Keysight IEEE MultiGBASE TX EQ Training

User Guide

KEYSIGHT TECHNOLOGIES

Notices

Manual Part Number

M8070-91020

Edition

Edition 3.0, March 2019

Keysight Technologies Deutschland GmbH
Herrenberger Strasse 130,
71034 Böblingen, Germany

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

U.S. Government Rights

The Software is “commercial computer software,” as defined by Federal Acquisition Regulation (“FAR”) 2.101. Pursuant to FAR 12.212 and 27.405-3 and Department of Defense FAR Supplement

(“DFARS”) 227.7202, the U.S. government acquires commercial computer software under the same terms by which the software is customarily provided to the public. Accordingly, Keysight provides the Software to U.S. government customers under its standard commercial license, which is embodied in its End User License Agreement (EULA), a copy of which can be found at http://www.keysight.com/find/sweula. The license set forth in the EULA represents the exclusive authority by which the U.S. government may use, modify, distribute, or disclose the Software. The EULA and the license set forth therein, does not require or permit, among other things, that Keysight: (1) Furnish technical information related to commercial computer software or commercial computer software documentation that is not customarily provided to the public; or (2) Relinquish to, or otherwise provide, the government rights in excess of these rights customarily provided to the public to use, modify, reproduce, release, perform, display, or disclose commercial computer software or commercial computer software documentation. No additional government requirements beyond those set forth in the EULA shall apply, except to the extent that those terms, rights, or licenses are explicitly required from all providers of commercial computer software pursuant to the FAR and the DFARS and are set forth specifically in writing elsewhere in the EULA. Keysight shall be under no obligation to update, revise or otherwise modify the Software. With respect to any technical data as defined by FAR 2.101, pursuant to FAR 12.211 and 27.404.2 and DFARS 227.7102, the U.S. government acquires no greater than Limited Rights as defined in FAR 27.401 or DFAR 227.7103-5 (c), as applicable in any technical data.

Warranty

THE MATERIAL CONTAINED IN THIS DOCUMENT IS PROVIDED "AS IS," AND IS SUBJECT TO BEING CHANGED, WITHOUT NOTICE, IN FUTURE EDITIONS. FURTHER, TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, KEYSIGHT DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED WITH REGARD TO THIS MANUAL AND ANY INFORMATION CONTAINED HEREIN, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. KEYSIGHT SHALL NOT BE LIABLE FOR ERRORS OR FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, USE, OR PERFORMANCE OF THIS DOCUMENT OR ANY INFORMATION CONTAINED HEREIN. SHOULD KEYSIGHT AND THE USER HAVE A SEPARATE WRITTEN AGREEMENT WITH WARRANTY TERMS COVERING THE MATERIAL IN THIS DOCUMENT THAT CONFLICT WITH THESE TERMS, THE WARRANTY TERMS IN THE SEPARATE AGREEMENT WILL CONTROL.

Safety Notices

CAUTION

A CAUTION notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a CAUTION notice until the indicated conditions are fully understood and met.

WARNING

A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.

Contents

1 Introduction

This chapter describes the different technologies named “10GBASE-KR” and “25GBASE-KR and 100GBASE-KR4” for transmission and reception of data over backplane Ethernet with a speed of 10 Gbps, 25 Gbps and 100 Gbps respectively. All the three specifications are collectively called MultiGBASE.

Overview

MultiGBASE are computer networking standards and are defined in IEEE 802.3.

These TX EQ Training applications are installed as a plug-in to the M8070B software. These plug-ins enable to generate the signal according to the specification, to put the Device Under Test (DUT) into a state where the receiver tolerance can be tested.

The MultiGBASE plug-ins have been developed for testing receiver tolerance of MultiGBASE devices against impairments such as jitter and interference.
For achieving this, link equalization training is performed between the instruments generator and one receiver of a single DUT. After successfully finishing this procedure as defined in the standard, it is possible to transition without signal interruption to a user-defined sequence. This sequence can then be used for testing the impairment tolerance of the user device.

- 10GBASE-KR can be interpreted as:
  - 10G - Data rate is 10 Gbps
  - BASE - Modulation type is baseband
  - K - Backplane application
  - R - Scrambled coding (64B/66B)
  - Operating over one lane

- 25GBASE-KR can be interpreted as:
  - 25G - Data rate is 25Gbps
  - BASE - Modulation type is baseband
  - K - Backplane application
  - R - Scrambled coding (64B/66B)
  - Operating over one lane

- 100GBASE-KR4 can be interpreted as:
  - 100G - Data rate is 100Gbps
  - BASE - Modulation type is baseband
  - K - Backplane application
  - R - Scrambled coding (64B/66B)
  - Operating over four lanes

Terminologies associated with Technology

Backplane Ethernet

Backplane Ethernet is used in applications such as blade servers and modular routers/switches with upgradeable line cards. It enables inter-operable solutions within a chassis and supports 10GBASE-KR and 25GBASE-KR at a speed of 10 Gbps and 25 Gbps respectively, over a single link. For 100GBASE-KR4 the speed is 100Gbps over four lanes.

Transmitter Equalization (TX EQ) Training

It is a process to establish and optimize a communication channel between source and sink devices. Generally, the TX EQ training procedure also called startup protocol, is performed after link speed auto negotiation. The MultiGBASE TX EQ Training plug-ins do not support auto negotiation, therefore it is expected that the DUT can be set to the MultiGBASE operating mode by other means.

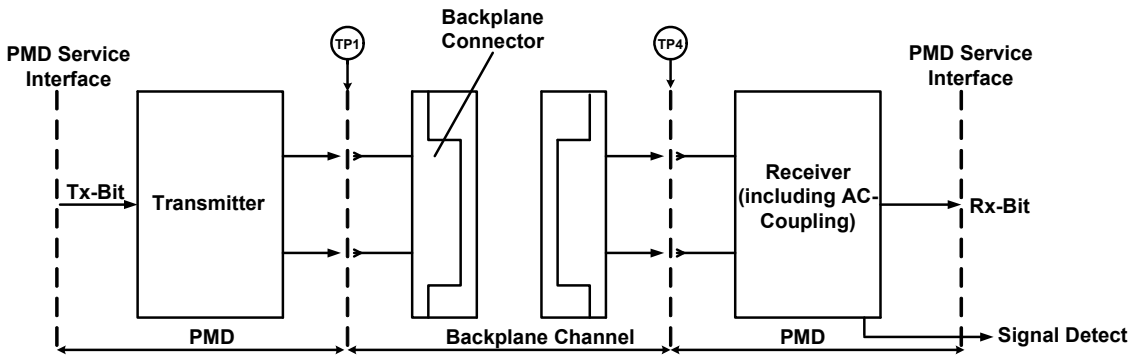


Figure 1 Link block diagram for 10GBASE-KR

The Physical Medium Dependent (PMD) sublayer defines the test points TP1 (TP0 for 25GBASE-KR) and TP4 (TP5 for 25GBASE-KR) for the purpose of system conformance testing as shown in Figure 1.

The PMD sub layer is a part of the physical layer that defines the details of transmission and reception of individual bits on a physical medium.

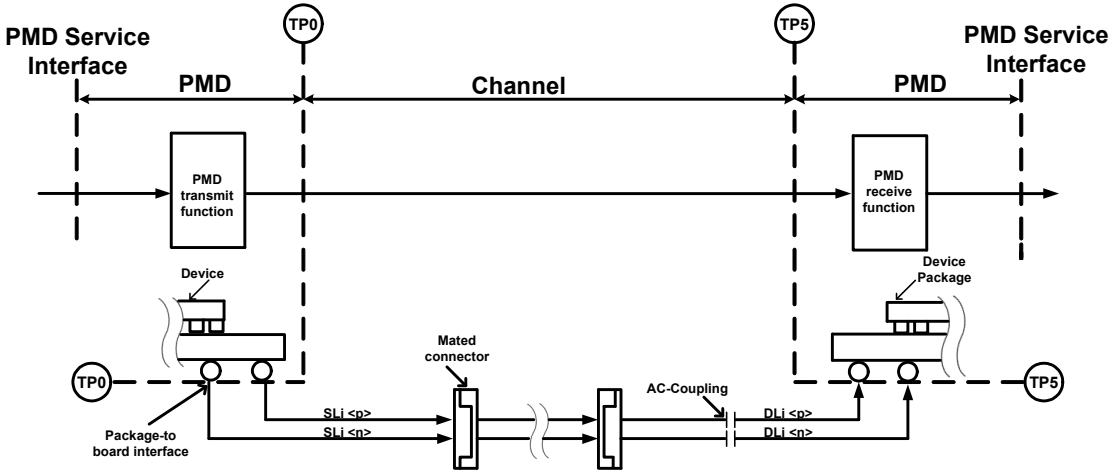


Figure 2 Link block diagram for 25GBASE-KR and 100GBASE-KR4 (one direction for one lane is illustrated)

- For 25GBASE-KR ‘i’ = 0, as it is single lane structure.
- For 100GBASE-KR4 ‘i’ goes from 0 to 3, as it is a four lane structure.

Basic Requirements for 10GBASE-KR TX EQ Training

Hardware Requirements

Required modules are:

- M8041A
- M8051A (cannot be operated without M8041A)

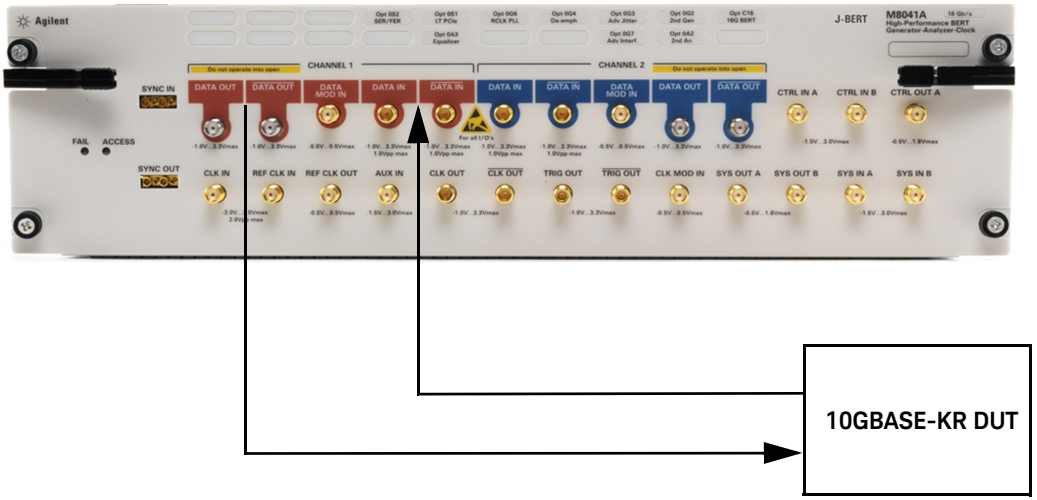


Figure 3 BERT connection setup with 10GBASE-KR DUT

NOTE **The 10GBASE-KR TX EQ Training plug-in requires M8041A or M8051A serial number - DE55300500 or modules with option 0G5 / UG5.**

1 Connect the probe/cable from DATA OUT (either from normal DATA OUT, complement DATA OUT or both, depending on the DUT type) of the M8041A to the DATA IN (receiver) of the 10GBASE-KR DUT.

2 Connect the probe/cable from DATA OUT (transmitter) of the 10GBASE-KR DUT to the DATA IN (either to normal DATA IN, complement DATA IN or both, depending on the DUT type) of the M8041A.

NOTE You are free to choose the connectors used by the 10GBASE-KR TX EQ Training plug-in. Connectors are represented as DATA IN and DATA OUT on the M8041A module.

Software Requirements

To install the 10GBASE-KR TX EQ Training plug-ins, the M8070B software (S6.0 or above) is required.

You can download the software from the following link:

http://www.keysight.com/find/M8070B

License/Option Requirements

The 10GBASE-KR TX EQ Training plug-in is a licensed feature. To enable them, the following licenses are required:

NOTE Some of the required licenses can be substituted by an upgraded version or alternative license combination.

Table 1 Required Licenses for M8041A

M8020A Structure	Description	M8020AU / M8062AU Structure	Trial Licenses	Prerequisites	Clock group-wide
M8041A-0G4	Multi-tap De-emphasis, Module-wide License	M8041A-UG4	M8041A-TG4	-	no
M8041A-0SX	10GBASE-KR Transmitter Equalization Training, Module-wide License	M8041A-USX	M8041A-TSX	ED1(16G)	no
M8041A-C16	BERT one Channel, Data Rate up to 16 Gb/s	-	-	-	no

Table 2 Required Licenses for M8051A

M8020A Structure	Description	M8020AU / M8062AU Structure	Trial Licenses	Prerequisites	Clock group-wide
M8051A-0G4	Multi-tap De-emphasis, Module-wide License	M8051A-UG4	M8051A-TG4	-	no
M8051A-0SX	10GBASE-KR Transmitter Equalization Training, Module-wide License	M8051A-USX	M8051A-TSX	ED1(16G)	no
M8051A-C16	BERT one Channel, Data Rate up to 16 Gb/s	-	-	-	no

Some licenses are optional that might be necessary for supporting device testing, depending on specific customer needs.

Table 3 Optional Licenses for M8041A

M8020A Structure	Description	M8020AU / M8062AU Structure	Trial Licenses	Prerequisites	Clock group-wide
M8041A-0G3	Advanced Jitter Sources for Receiver Characterization, Module-wide License	M8041A-UG3	M8041A-TG3	-	no
M8041A-0G5	Adjustable Inter-symbol Interference (ISI), Module-wide License	M8041A-UG5	M8041A-TG5	-	no
M8041A-0G7	Advanced Interference Sources for Receiver Characterization, Module-wide License	M8041A-UG7	M8041A-TG7	-	no

Table 4 Optional Licenses for M8051A

M8020A Structure	Description	M8020AU / M8062AU Structure	Trial Licenses	Prerequisites	Clock group-wide
M8051A-0G3	Advanced Jitter Sources for Receiver characterization, Module-wide License	M8051A-UG3	M8051A-TG3	-	no
M8051A-0G5	Adjustable Intersymbol Interference (ISI), Module-wide License	M8051A-UG5	M8051A-TG5	-	no
M8051A-0G7	Advanced Interference Sources for Receiver characterization, Module-wide License	M8051A-UG7	M8051A-TG7	-	no

Basic Requirements for 25GBASE-KR and 100GBASE-KR4 TX EQ Training

Hardware Requirements

Required modules are:

- M8041A
- M8062A

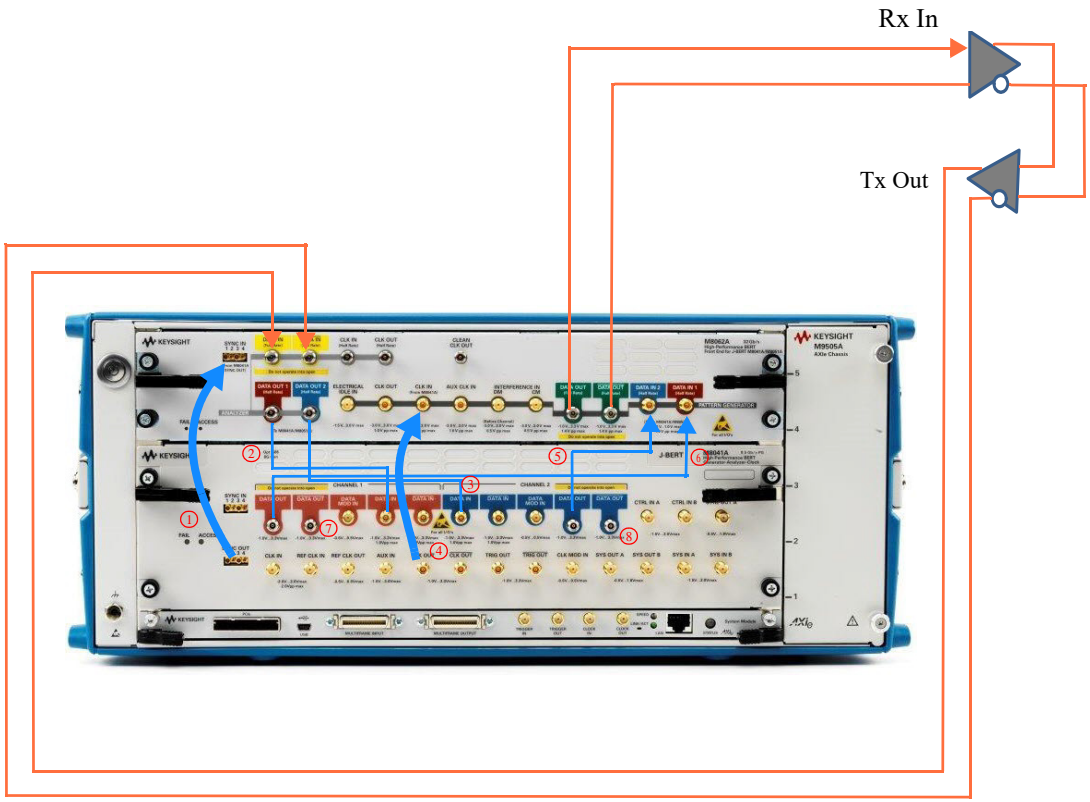


Figure 4 BERT Connection set-up with 25GBASE-KR and 100GBASE-KR4 DUT

Following are the setup of M8041A/M8062A connections:

1 M8041A SYNC OUT1 to M8062A SYNC IN 1

2 M8062A DATA OUT1 to M8041A DATA IN 1

3 M8062A DATA OUT 2 to M8041A DATA IN 2)

4 M8041A CLK OUT to M8062A CLK IN
5 M8041A DATA OUT 2 to M8062A DATA IN 2
6 M8041A DATA OUT 1 to M8062A DATA IN 1
7 Terminate M8041A DATA OUT 1 with 50 ohm
8 Terminate M8041A DATA OUT 2 with 50 ohm

Following are the setup of DUT connections:

- M8062A DATA OUTs to DUT Rx IN
- DUT TX OUTs to M8062A DATA IN

NOTE Generally, normal and complement inputs are considered as a single input or output.

Software Requirements

Same as described for 10GBASE-KR X EQ Training, see Software Requirements on page 12.

License/Option Requirements

The M8062A module requires the following options to be used with 25GBASE-KR and 100GBASE-KR4 TX EQ Training:

Table 5 Required Options for M8062A

Options	Module Features
M8062A-C32	32 Gb/s BERT Front End
M8062A-0G4	Multi-tap De-emphasis License
M8062A-0A4	Clock Recovery License

NOTE Only modules supporting the CDR feature can be used with this plug-in. To enable this feature, a hardware upgrade may also be required in modules with serial numbers MY55400300.

The following option can be useful for device characterization:

Table 6 Additional Options for M8062A

Options	Module Features
M8062A-0G5	Adjustable Intersymbol Interference (ISI) License

Interference Sources

M8062A does not contain interference sources but it is possible to supply differential and common mode interference signals from an external source using the respective inputs INTERFERENCE IN - DM and CM. Differential mode interference is coupled in prior to the internal ISI emulation while common mode interference is coupled in after the internal ISI emulation.

Similarly, the M8041A module requires the following options to drive an M8062A when being used with 25GBASE-KR and 100GBASE-KR4 TX EQ Training:

Table 7 Required Options for M8041A

Options	Module Features
M8041A-G16	Pattern Generator one Channel, Data Rate up to 16 Gb/s
M8041A-0G2	Second Channel for Pattern Generator
M8041A-0A2	Second Channel for Analyzer

The following option is not required for TX EQ Training but for performing Jitter Tolerance testing:

Table 8 Additional Option for M8041A

Options	Module Features
M8041A-0G3	Advanced Jitter Sources for Receiver Characterization

The M8062A module also requires the following licenses to be used with 25GBASE-KR and 100GBASE-KR4 TX EQ Training:

Table 9 Required Licenses for M8062A

License	Instrument Option	Description	Version no.
M8062A-OSC	OSC	100GBASE-KR4 and 25GBASE-KR Transmitter Equalization Training, module-wide License	1.0
M8062A-USC	USC	100GBASE-KR4 and 25GBASE-KR Transmitter Equalization Training, module-wide License	1.0
M8062A-TSC	TSC	100GBASE-KR4 and 25GBASE-KR Transmitter Equalization Training, module-wide License	1.0

Installing Plug-in

The MultiGBASE TX EQ Training plug-ins must be installed separately by the M8070B system software.

The installer for the MultiGBASE TX EQ Training plug-in can be downloaded from the following Keysight webpage:

www.keysight.com/IEEE 802.3 Transmitter Equalization Training

NOTE **The system must have M8070B software (version S6.0 or above) installed on it.**

Starting the TX EQ Training Plug-in

To access the installed TX EQ Training plug-in:

1 Click **Start** > **All Programs** > **Keysight M8070B**. The user interface for the M8070B system software appears.

2 From the M8070B user interface menu, click **Application** to view the list of all installed IEEE plug-ins.

3 For 10GBASE-KR TX EQ Training plug-in, select **IEEE** > **10GBASE-KR TX EQ Training** plug-in.

The 10GBASE-KR TX EQ Training Editor user interface appears as shown in Figure 5:

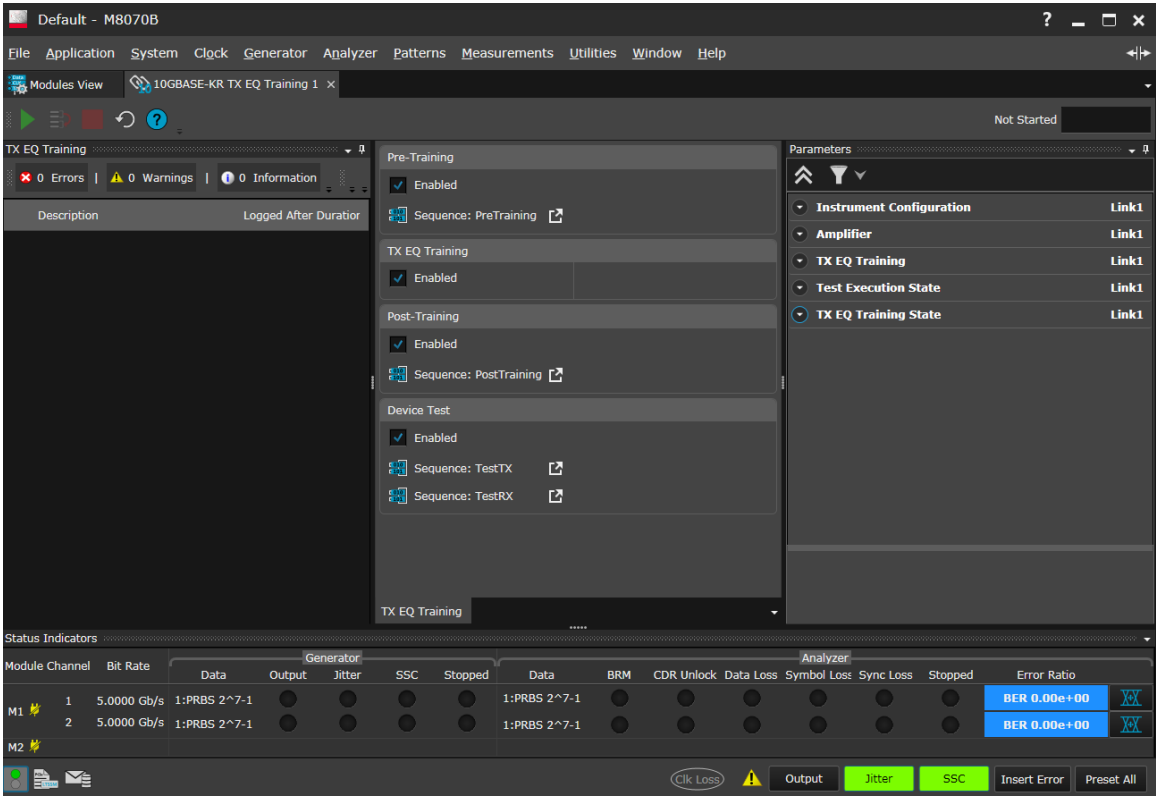


Figure 5 10GBASE-KR TX EQ Training Plug-in User Interface

4 For the 25GBASE-KR and 100GBASE-KR4 TX EQ Training plug-in, select **IEEE** > **25GBASE-KR and 100GBASE-KR4 TX EQ Training** plug-in.

The 25GBASE-KR and 100GBASE-KR4 TX EQ Training Editor user interface appears as shown in Figure 6:

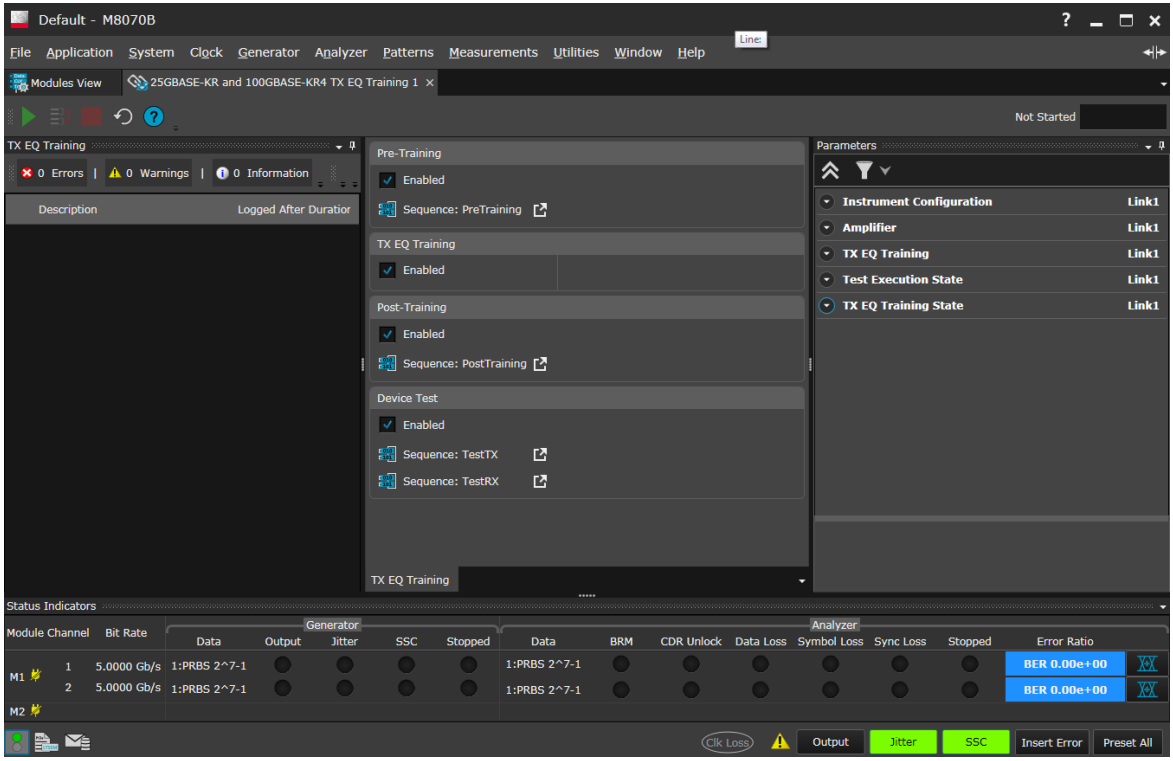


Figure 6 25GBASE-KR and 100GBASE-KR4 TX EQ Training Plug-in User Interface

Related Documents

For more details about the M8070B system software, refer to the M8070B documentation.

To locate the M8070B documents, click **Start** > **All Programs** > **Keysight M8070B** > **Keysight M8070B Documentation**. Alternatively, you may also visit www.keysight.com/find/M8070B to find the latest versions of the M8070B and related manuals.

Contacting Keysight Technologies

For more information on products, applications or services associated with Keysight Technologies, contact your local Keysight office. The complete list is available at: www.keysight.com/find/contactus.

2 MultiGBASE TX EQ Training Frames

Overview

The MultiGBASE TX EQ Training plug-in defines a small part of the complete process of link training, called “TX Equalization Training”.

10GBASE-KR, 25GBASE-KR and 100GBASE-KR4 TX EQ Training start-up protocols are defined in IEEE Std 802.3 sub-clauses: 72.6.10.1, 111.7.10.1 and 93.7.12 respectively.

TX Equalization Training has four states:

- Pre-Training
- TX EQ Training
- Post-Training
- Device Test

TX Equalization Training States:

1 Pre-Training: In this state, the MultiGBASE TX EQ Training plug-ins prepare the DUT to accept their start-up protocol. The Pre-Training state brings the DUT from initialization mode to data-exchange (or communication) mode; using either the Pre-Training sequence blocks or external means.

NOTE **Default sequence is empty for Pre-Training state.**

2 TX EQ Training: In this state, the MultiGBASE TX EQ Training plug-ins negotiate the optimal equalization setting between the instrument’s generator and the DUT’s receiver. The MultiGBASE TX EQ Training plug-ins support the EQ Training through the continuous exchange of fixed-length training frames. For more information about training frames, refer to Training Frame Structure for 10GBASE-KR on page 28 and Training Frame Structure for 25GBASE-KR and 100GBASE-KR4 on page 33.

3 Post-Training: In this state, the MultiGBASE TX EQ Training plug-ins run an optional sequence intended to prepare the DUT for receiver tolerance testing. For example, one such step which may be run by any of the TX EQ Training plug-in, is to send a custom preparation frame, bringing the DUT into tolerance testing mode. The DUT can be brought into loopback mode either by using a special sequence within the Post-Training block or by proprietary means.

NOTE Default sequence is empty for Post-Training state.

4 Device-Test: In this state, the MultiGBASE TX EQ Training plug-ins run custom steps to perform device testing in loop-back mode. For example in this state, one such step run by the plug-in, is to run a jitter tolerance test.

NOTE Default sequence for Device-Test is:

- TX: PRBS 7-1, infinitely looping
- RX: PRBS 7-1, sync and loop

Training Frame Structure for 10GBASE-KR

The 10GBASE-KR TX EQ Training plug-in sends training frames continuously during the TX EQ Training. The length is fixed with (512+16+16+4) 548 octets as shown in Figure 7.

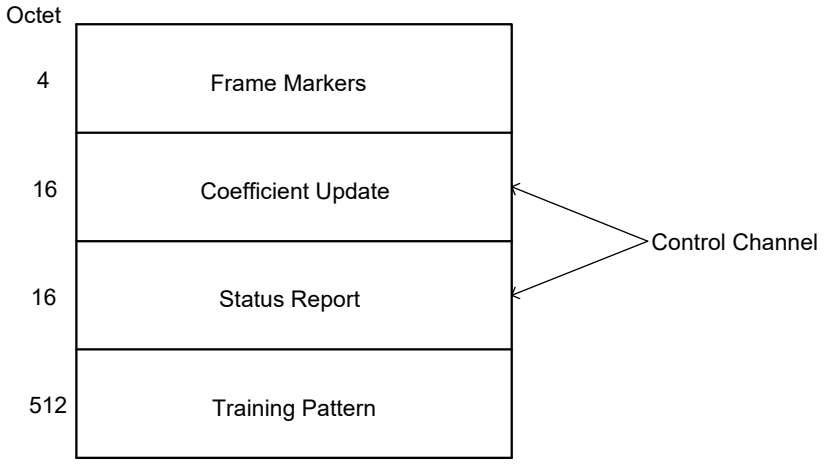


Figure 7 Training Frame structure

The training frame structure consists of the following frames:

Frame markers

Frame markers indicate the start of a training frame. They have a unique pattern, which does not appear in the Control channel or the Training Pattern. Training frames are delimited by the 32-bit pattern, 0xFFFF0000 (hexadecimals, ones transmitted first) when expressed as 10.3125 GBd NRZ data.

Control channel

Communicating with a DUT over the control channel is done using Differential Manchester Encoding (DME) at a data rate equal to one eighth of the 10GBASE-KR symbol rate.

Each DME symbol contains two DME transition positions and each transition position is four 10GBASE-KR UI. So, one control channel bit is transmitted in every eighth 10GBASE-KR UI. Thus, the length of one DME

coded data cell/bit is eight 10GBASE-KR UI and the total length of the control channel is {[(16+16) x 8] x 10GBASE-KR}UI, that is, (256 x 10GBASE-KR)UI.

The control channel consists of two fields – Coefficient Update and Status Report. For more details on the Control channel, refer to IEEE Std 802.3, sub-clause 72.6.10.2 – Training Frame Structure and sub-clause 72.6.10.2.2 Control Channel Encoding.

Coefficient update

The Coefficient Update field carries coefficient change requests from the DUT to the instrument's transmit equalizer and vice versa. This field provides request bits for setting the receiver to preset or initialize state or to update its coefficients.

Table 10 Coefficient Update field

Cell(s)	Name	Description
15 & 14	Reserved (R)	Always transmitted as 0 and ignored on reception.
13	Preset (P)	1 = Preset coefficients 0 = Normal operation
12	Initialize (I)	1 = Initialize coefficients 0 = Normal operation
11 to 6	Reserved (R)	Always transmitted as 0 and ignored on reception.
5 & 4	Coefficient (+1) update	5 4 1 1 = reserved 0 1 = increment 1 0 = decrement 0 0 = hold
3 & 2	Coefficient (0) update	3 2 1 1 = reserved 0 1 = increment 1 0 = decrement 0 0 = hold
1 & 0	Coefficient (–1) update	1 0 1 1 = reserved 0 1 = increment 1 0 = decrement 0 0 = hold

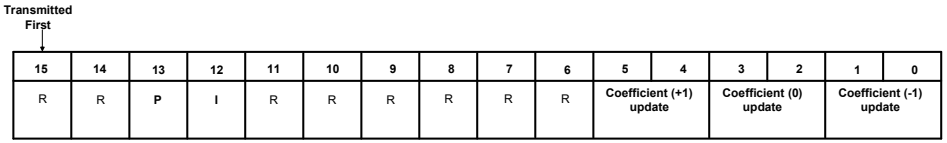


Figure 8 Coefficient Update Field

Description of the fields:

i Preset: A preset control request is sent to set the coefficients to a state where equalization is turned off. In this state, the pre-cursor (k=-1) and post-cursor (k=+1) coefficients are turned off, and the main cursor (k=0) is set to its maximum value.
For more details about the preset setting, refer to IEEE Std 802.3, sub-clause 72.6.10.2.3.1 – Preset

ii Initialize: An initialize request is sent to set the coefficients to configure the transmit equalizer to INITIALIZE. In this state, the taps should be set in such a manner that the values Rpre and Rpst of the transmitter equalizer are configured to 1.29 ± 10% and 2.57 ± 10% respectively, as described in section Pre- and Post-cursor equalization ratios on page 37.
For more details about the initialize setting, refer to IEEE Std 802.3, sub clause 72.6.10.2.3.2 – Initialize

iii Coefficient (k) update: The Coefficient (k) update is a 2-bit field which has three defined request states:

- Increment – request for the corresponding coefficient to be increased
- Decrement – request for the corresponding coefficient to be decreased
- Hold –default state for a given tap, which corresponds to no change in the coefficients

For more details about Coefficient update, refer to IEEE Std 802.3, sub clause 72.6.10.2.3.3 – Coefficient (k) update.

Status Report

The Status Report field sends the information about the signal state from the DUT to the 10GBASE-KR TX EQ Training plug-in.

Table 11 Status Report field

Cell(s)	Name	Description
15	Receiver ready (Rx_R)	1 = The DUT has determined that training is completed and that it is prepared to receive data. 0 = The DUT is requesting that training continues.
14 to 6	Reserved (R)	Transmitted as 0 and ignored on reception.
5 & 4	Coefficient (+1) status	5 4 1 1 = maximum 1 0 = minimum 0 1 = updated 0 0 = not updated
3 & 2	Coefficient (0) status	3 2 1 1 = maximum 1 0 = minimum 0 1 = updated 0 0 = not updated
1 & 0	Coefficient (–1) status	1 0 1 1 = Reserved 0 1 = increment 1 0 = decrement 0 0 = hold

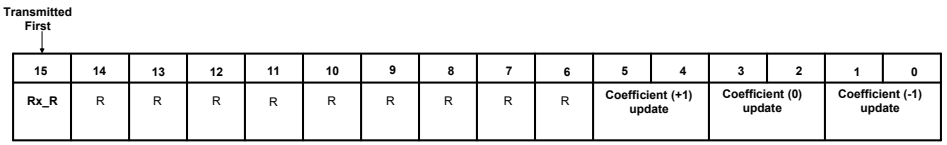


Figure 9 Status Report Field

i Receiver ready: This field indicates the current state of the DUT to the 10GBASE-KR TX EQ Training plug-in. When this field is asserted, it indicates that the training is completed and the DUT is ready to receive data. As long as this entry is set to 0 TX EQ training has not concluded.

ii Coefficient (k) status: The Coefficient (k) status is a 2-bit field that describes the status updates to the coefficients. The status is defined by either **not updated**, **updated**, **maximum** or **minimum**. Encoding of the coefficient update is shown in Table 11 on page 31 and the valid range for k is -1 to +1, where k = 0 represents the main tap.

Training pattern

The Training pattern has a length of 512 octets, which consists of 4094 bits from the output of a pseudo-random bit sequence of order 11 (PRBS11-1) generator shown in Figure 10, followed by two zeros. For more information about the training pattern, refer to IEEE Std 802.3, sub clause 72.6.10.2.6 – Training pattern.

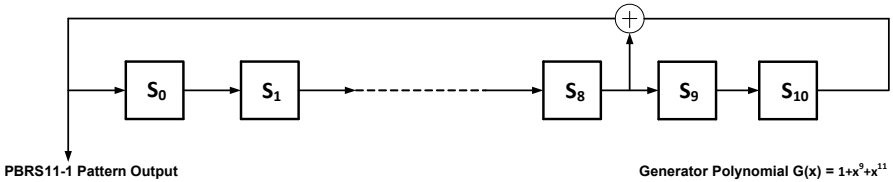


Figure 10 PRBS11-1 pattern generator

Training Frame Structure for 25GBASE-KR and 100GBASE-KR4

25GBASE-KR uses a single lane while 100GBASE-KR4 has a four lane structure (Lane 0, Lane 1, Lane 2 and Lane 3).

For both 25GBASE-KR and 100GBASE-KR4 each lane uses a training frame structure similar to 10GBASE-KR (Training Frame Structure for 10GBASE-KR on page 28). The differences are detailed in this following sub-section:

Symbol Rate

25GBASE-KR and 100GBASE-KR4 operate at a symbol rate of 25.78125 GBd.

Control channel

NOTE **25GBASE-KR uses the same control function as Lane 0 of 100GBASE-KR4.**

Communicating with a DUT over the control channel is also done using Differential Manchester Encoding (DME) but at a data rate equal to one eighth of the 25GBASE-KR and 100GBASE-KR4 symbol rate.

Similarly, each DME symbol contains two DME transition positions but each transition position is four 100GBASE-KR4 UI. So, one control channel bit is transmitted in every eighth 100GBASE-KR4 UI. Thus, the length of one DME coded data cell/bit is eight 100GBASE-KR4 UI and the total length of the control channel is {[(16+16) x 8] x 100GBASE-KR4}UI, that is, (256 x 100GBASE-KR4)UI.

The control channel also consists of two fields– Coefficient Update and Status Report as described for 10GBASE-KR.

Coefficient update

Coefficient update process is also similar to 10GBASE-KR but the period from receiving a new request to responding to that request shall be less than 2 ms, except during the first 50 ms following the beginning of the start-up protocol.

Description of the fields:

i Preset: Same as for 10GBASE-KR. For more details about the preset setting, refer to IEEE Std 802.3, sub-clause 93.8.1.5.2

ii Initialize: For more details about the initialize setting, refer to IEEE Std 802.3, sub clause 93.8.1.5.3

iii Coefficient (k) update: Same as for 10GBASE-KR. For more details about Coefficient update, refer to IEEE Std 802.3, sub clause 93.8.1.5.4

Status Report

Same as described for 10GBASE-KR Status Report on page 31.

Training pattern

For 25GBASE-KR and 100GBASE-KR4 the training pattern has been enhanced to minimize correlation between lanes when using more than one lane. Following are the specifications:

- The training pattern for each lane consists of 4094 bits from the output of a pseudo-random bit sequence (PRBS) generator followed by two zeros.
- 100GBASE-KR4 is specified for operation on four lanes. Each of these lanes uses a unique training pattern PRBS polynomial 'n' where ‘n’ where n goes from 0 to 3 as shown in Table 12 on page 35.
- A seed value of 0x000 is invalid. An example implementation of the PRBS generator for n = 0 with default settings is given in Figure 11. The first 32 bits of the training pattern for each polynomial is also provided in Table 12.

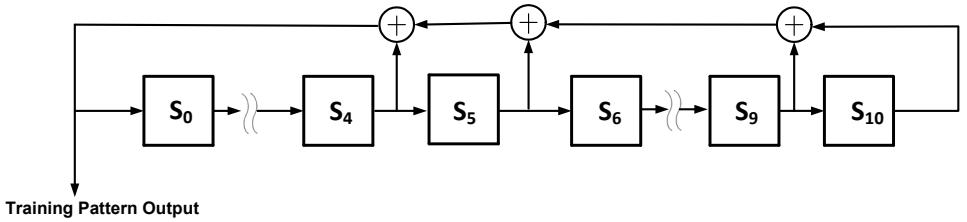


Figure 11 PRBS generator for polynomial n = 0

Table 12 PRBS parameters for each physical lane

n	Polynomial_n, G(x)
0	$1 + x^5 + x^6 + x^{10} + x^{11}$
1	$1 + x^5 + x^6 + x^9 + x^{11}$
2	$1 + x^4 + x^6 + x^8 + x^{11}$
3	$1 + x^4 + x^6 + x^7 + x^{11}$

Transmitter Equalization Training Procedure

Figure 12 shows a 11111111...00000000... waveform sent from a 10GBASE-KR TX EQ (or 25GBASE-KR and 100GBASE-KR4) Training plug-in. It shows the maximum and minimum values of amplitude that a training frame reaches during TX EQ Training, for a specific coefficient setting. According to the requests sent from the DUT, the plug-in changes its pre-cursor, post-cursor and main tap values within user definable or hardware dependent ranges. When the maximum or minimum voltage level is reached, the plug-in sends a training frame containing a status report to the DUT, indicating that it has reached its maximum or minimum voltage or coefficient limit. Further changes of coefficient value in the same direction will therefore not be applied.

For more details about Transmitter Output Waveform, refer to IEEE Std 802.3, sub clause 72.7.1.11 – Transmitter output waveform requirements.

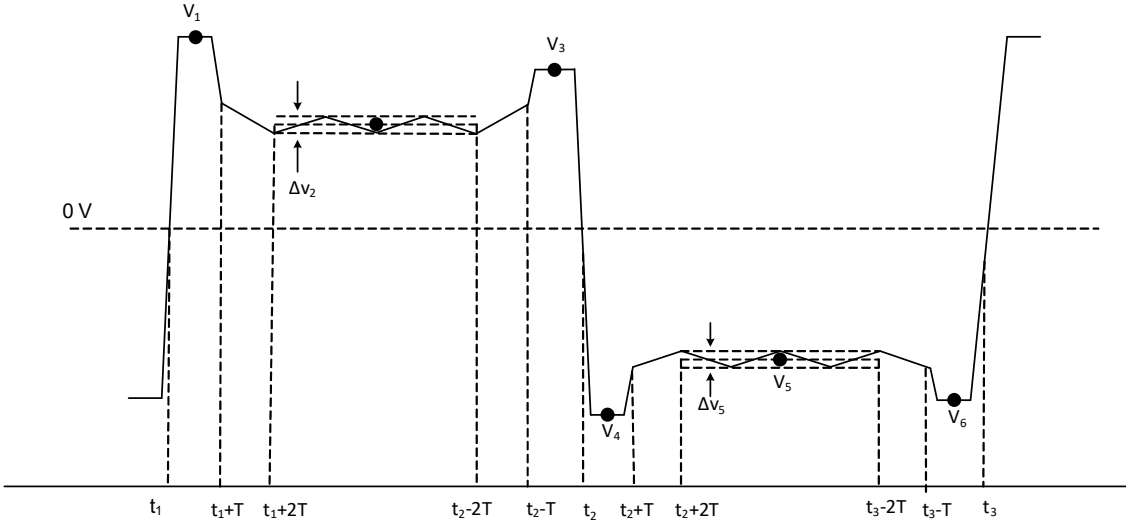


Figure 12 Transmitter Output waveform

In this figure:

T = symbol period

t_1 = zero-crossing point of the first rising edge of the AC-coupled signal

t_2 = zero-crossing point of the falling edge of the AC-coupled signal

t_3 = zero-crossing point of the second rising edge of the AC-coupled signal

v_1 = maximum voltage measured in the interval t_1 to $t_1 + T$

v_2 = positive steady-state voltage measured as the average voltage in the interval $t_1 + 2T$ to $t_2 - 2T$

v_3 = maximum voltage measured in the interval $t_2 - T$ to t_2

v_4 = minimum voltage measured in the interval t_2 to $t_2 + T$

v_5 = negative steady-state voltage measured as the average voltage in the interval $t_2 + 2T$ to $t_3 - 2T$

v_6 = minimum voltage measured in the interval $t_3 - T$ to t_3

Δv_2 = positive voltage ripple measured as the peak-to-peak value of the difference between the voltage in the range $t_1 + 2T$ to $t_2 - 2T$ and v_2

Δv_5 = negative voltage ripple measured as the peak-to-peak value of the difference between the voltage in the range $t_2 + 2T$ to $t_3 - 2T$ and v_5

Pre- and Post-cursor equalization ratios

From the value of the voltages, shown in Figure 12, you can calculate the pre- and post-cursor equalization ratios R_{pre} and R_{pst} respectively, using the following equations:

- $R_{pre} = v_3/v_2$
- $R_{pst} = v_1/v_2$

NOTE

For Preset: $R_{pre} = 1$ and $R_{pst} = 1$

For Initialize: $R_{pre} = 1.29 \pm 10\%$ and $R_{pst} = 2.57 \pm 10\%$

3 Using MultiGBASE TX EQ Training User Interface

Two different training plug-ins “10GBASE-KR TX EQ Training” and “25GBASE-KR and 100GBASE-KR4 TX EQ Training” have been developed for testing receiver tolerance of devices against impairments such as jitter and interference.

The user interface for both plug-ins has the same appearance and is described in this chapter.

MultiGBASE TX EQ Training User Interface

Figure 13 and Figure 14 show the user interface for the “10GBASE-KR TX EQ Training” and the “25GBASE-KR and 100GBASE-KR4 TX EQ Training” plug-in respectively:

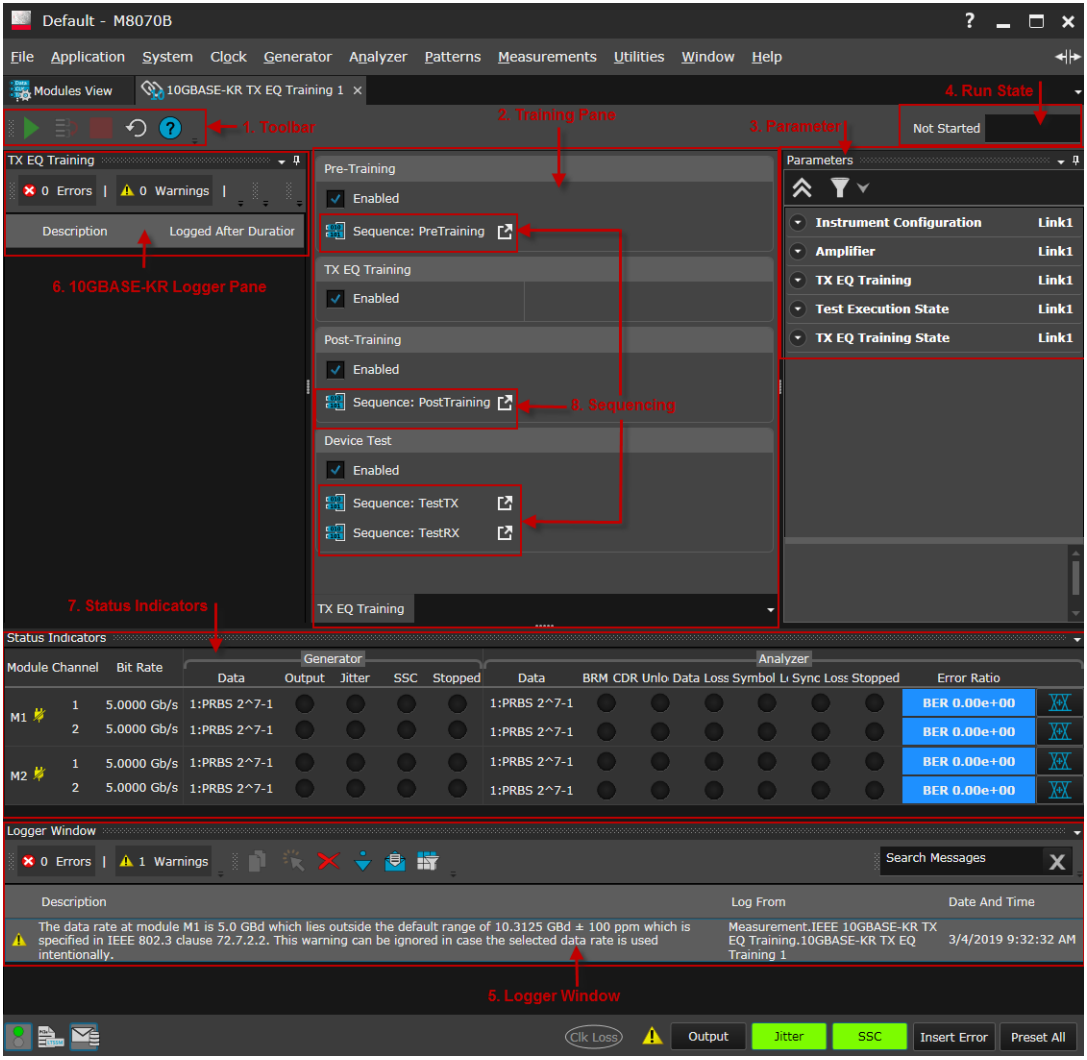


Figure 13 10GBASE-KR TX EQ Training Plug-in User Interface

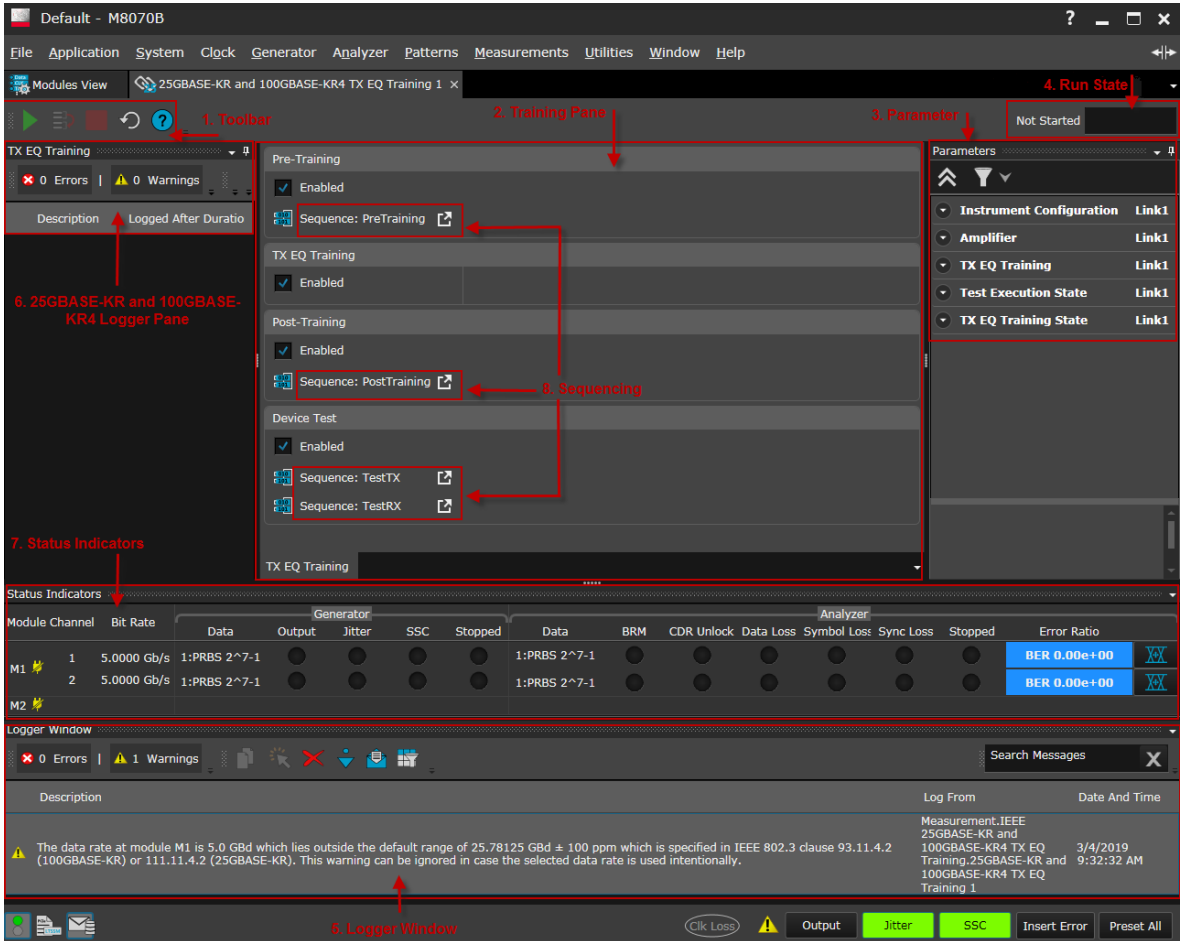


Figure 14 25GBASE-KR and 100GBASE-KR4 TX EQ Training Plug-in User Interface

Following are the elements of the user interface, which are common to both plug-ins:

1 Toolbar
2 Training Pane
3 Parameters
4 Run State
5 The Global Logger Window

6 Logger Pane (10GBASE-KR) or Logger Pane (25GBASE-KR and 100GBASE-KR4)
7 Status Indicators
8 Sequencing (MultiGBASE TX EQ Training - Sequence Editor)

Toolbar

The toolbar provides the following shortcuts:

Figure 15 Toolbar

Table 13 Toolbar commands description

	Name	Description
	Start	Sets the TX EQ Training plug-in into run mode. This command runs the Pre-Training, TX EQ Training, Post Training and Device Test sequentially, if the respective block is enabled.
	Break	Exits an infinite loop that is configured with a “manual” break condition. Sequence execution continues with the next block. Pressing this button will not influence execution of the TX EQ Training block.
	Stop	Stops the plug-in execution.
	Reset	Resets the plug-in to its default values.
	Open Manual	Opens the User Guide.

WARNING You may use the Break button from the TX EQ Training plug-in or from the M8070B Sequence Editor. Functionality is the same for both options.

Suppose, you are using multiple instances of the plug-in, such that the “10GBASE-KR TX EQ Training N” (or “25GBASE-KR and 100GBASE-KR4Training N”) corresponds to the 'Nth' TX EQ Training plug-in instance; the Break functionality suspends all breakable loop sequences in the system that may be running simultaneously, thereby affecting all other plug-in instances.

Training Pane

TX Equalization Training is performed by four blocks:

- Pre-Training
- TX EQ Training
- Post-Training
- Device Test

You can edit the sequence parameters in all blocks except in the TX EQ Training block. Refer to section MultiGBASE TX EQ Training - Sequence Editor on page 48 to know about how to modify the parameters for sequences.

Performing TX Equalization Training

- Click the Run  button from the toolbar to perform TX Equalization Training. For details, refer to Toolbar on page 42.
- In all different states of the TX Equalization Training, there is an option **Enabled** available as shown in Figure 16. If you want to skip the respective training state, clear the check box.

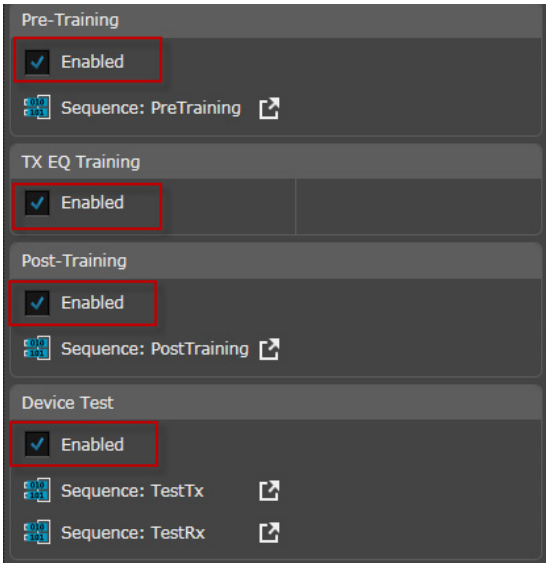


Figure 16 Different states of TX Equalization Training

For details on different states of the TX Equalization Training, refer to Training Pane on page 74 (chapter 4).

Parameters

The Parameters pane allows you to change the parameters related to the instrument configuration, amplifier settings and TX EQ training and TX Equalization Training parameters.

NOTE **During test execution, parameter changes are not permitted. Doing so can interfere with plug-in execution and lead to unexpected behavior.**

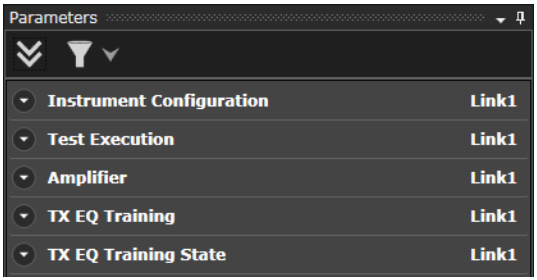


Figure 17 Parameters Pane

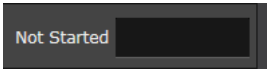
For details on plug-in specific parameters, refer to Parameters on page 78 for 10GBASE-KR and Parameters on page 91 for 25GBASE-KR and 100GBASE-KR4 (chapter 4).

Run State

The Run State shows the current state of the plug-in.

Not Started

Indicates that the plug-in is in idle mode.



Running

Indicates that the plug-in is currently executing its test sequence.

Stopped

Indicates that the plug-in execution has been stopped.

Finished

Indicates that the plug-in execution is finished.

The Global Logger Window

The global M8070B Logger Window displays the description of Error and Warning messages along with the applications from where they were generated and their time stamps.

For error and status messages dedicated to the TX EQ Training plug-in a separate logger window exists inside the plug-in view.

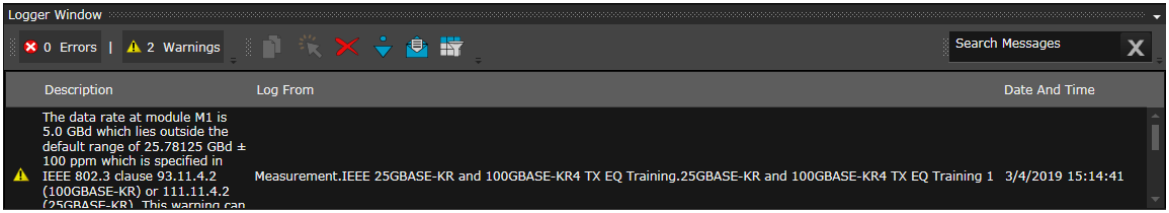


Figure 18 The Global Logger Window (when 10GBASE-KR TX EQ Training plug-in is running)

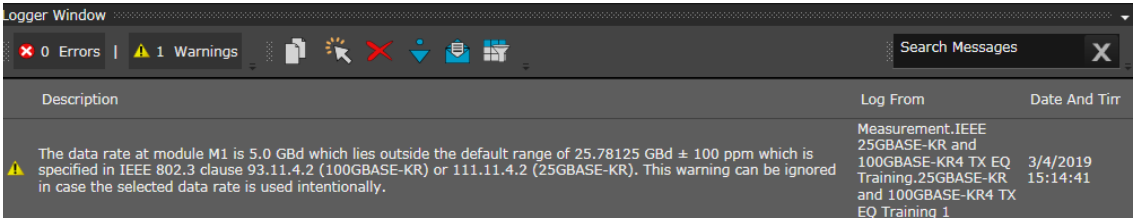


Figure 19 The Global Logger Window (when 25GBASE-KR and 100GBASE-KR4 TX EQ Training plug-in is running)

Logger Pane (10GBASE-KR)

Refer to Logger Pane (10GBASE-KR) on page 85 (chapter 4).

Logger Pane (25GBASE-KR and 100GBASE-KR4)

Refer to Logger Pane (25GBASE-KR and 100GBASE-KR4) on page 95 (chapter 4).

Status Indicators

The Status Indicators window displays the current status of the generator and analyzer ports for each channel of the connected modules. For more information on Status Indicators, refer to *Keysight M8070B User Guide*.

MultiGBASE TX EQ Training - Sequence Editor

Overview

MultiGBASE TX EQ Training - sequence editor allows you to create, edit and maintain sequences which are used in the following states of the TX Equalization Training:

- PreTraining
- PostTraining
- TestTx
- TestRx

Accessing the Sequence Editor

To access the *10GBASE-KR TX EQ Training - Sequence* window or *25GBASE-KR and 100GBASE-KR4 TX EQ Training - Sequence* window

- Click any of the sequences in the user interface window of the plug-in as shown (red boxes) in Figure 20:

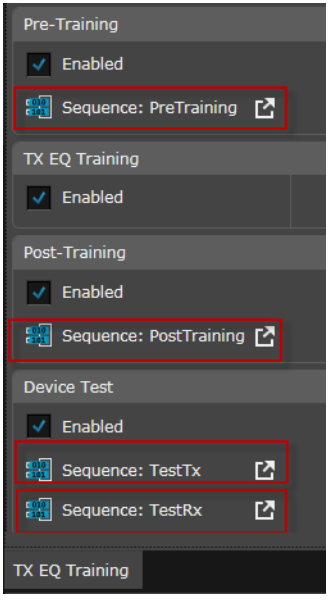


Figure 20 Sequence Editor Window

- *MultiGBASE TX EQ Training - Sequence* editor window opens in a new tab as shown in Figure 21.

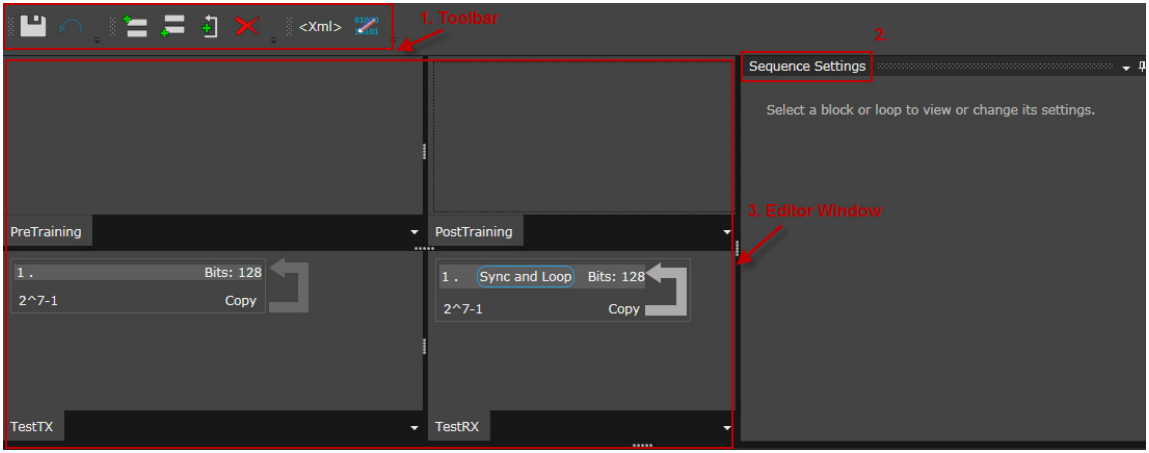


Figure 21 MultiGBASE TX-EQ Training-Sequence Editor User Interface

The sequence editor user interface includes the following elements showing the sequence played back during execution of the respective block:

1 Toolbar

2 Sequence Settings

3 Editor Window: It is divided into four panes:

- PreTraining
- PostTraining
- TestTX
- TestRx

Toolbar of MultiGBASE TX EQ Training - Sequence Editor

The toolbar of the MultiGBASE TX EQ Training - Sequence Editor provides the following sequence editing functions:

Table 14 MultiGBASE-KR TX EQ Training - Sequence Editor Toolbar

Icon	Name	Description
	Save	Saves the changes made in the sequence editor. An orange icon  indicates that modifications are not yet used during plug-in execution. Once you save the changes, it returns to its default color (grey).
	Discard changes and restore sequences	Discards the changes made in the sequence editor and restore its default values. For more details, refer to Discard changes and restore sequence default.
	Add Block Before Selected Block	Adds a block before the selected sequence block.
	Add Block After Selected Block	Adds a block after the selected sequence block.
	Add New Loop	Creates a loop around a sequence block. For details, refer to Add New Loop.
	Delete Selected block or Loop	Deletes the selected block from the sequence. Deleting a block will also remove all loops that are associated with this block.
	XML	Toggles between a graphical and a textual XML sequence representation. The changes made in one representation will reflect into the other representation.
	Show Memory Pattern	In case the selected block uses a memory pattern clicking this icon opens the pattern in a new Pattern Editor View.

Using the MultiGBASE TX EQ Training - Sequence Editor Toolbar

To understand how to use the toolbar, click any one of the four panes. For example, you may click **PreTraining**. You can perform the following operations:

- Add block(s)
  - i Click the  **Add Block Before Selected Block** icon to add a block before the selected sequence block.
  - ii Click the  **Add Block After Selected Block** icon to add a block after the selected sequence block.
- Delete block
  - i Select the block you want to delete.
  - ii Click the  icon.
- Delete loop
  - i Select the loop indicator you want to delete.
  - ii Click the  icon.
- Add New Loop
  - i Click the  **Add new Loop** icon. A **Create Loop** dialog is displayed just below the toolbar as shown in the following figure:

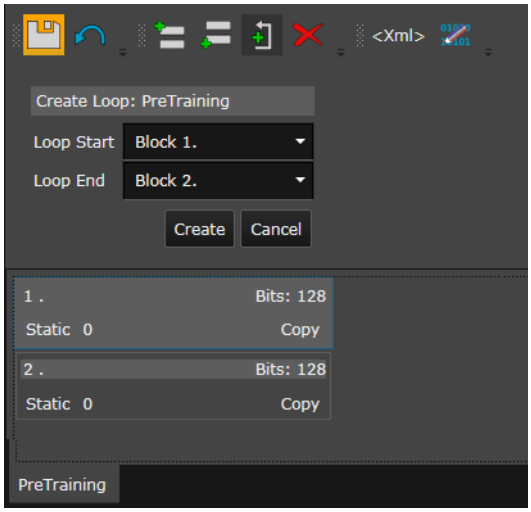


Figure 22 Creating a Loop

ii Specify the start and end blocks for the loop in the sequence. For example, select *Loop Start – Block 1* and *Loop End – Block 2*.

iii Click **Create** to create a loop around the specified blocks.

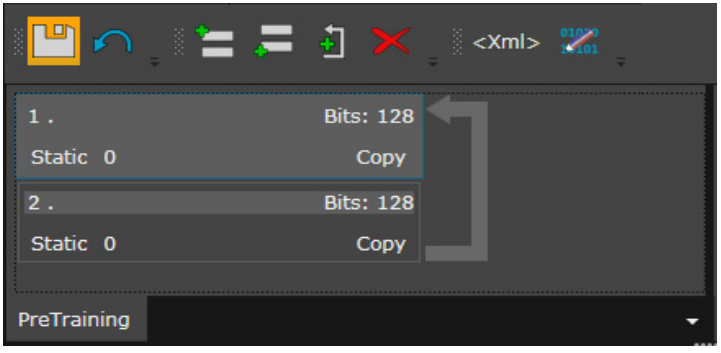


Figure 23 Created Loop

iv Click the loop indicator. A **Loop Settings** functional block appears in the **Sequence Settings** pane, where you can specify the loop count when enabling the Counted Loop option.

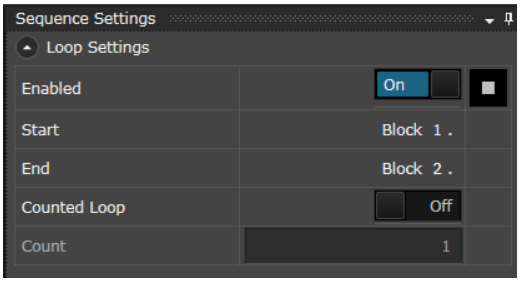


Figure 24 Loop Settings in Sequence Settings

- Discard changes and restore sequence default

  i Click  Discard changes and restore sequences icon.

  This opens a **Sequence Editor** dialog, which allows to discard the changes made in the sequence editor and restores its default values.

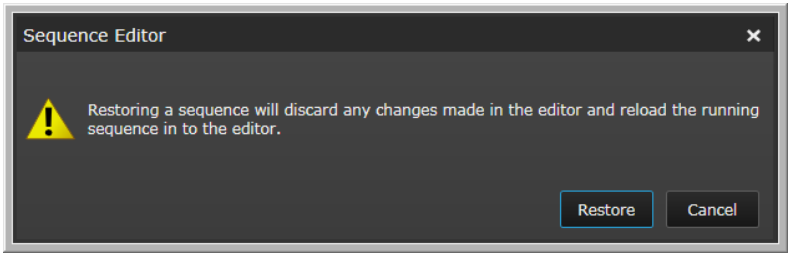


Figure 25 Sequence Editor Dialog

  ii Click **Restore** if you want to discard the changes and set it to the default value. Otherwise click **Cancel**.

NOTE

Click  to apply the changes. These changes will be applied once the plug-in is started.

Sequence Settings Sidebar

The Sequence Setting sidebar allows you to set the properties for the selected block and sequence. By using this sidebar, the following settings can be specified:

- Block Data. For details, refer to Block Data on page 55.
- Block Settings. For details, refer to Block Settings on page 56.
- Block Branches. For details, refer to Block Branches on page 57.
- Block Controls. For details, refer to Block Controls on page 58.

NOTE Some of the features (Serialize and Copy Plus Phase Adjust) of Sequence Settings might not be compatible with M8062A, but they are available as a selection in the GUI.

Set to Default check box

Most of the options in Block Data and Block Settings in the Sequence Setting sidebar contain a *Set to Default* check box, which gets highlighted when some modifications are done. Select this check-box to change the settings to their default values.

The following figure shows how the check box gets highlighted when the setting **Squelch** is set to **On**.

Figure 26 Using the Set to Default check box

If you click the **Set to Default** check box, the setting for **Squelch** changes to **Off,** which is its default value.

Figure 27 Example for setting parameter to default value

Block Data

Figure 28 Block Data

The Block Data section allows you to:

- Provide a block *Name*.
- Provide a block *Length*.
- Select *Block Type*. The available options are:
  - Clock: For block type as 'Clock', specify the parameter 'Divider' and 'Replicate'. The 'Replicate' feature shows how the serial patterns are split to multiple locations. It has the following options:
    - Serialize: This setting is not compatible with the plug-in use model.
    - Copy: In this option, each location gets a copy of the pattern.
    - Copy Plus Phase Adjust: This setting is not compatible with the plug-in use model.
  - Pulse: For block type as 'Pulse', specify the 'Width', 'Replicate' and 'Offset' features.
  - Prbs: For block type as 'PRBS', you need to specify the 'Polynomial', 'Replicate', 'Invert' and 'Seed (Hex)' features.
  - Static: For block type as 'Static', specify the 'Signal Value', 'Replicate and the 'Squelch' features.
  - Memory Pattern: If you select Block Type as *Memory Pattern*, a **Select Pattern** dialog appears, which allows you to select a memory pattern.

Compare

The Compare feature allows you to compare the block data for a particular sequence. It provides the flexibility to modify the sequence without deleting the blocks.

- When to use/not use this feature: Suppose you have created a sequence with multiple blocks. Now, if you want to exclude particular block(s) from that sequence while comparing, instead of deleting the whole sequence, just set the compare functionality to “OFF”. It disables these particular block(s) from the sequence for comparison.

NOTE The Compare feature is used with the TestRx sequence block only. This block is intended for measuring a BER after successfully performing TX EQ Training.

Block Settings

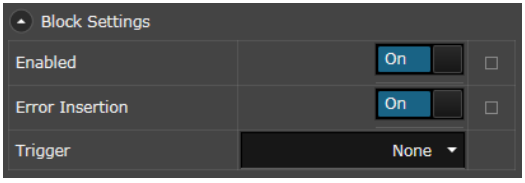


Figure 29 Block Settings

The Block Settings section has the following features:

- Enabled: Click the corresponding **ON/OFF** toggle button to enable the block in the sequence.
- CDR: Click the corresponding **ON/OFF** toggle button to enable/disable the CDR during execution of this block. This parameter only has an effect if the CDR control has been set to sequence driven.
- Sync and Loop: Click the corresponding **ON/OFF** toggle button to turn on Sync and Loop for the corresponding Data In block.

NOTE Pattern sync is automatically performed when activating the Sync and Loop feature upon every iteration of the corresponding block.

- Trigger: Use the drop down option to specify whether you want to apply trigger on either ‘Pulse’ or ‘Pulse or PRBS Match’.
  - If you want to apply trigger on ‘Pulse’, you have to set the values of Offset and Width.
  - If you want to apply trigger on ‘Pulse or PRBS Match’, you need to set the values of ‘Polynomial’, ‘Invert’ and ‘Width’.

Block Branches

Used to vary sequence execution depending on external conditions. You can add up to two branches in a sequence.

To add a branch, click the *Add Branch* button.

i Go to Block: Specify the block name to jump to.

ii Click Add Branch if you want to add another branch. Up to two branches can be added within the same block.

iii Click 

to delete the branch.

iv Once branching is enabled in a block, the Block Branches 

icon appears on the sequence block as shown in the following figure:

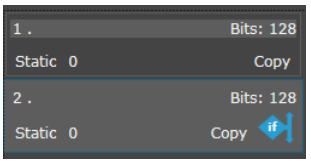


Figure 30 Example of enabling Block Branches

The Block Branches section appears as shown in the following figure:

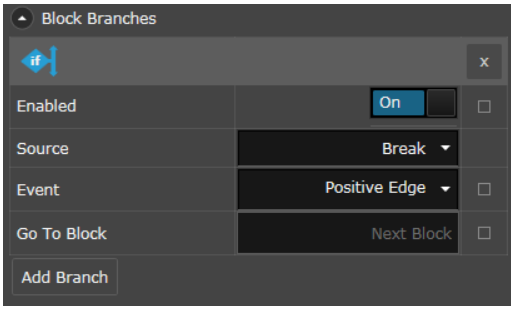


Figure 31 Block Branches

The Block Branches section provides the following settings:

- Enabled: Use the **ON/OFF** toggle button to enable the branching option.
- Source: Use the drop-down list to specify the source for the branch, which has the following options:
  1 Break
  2 Ctrl In A
  3 Error
  4 Ctrl In B
- Event: Specify the event for the branch. It has the following options:
  1 Positive Edge
  2 Negative Edge
  3 High
  4 Low

Block Controls

The Block Controls section allows to output a specific signal at the target connector (CTRL OUT A, SYS OUT A, SYS OUT B) during playback of the selected sequence block. Possible signal values are Low, High and Pulse. Click **Add Control** to add more block controls. You can add up to four block controls.

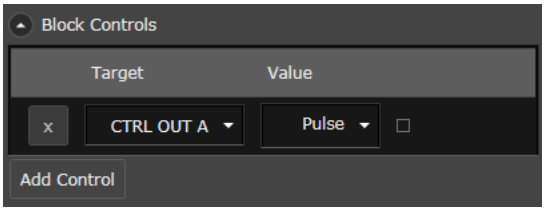


Figure 32 Block Controls

NOTE

The functionality of the MultiGBASE TX EQ Training - Sequence editor is similar to the M8070B Sequence Editor. To know how to access the M8070B Sequence Editor, refer to section M8070B Sequence Editor on page 60. For more information, refer to the M8070B User Guide.

There are also some differences between the M8070B Sequence Editor and the plug-in specific Sequence Editor, such as:

- The plug-in specific sequence editor does not allow to define any locations for the sequences as these are defined by the selected generator and analyzer values of the TX EQ Training plug-in.
- Through the plug-in specific Sequence Editor, it is also not possible to access the functional blocks of Instrument Configuration and Sequence Configuration.

Editor Window

The Editor window provides an interactive user interface for creating, editing and exporting the sequences in all four panes. For details, see section Using the MultiGBASE TX EQ Training - Sequence Editor Toolbar on page 51.

M8070B Sequence Editor

The M8070B Sequence Editor allows to create and maintain sequences. In addition to this, it allows to edit patterns.

To access the M8070B **Sequence Editor**:

Go to the **Menu Bar** > **Patterns** and select **Sequence Editor**. The **Sequence Editor** window appears as shown in the following figure:

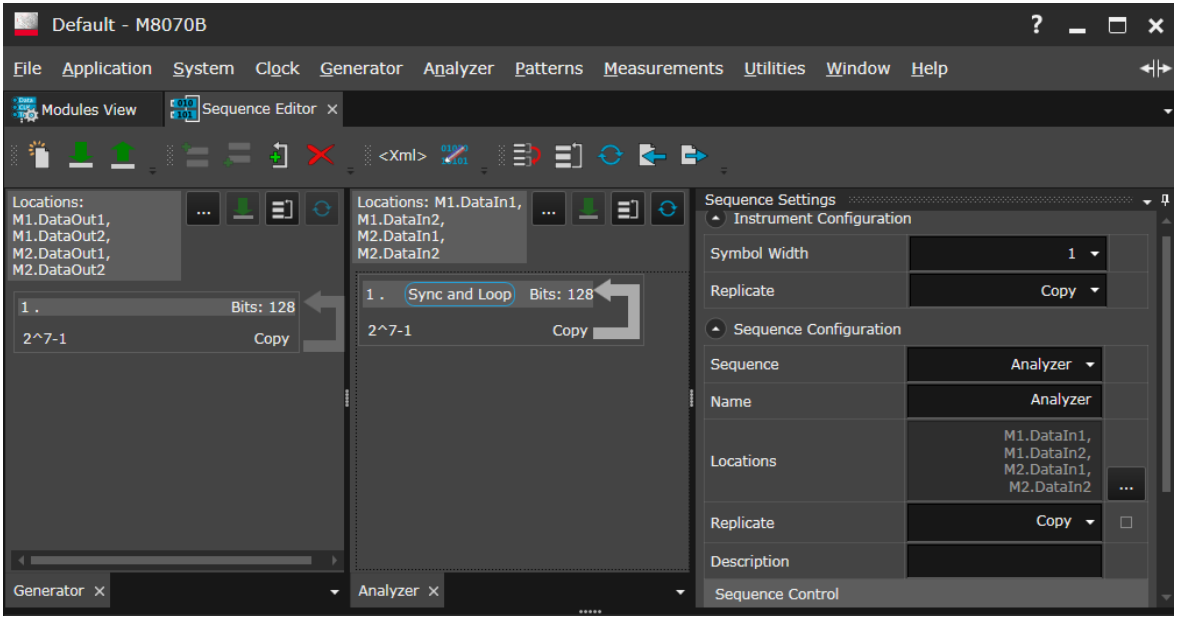


Figure 33 Sequence Editor Window

NOTE Modifying parameters of M8070B Sequence Editor during plug-in test execution is not recommended, else the plug-in may lead to malfunctions.

NOTE Restarting the automatically generated sequence or skipping a sequence using the M8070B Sequence Editor or using the respective SCPI commands must be done with care while the TX Equalization Training is running. It might lead to inconsistent/faulty behavior, in the plug-in.

4 Using Plug-in specific TX EQ Training User Interface

This chapter familiarizes you with the user interface of the “10GBASE-KR TX EQ Training” and “25GBASE-KR and 100GBASE-KR4 TX EQ Training” plug-ins. Additionally, it also describes some relevant parameters in the Module View, which are very important for TX EQ Training plug-ins.

Relevant Modules Parameters

The default view of the M8070B is the Modules View. However, if it is not available or it is closed, you can still view it.

To view the Modules View:

- Go to the **Menu Bar** > **System** and select **Modules View**.

Figure 34 shows an example of Modules View for 10GBASE-KR TX EQ Training plug-in when M8041A (M1) and M8051A (M2) modules are connected:

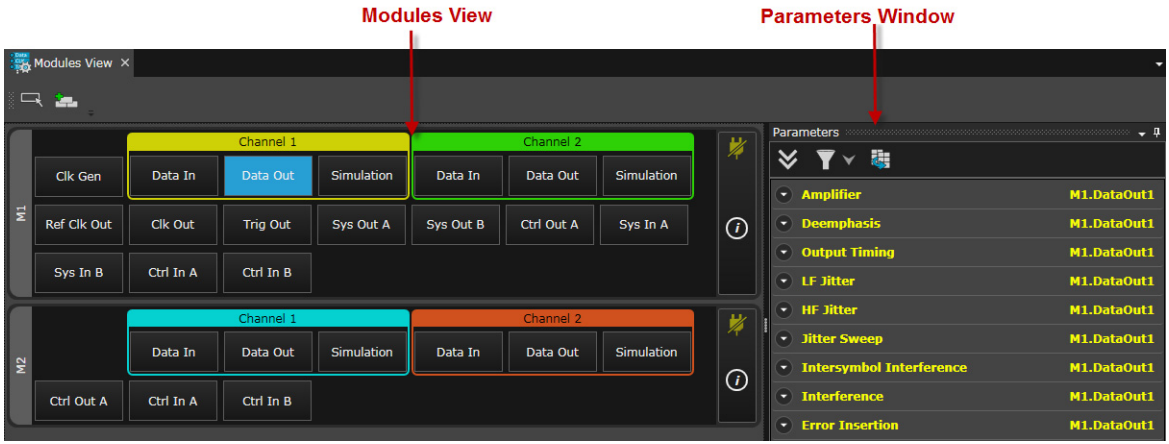


Figure 34 Module View for 10GBASE-KR TX EQ Training plug-in

Figure 35 shows an example of Modules View for 25GBASE-KR and 100GBASE-KR4 TX EQ Training plug-in when M8041A (M1) and M8062A (M2) modules are connected:

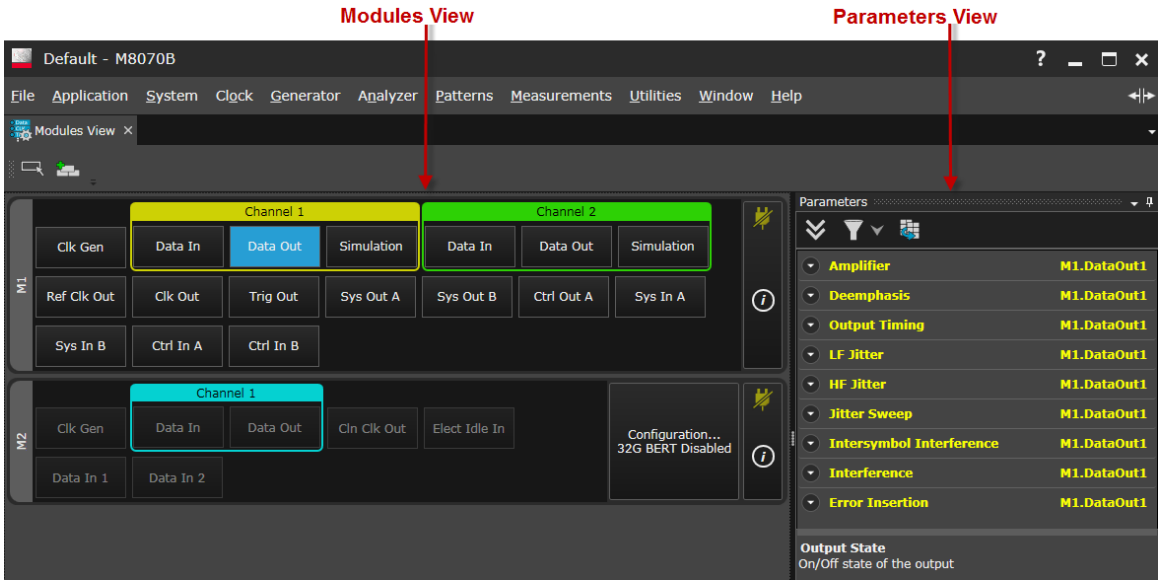


Figure 35 Module View for 25GBASE-KR and 100GBASE-KR4 TX EQ Training plug-in

In both figures (Figure 34 and Figure 35), the left pane shows the connected modules and the right pane shows the Parameters window. Each module has input and output ports, which can be configured through the Parameters window. For more details, refer to the *M8070B User Guide*.

This section consists of the parameters that are relevant for the plug-ins.

Plug-in Controlled Parameters

Plug-in controlled parameters are those parameters whose values are indirectly controlled by the plug-in while it is running.

NOTE **Modifying parameters during plug-in test execution is not recommended and must be handled with care, else the plug-in may lead to malfunctions.**

For (both) 10GBASE-KR & 25GBASE-KR and 100GBASE-KR4

The following parameters are modified by the plug-in, while it is running and must not be changed manually during runtime.

- Amplifier
  - Amplitude
  - Auto Range (Preset: Off)
  - Amplitude Range (Preset: 0 mV - 1.200 V)
- Deemphasis
  - Preset Enable
  - Preset Register Number
  - Unit
  - Pre-Cursor1
  - Post-Cursor1
- Sequencer

For (only) 25GBASE-KR and 100GBASE-KR4

The M8041A (16Gb/s Generator, Analyzer, Clock Module) has two channels and two Data Out ports. While the M8062A (32Gb/s BERT Front End) has only one channel and one Data Out port.

The following parameters of Data In port are modified by the plug-in. These values are changed only during TX EQ Training to give optimal results. After execution the original value are restored.

The highlighted parameter state indicates the default state for 25GBASE-KR and 100GBASE-KR4 operation.

- CDR
  - Auto Re-Lock: During link training, this parameter is by default set to **On**. If it loses synchronization with the received data stream, the CDR is re-acquired.
- Analyzer
  - Re-Sync: During execution of the TX EQ Training block, this parameter by default set to **Automatic**. Note that this changed after executing the previous state, the parameter is restored.

User Controlled Parameters

These parameters are not changed by the TX EQ training plug-in but are directly relevant to its usage.

For 10GBASE-KR

Following are the relevant parameters for TX Equalization Training:

1 Clk Gen
   - Synthesizer
     i Source: Allows you to select the trigger source of the instrument. In general, select the **Internal** option.
     ii Reference Frequency: For the Internal clock source, use **Internal 100 MHz**.
     iii Frequency: Use **10.3125 GHz**.
     Default signaling speed range for 10GBASE-KR is 10.3125 GBd ±100 ppm as defined in IEEE Std 802.3, sub-clause 72.7.1.3.
2 Data Out: This acts as the output port for the generator, which may be connected to the DUT.
   - Amplifier
     i Output State: Turn it **On**, otherwise the output will be off.
     ii Coupling: Select **DC** for TX Equalization Training.
     iii Termination Model: Select **Balanced** into differential 100 Ω.
     iv Transition Time: Controls the transition time of the output signal. Select **Smooth** for emulating a 10GBASE-KR TX.
3 Data In: This acts as the input port for the Analyzer. This port is connected to the data signal output by the DUT. Relevant parameters of the Data In port are:

- Comparator
  - i Coupling: Use **AC** for 10GBASE-KR.
  - ii Termination Configuration: Select **Balanced** mode, in which the input sets the termination voltage automatically to the same value as the given common mode voltage.
  - iii Common Mode Voltage: Select its value to **0 V** for TX Equalization Training.
- CDR
  - i CDR State: Set this to **On** always for 10GBASE-KR.

For 25GBASE-KR and 100GBASE-KR4

Following are the relevant parameters for TX Equalization Training:

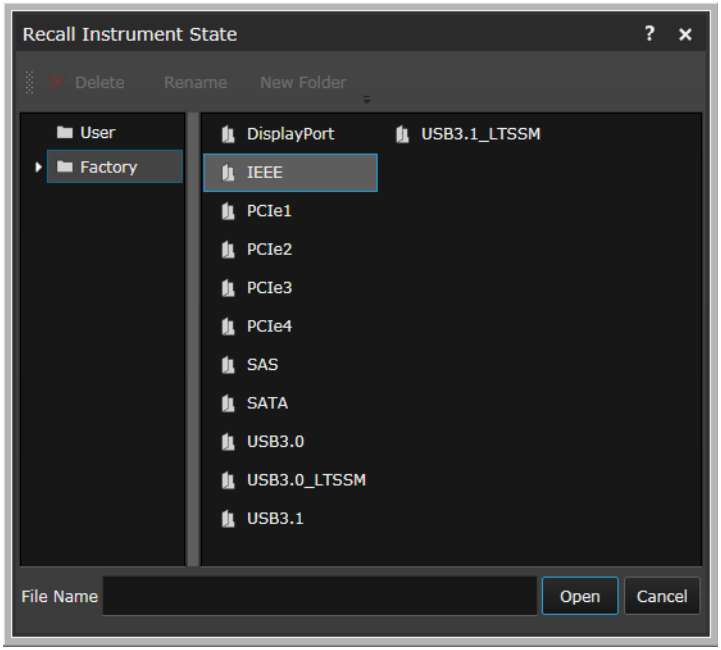
1 Clk Gen
   - Synthesizer
     - i Source: Allows you to select the trigger source of the instrument. In general, select the **Internal** option.
     - ii Reference Frequency: For the Internal clock source, use **Internal 100 MHz**.
     - iii Frequency: Use **25.78125 GHz**.
       Default signaling speed range for 25GBASE-KR and 100GBASE-KR4 is 25.78125 GBd.
2 Data Out: This acts as the output port for the generator, which may be connected to the DUT.
   - Amplifier
     - i Output State: Turn it **On**, otherwise the output will be off.
     - ii Coupling: Select **DC** for TX Equalization Training.
     - iii Termination Model: Select **Balanced** into differential 100 Ω.
3 Data In: This acts as the input port for the Analyzer. This port is connected to the data signal output by the DUT. Relevant parameters of the Data In port are:
   - Comparator
     - i Coupling: Use **AC** for 25GBASE-KR and 100GBASE-KR4.
     - ii Termination Configuration: Select **Balanced** mode, in which the input sets the termination voltage automatically to the same value as the given common mode voltage.
     - iii Common Mode Voltage: Select its value to **0 V** for TX Equalization Training.

- CDR
  - i CDR State: Set this to **On** always for 25GBASE-KR and 100GBASE-KR4.

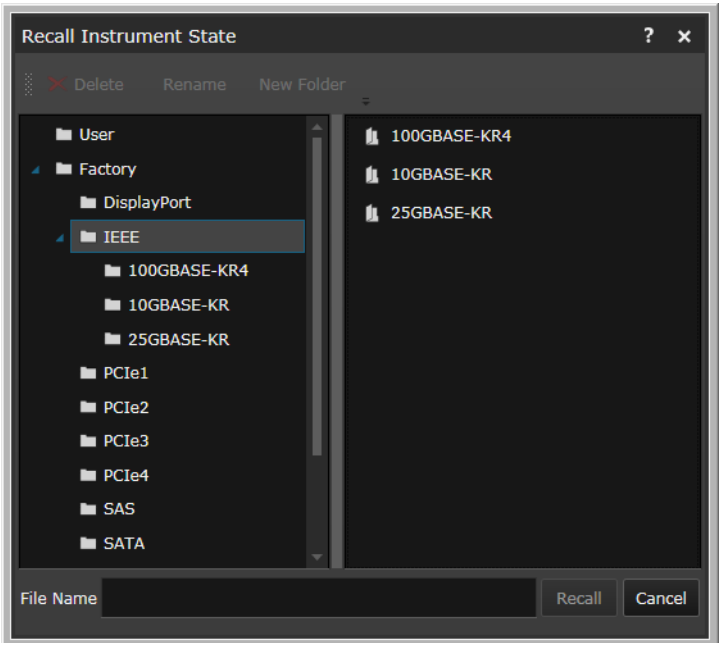
NOTE If the instrument and the DUT are not directly clocked by the same source, a slight frequency offset occurs between the two devices. To compensate this offset, the CDR must be enabled.

You may also apply all the settings mentioned above, which are required for the 10GBASE-KR, 25GBASE-KR and 100GBASE-KR4 operations, using the following steps:

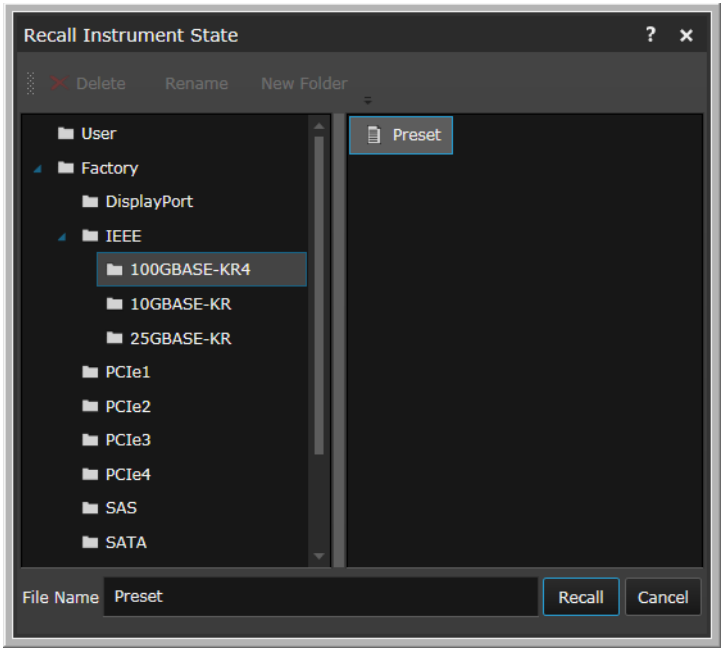
- Go to the **Menu Bar** > **File** and select **Recall Instrument State**. This opens a **Recall Instrument State** dialog as shown in the following figure.
- Select **Factory** > **IEEE**, click **Open**



- Select any of the operation: **100GBASE-KR4**, **10GBASE-KR** or **25GBASE-KR**



- For example you have selected 100GBASE-KR4. A setting file named **Preset** appears, select it and click **Recall** for applying the predefined settings otherwise click **Cancel**.


Recall Instrument State
Delete
Rename
New Folder
User
Factory
DisplayPort
IEEE
100GBASE-KR4
10GBASE-KR
25GBASE-KR
PCIe1
PCIe2
PCIe3
PCIe4
SAS
SATA
Preset
File Name
Preset
Recall
Cancel

Relevant Features of Status Indicators Window

For both the plug-ins, the status bar is located at the bottom of the user interface. The Status Indicators window displays the status for the generator and the analyzer. Some of the global features must be enabled as indicated by green for TX Equalization Training.

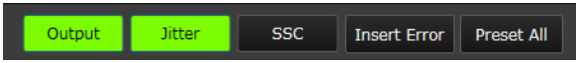


Figure 36 Status Indicators

- Output: The global *Output* state must be set on to obtain a signal out of the BERT generator.
- Jitter: For jitter tolerance testing, the global *Jitter* button should be enabled. Else jitter will be switched off globally.

10GBASE-KR TX EQ Training User Interface

Figure 37 shows the user interface for the “10GBASE-KR TX EQ Training” plug-in:

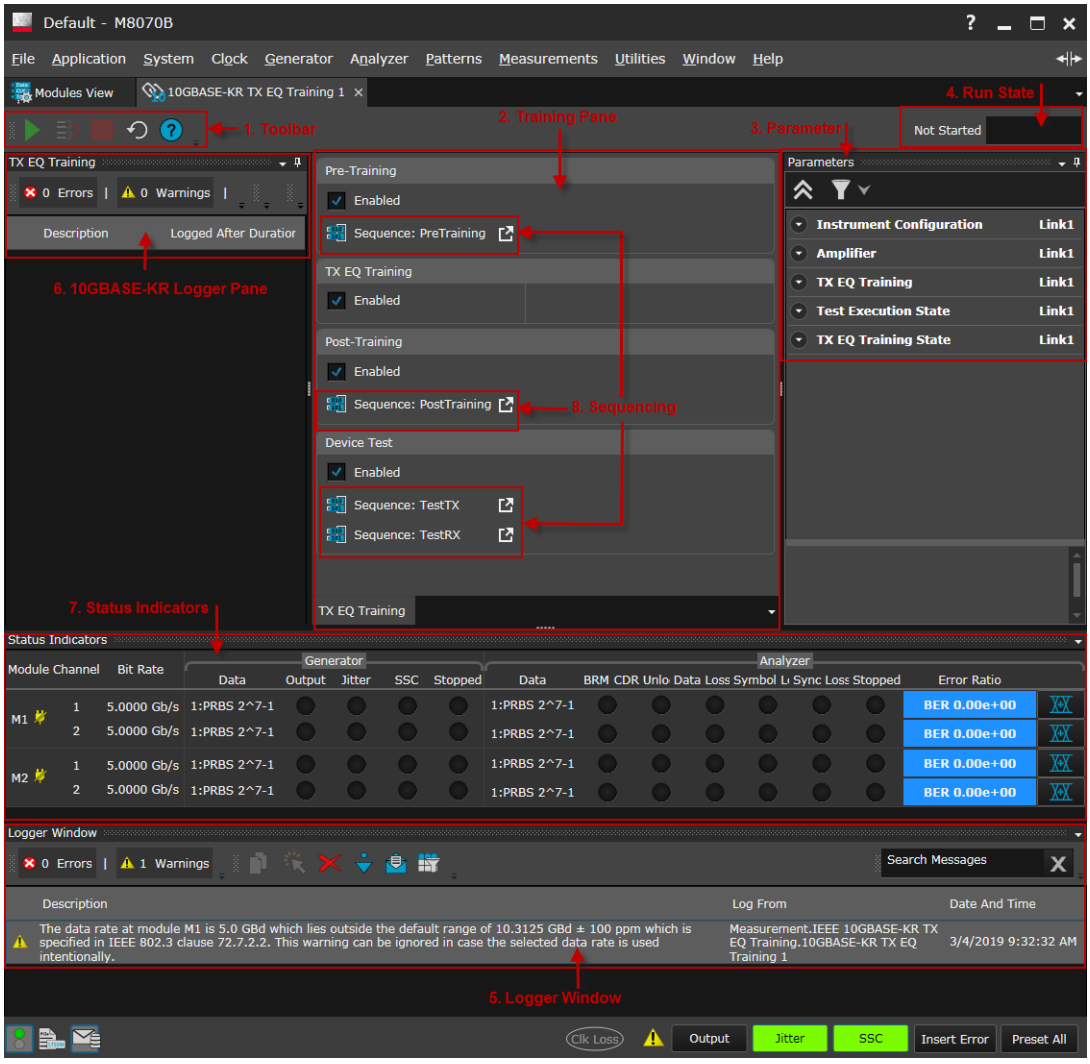


Figure 37 10GBASE-KR TX EQ Training Plug-in User Interface

Following are the elements of user interface:

1 Toolbar
2 Training Pane
3 Parameters
4 Run State
5 Logger Window
6 Logger Pane (10GBASE-KR)
7 Status Indicators
8 Sequencing (MultiGBASE TX EQ Training - Sequence Editor on page 48)

Toolbar

Refer to Toolbar on page 42 (chapter 3).

Training Pane

TX Equalization Training is performed by four blocks:

- Pre-Training
- TX EQ Training
- Post-Training
- Device Test

Pre-Training

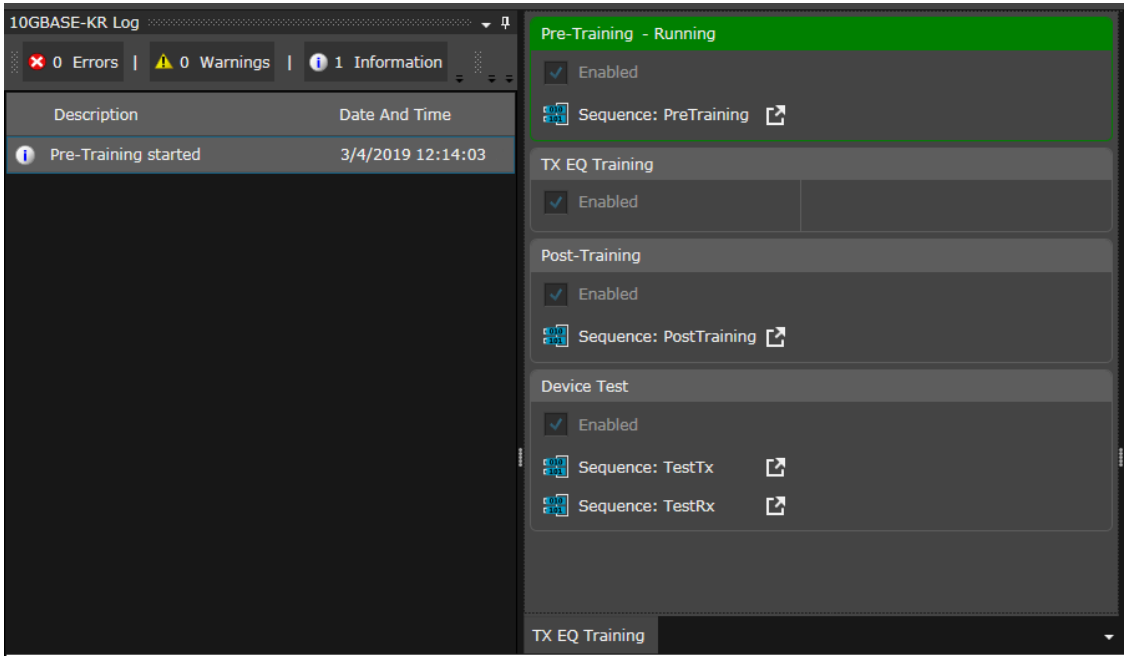


Figure 38 Running Pre-Training

In this state, the10GBASE-KR TX EQ Training plug-in prepares the DUT to accept its start-up protocol. The Pre-Training state brings the DUT from initialization mode to data-exchange (or communication) mode; using either the Pre-Training sequence block or by external means.

NOTE Default sequence is empty for Pre-Training state.

TX EQ Training

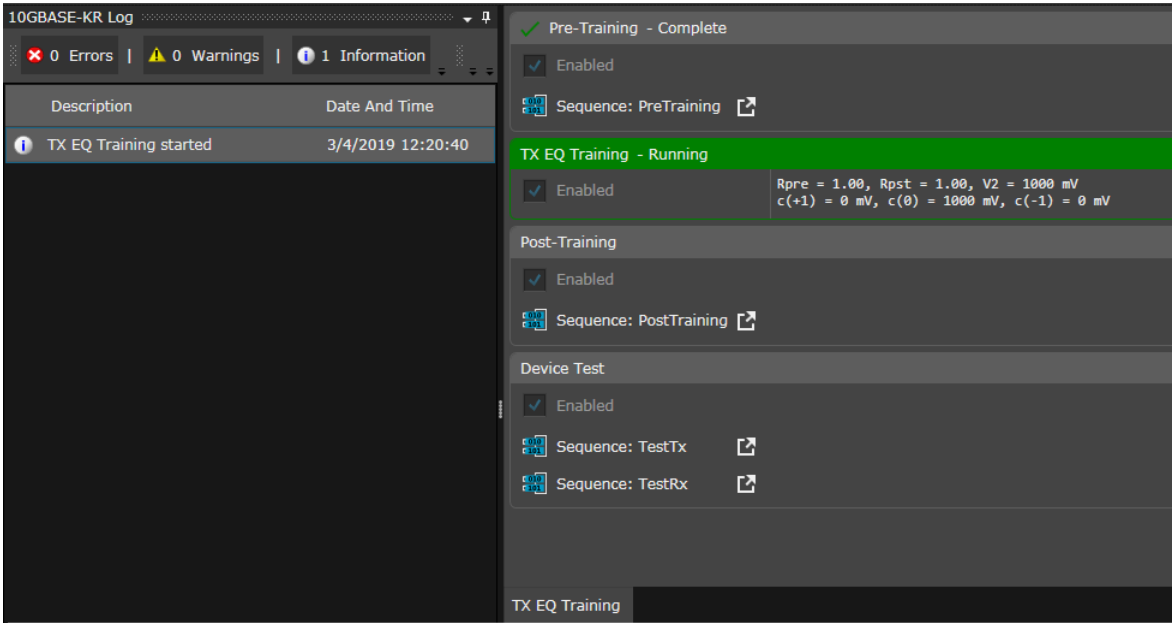


Figure 39 Running TX EQ Training

The purpose of the TX EQ Training block is to configure the ideal de-emphasis setting that fulfills the needs of the DUT. The settings are requested in the training frames sent from the DUT to the instrument's generator. The best values are determined by the data included in the training frames, refer to Training Frame Structure for 10GBASE-KR on page 28 and Training Frame Structure for 25GBASE-KR and 100GBASE-KR4 on page 33.

You can see the values of the De-emphasis settings displayed in the module view. Please be aware that these settings might be shown in a different format than what is expected from the 10GBASE-KR definition. For details, refer to Plug-in Controlled Parameters on page 66.

TX EQ Training Steps

1 The DUT sends a request to the instrument's generator to specify whether it wants to start the TX Equalization Training by Preset or Initialize setting. The same is true for the instrument, which also requests from the DUT to either use the Preset or Initialize setting for its own transmitter. For more information about Preset and Initialize settings, refer to Coefficient update on page 29.

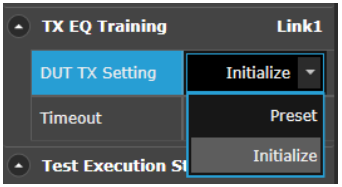


Figure 40 TX EQ Training

2 After the coefficient start condition is defined, the DUT continues to request from the instrument's generator to change its de-emphasis coefficient.

3 This process continues until the DUT receives a signal of sufficient quality at its receiver on the instrument side and indicates to the DUT that it is not capable to fulfill the change request.

4 After the DUT is done with its processing, it sends its receiver ready status to the instrument, which in turn changes its receiver ready state to True and updates the DUT.

5 Once the plug-in completes the de-emphasis settings, the instrument's generator notifies the DUT and sends the settings to the DUT. When the instrument sends the Ready status to the receiver, the TX EQ Training is finished and the sequencer continues to run the Post-Training or Device Test sequence, if defined; else the Training sequence is completed.

Post Training

In this state, the 10GBASE-KR TX EQ Training plug-in runs an optional sequence intended to prepare the DUT for receiver tolerance testing. For example, one such step, which may be run by the 10GBASE-KR TX EQ Training plug-in, is to send a custom preparation frame, bringing the DUT into tolerance testing mode. The DUT can be brought into loopback mode either by using a special sequence within the Post-Training block or by proprietary means.

NOTE Default sequence is empty for Post-Training state.

Device Test

In this state, the 10GBASE-KR TX EQ Training plug-in can run a custom sequence to perform device testing in loop-back mode. For example, this state is intended to perform a jitter tolerance test against the receiver of the DUTs.

NOTE The default sequence for Device Test are:

- TX: PRBS 7-1, infinitely looping
- RX: PRBS 7-1, sync and loop

Parameters

The Parameters pane allows you to change the parameters related to TX Equalization Training, the instruments configuration, amplifier settings and EQ training parameters.

NOTE During test execution, parameter changes are not permitted.

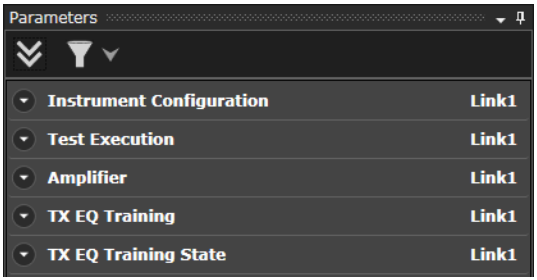


Figure 41 Parameters Pane

Instrument Configuration

Allows you to select a compatible Generator and Analyzer. The 10GBASE-KR TX EQ Training plug-in currently supports two types of modules:

- M8041A 16 Gb/s Generator - Analyzer-Clock
- M8051A 16 Gb/s Generator - Analyzer

NOTE You can use more than one M8051A in a single instrument.

It also provides a shortcut to execute the Analyzer Auto Align.

NOTE

Use the functionality of Analyzer Auto Align with care during plug-in execution as it might have a negative effect on the TX EQ Training procedure.

If auto-alignment is performed during TX EQ Training, the Instruments Analyzer is temporarily unable to capture any data. While auto-alignment is performed, the link training procedure gets into a stalled state but continues uninterrupted after auto-alignment is done.

In case the DUT being tested has a low timeout for TX EQ Training, the duration that the process of auto-alignment takes might be long enough for the DUT to reset its link training procedure, which eventually leads to the interruption in the training.

The parameters provided under the Instrument Configuration branch are shown in Figure 42 and described in Table 15.

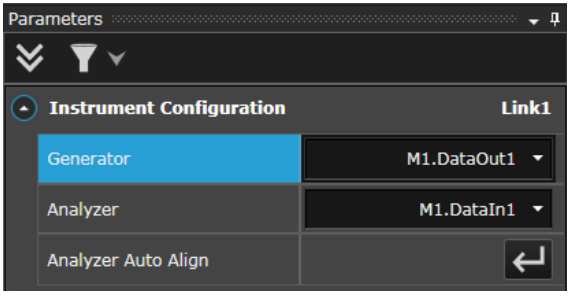


Figure 42 Instrument Configuration Parameters

Table 15 Instrument Configuration Parameters

Name	Description
Generator	Generates an output signal based on the data pattern. It selects the instruments transmitter, used to send data to the receiver of the DUT. Here are some examples: ▪ M1.DataOut1 ▪ M1.DataOut2 ▪ M2.DataOut1 ▪ M2.DataOut2
Analyzer	Examines an incoming bit stream, compares it to the expected pattern and locates inconsistencies. It captures the frames of the DUT's transmitter. Here are some examples: ▪ M1.DataIn1 ▪ M1.DataIn2 ▪ M2.DataIn1 ▪ M2.DataIn2
Analyzer Auto Align	Click the  button to automatically align the selected input to the received signal.

Test Execution State

The parameter provided by Test Execution State is shown in Figure 43 and described in Table 16.

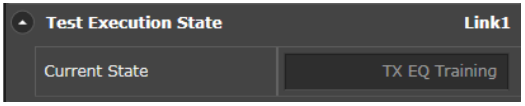


Figure 43 Test Execution State

Table 16 Test Execution State Parameters

Name	Description
Current state	Shows the currently executed test execution state block, following are the possible states: ▪ Idle ▪ PreTraining ▪ TX EQ Training ▪ PostTraining ▪ DeviceTest ▪ Error ▪ Timeout ▪ Aborted

Amplifier

Defines the different voltage levels of the transmitted output waveform of the 10GBASE-KR TX EQ Training plug-in. It has been designed in a way that the voltage output levels are defined and handled as described in the specification for 10GBASE-KR. For more details, refer to Transmitter Equalization Training Procedure on page 36.

The parameters provided under Amplifier are shown in Figure 44 and described in Table 17.

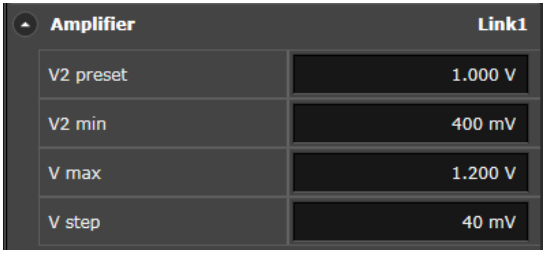


Figure 44 Amplifier Parameter

Table 17 Amplifier Parameter

Name	Description
V2 preset	Differential steady state voltage, used for Preset setting. The preset value does not change. It is applied whenever the DUT requests the preset setting.
V2 min	Minimum differential steady state voltage allowed by the instrument during TX Equalization Training. It is the minimum amplitude, the main cursor of the instrument is allowed to reach. If V2 reaches the boundaries imposed by this setting, the plug-in updates the status report field of its training frame with the corresponding coefficient limits having been reached. This tells the DUT that the instrument cannot apply the previous request.
V max	Maximum differential peak to peak voltage allowed by the test equipment. This maximum value is intended to limit the amplitude range to protect the DUT's receiver circuit from over voltage. If the output amplitude reaches the boundaries imposed by this setting, the plug-in updates its training frame status report field with the corresponding coefficient limits having been reached. This tells the DUT that the Instrument cannot apply the previous request.
V step	Differential step size (Vn(k)-Vn(k-1)) to be used when changing equalization coefficients during TX Equalization Training. It specifies the actual change in voltage between two consecutive de-emphasis cursor settings.

TX EQ Training

TX EQ Training defines which TX state (Preset or Initialize) will be requested from the DUT and the timeout for the TX Equalization Training procedure.

The timeout is reset whenever there is communication between instrument and DUT. Therefore, a timeout occurs only when no answer is received from the DUT for the specified timeout duration.

The respective parameters are shown in Figure 45 and described in Table 18.

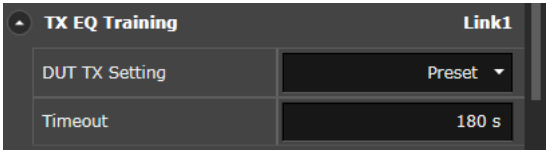


Figure 45 TX EQ Training Parameters

Table 18 TX EQ Training Parameters

Name	Description
DUT TX Setting	Defines which state is requested from the DUT for its TX. It has the following states (for more information on states refer to Coefficient update): ▪ Preset: A preset control request is sent to set the coefficients of the DUTs TX to a state where equalization is turned off. ▪ Initialize: An initialize request is sent to set the coefficients to configure the DUT's transmitter equalizer to INITIALIZE.
Time Out	It occurs when there is no DUT or no answer received in the set time. You can define the timeout from 1 to 3600 seconds.

TX EQ Training State

This functional block shows the parameters of the TX EQ training state corresponding to the definition in IEEE 802.3 subclause 72.7.1.10 onwards. These parameters cannot be controlled directly by any user. They are auto calculated during the interaction between the instrument and DUT that occurs during the process of TX EQ Training. The parameters are directly related to the settings of the de-emphasis parameters.

The parameters provided by TX EQ Training State are shown in Figure 46 and described in Table 19.

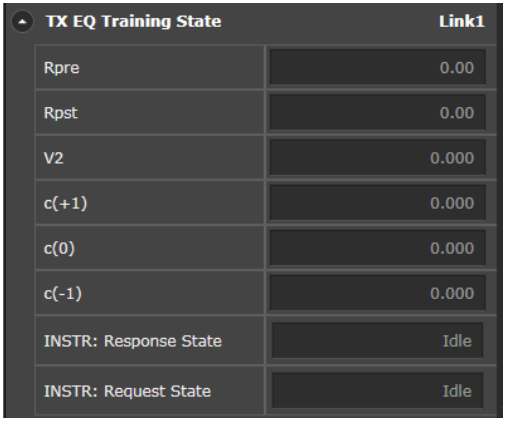


Figure 46 TX EQ Training State Parameters

Table 19 TX EQ Training State Parameters

Name	Description
Rpre	Current pre-cursor equalization ratio used by the plug-in. For more information, refer to Pre- and Post-cursor equalization ratios.
Rpst	Current post-cursor equalization ratio used by the plug-in. For more information, refer to Pre- and Post-cursor equalization ratios.
V2	Current differential steady state voltage used by the instrument.
c(+1)	Current differential post-cursor (deemphasis) value used by the instrument.
c(0)	Current differential main cursor value used by the instrument.
c(-1)	Current differential pre-cursor (Pre-shoot) value used by the instrument.
INSTR: Response State	Current state of the response blocks inside the training frame as seen from the instrument. Following are the possible response states: ▪ Initial Hold State ▪ Requested Update ▪ Acknowledged Update ▪ Idle ▪ Rx Training Finished ▪ Timeout
INSTR: Request State	Current state of the request blocks inside the training frame as seen from the instrument. Possible request states are the same as the response state described above.

Run State

The Run State shows the current state of the plug-in. For details, refer to Run State on page 45 (Chapter 3).

Logger Window

For details, refer to The Global Logger Window on page 46 (Chapter 3).

Logger Pane (10GBASE-KR)

The 10GBASE-KR Log displays the description of different messages types: Errors, Warnings, Information and Debugs. To activate these messages, press the corresponding button. For details, see Figure 47 (shown in red). By default during runtime of the plug-in, the debug messages are disabled. However, if more in depth TX EQ Training details

are required, activate the debug messages by clicking the Debugs button. Once the Debugs option is activated, every step of the TX EQ training procedure is displayed in a separate log message.

NOTE Depending on the screen and windows size, the Debugs button might be hidden. To activate the additional messages, click the small expander arrow on the right side of the visible message buttons.

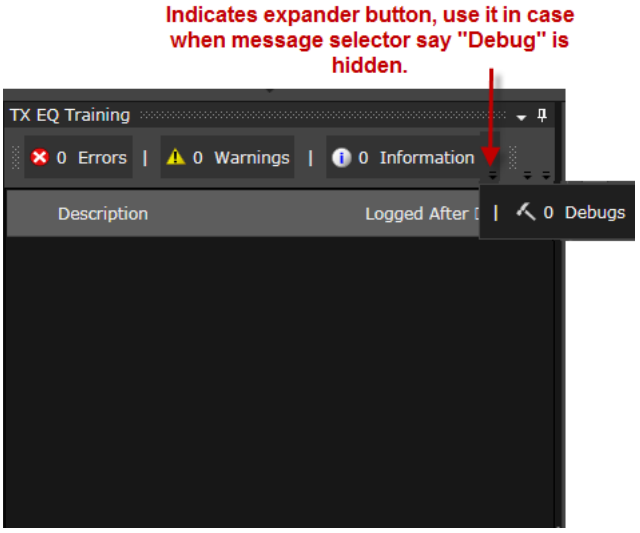


Figure 47 10GBASE-KR Log Expanded View

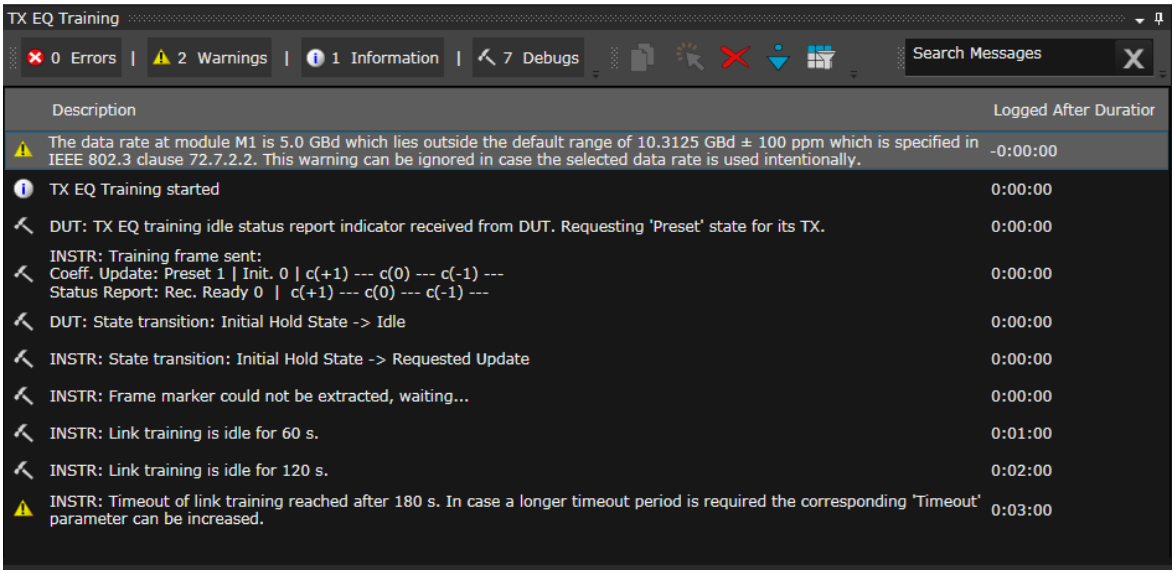


Figure 48 10GBASE-KR Log with activated Debugs

Table 20 10GBASE-KR Log pane description

	Name	Description
	Errors	Shows/hides the errors that occur during execution of the TX Equalization Training.
	Warning	Shows/hides the warnings that occur during execution of the TX Equalization Training.
	Information	Shows/hides information message such as start or stop of the TX EQ training.
	Debugs	Shows/hides debug messages such as the state description, coefficient description, status of frame markers etc.

Toolbar of 10GBASE-KR Log pane

It provides the following shortcuts:

Figure 49 Toolbar of 10GBASE-KR Log

Table 21 Toolbar commands description

	Name	Description
	Copy	Copies the desired messages.
	Select All	Selects all messages.
	Clear Messages	Deletes all messages.
	Enable Auto Scroll To Latest Message	Highlights the most recent message.
	Column Option	Selects/clears the check box to show or hide the following options: ▪ Show Log From ▪ Show Time Stamp

Status Indicators

The Status Indicators window displays the current status of the generator and analyzer ports for each channel of the connected modules. For more information on Status Indicators, refer to *Keysight M8070B User Guide*.

10GBASE-KR TX EQ Training - Sequence Editor

Functionality is the same as described for MultiGBASE TX EQ Training - Sequence Editor on page 48 (Chapter 3).

25GBASE-KR and 100GBASE-KR4 TX EQ Training User Interface

Figure 37 shows the user interface of the “25GBASE-KR and 100GBASE-KR4” plug-in:

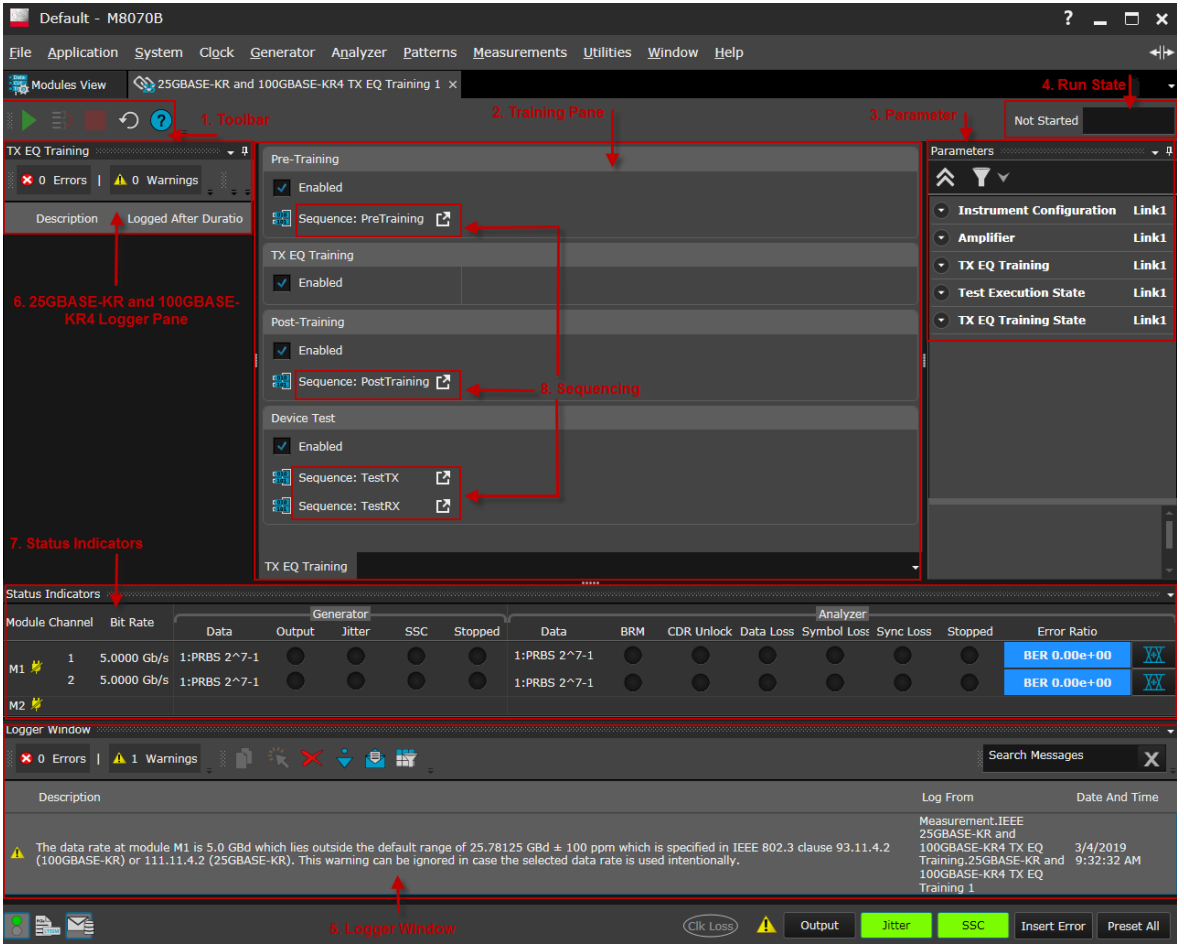


Figure 50 25GBASE-KR and 100GBASE-KR4 TX EQ Training Plug-in User Interface

Following are the elements of the user interface:

1 Toolbar
2 Training Pane
3 Parameters
4 Run State
5 Logger Window
6 Logger Pane (25GBASE-KR and 100GBASE-KR4)
7 Status Indicators
8 Sequencing (MultiGBASE TX EQ Training - Sequence Editor on page 48)

Toolbar

Refer to Toolbar on page 42 (chapter 3).

Training Pane

TX Equalization Training is performed by four blocks:

- Pre-Training
- TX EQ Training
- Post-Training
- Device Test

Pre-Training

Functionality is the same as described for 10GBASE-KR Pre-Training on page 75.

TX EQ Training

Functionality is the same as described for 10GBASE-KR TX EQ Training on page 76.

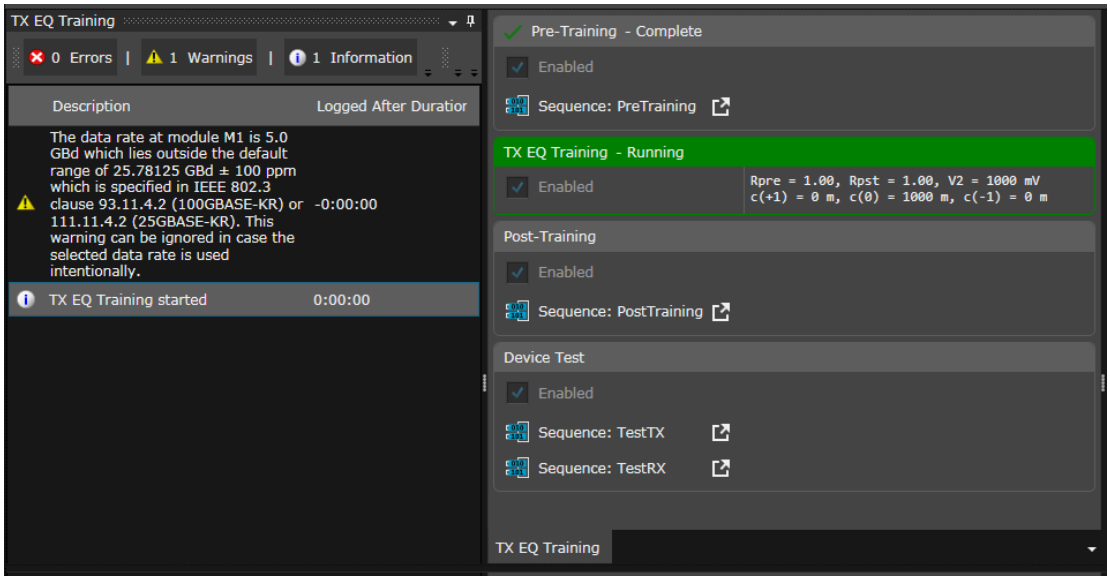


Figure 51 Running the TX EQ Training

Post Training

Functionality is the same as described for 10GBASE-KR Post Training on page 78.

Device Test

Functionality is the same as described for 10GBASE-KR Device Test on page 78.

Parameters

The Parameters pane allows you to change the parameters related to TX Equalization Training, the instruments configuration, amplifier settings and TX EQ training parameters.

NOTE **During test execution, parameter changes are not permitted.**

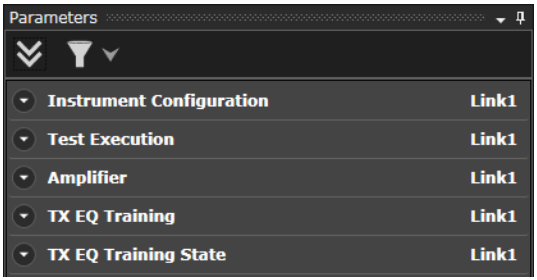


Figure 52 Parameters Pane

Instrument Configuration

Allows you to select a compatible Generator and Analyzer. The 25GBASE-KR and 100GBASE-KR4 TX EQ Training plug-in currently supports two types of modules:

- M8041A 16 Gb/s Generator - Analyzer-Clock
- M8062A 32 Gb/s Generator - Analyzer

It also provides a shortcut to execute the Analyzer Auto Align.

NOTE **Use the functionality of Analyzer Auto Align with care during plug-in execution as it might have a negative effect on the TX EQ Training procedure.**

The parameters provided under the Instrument Configuration branch are shown in Figure 53 and described in Table 22.

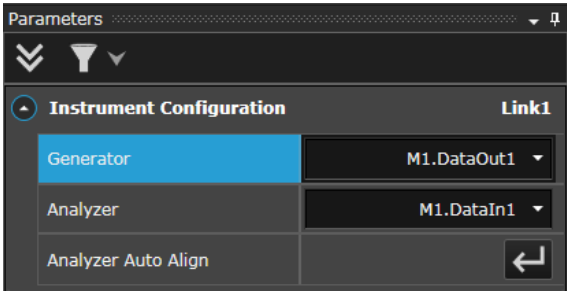


Figure 53 Instrument Configuration Parameters

Table 22 Instrument Configuration Parameters

Name	Description
Generator	Generates an output signal based on the data pattern. It selects the instruments transmitter, used to send data to the receiver of the DUT.
Analyzer	Examines an incoming bit stream, compares it to the expected pattern and locates inconsistencies. It captures the frames of the DUT's transmitter.
Analyzer Auto Align	Click the  button to automatically align the selected input to the received signal.

NOTE

The available channel identifiers depend on the module configuration inside the system. Generally, for the 25GBASE-KR and 100GBASE-KR4 Training plug-in, the default channel number will be M2.DataOut and M2.DataIn (Without a number after DataOut or DataIn) because the M8062A module contains an M8041A module below. Therefore, the M8041A module resides in slots 1 to 3 and the M8062A module in slots 4 to 5. This is how the channel identifier values mentioned above are attained. However, depending on the M8062A module number, the channel identifier value may be different, such as M3.DataOut and so on.

Amplifier

Functionality is the same as described for 10GBASE-KR Amplifier on page 82.

TX EQ Training

TX EQ Training defines which TX state (Preset or Initialize) will be requested from the DUT and the timeout for the TX Equalization Training procedure.

The timeout is reset whenever there is communication between instrument and DUT. Therefore, a timeout occurs only when no answer is received from the DUT for the specified timeout duration.

The respective parameters are shown in Figure 54 and described in Table 23.

Figure 54 TX EQ Training Parameters

Table 23 TX EQ Training Parameters

Name	Description
DUT TX Setting	Defines which state is requested from the DUT for its TX. It has the following states (for more information on states refer to Coefficient update): ▪ Preset: A preset control request is sent to set the coefficients to a state where equalization is turned off. For more details, refer to ▪ Initialize: An initialize request is sent to set the coefficients to configure the transmit equalizer to INITIALIZE.
Time Out	It occurs when there is no DUT or no answer received in the set time. You can define the timeout from 1 sec to 3600 seconds.
PRBS Type	Specifies which PRBS polynomial is used for the training pattern. For 25GBASE-KR there is only one lane “Lane 0” and for 100GBASE-KR4 there are four lanes i.e. “Lane 0, Lane 1, Lane 2 and Lane 3”. Following are the polynomials for different lanes: ▪ Lane 0: $1 + x^5 + x^6 + x^{10} + x^{11}$ ▪ Lane 1: $1 + x^5 + x^6 + x^9 + x^{11}$ ▪ Lane 2: $1 + x^4 + x^6 + x^8 + x^{11}$ ▪ Lane 3: $1 + x^4 + x^6 + x^7 + x^{11}$

Test Execution State

Functionality is the same as described for 10GBASE-KR Test Execution State on page 81.

TX EQ Training State

Functionality is the same as described for 10GBASE-KR TX EQ Training State on page 84.

Run State

The Run State shows the current state of the plug-in. For details, refer to Run State on page 45 (Chapter 3).

Logger Window

Refer to The Global Logger Window on page 46 (Chapter 3).

Logger Pane (25GBASE-KR and 100GBASE-KR4)

Functionality is the same as described for Logger Pane (10GBASE-KR) on page 85.

Status Indicators

The Status Indicators window displays the current status of the generator and analyzer ports for each channel of the connected modules. For more information on Status Indicators, refer to *Keysight M8070B User Guide*.

25GBASE-KR and 100GBASE-KR4 TX EQ Training - Sequence Editor

Functionality is the same as described for MultiGBASE TX EQ Training - Sequence Editor on page 48 (Chapter 3).

Device Tolerance Testing

To perform tolerance testing of a test chip, you must perform some steps that are specific to each chip type. A tolerance test can be executed by sending a signal from the BERT to the DUT which receives and samples this signal and loops it back to the BERT. Activating this functionality depends on the specific DUT.

NOTE The procedure for the Device tolerance testing is the same for both plug-ins. In this section, the test procedure for 10GBASE-KR DUT only has been described.

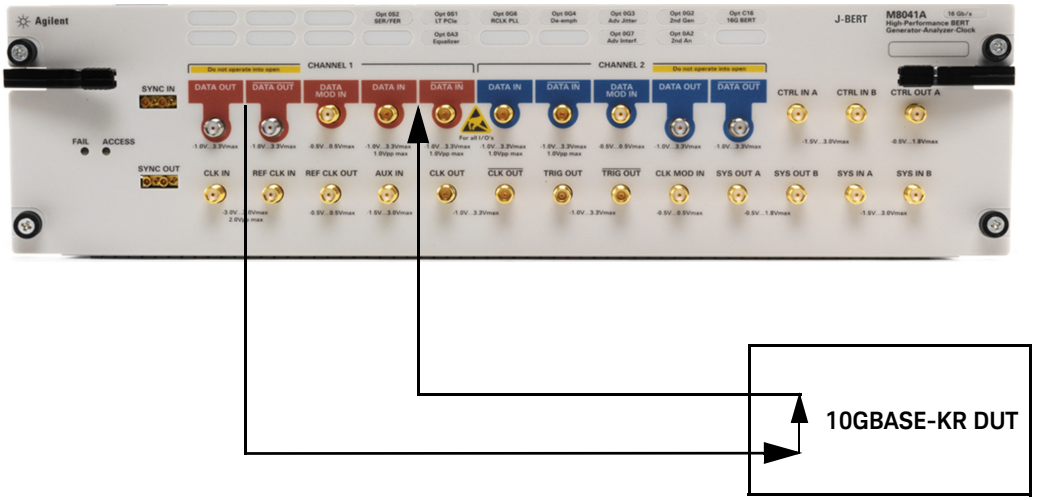


Figure 55 BERT Connection set-up with 10GBASE-KR DUT for Device Tolerance Testing

The BERT is expected to receive the same data as it had sent earlier to the DUT. If the looped back signal contains bit errors, it indicates that the receiver of the DUT is unable to tolerate the amount of jitter or interference and therefore, could not interpret the data correctly.

General Procedure

1 Connect the Generator of the BERT to the receiver of the DUT and the Analyzer of the BERT to the transmitter of the DUT.

2 Configure the DUT to attain a state where it expects TX EQ training frames to be received. To achieve this, you may be required to use a serial programming interface or to enable a special bit sequence. In the latter case, use the Pre-Training sequence block of the plug-in to setup such a sequence.

3 Edit the test pattern in the Post-Training and the TX EQ Training blocks to allow for loopback testing. The settings are:

   a Post-Training block - No sequence

   b TestTX - An infinitely looped PRBS7-1

   c TestRX - An infinitely looped PRBS7-1 setup as sync and loop to allow the BERT to synchronize to the DUT's loopback signal.

   The default state of 10GBASE-KR TX EQ Training - Sequence window contains the above mentioned sequence snippets.

4 Start the execution of the TX EQ Link Training plug-in and let it train the DUT's receiver. Once the DUT sends the finished bit inside the training frame to the BERT, the TX EQ Training plug-in also performs the same action and thereby concludes the equalization training.

5 After TX EQ Training succeeded, the BERT plays back the Post-Training block (empty sequence as default) and then the Device Test block (Default: PRBS7-1 sequence snippet used for both Generator and Analyzer).

6 Program the DUT by custom means into loopback mode is required.

7 Start with application/device specific tolerance testing.

Jitter Tolerance

To conveniently perform jitter tolerance testing, there is a separate application specific view available.

To launch this interface: Go to **Measurements** > **Jitter Tolerance**

This measurement can be used in conjunction with the 10GBASE-KR plug-in, starting at step 7. For more information on how to use this view, refer to the *M8070B User Guide*.

Figure 56 Jitter Tolerance

5 SCPI Command Reference

KEYSIGHT TECHNOLOGIES

Common SCPI Commands

For each GUI control there is a corresponding SCPI command/query. The SCPI programming reference contains only simple descriptions for each command/query, for full details, refer to the corresponding GUI sections of this User Guide.

The SCPI Commands are common for both plug-ins ('10GBASE-KR' & '25GBASE-KR and 100GBASE-KR4'). But in SCPI syntax you need to do the following changes for the specific plug-in:

- Plug-in name (represented by [...] in general SCPI syntax)
  - For 10GBASE-KR: "LTXGKR"
  - For 25GBASE-KR and 100GBASE-KR4: "LTCGKR"
- Identifier example
  - For 10GBASE-KR: '10GBASE-KR TX EQ Training 1'
  - For 25GBASE-KR and 100GBASE-KR4: '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

Table 24 Common SCPI Commands

Command	Description	Reference
Plug-in Framework Commands		For details, see Plug-in Framework Commands on page 102
:PLUGin:[...]:AMPlifier:VMAXimum[?]	Defines the maximum differential peak-peak voltage allowed by the test equipment.	For details, see page no.108
:PLUGin:[...]:AMPlifier:VMINSteady[?]	Defines the minimum differential steady state voltage allowed by the instrument.	For details, see page no.109
:PLUGin:[...]:AMPlifier:VSTEady[?]	Defines the differential steady state voltage used for preset setting.	For details, see page no. 109
:PLUGin:[...]:AMPlifier:VSTEPsize[?]	Defines the differential step size to be used when changing equalization coefficients during TX Equalization Training.	For details, see page no. 110
:PLUGin:[...]:BLOCk:POSTraining:ENABled[?]	Enables or disables the post training state.	For details, see page no. 111
:PLUGin:[...]:BLOCk:POSTraining:SEQuence:TX:VALue[?]	Sets the sequence settings for the Post Training.	For details, see page no. 111
:PLUGin:[...]:BLOCk:POSTraining:STATe?	Returns the state of post training.	For details, see page no. 112
:PLUGin:[...]:BLOCk:PRETraining:ENABled[?]	Enables or disables the Pre-Training state.	For details, see page no. 113

Command	Description	Reference
:PLUGin:[...]:BLOCk:PRETraining:SEQuence:TX:VALue[?]	Sets the sequence settings for the Pre-Training.	For details, see page no. 113
:PLUGin:[...]:BLOCk:PRETraining:STATe?	Returns the state of Pre-Training.	For details, see page no. 114
:PLUGin:[...]:BLOCk:SEQuence:BREak	Breaks out all the loops which are defined as breakable and currently running.	For details, see page no. 115
:PLUGin:[...]:BLOCk:TEST:ENABled[?]	Enables or disables the device test state.	For details, see page no. 115
:PLUGin:[...]:BLOCk:TEST:SEQuence:RX:VALue[?]	Sets the sequence settings to the receiver of device.	For details, see page no. 116
:PLUGin:[...]:BLOCk:TEST:SEQuence:TX:VALue[?]	Sets the sequence settings to the transmitter of device.	For details, see page no. 117
:PLUGin:[...]:BLOCk:TEST:STATe?	Returns the state of post training.	For details, see page no. 118
:PLUGin:[...]:BLOCk:TXEQ:ENABled[?]	Enables or disables the TX EQ state.	For details, see page no. 118
:PLUGin:[...]:BLOCk:TXEQ:STATe?	Returns current state of TX EQ training.	For details, see page no. 119
:PLUGin:[...]:INSTrument:DINput[?]	Defines the local receiver which is used to capture frames from the transmitter of the DUT.	For details, see page no. 119
:PLUGin:[...]:INSTrument:DINput:ALIGnment	Aligns the selected input to the received signal.	For details, see page no. 120
:PLUGin:[...]:INSTrument:DOUTput[?]	Defines the local transmitter which is used to communicate with the receiver of the DUT.	For details, see page no. 120
:PLUGin:[...]:LTraining:DUTState[?]	Defines which state is requested from the DUT for its transmitter.	For details, see page no. 121
:PLUGin:[...]:LTraining:RESult?	Returns the result of TX Equalization Training.	For details, see page no. 121
:PLUGin:[...]:LTraining:STATe:CMAIn?	Returns the current differential main cursor value used by the instrument.	For details, see page no. 122
:PLUGin:[...]:LTraining:STATe:CMinus?	Returns the current differential pre-cursor (pre-shoot) value used by the instrument.	For details, see page no. 122
:PLUGin:[...]:LTraining:STATe:CPlus?	Returns the current differential steady state voltage setting used by the instrument.	For details, see page no. 122
:PLUGin:[...]:LTraining:STATe:IREQuest?	Returns the current state of the request blocks inside the training frame as see page no.n from the plug-in.	For details, see page no. 123
:PLUGin:[...]:LTraining:STATe:IRESponse?	Returns the current state of the response blocks inside the training frame as see page no.n from the plug-in.	For details, see page no. 123
:PLUGin:[...]:LTraining:STATe:RPRE?	Returns the value of current pre-cursor equalization ratio used by the instrument.	For details, see page no. 124
:PLUGin:[...]:LTraining:STATe:RPST?	Returns the value of current post-cursor equalization ratio used by the instrument.	For details, see page no. 124

Command	Description	Reference
:PLUGin:[...]:LTraining:STATe:VMAIn?	Returns the value of current differential steady state voltage setting used by the instrument.	For details, see page no. 124
:PLUGin:[...]:LTraining:TIMEout[?]	Defines the maximum time which may be spent inside any of the TX Equalization Training States, before training is aborted.	For details, see page no. 125
:PLUGin:[...]:TEXEcution:STATe?	Returns current training state.	For details, see page no. 125

Plug-in Framework Commands

Plug-in framework commands are a set of "general" SCPI commands for controlling the TX EQ Training plug-ins.

Table 25 Plug-in Framework Commands

Command	Description	Reference
:PLUGin:[...]:CATalog?	Returns the active instance name of the TX EQ Training plug-in. If multiple plug-in instances are open, it returns the name of most recent plug-in instance.	For details, see page no.103
:PLUGin:[...]:DELete	Closes the active instance of the TX EQ Training plug-in. If multiple plug-in instances are open, it closes the last plug-in instance.	For details, see page no.103
:PLUGin:[...]:NEW	Opens a new instance of TX EQ Training plug-in.	For details, see page no.104
:PLUGin:[...]:RESet	Resets the state of TX EQ Training plug-in to its default values.	For details, see page no.104
:PLUGin:[...]:RUN:LOG?	Returns the logs of the plug-in.	For details, see page no. 105
:PLUGin:[...]:RUN:MESSage?	Returns a value describing the state of the plug-in.	For details, see page no. 105
:PLUGin:[...]:RUN:PROGress?	Returns a number in the range of 0.0 to 1.0 to indicate the progress of the plug-in. A '0.0' indicates that the plug-in is idle and '1.0' indicates that the plug-in execution has been finished.	For details, see page no. 106
:PLUGin:[...]:RUN:STATus?	Returns the running status of the plug-in. A '0' indicates the plug-in is not running and a '1' indicates that the plug-in execution has been finished.	For details, see page no. 107
:PLUGin:[...]:STARt	Used to start the plug-in execution.	For details, see page no. 107
:PLUGin:[...]:STOP	Used to stop the plug-in execution.	For details, see page no. 108

:PLUGin:[...]:CATalog?

Query :PLUGin:LTXGKR:CATalog?

:PLUGin:LTCGKR:CATalog?

Return Value Name of each active plug-in instance, such as '10GBASE-KR TX EQ Training 1' or '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'.

Description Based on the variable used (LTCGKR or LTXGKR) in the command, this query returns the respective names of all active plug-in instances of the TX EQ Training plug-in.

For example, if three plug-in instances of '25GBASE-KR and 100GBASE-KR4 TX EQ Training' are active, the variable 'LTCGKR' in the query would return '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1','25GBASE-KR and 100GBASE-KR4 TX EQ Training 2','25GBASE-KR and 100GBASE-KR4 TX EQ Training 3'.

Example :PLUGin:LTXGKR:CATalog?

:PLUGin:LTCGKR:CATalog?

:PLUGin:[...]:DELete

Command :PLUGin:LTXGKR:DELete

:PLUGin:LTCGKR:DELete

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it deletes the active plug-in instances of the TX EQ Training plug-in. If multiple plug-in instances are active, for example three plug-in instances of '25GBASE-KR and 100GBASE-KR4 TX EQ Training' are active, the variable 'LTCGKR' in the command would delete the initially active plug-in instance.

The syntax for this command includes an optional parameter called '<Instance Name>'. If you specify this variable in the command syntax, it deletes the specified plug-in instance instead of the initially active plug-in instance.

Example :PLUGin:LTXGKR:DELete

:PLUGin:LTCGKR:DELete

:PLUGin:LTXGKR:DELete '10GBASE-KR TX EQ Training 2'

:PLUGin:LTCGKR:DELete '25GBASE-KR and 100GBASE-KR4 TX EQ Training 3'

:PLUGin:[...]:NEW

Command :PLUGin:LTXGKR:NEW '<Instance Name>'

:PLUGin:LTCGKR:NEW '<Instance Name>'

Input Parameters '<Instance Name>': Assign any name to the plug-in instance.

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it opens a new instance of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:NEW '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:NEW '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:RESet

Command :PLUGin:LTXGKR:RESet

:PLUGin:LTCGKR:RESet

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it resets the active plug-in instances of the TX EQ Training plug-in to its default values. If multiple plug-in instances are active, for example three plug-in instances of '10GBASE-KR TX EQ Training' are active, the variable 'LTXGKR' in the command would reset the initially active plug-in instance.

The syntax for this command includes an optional parameter called '<Instance Name>'. If you specify this variable in the command syntax, it resets the specified plug-in instance instead of the initially active plug-in instance.

Example :PLUGin:LTXGKR:RESet

:PLUGin:LTCGKR:RESet

:PLUGin:LTXGKR:RESet '10GBASE-KR TX EQ Training 2'

:PLUGin:LTCGKR:RESet '25GBASE-KR and 100GBASE-KR4 TX EQ Training 3'

:PLUGin:[...]:RUN:LOG?

Command :PLUGin:LTXGKR:RUN:LOG?

:PLUGin:LTCGKR:RUN:LOG?

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns logs of the active plug-in instances of the TX EQ Training plug-in. If multiple plug-in instances are active, for example three plug-in instances of '10GBASE-KR TX EQ Training' are active, the variable 'LTXGKR' in the query would return the logs for initially active plug-in instance.

The syntax for this command includes an optional parameter called '<Instance Name>'. If you specify this variable in the query syntax, it returns the logs for the specified plug-in instance instead of the initially active instance.

Example :PLUGin:LTXGKR:RUN:LOG?

:PLUGin:LTCGKR:RUN:LOG?

:PLUGin:LTXGKR:RUN:LOG? '10GBASE-KR TX EQ Training 2'

:PLUGin:LTCGKR:RUN:LOG? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 3'

:PLUGin:[...]:RUN:MESSage?

Command :PLUGin:LTXGKR:RUN:MESSage?

:PLUGin:LTCGKR:RUN:MESSage?

Return Value NotStarted | Running | Finished | Error | Stopped

Description This query returns a value describing the state of the plug-in.

Based on the variable used (LTCGKR or LTXGKR) in the query, it returns a value describing the state of the active plug-in instances of the TX EQ Training plug-in. But if multiple plug-in instances are active, for example three plug-in instances of '25GBASE-KR and 100GBASE-KR4 TX EQ Training' are active, the variable 'LTCGKR' in the command would return a value describing the state of the initially active plug-in instance.

The syntax for this command includes an optional parameter called '<Instance Name>'. If you specify this variable in the command syntax, it return a value describing the state of the specified plug-in instance instead of the initially active plug-in instance.

Example :PLUGin:LTXGKR:RUN:MESSage? '

:PLUGin:LTCGKR:RUN:MESSage? '

:PLUGin:LTXGKR:RUN:MESSage? '10GBASE-KR TX EQ Training 3'

:PLUGin:LTCGKR:RUN:MESSage? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 2'

:PLUGin:[...]:RUN:PROGress?

Command **:PLUGin:LTXGKR:RUN:PROGress?** '

:PLUGin:LTCGKR:RUN:PROGress? '

Return Value 0.0 | 1.0

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns a number in the range of 0.0 to 1.0 to indicate the progress of the active plug-in instances of the TX EQ Training plug-in. A '0.0' indicates that the plug-in is idle and '1.0' indicates that the plug-in execution has been finished.

If multiple plug-in instances are active, for example three plug-in instances of '25GBASE-KR and 100GBASE-KR4 TX EQ Training' are active, the variable 'LTCGKR' in the query would return the progress for the initially active plug-in instance.

The syntax for this command includes an optional parameter called '<Instance Name>'. If you specify this variable in the command syntax, it returns the progress for the specified plug-in instance instead of the initially active plug-in instance.

Example :PLUGin:LTXGKR:RUN:PROGress?

:PLUGin:LTCGKR:RUN:PROGress?

:PLUGin:LTXGKR:RUN:PROGress? '10GBASE-KR TX EQ Training 4'

:PLUGin:LTCGKR:RUN:PROGress? '25GBASE-KR and 100GBASE-KR4 3'

:PLUGin:[...]:RUN:STATus?

Command :PLUGin:LTXGKR:RUN:STATus?

:PLUGin:LTCGKR:RUN:STATus?

Return Value 0 |1

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns the running status of the active plug-in instances of the TX EQ Training plug-in. A '0' indicates the plug-in is not running and a '1' indicates that the plug-in execution has been finished.

If multiple plug-in instances are active, for example four plug-in instances of '25GBASE-KR and 100GBASE-KR4 TX EQ Training' are active, the variable 'LTCGKR' in the command would returns the running status of the initially active plug-in instance.

The syntax for this command includes an optional parameter called '<Instance Name>'. If you specify this variable in the command syntax, it would returns the running status of the specified plug-in instance instead of the initially active plug-in instance.

Example :PLUGin:LTXGKR:RUN:STATus?

PLUGin:LTCGKR:RUN:STATus?

:PLUGin:LTXGKR:RUN:STATus? '10GBASE-KR4 TX EQ Training 2'

PLUGin:LTCGKR:RUN:STATus? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 2'

:PLUGin:[...]:STARt

Command :PLUGin:LTXGKR:STARt

:PLUGin:LTCGKR:STARt

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it starts the plug-in execution the active plug-in instances of the TX EQ Training plug-in. If multiple plug-in instances are active, for example three plug-in instances of '25GBASE-KR and 100GBASE-KR4 TX EQ Training' are active, the variable 'LTCGKR' in the command would start the plug-in execution of the initially active plug-in instance.

The syntax for this command includes an optional parameter called '<Instance Name>'. If you specify this variable in the command syntax, it starts the plug-in execution of the specified plug-in instance instead of the initially active plug-in instance.

Example :PLUGin:LTXGKR:STARt

:PLUGin:LTCGKR:STARt

:PLUGin:LTXGKR:STARt '10GBASE-KR4 TX EQ Training 2'

:PLUGin:LTCGKR:STARt '25GBASE-KR and 100GBASE-KR4 TX EQ Training 2'

:PLUGin:[...]:STOP

Command **:PLUGin:LTXGKR:STOP**

:PLUGin:LTCGKR:STOP

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it stops the plug-in execution the active plug-in instances of the TX EQ Training plug-in. If multiple plug-in instances are active, for example three plug-in instances of '25GBASE-KR and 100GBASE-KR4 TX EQ Training' are active, the variable 'LTCGKR' in the command would stop the plug-in execution of the initially active plug-in instance.

The syntax for this command includes an optional parameter called '<Instance Name>'. If you specify this variable in the command syntax, it stops the plug-in execution of the specified plug-in instance instead of the initially active plug-in instance.

Example :PLUGin:LTXGKR:STOP

:PLUGin:LTCGKR:STOP

:PLUGin:LTXGKR:STOP '10GBASE-KR4 TX EQ Training 2'

:PLUGin:LTCGKR:STOP '25GBASE-KR and 100GBASE-KR4 TX EQ Training 2'

:PLUGin:[...]:AMPlifier:VMAXimum[?]

Command **:PLUGin:LTXGKR:AMPlifier:VMAXimum 'Identifier', <Nrf>**

:PLUGin:LTCGKR:AMPlifier:VMAXimum 'Identifier', <Nrf>

Query **:PLUGin:LTXGKR:AMPlifier:VMAXimum? 'Identifier'**

:PLUGin:LTCGKR:AMPlifier:VMAXimum? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is 25GBASE-KR and 100GBASE-KR4'

<Nrf>: Enter the value of maximum differential voltage.

Return Range 0 to 2.4 V. The range depends on setting of V2 min and V max.

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it defines the maximum differential peak-peak voltage allowed by the test equipment of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:AMPlifier:VMAXimum '10GBASE-KR TX EQ Training 1', 2

:PLUGin:LTCGKR:AMPlifier:VMAXimum '25GBASE-KR and 100GBASE-KR4TX EQ Training 1', 1

:PLUGin:LTXGKR:AMPlifier:VMAXimum? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:AMPlifier:VMAXimum? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:AMPlifier:VMINSteady[?]

Command :PLUGin:LTXGKR:AMPlifier:VMINSteady 'Identifier', <Nrf>

:PLUGin:LTCGKR:AMPlifier:VMINSteady 'Identifier', <Nrf>

Query :PLUGin:LTXGKR:AMPlifier:VMINSteady? 'Identifier'

:PLUGin:LTCGKR:AMPlifier:VMINSteady 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<Nrf>: Enter the value of minimum differential steady state voltage.

Return Range 0 mV to 990 mV

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it defines the minimum differential steady state voltage allowed by the test equipment of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:AMPlifier:VMINSteady '10GBASE-KR TX EQ Training 1', 60 mV

:PLUGin:LTCGKR:AMPlifier:VMINSteady '25GBASE-KR and 100GBASE-KR4TX EQ Training 1', 60 mV

:PLUGin:LTXGKR:AMPlifier:VMINSteady? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:AMPlifier:VMINSteady? '25GBASE-KR and 100GBASE-KR4TX EQ Training 1'

:PLUGin:[...]:AMPlifier:VSTEady[?]

Command :PLUGin:LTXGKR:AMPlifier:VSTEady 'Identifier', <Nrf>

:PLUGin:LTCGKR:AMPlifier:VSTEady 'Identifier', <Nrf>

Query :PLUGin:LTXGKR:AMPlifier:VSTEady? 'Identifier'

:PLUGin:LTCGKR:AMPlifier:VSTEady? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<Nrf>: Enter the value of differential voltage for preset setting.

Return Range 0 to 2.4 V

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it defines the differential steady state voltage used for preset setting of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:AMPlifier:VSTEady '10GBASE-KR TX EQ Training 1', 1.1

:PLUGin:LTCGKR:AMPlifier:VSTEady '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', 1.1

:PLUGin:LTXGKR:AMPlifier:VSTEady? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:AMPlifier:VSTEady? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:AMPlifier:VSTEPsize[?]

Command **:PLUGin:LTXGKR:AMPlifier:VSTEPsize 'Identifier', <Nrf>**

:PLUGin:LTCGKR:AMPlifier:VSTEPsize ''Identifier', <Nrf>

Query **:PLUGin:LTXGKR:AMPlifier:VSTEPsize? 'Identifier'**

:PLUGin:LTCGKR:AMPlifier:VSTEPsize? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<Nrf>: Enter the value of differential step size voltage.

Return Range 1 mV to 100 mV

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it defines the differential step size to be used when changing equalization coefficients during TX Equalization Training of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:AMPlifier:VSTEPsize '10GBASE-KR TX EQ Training 1', 50 mV

:PLUGin:LTCGKR:AMPlifier:VSTEPsize '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', 25 mV

:PLUGin:LTXGKR:AMPlifier:VSTEPsize? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:AMPlifier:VSTEPsize? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:POSTraining:ENABled[?]

Command :PLUGin:LTXGKR:BLOCk:POSTraining:ENABled 'Identifier', <Bool>

:PLUGin:LTCGKR:BLOCk:POSTraining:ENABled 'Identifier', <Bool>

Query :PLUGin:LTXGKR:BLOCk:POSTraining:ENABled? 'Identifier'

:PLUGin:LTCGKR:BLOCk:POSTraining:ENABled? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<Bool>: Enter 1/ON or 0/OFF to enable or disable the post training state.

Return Value 1 | 0

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it enables or disables the post training state of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:BLOCk:POSTraining:ENABled '10GBASE-KR TX EQ Training 1', ON

:PLUGin:LTCGKR:BLOCk:POSTraining:ENABled '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', ON

:PLUGin:LTXGKR:BLOCk:POSTraining:ENABled? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:POSTraining:ENABled? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:POSTraining:SEQuence:TX:VALue[?]

Command :PLUGin:LTXGKR:BLOCk:POSTraining:SEQuence:TX:VALue 'Identifier', <sequence setting>

:PLUGin:LTCGKR:BLOCk:POSTraining:SEQuence:TX:VALue 'Identifier', <sequence setting>

Query :PLUGin:LTXGKR:BLOCk:POSTraining:SEQuence:TX:VALue? 'Identifier'

:PLUGin:LTCGKR:BLOCk:POSTraining:SEQuence:TX:VALue 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<sequence setting>: Specify sequence setting.

Return Value Sequence setting.

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it sets the sequence settings for the Post Training of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:BLOCk:POSTraining:SEQuence:TX:VALue '10GBase-KR TX EQ Training 1', '<?xml version="1.0" encoding="utf-16"?><sequenceDefinition xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.keysight.com/schemas/M8000/DataSequence"><sequence><loop><block length="128"><prbs polynomial="2^7-1" /></block></loop></sequence></sequenceDefinition>'

:PLUGin:LTCGKR:BLOCk:POSTraining:SEQuence:TX:VALue '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', '<?xml version="1.0" encoding="utf-16"?><sequenceDefinition xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.keysight.com/schemas/M8000/DataSequence"><sequence><loop><block length="128"><prbs polynomial="2^7-1" /></block></loop></sequence></sequenceDefinition>'

:PLUGin:LTXGKR:BLOCk:POSTraining:SEQuence:TX:VALue? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:POSTraining:SEQuence:TX:VALue? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:POSTraining:STATe?

Query **:PLUGin:LTXGKR:BLOCk:POSTraining:STATe?** '**Identifier**'

PLUGin:LTCGKR:BLOCk:POSTraining:STATe? '**Identifier**'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is 25GBASE-KR and 100GBASE-KR4'

Return Value Not Yet Run | Not Completed | Completed

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns the state of post training of the TX EQ Training plug-in.

This query returns the state of post training.

Example :PLUGin:LTXGKR:BLOCk:POSTraining:STATe? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:POSTraining:STATe? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:PRETraining:ENABled[?]

Command :PLUGin:LTXGKR:BLOCk:PRETraining:ENABled 'Identifier', <Bool>

:PLUGin:LTCGKR:BLOCk:PRETraining:ENABled 'Identifier', <Bool>

Query :PLUGin:LTXGKR:BLOCk:PRETraining:ENABled? 'Identifier'

:PLUGin:LTCGKR:BLOCk:PRETraining:ENABled? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<Bool>: Enter 1/ON or 0/OFF to enable or disable the pre-training state.

Return Value 1 | 0

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it enables or disables the Pre-Training state of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:BLOCk:PRETraining:ENABled '10GBASE-KR TX EQ Training 1', 1

:PLUGin:LTCGKR:BLOCk:PRETraining:ENABled '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', 1

:PLUGin:LTXGKR:BLOCk:PRETraining:ENABled? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:PRETraining:ENABled? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:PRETraining:SEQuence:TX:VALue[?]

Command :PLUGin:LTXGKR:BLOCk:PRETraining:SEQuence:TX:VALue 'Identifier', <sequence setting>

:PLUGin:LTCGKR:BLOCk:PRETraining:SEQuence:TX:VALue 'Identifier', <sequence setting>

Query :PLUGin:LTXGKR:BLOCk:PRETraining:SEQuence:TX:VALue? Identifier'

:PLUGin:LTCGKR:BLOCk:PRETraining:SEQuence:TX:VALue? Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<sequence setting>: Specify sequence setting

Return Value Sequence settings.

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it sets the sequence settings for the Pre-Training of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:BLOCk:PRETraining:SEQuence:TX:VALue '10GBASE-KR TX EQ Training 1', '<?xml version="1.0" encoding="utf-16"?><sequenceDefinition xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.keysight.com/schemas/M8000/DataSequence"><sequence><loop><block length="128"><prbs polynomial="2^7-1" /></block></loop></sequence></sequenceDefinition>'

:PLUGin:LTCGKR:BLOCk:PRETraining:SEQuence:TX:VALue '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', '<?xml version="1.0" encoding="utf-16"?><sequenceDefinition xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.keysight.com/schemas/M8000/DataSequence"><sequence><loop><block length="128"><prbs polynomial="2^7-1" /></block></loop></sequence></sequenceDefinition>'

:PLUGin:LTXGKR:BLOCk:POSTraining:SEQuence:TX:VALue?'10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:POSTraining:SEQuence:TX:VALue? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:PRETraining:STATe?

Query **:PLUGin:LTXGKR:BLOCk:PRETraining:STATe? 'Identifier'**

:PLUGin:LTCGKR:BLOCk:PRETraining:STATe? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value Not Yet Run | Not Completed | Completed

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it returns the state of Pre-Training of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:BLOCk:PRETraining:STATe? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:PRETraining:STATe? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:SEQuence:BREak

Command :PLUGin:LTXGKR:BLOCk:SEQuence:BREak 'Identifier'

:PLUGin:LTCGKR:BLOCk:SEQuence:BREak 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it breaks out all the loops which are defined as breakable and currently running of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:BLOCk:SEQuence:BREak '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:SEQuence:BREak '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

NOTE Presently, there is no way to limit the break functionality to have its effect only on the selected generator.

:PLUGin:[...]:BLOCk:TEST:ENABled[?]

Command :PLUGin:LTXGKR:BLOCk:TEST:ENABled 'Identifier', <Bool>

:PLUGin:LTCGKR:BLOCk:TEST:ENABled 'Identifier', <Bool>

Query :PLUGin:LTXGKR:BLOCk:TEST:ENABled? 'Identifier'

:PLUGin:LTCGKR:BLOCk:TEST:ENABled? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<Bool>: Enter 1/ON or 0 /OFF to enable or disable the device test state.

Return Value 1 | 0

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it enables or disables the device test state of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:BLOCk:TEST:ENABled '10GBASE-KR TX EQ Training 1', ON

:PLUGin:LTCGKR:BLOCk:TEST:ENABled '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', ON

:PLUGin:LTXGKR:BLOCk:TEST:ENABled? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:TEST:ENABled? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:TEST:SEQuence:RX:VALue[?]

Command :PLUGin:LTXGKR:BLOCk:TEST:SEQuence:RX:VALue 'Identifier', <sequence setting>

:PLUGin:LTCGKR:BLOCk:TEST:SEQuence:RX:VALue 'Identifier', <sequence setting>

Query :PLUGin:LTXGKR:BLOCk:TEST:SEQuence:RX:VALue? 'Identifier'

:PLUGin:LTCGKR:BLOCk:TEST:SEQuence:RX:VALue? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<sequence setting>: Specify sequence setting

Return Value Sequence settings.

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it sets the sequence settings to the receiver of device of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:BLOCk:TEST:SEQuence:RX:VALue '10GBase-KR TX EQ Training 1', '<?xml version="1.0" encoding="utf-16"?><sequenceDefinition xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.keysight.com/schemas/M8000/DataSequence"><sequence><loop><block length="128"><prbs polynomial="2^7-1" /></block></loop></sequence></sequenceDefinition>'

:PLUGin:LTCGKR:BLOCk:TEST:SEQuence:RX:VALue '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', '<?xml version="1.0" encoding="utf-16"?><sequenceDefinition xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.keysight.com/schemas/M8000/DataSequence"><sequence><loop><block length="128"><prbs polynomial="2^7-1" /></block></loop></sequence></sequenceDefinition>'

:PLUGin:LTXGKR:BLOCk:TEST:SEQuence:RX:VALue? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:TEST:SEQuence:RX:VALue? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:TEST:SEQuence:TX:VALue[?]

Command :PLUGin:LTXGKR:BLOCk:TEST:SEQuence:TX:VALue 'Identifier', <sequence setting>

:PLUGin:LTCGKR:BLOCk:TEST:SEQuence:TX:VALue 'Identifier', <sequence setting>

Query :PLUGin:LTXGKR:BLOCk:TEST:SEQuence:TX:VALue? 'Identifier'

PLUGin:LTCGKR:BLOCk:TEST:SEQuence:TX:VALue? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<sequence setting>: Specify sequence setting

Return Value Sequence settings.

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it sets the sequence settings to the transmitter of device of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:BLOCk:TEST:SEQuence:TX:VALue "10GBase-KR TX EQ Training 1", '<?xml version="1.0" encoding="utf-16"?><sequenceDefinition xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.keysight.com/schemas/M8000/DataSequence"><sequence><loop><block length="128"><prbs polynomial="2^7-1" /></block></loop></sequence></sequenceDefinition>'

:PLUGin:LTCGKR:BLOCk:TEST:SEQuence:TX:VALue '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', '<?xml version="1.0" encoding="utf-16"?><sequenceDefinition xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.keysight.com/schemas/M8000/DataSequence"><sequence><loop><block length="128"><prbs polynomial="2^7-1" /></block></loop></sequence></sequenceDefinition>'

:PLUGin:LTXGKR:BLOCk:TEST:SEQuence:TX:VALue? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:TEST:SEQuence:TX:VALue? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:TEST:STATe?

Query :PLUGin:LTXGKR:BLOCk:TEST:STATe? 'Identifier'

:PLUGin:LTCGKR:BLOCk:TEST:STATe? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value Not Yet Run | Not Completed | Completed

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns the state of post training of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:BLOCk:TEST:STATe? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:TEST:STATe? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:TXEQ:ENABled[?]

Command :PLUGin:LTXGKR:BLOCk:TXEQ:ENABled 'Identifier', <Bool>

:PLUGin:LTCGKR:BLOCk:TXEQ:ENABled 'Identifier', <Bool>

Query :PLUGin:LTXGKR:BLOCk:TXEQ:ENABled? 'Identifier'

:PLUGin:LTCGKR:BLOCk:TXEQ:ENABled? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<Bool>: Enter 1/ON or 0/OFF to enable or disable the TX EQ Training state.

Return Value 1 | 0

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it enables or disables the TX EQ state of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:BLOCk:TXEQ:ENABled '10GBASE-KR TX EQ Training 1', OFF

:PLUGin:LTCGKR:BLOCk:TXEQ:ENABled '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', OFF

:PLUGin:LTXGKR:BLOCk:TXEQ:ENABled? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:TXEQ:ENABled? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:BLOCk:TXEQ:STATe?

Query :PLUGin:LTXGKR:BLOCk:TXEQ:STATe? 'Identifier'

:PLUGin:LTCGKR:BLOCk:TXEQ:STATe? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value Not Yet Run | Not Completed | Completed

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns current state of TX EQ training of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:BLOCk:TXEQ:STATe? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:BLOCk:TXEQ:STATe? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:INSTrument:DINput[?]

Command :PLUGin:LTXGKR:INSTrument:DINput 'Identifier', <Local receiver>

:PLUGin:LTCGKR:INSTrument:DINput 'Identifier', <Local receiver>

Query :PLUGin:LTXGKR:INSTrument:DINput? 'Identifier'

:PLUGin:LTCGKR:INSTrument:DINput 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<Local receiver>: Enter 'M1.DataIn1' | 'M1.DataIn2' | 'M2.DataIn1' | 'M2.DataIn2'.

Return Value 'M1.DataIn1' | 'M1.DataIn2' | 'M2.DataIn1' | 'M2.DataIn2'

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it defines the local receiver which is used to capture frames from the transmitter of the DUT of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:INSTrument:DINput '10GBASE-KR TX EQ Training 1', 'M1.DataIn2'

:PLUGin:LTCGKR:INSTrument:DINput '25GBASE-KR and 100GBASE-KR4', 'M1.DataIn2'

:PLUGin:LTXGKR:INSTrument:DINput? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:INSTrument:DINput? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:INSTrument:DINput:ALIGnment

Command :PLUGin:LTXGKR:INSTrument:DINput:ALIGnment 'Identifier'

:PLUGin:LTCGKR:INSTrument:DINput:ALIGnment 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it aligns the selected input to the received signal of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:INSTrument:DINput:ALIGnment '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:INSTrument:DINput:ALIGnment '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:INSTrument:DOUTput[?]

Command :PLUGin:LTXGKR:INSTrument:DOUTput 'Identifier', <Local transmitter>

:PLUGin:LTCGKR:INSTrument:DOUTput 'Identifier', <Local transmitter>

Query :PLUGin:LTXGKR:INSTrument:DOUTput? 'Identifier'

:PLUGin:LTCGKR:INSTrument:DOUTput? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<Local transmitter>: Enter 'M*.DataOut*'.

Return Value “M*.DataOut*”

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it defines the local transmitter which is used to communicate with the DUT’s receiver of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:INSTrument:DOUTput '10GBASE-KR TX EQ Training 1', 'M1.DataOut1'

:PLUGin:LTCGKR:INSTrument:DOUTput '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', 'M1.DataOut1'

:PLUGin:LTXGKR:INSTrument:DOUTput? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:INSTrument:DOUTput? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:DUTState[?]

Command :PLUGin:LTXGKR:LTraining:DUTState 'Identifier', <PRESet | INITialize>

:PLUGin:LTCGKR:LTraining:DUTState 'Identifier', <PRESet | INITialize>

Query :PLUGin:LTXGKR:LTraining:DUTState? 'Identifier'

:PLUGin:LTCGKR:LTraining:DUTState? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value PRES | INIT

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it defines which state is requested from the DUT for its transmitter of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:LTraining:DUTState '10GBASE-KR TX EQ Training 1', INIT

:PLUGin:LTXGKR:LTraining:DUTState '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:LTXGKR:LTraining:DUTState? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTXGKR:LTraining:DUTState '125GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:RESult?

Query :PLUGin:LTXGKR:LTraining:RESult? 'Identifier'

:PLUGin:LTCGKR:LTraining:RESult? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value TX EQ Training has two states, so two types of return values are:

- "N/A": for the state ‘Not yet executed’.
- Rpre = 1.29, Rpst = 2.58, V2 = 252 mV c(+1) = -199 mV, c(0) = 488 mV, c(-1) = -36 mV: for the state ‘Executed’

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns the result of TX Equalization Training of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:LTraining:RESult? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:LTraining:RESult? '125GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:STATe:CMAIn?

Query :PLUGin:LTXGKR:LTraining:STATe:CMAIn? 'Identifier'

:PLUGin:LTCGKR:LTraining:STATe:CMAIn? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value It returns the value of c(0) that is currently assigned for the instrument's DATA OUT as requested from DUT.

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it returns the current differential main cursor value used by the instrument. of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:LTraining:STATe:CMAIn? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:LTraining:STATe:CMAIn? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:STATe:CMinus?

Query :PLUGin:LTXGKR:LTraining:STATe:CMinus? 'Identifier'

:PLUGin:LTCGKR:LTraining:STATe:CMinus? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value It returns the value of c(-1) that is currently assigned for the instrument's DATA OUT as requested from DUT.

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns the current differential pre-cursor (pre-shoot) value used by the instrument of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:LTraining:STATe:CMinus? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:LTraining:STATe:CMinus? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:STATe:CPlus?

Query :PLUGin:LTXGKR:LTraining:STATe:CPlus? 'Identifier'

:PLUGin:LTCGKR:LTraining:STATe:CPlus? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Range It returns the value of c(+1) that is currently assigned for the instrument's DATA OUT as requested from DUT.

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns the current differential steady state voltage setting used by the instrument of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:LTraining:STATe:CPlus? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:LTraining:STATe:CPlus? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:STATe:IREQuest?

Query **:PLUGin:LTXGKR:LTraining:STATe:IREQuest? 'Identifier'**

:PLUGin:LTCGKR:LTraining:STATe:IREQuest? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value Initial Hold State | Requested Update | Acknowledged Update | Idle | Rx Training Finished | Timeout

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it returns the current state of the request blocks inside the training frame as seen from the plug-in of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:LTraining:STATe:IREQuest? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTXGKR:LTraining:STATe:IREQuest? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:STATe:IRESponse?

Query **:PLUGin:LTXGKR:LTraining:STATe:IRESponse? 'Identifier'**

:PLUGin:LTCGKR:LTraining:STATe:IRESponse? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value Initial Hold State| Requested Update |Acknowledged Update | Idle | Rx Training Finished | Timeout

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns the current state of the response blocks inside the training frame as seen from the plug-in of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:LTraining:STATe:IRESponse? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:LTraining:STATe:IRESponse? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:STATe:RPRE?

Query :PLUGin:LTXGKR:LTraining:STATe:RPRE? 'Identifier'

:PLUGin:LTCGKR:LTraining:STATe:RPRE? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it returns the value of current pre-cursor equalization ratio used by the instrument of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:LTraining:STATe:RPRE? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:LTraining:STATe:RPRE? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:STATe:RPST?

Query :PLUGin:LTXGKR:LTraining:STATe:RPST? 'Identifier'

PLUGin:LTCGKR:LTraining:STATe:RPST? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it returns the value of current post-cursor equalization ratio used by the instrument of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:LTraining:STATe:RPST? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:LTraining:STATe:RPST? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:STATe:VMAIn?

Query :PLUGin:LTXGKR:LTraining:STATe:VMAIn? 'Identifier'

:PLUGin:LTCGKR:LTraining:STATe:VMAIn? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it returns the value of current differential steady state voltage setting used by the instrument of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:LTraining:STATe:VMAIn?'10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:LTraining:STATe:VMAIn? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:LTraining:TIMEout[?]

Command **:PLUGin:LTXGKR:LTraining:TIMEout** 'Identifier', <NR1>

:PLUGin:LTCGKR:LTraining:TIMEout 'Identifier', <NR1>

Query **:PLUGin:LTXGKR:LTraining:TIMEout?** 'Identifier'

:PLUGin:LTCGKR:LTraining:TIMEout? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

<NR1>: Enter the time out in seconds.

Return Range 1 second to 3600 seconds

Description Based on the variable used (LTCGKR or LTXGKR) in the command, it defines the maximum time which may be spent inside any of the TX Equalization Training States, before training is aborted of the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTXGKR:LTraining:TIMEout '10GBASE-KR TX EQ Training 1', 80

:PLUGin:LTCGKR:LTraining:TIMEout '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', 80

:PLUGin:LTXGKR:LTraining:TIMEout? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:LTraining:TIMEout? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

:PLUGin:[...]:TEXEcution:STATe?

Query **:PLUGin:LTXGKR:TEXEcution:STATe?** 'Identifier'

:PLUGin:LTCGKR:TEXEcution:STATe? 'Identifier'

Input Parameters For LTXGKR 'Identifier' is '10GBASE-KR TX EQ Training 1'

For LTCGKR 'Identifier' is '25GBASE-KR and 100GBASE-KR4'

Return Value Idle | TX EQ Training

Description Based on the variable used (LTCGKR or LTXGKR) in the query, it returns current training state of the TX EQ Training plug-in.

Example :PLUGin:LTXGKR:TEXEcution:STATe? '10GBASE-KR TX EQ Training 1'

:PLUGin:LTCGKR:TEXEcution:STATe? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

SCPIs for 25GBASE-KR and 100GBASE-KR4

:PLUGin:LTCGKR:LTraining:PRBS[?]

Command :PLUGin:LTCGKR:LTraining:PRBS 'Identifier', <Boolean>

Query :PLUGin:LTCGKR:LTraining:PRBS? 'Identifier'

Input Parameters 'Identifier': '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

<Boolean>: LAN0|LAN1|LAN2|LAN3

Description This command sets the lane number with specific PBRS Type (e.g. Lane 0: 1+x^5+x^6+x^10+x^11) to the TX EQ Training plug-in.

This query returns the present setting.

Example :PLUGin:LTCGKR:LTraining:PRBS '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1', LAN2

:PLUGin:LTCGKR:LTraining:PRBS? '25GBASE-KR and 100GBASE-KR4 TX EQ Training 1'

Executing a SCPI through M8070B SCPI Editor

This section describes how a SCPI command can be executed through M8070B SCPI Editor.

Following are the required steps:

1 From the M8070B user interface menu, click *Utilities* to view the list.

2 Select the SCPI *Editor*.

The *SCPI Editor* user interface appears as shown in the following figure:

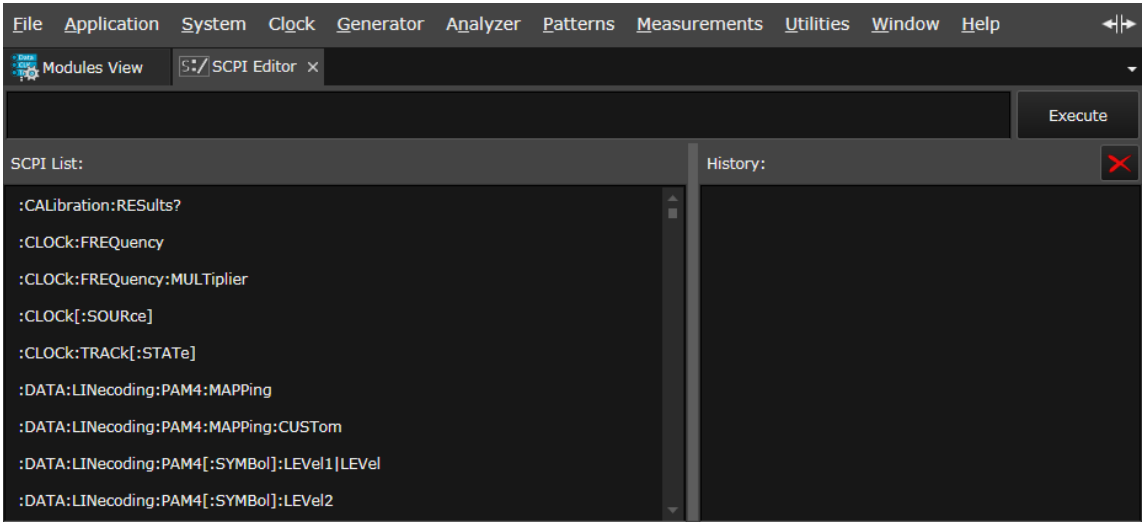


Figure 57 SCPI Editor

3 Select the SCPI from the given list. You may also type the SCPI in the provided text box to expedite the command search. Just above the *SCPI List:,* space is provided to write the SCPIs command.

4 Use the proper SCPI command syntax along with the command separators semicolon (“;”).

5 The following example shows how to execute a SCPI command to enable Pre-Training Block:

- Write the following syntax to open a plug-in instance-
:PLUGin:LTXGKR:NEW '10GBASE-KR TX EQ Training 1'.
A new plug-in instance named “10GBASE-KR TX EQ Training 1” opens in a new tab.

- Write the command you want to run. E.g. :PLUGin:LTXGKR:BLOCk:PRETraining:ENABled '10GBASE-KR TX EQ Training 1', 1

6 Click Execute. The output of the SCPI command is displayed in the History pane as shown in the following figure:

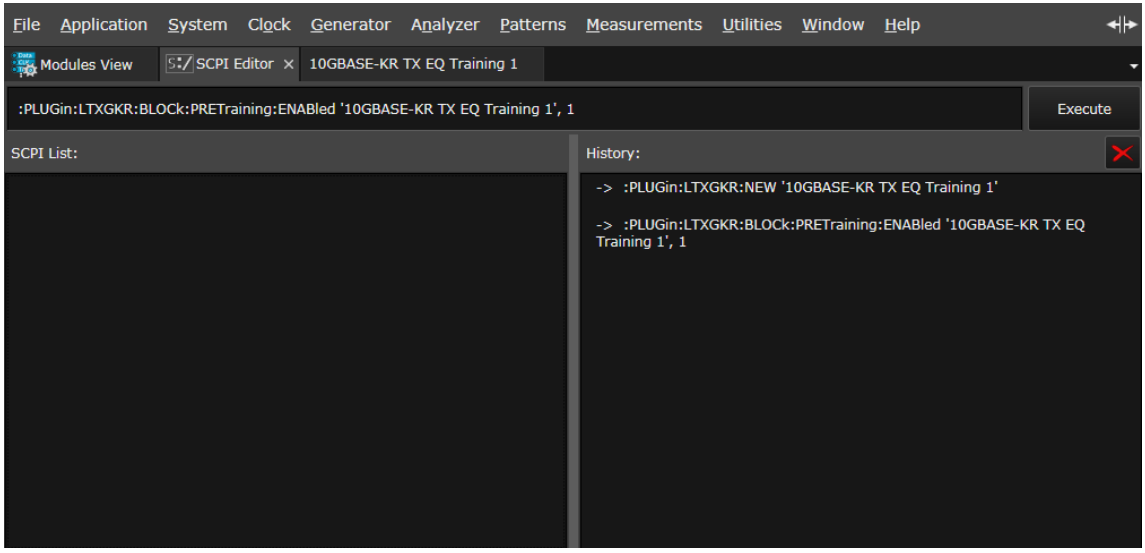


Figure 58 Executing SCPI Command

- Click the  Clear History icon to clear the contents of History pane.

Edition 3.0, March 2019

M8070-91020

www.keysight.com

KEYSIGHT
TECHNOLOGIES