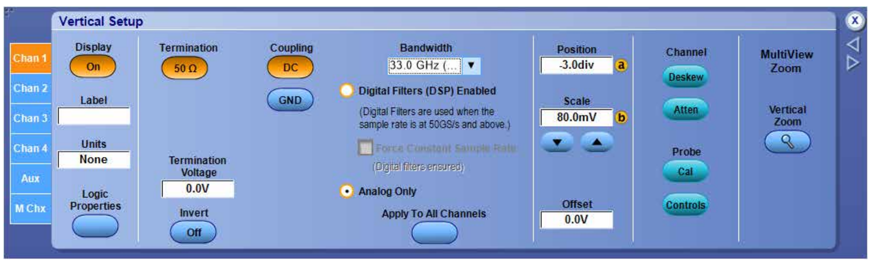
Following the following steps based on the setup:

a. Single-ended input:

i. On the TekScope menu, go to the Vertical menu for the live channel connected to the intended live signal.

ii. Set the Position control to several different values across the screen, e.g., -3 div, 0 div and +3 div.





iii. Measure the noise RMS value for each position using steps 9 to 11.

iv. Calculate the average of the different readings as the final scope noise RMS value.

b. Differential inputs:

i. Suppose Ch1 and Ch3 are used, and are combined using a Math expression (possibly combined with de-embedding). Go to Vertical menu for Ch1 and set the Position to a value, e.g., as +3 div. Go to Ch3 and set the position to the inverse value (-3 div in this example).

ii. Now measure the scope noise RMS using Steps 9 to 11 on the corresponding Math channel.

iii. Repeat Steps 12.b.i and 12.b.ii with similar combinations, e.g., {-3 div, +3 div}, {0 div, 0 div}, etc.

iv. Calculate the average of the different readings as the final scope noise RMS value.

Measuring scope noise for optical signals

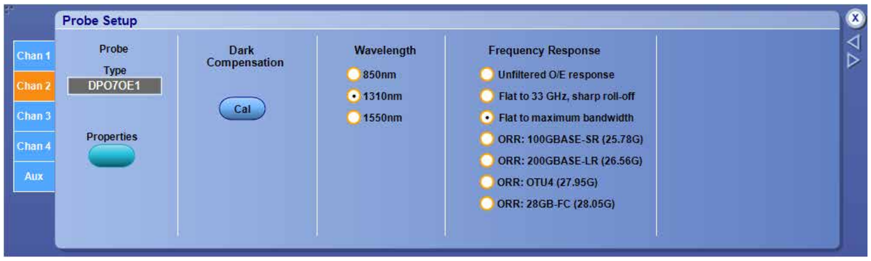
The following optical measurements require the scope noise value (to be entered on the Prefs tab) for most accurate results.

- EH
- RIN_xOMA
- TDECQ
- OMA-TDECQ

Optical measurement usually requires the application of an optical reference receiver ORR filter. For IEEE compliance, the ORR is a 4^{th}-order Bessel-Thomson filter with a bandwidth of 0.5 of the actual symbol rate (IEEE 8023bs).

1. Connect the live signal to one of the available channels.

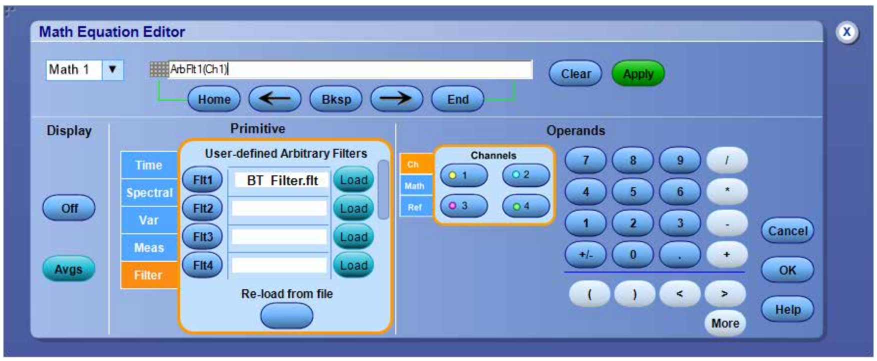
2. Configure the O/E probe to ensure it uses Flat to maximum bandwidth frequency response.



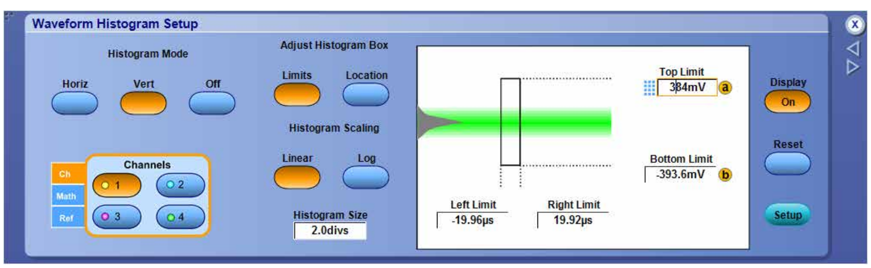
3. Adjust the vertical scale such that the signal occupies the full vertical range.
4. If Bessel-Thomson filter is used, then this filter needs to be applied to the live signal using Math->Arbflt.

   Let the output of this filter be Math1.

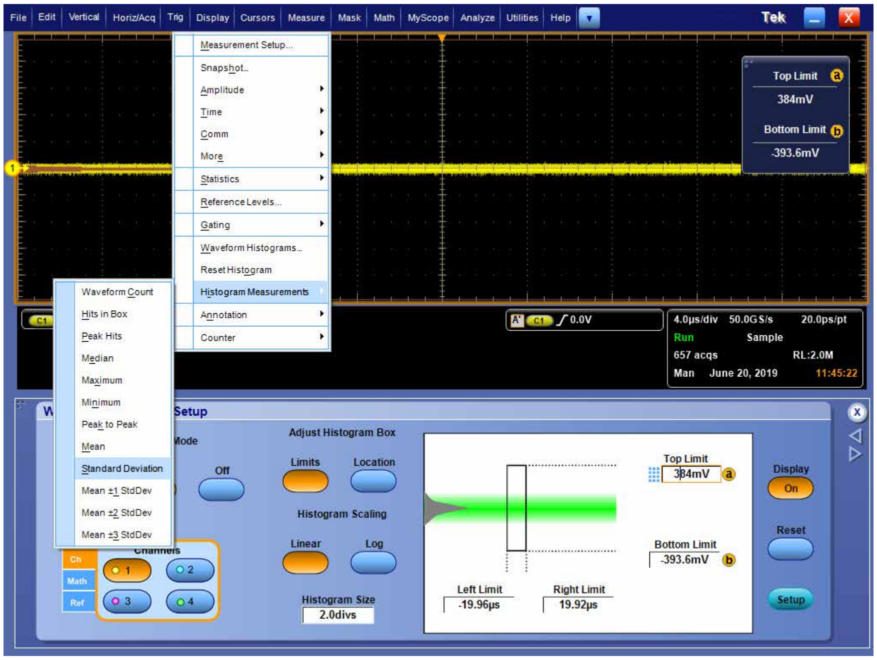
   (If you have sequenced the application at least once in the current session with its Bessel-Thomson filter enabled, the created filter file can be found in the path specified. Refer to the *Receive filter* on page 20 section for more information.



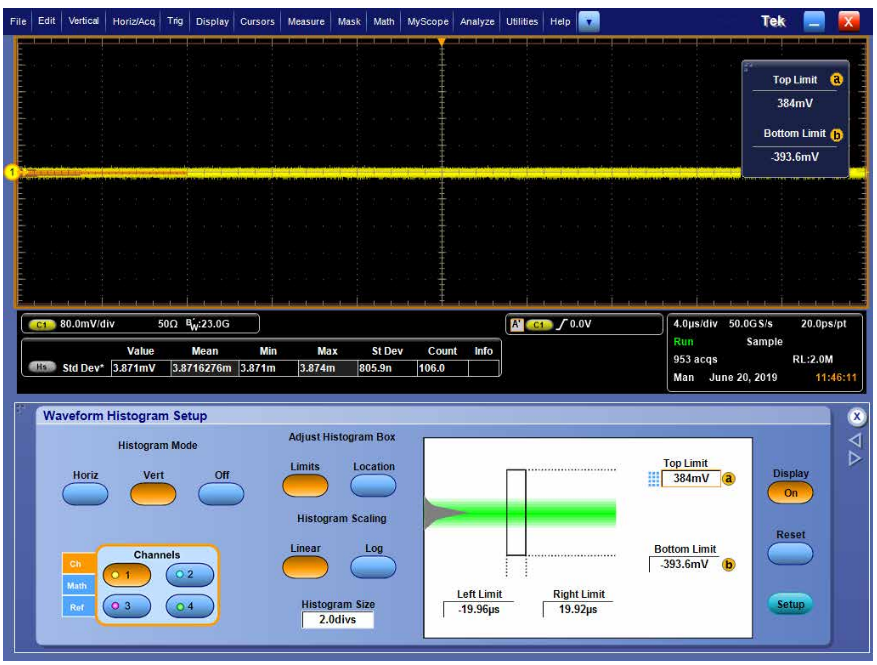
5. Disconnect the live signal from the scope.
6. Switch to Math1.
7. On the TekScope menu, go to Measure>Waveform Histograms.
8. Set Histogram Mode to Vert. Set the source to the channel selected in Step 6.
9. Set the Left, Right, Top and Bottom limits to cover the entire graticule.



10. On the TekScope menu, select Measure>Histogram Measurements>Standard Deviation.



11. “Hs Std Dev” is now be visible in the Measurement badge – The Mean value of this is the scope noise RMS value.



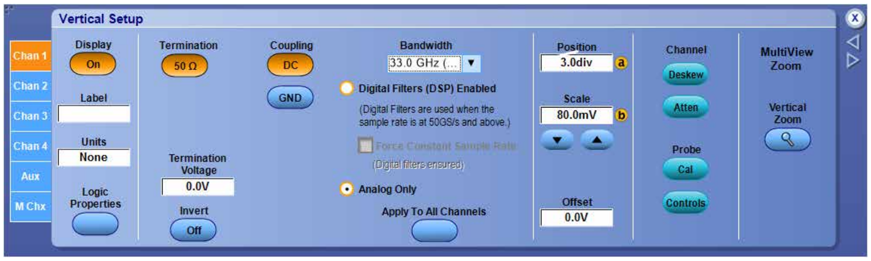
12. The scope noise RMS value should be measured for several different vertical positions using Steps 9 to 11. Calculate the average of these values to obtain the final scope noise RMS value.

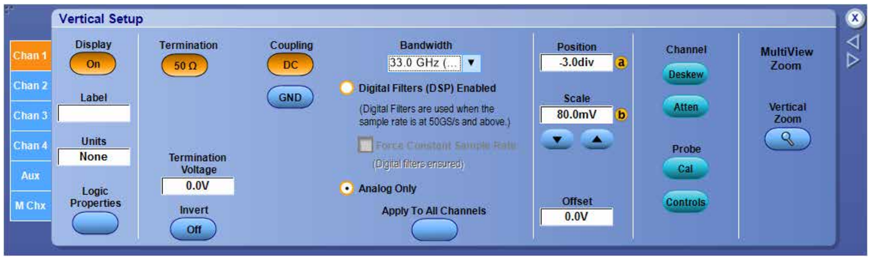
Following the following steps based on the setup:

a. Single-ended input:

i. On the TekScope menu, go to Vertical menu for the live channel connected to the intended live signal.

ii. Set the Position control to several different values across the screen, e.g., -3 div, 0 div and +3 div.





iii. Measure the noise RMS value for each position using steps 9 to 11.

iv. Calculate the average of the different readings as the final scope noise RMS value.

b. Differential inputs:

i. Suppose Ch1 and Ch3 are used, and are combined using a Math expression (possibly combined with de-embedding). Go to Vertical menu for Ch1 and set the Position to a value, e.g., as +3 div. Go to Ch3 and set the position to the inverse value (-3 div in this example).

ii. Now measure the scope noise RMS using Steps 9 to 11 on the corresponding Math channel.

iii. Repeat Steps 12.b.i and 12.b.ii with similar combinations, e.g., {-3 div, +3 div}, {0 div, 0 div}, etc.

iv. Calculate the average of the different readings as the final scope noise RMS value.

Measurements

This section describes measurements used by the PAMJET software.

General full waveform measurements

Measurement	Description
Unit interval	The mean duration of a PAMJET symbol (corresponding to two bits of information)
Correlated waveform analysis	Correlated waveform analysis
Symbol rate	The number of symbols transmitted per second, equal to the reciprocal of the Unit Interval. A symbol is also known as a 'baud', so the conventional units are GBd or GigaBaud.
Equivalent bit rate	The number of bits transmitted per second. Since there are two bits in each symbol, this is numerically twice the symbol rate.
Pattern length	If the signal bears a repeating pattern of symbols, this measurement gives the number of symbols in a single repeat.
Symbol population	The total number of symbols accumulated since the results were last cleared.
Symbol errors	The number of detected symbol errors accumulated since the results were last cleared.
SER (actual)	Actual (not estimated) Symbol Error Ratio: The ratio of accumulated erroneous symbols to total accumulated symbols since the results were last cleared.
BER (actual)	Actual (not estimated) Bit Error Ratio: The ratio of accumulated erroneous bits to total accumulated bits since the results were last cleared.
Linearity (R_{LM})	This measurement is defined in IEEE specification 802.3bj, and is a measure of how close the four symbol levels are to being equally spaced in the vertical dimension. Perfectly even spacing corresponds to a value of 100%. To be compliant, this measurement must be made on a specific 160-bit pattern. If the acquired waveform contains the required pattern, the measurement is made according to the specification. If the waveform contains any other pattern, this measurement uses an adaptation of the formal definition by using samples from the center of each and every symbol period, regardless of the pattern. The measurement is computed as: $R_{LM} = \frac{6 * S_{min}}{V_D - V_A}$ Where: $S_{min} = \frac{\min (V_D - V_C, V_C - V_B, V_B - V_A)}{2}$ And V_A - V_D are as described below.
EW	The Eye Width at BER as defined by CEI-56G-VSR-PAMJET (oif2014.230 draft 10, dated 17-Feb-2017), assuming that the Statistical Eye Analysis BER Target is left at the default, specification-compliant value of 1e-5. If the Statistical Eye Analysis BER Target is changed to another value B, then the measurement name will dynamically change to EW<-log $_{10}$ (B)>. For example, if the BER target is changed to 1e-6, the name will change to EW6.

Table continued…

Measurement	Description
EH	The Eye Height at BER, after compensation for scope noise (*Refer note*). If you configure the BER to 1e-6, the measurement corresponds to EH6 as defined by CEI-56G-VSRPAMJET(oif2014.230 draft 10, dated 17-Feb-2017). See EW, above, for a description of how the measurement's name automatically adapts as the Statistical Eye Analysis BER target is changed. **Note:** Scope noise compensation is performed by post-processing the set of amplitude samples formed by taking the horizontal center of each symbol, using these steps: • Remove deterministic data-dependent noise from each sample • Use a spectral approach to identify and remove periodic amplitude components not correlated to the pattern • Use a qscale-based approach to separate any nonperiodic bounded noise that may be present • Quantify the sigma value for the remaining Gaussian noise • Subtract from this sigma the scope noise sigma (using root-sum-squares subtraction) • Option to consider the noise due to reference receiver as defined in the IEEE 802.3 Annex 120G.5. If this option is enabled, then the sigma of the reference receiver is added to the sigma computed above. • Options are available to enable windowing techniques while computing EH. By default, it is set to "None" to consider only the vertical probability density function (PDF) at the eye center. However, different window weighting functions can also be defined, where a set of vertical PDFs around the eye center configured by the window width is considered. For "Rectangular" window, the vertical PDFs around the eye center are weighted uniformly. For "Gaussian" window, the vertical PDFs around the eye center are weighted using a gaussian function as defined in the IEEE 802.3 Annex 120G.5. • Calculate the deviation in the sigma value based on the methods. Then, use the Q-factor and BER relationship to compute the deviation in the EH at the prescribed BER.
VEC	The Vertical Eye Closure as defined by CEI-56G-VSR-PAMJET (oif2014.230 draft 10, dated 17-Feb-2017). The value is only compliant if the Statistical Eye Analysis BER Target is 1e-5, which is the default value.
SNDR	Signal-to-noise-and-distortion ratio, defined by: $SNDR = 10\log_{10}\left(\frac{p_{max}^{2}}{\sigma_e^{2}+\sigma_n^{2}}\right) dB$
p_{max}	Peak amplitude of the linear fit pulse response.
σ_e	Linear fit error, computed as RMS error between measured signal and linear fit.
σ_n	Vertical noise, computed as an average of RMS deviation from mean voltage at a fixed point in a run of 8 ones and 8 zeroes.
UUGJ	Uncorrelated unbounded Gaussian jitter as defined by CEI-56G-MR-PAMJET (OIF2014.245 draft 7, dated 27-June-2016). Requires a minimal symbol population of 10^7. This measurement is accumulated until the results are cleared. For more details, read *OIF-CEI jitter measurements: UUGJ, UBHPJ and Even-Odd Jitter* on page 35.

Table continued…

Measurement	Description
UBHPJ	Uncorrelated bounded high probability jitter as defined by CEI-56G-MR-PAMJET (OIF2014.245 draft 7, dated 27-June-2016). Requires a minimal symbol population of 10^7. This measurement is accumulated until the results are cleared. For more details, read *OIF-CEI jitter measurements: UUGJ, UBHPJ and Even-Odd Jitter* on page 35.
Even-odd jitter	As defined by CEI-56G-MR-PAMJET (OIF2014.245 draft 7, dated 27-June-2016). Requires the availability of Statistical Eye Analysis results, which meets the Statistical Eye Analysis BER target). As of Aug 2017, this measurement is obsolete and removed from the OIF-CEI spec. For more details, read *OIF-CEI jitter measurements: UUGJ, UBHPJ and Even-Odd Jitter* on page 35
J_{rms}	Implementation of the measurement defined in IEEE 802.3bs (draft 3.5, dated 10-Oct-2017). Requires the use of PRBS13Q compliant pattern. For more details, read *IEEE electrical measurements: Jrms, J4u, J3u, J3u03, J4u04, Even-Odd Jitter, Rise/Fall Time and SNRISI* on page 36.
J_{4u}	Implementation of the measurement defined in IEEE 802.3bs (draft 3.5, dated 10-Oct-2017). Requires the use of PRBS13Q compliant pattern. For more details, read *IEEE electrical measurements: Jrms, J4u, J3u, J3u03, J4u04, Even-Odd Jitter, Rise/Fall Time and SNRISI* on page 36.
J_{3u}	Implementation of the measurement defined in IEEE P802.3cd (draft 3.5, dated 18-September-2018), which refers to IEEE 802.3bs (draft 3.5, dated 10-October-2017). For more details, read *IEEE electrical measurements: Jrms, J4u, J3u, J3u03, J4u04, Even-Odd Jitter, Rise/Fall Time and SNRISI* on page 36.
EOJ (IEEE) per edge	EOJ measurements for each of the twelve possible transitions. This allows for understanding the contribution of EOJ by each of the twelve signal transitions. For more details, read *IEEE electrical measurements: Jrms, J4u, J3u, J3u03, J4u04, Even-Odd Jitter, Rise/Fall Time and SNRISI* on page 36. Requires the use of PRBS13Q compliant pattern.
Sigma N	Vertical noise, defined by $\sigma_n = \frac{(\sigma_{n(0)}+\sigma_{n(1)}+\sigma_{n(2)}+\sigma_{n(3)})}{4}$ Vertical noise for PAMJET level k (k ∈ {0, 1, 2, 3}), computed as the RMS deviation from mean voltage at a fixed point in a consecutive run of eight k symbols. This measurement is compensated for the scope's random vertical noise by computing the compensated value as the square root of [(uncompensated $\sigma_{n(k)})^2$ – (scope rms noise)^2] .
Even-odd jitter (IEEE)	As defined by IEEE 802.3bs (draft 3.3, dated 28-July-2017). Requires the use of PRBS13Q compliant pattern. For more details, read *IEEE electrical measurements: Jrms, J4u, J3u, J3u03, J4u04, Even-Odd Jitter, Rise/Fall Time and SNRISI* on page 36.

Table continued…

Measurement	Description
Rise/Fall time	As defined by IEEE 802.3bs (draft 3.3, dated 28-July-2017). Requires the use of PRBS13Q compliant pattern. For more details, read *IEEE electrical measurements: Jrms, J4u, J3u, J3u03, J4u04, Even-Odd Jitter, Rise/Fall Time and SNRISI* on page 36.
SNR_{ISI}	As defined by IEEE 802.3bs (draft 3.3, dated 28-July-2017). Since this measurement depends on the availability of SNDR results, the SNDR measurement or one of its subcomponents must also be selected. For more details, read *IEEE electrical measurements: Jrms, J4u, J3u, J3u03, J4u04, Even-Odd Jitter, Rise/Fall Time and SNRISI* on page 36.
$J3u_{03}$	Implementation of the measurement is defined in IEEE P802.3ck. For more details read *IEEE electrical measurements: Jrms, J4u, J3u, Even-Odd Jitter, Rise/Fall Time and SNRISI*
$J4u_{03}$	Implementation of the measurement is defined in IEEE P802.3ck. For more details read *IEEE electrical measurements: Jrms, J4u, J3u, Even-Odd Jitter, Rise/Fall Time and SNRISI*
$T_{TX\text{-}uTJ}$	Uncorrelated total jitter. Implementation of the measurement is defined in PCI Express Base Spec.
$T_{TX\text{-}RJ}$	Random jitter. Implementation of the measurement is defined in PCI Express Base Spec.
$T_{TX\text{-}uDJdd}$	Uncorrelated deterministic jitter. Implementation of the measurement is defined in PCI Express Base Spec.
$T_{TX\text{-}UPW\text{-}TJ}$	Uncorrelated Total Pulse Width Jitter. Implementation of the measurement is defined in PCI Express Base
$T_{TX\text{-}UPW\text{-}RJ}$	Uncorrelated random pulse width jitter that corresponds to $T_{TX\text{-}UPW\text{-}DJDD}$ and $T_{TX\text{-}UPW\text{-}TJ}$
$T_{TX\text{-}UPW\text{-}DJDD}$	Uncorrelated Deterministic Pulse Width Jitter. Implementation of the measurement is defined in PCI Express Base Spec
JNu(PCIe)	Uncorrelated total jitter @ BER 1e-N
Jrms(PCIe)	Standard deviation of the uncorrelated total jitter
SNDR(PCIe)	Signal-to-Noise-Distortion ratio. Implementation of the measurement is defined in PCI Express Base Spec.
R_{LM}(PCIe)	Level Separation Mismatch Ratio for PCIe. Implementation of the measurement is defined in PCI Express Base Spec .
$V_{TX\text{-}DIFF\text{-}PP}$	Differential peak-peak Tx voltage swing for full swing operation. Implementation of the measurement is defined in PCI Express Base Spec.
$V_{TX\text{-}EIEOS}$	Minimum voltage swing during EIEOS. Implementation of the measurement is defined in PCI Express Base Spec .

Table continued…

Measurement	Description
Tx Preset Equalization	Preshoot 2, Preshoot 1, and De-emphasis for the current PCIe Gen6 Tx Equalization Preset, reported in dB.
ER	The ratio of the highest and lowest optical power levels of a PAMJET optical signal, reported in dB. For more details, read *Optical measurements* on page 38.
OMA_{outer}	The difference between the highest and lowest optical power levels of a PAMJET optical signal, reported in W. For more details, read *Optical measurements* on page 38.
AOP	The average power of a PAMJET optical signal, reported in dBm. For more details, read *Optical measurements* on page 38.
TDECQ	As defined by IEEE 802.3bs (draft 3.5, dated 10-October-2017). For more details, read *IEEE optical measurements* on page 39.
SSC Modulation Rate	Modulating frequency of spread spectrumclock. It is the rate at which clock frequency changes
SSC Phase Deviation	Phase jitter associated with SSC modulation
SSC Min Frequency Deviation	Minimum frequency shift in SSC profile as a function of time. It represents the frequency deviation in terms of ppm
SSC Max Frequency Deviation	Maximum frequency shift in SSC profile as a function of time. It represents the frequency deviation in terms of ppm
V_{di}(pk-pk)	Differential peak to peak voltage measured on correlated waveform
dv_f	Difference steady state voltage dv_f is defined mathematically as $dv_f = v_f^{(meas)} - v_f^{(ref)}$ where $v_f^{(meas)}$ is steady state voltage of linear fit pulse got from acquired waveform $v_f^{(ref)}$ is steady state voltage of linear fit pulse got after passing it through reference transmitter and package models.

Table continued…

Measurement	Description
dR_{peak}	Difference pulse peak ratio is defined mathematically as $$dR_{peak} = R_{peak}^{(meas)} - R_{peak}^{(ref)}$$ Where $R_{peak}^{(meas)}$ is the pulse peak ratio of linear fit pulse got from acquired waveform and it is mathematically defined as, $$R_{peak}^{(meas)} = \frac{v_{peak}^{(meas)}}{v_f^{(meas)}}$$ $R_{peak}^{(ref)}$ is the pulse peak ratio of linear fit pulse got after passing it through reference transmitter and package models and it is mathematically defined as, $$R_{peak}^{(ref)} = \frac{v_{peak}^{(ref)}}{v_f^{(ref)}}$$

CTLE Optimization

The CTLE Optimization is designed to quickly sweep through all the CTLE presets of the selected CTLE family and identify the preset that maximizes the eye height on the current waveform. The user has the option to select any available CTLE family and can also combine CTLE optimization with a LFPR-based DFE (if available).

Currently the LFPR-based DFE methods supported are:

- PCIe Gen6
- IEEE 802.3ck

⚠ **Note:** The optimum CTLE that maximizes eye height will be identified but will not be automatically used for any currently selected measurements. The user will have to record the CTLE preset that was selected > **disable CTLE** Optimization in the Select menu > **enable CTLE** in the Setup menu with the optimal CTLE Preset and then proceed with running an analysis cycle.

The estimated eye height(s) computed during the CTLE Optimization will be saved in the report as a table. The picked CTLE will be mentioned in the results and will be marked with an asterisk in the report table. These estimated eye heights may not representative of actual eye height results and should only be used to pick the optimal CTLE preset.

⚠ **Note:** Sometimes a negative estimated eye height will be reported because we assume a worst-case noise estimate. This may also mean the eye is closed. The negative value is still significant as it informs the user of the relative performance of the CTLE presets. The relatively best performing CTLE in this case can still yield an open eye in a regular analysis cycle.

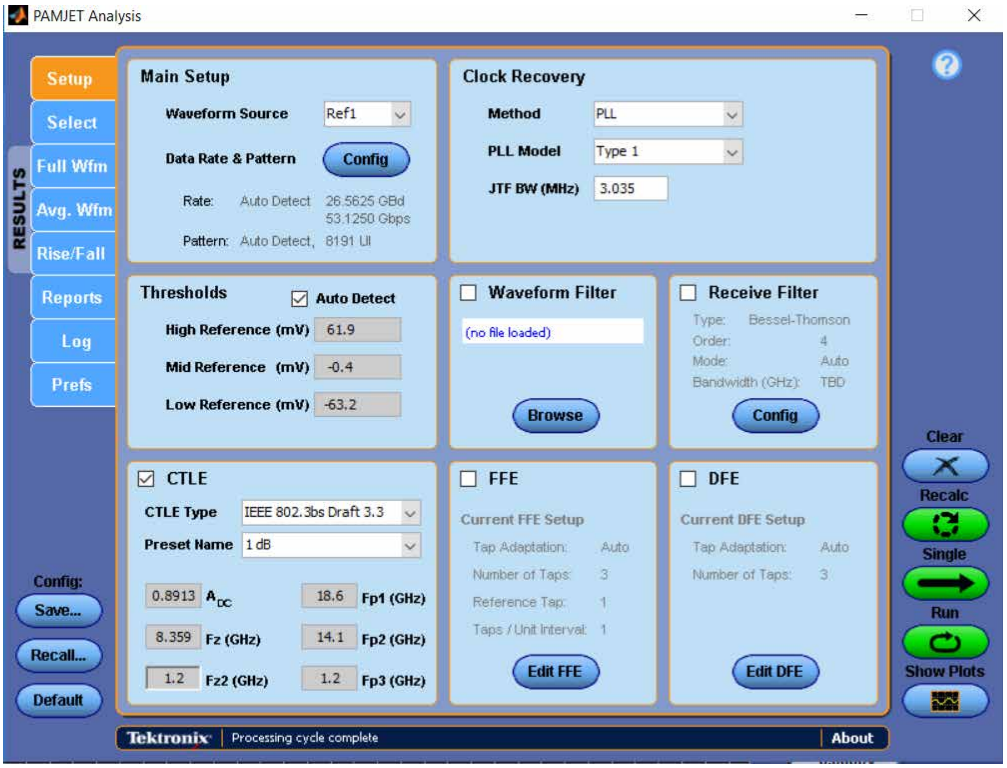
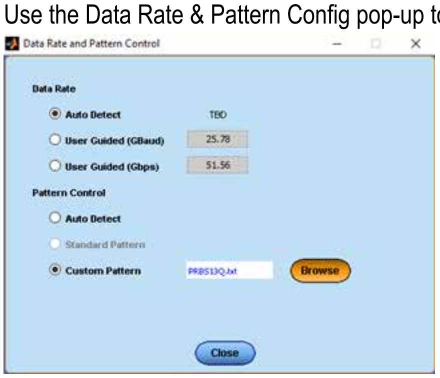


Figure 1: CTLE Optimization Setup

Before performing CTLE optimization, configure the Setup panel as follows:

1. Use the Data Rate & Pattern Config pop-up to enable Pattern Control > **Custom Pattern**, with the proper pattern file selected.

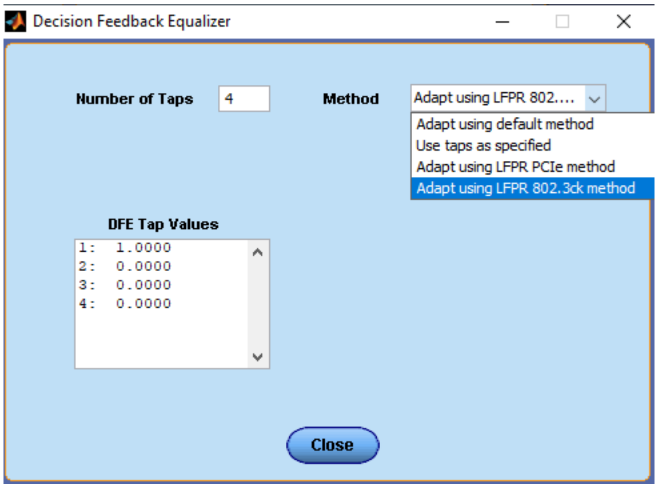


Note: The pattern file is not required but the lack of an accurate pattern can lead to a sub-optimal preset being picked.

2. Use the **CTLE Type** drop-down to pick the desired CTLE family. Leaving the **CTLE Type** as Custom will lead to the CTLE Optimization being skipped.
3. De-select the CTLE checkbox.

Note: CTLE may not turned on when CTLE optimization is selected. PAMJET will not allows you to proceed with an analysis cycle when both are selected at the same time.

4. If the LFPR-based DFE adaptation is available, use the Edit DFE pop-up menu to select it and set the number of DFE taps [Optional] .



5. De-select the DFE checkbox (not required but leads to faster processing and faster result reporting.)

Measurements per level in the PAMJET eye

Measurement	Description
V_D(3)	For the top symbol level, this is the voltage at the center of the unit interval across all instances of that symbol in the complete waveform. The mean, standard deviation and peak-to-peak range are shown. Only the mean value is used in the computation of R_{LM} .
V_C(2)	Same as V_D(3), except for the second level from the top.
V_B(1)	Same as V_D(3), except for the second level from the bottom.
V_A(0)	Same as V_D(3), except for the bottom level.

Measurements per eye opening (upper, middle, lower) in the PAMJET eye

Measurement	Description
Threshold	The reference voltage level at which the TJ, RJ, and DJ measurements for that level are made. It may be a value calculated automatically by the PAMJET application or a manually-entered value.
Offset	The horizontal offset, in time, from the center of the PAMJET eye (calculated as ½ of a symbol interval from the recovered clock edges) to the measured center of the eye as rendered, at the corresponding reference voltage.
TJ@-N	The total jitter at a selected Bit Error Rate (BER). This measurement is typically signified as TJ@BER or TJ@1E-12 (for example), but due to space constraints is it shown as TJ@-12 in the graphical user interface's results table. There are three TJ columns, and although the BER exponents default to -5, -10 and -15, these values are configurable to any value between -1 and -18 on the user preferences tab. The measurements are made at the reference voltage for the corresponding level.
RJ(d-d)	The random jitter, according to the dual-dirac jitter model, at the corresponding reference voltage.
DJ(d-d)	The deterministic jitter, according to the dual-dirac jitter model, at the corresponding reference voltage.
Width	The actual eye width of the rendered eye, at the corresponding reference voltage. This column is only displayed if the Noise Analysis feature (Prefs tab) is disabled.

Table continued…

Measurement	Description
Height	The actual eye height of the rendered eye at the center of the unit interval, for the corresponding (upper, middle or lower) eye. This column is only displayed if the Noise Analysis feature (Prefs tab) is disabled.
H_eye	The horizontal eye opening as defined by CEI-56G-VSR-PAMJET (oif2014.230 draft 7, dated 10-June-2016). The value will represent either Hupp, Hmid or Hlow for the upper, middle or lower eye respectively. This column is only displayed if the Noise Analysis feature (Prefs tab) is enabled.
V_eye	The vertical eye opening as defined by CEI-56G-VSR-PAMJET (oif2014.230 draft 7, dated 10-June-2016). The value will represent either Vupp, Vmid or Vlow for the upper, middle or lower eye respectively. This column is only displayed if the Noise Analysis feature (Prefs tab) is enabled.

Measurements for the correlated waveform

Measurement	Description
Correlated (averaged) waveform	Please see the Glossary
Level deviation	A measure of the deviation of the vertical intervals between levels from perfectly equal spacing, where 0% represents perfect spacing. The measurement is computed as: $$\mathrm{Level\,Deviation}=\frac{1}{3}\sum_{ij=10,21,32}\frac{\left\| d_{ij}-\frac{(Pk-PK)}{3}\right\|}{\frac{(Pk-Pk)}{3}}\times 100\%$$ Where d_{ij} is d_i - d_j, and d_i is the amplitude of level i as described in the per-level table below.
Level thickness	An overall measure of the vertical thickness of the symbol levels in the correlated waveform, where an ideal signal with maximally open eyes would have a thickness of 0%. The measurement is computed as: $$\mathrm{Level\,Thickness}=\frac{1}{4}\sum_{i=0}^{3}\frac{\sigma_i}{(Pk-Pk)\div 2}\times 100\%$$ Where σ_i is the standard deviation of level i as defined in the table below.
Time deviation (origin)	A measure of how well-centered the four symbol levels are with respect to the nominal center of the eye, in time. The measurement is calculated as: $$\mathrm{Time\,Deviation}-\frac{1}{4}\sum_{i=0}^{3}\frac{\|e_i\|}{UI}\times 100\%$$ Where e_i is the horizontal offset from the center of the UI as defined in the table below and UI is the nominal Unit Interval.
Time deviation (mean)	The Time Deviation (Origin) is somewhat sensitive to the exact choice of the center of the symbol, which depends on details of the clock recovery. Because of this, a slight change in the clock recovery position can inflate the time deviation result even if the three individual offsets e_i are nearly equal to each other. The time deviation (mean) removes the mean value of the three e_i values before computing the time deviation according to the equation above. This yields a measurement that is unbiased by precise clock recovery position.

Table continued…

Measurement	Description
Peak-Peak	The signal amplitude between a nominal level 3 and a nominal level 0. The nominal level for level N is defined as follows: The longest sequence adjacent bits of level N are found in the correlated waveform, and the nominal level is the level in the center of that run of bits.

Measurements per level in the correlated PAMJET eye

Measurement	Description
Minimum ISI point	For each level (0 - 3) in the correlated eye, there exists a horizontal position across the unit interval where the standard deviation of the waveform voltage is minimized, so that the level is thinnest at this point. (In theory there may be more than one point for each level that matches this description, but in practice that is seldom a problem.)
Time (e)	For each level, the horizontal offset from the Minimum ISI Point to the nominal center of the eye.
Amplitude (d)	For each level, the mean voltage at the Minimum ISI Point
Standard deviation (σ)	For each level, the standard deviation at the Minimum ISI Point

Measurements for rise times and fall times

Measurement	Description
0->3	A set of three measurements of the rise time of the correlated waveform, from the 0 (lowest) level to the 3 (highest) level. From top to bottom the three measurements within the set are Maximum, Mean, and Minimum. The measurements are from the 20% to the 80% point of the nominal symbol levels shown in the Amp (d) column of the right-hand table on the Results for Correlated Waveform tab. The other five possible rising transitions within the PAMJET eye are similarly named. For example, the group labeled 1->2 measures the rise times from the 1 level to the 2 level.
3->0	A set of three measurements of the fall time of the correlated waveform, from the 3 (highest) level to the 0 (lowest) level. The measurements are from the 80% to the 20% level. See the 0->3 measurement for additional details.
Minimum	For a typical test pattern, there are several or many transitions in each pattern-repeat that make a given transition type, such as 0->3. The minimum rise or fall time represents the fastest such event. If a given transition type is not represented in the repeating pattern, the minimum is shown as a series of dashes, "---".
Maximum	The maximum value for a given transition type is the slowest rise or fall time across all the transitions of this type. In cases of marginal signal quality, there may be some transitions of a given type that never reach the 20% or 80% value. In this case, the maximum is considered undefined and the value is displayed as three dashes, "---".
Mean	The Mean value for a given transition type is the average rise or fall time across all the transitions of this type. If one or more transitions of a given type never reach the 20% or 80% value, it is not possible to calculate a mean value across the population. In this case the mean value is displayed as three dashes, "---".
Count	For a given transition type, such as 0->3, the count is the number of times that transition appears in one repetition of the pattern. If a transition type doesn't appear at all in a pattern, the count is 0. The sum of the counts from the 12 transition types may be equal to the pattern length, but it is frequently less. This occurs because wherever the pattern has two or more adjacent symbols of the same value, there is no transition.

GPIB command reference

You can use GPIB commands to control the application remotely, and to monitor the application activity using the application menus. The GPIB commands listed are separate from, and handled differently than, the DPO70000 Series Oscilloscope GPIB commands. (Note that a TekScope command, APPLICATION:ACTIVATE “PAM Analysis (PAMJET)”, is used to start the application.) Press here to see the set of available commands (*GPIB command reference* on page 65).

Handshaking protocol

The application handles GPIB communications through its own protocol handshaking.

The requirements for GPIB communications with a controller are as follows:

1. Once the application has started, it writes an “OK” status to the application handshake variable. This tells the controller that it may now write a valid commands into the “pam4” variable.
2. The GPIB controller polls the handshake variable (VARIABLE:VALUE? “pam4”) until it detects the OK status.
3. The GPIB controller writes a command string into the application handshake variable. For example, sending the command ‘VARIABLE:VALUE “pam4”, “single”’ writes the string “single” into the variable “pam4”.
4. The application GPIB function polls the handshake variable, reads the command string and interprets it as a command. If the command is not understood, it writes an ERROR handshake value to the variable.
5. A good command is parsed and executed. On successful execution, the application writes an OK to the handshake variable. When the GPIB controller reads the OK status, it may send a new command string.

Best practice when programming using GPIB commands

To help users get started with using GPIB commands to operate PAMJET Transmitter Analysis, an example of automatic testing script written in Python 3.5 is included at the end of this chapter. Although Python is selected here for its readiness to support instrument control, most programming languages can be used.

To establish sufficient timing for PAMJET Transmitter Analysis and TekVISA to correctly transmit, receive and process GPIB commands, it is recommended to insert 0.5 to 1 second delay between consecutive commands during scripting.

You are also advised to routinely query PAMJET status before executing additional commands, especially while running time-consuming tasks, such as performing a single processing cycle. Delay or loss of commands can happen if one is sent while PAMJET Transmitter Analysis is busy executing a previous command.

GPIB commands

APPLICATION:ACTIVATE "PAM Analysis (PAMJET)"

APPLICATION:ACTIVATE "PAM Analysis (PAMJET)"	This command instructs the oscilloscope to start the PAM4 application. It is a set-only parameter. (Note that this is a TekScope command, not a PAM4 command, and that the format and capitalization of the quoted string must be exactly as shown).
Syntax	APPLICATION:ACTIVATE "PAM Analysis (PAMJET)"
Returns	None

VARIABLE:VALUE "pam4","version?"

VARIABLE:VALUE "pam4","version?"	Instructs the PAM4 application to report its version number in response to a subsequent "VARIABLE:VALUE? "pam4"" query.
Syntax	VARIABLE:VALUE "pam4","version?" followed by query

	VARIABLE:VALUE? "pam4"
Returns	Version number of the currently installer PAM4 Transmitter Analysis. E.g., "OK:10.0.1.1".

VARIABLE:VALUE "pam4","exit"

VARIABLE:VALUE "pam4","exit"	Closes the PAM4 application. The current state of the application is not saved.
Syntax	VARIABLE:VALUE "pam4","exit"
Arguments	"exit" forces the application to close.

VARIABLE:VALUE? "pam4"

VARIABLE:VALUE? "pam4"	Reads the value of the PAM4 handshake variable. If the handshake variable begins with "OK" or "ERROR", the PAM4 application is ready to receive another command. Any other value means that PAM4 has not finished processing a prior command.
Syntax	VARIABLE:VALUE? "pam4"
Arguments	None
Returns	OK[:<message>] The PAM4 application is ready for a command.
	ERROR[:<message>] The PAM4 application encountered a problem with the prior command but is ready to accept a new command.

VARIABLE:VALUE "pam4","clear"

VARIABLE:VALUE "pam4","clear"	Clears all measurement result values from prior analysis.
Syntax	VARIABLE:VALUE "pam4","clear"

VARIABLE:VALUE "pam4","datasource [ch1|ch2|ch3|ch4|math1|ref1]"

VARIABLE:VALUE "pam4","datasource [ch1|ch2|ch3|ch4|math1|ref1]"	Sets the data source (waveform source) to the one designated by the command. May be issued with a question mark, in which case a subsequent query of the PAM4 handshake variable will return "OK:<source>".
Syntax 1	VARIABLE:VALUE "pam4","datasource [ch1|ch2|ch3|ch4|math1|ref1]"
Syntax 2	VARIABLE:VALUE "pam4","datasource?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "datasource ch2"
Example 2	VARIABLE:VALUE "pam4", "datasource?"
	VARIABLE:VALUE? "pam4" (query returns "OK:CH2")

VARIABLE:VALUE "pam4","default"

VARIABLE:VALUE "pam4","default"	Instructs the PAM4 application to recall a factory-default setup file.
Syntax	VARIABLE:VALUE "pam4","default"

VARIABLE:VALUE "pam4","lasterror?"

VARIABLE:VALUE "pam4","lasterror?"	Instructs the PAM4 application to export the latest encountered error.
Syntax	VARIABLE:VALUE "pam4","lasterror?"
Example	VARIABLE:VALUE "pam4","lasterror?" VARIABLE:VALUE? "pam4" (query returns "OK:Exception: Out of memory")

VARIABLE:VALUE "pam4", "probetype <value>"

VARIABLE:VALUE "pam4", "probetype [electrical|optical]"	Sets the measurement category to Electrical or Optical from the Select panel. The application shows the electrical measurement tree in electrical mode and shows the optical tree when optical mode is selected.
Syntax 1	VARIABLE:VALUE "pam4", "probetype [electrical|optical]"
Syntax 2	VARIABLE:VALUE "pam4", "probetype?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "probetype electrical"
Example 2	VARIABLE:VALUE "pam4", "probetype?" VARIABLE:VALUE? "pam4" (query returns "OK:Electrical")

VARIABLE:VALUE "pam4","recalc"

VARIABLE:VALUE "pam4","recalc"	Instructs the PAM4 application to analyze a single waveform which is already present in the scope memory. Until the processing cycle is complete, the only additional commands that will be processed are "state?", "stop" and "exit". All other commands will report an error.
Syntax	VARIABLE:VALUE "pam4","recalc"

VARIABLE:VALUE "pam4","recall <filename>"

VARIABLE:VALUE "pam4","recall <filename>"	Instructs the PAM4 application to recall a previously-saved setup file, thereby restoring the prior configuration. If the filename does not have a ".psf" extension, one will be appended before recall is attempted. Note that the scope configuration is not affected by this operation.
Syntax	VARIABLE:VALUE "pam4","recall <filename>"
Example	VARIABLE:VALUE "pam4","recall C:\Users\<username>\p4setups\r1_noEq.psf"

VARIABLE:VALUE "pam4","run"

VARIABLE:VALUE "pam4","run"	Instructs the PAM4 application to run the application continuously until it meets the population control requirement. It is recommended to set up the population control before using this command. Otherwise, the PAM4 application will keep running forever.
Syntax	VARIABLE:VALUE "pam4","run"

VARIABLE:VALUE "pam4","state?"

VARIABLE:VALUE "pam4","state?"	Instructs the PAM4 application to report its current processing state in response to a subsequent "VARIABLE:VALUE? "pam4"" query.
Syntax	VARIABLE:VALUE "pam4","state?" followed by query

VARIABLE:VALUE? "pam4"

Returns "OK:single", "OK:recalc" or "OK:stop"

VARIABLE:VALUE "pam4","saveconfig <filename>"

VARIABLE:VALUE "pam4","saveconfig <filename>" Instructs the PAM4 application to save the current application state in a setup file.

Syntax VARIABLE:VALUE "pam4","saveconfig <filename>"

Example VARIABLE:VALUE "pam4","saveconfig C:\ Users\<username>\p4setups\r1_noEq.psf"

VARIABLE:VALUE "pam4","single"

VARIABLE:VALUE "pam4","single" Instructs the PAM4 application to acquire and analyze a single waveform. Until the processing cycle is complete, the only additional commands that will be processed are "state?", "stop" and "exit". All other commands will report an error.

Syntax VARIABLE:VALUE "pam4","single"

VARIABLE:VALUE "pam4","savereport <filename>"

VARIABLE:VALUE "pam4","savereport <filename>" Instructs the PAM4 application to save an HTML-style report file with the provided name. If the name does not have a ".mht" extension, one will be appended.

Syntax VARIABLE:VALUE "pam4","savereport <filename>"

Example VARIABLE:VALUE "pam4","savereport C:\ Users\<username>\p4reports\trial3.mht"

VARIABLE:VALUE "pam4","meascategory [electrical|optical]"

VARIABLE:VALUE "pam4","meascategory [electrical|optical]" Sets the measurement type to be "Electrical" or "Optical" by the command. Measurement type should correspond to the related waveform, electrical waveform or optical waveform.

Syntax 1 VARIABLE:VALUE "pam4"," meascategory [electrical|optical]"

Syntax 2 VARIABLE:VALUE "pam4"," meascategory?" followed by query

VARIABLE:VALUE? "pam4"

Example 1 VARIABLE:VALUE "pam4", " meascategory electrical"

Example 2 VARIABLE:VALUE "pam4", " meascategory?"

VARIABLE:VALUE? "pam4" (query returns "OK:Electrical")

VARIABLE:VALUE "pam4", "technology [400GCK | PCIe Gen 6]”

VARIABLE:VALUE "pam4", "technology [400GCK | PCIe Gen 6]” Selects the measurement tree based on the available technology licenses. Selections are “400GCK” and “PCIe Gen 6”.

Syntax 1 VARIABLE:VALUE "pam4", "technology [400GCK | PCIe Gen 6]”

Syntax 2 VARIABLE:VALUE "pam4", "technology?" followed by query

VARIABLE:VALUE? "pam4"

Example 1 VARIABLE:VALUE “pam4”, “technology 400GCK”

Example 2 VARIABLE:VALUE "pam4", "technology?"

VARIABLE:VALUE? "pam4" (query returns "OK:400GCK")

VARIABLE:VALUE "pam4","nominalsymbolrate <value>"

VARIABLE:VALUE "pam4","nominalsymbolrate <value>" Changes the nominal symbol rate to the provided value, using the units of GigaBaud. May be issued with a question mark, in which case a subsequent query of the PAM4 handshake variable will return "OK:<current value>".

Syntax 1 VARIABLE:VALUE "pam4", "nominalsymbolrate <value>"

Syntax 2 VARIABLE:VALUE "pam4", "nominalsymbolrate?" followed by query VARIABLE:VALUE? "pam4"

Example 1 VARIABLE:VALUE "pam4", "nominalsymbolrate 25.0"

Example 2 VARIABLE:VALUE "pam4", "nominalsymbolrate?"

VARIABLE:VALUE? "pam4" (query returns "OK:25000000000")

VARIABLE:VALUE "pam4","nominalbitrate <value>"

VARIABLE:VALUE "pam4","nominalbitrate <value>" Changes the bit rate to the provided value using the units, Gigabit. It refers to the "User Guided" (Gbps) under “Data Rate” in the UI, as shown the figure below. It may be issued with a question mark, in which case a subsequent query of the PAM4 handshake variable will return "OK:<current value>". Symbolrate will be automatically updated to be half the value of nominalbitrate.

Syntax 1 VARIABLE:VALUE "pam4", "nominalbitrate <value>"

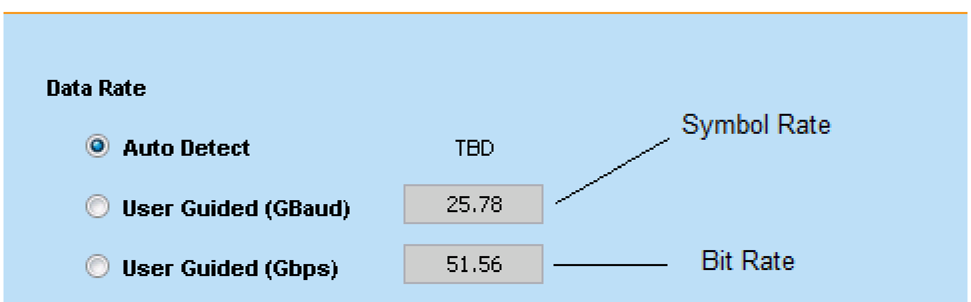
Syntax 2 VARIABLE:VALUE "pam4", "nominalbitrate?" followed by query

VARIABLE:VALUE? "pam4"

Example 1 VARIABLE:VALUE "pam4", "nominalbitrate 50.0"

Example 2 VARIABLE:VALUE "pam4", "nominalbitrate?"

VARIABLE:VALUE? "pam4" (query returns "OK:50000000000")



VARIABLE:VALUE "pam4", "thresholdauto <value>"

VARIABLE:VALUE "pam4", "thresholdauto <value>" Sets the thresholds to be automatically or manually adjusted. The values are [0 1]. One means thresh hold auto adjust is on, zero means it is off.

Syntax 1 VARIABLE:VALUE "pam4", "thresholdauto <value>"

Syntax 2 VARIABLE:VALUE "pam4", "thresholdauto?" followed by query VARIABLE:VALUE? "pam4"

Example 1 VARIABLE:VALUE "pam4", "thresholdauto 0"

Example 2	VARIABLE:VALUE "pam4", "thresholdauto?" VARIABLE:VALUE? "pam4" (query returns "OK:0")

VARIABLE:VALUE "pam4", "thresholdhigh <value>"

VARIABLE:VALUE "pam4", "thresholdhigh <value>"	Sets the thresholds high reference in mV.
Syntax 1	VARIABLE:VALUE "pam4", "thresholdhigh <value>"
Syntax 2	VARIABLE:VALUE "pam4", "thresholdhigh?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "thresholdhigh 78"
Example 2	VARIABLE:VALUE "pam4", "thresholdhigh?" VARIABLE:VALUE? "pam4" (query returns "OK:78")

VARIABLE:VALUE "pam4", "thresholdmid <value>"

VARIABLE:VALUE "pam4", "thresholdmid <value>"	Sets the thresholds mid reference in mV.
Syntax 1	VARIABLE:VALUE "pam4", "thresholdmid <value>"
Syntax 2	VARIABLE:VALUE "pam4", "thresholdmid?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "thresholdmid 1"
Example 2	VARIABLE:VALUE "pam4", "thresholdmid?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "thresholdlow <value>"

VARIABLE:VALUE "pam4", "thresholdlow <value>"	Sets the thresholds low reference in mV.
Syntax 1	VARIABLE:VALUE "pam4", "thresholdlow <value>"
Syntax 2	VARIABLE:VALUE "pam4", "thresholdlow?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "thresholdlow -20"
Example 2	VARIABLE:VALUE "pam4", "thresholdlow?" VARIABLE:VALUE? "pam4" (query returns "OK:-20")

VARIABLE:VALUE "pam4", "waveformfilteron <value>"

VARIABLE:VALUE "pam4", "waveformfilteron <value>"	Turns the waveform filter on/off . The value can be zero or one. One means waveform filter is on, zero means it is off.
Syntax 1	VARIABLE:VALUE "pam4", "waveformfilteron <value>"
Syntax 2	VARIABLE:VALUE "pam4", "waveformfilteron?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "waveformfilteron 1"
Example 2	VARIABLE:VALUE "pam4", "waveformfilteron?"

VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "loadfilter <filtername>"

VARIABLE:VALUE "pam4", "loadfilter <filtername>"	Instructs the PAM4 application to load a previously-saved filter file. If the filename does not have a ".flt" extension, one will be appended before recall is attempted. Note that the scope configuration is not affected by this operation.
Syntax 1	VARIABLE:VALUE "pam4","loadfilter <filter name>"
Syntax 2	VARIABLE:VALUE "pam4", "loadfilter?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4","loadfilter C:\Users\<username>\filter\test.flt"
Example 2	VARIABLE:VALUE "pam4", "loadfilter?" VARIABLE:VALUE? "pam4" (query returns "OK: C:\Users\<username>\filter\test.flt")

VARIABLE:VALUE "pam4", "ctleon <value>"

VARIABLE:VALUE "pam4", "ctleon <value>"	Turns CTLE on/off . The value can be zero or one. One means CTLE is on, zero means it is off.
Syntax 1	VARIABLE:VALUE "pam4", "ctleon [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "ctleon?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ctleon 1"
Example 2	VARIABLE:VALUE "pam4", "ctleon?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "ctletype <value>"

VARIABLE:VALUE "pam4", "ctletype <value>"

Sets the type of CTLE you wish to use. The normally available types are [Custom, OIF-CEI Gen I, OIF-CEI Gen II, IEEE 802.3bs Draft 3.3, IEEE 802.3bs Final, IEEE 802.3ck TP1a, IEEE 802.3ck TP4 Far-End, IEEE 802.3ck TP4 Near-End, PCIe Gen6 64G 0p7, USB-Gen4 v2, PCIe PCIe Gen6 64G 0p7 Extended 0.25dB steps], which have the following characteristics:

- Custom: Allows free access to individual CTLE parameters
- OIF-CEI Gen I: Offers preset parameter sets for 2-pole, 1-zero CTLE designs published by the early OIF-CEI standards
- OIF-CEI Gen II: Offers preset parameter sets for 3-pole, 2-zero CTLE designs published in OIF-CEI drafts as of March 2017
- IEEE 802.3bs Draft 3.1: Offers preset parameter sets for 3-pole, 2-zero CTLE designs published in IEEE 802.3bs Draft 3.1

Groups of CTLE presets are controlled by xml files, so more types may be added periodically. If additional entries appear in the CTLE Type field in the graphical user interface when you run the application, you can select any of those types by using the same string in the ctletype remote command.

Syntax 1	VARIABLE:VALUE "pam4", "ctletype [Custom|OIF-CEI Gen I|OIF-CEI Gen II|IEEE 802.3bs Draft 3.3|IEEE 802.3bs Final|IEEE 802.3ck TP1a|IEEE 802.3ck TP4 Far-End|IEEE 802.3ck TP4 Near-End|PCIe Gen6 64G 0p7|USB-Gen4 v2| PCIe Gen6 64G 0p7 Extended 0.25dB steps]"
Syntax 2	VARIABLE:VALUE "pam4", "ctletype?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ctletype OIF-CEI Gen II"
Example 2	VARIABLE:VALUE "pam4", "ctletype?" VARIABLE:VALUE? "pam4" (query returns "OK:OIF-CEI Gen II")

VARIABLE:VALUE "pam4", "ctledesign <value>"

VARIABLE:VALUE "pam4", "ctledesign <value>"	If the CTLE Type is Custom, sets whether the CTLE offers 2 poles and 1 zero (generally used by early versions of both OIF-CEI and IEEE spec) or 3 poles and 2 zeros (used by later spec versions). The values can be [3 Poles + 2 Zeros, 2 Poles + 1 Zero]. If the CTLE Type is other than "Custom", the choice of CTLE Design can still be set or queried, but will not be active until the Type is changed to "Custom".
Syntax 1	VARIABLE:VALUE "pam4", "ctledesign [3 Poles + 2 Zeros|2 Poles + 1 Zero]"
Syntax 2	VARIABLE:VALUE "pam4", "ctledesign?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ctledesign 2 Poles + 1 Zero"
Example 2	VARIABLE:VALUE "pam4", "ctledesign?" VARIABLE:VALUE? "pam4" (query returns "OK:2 Poles + 1 Zero")

VARIABLE:VALUE "pam4", "ctlepreset <value>"

VARIABLE:VALUE "pam4", "ctlepreset <value>"	Selects one of the CTLE Presets defined for the currently selected CTLE Type. Each Preset sets all of the CTLE parameters (gain, pole and zero frequencies) as a group. The <value> is a string matching any one of the items appearing in the Preset Name control in the graphical user interface, for the currently-selected CTLE Type. For the default Types, the Presets are generally of the form [1 dB, 2 dB, etc]. Whenever the CTLE Type is changed (either via the graphical user interface or the remote interface), the Preset will default to the top selection in the Preset Name control, usually "1 dB". You can then set it to some other value. For backward compatibility, long-form value strings accepted by prior versions of the application will be accepted, and will generally map to the "2 Poles + 1 Zero" design. Value strings consisting only of a number, which were accepted in prior versions, are no longer allowed.
Syntax 1	VARIABLE:VALUE "pam4", "ctlepreset <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ctlepreset?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ctlepreset "9 dB"
Example 2	VARIABLE:VALUE "pam4", "ctlepreset "OIF CEI: 9dB" (deprecated)
Example 3	VARIABLE:VALUE "pam4", "ctlepreset?" VARIABLE:VALUE? "pam4" (query returns "OK:9 dB")

VARIABLE:VALUE "pam4", "ctleadc <value>"

VARIABLE:VALUE "pam4", "ctleadc <value>"	Sets CTLE Adc value. It can only be set when CTLE Type is "Custom".
Syntax 1	VARIABLE:VALUE "pam4", "ctleadc <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ctleadc?" followed by query VARIABLE:VALUE? "pam4"
Example 1	Set CTLE preset to custom mode" first. VARIABLE:VALUE "pam4", "ctleadc 0.79"
Example 2	VARIABLE:VALUE "pam4", "ctleadc?" VARIABLE:VALUE? "pam4" (query returns "OK:0.79")

VARIABLE:VALUE "pam4", "ctlefp1 <value>"

VARIABLE:VALUE "pam4", "ctlefp1 <value>"	Sets CTLE Fp1 value in GHz. It only can be set when CTLE Type is "Custom".
Syntax 1	VARIABLE:VALUE "pam4", "ctlefp1 <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ctlefp1?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ctlefp1 15"
Example 2	VARIABLE:VALUE "pam4", "ctlefp1?" VARIABLE:VALUE? "pam4" (query returns "OK:15")

VARIABLE:VALUE "pam4", "ctlefp2 <value>"

VARIABLE:VALUE "pam4", "ctlefp2 <value>"	Sets CTLE Fp2 value in GHz. It only can be set when CTLE Type is "Custom".
Syntax 1	VARIABLE:VALUE "pam4", "ctlefp2 <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ctlefp2?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ctlefp2 20"
Example 2	VARIABLE:VALUE "pam4", "ctlefp2?" VARIABLE:VALUE? "pam4" (query returns "OK:20")

VARIABLE:VALUE "pam4", "ctlefp3 <value>"

VARIABLE:VALUE "pam4", "ctlefp3 <value>"	Sets CTLE Fp3 value in GHz. It only can be set when CTLE Type is "Custom" and CTLE Design is "3 Poles + 2 Zeros".
Syntax 1	VARIABLE:VALUE "pam4", "ctlefp3 <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ctlefp3?" followed by query VARIABLE:VALUE? "pam4"

Example 1	VARIABLE:VALUE "pam4", "ctlefp3 1.2"
Example 2	VARIABLE:VALUE "pam4", "ctlefp3?"
	VARIABLE:VALUE? "pam4" (query returns "OK:1.2")

VARIABLE:VALUE "pam4", "ctlefz <value>"

VARIABLE:VALUE "pam4", "ctlefz <value>"	Sets CTLE Fz value in GHz. It only can be set when CTLE Type is "Custom".
Syntax 1	VARIABLE:VALUE "pam4", "ctlefz <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ctlefz?" followed by query
	VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ctlefz 7.55"
Example 2	VARIABLE:VALUE "pam4", "ctlefz?"
	VARIABLE:VALUE? "pam4" (query returns "OK:7.55")

VARIABLE:VALUE "pam4", "ctlefz2 <value>"

VARIABLE:VALUE "pam4", "ctlefz2 <value>"	Sets CTLE Fz2 value in GHz. It only can be set when CTLE Type is "Custom" and CTLE Design is "3 Poles + 2 Zeros".
Syntax 1	VARIABLE:VALUE "pam4", "ctlefz2 <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ctlefz2?" followed by query
	VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ctlefz2 1.05"
Example 2	VARIABLE:VALUE "pam4", "ctlefz2?"
	VARIABLE:VALUE? "pam4" (query returns "OK:1.05")

VARIABLE:VALUE "pam4", "dfeon <value>"

VARIABLE:VALUE "pam4", "dfeon <value>"	Turns DFE on/off . The value can be zero or one. One means DFE is on, zero means it is off.
Syntax 1	VARIABLE:VALUE "pam4", "dfeon [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "dfeon?" followed by query
	VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "dfeon 1"
Example 2	VARIABLE:VALUE "pam4", "dfeon?"
	VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "dfeadaptmode <value>"

VARIABLE:VALUE "pam4", "dfeadaptmode <value>"	Sets DFE tap adaptation mode. Selections are "Auto","None", “LFPck” and “LFPpcie”. The “LFPck” and “LFPpcie” options are dependent on the technology licenses.
Syntax 1	VARIABLE:VALUE "pam4", "dfeadaptmode [Auto|None|LFPck|LFPpcie]"

Syntax 2	VARIABLE:VALUE "pam4", "dfeadaptmode?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "dfeadaptmode None"
Example 2	VARIABLE:VALUE "pam4", "dfeadaptmode?" VARIABLE:VALUE? "pam4" (query returns "OK:None")

VARIABLE:VALUE "pam4", "dfenumtaps <value>"

VARIABLE:VALUE "pam4", "dfenumtaps <value>"	Sets DFE number of taps. The range of the taps is [1 16].
Syntax 1	VARIABLE:VALUE "pam4", "dfenumtaps <value>"
Syntax 2	VARIABLE:VALUE "pam4", "dfenumtaps?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "dfenumtaps 3"
Example 2	VARIABLE:VALUE "pam4", "dfenumtaps?" VARIABLE:VALUE? "pam4" (query returns "OK:3")

VARIABLE:VALUE "pam4", "dfetapvalues <filename>"

VARIABLE:VALUE "pam4", "dfetapvalues <filename>"	Sets the DFE tap values as imported from a text file containing multiple rows, each of which correspond to a tap value. If the number of rows does not equate to number of DFE taps, an error message will be returned. In query mode, return the DFE tap values as displayed in DFE configuration window. Due to length limitation of return string length, only the first 140 characters will be displayed. If no tap values are available or DFE is disabled, an error message will be returned.
Syntax 1	VARIABLE:VALUE "pam4", "dfetapvalues <filename>"
Syntax 2	VARIABLE:VALUE "pam4", "dfetapvalues?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "dfetapvalues C:\temp\filename.txt"
Example 2	VARIABLE:VALUE "pam4", "dfetapvalues?" VARIABLE:VALUE? "pam4" (query returns "OK:0.0124 0.0045")

VARIABLE:VALUE "pam4", "ffeon <value>"

VARIABLE:VALUE "pam4", "ffeon <value>"	Turns FFE on/off. The value can be zero or one. One means FFE is on, zero means it is off.
Syntax 1	VARIABLE:VALUE "pam4", "ffeon [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "ffeon?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ffeon 1"
Example 2	VARIABLE:VALUE "pam4", "ffeon?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "ffeadaptmode <value>"

VARIABLE:VALUE "pam4", "ffeadaptmode <value>"	Sets FFE tap adaptation mode. Selections are "Auto" and "None".
Syntax 1	VARIABLE:VALUE "pam4", "ffeadaptmode [Auto|None]"
Syntax 2	VARIABLE:VALUE "pam4", "ffeadaptmode?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ffeadaptmode None"
Example 2	VARIABLE:VALUE "pam4", "ffeadaptmode?" VARIABLE:VALUE? "pam4" (query returns "OK:None")

VARIABLE:VALUE "pam4", "ffenumtaps <value>"

VARIABLE:VALUE "pam4", "ffenumtaps <value>"	Sets FFE number of taps. The range of the taps is [0 25].
Syntax 1	VARIABLE:VALUE "pam4", "ffenumtaps <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ffenumtaps?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ffenumtaps 7"
Example 2	VARIABLE:VALUE "pam4", "ffenumtaps?" VARIABLE:VALUE? "pam4" (query returns "OK:7")

VARIABLE:VALUE "pam4", "ffereftap <value>"

VARIABLE:VALUE "pam4", "ffereftap <value>"	Sets number of precursor taps for FFE. The range of the value is [1 10]. There are restrictions about its value, please refer to online help FFE section.
Syntax 1	VARIABLE:VALUE "pam4", "ffereftap <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ffereftap?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ffereftap 1"
Example 2	VARIABLE:VALUE "pam4", "ffereftap?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "ffetapspersymbol <value>"

VARIABLE:VALUE "pam4", "ffetapspersymbol <value>"	Sets number of taps per unit interval (symbol) for FFE. The range of the value is [1 10]. There are some restrictions about its value, please refer to online help FFE section.
Syntax 1	VARIABLE:VALUE "pam4", "ffetapspersymbol <value>"
Syntax 2	VARIABLE:VALUE "pam4", "ffetapspersymbol?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "ffetapspersymbol 2"

Example 2

VARIABLE:VALUE "pam4", "ffetapspersymbol?"

VARIABLE:VALUE? "pam4" (query returns "OK:2")

VARIABLE:VALUE "pam4", "ffetapvalues <filename>"

VARIABLE:VALUE "pam4", "ffetapvalues <filename>"

Sets the FFE tap values as imported from a text file containing multiple rows, each of which correspond to a tap value. If the number of rows does not equate to number of FFE taps, an error message will be returned. In query mode, return the FFE tap values as displayed in FFE configuration window. Due to length limitation of return string length, only the first 140 characters will be displayed. If no tap values are available or FFE is disabled, an error message will be returned.

Syntax 1

VARIABLE:VALUE "pam4", "ffetapvalues <filename>"

Syntax 2

VARIABLE:VALUE "pam4", "ffetapvalues?" followed by query

VARIABLE:VALUE? "pam4"

Example 1

VARIABLE:VALUE "pam4", "ffetapvalues C:\temp\filename.txt"

Example 2

VARIABLE:VALUE "pam4", "ffetapvalues?"

VARIABLE:VALUE? "pam4" (query returns "OK:1.0644 -0.1053")

VARIABLE:VALUE "pam4", "savelog <filename>"

VARIABLE:VALUE "pam4", "savelog <filename>"

Saves the log to a file. Filename includes the file path as well. If the file path or file name is not correct, it will return an error message when performing query.

Syntax 1

VARIABLE:VALUE "pam4", "savelog <filename>"

Syntax 2

VARIABLE:VALUE "pam4", "savelog?" followed by query

VARIABLE:VALUE? "pam4"

Example 1

VARIABLE:VALUE "pam4", "savelog

C:\Users\<username>\Tektronix\TekApplications\PAM4\test.txt"

Example 2

VARIABLE:VALUE "pam4", "savelog?"

VARIABLE:VALUE? "pam4"

(query returns "OK:

C:\Users\<username>\Tektronix\TekApplications\PAM4\test.txt")

VARIABLE:VALUE "pam4", "clearlog"

VARIABLE:VALUE "pam4", "clearlog"

Clears the log in the log tab.

Syntax

VARIABLE:VALUE "pam4", "clearlog"

Example

VARIABLE:VALUE "pam4", "clearlog"

VARIABLE:VALUE "pam4", "noiseanalysison <value>"

VARIABLE:VALUE "pam4", "noiseanalysison <value>"

Note: This is a deprecated command. Refer to the "select" commands:

VARIABLE:VALUE "pam4", "select <mnames>"

VARIABLE:VALUE "pam4", "deselect <mnames>"

VARIABLE:VALUE "pam4", "isselect <mnames>"

Enable/Disable statistical eye analysis (formerly known as noise analysis). The value can be zero or one. One means the statistical eye is on, zero means it is off.

Syntax 1 VARIABLE:VALUE "pam4", "noiseanalysison <value>"

Syntax 2 VARIABLE:VALUE "pam4", "noiseanalysison?" followed by query

VARIABLE:VALUE? "pam4"

Example 1 VARIABLE:VALUE "pam4", "noiseanalysison 1"

Example 2 VARIABLE:VALUE "pam4", "noiseanalysison?"

VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "noiseber <value>"

VARIABLE:VALUE "pam4", "noiseber <value>" Sets the noise BER rate in scientific format.

Syntax 1 VARIABLE:VALUE "pam4", "noiseber <value>"

Syntax 2 VARIABLE:VALUE "pam4", "noiseber?" followed by query

VARIABLE:VALUE? "pam4"

Example 1 VARIABLE:VALUE "pam4", "noiseber 1e-06"

Example 2 VARIABLE:VALUE "pam4", "noiseber?"

VARIABLE:VALUE? "pam4" (query returns "OK:1e-06")

VARIABLE:VALUE "pam4", "halteyeclosureon <value>"

VARIABLE:VALUE "pam4", "halteyeclosureon <value>" Enable/Disable halt on closed center eye. The value can be zero or one. One means halt on eye closure is on, zero means it is off.

Syntax 1 VARIABLE:VALUE "pam4", "halteyeclosureon <value>"

Syntax 2 VARIABLE:VALUE "pam4", "halteyeclosureon?" followed by query

VARIABLE:VALUE? "pam4"

Example 1 VARIABLE:VALUE "pam4", "halteyeclosureon 1"

Example 2 VARIABLE:VALUE "pam4", "halteyeclosureon?"

VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "popcontrolmethod <value>"

VARIABLE:VALUE "pam4", "popcontrolmethod <value>" Sets the population control methods: [No limit, Number of symbols, Number of Acquisitions].

Syntax 1 VARIABLE:VALUE "pam4", "popcontrolmethod <value>"

Syntax 2 VARIABLE:VALUE "pam4", "popcontrolmethod?" followed by query

VARIABLE:VALUE? "pam4"

Example 1	VARIABLE:VALUE "pam4", "popcontrolmethod No limit"
Example 2	VARIABLE:VALUE "pam4", "popcontrolmethod?" VARIABLE:VALUE? "pam4" (query returns "OK:No limit")

VARIABLE:VALUE "pam4", "popcontrolsymbols <value>"

VARIABLE:VALUE "pam4", "popcontrolsymbols <value>"	Sets the number of symbols for population control.
Syntax 1	VARIABLE:VALUE "pam4", "popcontrolsymbols <value>"
Syntax 2	VARIABLE:VALUE "pam4", "popcontrolsymbols?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "popcontrolsymbols 400000"
Example 2	VARIABLE:VALUE "pam4", "popcontrolsymbols?" VARIABLE:VALUE? "pam4" (query returns "OK:400000")

VARIABLE:VALUE "pam4", "popcontrolacquisitions <value>"

VARIABLE:VALUE "pam4", "popcontrolacquisitions <value>"	Sets the number of acquisitions for population control.
Syntax 1	VARIABLE:VALUE "pam4", "popcontrolacquisitions <value>"
Syntax 2	VARIABLE:VALUE "pam4", "popcontrolacquisitions?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "popcontrolacquisitions 400"
Example 2	VARIABLE:VALUE "pam4", "popcontrolacquisitions?" VARIABLE:VALUE? "pam4" (query returns "OK:400")

VARIABLE:VALUE "pam4", "enablesavecsv <value>"

VARIABLE:VALUE "pam4", "enablesavecsv <value>"	Enable/Disable saving the measurement logging file in csv format. The input value can be zero or one. One means enable saving, zero means disable saving. It saves the measurement logging information at: C:\Users\<username>\Tektronix\TekApplications\PAM4\Log\ PAM4_Results.csv
Syntax 1	VARIABLE:VALUE "pam4", "enablesavecsv <value>"
Syntax 2	VARIABLE:VALUE "pam4", "enablesavecsv?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "enablesavecsv 1"
Example 2	VARIABLE:VALUE "pam4", "enablesavecsv?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "clockmethod <value>"

VARIABLE:VALUE "pam4", "clockmethod <value>"	Sets the clock recovery method. The method can be PLL or Behavioral.
Syntax 1	variable:value "pam4", "clockmethod [PLL, Behavioral]"
Syntax 2	variable:value "pam4", "clockmethod?” followed by query variable:value "pam4"
Example 1	variable:value "pam4", "clockmethod PLL"
Example 2	variable:value "pam4", "clockmethod?” variable:value? "pam4" (query returns "OK:PLL")

VARIABLE:VALUE "pam4", "clockmodel <value>"

VARIABLE:VALUE "pam4", "clockmodel <value>"	Sets the behavioral CDR model. The <value> is a string matching any one of the items appearing in the CDR Model control in the graphical user interface. CDR Model appears only when Behavioral CDR is selected as the Clock Method.
Syntax 1	variable:value "pam4", "clockmodel PCIe Gen6 64G 0p7"
Syntax 2	variable:value "pam4", "clockmodel?” followed by query variable:value? "pam4"
Example 1	variable:value "pam4", "clockmodel PCIe Gen6 64G 0p7"
Example 2	variable:value "pam4", "clockmodel?” variable:value? "pam4" (query returns "OK:PCIe Gen6 64G 0p7")

VARIABLE:VALUE "pam4", "clockpllmodel <value>"

VARIABLE:VALUE "pam4", "clockpllmodel <value>"	Sets the PLL model type. The type can be type 1 or type 2.
Syntax 1	VARIABLE:VALUE "pam4", "clockpllmodel [type 1|type 2]"
Syntax 2	VARIABLE:VALUE "pam4", "clockpllmodel?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "clockpllmodel type 1"
Example 2	VARIABLE:VALUE "pam4", "clockpllmodel?" VARIABLE:VALUE? "pam4" (query returns "OK:type1")

VARIABLE:VALUE "pam4", "clockplldamp <value>"

VARIABLE:VALUE "pam4", "clockplldamp <value>"	Sets the PLL damping factor for type 2 PLL.
Syntax 1	VARIABLE:VALUE "pam4", "clockplldamp <value>"
Syntax 2	VARIABLE:VALUE "pam4", "clockplldamp?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "clockplldamp 0.700"
Example 2	VARIABLE:VALUE "pam4", "clockplldamp?" VARIABLE:VALUE? "pam4" (query returns "OK:0.7")

VARIABLE:VALUE "pam4", "clockpllbw <value>"

VARIABLE:VALUE "pam4", "clockpllbw <value>"	This command is deprecated and may be removed in the future, please use the PI command "clockjtfbw" instead. This command sets PLL JTF bandwidth in MHz.
Syntax 1	VARIABLE:VALUE "pam4", "clockpllbw <value>"
Syntax 2	VARIABLE:VALUE "pam4", "clockpllbw?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "clockpllbw 17"
Example 2	VARIABLE:VALUE "pam4", "clockpllbw?" VARIABLE:VALUE? "pam4" (query returns "OK:17")

VARIABLE:VALUE "pam4", "clockjtfbw <value>"

VARIABLE:VALUE "pam4", "clockjtfbw <value>"	Sets the PLL JTF bandwidth in MHz.
Syntax 1	VARIABLE:VALUE "pam4", "clockjtfbw <value>"
Syntax 2	VARIABLE:VALUE "pam4", "clockjtfbw?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "clockjtfbw 17"
Example 2	VARIABLE:VALUE "pam4", "clockjtfbw?" VARIABLE:VALUE? "pam4" (query returns "OK:17")

VARIABLE:VALUE "pam4", "results ceijittermeas?"

VARIABLE:VALUE "pam4", "results ceijittermeas?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns OIF-CEI jitter measurements UUGJ, UBHPJ and even-odd jitter (EOJ).
Syntax	VARIABLE:VALUE "pam4", "results ceijittermeas?"
Example	VARIABLE:VALUE "pam4", "results ceijittermeas?" VARIABLE:VALUE? "pam4" (query returns "OK:UUGJ 4.267ps;UBHPJ 4.128ps;EOJ 3.015fs;")

VARIABLE:VALUE "pam4", "results corrmeas?"

VARIABLE:VALUE "pam4", "results corrmeas?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the content of upper left table on the average waveform page.
Syntax	VARIABLE:VALUE "pam4", "results corrmeas?"
Example	VARIABLE:VALUE "pam4", "results corrmeas?" VARIABLE:VALUE? "pam4" (query returns "OK:Level Deviation 598.0m%;Level Thickness 3.811%;TimeDeviationOrig 3.571%;TimeDeviationMean 892.9m%;Pk-Pk 704.3mV; ---;")

VARIABLE:VALUE "pam4", "results corrlevel1?"

VARIABLE:VALUE "pam4", "results corrlevel1?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns 1st and 2nd rows of the upper left table on the average waveform page.
Syntax	VARIABLE:VALUE "pam4", "results corrlevel1?"
Example	VARIABLE:VALUE "pam4", "results corrlevel1?" VARIABLE:VALUE? "pam4" (query returns "OK:Level 3 Time (e) 4.156ps;Level 3 Amp (d) 119.3mV;Level 3 Std Dev 4.385mV;Level 2 Time (e) -5.541ps;Level 2 Amp (d) 39.90mV;Level 2 Std Dev 4.216mV")

VARIABLE:VALUE "pam4", "results corrlevel2?"

VARIABLE:VALUE "pam4", "results corrlevel2?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns 3rd and 4th rows of the upper right table on the average waveform page, beginning with Level 1.
Syntax	VARIABLE:VALUE "pam4", "results corrlevel2?"
Example	VARIABLE:VALUE "pam4", "results corrlevel2?" VARIABLE:VALUE? "pam4" (query returns "OK:Level 1 Time (e) 4.156ps;Level 1 Amp (d) -41.03mV;Level 1 Std Dev 3.716mV;Level 0 Time (e) 4.156ps;Level 0 Amp (d) -120.6mV;Level 0 Std Dev 3.353mV")

VARIABLE:VALUE "pam4", "results eyelow?"

VARIABLE:VALUE "pam4", "results eyelow?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns lower eye information in lower table of full waveform page.
Syntax	VARIABLE:VALUE "pam4", "results eyelow?"
Example	VARIABLE:VALUE "pam4", "results eyelow?" VARIABLE:VALUE? "pam4" (query returns "OK:Lower Eye Thresh -80.92mV;Lower Eye Offset -2.030ps;Lower Eye TJ@-5 ---;Lower Eye TJ@-10 ---;Lower Eye TJ@-15 ---;Lower Eye RJ(d-d) ---;Lower Eye DJ(d-d) ---;Lower Eye Width n/a *;Lower Eye Height n/a *;")

VARIABLE:VALUE "pam4", "results eyemid?"

VARIABLE:VALUE "pam4", "results eyemid?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns middle eye information in lower table of full waveform page.
Syntax	VARIABLE:VALUE "pam4", "results eyemid?"
Example	VARIABLE:VALUE "pam4", "results eyemid?" VARIABLE:VALUE? "pam4" (query returns "OK:Middle Eye Thresh -1.028mV;Middle Eye Offset -2.337ps;Middle Eye TJ@-5 ---;Middle Eye TJ@-10 ---;Middle Eye TJ@-15 ---;Middle Eye RJ(d-d) ---;Middle Eye DJ(d-d) ---;Middle Eye Width n/a *;Middle Eye Height n/a *;")

VARIABLE:VALUE "pam4", "results eyeupp?"

VARIABLE:VALUE "pam4", "results eyeupp?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns upper eye information in lower table of full waveform page.
Syntax	VARIABLE:VALUE "pam4", "results eyeupp?"
Example	VARIABLE:VALUE "pam4", "results eyeupp?" VARIABLE:VALUE? "pam4" (query returns "OK:Upper Eye Thresh 78.59mV;Upper Eye Offset -1.610ps;Upper Eye TJ@-5 ---;Upper Eye TJ@-10 ---;Upper Eye TJ@-15 ---;Upper Eye RJ(d-d) ---;Upper Eye DJ(d-d) ---;Upper Eye Width n/a *;Upper Eye Height n/a *;")

VARIABLE:VALUE "pam4", "results fall1?"

VARIABLE:VALUE "pam4", "results fall1?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the 2- and 3-level transition fall time results from Rise/Fall page.
Syntax	VARIABLE:VALUE "pam4", "results fall1?"
Example	VARIABLE:VALUE "pam4", "results fall1?" VARIABLE:VALUE? "pam4" (query returns "OK:Fall(3-0) Max 12.8ps;Fall(3-0) Mean 12.5ps;Fall(3-0) Min 11.9ps;Fall(2-0) Max 15.2ps;Fall(2-0) Mean 13.2ps;Fall(2-0) Min 11.4ps;Fall(3-1) Max 12.7ps;Fall(3-1) Mean 12.1ps;Fall(3-1) Min 11.6ps;")

VARIABLE:VALUE "pam4", "results fall2?"

VARIABLE:VALUE "pam4", "results fall2?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the 1-level transition fall time results from Rise/Fall page.
Syntax	VARIABLE:VALUE "pam4", "results fall2?"
Example	VARIABLE:VALUE "pam4", "results fall2?" VARIABLE:VALUE? "pam4" (query returns "OK:Fall(1-0) Max ---;Fall(1-0) Mean ---;Fall(1-0) Min 9.70ps;Fall(2-1) Max 14.5ps;Fall(2-1) Mean 12.1ps;Fall(2-1) Min 10.1ps;Fall(3-2) Max 12.2ps;Fall(3-2) Mean 11.4ps;Fall(3-2) Min 10.2ps;")

VARIABLE:VALUE "pam4", "results fullmeas?"

VARIABLE:VALUE "pam4", "results fullmeas?"	⚠ **Note:** This command is deprecated. Use the commands: VARIABLE:VALUE "pam4", "results fullmeas1?" VARIABLE:VALUE "pam4", "results fullmeas2?" This command uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the upper left table contents on the full waveform page, beginning with unit interval.
Syntax	VARIABLE:VALUE "pam4", "results fullmeas?"
Example	VARIABLE:VALUE "pam4", "results fullmeas?" VARIABLE:VALUE? "pam4"

(query returns "OK:Unit Interval 38.79ps;Symbol Rate 25.78GBd;Equiv Bitrate (2xBd) 51.56Gbps;Pattern Length 2047;Symbol Population 123099;Linearity (R_LM) 91.44%;EW6 insufficient data *;EH6 insufficient data *;VEC insufficient data *;")

VARIABLE:VALUE "pam4", "results fullmeas1?"

VARIABLE:VALUE "pam4", "results fullmeas1?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the first half of the upper left table contents on the full waveform page.
Syntax	VARIABLE:VALUE "pam4", "results fullmeas1?"
Example	VARIABLE:VALUE "pam4", "results fullmeas1?" VARIABLE:VALUE? "pam4" (query returns "OK:Unit Interval 38.79ps;Symbol Rate 25.78GBd;Equiv Bitrate (2xBd) 51.56Gbps;Pattern Length 127;Symbol Population 106159;Symbol Errors 0;Symbol Error Ratio 0.0000e+00;Bit Error Ratio 0.0000e+00;")

VARIABLE:VALUE "pam4", "results fullmeas2?"

VARIABLE:VALUE "pam4", "results fullmeas2?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the second half of the upper left table contents on the full waveform page.
Syntax	VARIABLE:VALUE "pam4", "results fullmeas2?"
Example	VARIABLE:VALUE "pam4", "results fullmeas2?" VARIABLE:VALUE? "pam4" (query returns "OK:Linearity R_LM 94.62%;EW6 insufficient data *;EH6 insufficient data *;VEC insufficient data *;SNDR Disabled;Peak Disabled;Sigma_e Disabled;Sigma_n Disabled;")

VARIABLE:VALUE "pam4", "results fulllevel1?"

VARIABLE:VALUE "pam4", "results fulllevel1?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns 1st and 2nd rows of upper right table on the full waveform page, beginning with V_D(3).
Syntax	VARIABLE:VALUE "pam4", "results fulllevel1?"
Example	VARIABLE:VALUE "pam4", "results fulllevel1?" VARIABLE:VALUE? "pam4" ("OK:V_D(3) Mean 118.0mV;V_D(3) StdDev 7.202mV;V_D(3) Pk-Pk 59.14mV;V_C(2) Mean 40.21mV;V_C(2) StdDev 6.257mV;V_C(2) Pk-Pk 48.74mV;")

VARIABLE:VALUE "pam4", "results fulllevel2?"

VARIABLE:VALUE "pam4", "results fulllevel2?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns 3rd and 4th rows of upper right table on the full waveform page, beginning with V_B(1).
Syntax	VARIABLE:VALUE "pam4", "results fulllevel2?"
Example	VARIABLE:VALUE "pam4", "results fulllevel2?" VARIABLE:VALUE? "pam4" (query returns "OK:V_B(1) Mean -41.66mV;V_B(1) StdDev 5.950mV;V_B(1) Pk-Pk 52.32mV;V_A(0) Mean -120.0mV;V_A(0) StdDev 6.292mV;V_A(0) Pk-Pk 48.65mV;")

VARIABLE:VALUE "pam4","results hlowleft?"

VARIABLE:VALUE "pam4","results hlowleft?"	The command is used in Eye Symmetric Mask Width (ESMW) calculation. It returns the left corner of lower eye plot in time unit. After this command, it uses VARIABLE:VALUE? "PAM4" query to obtain the value.
Syntax	VARIABLE:VALUE "pam4", "results hlowleft?"
Example	VARIABLE:VALUE "pam4", "results hlowleft?" VARIABLE:VALUE? "pam4" (query returns "OK:-8.734ps")

VARIABLE:VALUE "pam4","results hlowright?"

VARIABLE:VALUE "pam4","results hlowright?"	The command is used in Eye Symmetric Mask Width (ESMW) calculation. It returns the right corner of lower eye in time unit. After this command, it uses VARIABLE:VALUE? "PAM4" query to obtain the value.
Syntax	VARIABLE:VALUE "pam4", "results hlowright?"
Example	VARIABLE:VALUE "pam4", "results hlowright?" VARIABLE:VALUE? "pam4" (query returns "OK:2.56ps")

VARIABLE:VALUE "pam4","results hmidleft?"

VARIABLE:VALUE "pam4","results hmidleft?"	The command is used in Eye Symmetric Mask Width (ESMW) calculation. It returns the left corner of mid eye in time unit. After this command, it uses VARIABLE:VALUE? "PAM4" query to obtain the value.
Syntax	VARIABLE:VALUE "pam4", "results hmidleft?"
Example	VARIABLE:VALUE "pam4", "results hmidleft?" VARIABLE:VALUE? "pam4" (query returns "OK:8.734ps")

VARIABLE:VALUE "pam4","results hmidright?"

VARIABLE:VALUE "pam4","results hmidright?"	The command is used in Eye Symmetric Mask Width (ESMW) calculation. It returns the right corner of mid eye in time unit. After this command, it uses VARIABLE:VALUE? "PAM4" query to obtain the value.
Syntax	VARIABLE:VALUE "pam4", "results hmidright?"
Example	VARIABLE:VALUE "pam4", "results hmidright?" VARIABLE:VALUE? "pam4" (query returns "OK: 602.4fs")

VARIABLE:VALUE "pam4","results huppleft?"

VARIABLE:VALUE "pam4","results huppleft?"	The command is used in Eye Symmetric Mask Width (ESMW) calculation. It returns the left corner of upper eye in time unit. After this command, it uses VARIABLE:VALUE? "PAM4" query to obtain the value.
Syntax	VARIABLE:VALUE "pam4", "results huppleft?"
Example	VARIABLE:VALUE "pam4", "results huppleft?" VARIABLE:VALUE? "pam4" (query returns "OK: 2.4fs")

VARIABLE:VALUE "pam4","results huppright?"

VARIABLE:VALUE "pam4","results huppright?"	The command is used in Eye Symmetric Mask Width (ESMW) calculation. It returns the right corner of upper eye in time unit. After this command, it uses VARIABLE:VALUE? "PAM4" query to obtain the value.

Syntax	VARIABLE:VALUE "pam4", "results hupPright?"
Example	VARIABLE:VALUE "pam4", "results huppright?"
	VARIABLE:VALUE? "pam4" (query returns "OK: 602.4fs")

VARIABLE:VALUE "pam4", "results ieeemeas?"

VARIABLE:VALUE "pam4", "results ieeemeas?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the IEEE measurements including Jrms, J4u, J3u, even and odd jitter, rise and fall time, SNR_ISI, dvf and dRpeak.
Syntax	VARIABLE:VALUE "pam4", "results ieeemeas?"
Example	VARIABLE:VALUE "pam4", "results ieeemeas?"
	VARIABLE:VALUE? "pam4"
	(query returns "OK:Jrms ---;J3u ---;J4u ---;EOJ (IEEE) ---;IEEE Rise Time ---;IEEE Fall Time ---;SNR_ISI---;dvf---;dRpeak---;")

VARIABLE:VALUE "pam4", "results opticalmeas?"

VARIABLE:VALUE "pam4", "results opticalmeas?"	Uses VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the optical measurements that include ER, OMA, AOP, RINxOMA, TDECQ, OMA(outer) – TDECQ, Ceq.
Syntax	VARIABLE:VALUE "pam4", "results opticalmeas?"
Example	VARIABLE:VALUE "pam4", "results opticalmeas?"
	VARIABLE:VALUE? "pam4"
	(query returns "OK:ER ---;OMA ---;AOP ---;RINxOMA ---;TDECQ ---;OMA(outer) - TDECQ ---;Ceq ---;")

VARIABLE:VALUE "pam4", "eradjustison <value>"

VARIABLE:VALUE "pam4", "eradjustison <value>"	Instructs the PAM4 application to enable/disable the ER adjust checkbox.
Syntax 1	VARIABLE:VALUE "pam4", "eradjustison [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "eradjustison?"VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", " eradjustison 1"
Example 2	VARIABLE:VALUE "pam4", " eradjustison?"VARIABLE:VALUE? "pam4" (query returns "OK: 1")

VARIABLE:VALUE "pam4", "eradjust <value>"

VARIABLE:VALUE "pam4", "eradjust <value>"	Instructs the PAM4 application to set the ER adjust value.
Syntax 1	VARIABLE:VALUE "pam4", " eradjust <value>"
Syntax 2	VARIABLE:VALUE "pam4", " eradjust?"VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", " eradjust 2"
Example 2	VARIABLE:VALUE "pam4", " eradjust?"VARIABLE:VALUE? "pam4" (query returns "OK: 2")

VARIABLE:VALUE "pam4", "results rise1?"

VARIABLE:VALUE "pam4", "results rise1?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the 2- and 3-level transition rise time results from Rise/Fall page.
Syntax	VARIABLE:VALUE "pam4", "results rise1?"
Example	VARIABLE:VALUE "pam4", "results rise1?" VARIABLE:VALUE? "pam4" (query returns "OK:Rise(0-3) Max 14.4ps;Rise(0-3) Mean 13.4ps;Rise(0-3) Min 12.1ps;Rise(0-2) Max 15.2ps;Rise(0-2) Mean 12.7ps;Rise(0-2) Min 11.1ps;Rise(1-3) Max 14.8ps;Rise(1-3) Mean 13.3ps;Rise(1-3) Min 12.1ps;")

VARIABLE:VALUE "pam4", "results rise2?"

VARIABLE:VALUE "pam4", "results rise2?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the 1-level transition rise time results from Rise/Fall page.
Syntax	VARIABLE:VALUE "pam4", "results rise2?"
Example	VARIABLE:VALUE "pam4", "results rise2?" VARIABLE:VALUE? "pam4" (query returns "OK:Rise(0-1) Max 14.1ps;Rise(0-1) Mean 12.2ps;Rise(0-1) Min 9.88ps;Rise(1-2) Max 15.0ps;Rise(1-2) Mean 11.9ps;Rise(1-2) Min 9.57ps;Rise(2-3) Max 16.2ps;Rise(2-3) Mean 14.4ps;Rise(2-3) Min 11.9ps;")

VARIABLE:VALUE "pam4", "results sigmanperlevel?"

VARIABLE:VALUE "pam4", "results sigmanperlevel?"	Uses VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns sigma N values for each level.
Syntax	VARIABLE:VALUE "pam4", "results sigmanperlevel?"
Example	VARIABLE:VALUE "pam4", "results sigmanperlevel?" VARIABLE:VALUE? "pam4" (query returns "OK:Sigma n (3) ---;Sigma n (2) ---;Sigma n (1) ---;Sigma n (0) ---;")

VARIABLE:VALUE "pam4", "results sndrtp?"

VARIABLE:VALUE "pam4", "results sndrtp?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameter afterwards; it returns tp, the index of Pmax computed for SNDR measurements.
Syntax	VARIABLE:VALUE "pam4", "results sndrtp?"
Example	VARIABLE:VALUE "pam4", "results sndrtp?" VARIABLE:VALUE? "pam4" (query returns "OK: 46")

VARIABLE:VALUE "pam4", "results sndrvoltage?"

VARIABLE:VALUE "pam4", "results sndrvoltage?"	Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the steady state voltage.
Syntax	VARIABLE:VALUE "pam4", "results sndrvoltage?"

Example	VARIABLE:VALUE "pam4", "results sndrvoltage?" VARIABLE:VALUE? "pam4" (query returns "OK: 0.5")

VARIABLE:VALUE "pam4","results tmid?"

VARIABLE:VALUE "pam4","results tmid?"	The command is used in Eye Symmetric Mask Width (ESMW) calculation. It returns the midpoint of a PAM4 eye plot in time unit. After this command, it uses VARIABLE:VALUE? "PAM4" query to obtain the value.
Syntax	VARIABLE:VALUE "pam4", "results tmid?"
Example	VARIABLE:VALUE "pam4", "results tmid?" VARIABLE:VALUE? "pam4" (query returns "OK:-75.29fs")

VARIABLE:VALUE "pam4", "results tx preset equalization?"

VARIABLE:VALUE "pam4", “results tx preset equalization?"	Together with a subsequent VARIABLE:VALUE? "PAM4" query, this command returns the Preset 2, Preset 1, and De-emphasis values for the current Tx Preset.
Syntax	VARIABLE:VALUE "pam4", "results tx preset equalization?”
Example	VARIABLE:VALUE "pam4", " results tx preset equalization?" VARIABLE:VALUE? "pam4" (query returns "OK: Tx Preset Equalization -0.0dB 2.1dB 0.1dB;")

VARIABLE:VALUE "pam4", "scopenoisemvrms <value>"

VARIABLE:VALUE "pam4", "scopenoisemvrms <value>"	Sets the oscilloscope noise in mV for SNDR, Sigma N, Sigma N per level, Jrms, J3u, J4u, and Eye Height. The value should be larger than zero and smaller than 5.
Syntax 1	VARIABLE:VALUE "pam4", "scopenoisemvrms <value>"
Syntax 2	VARIABLE:VALUE "pam4", "scopenoisemvrms?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "scopenoisemvrms 2"
Example 2	VARIABLE:VALUE "pam4", "scopenoisemvrms?" VARIABLE:VALUE? "pam4" (query returns "OK: 2")

VARIABLE:VALUE "pam4", "sndron <value>"

VARIABLE:VALUE "pam4", "sndron <value>"	Enable SNDR Signal Noise Distribution Ratio (SNDR). The values can be 1 or 0. One means SNDR is on, zero means it is off.
Syntax 1	VARIABLE:VALUE "pam4", "sndron [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "sndron?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "sndron 1"
Example 2	VARIABLE:VALUE "pam4", "sndron?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "sndrsamplenumber <value>"

VARIABLE:VALUE "pam4", "sndrsamplenumber <value>"	Sets the SNDR samples per symbol (M). The value is in the range [32 200].
Syntax 1	VARIABLE:VALUE "pam4", "sndrsamplenumber <value>"
Syntax 2	VARIABLE:VALUE "pam4", "sndrsamplenumber?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "sndrsamplenumber 32"
Example 2	VARIABLE:VALUE "pam4", "sndrsamplenumber?" VARIABLE:VALUE? "pam4" (query returns "OK:32")

VARIABLE:VALUE "pam4", "sndrfilterlength <value>"

VARIABLE:VALUE "pam4", "sndrfilterlength <value>"	Sets the SNDR linear fit pulse filter length (Np). The value is in the range [2 100].
Syntax 1	VARIABLE:VALUE "pam4", "sndrfilterlength <value>"
Syntax 2	VARIABLE:VALUE "pam4", "sndrfilterlength?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "sndrfilterlength 64"
Example 2	VARIABLE:VALUE "pam4", "sndrfilterlength?" VARIABLE:VALUE? "pam4" (query returns "OK:64")

VARIABLE:VALUE "pam4", "sndrdelay <value>"

VARIABLE:VALUE "pam4", "sndrdelay <value>"	Sets the SNDR linear fit pulse filter delay (Dp) in symbols. The value is in the range [1 100].
Syntax 1	VARIABLE:VALUE "pam4", "sndrdelay <value>"
Syntax 2	VARIABLE:VALUE "pam4", "sndrdelay?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "sndelay 65"
Example 2	VARIABLE:VALUE "pam4", "sndrdelay?" VARIABLE:VALUE? "pam4" (query returns "OK:65")

VARIABLE:VALUE "pam4", "sndrnb <value>"

VARIABLE:VALUE "pam4", "sndrnb <value>"	Sets Nb for SNR_{ISI} computation. The value is in the range 5 to 50.
Syntax 1	VARIABLE:VALUE "pam4", "sndrnb <value>"
Syntax 2	VARIABLE:VALUE "pam4", "sndrnb?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "sndrnb 12"
Example 2	VARIABLE:VALUE "pam4", "sndrnb?"

VARIABLE:VALUE? "pam4" (query returns "OK:12")

VARIABLE:VALUE "pam4", "sndrbtfilteron <value>"

VARIABLE:VALUE "pam4", "sndrbtfilteron <value>"	This command is deprecated and may be removed in the future. The Bessel-Thomson filter is redesigned and can be enabled/disabled by "btfilteron". Enable/Disable the Bessel-Thomson filter. The values can be 1 or 0. One means the filter is on, zero means it is off.
Syntax 1	VARIABLE:VALUE "pam4", "sndrbtfilteron [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "sndrbtfilteron?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "sndrbtfilteron 1"
Example 2	VARIABLE:VALUE "pam4", "sndrbtfilteron?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "sndrdebugplotson <value>"

VARIABLE:VALUE "pam4", "sndrdebugplotson <value>"	Allows the SNDR to show more debug plots. The values can be 1 or 0. One means the debug plots are shown, zero means they are not.
Syntax 1	VARIABLE:VALUE "pam4", "sndrdebugplotson [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "sndrdebugplotson?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "sndrdebugplotson 1"
Example 2	VARIABLE:VALUE "pam4", "sndrdebugplotson?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "tx_preset <value>"

VARIABLE:VALUE "pam4", “tx_preset <value>"	Sets the Tx FFE preset to be tested. The range of the presets is [Q0 – Q10].
Syntax 1	VARIABLE:VALUE "pam4", "tx_preset <value>
Syntax 2	VARIABLE:VALUE "pam4", "tx_preset?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "tx_preset Q7"
Example 2	VARIABLE:VALUE "pam4", "tx_preset?" VARIABLE:VALUE? "pam4" (query returns "OK:q7")

VARIABLE:VALUE "pam4", "errnavenabled <value>"

VARIABLE:VALUE "pam4", "errnavenabled <value>"	Enable Error Navigator when it is 1. The values can be 1 or 0. One means Error Navigator is enabled, zero means it is not.
Syntax 1	VARIABLE:VALUE "pam4", "errnavenabled [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "errnavenbled?" followed by query

	VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "errnavenabled 1"
Example 2	VARIABLE:VALUE "pam4", "errnavenabled ?"
	VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "enablerelativeunit <value>"

VARIABLE:VALUE "pam4", "enablerelativeunit <value>"	Enable/Disable the use of relatively time unit (UI instead of second) for selected measurements. The values can be 1 or 0. One enables the relative time unit.
Syntax 1	VARIABLE:VALUE "pam4", "enablerelativeunit [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "enablerelativeunit?" followed by query
	VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "enablerelativeunit 1"
Example 2	VARIABLE:VALUE "pam4", "enablerelativeunit?"
	VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4","export <plot title>:<filename>"

VARIABLE:VALUE "pam4","export <plot title>:<filename>"

Instructs the PAM4 application to export a plot to a file. The plot title corresponds to the title displayed in the main plot, which includes: accumulated, correlated, linear, and TDECQ.

Use the following plot title to export the respective plot:

- Select Accumulated for Accumulated Eye plot
- Select Correlated for Correlated Eye plot
- Select Linear for Linear fit pulse response plot
- Select TDCEQ for TDCEQ eye plot

The export file type includes:

*.bmp; *.emf;.*.eps; *.jpg; *.png; *.tif ; *.fig; *.mat; *.csv

Syntax	VARIABLE:VALUE "pam4","export <plot title>: <filename>"
Example	VARIABLE:VALUE "pam4","export accumulated:c:\temp\test.fig"

VARIABLE:VALUE "pam4", "loggpibtraffic <value>"

VARIABLE:VALUE "pam4", "loggpibtraffic <value>"	Enable/Disable the function to log GPIB traffic in the "Log" panel. The value can be 1 or 0. One enables logging.
Syntax 1	VARIABLE:VALUE "pam4", "loggpibtraffic [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "loggpibtraffic?" followed by query
	VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "loggpibtraffic 1"
Example 2	VARIABLE:VALUE "pam4", "loggpibtraffic?"
	VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "btfilteron <value>"

VARIABLE:VALUE "pam4", "btfilteron <value>"	⚠ **Note:** All BT Filter commands are depreciated. Use the Receive Filter commands Turns the Bessel-Thomson filter on/off in the Setup panel. The value can be 1 or 0. One enables the filter.
Syntax 1	VARIABLE:VALUE "pam4", "btfilteron [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "btfilteron?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "btfilteron 1"
Example 2	VARIABLE:VALUE "pam4", "btfilteron?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "btfilterbw <value>"

VARIABLE:VALUE "pam4", "btfilterbw <value>"	Sets the Bessel-Thomson Filter bandwidth value in GHz. It is shown at the Bessel-Thomson Filter popup menu.
Syntax 1	VARIABLE:VALUE "pam4", "btfilterbw <value>"
Syntax 2	VARIABLE:VALUE "pam4", "btfilterbw?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "btfilterbw 25"
Example 2	VARIABLE:VALUE "pam4", "btfilterbw?" VARIABLE:VALUE? "pam4" (query returns "OK:25")

VARIABLE:VALUE "pam4", "btfiltermode <value>"

VARIABLE:VALUE "pam4", "btfiltermode <value>"	Sets the Bessel-Thomson Filter bandwidth mode. It can be "auto" or "manual". It is shown at the Bessel-Thomson Filter popup menu. The application calculates the filter bandwidth automatically in "Auto" mode. The user inputs the filter bandwidth manually in "Manual" mode.
Syntax 1	VARIABLE:VALUE "pam4", "btfiltermode [auto|manual]"
Syntax 2	VARIABLE:VALUE "pam4", "btfiltermode?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "btfiltermode manual"
Example 2	VARIABLE:VALUE "pam4", "btfiltermode?" VARIABLE:VALUE? "pam4" (query returns "OK:manual")

VARIABLE:VALUE "pam4", "rxfilteron <value>"

VARIABLE:VALUE "pam4", "results rxfilteron <value>"	Turns the Receive filter on/off in the Setup panel. The value can be 1 or 0. One enables the filter.
Syntax 1	VARIABLE:VALUE "pam4", "rxfilteron [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "rxfilteron?" followed by query VARIABLE:VALUE? "pam4"

Example 1	VARIABLE:VALUE "pam4", "rxfilteron 1"
Example 2	VARIABLE:VALUE "pam4", "rxfilteron?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", rxfilterbw<value>"

VARIABLE:VALUE "pam4", "rxfilterbw<value>"	Sets the Receive Filter bandwidth value in GHz. It is shown at the ReceiveFilter popup menu.
Syntax 1	VARIABLE:VALUE "pam4", "rxfilterbw <value>"
Syntax 2	VARIABLE:VALUE "pam4", "rxfilterbw?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "rxfilterbw 25"
Example 2	VARIABLE:VALUE "pam4", "rxfilterbw?" VARIABLE:VALUE? "pam4" (query returns "OK:25")

VARIABLE:VALUE "pam4", "rxfiltertype <value>"

VARIABLE:VALUE "pam4", "rxfiltertype <value>""	Sets the type of Receive Filter. Selections are “Bessel-Thomson” and “Butterworth”. It is shown at the ReceiveFilter popupmenu.
Syntax 1	VARIABLE:VALUE "pam4", "rxfiltertype <Bessel-Thomson|Butterworth>"
Syntax 2	VARIABLE:VALUE "pam4", " rxfiltertype?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", " rxfiltertype Butterworth "
Example 2	VARIABLE:VALUE "pam4", " rxfiltertype?" VARIABLE:VALUE? "pam4" (query returns "OK: Butterworth ")

VARIABLE:VALUE "pam4", "rxfiltermode <value>"

VARIABLE:VALUE "pam4", "rxfiltermode <value>""	Sets the Receive Filter bandwidth mode. It can be "auto" or "manual". It is shown at the Receive Filter popup menu. The application calculates the filter bandwidth automatically in "Auto" mode. The user inputs the filter bandwidth manually in "Manual" mode.
Syntax 1	VARIABLE:VALUE "pam4", "rxfiltermode [auto|manual]"
Syntax 2	VARIABLE:VALUE "pam4", "rxfiltermode?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "rxfiltermode manual"
Example 2	VARIABLE:VALUE "pam4", "rxfiltermode?" VARIABLE:VALUE? "pam4" (query returns "OK:manual")

VARIABLE:VALUE "pam4", "symbolratemode <value>"

VARIABLE:VALUE "pam4", "symbolratemode <value>"	⚠ **Note:** This is a deprecated command. Use the following command: VARIABLE:VALUE "pam4", "dataratemode <value>" Sets the symbol rate mode to be auto, user and userbit. It is shown at the Main Setup panel of Config menu. The application detect symbol rate automatically in Auto mode. The user needs to input the symbol rate in user mode.
Syntax 1	VARIABLE:VALUE "pam4", "symbolratermode <value>"

Syntax 2	VARIABLE:VALUE "pam4", "symbolratemode?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "symbolratemode user"
Example 2	VARIABLE:VALUE "pam4", "symbolratemode?" VARIABLE:VALUE? "pam4" (query returns "OK:user")

VARIABLE:VALUE "pam4", "dataratemode <value>"

VARIABLE:VALUE "pam4", "dataratemode <value>"	Sets the data rate mode to "auto", "user" or "userbit". It is shown at the "Main Setup" panel of the "Config" menu. The application detects the symbol rate automatically when in "Auto" mode. The user needs to input the data rate in "User" mode (GBauds) and bit rate in "Userbit" mode (Gbps).
Syntax 1	VARIABLE:VALUE "pam4", "dataratermode [auto|user|userbit]"
Syntax 2	VARIABLE:VALUE "pam4", "dataratemode?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "dataratemode user"
Example 2	VARIABLE:VALUE "pam4", "dataratemode?" VARIABLE:VALUE? "pam4" (query returns "OK:user")

VARIABLE:VALUE "pam4","exporterrors <filename>"

VARIABLE:VALUE "pam4","exporterrors <filename>"	Instructs the PAM4 application to export an ASCII text file listing all symbol errors from the current analysis cycle, if a repeating pattern is detected in the waveform. The file has comma-separated values arranged in four columns: • Column 1: index of erroneous symbol (with index 1 representing the first symbol after clock recovery begins) • Column 2: expected symbol at this location • Column 3: actual symbol detected at this location • Column 4: time (at center of symbol) where this error occurred If the filename doesn't have an extension, ".txt" will be appended.
Syntax	VARIABLE:VALUE "pam4","exporterrors<filename>"
Example	VARIABLE:VALUE "pam4","exportpattern c:\temp\test.txt"

VARIABLE:VALUE "pam4","exportpatternon <value>"

VARIABLE:VALUE "pam4","exportpatternon <value>	Instructs the PAM4 application to enable/disable the export of detected pattern during analysis to a text file.If this option is enabled, during analysis the pattern will be exported to file called "DetectedPattern.txt" in "C:\Users\<Current user>\Tektronix\TekApplications\PAM4\Patterns" Export Pattern Checkbox (Enabled by default).
Syntax 1	VARIABLE:VALUE "pam4", "exportpatternon [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "exportpatternon?"
Example 1	VARIABLE:VALUE "pam4", "exportpatternon 1"
Example 2	VARIABLE:VALUE "pam4", "exportpatternon?

VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4","exportpattern <filename>"

VARIABLE:VALUE "pam4","exportpattern <filename>"

Instructs the PAM4 application to export an ASCII text file listing one cycle of the detected repeating pattern, as a comma-separated row vector using the symbols {0, 1, 2, 3}. If the filename does not have an extension, extension ".txt" will be appended. The symbol pattern can be saved as ".txt" or ".csv" file. If the application doesn't detect repeating patterns, the file will contain all symbols within the range of clock recovery.

Syntax

VARIABLE:VALUE "pam4","exportpattern <filename>"

Example

VARIABLE:VALUE "pam4","exportpattern c:\temp\test.txt"

VARIABLE:VALUE? "pam4" (query returns "ERROR: no pattern detect") if there is no pattern found.

VARIABLE:VALUE "pam4","importpattern <filename>"

VARIABLE:VALUE "pam4","importpattern <filename>"

Instructs the PAM4 application to load the pattern file. The file will be used when the pattern control is in custom mode. The loaded pattern file is in *.txt file format.

By query using VARIABLE:VALUe importpattern?", it will return the import file name.

Syntax

VARIABLE:VALUE "pam4","importpattern <filename>"

Example 1

VARIABLE:VALUE "pam4","importpattern c:\temp\test.txt"

Example 2

VARIABLE:VALUE "pam4","importpattern? "

VARIABLE:VALUE? "pam4" (query returns "OK c:\temp\test.txt")

VARIABLE:VALUE "pam4", "patterncontrol <value>"

VARIABLE:VALUE "pam4", "patterncontrol <value>"

Sets the pattern control mode to Auto, Standard, or Custom. When the Config button is clicked on the Main Setup panel, it is displayed in a popup menu. The application detects pattern length automatically in Auto mode. The user selects the pattern type in Standard mode. The user loads a pattern file in Custom mode.

Syntax 1

VARIABLE:VALUE "pam4", "patterncontrol [auto|standard|custom]"

Syntax 2

VARIABLE:VALUE "pam4", "patterncontrol?" followed by query

VARIABLE:VALUE? "pam4"

Example 1

VARIABLE:VALUE "pam4", "patterncontrol auto"

Example 2

VARIABLE:VALUE "pam4", "patterncontrol?"

VARIABLE:VALUE? "pam4" (query returns "OK:auto")

VARIABLE:VALUE "pam4", "patternstandard?"

VARIABLE:VALUE "pam4", "patternstandard?"

Note: This feature is currently unavailable.

Uses the VARIABLE:VALUE? "PAM4" query to get the parameters afterwards; it returns the standard pattern type.

Syntax

VARIABLE:VALUE "pam4", "patternstandard?

Example	VARIABLE:VALUE "pam4", "patternstandard?" VARIABLE:VALUE? "pam4" (query returns "OK: None")

VARIABLE:VALUE "pam4", "tdecqffetapnum <value>"

VARIABLE:VALUE "pam4", "tdecqffetapnum <value>"	Sets the TDECQ FFE tap number. According to the specification, the recommended value is five.
Syntax 1	VARIABLE:VALUE "pam4", "tdecqffetapnum <value>"
Syntax 2	VARIABLE:VALUE "pam4", "tdecqffetapnum?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "tdecqffetapnum 5"
Example 2	VARIABLE:VALUE "pam4", "tdecqffetapnum?" VARIABLE:VALUE? "pam4" (query returns "OK: 5")

VARIABLE:VALUE "pam4", "tdecqffetapui <value>"

VARIABLE:VALUE "pam4", "tdecqffetapui <value>"	Sets the TDECQ FFE taps per UI. According to the specification, the recommended value is one.
Syntax 1	VARIABLE:VALUE "pam4", "tdecqffetapui <value>"
Syntax 2	VARIABLE:VALUE "pam4", "tdecqffetapui?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "tdecqffetapui 2"
Example 2	VARIABLE:VALUE "pam4", "tdecqffetapui?" VARIABLE:VALUE? "pam4" (query returns "OK: 2")

VARIABLE:VALUE "pam4", "tdecqffeautoset <value>"

VARIABLE:VALUE "pam4", "tdecqffeautoset <value>"	This command turns on/off FFE autoset for TDECQ. The value is zero or one.
Syntax 1	VARIABLE:VALUE "pam4", "tdecqffeautoset <value>"
Syntax 2	VARIABLE:VALUE "pam4", "tdecqffeautoset?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", " tdecqffeautoset1"
Example 2	VARIABLE:VALUE "pam4", " tdecqffeautoset?"

VARIABLE:VALUE "pam4", "tdecqffetapvalues?"

VARIABLE:VALUE "pam4", "tdecqffetapvalues?"	This command sets the TDECQ FFE tap values as imported from a text file containing multiple rows, each of which corresponds to a tap value. If the number of rows is not equal to the number of FFE taps, an error message will be returned. In query mode, the TDECQ FFE tap values displayed in the FFE configuration window are returned. Due to length limitation of a returned string, only the first 140 characters will be displayed. If no tap values are available or TDECQ is disabled, an error message will be returned.
Syntax 1	VARIABLE:VALUE "pam4", "tdecqffetapvalues <filename>"
Syntax 2	VARIABLE:VALUE "pam4", "tdecqffetapvalues?" followed by query VARIABLE:VALUE? "pam4"

Example 1	VARIABLE:VALUE "pam4", "tdecqffetapvalues C:\temp\filename.txt"
Example 2	VARIABLE:VALUE "pam4", " tdecqffetapvalues?"
	VARIABLE:VALUE? "pam4" (query returns "OK:0 0 1 0 0")

VARIABLE:VALUE "pam4", "tdecqsamplesui <value>"

VARIABLE:VALUE "pam4", "tdecqsamplesui <value>"	Sets the TDECQ waveform samples per UI. This value should be between 10 and 200. The recommended setting is 10.
Syntax 1	VARIABLE:VALUE "pam4", "tdecqsamplesui <value>"
Syntax 2	VARIABLE:VALUE "pam4", "tdecqsamplesui?" followed by query
	VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "tdecqsamplesui 10"
Example 2	VARIABLE:VALUE "pam4", "tdecqsamplesui?"
	VARIABLE:VALUE? "pam4" (query returns "OK: 10")

VARIABLE:VALUE "pam4", "tdecqmaxprecursor <value>"

VARIABLE:VALUE "pam4", "tdecqmaxprecursor <value>"	Sets TDECQ FFE max-precursor value. The recommended value is 2.
Syntax 1	VARIABLE:VALUE "pam4", "tdecqmaxprecursor <value>"
Syntax 2	VARIABLE:VALUE "pam4", "tdecqmaxprecursor?" followed by query
	VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "tdecqmaxprecursor 2"
Example 2	VARIABLE:VALUE "pam4", "tdecqmaxprecursor?"
	VARIABLE:VALUE? "pam4" (query returns "OK: 2")

VARIABLE:VALUE "pam4", "tdecqscopenoise <value>"

VARIABLE:VALUE "pam4", "tdecqscopenoise <value>"	Sets the oscilloscope noise in µW for TDECQ, RINxOMA, OMA - TDECQ, and Eye Height. The value should be larger than zero and smaller than 1000.
Syntax 1	VARIABLE:VALUE "pam4", "tdecqscopenoise <value>"
Syntax 2	VARIABLE:VALUE "pam4", "tdecqscopenoise?" followed by query
	VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "tdecqscopenoise 2"
Example 2	VARIABLE:VALUE "pam4", "tdecqscopenoise?"
	VARIABLE:VALUE? "pam4" (query returns "OK: 2")

VARIABLE:VALUE "pam4", "tdecqextendedsearch <value>"

VARIABLE:VALUE "pam4", "tdecqextendedsearch <value>"	Enables or disables extended search for TDECQ. The value can be 1 or 0. One enables extended search.

Syntax 1	VARIABLE:VALUE "pam4", "tdecqextendedsearch [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "tdecqextendedsearch?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "tdecqextendedsearch 1"
Example 2	VARIABLE:VALUE "pam4", "tdecqextendedsearch?" VARIABLE:VALUE? "pam4" (query returns "OK: 1")

VARIABLE:VALUE "pam4", "tdecqverticaladjust <value>"

VARIABLE:VALUE "pam4", "tdecqverticaladjust <value>"	Enables or disables vertical threshold adjustment for TDECQ. The value can be 1 or 0. One enables vertical threshold adjustment.
Syntax 1	VARIABLE:VALUE "pam4", "tdecqverticaladjust [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "tdecqverticaladjust?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "tdecqverticaladjust 1"
Example 2	VARIABLE:VALUE "pam4", "tdecqverticaladjust?" VARIABLE:VALUE? "pam4" (query returns "OK: 1")

VARIABLE:VALUE "pam4", "select <mnames>"

VARIABLE:VALUE "pam4", "select <mnames>"	Selects the measurement for the tree. The measurement category could be for Electrical or Optical depending on the probe type. (See *Measurement names lists* for values.)
Syntax	VARIABLE:VALUE "pam4", "select <mnames>"
Example	VARIABLE:VALUE "pam4", "select R_LM"

VARIABLE:VALUE "pam4", "deselect <mnames>"

VARIABLE:VALUE "pam4", "deselect <mnames>"	Deselects the measurement from the tree. The measurement category could be for Electrical or Optical depending on the probe type. (See *Measurement names lists* for values.)
Syntax	VARIABLE:VALUE "pam4", "deselect <mnames>"
Example	VARIABLE:VALUE "pam4", "deselect R_LM"

VARIABLE:VALUE "pam4", "isselect <mnames>"

VARIABLE:VALUE "pam4", "isselect <mnames>"	Queries if the measurement is selected from the tree or not. The measurement category could be for Electrical or Optical depending on the probe type. If the specified measurement is selected on the tree, it returns one, otherwise it returns zero. (See *Measurement names lists* for values.)
Syntax	VARIABLE:VALUE "pam4", "isselect <mnames>"
Example	VARIABLE:VALUE "pam4", "isselect SNDR?" followed by query VARIABLE:VALUE? "pam4" (query returns "OK:1”)

VARIABLE:VALUE "pam4", "pcie_jitterber <value>"

VARIABLE:VALUE "pam4", "pcie_jitterber <value>"	Sets the jitter BER rate in scientific format for all PCIe jitter measurements.
Syntax 1	VARIABLE:VALUE "pam4", "pcie_jitterber <value>"
Syntax 2	VARIABLE:VALUE "pam4", "pcie_jitterber?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", " pcie_jitterber 1e-08"
Example 2	VARIABLE:VALUE "pam4", "pcie_jitterber?" VARIABLE:VALUE? "pam4" (query returns "OK:1e-08")

VARIABLE:VALUE "pam4", "pcie_jitterdebugplotson <value>"

VARIABLE:VALUE "pam4", "pcie_jitterdebugplotson <value>"	Allows the PCIe Jitter measurements to show more debug plots. The values can be 1 or 0. One means the debug plots are shown, zero means they are not.
Syntax 1	VARIABLE:VALUE "pam4", " pcie_jitterdebugplotson [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "pcie_jitterdebugplotson?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "pcie_jitterdebugplotson 1"
Example 2	VARIABLE:VALUE "pam4", "pcie_jitterdebugplotson?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "pcie_sndron <value>"

VARIABLE:VALUE "pam4", "pcie_sndron <value>"	Enable SNDR (PCIe). The values can be 1 or 0. One means PCIe SNDR is on, zero means it is off.
Syntax 1	VARIABLE:VALUE "pam4", " pcie_sndron [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "pcie_sndron?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "pcie_sndron 1"
Example 2	VARIABLE:VALUE "pam4", "pcie_sndron?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "pcie_sndrsamplenumber <value>"

VARIABLE:VALUE "pam4", "pcie_sndrsamplenumber <value>"	Sets the SNDR (PCIe) samples per symbol (M). The value is in the range [32 200].
Syntax 1	VARIABLE:VALUE "pam4", "pcie_sndrsamplenumber <value>"
Syntax 2	VARIABLE:VALUE "pam4", "pcie_sndrsamplenumber?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "pcie_sndrsamplenumber 32"
Example 2	VARIABLE:VALUE "pam4", "pcie_sndrsamplenumber?" VARIABLE:VALUE? "pam4" (query returns "OK:32")

VARIABLE:VALUE "pam4", " pcie_sndrfilterlength <value>"

VARIABLE:VALUE "pam4", " pcie_sndrfilterlength <value>"	Sets the SNDR (PCIe) linear fit pulse filter length (Np). The value is in the range [2 10000].
Syntax 1	VARIABLE:VALUE "pam4", " pcie_sndrfilterlength <value>"
Syntax 2	VARIABLE:VALUE "pam4", "pcie_sndrfilterlength?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", " pcie_sndrfilterlength 64"
Example 2	VARIABLE:VALUE "pam4", "pcie_sndrfilterlength?" VARIABLE:VALUE? "pam4" (query returns "OK:64")

VARIABLE:VALUE "pam4", "pcie_sndrdelay <value>"

VARIABLE:VALUE "pam4", "pcie_sndrdelay <value>"	Sets the SNDR (PCIe) linear fit pulse filter delay (Dp) in symbols. The value is in the range [1 100].
Syntax 1	VARIABLE:VALUE "pam4", " pcie_sndrdelay <value>"
Syntax 2	VARIABLE:VALUE "pam4", "pcie_sndrdelay?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", " pcie_sndrdelay 65"
Example 2	VARIABLE:VALUE "pam4", "pcie_sndrdelay?" VARIABLE:VALUE? "pam4" (query returns "OK:65")

VARIABLE:VALUE "pam4", "pcie_sndrdebugplotson [1|0]"

VARIABLE:VALUE "pam4", "pcie_sndrdebugplotson [1|0]"	Allows the SNDR (PCIe) to show more debug plots. The values can be 1 or 0. One means the debug plots are shown, zero means they are not.
Syntax 1	VARIABLE:VALUE "pam4", "pcie_sndrdebugplotson [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "pcie_sndrdebugplotson?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "pcie_sndrdebugplotson 1"
Example 2	VARIABLE:VALUE "pam4", "pcie_sndrdebugplotson?" VARIABLE:VALUE? "pam4" (query returns "OK:1")

VARIABLE:VALUE "pam4", "isiresdp <value>"

VARIABLE:VALUE "pam4", "isiresdp <value>"	Sets the linear fit pulse filter delay (Dp) for residual ISI computation. The value is in the range [2 100].
Syntax 1	variable:value "pam4", "isiresdp <value>"
Syntax 2	VARIABLE:VALUE "pam4", " isiresdp?” followed by query VARIABLE:VALUE "pam4"
Example 1	variable:value "pam4", "isiresdp 4"
Example 2	VARIABLE:VALUE "pam4", " isiresdp?” VARIABLE:VALUE "pam4" (query returns "OK:4") followed by query

VARIABLE:VALUE "pam4", "isiresnp <value>"

VARIABLE:VALUE "pam4", "isiresnp <value>"	Sets the linear fit pulse filter length (Np) for residual ISI computation. The value is in the range [2 10000].
Syntax 1	VARIABLE:VALUE "pam4", "isiresnp <value>"
Syntax 2	variable:value "pam4", " isiresnp?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "isiresnp 64"
Example 2	variable:value "pam4", " isiresnp?" VARIABLE:VALUE? "pam4" (query returns "OK:64") followed by query

VARIABLE:VALUE "pam4", "weightedwindowtype <value>"

VARIABLE:VALUE "pam4", "weightedwindowtype <value>	Sets the Weighting Window Type in the Statistical Eye Analysis sub panel. Selections are “None”, “Rectangular”, “Gaussian”
Syntax 1	VARIABLE:VALUE "pam4", "weightedwindowtype <None|Rectangular|Gaussian>"
Syntax 2	VARIABLE:VALUE "pam4", "weightedwindowtype?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "weightedwindowtype None"
Example 2	VARIABLE:VALUE "pam4", " weightedwindowtype?" VARIABLE:VALUE? "pam4" (query returns "OK:None")

VARIABLE:VALUE "pam4", "enablereferencereceiverieee802d3ck <value>"

VARIABLE:VALUE "pam4", "enablereferencereceiverieee802d3ck <value>""	Compute eye opening including the effect of a reference receiver as per IEEE802.3 Annex 120G.5 using the configured CTLE and RX filters. Turns this option on/off. The value can be zero or one. One means options is on, zero means it is off.
Syntax 1	VARIABLE:VALUE "pam4", "enablereferencereceiverieee802d3ck [1|0]"
Syntax 2	VARIABLE:VALUE "pam4", "enablereferencereceiverieee802d3ck?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE "pam4", "enablereferencereceiverieee802d3ck 1"
Example 2	VARIABLE:VALUE "pam4", "enablereferencereceiverieee802d3ck?" VARIABLE:VALUE? "pam4" (query returns "OK:1") followed by query

VARIABLE:VALUE "pam4", " fixtures4pfile<filename>"

VARIABLE:VALUE "pam4", "results fixtures4pfile<file name >"	Querying and setting “fixtures4pfile”
Syntax 1	VARIABLE:VALUE "pam4", "results fixtures4pfile<file name >"
Syntax 2	VARIABLE:VALUE "pam4","fixtures4pfile?" followed by query VARIABLE:VALUE? "pam4"
Example 1	VARIABLE:VALUE “pam4”,"fixtures4pfile c:\temp\test.txt"
Example 2	VARIABLE:VALUE "pam4", "fixtures4pfile?" VARIABLE:VALUE? "pam4" (query returns "OK:c:\temp\test.txt")

Measurement names <mnames> tables

Measurement names tables

Table 1: Electrical measurement names

Tag <mnames>	Explanation
SymErrors	Symbol Errors
SER	Symbol Error Ratio (actual)
BER	Bit Error Ratio (actual)
R_LM	Linearity
RJDJ	Jitter: RJ / DJ / TJ@BER
VEC	Vertical Eye Closure (VEC)
EW	EW depends on target BER (currently BER 10^{-5})
EH	EH depends on target BER (currently BER 10^{-5})
Vupp	Vertical measurement of upper eye
Hupp	Horizontal measurement of upper eye
Vmid	Vertical measurement of mid eye
Hmid	Horizontal measurement of mid eye
Vlow	Vertical measurement of lower eye
Hlow	Horizontal measurement of lower eye
SNDR	SNDR measurement value
Pmax	SNDR measurement Pmax
SigmaE	SNDR measurement σ_E
SigmaN	SNDR measurement σ_N
SigmaN_3	SigmaN per level
SigmaN_2	
SigmaN_1	
SigmaN_0	
UUGJ	UUGJ (rms)
UBHPJ	UBHPJ (p-p)
CEI_EOJ	EOJ
Jrms	Jrms
J3u	J3u (IEEE) jitter measurement
J3u03	$J3u_{03,}$ which is J3u measured on R03 and F30 transitions
J4u	J4u
J4u03	$J4u_{03,}$ which is J4u measured on R03 and F30 transitions
IEEE_EOJ	EOJ (IEEE)

Table continued…

Tag <mnames>	Explanation
IEEE_EOJ_01	EOJ measurements for each of the twelve possible transitions
IEEE_EOJ_02	
IEEE_EOJ_03	
IEEE_EOJ_12	
IEEE_EOJ_13	
IEEE_EOJ_23	
IEEE_EOJ_32	
IEEE_EOJ_31	
IEEE_EOJ_30	
IEEE_EOJ_21	
IEEE_EOJ_20	
IEEE_EOJ_10	
IEEE_Rise	Rise Time (IEEE)
IEEE_Fall	Fall Time (IEEE)
SNR_ISI	SNR_{ISI}
VMA	VMA
VDI	$V_{di(pk-pk)}$
Rise_Fall	Rise & Fall (All Edges)
CTLE_Optimization	CTLE Optimization
Correlated_Waveform_Analysis	Correlated waveform analysis

Table 2: Optical measurement names

Tag <mnames>	Explanation
SymErrors	Symbol Errors
SER	Symbol Error Ratio (actual)
BER	Bit Error Ratio (actual)
R_LM	Linearity
RJDJ	Jitter: RJ / DJ / TJ@BER
VEC	Vertical Eye Closure (VEC)
EW	EW depends on target BER (currently BER 10^{-5})
EH	EH depends on target BER (currently BER 10^{-5})
Vupp	Vertical measurement of upper eye
Hupp	Horizontal measurement of upper eye
Vmid	Vertical measurement of mid eye
Hmid	Horizontal measurement of mid eye
Vlow	Vertical measurement of lower eye
Hlow	Horizontal measurement of lower eye

Table continued…

Tag <mnames>	Explanation
ER	Extinction Ratio
OMA_{outer}	Outer Optical Modulation Amplitude
OMA_{outer}–TDECQ	Outer Optical Modulation Amplitude after TDECQ FFE
AOP	Average Optical Power
C_{eq}	Equalizer noise enhancement coefficient.
TDECQ	TDECQ
Rise_Fall	Rise & Fall (All Edges)
Correlated_Waveform_Analysis	Correlated waveform analysis

Table 3: PCIe measurement names

Tag <mnames>	Explanation	Result Query
PCIe_TTX_UTJ	T_{TX-uTJ}	variable:value "pam4","results <html>T_{TX-uTJ} (PCIe)</html>?"
PCIe_TTX_UDJDD	T_{TX-RJ}	variable:value "pam4","results <html>T_{TX-uDJDD} (PCIe)</html>?"
PCIe_TTX_RJ	$T_{TX-uDJdd}$	variable:value "pam4","results <html>T_{TX-RJ} (PCIe)</html>?"
PCIe_TTX_UPW_TJ	$T_{TX-UPW-TJ}$	variable:value "pam4","results <html>T_{TX-UPW-TJ} (PCIe)</html>?"
PCIe_TTX_UPW_DJDD	$T_{TX-UPW-DJDD}$	variable:value "pam4","results <html>T_{TX-UPW-DJDD} (PCIe)</html>?"
PCIe_JNu	JNu(PCIe)	variable:value "pam4","results JNu (PCIe)?"
PCIe_Jrms	Jrms(PCIe)	variable:value "pam4","results Jrms (PCIe)?"
PCIe_VTX_ DIFF_PP	$V_{TX-DIFF-PP}$	variable:value "pam4","results <html>V_{TX-DIFF-PP}</html>?"
PCIe_VTX_EIEOS	$V_{TX-EIEOS}$	variable:value "pam4","results <html>V_{TX-EIEOS}</html>?"
Tx Preset Equalization	Preshoot 2, Preshoot 1, and De-emphasis for the current PCIe Gen6 Tx Equalization Preset, reported in dB.	VARIABLE:VALUE "pam4", "results Tx Preset Equalization?"
PCIe_R_LM	R_{LM}(PCIe)	variable:value "pam4","results <html>R_{LM} (PCIe)</html>?"
PCIe_SNDR	SNDR(PCIe)	variable:value "pam4","results SNDR (PCIe)?"
PCIe_Pmax	SNDR(PCIe) measurement Pmax	variable:value "pam4","results <html>P_{max} (PCIe)</html>?"
PCIe_SigmaE	SNDR(PCIe) measurement σE	variable:value "pam4","results <html>&#963_e (PCIe)</html>?"
PCIe_SigmaN	SNDR(PCIe) measurement σN	variable:value "pam4","results <html>&#963_n (PCIe) </html>?"

Table continued…

Tag <mnames>	Explanation	Result Query
PCIe_SigmaN_3	SigmaN(PCIe) per level	variable:value "pam4","results <html>&#963_{n(3)} (PCIe)</html>?"
PCIe_SigmaN_2		variable:value "pam4","results <html>&#963_{n(2)} (PCIe)</html>?"
PCIe_SigmaN_1		variable:value "pam4","results <html>&#963_{n(1)} (PCIe)</html>?"
PCIe_SigmaN_0		variable:value "pam4","results <html>&#963_{n(0)} (PCIe)</html>?"
PCIe_TTX_UPW_RJ	TTX-UPW-RJ	variable:value "pam4","results <html>T_{TX-UPW-RJ} (PCIe)</html>?"

Table 4: Additional 400GCK measurement names

Tag <mnames>	Explanation	Result Query
VMA	Voltage Modulation Amplitude	VMA
$V_{di(pk-pk)}$	Peak to peak differential voltage	VARIABLE:VALUE "pam4","results <html>V_{di} (pk-pk)</html>?"
ISI_RES	Residual ISI	VARIABLE:VALUE “pam4”,”results <html>ISI_{RES}</html>?”
dv_f	Difference steady state voltage	VARIABLE:VALUE "pam4","results <html>dv_f</html>?"
dR_{peak}	Difference pulse peak ratio	VARIABLE:VALUE "pam4","results <html>dR_{peak}</html>?"
CTLE_Optimization	CTLE Optimization	VARIABLE:VALUE "pam4","results CTLE Optimization?"

Sample GPIB script (Python 3.5)

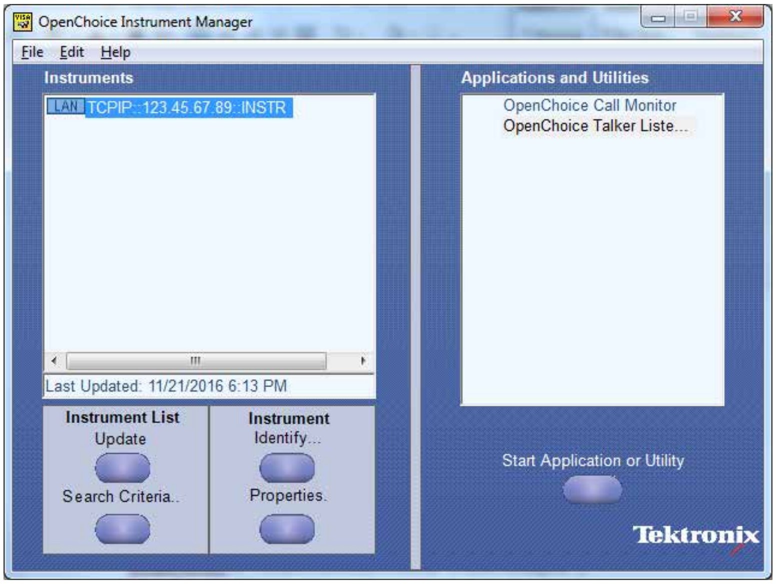
About this task

This example demonstrates using the PAMJET Transmitter Analysis Software.

It is important to insert 0.5 to 1 second delay between consecutive commands during scripting. This gives PAMJET Transmitter Analysis and TekVISA sufficient time to establish reliable communication.

Procedure

1. Recall "32G_PRBS7_9MRL.wfm" into scope memory REF1.
2. Obtain the VISA address of the target instrument. This can be viewed in OpenChoice Instrument Manager:



3. Execute the following Python script (using the instrument VISA address obtained in Step 2):

```
# python_pam4demo.py
# Use Python scripting to automate PAMJET Transmitter Analysis
import time
import visa

# Start VISA session with the instrument
visa_address = 'TCPIP::123.45.67.89::INSTR'
rm = visa.ResourceManager()
scope = rm.open_resource(visa_address)
scope.timeout = 60000
print(scope.query('*idn?').strip())
waittime = 1

# Start PAMJET Transmitter Analysis
scope.write('APPLICATION:ACTIVATE "PAM Analysis (PAMJET)"')
time.sleep(waittime)
while(scope.query('variable:value?
"pam4"').strip()!='"OK"'):
        time.sleep(5)

# Set waveform source
scope.write('variable:value "pam4","datasource ref1"')
time.sleep(waittime)
scope.write('variable:value "pam4","datasource?"')
time.sleep(waittime)
while(scope.query('variable:value? "pam4"').strip()!='"OK:REF1"'):
        time.sleep(waittime)
        scope.write('variable:value "pam4","datasource?"')
        time.sleep(waittime)

# Set data rate
scope.write('variable:value "pam4","nominalsymbolrate 32"')
time.sleep(waittime)
```

```
scope.write('variable:value "pam4","nominalsymbolrate?"')
time.sleep(waittime)
while(scope.query('variable:value? "pam4"').strip()!='"OK:32000000000"'):
            time.sleep(waittime)
            scope.write('variable:value "pam4","nominalsymbolrate?"')
            time.sleep(waittime)

# Set CTLE
## Turn on CTLE
scope.write('variable:value "pam4","ctleon 1"')
time.sleep(waittime)
scope.write('variable:value "pam4","ctleon?"')
time.sleep(waittime)
while(scope.query('variable:value? "pam4"').strip()!='"OK:1"'):
            time.sleep(waittime)
            scope.write('variable:value "pam4","ctleon?"')
            time.sleep(waittime)

## Set CTLE type
scope.write('variable:value "pam4","ctletype OIF-CEI Gen I"')
time.sleep(waittime)
scope.write('variable:value "pam4","ctletype?"')
time.sleep(waittime)
while(scope.query('variable:value? "pam4"').strip()!='"OK:OIF-CEI Gen I"'):
            time.sleep(waittime)
            scope.write('variable:value "pam4","ctletype?"')
            time.sleep(waittime)

## Set CTLE preset
scope.write('variable:value "pam4","ctlepreset 2 dB"')
time.sleep(waittime)
scope.write('variable:value "pam4","ctlepreset?"')
time.sleep(waittime)
while(scope.query('variable:value? "pam4"').strip()!='"OK:2 dB"'):
            time.sleep(waittime)
            scope.write('variable:value "pam4","ctlepreset?"')
            time.sleep(waittime)

# Run Single
scope.write('variable:value "pam4","single"')
time.sleep(waittime)
scope.write('variable:value "pam4","state?"')
time.sleep(waittime)
while(scope.query('variable:value? "pam4"').strip()!='"OK:stop"'):
            time.sleep(waittime)
            scope.write('variable:value "pam4","state?"')
            time.sleep(waittime)

# Exit
scope.write('variable:value "pam4","exit"')
time.sleep(waittime)
wait = input("PRESS ENTER TO FINISH...")
scope.close()
```

```
rm.close()
print("\nEnd of demonstration.")
```

Glossary

BER	Bit Error Rate or Ratio. A statistical measure of the frequency of bit errors, expressed as a ratio between error bits and total bits.
Correlated (averaged) waveform	The correlated or averaged waveform is only defined for waveforms with repeating patterns of bits or symbols. It is a single repeat of the pattern sequence that shows only the deterministic effects (ISI) without any effects due to uncorrelated impairments such as random jitter or noise, or periodic effects at rates not integrally related to the bit rate or pattern rate. Conceptually, the correlated waveform for a long real-time waveform with a repeating pattern of PAMJET symbols is found as follows: **1.** The waveform is separated into segments, each holding one full pattern-repeat; **2.** The segments are overlaid so that the symbols align and the recovered clock times exactly coincide; **3.** At each (cyclostationary) instant in time across the duration of the repeat, the mean value across all the segments is computed and becomes the correlated waveform value at that instant. The actual process must account for time-variant symbol rates and for oscilloscope samples that do not coincide.
CTLE	Continuous Time Linear Equalizer: a linear equalization method that uses frequency-domain amplitude and phase shaping to counter the effects of a lossy channel or other impairments. It does not require clock or symbol recovery.
DFE	Decision Feedback Equalizer: A non-linear equalization method that performs clock recovery and uses the recovered states from several prior symbols to predictively correct the next symbol.
DPOJET	A Tektronix analysis application that performs Jitter, Eye diagram, and Timing measurements on DPO (and other) series oscilloscopes.
Equalizer	A circuit designed to compensate (equalize) the detrimental effects of a transmission channel. Three types of equalizers can be modeled: CTLE (continuous-time linear equalizer), FFE (feed-forward equalizer) and DFE (decision-feedback equalizer).
FFE	Feed Forward Equalizer: A linear equalization method that uses FIR with spaced taps to counter the effects of a lossy channel or other impairments. It can be performed with and without clock recovery.
ISI	Inter-Symbol Interference: A form of waveform distortion in which events (typically waveform transitions) prior to a point in time cause time-delayed distortions that add or subtract to affect that point.
PAM	Pulse Amplitude Modulation, in which a signal with a nominally constant symbol rate uses one discrete amplitude level during each symbol interval to represent one or more bits.
PAM4	Pulse Amplitude Modulation using four possible voltage levels. Since each symbol can have one of four voltage, it can carry two binary bits of information (conceptually, 00, 01, 10, or 11).
SDLA	Serial Data Link Analyzer, a Tektronix analysis application for advanced modeling and simulation of serial data waveforms, supporting fixture de-embedding, channel embedding and equalization.
SNDR	Signal-to-Noise-and-Distortion Ratio: A set of measurements defined by 100G CR4 and KR4 standards. SNDR is the ratio between the square of the linear fit pulse peak and the sum of squares of linear fit error and vertical noise.
UI	Unit interval; the duration of one transmitted symbol which carries two bits of information.
PAMJET	A Tektronix analysis application that performs Jitter, Eye diagram, and Timing measurements on PAM4 signal acquired on Tektronix real time oscilloscope.