



**High Speed Serial
Plug-in Application
Programmer Manual**



077-1246-00



**High Speed Serial
Plug-in Application
Programmer Manual**

Copyright © Tektronix. All rights reserved. Licensed software products are owned by Tektronix or its subsidiaries or suppliers, and are protected by national copyright laws and international treaty provisions.

Tektronix products are covered by U.S. and foreign patents, issued and pending. Information in this publication supersedes that in all previously published material. Specifications and price change privileges reserved.

TEKTRONIX and TEK are registered trademarks of Tektronix, Inc.

SourceXpress[®] is a registered trademark of Tektronix, Inc.

Contacting Tektronix

Tektronix, Inc.
14150 SW Karl Braun Drive
P.O. Box 500
Beaverton, OR 97077
USA

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

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

Table of Contents

Getting Started

Introduction	1-1
Documentation	1-2

Syntax and Commands

Command Syntax	2-1
Command Groups	2-3
Batch Compile commands	2-3
Channel commands	2-4
Compile commands	2-7
Control commands	2-8
Pattern Basic Settings commands	2-8
Pattern Duty Cycle Distortion commands	2-9
Pattern Encoding/Modulation commands	2-9
Pattern Markers commands	2-9
Pattern Scrambling commands	2-10
Pattern Step Response commands	2-10
Transmitter Noise commands	2-11
Transmitter Periodic Jitter commands	2-11
Transmitter Pre/De-emphasis commands	2-11
Transmitter Random Jitter commands	2-12
Transmitter Spread Spectrum Clocking commands	2-13
Command Descriptions	2-15

Getting Started

Introduction

This programmer manual provides information on how to use commands for remotely controlling the High Speed Serial plug-in application.

The High Speed Serial plug-in can be installed in either the SourceXpress software application or an AWG70000A series arbitrary waveform generator.

Communication with the plug-in is through the SourceXpress programmatic interface or the AWG70000A series instrument's programmatic interface. Using a single VISA or raw socket session, it is possible to communicate with both SourceXpress and AWG70000A series instruments.

For information on the Remote Control, GPIB Parameters, LAN Parameters, Connecting to the Instrument using GPIB, and Setting up GPIB Communication, refer to the *AWG70000A series Arbitrary Waveform Generators Programmer Manual*.

Documentation

In addition to this High Speed Serial Programmer Guide, the following documentation is included with this application:

- High Speed Serial Help. The help provides in-depth operation and user interface help.
- High Speed Serial User Manual (PDF). This is adapted from the High Speed Serial help system.

Syntax and Commands

Command Syntax

For information on the Syntax Overview, Command and Query Structure, Clearing the Instrument, Command Entry, Parameter Types, SCPI Commands and Queries, refer to the *AWG70000A Series Arbitrary Waveform Generators Programmer Manual*.

Command Groups

Batch Compile commands

Table 2-1: Batch Compile commands and their descriptions

Command	Description
HSSerial:BCOMpile:PJTter:DEViation:MDEViation	Sets or returns the percent of deviation when the Deviation for the Integer Cycles is set to Manual in Batch Compile.
HSSerial:BCOMpile:PJTter:DEViation:TYPE	Sets or returns the Periodic Jitter deviation type selected for Batch Compile.
HSSerial:BCOMpile:PJTter:ICENable	Sets or returns the Integer Cycles for Periodic Jitter state (enabled or disabled) in Batch Compile.
HSSerial:BCOMpile:PJ[n]:ENABLE	Sets or returns the specified Periodic Jitter state (enabled or disabled) in Batch Compile.
HSSerial:BCOMpile:PJ[n]:FREQUENCY:HIGH	Sets or returns the High Frequency value of the specified Periodic Jitter in Batch Compile.
HSSerial:BCOMpile:PJ[n]:FREQUENCY:INCRement	Sets or returns the Increment Frequency value of the specified Periodic Jitter in Batch Compile.
HSSerial:BCOMpile:PJ[n]:FREQUENCY:LOW	Sets or returns the Low Frequency value of the specified Periodic Jitter in batch compile.
HSSerial:BCOMpile:PJ[n]:MAGNitude:END	Sets or returns the End Magnitude value of the specified Periodic Jitter in Batch Compile.
HSSerial:BCOMpile:PJ[n]:MAGNitude:INCRement	Sets or returns the Magnitude Increment value of the specified Periodic Jitter in Batch Compile.
HSSerial:BCOMpile:PJ[n]:MAGNitude:START	Sets or returns the Start Magnitude value of the specified Periodic Jitter in Batch Compile.
HSSerial:BCOMpile:RJ1:ENABLE	Sets or returns the Random Jitter state (enabled or disabled) in Batch Compile.
HSSerial:BCOMpile:RJ1:FREQUENCY:HIGH	Sets or returns the High Frequency value of the Random Jitter in Batch Compile.
HSSerial:BCOMpile:RJ1:FREQUENCY:LOW	Sets or returns the Low Frequency value of the Random Jitter in batch compile.
HSSerial:BCOMpile:RJ1:MAGNitude:END	Sets or returns the End Magnitude value of the Random Jitter in batch compile.
HSSerial:BCOMpile:RJ1:MAGNitude:INCRement	Sets or returns the Magnitude Increment value of the Random Jitter in Batch Compile.
HSSerial:BCOMpile:RJ1:MAGNitude:START	Sets or returns the Start Magnitude value of the Random Jitter in Batch Compile.

Channel commands

Table 2-2: Channel commands and their descriptions

Command	Description
<code>HSSerial:CHANnel:ENABle</code>	Sets or returns the Channel emulation model state (enabled or disabled).
<code>HSSerial:CHANnel:ISI:BW</code>	Sets or returns the Intersymbol Interference (ISI) Normalized Channel Bandwidth.
<code>HSSerial:CHANnel:ISI:TYPE</code>	Sets or returns the Intersymbol Interference (ISI) type.
<code>HSSerial:CHANnel:ISI:VALue</code>	Sets or returns the Intersymbol Interference (ISI) magnitude.
<code>HSSerial:CHANnel:SPARameter:BANDwidth</code>	Sets or returns the S-Parameter bandwidth when setting manually.
<code>HSSerial:CHANnel:SPARameter:BANDwidth:AUTO</code>	Sets or returns the S-Parameter automatic bandwidth calculation setting.
<code>HSSerial:CHANnel:SPARameter:CASCading:AGGRessor2[:ENABle]</code>	Sets or returns whether the aggressor 2 signal type state (enabled or disabled) in Cascading mode.
<code>HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:AMPLitude</code>	Sets or returns the specified Aggressor's amplitude, in Cascading mode.
<code>HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:CTALK</code>	Sets or returns the specified Aggressor's crosstalk type, in Cascading mode.
<code>HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:DRATe</code>	This command sets or returns the specified Aggressor's data rate, in Cascading mode.
<code>HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal</code>	Sets or returns specified Aggressor's signal type, in Cascading mode.
<code>HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal:FILE</code>	This command sets or returns the filepath to the aggressor file for the specified Aggressor, in Cascading mode.
<code>HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal:PRBS</code>	Sets or returns the specified Aggressor's PRBS signal type, in Cascading mode.
<code>HSSerial:CHANnel:SPARameter:CASCading:DEEMbed</code>	Sets or returns whether the Cascading S-Parameters is to de-embed (invert) the S-Parameters, in Cascading mode.
<code>HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:DRX[n]</code>	Sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified receiver port number (Rx-Port) in Cascading mode and Differential Signalling Scheme (where applicable).
<code>HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:DTX[n]</code>	Sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified transmission port number (Tx-Port) in Cascading mode and Differential Signalling Scheme (where applicable).

Table 2-2: Channel commands and their descriptions (cont.)

Command	Description
HSSerial:CHANnel:SPARameter: CASCading:STAGe[m]:RX[n]	Sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified receiver port number (Rx-Port) in Cascading mode and Single-Ended Signalling Scheme (where applicable).
HSSerial:CHANnel:SPARameter: CASCading:STAGe[m]:TX[n]	Sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified transmission port number (Tx-Port) in Cascading mode and Single-Ended Signalling Scheme (where applicable).
HSSerial:CHANnel:SPARameter: CASCading:STAGe[m]:ENABLE	Sets or returns the state of the specified Cascaded S-Parameter stage (enabled or disabled).
HSSerial:CHANnel:SPARameter: CASCading:STAGe[m]:FILE	Sets or returns the Filepath for the specified S-Parameters Cascading Stage, in Cascading mode.
HSSerial:CHANnel:SPARameter: CASCading:STAGe[m]:SSCHeme	Sets or returns the S-Parameter Signalling Scheme, in Cascading mode. Signalling Scheme is only available when the Number of Ports is set to 4, 8, or 12.
HSSerial:CHANnel:SPARameter: CASCading:SType	Sets or returns S-Parameter signal type (victim or aggressor), in Cascading mode. The number of ports must be either 8 or 12.
HSSerial:CHANnel:SPARameter: CASCading:Type	Sets or returns the S-Parameter number of ports, in Cascading mode.
HSSerial:CHANnel:SPARameter:MODE	Sets or returns the S-Parameter mode (Cascading or Non-Cascading).
HSSerial:CHANnel:SPARameter: NCAScading:AGGRessor2[:ENABLE]	Sets or returns the aggressor 2 signal type state (enabled or disabled) in Non-Cascading mode. Aggressor2 signals are available when the number of ports is set to 12.
HSSerial:CHANnel:SPARameter: NCAScading:AGGRessor[n]:AMPLitude	Sets or returns the specified Aggressor's amplitude, in Non-Cascading mode.
HSSerial:CHANnel:SPARameter: NCAScading:AGGRessor[n]:CTAlk	Sets or returns the specified Aggressor's crosstalk type, in Non-Cascading mode.
HSSerial:CHANnel:SPARameter: NCAScading:AGGRessor[n]:DRATe	Sets or returns the specified Aggressor's data rate, in Non-Cascading mode.
HSSerial:CHANnel:SPARameter: NCAScading:AGGRessor[n]:SIGNal	Sets or returns specified Aggressor's signal type, in Non-Cascading mode.
HSSerial:CHANnel:SPARameter: NCAScading:AGGRessor[n]:SIGNal:FILE	Sets or returns the filepath to the aggressor file for the specified Aggressor, in Non-Cascading mode.

Table 2-2: Channel commands and their descriptions (cont.)

Command	Description
HSSerial:CHANnel:SPARameter: NCAScading:AGGRessor[n]:SIGNal:PRBS	Sets or returns the specified Aggressor's PRBS signal type, in Non-Cascading mode.
HSSerial:CHANnel:SPARameter: NCAScading:DEEMbed	Sets or returns whether the Non-Cascading S-Parameters is to de-embed (invert) the S-Parameters, in Non-Cascading mode.
HSSerial:CHANnel:SPARameter: NCAScading:DRX[n]	Sets or returns the S-Parameter port assignment of the channel's specified receiver port number (Rx-Port) in Non-Cascading mode and Differential Signalling Scheme (where applicable).
HSSerial:CHANnel:SPARameter: NCAScading:DTX[n]	Sets or returns the S-Parameter port assignment of the channel's specified transmission port number (Tx-Port) in Non-Cascading mode and Differential Signalling Scheme (where applicable).
HSSerial:CHANnel:SPARameter: NCAScading:FILE	Sets or returns the filepath and file name of the S-Parameter file, in Non-Cascading mode.
HSSerial:CHANnel:SPARameter: NCAScading:LAYout	Sets or returns the 4 port S-Parameter Matrix Configuration, in Non-Cascading mode.
HSSerial:CHANnel:SPARameter: NCAScading:RX[n]	Sets or returns the S-Parameter port assignment of the channel's specified receiver port number (Rx-Port) in Non-Cascading mode and Single-Ended Signalling Scheme (where applicable).
HSSerial:CHANnel:SPARameter: NCAScading:SSCHeme	Sets or returns the S-Parameter Signalling Scheme, in Non-Cascading mode. Signalling Scheme is only available when the Number of Ports is set to 4, 8, or 12.
HSSerial:CHANnel:SPARameter: NCAScading:STYPe	Sets or returns S-Parameter signal type (victim or aggressor), in Non-Cascading mode. The number of ports must be either 8 or 12.
HSSerial:CHANnel:SPARameter: NCAScading:TX[n]	Sets or returns the S-Parameter port assignment of the channel's specified transmission port number (Tx-Port) in Non-Cascading mode and Single-Ended Signalling Scheme (where applicable).
HSSerial:CHANnel:SPARameter: NCAScading:TYPE	Sets or returns the S-Parameter number of ports, in Non-Cascading mode.
HSSerial:CHANnel:TYPE	Sets or returns the Channel emulation type.

Compile commands

Table 2-3: Compile commands and their descriptions

Command	Description
HSSerial:COMPILE	Compiles and generates a waveform using the High Speed Serial plug-in compile settings.
HSSerial:COMPILE:ATSequence:ENABLE	Sets or returns the Add to sequence compile state (enabled or disabled) for the Compile Settings.
HSSerial:COMPILE:ATSequence:SEQUENCE	Sets or returns the sequence name in the sequence list when the Add to an existing sequence is enabled for the Compile Settings.
HSSerial:COMPILE:ATSequence:TRACK	Sets or returns the track name when the Add to an existing sequence is enabled for the Compile Settings.
HSSerial:COMPILE:CANCEL	Cancels a compilation currently in progress.
HSSerial:COMPILE:CORREction:APPLY	Enables or disables applying a corrections file during compile.
HSSerial:COMPILE:CORREction:PATH	Sets or returns the path of the corrections file to use during compile.
HSSerial:COMPILE:NAME	Sets or returns the name of the compiled waveform.
HSSerial:COMPILE:OPTions	Sets or returns the “Compile and assign” setting for the Compile Settings.
HSSerial:COMPILE:OVERwrite	Enables or disables overwriting an existing waveform name.
HSSerial:COMPILE:PLAY	Sets or returns the Play after assign setting for the Compile Settings.
HSSerial:COMPILE:REPeat	Sets or returns the Manual Repeat Count setting for the Compile Settings.
HSSerial:COMPILE:REPeat:AUTO	Sets or returns the Auto Repeat Count setting for the Compile Settings.
HSSerial:COMPILE:REPeat:MTYPE	Sets or returns the Manual Repeat Count type for the Compile Settings.
HSSerial:COMPILE:SPUI	Sets or returns the value of the Samples/Unit Interval for the Compile Settings.
HSSerial:COMPILE:SRATE	Sets or returns the sampling rate for the Compile Settings.
HSSerial:COMPILE:STYPE	Sets or returns the Sampling Rate type for the Compile Settings.

Control commands

Table 2-4: Control commands and their descriptions

Command	Description
HSSerial:RESet	Resets the High Speed Serial application plug-in to its default values.
WPLugin:ACTive	Sets or returns the active waveform plug-in. To use the High Speed Serial commands in this document, the active waveform plug-in must be set to High Speed Serial.

Pattern Basic Settings commands

Table 2-5: Pattern Basic Settings commands and their descriptions

Command	Description
HSSerial:BDATa	Sets or returns the Base Pattern Data type.
HSSerial:BDATa:INVert	Sets or returns the Invert Bits setting.
HSSerial:BDATa:PATtern	Sets or returns the user defined pattern value when the Base Data Pattern is set to Pattern.
HSSerial:BDATa:FILE	Sets or returns the path of the file to use when the Base Data Pattern is set to File.
HSSerial:BDATa:PATtern:TYPE	Sets or returns the user defined pattern type when the Base Data Pattern is set to Pattern.
HSSerial:BDATa:PRBS	Sets or returns the PRBS type when the Base Data Pattern is set to PRBS.
HSSerial:BDATa:PRBS:UDEFined:POLYnomial	Sets or returns the PRBS Polynomial Expression when the Base Pattern Data is set to User Defined PRBS.
HSSerial:BDATa:PRBS:UDEFined:SREGister	Sets or returns the Shift register initial value when the Base Pattern Data is set to User Defined PRBS.
HSSerial:DRATe	Sets or returns the Data Rate.
HSSerial:ISTate:OFFSet	Sets or returns the idle state amplitude offset.
HSSerial:ISTate:VALue	Sets or returns the idle state time.
HSSerial:AMPLitude:MAXimum	Sets or returns the maximum amplitude.
HSSerial:AMPLitude:MINimum	Sets or returns the minimum amplitude.

Pattern Duty Cycle Distortion commands

Table 2-6: Pattern Duty Cycle Distortion commands and their descriptions

Command	Description
HSSerial:DCD:ENABle	Sets or returns the Duty Cycle Distortion (DCD) state (enabled or disabled).
HSSerial:DCD:VALue	Sets or returns the Duty Cycle Distortion (DCD) value.

Pattern Encoding/Modulation commands

Table 2-7: Pattern Encoding/Modulation commands and their descriptions

Command	Description
HSSerial:ENCode:ENCo8b10b:DISParity	Sets or returns the 8B/10B encoding disparity setting.
HSSerial:ENCode:ENCo8b10b:ENABle	Sets or returns the 8B/10B encoding state (enabled or disabled).
HSSerial:ENCode:PAM:LEVelS	Sets or returns the Pulse-amplitude Modulation (PAM) encoding.
HSSerial:ENCode:PAM:NVALue	Sets or returns the PAM Normalized levels value for the specified row.
HSSerial:ENCode:PWM:ENABle	Sets or returns Pulse Width Modulation (PWM) state (enabled or disabled).
HSSerial:ENCode:PWM:TMINor	Sets or returns the T_Minor value of the PWM.
HSSerial:ENCode:SCHeme	Sets or returns the Signaling Scheme on the base data pattern.

Pattern Markers commands

Table 2-8: Pattern Markers commands and their descriptions

Command	Description
HSSerial:MARKer[n]:CLOCK:FREQuency	Sets or returns the user defined clock frequency for the specified marker.
HSSerial:MARKer[n]:CLOCK:TYPE	Sets or returns the marker clock type for the specified marker.
HSSerial:MARKer[n]:ENABle	Sets or returns the specified marker's state (enabled or disabled).
HSSerial:MARKer[n]:HLOW:TYPE	Sets or returns the High/Low type of the specified marker.

Table 2-8: Pattern Markers commands and their descriptions (cont.)

Command	Description
HSSerial:MARKer[n]:TRIGger:LENGth	Sets or returns the number of samples of the specified marker when the marker type is set to Trigger.
HSSerial:MARKer[n]:TRIGger:TYPE	Sets or returns the marker High/Low trigger type setting for the specified marker.
HSSerial:MARKer[n]:TYPE	Sets or returns the marker type for the specified marker.

Pattern Scrambling commands

Table 2-9: Pattern Scrambling commands and their descriptions

Command	Description
HSSerial:SCRamble:ENABLE	Sets or returns the Scrambling enable state (enabled or disabled).
HSSerial:SCRamble:POLYnomial	Sets or returns the Scrambling Polynomial value.
HSSerial:SCRamble:RINit	Sets or returns the Register Initial Value.
HSSerial:SCRamble:RTYPE	Sets or returns the Register Initial Value Type.
HSSerial:SCRamble:TYPE	Sets or returns the Scrambling Type.

Pattern Step Response commands

Table 2-10: Pattern Step Response commands and their descriptions

Command	Description
HSSerial:SRESponse:ENABLE	Sets or returns the Step Response state (enabled or disabled).
HSSerial:SRESponse:FTIME	Sets or returns the Step Response Fall Time value.
HSSerial:SResponse:FTYPE	Sets or returns the Step Response Fall Time Type.
HSSerial:SRESponse:RFTYPE	Sets or returns the Rise/Fall Time type (10/90 or 20/80).
HSSerial:SRESponse:RTIME	Sets or returns the Step Response Rise Time value.
HSSerial:SRESponse:RTYPE	Sets or returns the Step Response Rise Time Type.

Transmitter Noise commands

Table 2-11: Transmitter Noise commands and their descriptions

Command	Description
HSSerial:NOISe:BW	Sets or returns the Single Tone Noise frequency setting.
HSSerial:NOISe:ENABle	Sets or returns the Noise state (enabled or disabled).
HSSerial:NOISe:LOCation	Sets or returns the Transmitter Noise type.
HSSerial:NOISe:TYPE	Sets or returns the noise Frequency setting.
HSSerial:NOISe:VALue	Sets or returns the noise magnitude.

Transmitter Periodic Jitter commands

Table 2-12: Transmitter Periodic Jitter commands and their descriptions

Command	Description
HSSerial:PJITter:DEViation:MDEViation	Sets or returns the percent of deviation when the Deviation for the Integer Cycles is set to Manual.
HSSerial:PJITter:DEViation:TYPE	Sets or returns the Periodic Jitter Deviation type selected.
HSSerial:PJITter:ICENable	Sets or returns the Integer Cycles for Periodic Jitter state (enabled or disabled) when using the Transmitter settings.
HSSerial:PJ[n]:ENABle	Sets or returns the Transmitter Periodic Jitter state (enabled or disabled) for the specified Periodic Jitter.
HSSerial:PJ[n]:FREQuency	Sets or returns the frequency of the specified Transmitter Periodic Jitter.
HSSerial:PJ[n]:MAGNitude	Sets or returns the Magnitude (pk-pk) of the specified Transmitter Periodic Jitter.
HSSerial:PJ[n]:PHASe	Sets or returns the Phase of the specified Transmitter Periodic Jitter.

Transmitter Pre/De-emphasis commands

Table 2-13: Transmitter Pre/De-emphasis commands and their descriptions

Command	Description
HSSerial:PREemphasis:COPReemp	Sets or returns the Transmitter Pre/De-emphasis coefficients.
HSSerial:PREemphasis:DPRReemp	Deletes last active Pre/De-emphasis tap.

Table 2-13: Transmitter Pre/De-emphasis commands and their descriptions (cont.)

Command	Description
HSSerial:PREemphasis:ENABLE	Sets or returns the Transmitter Pre/De-Emphasis state (enabled or disabled).
HSSerial:PREemphasis:TPReemph	The set form of this command adds a new Pre/De-Emphasis tap. The query form of this command returns the number of active Pre/De-Emphasis taps.
HSSerial:PREemphasis:TYPE	Sets or returns the Pre/De-Emphasis type.
HSSerial:PREemphasis:UNIT	Sets or returns the Pre/De-Emphasis units.
HSSerial:PREShoot:COPShoot	Sets or returns the Transmitter Preshoot coefficients.
HSSerial:PREShoot:ENABLE	Sets or returns the Transmitter Preshoot state (enabled or disabled).
HSSerial:PREShoot:TYPE	Sets or returns the Preshoot type.

Transmitter Random Jitter commands

Table 2-14: Transmitter Random Jitter commands and their descriptions

Command	Description
HSSerial:RJ1Cfactor:ENABLE	Sets or returns the Random Jitter RJ1 Crest Factor state (enabled or disabled).
HSSerial:RJ1Cfactor:TISigma	Sets or returns the RJ1 Crest Factor value.
HSSerial:RJ[n]:ENABLE	Sets or returns the Transmitter Random Jitter state (enabled or disabled) for the specified Random Jitter .
HSSerial:RJ[n]:FREQuency:END	Sets or returns the High Frequency value of the specified Transmitter Random Jitter.
HSSerial:RJ[n]:FREQuency:START	Sets or returns the Low Frequency value of the specified Transmitter Random Jitter.
HSSerial:RJ[n]:MAGNitude	Sets or returns the Magnitude of the specified Transmitter Random Jitter.
HSSerial:RSEed:ENABLE	Sets or returns the Random Jitter RJ1 Seed state (enabled or disabled).
HSSerial:RSEed:VALue	Sets or returns the RJ1 Seed value.

Transmitter Spread Spectrum Clocking commands

Table 2-15: Transmitter Spread Spectrum Clocking commands and their descriptions

Command	Description
HSSerial:SSC:DFDT	This command sets or returns the Error Option df/dt value of Spread Spectrum Clocking.
HSSerial:SSC:DFDT:DURation	Sets or returns the Error Option Minimum Duration value of Spread Spectrum Clocking.
HSSerial:SSC:DFDT:LOCation	Sets or returns the Error Option Location value of Spread Spectrum Clocking.
HSSerial:SSC:ENABLE	Sets or returns the Spread Spectrum Clocking state (enabled or disabled).
HSSerial:SSC:EOPTion:ENABLE	Sets or returns the Error Option state (enabled or disabled) in Spread Spectrum Clocking.
HSSerial:SSC:FREQuency:DEViation	Sets or returns the Frequency Deviation value of Spread Spectrum Clocking.
HSSerial:SSC:FREQuency:MODulation	Sets or returns the Frequency Modulation value of Spread Spectrum Clocking.
HSSerial:SSC:SHApe	Sets or returns the Shape profile of Spread Spectrum Clocking.
HSSerial:SSC:SPRead	Sets or returns the Spread of the Spread Spectrum Clocking profile.
HSSerial:SSC:USPRead:PERCentage	Sets or returns the Downward or Upward Percentage of the Unequal Spread of Spread Spectrum Clocking.

Command Descriptions

HSSerial:AMPLitude:MAXimum

This command sets or returns the maximum amplitude.

Group	Pattern Basic Settings
Syntax	<code>HSSerial:AMPLitude:MAXimum <max_amplitude></code> <code>HSSerial:AMPLitude:MAXimum?</code>
Arguments	<code><max_amplitude>::= <NR3> value.</code>
Returns	A single <code><NR3></code> value.
Examples	<code>HSSerial:AMPLITUDE:MAXIMUM 200E-3</code> sets the maximum amplitude to 200 mV. <code>HSSerial:AMPLITUDE:MAXIMUM?</code> might return <code>250.0000000000E-3</code> , indicating the amplitude is set to 250 mV.

HSSerial:AMPLitude:MINimum

This command sets or returns the minimum amplitude.

Group	Pattern Basic Settings
Syntax	<code>HSSerial:AMPLitude:MINimum <min_amplitude></code> <code>HSSerial:AMPLitude:MINimum?</code>
Arguments	<code><min_amplitude>::= <NR3> value.</code>
Returns	A single <code><NR3></code> value.
Examples	<code>HSSerial:AMPLITUDE:MINIMUM -200E-3</code> sets the minimum amplitude to -200 mV. <code>HSSerial:AMPLITUDE:MINIMUM?</code> might return <code>-250.0000000000E-3</code> , indicating the amplitude is set to -250 mV.

HSSerial:BCOMpile:PJITter:DEViation:MDEViation

This command sets or returns the percent of deviation when the Deviation for the Integer Cycles is set to Manual in Batch Compile.

Group	Batch Compile
Syntax	HSSerial:BCOMpile:PJITter:DEViation:MDEViation <percent> HSSerial:BCOMpile:PJITter:DEViation:MDEViation?
Arguments	<percent>::=<NR3> value. Range: 0 to 50.
Returns	A single <NR1> value.
Examples	HSSERIAL:BCOMPILE:PJITTER:DEVIATION:MDEVIATION 20 sets the deviation to 20% when the Integer Cycles Deviation type is set to Manual in Batch Compile. HSSERIAL:BCOMPILE:PJITTER:DEVIATION:MDEVIATION? might return 20, indicating the deviation is set to 20% when the Integer Cycles Deviation type is set to Manual in Batch Compile.

HSSerial:BCOMpile:PJITter:DEViation:TYPE

This command sets or returns the Periodic Jitter deviation type selected for Batch Compile.

Group	Batch Compile
Syntax	HSSerial:BCOMpile:PJITter:DEViation:TYPE {AUTO MANUal} HSSerial:BCOMpile:PJITter:DEViation:TYPE?
Arguments	AUTO – Auto deviation type. MANUal – Manual deviation type.
Returns	AUTO MANU

Examples `HSSERIAL:BCOMPILE:PJITTER:DEVIATION:TYPE AUTO` sets the deviation type to auto.

`HSSERIAL:BCOMPILE:PJITTER:DEVIATION:TYPE?` might return `MANUAL`, indicating the Periodic Jitter deviation type is manual.

HSSerial:BCOMpile:PJITter:ICENable

This command sets or returns the Integer Cycles for Periodic Jitter state (enabled or disabled) in Batch Compile.

Group Batch Compile

Syntax `HSSerial:BCOMpile:PJITter:ICENable {0|1|OFF|ON}`
`HSSerial:BCOMpile:PJITter:ICENable?`

Arguments OFF or 0 disables Integer Cycles. OFF or 0 is the default value.
 ON or 1 enables the Integer Cycles.

Returns A single <Boolean> value.

Examples `HSSERIAL:BCOMPILE:PJITTER:ICENABLE 0` disables integer cycles for Periodic Jitter when using Batch Compile.

`HSSERIAL:BCOMPILE:PJITTER:ICENABLE?` might return 1, indicating that the integer cycles is enabled for Periodic Jitter when using Batch Compile.

HSSerial:BCOMpile:PJ[n]:ENABLE

This command sets or returns the specified Periodic Jitter state (enabled or disabled) in Batch Compile .

Conditions PJ1 must be enabled in order to use PJ2.

Group Batch Compile

Syntax `HSSerial:BCOMpile:PJ[n]:ENABLE {0|1|OFF|ON}`
`HSSerial:BCOMpile:PJ[n]:ENABLE?`

Arguments	[n] ::= {1 2} ("n" determines PJ1 or PJ2. If omitted, interpreted as 1.) OFF or 0 disables the selected Periodic Jitter. OFF or 0 is the default value. ON or 1 enables the selected Periodic Jitter.
Returns	A single <Boolean> value.
Examples	HSSerial:BCOMPILE:PJ2:ENABLE 1 enables PJ2 in batch compile. HSSerial:BCOMPILE:PJ1:ENABLE? might return 0, indicating PJ1 in batch compile is not enabled.

HSSerial:BCOMpile:PJ[n]:FREQuency:HIGH

This command sets or returns the High Frequency value of the specified Periodic Jitter in Batch Compile.

Group	Batch Compile
Syntax	HSSerial:BCOMpile:PJ[n]:FREQuency:HIGH <value> HSSerial:BCOMpile:PJ[n]:FREQuency:HIGH?
Arguments	[n] ::= {1 2} ("n" determines PJ1 or PJ2. If omitted, interpreted as 1.) <value> ::= <NR3> value.
Returns	A single <NR3> value.
Examples	HSSerial:BCOMPILE:PJ2:FREQUENCY:HIGH 3E6 sets the PJ2 high frequency value to 3 MHz for batch compile. HSSerial:BCOMPILE:PJ1:FREQUENCY:HIGH? might return 10.0000000000E+3, indicating that the high frequency value of PJ1 for batch compile is set to 10 kHz.

HSSerial:BCOMpile:PJ[n]:FREQuency:INCRe ment

This command sets or returns the Increment Frequency value of the specified Periodic Jitter in Batch Compile.

Group	Batch Compile
--------------	---------------

Syntax	<code>HSSerial:BCOMpile:PJ[n]:FREQUENCY:INCRement <value></code> <code>HSSerial:BCOMpile:PJ[n]:FREQUENCY:INCRement?</code>
Arguments	<code>[n] ::= {1 2}</code> ("n" determines PJ1 or PJ2. If omitted, interpreted as 1.) <code><value> ::= <NR3> value.</code>
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:BCOMPILE:PJ2:FREQUENCY:INCREMENT 10E3</code> sets the PJ2 Increment frequency value to 10 kHz for batch compile. <code>HSSERIAL:BCOMPILE:PJ1:FREQUENCY:INCREMENT?</code> might return <code>10.000000000E+3</code> , indicating that the PJ1 Increment frequency value for batch compile is 10 kHz

HSSerial:BCOMpile:PJ[n]:FREQUENCY:LOW

This command sets or returns the Low Frequency value of the specified Periodic Jitter in Batch Compile.

Group	Batch Compile
Syntax	<code>HSSerial:BCOMpile:PJ[n]:FREQUENCY:LOW <value></code> <code>HSSerial:BCOMpile:PJ[n]:FREQUENCY:LOW?</code>
Arguments	<code>[n] ::= {1 2}</code> ("n" determines PJ1 or PJ2. If omitted, interpreted as 1.) <code><value> ::= <NR3> value.</code>
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:BCOMPILE:PJ2:FREQUENCY:LOW 3E6</code> sets the PJ2 low frequency value to 3 MHz for batch compile. <code>HSSERIAL:BCOMPILE:PJ1:FREQUENCY:LOW?</code> might return <code>10.000000000E+3</code> , indicating that the low frequency value of PJ1 for batch compile is set to 10 kHz.

HSSerial:BCOMpile:PJ[n]:MAGNitude:END

This command sets or returns the End Magnitude value of the specified Periodic Jitter in Batch Compile.

Group	Batch Compile
Syntax	HSSerial:BCOMpile:PJ[n]:MAGNitude:END <value> HSSerial:BCOMpile:PJ[n]:MAGNitude:END?
Arguments	[n] ::= {1 2} ("n" determines PJ1 or PJ2. If omitted, interpreted as 1.) <value> ::= <NR3> value.
Returns	A single <NR3> value.
Examples	HSSERIAL:BCOMPILE:PJ1:MAGNITUDE:END 10 sets the PJ1 End Magnitude value to 10 UI for batch compile. HSSERIAL:BCOMPILE:PJ1:MAGNITUDE:END? might return 20.0000000000E-3, indicating that the PJ1 End Magnitude value in batch compile is set to 0.02 UI.

HSSerial:BCOMpile:PJ[n]:MAGNitude:INCRement

This command sets or returns the Magnitude Increment value of the specified Periodic Jitter in Batch Compile.

Group	Batch Compile
Syntax	HSSerial:BCOMpile:PJ[n]:MAGNitude:INCRement <value> HSSerial:BCOMpile:PJ[n]:MAGNitude:INCRement?
Arguments	[n] ::= {1 2} ("n" determines PJ1 or PJ2. If omitted, interpreted as 1.) <value> ::= <NR3> value.
Returns	A single <NR3> value.
Examples	HSSERIAL:BCOMPILE:PJ2:MAGNITUDE:INCREMENT 0.01 sets the PJ2 Magnitude Increment value to 0.01 UI for batch compile.

HSSerial:BCOMPILE:PJ1:MAGNITUDE:INCREMENT? might return 10.000000000E-3, indicating the PJ1 Magnitude Increment value in batch compile is 0.01 UI.

HSSerial:BCOMpile:PJ[n]:MAGNitude:START

This command sets or returns the Start Magnitude value of the specified Periodic Jitter in Batch Compile.

Group	Batch Compile
Syntax	HSSerial:BCOMpile:PJ[n]:MAGNitude:START <value> HSSerial:BCOMpile:PJ[n]:MAGNitude:START?
Arguments	[n] ::= {1 2} ("n" determines PJ1 or PJ2. If omitted, interpreted as 1.) <value> ::= <NR3> value.
Returns	A single <NR3> value.
Examples	HSSerial:BCOMPILE:PJ1:MAGNITUDE:END 10 sets the PJ1 Start Magnitude value to 10 UI for batch compile. HSSerial:BCOMPILE:PJ1:MAGNITUDE:END? might return 20.000000000E-3, indicating that the PJ1 Start Magnitude value in batch compile is set to 0.02 UI.

HSSerial:BCOMpile:RJ1:ENABLE

This command sets or returns the Random Jitter state (enabled or disabled) in Batch Compile.

Group	Batch Compile
Syntax	HSSerial:BCOMpile:RJ1:ENABLE {0 1 OFF ON} HSSerial:BCOMpile:RJ1:ENABLE?
Arguments	OFF or 0 disables Random Jitter. OFF or 0 is the default value. ON or 1 enables Random Jitter.

Returns A single <Boolean> value.

Examples `HSSerial:BCOMPILE:RJ1:ENABLE 1` enables RJ1 in batch compile
`HSSerial:BCOMPILE:RJ1:ENABLE?` might return 0, indicating that RJ1 in batch compile is disabled.

HSSerial:BCOMpile:RJ1:FREQuency:HIGH

This command sets or returns the High Frequency value of the Random Jitter in Batch Compile.

Group Batch Compile

Syntax `HSSerial:BCOMpile:RJ1:FREQuency:HIGH <high_frequency>`
`HSSerial:BCOMpile:RJ1:FREQuency:HIGH?`

Arguments <high_frequency>::= <NR3> value.

Returns A single <NR3> value.

Examples `HSSerial:BCOMPILE:RJ1:FREQUENCY:HIGH 1E6` sets the RJ1 High Frequency value to 1 MHz for batch compile.
`HSSerial:BCOMPILE:RJ1:FREQUENCY:HIGH?` might return `250.0000000000E+3`, indicating that the RJ1 High Frequency for batch compile is set to 250 kHz.

HSSerial:BCOMpile:RJ1:FREQuency:LOW

This command sets or returns the Low Frequency value of the Random Jitter in Batch Compile.

Group Batch Compile

Syntax `HSSerial:BCOMpile:RJ1:FREQuency:LOW <low_frequency>`
`HSSerial:BCOMpile:RJ1:FREQuency:LOW?`

Arguments <low_frequency>::= <NR3> value.

Returns A single <NR3> value.

Examples `HSSerial:BCOMPILE:RJ1:FREQUENCY:LOW 3E6` sets the RJ1 Low Frequency value to 3 MHz for batch compile.

`HSSerial:BCOMPILE:RJ1:FREQUENCY:LOW?` might return `10.000000000E+3`, indicating that the RJ1 Low Frequency value for batch compile is set to 10 kHz.

HSSerial:BCOMpile:RJ1:MAGNitude:END

This command sets or returns the End Magnitude value of the Random Jitter in batch compile.

Group Batch Compile

Syntax `HSSerial:BCOMpile:RJ1:MAGNitude:END <magnitude>`
`HSSerial:BCOMpile:RJ1:MAGNitude:END?`

Arguments <magnitude>::= <NR3> value.

Returns A single <NR3> value.

Examples `HSSerial:BCOMPILE:RJ1:MAGNITUDE:END 30E-3` sets the RJ1 End Magnitude value to 0.03 UI for batch compile.

`HSSerial:BCOMPILE:RJ1:MAGNITUDE:END?` might return `20.000000000E-3`, indicating that the RJ1 End Magnitude value in batch compile is set to 0.02 UI.

HSSerial:BCOMpile:RJ1:MAGNitude:INCRement

This command sets or returns the Magnitude Increment value of the Random Jitter in batch compile.

Group Batch Compile

Syntax `HSSerial:BCOMpile:RJ1:MAGNitude:INCRement <increment>`
`HSSerial:BCOMpile:RJ1:MAGNitude:INCRement?`

Arguments	<code><increment>::= <NR3> value.</code>
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:BCOMPILE:RJ1:MAGNITUDE:INCREMENT 0.01</code> sets the RJ Magnitude Increment value to 0.01 UI for batch compile. <code>HSSERIAL:BCOMPILE:RJ1:MAGNITUDE:INCREMENT?</code> might return <code>10.0000000000E-3</code>, indicating the RJ Magnitude Increment value in batch compile is 0.01 UI.

HSSerial:BCOMpile:RJ1:MAGNitude:START

This command sets or returns the Start Magnitude value of the Random Jitter in batch compile.

Group	Batch Compile
Syntax	<code>HSSerial:BCOMpile:RJ1:MAGNitude:START <start_magnitude></code> <code>HSSerial:BCOMpile:RJ1:MAGNitude:START?</code>
Arguments	<code><start_magnitude>::= <NR3> value.</code>
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:BCOMPILE:RJ1:MAGNITUDE:START 10E-3</code> sets the RJ1 Start Magnitude value to 0.01 UI for batch compile. <code>HSSERIAL:BCOMPILE:RJ1:MAGNITUDE:START?</code> might return <code>20.0000000000E-3</code>, indicating that the RJ1 Start Magnitude value in batch compile is set to 0.02 UI.

HSSerial:BDATa

This command sets or returns the Base Pattern Data type.

Group	Pattern Basic Settings
Syntax	<code>HSSerial:BDATa {CLOCK AZERO AONE ISTate PRBS PATTERN FILE}</code> <code>HSSerial:BDATa?</code>

Related Commands	HSSerial:BDATa:PRBS HSSerial:BDATa:FILE HSSerial:BDATa:PATtern:TYPE
Arguments	CLOCK generates a waveform with a clock-like bit pattern of alternate 0s and 1s. AZERo generates a waveform with an amplitude corresponding to the maximum negative voltage level. AONE generates a waveform with an amplitude corresponding to maximum positive voltage level. ISTate sets the base data type to Idle State. During this state, the value is neither 0 or 1. It is a state during which no valid data is transferred between the transmitter and the receiver. SourceXpress supports Idle state waveforms with idle value and idle offset. PRBS sets the base data type to PRBS. Select a PRBS pattern. PATtern sets the base data type to a Binary, Hex, or Symbol pattern. Select a pattern type. FILE sets the base data type to use a supplied pattern file.
Returns	CLOC (Clock) AZER (All Zeros) AONE (All Ones) IST (Idle State) PRBS PATT (Pattern) FILE
Examples	HSSERIAL:BDATA IState set the Base Data Pattern to IdleState. HSSERIAL:BDATA? might return ALL, indicating the Base Data Pattern type is set to All Ones.

HSSerial:BDATa:INVert

This command sets or returns the Invert Bits setting.

Group Pattern Basic Settings

Syntax HSSerial:BDATa:INVert {0|1|OFF|ON}
HSSerial:BDATa:INVert?

Arguments	OFF or 0 disables Invert Bits. OFF or 0 is the default value. ON or 1 enables Invert Bits.
Returns	A single <Boolean> value.
Examples	HSSERIAL:BDATA:INVERT 1 enables the Invert Bits. HSSERIAL:BDATA:INVERT? might return 0, indicating that Invert Bits is disabled.

HSSerial:BDATa:PATtern

This command sets or returns the user defined pattern value when the Base Data Pattern is set to Pattern. The pattern must be valid for the selected Pattern type set using the command [HSSerial:BDATa:PATtern:TYPE](#).

Group	Pattern Basic Settings
Syntax	HSSerial:BDATa:PATtern <pattern> HSSerial:BDATa:PATtern?
Related Commands	HSSerial:BDATa:PATtern:TYPE
Arguments	<pattern>::=<string> defines the user pattern.
Returns	A single <pattern> string.
Examples	HSSERIAL:BDATA:PATTERN "FF" sets the pattern to FF for the Hexadecimal pattern. HSSERIAL:BDATA:PATTERN? might return "FF", indicating the pattern is set to FF for the Hexadecimal pattern.

HSSerial:BDATa:FILE

This command sets or returns the path of the file to use when the Base Data Pattern is set to File.

Group	Pattern Basic Settings
--------------	------------------------

Syntax HSSerial:BDATA:FILE <filepath>
HSSerial:BDATA:FILE?

Related Commands [HSSerial:BDATA](#)

Arguments <filepath>::=<string> defines the file path.

Returns A single <filepath> string.

Examples HSSERIAL:BDATA:FILE "C:\test.txt" sets the File Path to C:\test.txt.
HSSERIAL:BDATA:FILE? might return "C:\test.txt", indicating the set file path.

HSSerial:BDATA:PATtern:TYPE

This command sets or returns the user defined pattern type when the Base Data Pattern is set to Pattern.

Group Pattern Basic Settings

Syntax HSSerial:BDATA:PATtern:TYPE {BINary|HEX|SYMBol}
HSSerial:BDATA:PATtern:TYPE?

Related Commands [HSSerial:BDATA](#)

Arguments BINary sets the user pattern type to binary.
HEX sets the user pattern type to hexadecimal.
SYMBol sets the user pattern type to symbol.

Returns BIN (Binary)
HEX (Hexadecimal)
SYMB (Symbol)

Examples HSSERIAL:BDATA:PATTERN:TYPE HEX sets the Base Pattern Data type to Hexadecimal.
HSSERIAL:BDATA:PATTERN:TYPE? might return BIN, indicating that the Base Pattern Data type is set to Binary.

HSSerial:BDATa:PRBS

This command sets or returns the PRBS type when the Base Data Pattern is set to PRBS.

Group Pattern Basic Settings

Syntax HSSerial:BDATa:PRBS
{PRBS7|PRBS9|PRBS15|PRBS16|PRBS20|PRBS21|PRBS23|PRBS29|PRBS31|UDEF}
HSSerial:BDATa:PRBS?

Related Commands [HSSerial:BDATa](#)

Arguments PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31 sets the Base Data PRBS pattern to the selected PRBS type.

UDEF sets the Base Data PRBS pattern to User Defined.

Returns The PRBS type or UDEF.

Examples HSSerial:BDATa:PRBS PRBS15 sets the Base Data PRBS pattern to PRBS15.
HSSerial:BDATa:PRBS? might return PRBS7, indicating that the Base Data PRBS pattern is PRBS7.

HSSerial:BDATa:PRBS:UDEFined:POLYnomial

This command sets or returns the PRBS Polynomial Expression when the Base Pattern Data is set to User Defined PRBS.

Group Pattern Basic Settings

Syntax HSSerial:BDATa:PRBS:UDEFined:POLYnomial <polynomial>
HSSerial:BDATa:PRBS:UDEFined:POLYnomial?

Related Commands [HSSerial:BDATa](#)

Arguments <polynomial>::= <string> defines the PRBS polynomial.

Returns A single <polynomial> string.

Examples `HSSerial:BDATA:PRBS:UDEFINED:POLYNOMIAL "X7+X6+1"` sets the PRBS Polynomial to be X^7+X^6+1 .

`HSSerial:BDATA:PRBS:UDEFINED:POLYNOMIAL?` might return the PRBS Polynomial as $"X^9+X^5+1"$.

HSSerial:BDATa:PRBS:UDEFined:SREGister

This Command sets or returns the Shift register initial value when the Base Pattern Data is set to User Defined PRBS.

Group Pattern Basic Settings

Syntax `HSSerial:BDATa:PRBS:UDEFined:SREGister <register>`
`HSSerial:BDATa:PRBS:UDEFined:SREGister?`

Related Commands [HSSerial:BDATa](#)

Arguments <register>::= <string> defines the PRBS shift register initial value.

Returns A single <register> string.

Examples `HSSerial:BDATA:PRBS:UDEFINED:SREGISTER "0000001"` sets the PRBS Shift register initial value to be 0000001.

`HSSerial:BDATA:PRBS:UDEFINED:SREGISTER?` might return the PRBS ShiftRegister value as $"1111111"$.

HSSerial:CHANnel:ENABle

This command sets or returns the Channel emulation model state (enabled or disabled).

Group Channel

Syntax `HSSerial:CHANnel:ENABle {0|1|OFF|ON}`
`HSSerial:CHANnel:ENABle?`

Arguments	OFF or 0 disables Channel emulation. OFF or 0 is the default value. ON or 1 enables Channel emulation.
Returns	A single <Boolean> value.
Examples	HSSERIAL:CHANNEL:ENABLE 0 disables Channel emulation. HSSERIAL:CHANNEL:ENABLE? might return 1, indicating that Channel emulation is enabled.

HSSerial:CHANnel:ISI:BW

This command sets or returns the Intersymbol Interference (ISI) Normalized Channel Bandwidth.

Conditions	Requires an S-Parameters license.
Group	Channel
Syntax	HSSerial:CHANnel:ISI:BW <bandwidth>
Related Commands	HSSerial:CHANnel:ISI:TYPE
Arguments	<bandwidth>::=<NR3> value.
Returns	A single <NR3> value.
Examples	HSSERIAL:CHANNEL:ISI:BW 0.5 sets the ISI Normalized Channel Bandwidth to 0.5. HSSERIAL:CHANNEL:ISI:BW? might return 300.0000000000E-3, indicating that the ISI Normalized Channel Bandwidth is set to 0.3.

HSSerial:CHANnel:ISI:TYPE

This command sets or returns the Intersymbol Interference (ISI) type.

Conditions	Requires an S-Parameters license.
-------------------	-----------------------------------

Group	Channel
Syntax	<code>HSSerial:CHANNEL:ISI:TYPE {VALUE BW}</code>
Arguments	VALUE sets the ISI to use a value in Unit Intervals or seconds. BW sets the ISI to use a Normalized Channel Bandwidth.
Returns	VAL (Value) BW (Bandwidth)
Examples	<code>HSSERIAL:CHANNEL:ISI:TYPE BW</code> sets the ISI to use a Normalized Channel Bandwidth setting. <code>HSSERIAL:CHANNEL:ISI:TYPE?</code> might return VAL, indicating that the Intersymbol Interference is set to Intersymbol Interference Value.

HSSerial:CHANnel:ISI:VALue

This command sets or returns the Intersymbol Interference (ISI) magnitude.

Conditions	The default units is UI (Unit Intervals) and is typically how this control is defined. Optionally, the units can be set as seconds. When setting the value, you must send values appropriate for the user interface setting. Units is set directly from the user interface for this value. See the examples for using the set and query forms as either UI or seconds. Requires an S-Parameters license.
-------------------	---

Group	Channel
Syntax	<code>HSSerial:CHANNEL:ISI:VALue <magnitude></code>
Related Commands	HSSerial:CHANnel:ISI:TYPE
Arguments	<magnitude>::=<NR3> value. Range in UI: 0.01 to 0.5 Range in seconds: 10 ps to 500 ps

Returns A single <NR3> value.

Examples Unit Intervals

HSSERIAL:CHANNEL:ISI:VALUE 0.5 sets the ISI Value to 0.5 UI.

HSSERIAL:CHANNEL:ISI:VALUE? might return 25.0000000000E-3, indicating that the ISI Value is set to 0.025 UI.

Seconds

HSSERIAL:CHANNEL:ISI:VALUE 25E-12 sets the ISI Value to 25 ps.

HSSERIAL:CHANNEL:ISI:VALUE? might return 10.0000000000E-12, indicating that the ISI Value is set to 10 ps.

HSSerial:CHANnel:SPARAmeter:BANDwidth

This command sets or returns the S-Parameter bandwidth when setting manually.

Conditions Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARAmeter:BANDwidth {FULL|<bandwidth>}
HSSerial:CHANnel:SPARAmeter:BANDwidth?

Related Commands [HSSerial:CHANnel:SPARAmeter:BANDwidth:AUTO](#)

Arguments FULL – The bandwidth is set to ½ of the waveform’s sample rate (i.e. Nyquist Frequency).
<bandwidth>::= <NR3> value.
Range: 1 Hz to ½ of the maximum sample rate of the instrument.
If the set Bandwidth is greater than the Nyquist (Sample rate of the waveform/2), then the software limits the bandwidth to ½ of the waveform’s sample rate.

Returns FULL
A single <NR3> value.

Examples HSSERIAL:CHANNEL:SPARAMETER:BANDWIDTH 60E6 sets the S-Parameter Bandwidth Value to 60 MHz.

HSSERIAL:CHANNEL:SPARAMETER:BANDWIDTH? might return
1.000000000E+9, indicating the S-Parameter Bandwidth is set to 1 GHz.

HSSerial:CHANnel:SPARAmeter:BANDwidth:AUTO

This command sets or returns the S-Parameter automatic bandwidth calculation setting. The bandwidth is defined at the point where the signal rolls off to –60 dB. If this results in a bandwidth greater than the instrument supports, the bandwidth is set to ½ of the waveform's sample rate (i.e. Nyquist Frequency).

Conditions Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARAmeter:BANDwidth:AUTO {0|1|OFF|ON}
HSSerial:CHANnel:SPARAmeter:BANDwidth:AUTO?

Related Commands [HSSerial:CHANnel:SPARAmeter:BANDwidth](#)

Arguments ON or 1 enables automatic bandwidth calculation. ON or 1 is the default value.
OFF or 0 disables automatic bandwidth calculation and sets the Bandwidth setting to Manual, requiring a value.

Returns A single <Boolean> value.

Examples HSSERIAL:CHANNEL:SPARAMETER:BANDWIDTH:AUTO 0 disables the
S-Parameter automatic bandwidth calculation and sets it to use a manual value.
HSSERIAL:CHANNEL:SPARAMETER:BANDWIDTH:AUTO? might return 1,
indicating the S-Parameter automatic bandwidth calculation is enabled.

HSSerial:CHANnel:SPARAmeter:CASCading:AGGRessor2[:ENABLE]

This command sets or returns whether the aggressor 2 signal type state (enabled or disabled) in Cascading mode. Aggressor2 signals are available when the number of ports is set to 12.

Conditions S-Parameter Mode must be set to Cascading.

Number of Ports must be set to 12.

Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2[:ENABLE]
{0|1|ON|OFF}
HSSerial:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2[:ENABLE]?`

Related Commands [HSSerial:CHANNEL:SPARAMETER:MODE](#)

Arguments OFF or 0 disables the aggressor 2 signal type. OFF or 0 is the default value.
ON or 1 enables the aggressor 2 signal type.

Returns A single <Boolean> value.

Examples `HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2:ENABLE ON`
enables the aggressor 2 signal type, in Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2:ENABLE?`
might return 0, indicating that the aggressor 2 signal type is disabled, in Cascading mode.

HSSerial:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR[n]:AMPLITUDE

This command sets or returns the specified Aggressor's amplitude, in Cascading mode.

Conditions S-Parameter Mode must be set to Cascading.

Number of ports must be either 8 or 12.

Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR[n]:AMPLITUDE
<amplitude>
HSSerial:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR[n]:
AMPLITUDE?`

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments `[n] ::= {1|2}` (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

`<amplitude> ::= <NRf>`

Returns A single `<NR3>` value.

Examples `HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR1:AMPLITUDE 200E-3` sets the first Aggressor's amplitude to 200 mV, in Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2:AMPLITUDE?` might return `100.0000000000E-3`, indicating that the 2nd Aggressor's amplitude is set to 100 mV, in Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:CTALk

This command sets or returns the specified Aggressor's crosstalk type, in Cascading mode.

Conditions S-Parameter Mode must be set to Cascading.

Number of ports must be either 8 or 12.

Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:CTALk {NEXT|FEXT|BOTH}`
`HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:CTALk?`

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments `[n] ::= {1|2}` (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

NEXT – Near-End Crosstalk

FEXT – Far-End Crosstalk
 BOTH – Near and Far-End Crosstalk

Returns NEXT
 FEXT
 BOTH

Examples HSSerial:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR1:CTALK FEXT
 sets the first Aggressor's Crosstalk type to Far End Crosstalk, in Cascading mode.

HSSerial:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2:CTALK? might
 return NEXT, indicating that the 2nd Aggressor crosstalk type is set to Far End
 Crosstalk, in Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:DRATe

This command sets or returns the specified Aggressor's data rate, in Cascading mode.

Conditions S-Parameter Mode must be set to Cascading.
 Number of ports must be either 8 or 12.
 Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:DRATe
 <data_rate>
 HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:DRATe?

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments [n] ::= {1|2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the
 number of ports is set to 12.)
 If omitted, n is interpreted as 1.
 <data_rate> ::= <NRf>

Returns A single <NR3> value.

Examples `HSSerial:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR1:DRATE 4E9`
 sets the first Aggressor's data rate to 4 Gbps, in Cascading mode.

`HSSerial:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2:DRATE?` might
 return `500.000000000E+3`, indicating that the 2nd Aggressor's data rate is
 set to 500 kbps, in Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal

This command sets or returns specified Aggressor's signal type, in Cascading mode.

Conditions S-Parameter Mode must be set to Cascading.
 Number of ports must be either 8 or 12.
 Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal`
`{CLOCK|PRBS|FILE|SAVictim}`
`HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal?`

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)
[HSSerial:CHANnel:SPARameter:CASCading:AGGRessor\[n\]:SIGNal:FILE](#)

Arguments `[n] ::= {1|2}` (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

CLOCK – Indicates that the aggressor signal is a clock pattern.

PRBS – Indicates that the aggressor signal is a PBRS pattern. You also must set the PBRS type.

FILE – Aggressor is set to use a file. You must set the file path.

SAVictim – Aggressor is the same as the victim.

Returns CLOC
 PRBS
 FILE
 SAV

Examples `HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR1:SIGNAL`
`SAVICTIM` sets the aggressor signal to be the same as the victim, in Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2:SIGNAL?`
might return `FILE`, indicating that 2nd Aggressor has a signal type set to use a file, in Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal:FILE

This command sets or returns the filepath to the aggressor file for the specified Aggressor, in Cascading mode.

Conditions S-Parameter Mode must be set to Cascading.
Number of ports must be either 8 or 12.
Aggressor signal type must be File.
Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal:`
`FILE <filepath>`
`HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal:`
`FILE?`

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)
[HSSerial:CHANnel:SPARameter:CASCading:AGGRessor\[n\]:SIGNal](#)

Arguments `[n] ::= {1|2}` ("n" determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.
`<filepath> ::= <string>` defines the path to the aggressor file.

Returns A single `<filepath>` string.

Examples `HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR1:SIGNAL:FILE`
`"C:\temp\myFile.txt"` sets the first Aggressor's file and filepath when the aggressor is set to use a file, in Cascading mode.

HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2:SIGNAL:FILE?
might return "C:\temp\myFile.txt" indicating that the 2nd Aggressor has a
signal type filepath set to "C:\temp\myFile.txt", in Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal:PRBS

This command sets or returns the specified Aggressor's PRBS signal type, in Cascading mode.

Conditions	S-Parameter Mode must be set to Cascading. Number of ports must be either 8 or 12. Aggressor signal type must be PRBS. Requires an S-Parameters license.
Group	Channel
Syntax	HSSerial:CHANnel:SPARameter: CASCading:AGGRessor[n]:SIGNal:PRBS {PRBS7 PRBS9 PRBS15 PRBS16 PRBS20 PRBS21 PRBS23 PRBS29 PRBS31} HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal: PRBS?
Related Commands	HSSerial:CHANnel:SPARameter:MODE HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal
Arguments	[n] ::= {1 2} ("n" determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.) If omitted, n is interpreted as 1. Patterns available include: PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31.
Returns	PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31
Examples	HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR1:SIGNAL:PRBS PRBS31 sets the first Aggressor's Signal type's PRBS value to PRBS31, in Cascading mode.

HSSERIAL:CHANNEL:SPARAMETER:CASCADING:AGGRESSOR2:SIGNAL:PRBS? might return PRBS15, indicating that the 2nd Aggressor has a signal type PRBS value set to PRBS15, in Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:DEEMbed

This command sets or returns whether the Cascading S-Parameters is to de-embed (invert) the S-Parameters, in Cascading mode.

Conditions S-Parameter Mode must be set to Cascading.
Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARameter:CASCading:DEEMbed {0|1|OFF|ON}
HSSerial:CHANnel:SPARameter:CASCading:DEEMbed?

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments OFF or 0 disables de-embedding. OFF or 0 is the default value.
ON or 1 enables de-embedding.

Returns A single <Boolean> value.

Examples HSSERIAL:CHANNEL:SPARAMETER:CASCADING:DEEMBED 1 will de-embed the S-Parameters for Cascading mode.

HSSERIAL:CHANNEL:SPARAMETER:CASCADING:DEEMBED? might return 0, indicating that S-Parameters will not be de-embedded for Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:DRX[n]

This command sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified receiver port number (Rx-Port) in Cascading mode and Differential Signalling Scheme (where applicable).

Conditions S-Parameter Mode must be set to Cascading.
S-Parameter Signalling Scheme must be set to Differential (where applicable).

Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANnel:SPARAmeter:CASCading:STAGe[m]:DRX[n] <port number>`
`HSSerial:CHANnel:SPARAmeter:CASCading:STAGe[m]:DRX[n]?`

Related Commands [HSSerial:CHANnel:SPARAmeter:MODE](#)
[HSSerial:CHANnel:SPARAmeter:CASCading:STYPe](#)
[HSSerial:CHANnel:SPARAmeter:CASCading:TYPE](#)
[HSSerial:CHANnel:SPARAmeter:CASCading:STAGe\[m\]:DTX\[n\]](#)

Arguments `[m]` ::= {1|2|3|4|5|6}. A variable value to define the Stage.
 If omitted, interpreted as 1

`[n]` ::= <NR1> value. A variable value to define the receiver port number (Rx-Port) of the channel.
 The actual range is dependent on the Number of Ports (Type).
 Type = 4, then `n` = {1}
 Type = 8 then `n` = {1 – 2}
 Type = 12 then `n` = {1 – 3}
 If omitted, `n` is interpreted as 1.

<port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Rx-Port of the channel.
 The actual range is dependent on the Number of Ports (Type).
 Type = 4 then <port number> = {1 – 2}
 Type = 8 then <port number> = {1 – 4}
 Type = 12 then <port number> = {1 – 6}

Returns A single <NR1> value.

Examples `HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE2:DRX2 4` assigns S-Parameter port 4 to the channel's receiver port 2 for Stage 2, in the Differential, Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE6:RX3?` might return 10, indicating that S-Parameter Port 10 is assigned to the channel's receiver port 3 for Stage 6, in the Differential, Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:DTX[n]

This command sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified transmission port number (Tx-Port) in Cascading mode and Differential Signalling Scheme (where applicable).

Conditions	S-Parameter Mode must be set to Cascading. S-Parameter Signalling Scheme must be set to Differential (where applicable). Requires an S-Parameters license.
Group	Channel
Syntax	HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:DTX[n] <port number> HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:DTX[n]?
Related Commands	HSSerial:CHANnel:SPARameter:MODE HSSerial:CHANnel:SPARameter:CASCading:STYPe HSSerial:CHANnel:SPARameter:CASCading:TYPE HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:DRX[n]
Arguments	[m] ::= {1 2 3 4 5 6}. A variable value to define the Stage. If omitted, interpreted as 1 [n] ::= <NR1> value. A variable value to define the transmission port number (Tx-Port) of the channel. The actual range is dependent on the Number of Ports (Type). Type = 4, then n = {1} Type = 8 then n = {1 – 2} Type = 12 then n = {1 – 3} If omitted, n is interpreted as 1. <port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel. The actual range is dependent on the Number of Ports (Type). Type = 4 then <port number> = {1 – 2} Type = 8 then <port number> = {1 – 4} Type = 12 then <port number> = {1 – 6}
Returns	A single <NR1> value.

- Examples**
- `HSSerial:CHANNEL:SPARAMETER:CASCADING:STAGE2:DTX2 4` assigns S-Parameter port 4 to the channel's transmission port 2 for Stage 2, in the Differential, Cascading mode.
- `HSSerial:CHANNEL:SPARAMETER:CASCADING:STAGE6:TX3?` might return 10, indicating that S-Parameter Port 10 is assigned to the channel's transmission port 3 for Stage 6, in the Differential, Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:RX[n]

This command sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified receiver port number (Rx-Port) in Cascading mode and Single-Ended Signalling Scheme (where applicable).

- Conditions**
- S-Parameter Mode must be set to Cascading.
 - S-Parameter Signalling Scheme must be set to Single-Ended (where applicable).
 - Requires an S-Parameters license.

Group Channel

Syntax

`HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:RX[n] <port number>`
`HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:RX[n]?`

Related Commands

- [HSSerial:CHANnel:SPARameter:MODE](#)
- [HSSerial:CHANnel:SPARameter:CASCading:SType](#)
- [HSSerial:CHANnel:SPARameter:CASCading:Type](#)
- [HSSerial:CHANnel:SPARameter:CASCading:STAGe\[m\]:TX\[n\]](#)

- Arguments**
- `[m]` ::= {1|2|3|4|5|6}. A variable value to define the Stage.
- If omitted, interpreted as 1
- `[n]` ::= <NR1> value. A variable value to define the receiver port number (Rx-Port) of the channel.
- The actual range is dependent on the Number of Ports (Type).
- Type = 1, then n = no value
 - Type = 2, then n = {1}
 - Type = 4 then n = {1 – 2}
 - Type = 6 then n = {1 – 3}
 - Type = 8 then n = {1 – 4}
 - Type = 12 then n = {1 – 6}

<port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 1, then <port number> = no value
 Type = 2, then <port number> = {1 – 2}
 Type = 4 then <port number> = {1 – 4}
 Type = 6 then <port number> = {1 – 6}
 Type = 8 then <port number> = {1 – 8}
 Type = 12 then <port number> = {1 – 12}

Returns A single <NR1> value.

Examples `HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE2:RX2 4` assigns S-Parameter port 4 to the channel's receiver port 2 for Stage 2, in the Single-Ended, Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE6:RX3?` might return 10, indicating that S-Parameter Port 10 is assigned to the channel's receiver port 3 for Stage 6, in the Single-Ended, Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:TX[n]

This command sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified transmission port number (Tx-Port) in Cascading mode and Single-Ended Signalling Scheme (where applicable).

Conditions S-Parameter Mode must be set to Cascading.
 S-Parameter Signalling Scheme must be set to Single-Ended (where applicable).
 Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:TX[n] <port number>`
`HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:TX[n]?`

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)
[HSSerial:CHANnel:SPARameter:CASCading:SType](#)
[HSSerial:CHANnel:SPARameter:CASCading:TYpe](#)
[HSSerial:CHANnel:SPARameter:CASCading:STAGe\[m\]:RX\[n\]](#)

Arguments	[m] ::= {1 2 3 4 5 6}. A variable value to define the Stage. If omitted, interpreted as 1 [n] ::= <NR1> value. A variable value to define the transmission port number (Tx-Port) of the channel. The actual range is dependent on the Number of Ports (Type). Type = 1, then n = no value Type = 2, then n = {1} Type = 4 then n = {1 – 2} Type = 6 then n = {1 – 3} Type = 8 then n = {1 – 4} Type = 12 then n = {1 – 6} <port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel. The actual range is dependent on the Number of Ports (Type). Type = 1, then <port number> = no value Type = 2, then <port number> = {1 – 2} Type = 4 then <port number> = {1 – 4} Type = 6 then <port number> = {1 – 6} Type = 8 then <port number> = {1 – 8} Type = 12 then <port number> = {1 – 12}
Returns	A single <NR1> value.
Examples	HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE2:TX2 4 assigns S-Parameter port 4 to the channel's transmission port 2 for Stage 2, in the Single-Ended, Cascading mode. HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE6:TX3? might return 10, indicating that S-Parameter Port 10 is assigned to the channel's transmission port 3 for Stage 6, in the Single-Ended, Cascading mode.

HSSerial:CHANnel:SPARAmeter:CASCading:STAGe[m]:ENABle

This command sets or returns the state of the specified Cascaded S-Parameter stage (enabled or disabled).

Conditions	S-Parameter Mode must be set to Cascading. Requires an S-Parameters license.
-------------------	--

Group	Channel
Syntax	<pre> HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:ENABle {0 1 OFF ON} HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:ENABle? </pre>
Related Commands	HSSerial:CHANnel:SPARameter:MODE HSSerial:CHANnel:SPARameter:NCAScading:FILE
Arguments	[m] ::= {1 – 6} ("m" determines the stage number) If omitted, m is interpreted as 1. OFF or 0 disables the specified Cascading Stage. OFF or 0 is the default value. ON or 1 enables the Stage.
Returns	A single <Boolean> value.
Examples	<pre> HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE6:ENABLE 1 enables Stage 6 in Cascading mode. HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE6:ENABLE? might return 0, indicating that Stage 6 is not enabled in Cascading mode. </pre>

HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:FILE

This command sets or returns the filepath for the specified S-Parameters Cascading Stage, in Cascading mode.

Conditions	S-Parameter Mode must be set to Cascading. Requires an S-Parameters license.
Group	Channel
Syntax	<pre> HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:FILE <filepath> HSSerial:CHANnel:SPARameter:CASCading:STAGe[m]:FILE </pre>
Related Commands	HSSerial:CHANnel:SPARameter:MODE

Arguments	[m] ::= {1 – 6} ("m" determines the stage number) If omitted, m is interpreted as 1. <filepath> ::= <string> defines the path to the S-Parameter file.
Returns	<filepath> ::= <string>.
Examples	HSSerial:CHANnel:SPARAmeter:CASCADING:STAGe1:FILE "C:\temp\myFile.s12p" sets the filepath to "C:\temp\myFile.s12p" for use during compilation for Stage 1. HSSerial:CHANnel:SPARAmeter:CASCADING:STAGe1:FILE? might return "C:\temp\myFile.s12p" indicating the filepath for Stage 1.

HSSerial:CHANnel:SPARAmeter:CASCading:STAGe[m]:SSCHeme

This command sets or returns the S-Parameter Signalling Scheme, in Cascading mode. Signalling Scheme is only available when the Number of Ports is set to 4, 8, or 12.

Conditions	S-Parameter Mode must be set to Cascading. Requires an S-Parameters license.
Group	Channel
Syntax	HSSerial:CHANnel:SPARAmeter:CASCading:STAGe[m]:SSCHeme {SENDED DIFFerential} HSSerial:CHANnel:SPARAmeter:CASCading:STAGe[m]:SSCHeme?
Related Commands	HSSerial:CHANnel:SPARAmeter:MODE
Arguments	[m] ::= {1 – 6} ("m" determines the stage number) If omitted, m is interpreted as 1. SENDED – Single Ended Signal Scheme DIFFerential – Differential Signal Scheme
Returns	SEND DIFF

Examples `HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE2:SSCHEME DIFF` sets the Stage 2 Signalling Scheme to Differential, in Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STAGE3:SSCHEME?` might return `SEND`, indicating that the Stage 3 Signalling Scheme is set to Single Ended, in Cascading mode.

HSSerial:CHANnel:SPARAmeter:CASCading:STYPe

This command sets or returns S-Parameter signal type (victim or aggressor), in Cascading mode. The number of ports must be either 8 or 12.

Conditions S-Parameter Mode must be set to Cascading.

Number of Ports must be either 8 or 12.

Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANnel:SPARAmeter:CASCading:STYPe`
 `{VICTim|AGGRessor|BOTH}`
 `HSSerial:CHANnel:SPARAmeter:CASCading:STYPe?`

Related Commands [HSSerial:CHANnel:SPARAmeter:MODE](#)

Arguments `VICTim` – enables the victim signal type.

`AGGRessor` – enables the aggressor signal type.

`BOTH` – enables the victim and aggressor signal types.

Returns `VICT`
 `AGGR`
 `BOTH`

Examples `HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STYPe BOTH` sets the signal type to include both the Victim and Aggressor signal types, in Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:CASCADING:STYPe?` might return `AGGR`, indicating that the S-Parameter signal type is currently set to be Aggressor, in Cascading mode.

HSSerial:CHANnel:SPARameter:CASCading:TYPE

This command sets or returns the S-Parameter number of ports, in Cascading mode.

Conditions Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARameter:CASCading:TYPE {2|4|6|8|12}
HSSerial:CHANnel:SPARameter:CASCading:TYPE?

Arguments {2|4|6|8|12} – defines the number of S-Parameter ports.

Returns A single <NR1> value.

Examples HSSerial:CHANnel:SPARameter:CASCADING:TYPE 12 sets the S-Parameter type to a 12-Port system for the cascading mode.

HSSerial:CHANnel:SPARameter:CASCADING:TYPE? might return 6, indicating that the S-Parameter type is a 6-Port system for Cascading mode.

HSSerial:CHANnel:SPARameter:MODE

This command sets or returns the S-Parameter mode (Cascading or Non-Cascading).

Conditions Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARameter:MODE {CASC|NCAS}
HSSerial:CHANnel:SPARameter:MODE?

Arguments CASCading sets the S-Parameter mode to cascading, allowing you to cascade up to six S-parameter files and apply the characteristics on the waveform.

NCASCading sets the S-Parameter mode to non-cascading, allowing you to apply S-parameter characteristics on the waveform from only one S-parameter file.

Returns CASC
NCASC

Examples HSSERIAL:CHANNEL:SPARAMETER:MODE CASCADING sets the S-Parameter mode to cascading.

HSSERIAL:CHANNEL:SPARAMETER:MODE? might return NCAS, indicating that the S-Parameter mode is set to Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor2[:ENABle]

This command sets or returns the aggressor 2 signal type state (enabled or disabled) in Non-Cascading mode. Aggressor2 signals are available when the number of ports is set to 12.

Conditions S-Parameter Mode must be set to Non-Cascading.
Number of Ports must be set to 12.
Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor2[:ENABle]
{0|1|ON|OFF}
HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor2[:ENABle]?

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments OFF or 0 disables the aggressor 2 signal type. OFF or 0 is the default value.
ON or 1 enables the aggressor 2 signal type.

Returns A single <Boolean> value.

Examples HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR2:ENABLE ON enables the aggressor 2 signal type, in Non-Cascading mode.

HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR2:ENABLE? might return 0, indicating that the aggressor 2 signal type is disabled, in Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor[n]:AMPLitude

This command sets or returns the specified Aggressor's amplitude, in Non-Cascading mode.

Conditions S-Parameter Mode must be set to Non-Cascading.
Number of ports must be either 8 or 12.
Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor[n]:
AMPLitude <amplitude>
HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor[n]:
AMPLitude?

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments [n] ::= {1|2} ("n" determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)
If omitted, n is interpreted as 1.
<amplitude> ::= <NR3> value.

Returns A single <NR3> value.

Examples HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR1:AMPLITUDE
200E-3 sets the 1st Aggressor's amplitude to 200 mV, in Non-Cascading mode.
HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR2:AMPLITUDE?
might return 100.0000000000E-3, indicating that the 2nd Aggressor's amplitude
is set to 100 mV, in Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor[n]:CTALk

This command sets or returns the specified Aggressor's crosstalk type, in Non-Cascading mode.

Conditions S-Parameter Mode must be set to Non-Cascading.
Number of ports must be either 8 or 12.

Requires an S-Parameters license.

Group	Channel
Syntax	<pre> HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor[n]:CTALK {NEXT FEXT BOTH} HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor[n]:CTALK? </pre>
Related Commands	HSSerial:CHANnel:SPARameter:MODE
Arguments	[n] ::= {1 2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.) If omitted, n is interpreted as 1. NEXT – Near-End Crosstalk FEXT – Far-End Crosstalk BOTH – Near and Far-End Crosstalk
Returns	NEXT FEXT BOTH
Examples	HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR1:CTALK FEXT sets the 1st Aggressor's Crosstalk type to Far End Crosstalk, in Non-Cascading mode. HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR2:CTALK? might return NEXT, indicating that the 2nd Aggressor crosstalk type is set to Near End Crosstalk, in Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor[n]:DRATe

This command sets or returns the specified Aggressor's data rate, in Non-Cascading mode.

Conditions	S-Parameter Mode must be set to Non-Cascading. Number of ports must be either 8 or 12. Requires an S-Parameters license.
-------------------	---

Group	Channel
Syntax	<pre> HSSerial:CHANNEL:SPARAmeter:NCAScading:AGGRessor[n]:DRATE <data_rate> HSSerial:CHANNEL:SPARAmeter:NCAScading:AGGRessor[n]:DRATE? </pre>
Related Commands	HSSerial:CHANnel:SPARAmeter:MODE
Arguments	[n] ::= {1 2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.) If omitted, n is interpreted as 1. <data_rate> ::= <NR3> value.
Returns	A single <NR3> value.
Examples	<pre> HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR1:DRATE 4E9 </pre> sets the 1st Aggressor's data rate to 4 Gbps, in Non-Cascading mode. <pre> HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR2:DRATE? </pre> might return 500.0000000000E+3, indicating that the 2nd Aggressor's data rate is set to 500 kbps, in Non-Cascading mode.

HSSerial:CHANnel:SPARAmeter:NCAScading:AGGRessor[n]:SIGNal

This command sets or returns specified Aggressor's signal type, in Non-Cascading mode.

Conditions	S-Parameter Mode must be set to Non-Cascading. Number of ports must be either 8 or 12. Requires an S-Parameters license.
Group	Channel
Syntax	<pre> HSSerial:CHANNEL:SPARAmeter:NCAScading:AGGRessor[n]:SIGNal {CLOCK PRBS FILE SAVictim} HSSerial:CHANNEL:SPARAmeter:NCAScading:AGGRessor[n]:SIGNal? </pre>

Related Commands	HSSerial:CHANnel:SPARameter:MODE HSSerial:CHANnel:SPARameter:CASCading:AGGRessor[n]:SIGNal:FILE
Arguments	[n] ::= {1 2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.) If omitted, n is interpreted as 1. CLOCK – Indicates that the aggressor signal is a clock pattern. PRBS – Indicates that the aggressor signal is a PBRs pattern. You also must set the PBRs type. FILE – Aggressor is set to use a file. You must set the file path. SAVictim – Aggressor is the same as the victim.
Returns	CLOC PRBS FILE SAV
Examples	HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR1:SIGNAL SAVICTIM sets the 1st aggressor signal to be the same as the victim, in Non-Cascading mode HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR2:SIGNAL? might return FILE, indicating that 2nd Aggressor has a signal type set to use a file, in Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:AGGRessor[n]:SIGNal:FILE

This command sets or returns the filepath to the aggressor file for the specified Aggressor, in Non-Cascading mode.

Conditions	S-Parameter Mode must be set to Non-Cascading. Number of ports must be either 8 or 12. Aggressor signal type must be File. Requires an S-Parameters license.
-------------------	--

Group	Channel
--------------	---------

Syntax HSSerial:CHANnel:SPARAmeter:NCAScading:AGGRessor[n]:SIGNal:
FILE <filepath>
HSSerial:CHANnel:SPARAmeter:NCAScading:AGGRessor[n]:SIGNal:
FILE?

Related Commands [HSSerial:CHANnel:SPARAmeter:MODE](#)
[HSSerial:CHANnel:SPARAmeter:NCAScading:AGGRessor\[n\]:SIGNal](#)

Arguments [n] ::= {1|2} ("n" determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

<filepath> ::= <string> defines the path to the aggressor file.

Returns A single <filepath> string.

Examples HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR1:SIGNAL:FILE
"C:\temp\myFile.txt" sets the 1st Aggressor's file and filepath when the aggressor is set to use a file, in Non-Cascading mode.

HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR2:SIGNAL:FILE?
might return "C:\temp\myFile.txt" indicating that the 2nd Aggressor has a signal type filepath set to "C:\temp\myFile.txt", in Non-Cascading mode.

HSSerial:CHANnel:SPARAmeter:NCAScading:AGGRessor[n]:SIGNal:PRBS

This command sets or returns the specified Aggressor's PRBS signal type, in Non-Cascading mode.

Conditions S-Parameter Mode must be set to Non-Cascading.

Number of ports must be either 8 or 12.

Aggressor signal type must be PRBS.

Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARAmeter:
NCAScading:AGGRessor[n]:SIGNal:PRBS
{PRBS7|PRBS9|PRBS15|PRBS16|PRBS20|PRBS21|PRBS23|PRBS29|PRBS31}
HSSerial:CHANnel:SPARAmeter:NCAScading:AGGRessor[n]:SIGNal:
PRBS?

Related Commands	HSSerial:CHANnel:SPARAmeter:MODE HSSerial:CHANnel:SPARAmeter:NCAScading:AGGRessor[n]:SIGNal
Arguments	[n] ::= {1 2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.) If omitted, n is interpreted as 1. Patterns available include: PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31
Returns	PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31
Examples	HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR1:SIGNAL:PRBS PRBS31 sets the 1st Aggressor's Signal type's PRBS value to PRBS31, in Non-Cascading mode. HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:AGGRESSOR2:SIGNAL:PRBS? might return PRBS15, indicating that the 2nd Aggressor has a signal type PRBS value set to PRBS15, in Non-Cascading mode.

HSSerial:CHANnel:SPARAmeter:NCAScading:DEEMbed

This command sets or returns whether the Non-Cascading S-Parameters is to de-embed (invert) the S-Parameters, in Non-Cascading mode.

Conditions	S-Parameter Mode must be set to Non-Cascading. Requires an S-Parameters license.
Group	Channel
Syntax	HSSerial:CHANnel:SPARAmeter:NCAScading:DEEMbed {0 1 OFF ON} HSSerial:CHANnel:SPARAmeter:NCAScading:DEEMbed?
Related Commands	HSSerial:CHANnel:SPARAmeter:MODE
Arguments	OFF or 0 disables de-embedding. OFF or 0 is the default value. ON or 1 enables de-embedding.
Returns	A single <Boolean> value.

- Examples** `HSSerial:CHANNEL:SPARAMETER:NCASCADING:DEEMBED 1` will de-embed the S-Parameters for Non-Cascading mode.
- `HSSerial:CHANNEL:SPARAMETER:NCASCADING:DEEMBED?` might return 0, indicating that S-Parameters will not be de-embedded for Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:DRX[n]

This command sets or returns the S-Parameter port assignment of the channel's specified receiver port number (Rx-Port) in Non-Cascading mode and Differential Signalling Scheme (where applicable).

- Conditions** S-Parameter Mode must be set to Non-Cascading.
S-Parameter Signalling Scheme must be set to Differential.
Requires an S-Parameters license.
- Group** Channel
- Syntax** `HSSerial:CHANNEL:SPARAMETER:NCASCADING:DRX[n] <port_number>`
`HSSerial:CHANNEL:SPARAMETER:NCASCADING:DRX[n]?`
- Related Commands** [HSSerial:CHANnel:SPARameter:MODE](#)
[HSSerial:CHANnel:SPARameter:CASCading:STYPe](#)
[HSSerial:CHANnel:SPARameter:NCAScading:TYPE](#)
[HSSerial:CHANnel:SPARameter:NCAScading:RX\[n\]](#)
- Arguments** `[n] ::= <NR1>`. A variable value to define the receiver port number (Rx-Port) of the channel.
- The actual range is dependent on the Number of Ports (Type).
- Type = 4, then `n = {1}`
Type = 8 then `n = {1 – 2}`
Type = 12 then `n = {1 – 3}`
- If omitted, `n` is interpreted as 1.
- `<port_number> ::= <NR1>` value. A variable value to define the S-Parameter Port assigned to the specified Rx-Port of the channel.
- The actual range is dependent on the Number of Ports (Type).
- Type = 4, then `n = {1 – 2}`
Type = 8 then `<port number> = {1 – 4}`
Type = 12 then `<port number> = {1 – 6}`

Returns A single <NR1> value.

Examples `HSSerial:CHANNEL:SPARAMETER:NCASCADING:DTX2 4` assigns S-Parameter port 4 to channel's receiver port 2, in the Differential, Non-Cascading mode.

`HSSerial:CHANNEL:SPARAMETER:NCASCADING:DTX3?` might return 6, indicating that S-Parameter Port 6 is assigned to the channel's receiver port 3, in the Differential, Non-Cascading mode.

HSSerial:CHANnel:SPARAmeter:NCAScading:DTX[n]

This command sets or returns the S-Parameter port assignment of the channel's specified transmission port number (Tx-Port) in Non-Cascading mode and Differential Signalling Scheme (where applicable).

Conditions S-Parameter Mode must be set to Non-Cascading.
S-Parameter Signalling Scheme must be set to Differential.
Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANNEL:SPARAMETER:NCASCADING:DTX[n] <port_number>`
`HSSerial:CHANNEL:SPARAMETER:NCASCADING:DTX[n]?`

Related Commands [HSSerial:CHANnel:SPARAmeter:MODE](#)
[HSSerial:CHANnel:SPARAmeter:NCAScading:STYPe](#)
[HSSerial:CHANnel:SPARAmeter:NCAScading:TYPE](#)
[HSSerial:CHANnel:SPARAmeter:NCAScading:DRX\[n\]](#)

Arguments `[n] ::= <NR1> value`. A variable value to define the transmission port number (Tx-Port) of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 4, then `n = {1}`
Type = 8 then `n = {1 – 2}`
Type = 12 then `n = {1 – 3}`

If omitted, `n` is interpreted as 1.

`<port_number> ::= <NR1> value`. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 4 then <port number> = {1 – 2}
 Type = 8 then <port number> = {1 – 4}
 Type = 12 then <port number> = {1 – 6}

Returns A single <NR1> value.

Examples `HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:DTX2 4` assigns S-Parameter port 4 to channel's transmission port 2, in the Differential, Non-Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:DTX3?` might return 6, indicating that S-Parameter Port 6 is assigned to the channel's transmission port 3, in the Differential, Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:FILE

This command sets or returns the filepath and file name of the S-Parameter file, in Non-Cascading mode.

Conditions S-Parameter Mode must be set to Non-Cascading.
 Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANnel:SPARameter:NCAScading:FILE <filepath>`

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments <filepath>::= <string> defines the path to the S-Parameter file.

Returns A single <filepath> string.

Examples `HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:FILE "C:\temp\myFile.s12p"` sets the filepath to "C:\temp\myFile.s12p" for use during compilation.

`HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:FILE?` might return "C:\temp\myOtherFile.s6p", indicating the current filepath.

HSSerial:CHANnel:SPARameter:NCAScading:LAYout

This command sets or returns the 4 port S-Parameter Matrix Configuration, in Non-Cascading mode.

Conditions S-Parameter Mode must be set to Non-Cascading.
Number of Ports must be set to 4.
Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARameter:NCAScading:LAYout
{TYPical|ALTErnate}
HSSerial:CHANnel:SPARameter:NCAScading:LAYout?

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments TYPical or ALTErnate: selects the S-Parameter Matrix.

S-Parmeter Matrix Typical				S-Parmeter Matrix Alternate			
SDD11	SDD12	SDC11	SDC12	SCC11	SCC12	SCD11	SCD12
SDD21	SDD22	SDC21	SDC22	SCC21	SCC22	SCD21	SCD22
SCD11	SCD12	SCC11	SCC12	SDC11	SDC12	SDD11	SDD12
SCD21	SCD22	SCC21	SCC22	SDC21	SDC22	SDD21	SDD22

Returns TYP
ALT

Examples HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:LAYOUT TYPICAL sets the 4 port configuration's Layout to Typical, in Non-Cascading mode.

HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:LAYOUT? might return TYP, indicating that configuration's Layout for port 4 is set to Typical, in Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:RX[n]

This command sets or returns the S-Parameter port assignment of the channel's specified receiver port number (Rx-Port) in Non-Cascading mode and Single-Ended Signalling Scheme (where applicable).

Conditions S-Parameter Mode must be set to Non-Cascading.
S-Parameter Signalling Scheme must be set to Single-Ended.
Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARameter:NCAScading:RX[n] <port number>
HSSerial:CHANnel:SPARameter:NCAScading:RX[n]?

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)
[HSSerial:CHANnel:SPARameter:CASCading:STYPe](#)
[HSSerial:CHANnel:SPARameter:NCAScading:TYPE](#)
[HSSerial:CHANnel:SPARameter:NCAScading:TX\[n\]](#)

Arguments [n] ::= <NR1>. A variable value to define the receiver port number (Rx-Port) of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 1, then n = no value
Type = 2, then n = {1}
Type = 4 then n = {1 – 2}
Type = 6 then n = {1 – 3}
Type = 8 then n = {1 – 4}
Type = 12 then n = {1 – 6}

If omitted, n is interpreted as 1.

<port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Rx-Port of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 1, then <port number> = no value
Type = 2, then <port number> = {1 – 2}
Type = 4 then <port number> = {1 – 4}
Type = 6 then <port number> = {1 – 6}
Type = 8 then <port number> = {1 – 8}
Type = 12 then <port number> = {1 – 12}

Returns A single <NR1> value.

Examples `HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:RX2 4` assigns S-Parameter port 4 to the channel's receiver port 2, in the Single-Ended, Non-Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:RX4?` might return 6, indicating that S-Parameter Port 6 is assigned to the channel's receiver port 4, in the Single-Ended, Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:SSCHeme

This command sets or returns the S-Parameter Signalling Scheme, in Non-Cascading mode. Signalling Scheme is only available when the Number of Ports is set to 4, 8, or 12.

Conditions S-Parameter Mode must be set to Non-Cascading.
Requires an S-Parameters license.

Group Channel

Syntax `HSSerial:CHANnel:SPARameter:NCAScading:SSCHeme {SENDeD|DIFFerential}`
`HSSerial:CHANnel:SPARameter:NCAScading:SSCHeme?`

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments SENDeD – Single Ended Signal Scheme
DIFFerential – Differential Signal Scheme

Returns SEND
DIFF

Examples `HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:SSCHEME DIFF` sets the Signalling Scheme to Differential, in Non-Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:SSCHEME?` might return SEND, indicating that the Signalling Scheme is set to Single Ended, in Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:STYPe

This command sets or returns S-Parameter signal type (victim or aggressor), in Non-Cascading mode. The number of ports must be either 8 or 12.

Conditions S-Parameter Mode must be set to Non-Cascading.
Number of Ports must be either 8 or 12.
Requires an S-Parameters license.

Group Channel

Syntax HSSerial:CHANnel:SPARameter:NCAScading:STYPe
{VICTim|AGGRessor|BOTH}
HSSerial:CHANnel:SPARameter:NCAScading:STYPe?

Related Commands [HSSerial:CHANnel:SPARameter:MODE](#)

Arguments VICTim – enables the victim signal type.
AGGRessor – enables the aggressor signal type.
BOTH – enables the victim and aggressor signal types.

Returns VICT
AGGR
BOTH

Examples HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:STYPE BOTH sets the signal type to include both the Victim and Aggressor signal types, in Non-Cascading mode.

HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:STYPe? might return AGGR, indicating that the S-Parameter signal type is currently set to be Aggressor, in Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:TX[n]

This command sets or returns the S-Parameter port assignment of the channel's specified transmission port number (Tx-Port) in Non-Cascading mode and Single-Ended Signalling Scheme (where applicable).

Conditions	S-Parameter Mode must be set to Non-Cascading. S-Parameter Signalling Scheme must be set to Single-Ended (where applicable). Requires an S-Parameters license.
Group	Channel
Syntax	<pre>HSSerial:CHANnel:SPARameter:NCAScading:TX[n] <port number> HSSerial:CHANnel:SPARameter:NCAScading:TX[n]?</pre>
Related Commands	HSSerial:CHANnel:SPARameter:MODE HSSerial:CHANnel:SPARameter:CASCading:STYPe HSSerial:CHANnel:SPARameter:NCAScading:TYPE HSSerial:CHANnel:SPARameter:NCAScading:RX[n]
Arguments	[n] ::= <NR1>. A variable value to define the transmission port number (Tx-Port) of the channel. The actual range is dependent on the Number of Ports (Type). Type = 1, then n = no value Type = 2, then n = {1} Type = 4 then n = {1 – 2} Type = 6 then n = {1 – 3} Type = 8 then n = {1 – 4} Type = 12 then n = {1 – 6} If omitted, n is interpreted as 1. <port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel. The actual range is dependent on the Number of Ports (Type). Type = 1, then <port number> = no value Type = 2, then <port number> = {1 – 2} Type = 4 then <port number> = {1 – 4} Type = 6 then <port number> = {1 – 6} Type = 8 then <port number> = {1 – 8} Type = 12 then <port number> = {1 – 12}
Returns	A single <NR1> value.
Examples	<pre>HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:TX2 4</pre> assigns S-Parameter port 4 to the channel's transmission port 2, in the Single-Ended, Non-Cascading mode.

`HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:TX4?` might return 6, indicating that S-Parameter Port 6 is assigned to the channel's transmission port 4, in the Single-Ended, Non-Cascading mode.

HSSerial:CHANnel:SPARameter:NCAScading:TYPE

This command sets or returns the S-Parameter number of ports, in Non-Cascading mode.

Conditions	Requires an S-Parameters license.
Group	Channel
Syntax	<code>HSSerial:CHANnel:SPARameter:NCAScading:TYPE {1 2 4 6 8 12}</code>
Arguments	{1 2 4 6 8 12} – defines the number of S-Parameter ports.
Returns	A single <NR1> value.
Examples	<code>HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:TYPE 12</code> sets the S-Parameter type to a 12-Port system for Non-Cascading mode. <code>HSSERIAL:CHANNEL:SPARAMETER:NCASCADING:TYPE?</code> might return 6, indicating that the S-Parameter type is a 6-Port system for Non-Cascading mode.

HSSerial:CHANnel:TYPE

This command sets or returns the Channel emulation type.

Group	Channel
Syntax	<code>HSSerial:CHANnel:TYPE {ISI SPAR}</code>
Arguments	ISI sets the Channel emulation type to Intersymbol Interference. SPAR sets the Channel emulation type to S-Parmeters.
Returns	ISI SPAR

Examples `HSSERIAL:CHANNEL:TYPE ISI` sets the Channel emulation type to Intersymbol Interference.

`HSSERIAL:CHANNEL:TYPE?` might return `SPAR`, indicating that the Channel emulation type is set to S-Parameters.

HSSerial:COMPIle (No Query Form)

This command compiles and generates a waveform using the High Speed Serial plug-in compile settings.

Conditions The active plug-in must be High Speed Serial.

This is an overlapping command. Overlapping commands run concurrently with other commands, allowing additional commands to start before the overlapping command has finished.

Group Compile

Syntax `HSSerial:COMPIle`

Related Commands [WPLugin:ACTive](#)

Examples `HSSERIAL:COMPILE` compiles and generates the waveforms.

HSSerial:COMPIle:ATSequence:ENABLE

This command sets or returns the Add to sequence compile state (enabled or disabled) for the Compile Settings.

Group Compile

Syntax `HSSerial:COMPIle:ATSequence:ENABLE {0|1|OFF|ON}`
`HSSerial:COMPIle:ATSequence:ENABLE?`

Arguments OFF or 0 disables Add to sequence. OFF or 0 is the default value.
 ON or 1 enables Add to sequence.

Returns A single <Boolean> value.

Examples `HSSERIAL:COMPILE:ATSEQUENCE:ENABLE 1` enables Add to sequence.
`HSSERIAL:COMPILE:ATSEQUENCE:ENABLE?` might return 0, indicating that Add to sequence is not enabled.

HSSerial:COMPILE:ATSequence:SEQUENCE

This command sets or returns the sequence name in the sequence list when the Add to an existing sequence is enabled for the Compile Settings.

To set, the sequence must exist in the Sequence List.

Group Compile

Syntax `HSSerial:COMPILE:ATSequence:SEQUENCE <sequence_name>`
`HSSerial:COMPILE:ATSequence:SEQUENCE?`

Related Commands [HSSerial:COMPILE:ATSequence:TRACK](#)

Arguments `<sequence_name>::=<string>` defines the sequence name in the sequence list.

Returns A single `<sequence_name>` string.

Examples `HSSERIAL:COMPILE:ATSEQUENCE:SEQUENCE "SEQUENCE_1"` sets the sequence named Sequence_1 as the target sequence when Add to an existing sequence is enabled for the compile settings.
`HSSERIAL:COMPILE:ATSEQUENCE:SEQUENCE?` might return "Sequence_2", indicating this is the selected sequence name.

HSSerial:COMPILE:ATSequence:TRACK

This command sets or returns the track name when the Add to an existing sequence is enabled for the Compile Settings.

To set, the track must exist in the named sequence.

Group Compile

Syntax `HSSerial:COMPILE:ATSequence:TRACK <track>`
`HSSerial:COMPILE:ATSequence:TRACK?`

Related Commands [HSSerial:COMPILE:ATSequence:SEquence](#)

Arguments <track>::=<string> defines the track name.

Returns A single <track> string.

Examples HSSERIAL:COMPILE:ATSEQUENCE:TRACK "Track 2" sets the track named Track 2 as the target track when Add to an existing sequence is enabled for the compile settings.

HSSERIAL:COMPILE:ATSEQUENCE:TRACK? might return "Track_2", indicating this is the selected track name.

HSSerial:COMPILE:CANCEL (No Query Form)

This command cancels a compilation currently in progress.

Conditions The active plug-in must be High Speed Serial.

Group Compile

Syntax HSSerial:COMPILE:CANCEL

Related Commands [WPLugin:ACTIVE](#)

Examples HSSERIAL:COMPILE:CANCEL immediately ends the current compile process.

HSSerial:COMPILE:CORREction:APPLY

This command enables or disables applying a corrections file during compile.

Conditions delete if no conditions

Group Compile

Syntax HSSerial:COMPILE:CORREction:APPLY {0|1|OFF|ON}
HSSerial:COMPILE:CORREction:APPLY?

Related Commands [HSSerial:COMPILE:CORRection:PATH](#)

Arguments OFF or 0 enables applying a corrections file. OFF or 0 is the default value.
ON or 1 disables applying a corrections file.

Returns A single <Boolean> value.

Examples `HSSERIAL:COMPILE:CORRECTION:APPLY ON` applies a correction file during compile.
`HSSERIAL:COMPILE:CORRECTION:APPLY?` might return 0, indicating that adding a correction file is disabled.

HSSerial:COMPILE:CORRection:PATH

This command sets or returns the path of the corrections file to use during compile.

Group Compile

Syntax `HSSerial:COMPILE:CORRection:PATH <filepath>`
`HSSerial:COMPILE:CORRection:PATH?`

Related Commands [HSSerial:COMPILE:CORRection:APPLY](#)

Arguments <filepath>::=<string> defines path to the corrections file.

Returns A single <filepath> string.

Examples `HSSERIAL:COMPILE:CORRECTION:PATH`
`"C:\temp\CorrectionFile.corr"` sets the Correction File's path and filename.
`HSSERIAL:COMPILE:CORRECTION:PATH?` might return
`"C:\temp\CorrectionFile.corr"`.

HSSerial:COMPILE:NAME

This commands sets or returns the name of the compiled waveform.

Group	Compile
Syntax	HSSerial:COMPILE:NAME <signal_name> HSSerial:COMPILE:NAME?
Arguments	<signal_name>::=<string> defines the signal name to be created.
Returns	A single <signal_name> string.
Examples	HSSERIAL:COMPILE:NAME "HSSWfm" sets the waveform name to HSSWfm. HSSERIAL:COMPILE:NAME? might return "HSSWfm".

HSSerial:COMPILE:OPTions

This command sets or returns the “Compile and assign” setting for the Compile Settings.

When Compile and assign is disabled, Compile only is enabled by default.

Group	Compile
Syntax	HSSerial:COMPILE:OPTions {0 1 OFF ON} HSSerial:COMPILE:OPTions?
Related Commands	HSSerial:COMPILE:PLAY
Arguments	OFF or 0 disables Compile and assign. OFF or 0 is the default value. ON or 1 enables Compile and assign.
Returns	A single <Boolean> value.
Examples	HSSERIAL:COMPILE:OPTIONS 1 enables Compile and assign. HSSERIAL:COMPILE:OPTIONS? might return 0, indicating that Compile and assign is disabled (which also means Compile only is enabled).

HSSerial:COMPILE:OVERwrite

This command enables or disables overwriting an existing waveform name.

Group	Compile
Syntax	<code>HSSerial:COMPILE:OVERwrite {0 1 OFF ON}</code> <code>HSSerial:COMPILE:OVERwrite?</code>
Arguments	OFF or 0 disables overwriting an existing waveform. OFF or 0 is the default value. ON or 1 enables Overwrite existing waveform.
Returns	A single <Boolean> value.
Examples	<code>HSSERIAL:COMPILE:OVERWRITE 1</code> enables the Overwrite the existing waveform setting. <code>HSSERIAL:COMPILE:OVERWRITE?</code> might return 0, indicating that the Overwrite existing waveform setting is not enabled.

HSSerial:COMPILE:PLAY

This command sets or returns the Play after assign setting for the Compile Settings.
Play after assign is active only when Compile and assign is enabled.

Group	Compile
Syntax	<code>HSSerial:COMPILE:PLAY {0 1 OFF ON}</code> <code>HSSerial:COMPILE:PLAY?</code>
Related Commands	HSSerial:COMPILE:OPTions
Arguments	OFF or 0 disables Play after assign. OFF or 0 is the default value. ON or 1 enables Play after assign.
Returns	A single <Boolean> value.

Examples `HSSERIAL:COMPILE:PLAY 1` enables Play after assign.
`HSSERIAL:COMPILE:PLAY?` might return 0, indicating that Play after assign is disabled.

HSSerial:COMPILE:REPEAT

This command sets or returns the Manual Repeat Count setting for the Compile Settings.

Conditions The manual setting is automatically disabled if the Auto Repeat Count is enabled.

Group Compile

Syntax `HSSerial:COMPILE:REPEAT <value>`
`HSSerial:COMPILE:REPEAT?`

Related Commands [HSSerial:COMPILE:REPEAT:MTYPE](#)
[HSSerial:COMPILE:REPEAT:AUTO](#)

Arguments `<value>::=<NR1>` when count type is set to count.
`<value>::=<NR3>` when count type is set to seconds.

Returns A single `<NR1>` value when count type is set to count.
A single `<NR3>` value when count type is set to seconds.

Examples `HSSERIAL:COMPILE:REPEAT 3` sets the Manual Repeat Count to 3.
`HSSERIAL:COMPILE:REPEAT?` might return `2.0000000000E-9`, indicating that Manual repeat is set to 2 ns.

HSSerial:COMPILE:REPEAT:AUTO

This command sets or returns the Auto Repeat Count setting.

Conditions The manual setting is automatically enabled if the Auto Repeat Count is disabled.

Group Compile

Syntax `HSSerial:COMPILE:REPEAT:AUTO {0|1|OFF|ON}`
`HSSerial:COMPILE:REPEAT:AUTO`

Related Commands [HSSerial:COMPILE:REPEAT](#)

Arguments OFF or 0 disables Auto Repeat Count.
ON or 1 enables Auto Repeat Count. On or 1 is the default value.

Returns A single <Boolean> value.

Examples `HSSERIAL:COMPILE:REPEAT:AUTO 0` disables the Auto Repeat Count setting.
`HSSERIAL:COMPILE:REPEAT:AUTO?` might return 1, indicating that Auto Repeat Count is enabled.

HSSerial:COMPILE:REPEAT:MTYPE

This command sets or returns the Manual Repeat Count type.

Group Compile

Syntax `HSSerial:COMPILE:REPEAT:MTYPE {COUNT|TIME}`

Related Commands [HSSerial:COMPILE:REPEAT](#)

Arguments COUNT allows the user to set the number of times to repeat the waveform.
TIME allows the user to enter a waveform length in time.

Returns COUN
TIME

Examples `HSSERIAL:COMPILE:REPEAT:MTYPE COUNT` sets the Manual repeat type to Count.
`HSSERIAL:COMPILE:REPEAT:MTYPE?` might return TIME, indicating that the Manual repeat type is set to Time.

HSSerial:COMPile:SPUI

This command sets or returns the value of the Samples/Unit Interval for the Compile Settings.

Group Compile

Syntax HSSerial:COMPile:SPUI <samples/UI>
HSSerial:COMPile:SPUI?

Related Commands [HSSerial:COMPile:STYPe](#)

Arguments <samples/UI>::=<NR3>

The default value is 6.

Range:	AWG70001A	1 to 50
	AWG70002A	1 to 25

Returns A single <NR3> value.

Examples HSSerial:COMPILE:SPUI 3 sets the Samples/Unit Interval to 3.

HSSerial:COMPILE:SPUI? might return 25.0000000000, indicating that the Samples/Unit Interval is set to 25.

HSSerial:COMPile:SRATe

This command sets or returns the sampling rate for the Compile settings.

Group Compile

Syntax HSSerial:COMPile:SRATe <NR3>
HSSerial:COMPile:SRATe?

Related Commands [HSSerial:COMPile:STYPe](#)

Arguments	A single <NR3> value.		
	Range:	AWG70001A	1.49 kS/s to 50 GS/s
		AWG70002A	1.49 kS/s to 25 GS/s
Returns	A single <NR3> value.		
Examples	HSSERIAL:COMPILE:SRATE 6E9 sets the Sampling Rate to 6 GS/s in the Compile settings. HSSERIAL:COMPILE:SRATE? might return 6.000000000E+9, indicating the Sampling Rate is set to 6 GS/s.		

HSSerial:COMPILE:SType

This command sets or returns the Sampling Rate type for the Compile Settings.

Group	Compile
Syntax	HSSerial:COMPILE:SType {AUTO SRATE SPUI} HSSerial:COMPILE:SType?
Related Commands	HSSerial:COMPILE:SPUI , HSSerial:COMPILE:SRATE
Arguments	AUTO: Sets the Sampling Rate to be automatically calculated by the system. SRATE: Sets the Sampling Rate to be manually set by the user. SPUI: Sets the Sampling Rate to Samples/Unit Interval.
Returns	AUTO SRAT SPUI
Examples	HSSERIAL:COMPILE:SType SPUI sets the Sampling Rate type to Samples/Unit Interval. HSSERIAL:COMPILE:SType? might return SRAT, indicating that the Sampling Rate is set to Manual.

HSSerial:DCD:ENABLe

This command sets or returns the Duty Cycle Distortion (DCD) state (enabled or disabled).

Group Pattern Duty Cycle Distortion

Syntax HSSerial:DCD:ENABLe {0|1|OFF|ON}
HSSerial:DCD:ENABLe?

Related Commands [HSSerial:DCD:VALue](#)

Arguments OFF or 0 disables Duty Cycle Distortion (DCD). OFF or 0 is the default value. ON or 1 enables Duty Cycle Distortion (DCD).

Returns A single <Boolean> value.

Examples HSSERIAL:DCD:ENABLE 1 enables DCD.
HSSERIAL:DCD:ENABLE? might return 0, indicating that DCD is disabled.

HSSerial:DCD:VALue

This command sets or returns the Duty Cycle Distortion (DCD) value.

Conditions The default units is UI (Unit Intervals) and is typically how this control is defined. Optionally, the units can be set as seconds. When setting the value, you must send values appropriate for the user interface setting. Units is set directly from the user interface for this value.

See the examples for using the set and query forms as either UI or seconds.

Group Pattern Duty Cycle Distortion

Syntax HSSerial:DCD:VALue <distortion>
HSSerial:DCD:VALue?

Related Commands [HSSerial:DCD:ENABLe](#)

Arguments	<distortion>::=<NR3> value. Range as UI: 0.01 UI to 0.5 UI Range in seconds: 10 ps to 500 ps
Returns	A single <NR3> value.
Examples	Unit Intervals HSSERIAL:DCD:VALUE 0.5 sets the DCD value to 0.5 UI. HSSERIAL:DCD:VALUE? might return 10.0000000000E-3, indicating that the DCD value is set to 0.01 UI . Seconds HSSERIAL:DCD:VALUE 10E-12 sets the DCD value to 10 ps. HSSERIAL:DCD:VALUE? might return 10.0000000000E-12, indicating that the DCD value is set to 10 ps.

HSSerial:DRATe

This command sets or returns the Data Rate.

Group	Pattern Basic Settings
Syntax	HSSerial:DRATe <rate> HSSerial:DRATe?
Arguments	<rate>::= <NR3>.
Returns	A single <NR3> value.
Examples	HSSERIAL:DRATE 6E9 sets the data rate to 6 GHz. HSSERIAL:DRATE? might return 1.0000000000E+9, indicating that the Data Rate is set to 1 Gbps.

HSSerial:ENCode:ENCo8b10b:DISParity

This command sets or returns the 8B/10B encoding state setting.

Group	Pattern Encoding/Modulation
Syntax	<code>HSSerial:ENCode:ENCo8b10b:DISParity {RDPLus RDMinus}</code> <code>HSSerial:ENCode:ENCo8b10b:DISParity?</code>
Related Commands	HSSerial:ENCode:ENCo8b10b:ENABle
Arguments	RDPLus is running disparity plus 1. RDMinus is running disparity minus 1.
Returns	RDPL RDM
Examples	<code>HSSERIAL:ENCODE:ENCO8B10B:DISPARITY RDPLus</code> sets the disparity to RD+. <code>HSSERIAL:ENCODE:ENCO8B10B:DISPARITY?</code> might return RDM, indicating the disparity is set to RD-.

HSSerial:ENCode:ENCo8b10b:ENABle

This command sets or returns the 8B/10B encoding state (enabled or disabled).

The Base Pattern Data must be set to Pattern and the Pattern type set to Symbol to use 8B/10B encoding.

Group	Pattern Encoding/Modulation
Syntax	<code>HSSerial:ENCode:ENCo8b10b:ENABle {0 1 OFF ON}</code> <code>HSSerial:ENCode:ENCo8b10b:ENABle?</code>
Related Commands	HSSerial:ENCode:ENCo8b10b:DISParity
Arguments	OFF or 0 disables 8B/10B encoding. OFF or 0 is the default value. ON or 1 enables 8B/10B encoding.
Returns	A single <Boolean> value.

Examples `HSSerial:ENCODE:ENCO8B10B:ENABLE 1` enables 8B/10B encoding.
`HSSerial:ENCODE:ENCO8B10B:ENABLE?` might return 0, indicating that 8B/10B encoding is disabled.

HSSerial:ENCode:PAM:LEVELs

This command sets or returns the Pulse-amplitude Modulation (PAM) encoding.

Group Pattern Encoding/Modulation

Syntax `HSSerial:ENCode:PAM:LEVELs {PAM4|PAM8|PAM16}`
`HSSerial:ENCode:PAM:LEVELs?`

Arguments PAM4 is 4 levels of PAM modulation.
PAM8 is 8 levels of PAM modulation.
PAM16 is 16 levels of PAM modulation.

Returns PAM4
PAM8
PAM16

Examples `HSSerial:ENCODE:PAM:LEVELS PAM8` sets the PAM encoding to eight levels.
`HSSerial:ENCODE:PAM:LEVELS?` might return PAM4, indicating that the PAM encoding is set to 4 levels.

HSSerial:ENCode:PAM:NVALue

This command sets or returns the PAM Normalized levels value for the specified row.

Group Pattern Encoding/Modulation

Syntax `HSSerial:ENCode:PAM:NVALue <rownum>,<value>`
`HSSerial:ENCode:PAM:NVALue? <rownum>`

Arguments `<rownum>::=<NR1>` is the row number of the PAM level. 0 specifies the first row.
`<value>::=<NR3>` is the PAM normalized levels value.

Returns A single <NR3> value.

Examples `HSSERIAL:ENCODE:PAM:NVALUE1,0.3` sets the PAM Normalized Level for the second row to 0.3.

`HSSERIAL:ENCODE:PAM:NVALUE?` might return `333.0000000000E-3`, indicating that the PAM normalized level value is 0.3.

HSSerial:ENCode:PWM:ENABLE

This command sets or returns Pulse Width Modulation (PWM) state (enabled or disabled).

Group Pattern Encoding/Modulation

Syntax `HSSerial:ENCode:PWM:ENABLe {0|1|OFF|ON}`
`HSSerial:ENCode:PWM:ENABLe?`

Related Commands [HSSerial:ENCode:PWM:TMINor](#)

Arguments OFF or 0 disables PWM. OFF or 0 is the default value.
 ON or 1 enables PWM.

Returns A single <Boolean> value.

Examples `HSSERIAL:ENCODE:PWM:ENABLE 1` enables the PWM.

`HSSERIAL:ENCODE:PWM:ENABLE?` might return 0, indicating that PWM is disabled.

HSSerial:ENCode:PWM:TMINor

This command sets or returns the T_Minor value of the PWM.

Group Pattern Encoding/Modulation

Syntax `HSSerial:ENCode:PWM:TMINor <t_minor>`
`HSSerial:ENCode:PWM:TMINor?`

Related Commands [HSSerial:ENCode:PWM:ENABle](#)

Arguments <t_minor>::=<NR3> value.

Returns A single <NR3> value.

Examples HSSERIAL:ENCODE:PWM:TMINOR 0.2 sets the T_Minor value to 0.2 UI.
HSSERIAL:ENCODE:PWM:TMINOR? might return 340.0000000000E-3,
indicating that the T Minor value is set to 0.34 UI.

HSSerial:ENCode:SCHeMe

This command sets or returns the Signaling Scheme on the base data pattern.

Group Pattern Encoding/Modulation

Syntax HSSerial:ENCode:SCHeMe {NRZ|NRZI|PAM}
HSSerial:ENCode:SCHeMe?

Arguments NRZ sets the signaling scheme to non-return to zero.
NRZI sets the signaling scheme to non-return to zero inverted encoding.
PAM sets the signaling scheme to pulse amplitude encoding.

Returns NRZ
NRZI
PAM

Examples HSSERIAL:ENCODE:SCHEME NRZI sets the signaling scheme to NRZI.
HSSERIAL:ENCODE:SCHEME? might return NRZ, indicating the signaling scheme
is NRZ.

HSSerial:IState:OFFSet

This command sets or returns the idle state amplitude offset.

Group Pattern Basic Settings

Syntax	<code>HSSerial:ISTate:OFFSet <offset></code> <code>HSSerial:ISTate:OFFSet?</code>
Related Commands	HSSerial:BDATa
Arguments	<code><offset> ::= <NR3> value.</code>
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:ISTATE:OFFSET 200E-3</code> sets the idle state offset to 200 mV. <code>HSSERIAL:ISTATE:OFFSET?</code> might return <code>20.0000000000E-3</code> indicating that the idle state offset is set to 20 mV.

HSSerial:ISTate:VALue

This command sets or returns the idle state time.

Group	Pattern Basic Settings
Syntax	<code>HSSerial:ISTate:VALue <time></code> <code>HSSerial:ISTate:VALue?</code>
Arguments	<code><time> ::= <NR3> value.</code>
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:ISTATE:VALUE 16E-9</code> sets the idle state time to 16 ns. <code>HSSERIAL:ISTATE:VALUE?</code> might return <code>10.0000000000E-9</code>, indicating the idle state time is set to 10 ns.

HSSerial:MARKer[n]:CLOCK:FREQuency

This command sets or returns the user defined clock frequency for the specified marker.

Group	Pattern Markers
--------------	-----------------

Syntax `HSSerial:MARKer[n]:CLOCK:FREQUENCY <frequency>`
`HSSerial:MARKer[n]:CLOCK:FREQUENCY?`

Related Commands [HSSerial:MARKer\[n\]:CLOCK:TYPE](#)

Arguments `[n] ::= {1|2}` ("n" determines marker 1 or 2. If omitted, interpreted as 1.)
`<frequency> ::= <NR3> value.`

Returns A single <NR3> value.

Examples `HSSerial:MARKER1:CLOCK:FREQUENCY 1E9` sets the marker 1 clock frequency to 1 GHz.
`HSSerial:MARKER2:CLOCK:FREQUENCY?` might return `10.000000000E+6`, indicating that the marker 2 clock frequency is set to 10 MHz.

HSSerial:MARKer[n]:CLOCK:TYPE

This command sets or returns the marker clock type for the specified marker.

Group Pattern Markers

Syntax `HSSerial:MARKer[n]:CLOCK:TYPE`
`{D1Rate|D2Rate|D4Rate|D8Rate|DURate}`
`HSSerial:MARKer[n]:CLOCK:TYPE?`

Arguments `[n] ::= {1|2}` ("n" determines marker 1 or 2. If omitted, interpreted as 1.)
`D1Rate` sets the clock frequency to the data rate.
`D2Rate` sets the clock frequency to $\frac{1}{2}$ the data rate.
`D4Rate` sets the clock frequency to $\frac{1}{4}$ the data rate.
`D8Rate` sets the clock frequency to $\frac{1}{8}$ the data rate.
`DURate` set the clock frequency to a user defined rate.

Returns `D1R`
`D2R`
`D4R`
`D8R`
`DUR`

Examples `HSSERIAL:MARKER1:CLOCK:TYPE D2R` sets the marker 1 clock type to ½ the data rate.

`HSSERIAL:MARKER2:CLOCK:TYPE?` might return `DUR`, indicating that the marker 2 type is set to a user defined rate.

HSSerial:MARKer[n]:ENABLE

This command sets or returns the specified marker's state (enabled or disabled).

Group Pattern Markers

Syntax `HSSerial:MARKer[n]:ENABle {0|1|OFF|ON}`
`HSSerial:MARKer[n]:ENABle?`

Arguments `[n] ::= {1|2}` ("n" determines marker 1 or 2. If omitted, interpreted as 1.)
 OFF or 0 disables the specified marker. OFF or 0 is the default value.
 ON or 1 enables the specified marker.

Returns A single <Boolean> value.

Examples `HSSERIAL:MARKER1:ENABLE 1` enables marker 1.
`HSSERIAL:MARKER2:ENABLE?` might return 0 indicating that marker 2 is disabled.

HSSerial:MARKer[n]:HLOW:TYPE

This command sets or returns the High/Low type of the specified marker.

Group Pattern Markers

Syntax `HSSerial:MARKer[n]:HLOW:TYPE {AHIGH|ALOW|TRIGger}`
`HSSerial:MARKer[n]:HLOW:TYPE?`

Arguments `[n] ::= {1|2}` ("n" determines marker 1 or 2. If omitted, interpreted as 1.)
 AHIGH sets the markers to all high.
 ALOW sets the markers to all low.

TRIGger sets the marker to transition (High To Low or Low To High) at the set number of samples.

Returns AHIG: All Low
 ALOW: All High
 TRIG: Transitions on trigger

Examples HSSERIAL:MARKER1:HLOW:TYPE ALOW sets the marker 1 type to All Low.
 HSSERIAL:MARKER2:HLOW:TYPE? might return ALOW, indicating the marker 2 type is set to All Low.

HSSerial:MARKer[n]:TRIGger:LENGth

This command sets or returns the number of samples of the specified marker when the marker type is set to Trigger.

Group Pattern Markers

Syntax HSSerial:MARKer[n]:TRIGger:LENGth <samples>
 HSSerial:MARKer[n]:TRIGger:LENGth?

Arguments [n] ::= {1|2} ("n" determines marker 1 or 2. If omitted, interpreted as 1.)
 <samples>::=<NR2> value.

Returns A single <NR2> value.

Examples HSSERIAL:MARKER1:TRIGGER:LENGTH 15 sets the marker 1 trigger length is set to 15 samples.
 HSSERIAL:MARKER2:TRIGGER:LENGTH? might return 22.0000000000, indicating the marker 2 number of trigger samples is set to 22.

HSSerial:MARKer[n]:TRIGger:TYPE

This command sets or returns the marker High/Low trigger type setting for the specified marker.

Group Pattern Markers

Syntax	<code>HSSerial:MARKer[n]:TRIGger:TYPE {LTHigh HTLow}</code> <code>HSSerial:MARKer[n]:TRIGger:TYPE?</code>
Arguments	<code>[n] ::= {1 2}</code> ("n" determines marker 1 or 2. If omitted, interpreted as 1.) LTHigh sets the trigger type to Low to High HTLow sets the trigger type to High to Low
Returns	LTH HTL
Examples	<code>HSSERIAL:MARKER1:TRIGGER:TYPE LTHIGH</code> sets the marker 1 trigger type to Low To High. <code>HSSERIAL:MARKER2:TRIGGER:TYPE?</code> might return HTL, indicating the marker 2 trigger type is set to High To Low.

HSSerial:MARKer[n]:TYPE

This command sets or returns the marker type for the specified marker.

Group	Pattern Markers
Syntax	<code>HSSerial:MARKer[n]:TYPE {BPATtern CLOCK HILOw}</code> <code>HSSerial:MARKer[n]:TYPE?</code>
Arguments	BPATtern sets the marker type to match the Base Pattern. CLOCK sets the marker type to be a function of the Data Rate or a User Defined value. HILOw sets the marker type to High/Low Trigger.
Returns	BPAT CLOC HIL
Examples	<code>HSSERIAL:MARKER1:TYPE BPATTERN</code> sets the marker type to Base Pattern. <code>HSSERIAL:MARKER2:TYPE?</code> might return BPAT, indicating the marker 2 type is set to match the Base Pattern.

HSSerial:NOISe:BW

This command sets or returns the Single Tone Noise frequency setting.

Group Transmitter Noise

Syntax HSSerial:NOISe:BW <frequency>
HSSerial:NOISe:BW?

Related Commands [HSSerial:NOISe:TYPE](#)

Arguments <frequency>::=<NR3> value.

Returns A single <NR3> value.

Examples HSSerial:NOISe:BW 1E6 sets the Noise Single Tone Bandwidth to 1 MHz.
HSSerial:NOISe:BW? might return 10.000000000E+6, indicating that the Noise Single Tone Bandwidth is set to 10 MHz.

HSSerial:NOISe:ENABLE

This command sets or returns the Noise state (enabled or disabled).

Group Transmitter Noise

Syntax HSSerial:NOISe:ENABLE {0|1|OFF|ON}
HSSerial:NOISe:ENABLE?

Arguments OFF or 0 disables Noise. OFF or 0 is the default value.
ON or 1 enables Noise.

Returns A single <Boolean> value.

Examples HSSerial:NOISe:ENABLE 1 enables Transmitter Noise.
HSSerial:NOISe:ENABLE? might return 0, indicating that Transmitter Noise is disabled.

HSSerial:NOISe:LOCation

This command sets or returns the Transmitter Noise type.

Group	Transmitter Noise
Syntax	<code>HSSerial:NOISe:LOCation {NEAR FAR}</code> <code>HSSerial:NOISe:LOCation?</code>
Arguments	NEAR sets the noise location to the near-end of the transmission simulation. FAR sets the noise location to the far-end of the transmission simulation.
Returns	NEAR FAR
Examples	<code>HSSerial:NOISe:LOCATION NEAR</code> applies the noise before the transmitter simulation. <code>HSSerial:NOISe:LOCATION?</code> might return FAR, indicating the noise is applied after the transmitter simulation.

HSSerial:NOISe:TYPE

This command sets or returns the noise Frequency setting.

Group	Transmitter Noise
Syntax	<code>HSSerial:NOISe:TYPE {FULL SINGle}</code> <code>HSSerial:NOISe:TYPE?</code>
Related Commands	HSSerial:NOISe:BW
Arguments	FULL sets the tone to full bandwidth. SINGle sets the tone to a single tone and allows you to set the tone frequency.
Returns	FULL SING

Examples `HSSERIAL:NOISE:TYPE SINGLE` sets the noise to Single Tone.
 `HSSERIAL:NOISE:TYPE?` might return `FULL`, indicating that the noise type is set to Full Bandwidth.

HSSerial:NOISe:VALue

This command sets or returns the noise Magnitude.

Group Transmitter Noise

Syntax `HSSerial:NOISe:VALue <magnitude>`
 `HSSerial:NOISe:VALue?`

Arguments `<magnitude>::=<NR3>` value.

Returns A single `<NR3>` value.

Examples `HSSERIAL:NOISE:VALUE 4E-2` sets the noise Magnitude to 40 mV.
 `HSSERIAL:NOISE:VALUE?` might return `1.0000000000E-3`, indicating that the noise Magnitude is set to 1 mV.

HSSerial:PJITter:DEViation:MDEViation

This command sets or returns the percent of deviation when the Deviation for the Integer Cycles is set to Manual.

Group Transmitter Periodic Jitter

Syntax `HSSerial:PJITter:DEViation:MDEViation <percent>`
 `HSSerial:PJITter:DEViation:MDEViation?`

Arguments `<percent>::=<NR3>` value.
 Range: 0 to 50.

Returns A single `<NR1>` value.

Examples `HSSERIAL:PJITTER:DEVIATION:MDEVIATION 20` sets the deviation to 20% when the Integer Cycles Deviation type is set to Manual.

`HSSERIAL:PJITTER:DEVIATION:MDEVIATION?` might return 20, indicating the deviation is set to 20% when the Integer Cycles Deviation type is set to Manual.

HSSerial:PJITter:DEVIation:TYPE

This command sets or returns the Periodic Jitter Deviation type selected.

Group Transmitter Periodic Jitter

Syntax `HSSerial:PJITter:DEVIation:TYPE {AUTO|MANual}`
`HSSerial:PJITter:DEVIation:TYPE?`

Related Commands [HSSerial:PJITter:DEVIation:MDEVIation](#)

Arguments AUTO lets the software automatically deviation percentage.
 MANual allows the user to set the deviation percentage manually.

Returns AUTO
 MAN

Examples `HSSERIAL:PJITTER:DEVIATION:TYPE AUTO` sets the Periodic Jitter Deviation type to automatic.

`HSSERIAL:PJITTER:DEVIATION:TYPE?` might return MAN, indicating that the deviation type is set to manual.

HSSerial:PJITter:ICENable

This command sets or returns the Integer Cycles for Periodic Jitter state (enabled or disabled) when using the Transmitter settings.

Group Transmitter Periodic Jitter

Syntax `HSSerial:PJITter:ICENable {0|1|OFF|ON}`
`HSSerial:PJITter:ICENable?`

Arguments	OFF or 0 disables Integer Cycles. OFF or 0 is the default value. ON or 1 enables the Integer Cycles.
Returns	A single <Boolean> value.
Examples	<code>HSSerial:PJITTER:ICENABLE 0</code> disables integer cycles for Periodic Jitter when using the Transmitter tab. <code>HSSerial:PJITTER:ICENABLE?</code> might return 1, indicating that the integer cycles is enabled for Periodic Jitter when using the Transmitter tab.

HSSerial:PJ[n]:ENABLE

This command sets or returns the Transmitter Periodic Jitter state (enabled or disabled) for the specified Periodic Jitter.

Group	Transmitter Periodic Jitter
Syntax	<code>HSSerial:PJ[n]:ENABLE {0 1 OFF ON}</code>
Arguments	<code>[n] ::= {1 2 3 4}</code> ("n" determines PJ1, PJ2, PJ3, and PJ4. If omitted, interpreted as 1.) OFF or 0 disables the selected Periodic Jitter. OFF or 0 is the default value. ON or 1 enables the selected Periodic Jitter.
Returns	A single <Boolean> value.
Examples	<code>HSSerial:PJ1:ENABLE 1</code> enables PJ1. <code>HSSerial:PJ2:ENABLE?</code> might return 0, indicating that PJ2 is disabled.

HSSerial:PJ[n]:FREQuency

This command sets or returns the frequency of the specified Transmitter Periodic Jitter.

Group	Transmitter Periodic Jitter
--------------	-----------------------------

Syntax	<code>HSSerial:PJ[n]:FREQUENCY <frequency></code> <code>HSSerial:PJ[n]:FREQUENCY?</code>
Arguments	<code>[n] ::= {1 2 3 4}</code> ("n" determines PJ1, PJ2, PJ3, and PJ4. If omitted, interpreted as 1.) <code><frequency> ::= <NR3> value.</code>
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:PJ1:FREQUENCY 30E3</code> sets PJ1 frequency to 30 kHz. <code>HSSERIAL:PJ2:FREQUENCY?</code> might return <code>20.0000000000E+3</code> , indicating the PJ2 frequency is 20 kHz.

HSSerial:PJ[n]:MAGNitude

This command sets or returns the Magnitude (pk-pk) of the specified Transmitter Periodic Jitter.

Conditions	The default units is UI (Unit Intervals) and is typically how this control is defined. Optionally, the units can be set as seconds. When setting the value, you must send values appropriate for the user interface setting. Units is set directly from the user interface for this value. See the examples for using the set and query forms as either UI or seconds.
Group	Transmitter Periodic Jitter
Syntax	<code>HSSerial:PJ[n]:MAGNitude <magnitude></code> <code>HSSerial:PJ[n]:MAGNitude?</code>
Arguments	<code>[n] ::= {1 2 3 4}</code> ("n" determines PJ1, PJ2, PJ3, and PJ4. If omitted, interpreted as 1.) <code><magnitude> ::= <NR3> value.</code> Range in UI: 0.001 to 50 Range in seconds: 1 ps to 50 ns
Returns	A single <code><NR3></code> value.

Examples Unit Intervals

`HSSERIAL:PJ1:MAGNITUDE 0.5` sets the PJ1 Magnitude to 0.5 UI.

`HSSERIAL:PJ2:MAGNITUDE?` might return `10.0000000000`, indicating that the PJ2 Magnitude is set to 10 UI.

Seconds

`HSSERIAL:PJ1:MAGNITUDE 8E-9` sets the PJ1 Magnitude to 8 ns_{p-p}.

`HSSERIAL:PJ2:MAGNITUDE?` might return `100.0000000000E-12`, indicating that the PJ2 Magnitude is set to 100 ps_{p-p}.

HSSerial:PJ[n]:PHASe

This command sets or returns the Phase of the specified Transmitter Periodic Jitter.

Group Transmitter Periodic Jitter

Syntax `HSSerial:PJ[n]:PHASe <phase>`
`HSSerial:PJ[n]:PHASe?`

Arguments `[n] ::= {1|2|3|4}` ("n" determines PJ1, PJ2, PJ3, and PJ4. If omitted, interpreted as 1.)
`<phase> ::= <NR1> value.`

Returns A single <NR1> value.

Examples `HSSERIAL:PJ1:PHASe 30` sets the PJ1 Phase to 30 degrees.
`HSSERIAL:PJ2:PHASe?` might return `11.0000000000`, indicating that the PJ2 Phase is set to 11 degrees.

HSSerial:PREemphasis:COPReemp

This command sets or returns the Transmitter Pre/De-emphasis coefficients.

Group Transmitter Pre/De-emphasis

Syntax `HSSerial:PREemphasis:COPReemp <row>,<column>,<value>`
`HSSerial:PREemphasis:COPReemp? <row>,<column>`

Arguments	<row>::=<NR1> representing the row of the coefficient and can range from 1 to 5. Limited to 1 if Preshoot is enabled. <column>::=<NR1> representing the column of the coefficient and can range from 1 to 6. <value>::=<NR3> representing the value of the coefficient. dB Range: -20 to +20. Linear Scale Range: 0.1 to 10.
Returns	A single <NR1> value when Units for Tap Coefficients is set to dB Scale. A single <NR3> value when Units for Tap Coefficients is set to Linear Scale.
Examples	HSSERIAL:PREEMPHASIS:COPREEMP 1,2,0.3 sets the Pre/De-emphasis tap coefficient of row 1 and column 2 to 0.3 (Linear Scale). HSSERIAL:PREEMPHASIS:COPREEMP? 1,2 might return 4.0000000000, indicating the Pre/De-emphasis tap for row 1 and column 2 is 4 dB (dB Scale).

HSSerial:PREemphasis:DPreemph (No Query Form)

This command deletes last active Pre/De-emphasis tap.

Group	Transmitter Pre/De-emphasis
Syntax	HSSerial:PREemphasis:DPreemph
Examples	HSSERIAL:PREEMPHASIS:DPREEMPH deletes the last active tap.

HSSerial:PREemphasis:ENABLE

This command sets or returns the Transmitter Pre/De-Emphasis state (enabled or disabled).

Group	Transmitter Pre/De-emphasis
Syntax	HSSerial:PREemphasis:ENABLE {0 1 OFF ON} HSSerial:PREemphasis:ENABLE?

Arguments	OFF or 0 disables Pre/De-Emphasis. OFF or 0 is the default value. ON or 1 enables Pre/De-Emphasis.
Returns	A single <Boolean> value.
Examples	<code>HSSerial:PREEMPHASIS:ENABLE 1</code> enables Pre/De-Emphasis. <code>HSSerial:PREEMPHASIS:ENABLE?</code> might return 0, indicating that Pre/De-Emphasis is disabled.

HSSerial:PREemphasis:TPReemph

The set form of this command adds a new Pre/De-Emphasis tap. The query form of this command returns the number of active Pre/De-Emphasis taps.

Group	Transmitter Pre/De-emphasis
Syntax	<code>HSSerial:PREemphasis:TPReemph</code> <code>HSSerial:PREemphasis:TPReemph?</code>
Returns	The number of active taps.
Examples	<code>HSSerial:PREEMPHASIS:TPREEMPH</code> adds a new Pre/De-Emphasis tap. <code>HSSerial:PREEMPHASIS:TPREEMPH?</code> might return 2, indicating the number of active Pre/De-Emphasis taps is 2.

HSSerial:PREemphasis:TYPE

This command sets or returns the Pre/De-Emphasis type.

Group	Transmitter Pre/De-emphasis
Syntax	<code>HSSerial:PREemphasis:TYPE {UIConstant UILinear FRActional}</code> <code>HSSerial:PREemphasis:TYPE?</code>
Arguments	UIConstant sets the pre-emphasis type to Constant UI. UILinear sets the pre-emphasis type to Linear UI. FRActional sets the pre-emphasis type to Fractional.

Returns UIC
UIL
FRAC

Examples HSSERIAL:PREEMPHASIS:TYPE UIConstant sets the Pre/De-Emphasis type to UIConstant.

HSSERIAL:PREEMPHASIS:TYPE? might return UIC, indicating the Pre/De-emphasis type is set to Constant UI.

HSSerial:PREemphasis:UNIT

This command sets or returns the Pre/De-Emphasis units.

Group Transmitter Pre/De-emphasis

Syntax HSSerial:PREemphasis:UNIT {DB|LINEar}
HSSerial:PREemphasis:UNIT?

Arguments DB sets the units scale to dB.
LINEar sets the units scale to linear.

Returns DB
LINE

Examples HSSERIAL:PREEMPHASIS:UNIT DB sets the Pre/De-Emphasis units to dB Scale.

HSSERIAL:PREEMPHASIS:UNIT? might return LINE, indicating the Pre/De-Emphasis units are set to Linear Scale.

HSSerial:PREShoot:COPShoot

This command sets or returns the Transmitter Preshoot coefficients.

Group Transmitter Pre/De-emphasis

Syntax HSSerial:PREShoot:COPShoot <row>,<column>,<value>
HSSerial:PREShoot:COPShoot? <row>,<column>

Arguments	<row>::=<NR1> representing the row of the coefficient and can only be 1. <column>::=<NR1> representing the column of the coefficient and can range from 1 to 6. <value>::=<NR3> representing the value of the coefficient. dB Range: -20 to +20. Linear Scale Range: 0.1 to 10.
Returns	A single <NR1> value when Units for Tap Coefficients is set to dB Scale. A single <NR3> value when Units for Tap Coefficients is set to Linear Scale.
Examples	HSSERIAL:PRESHOOT:COPSHOOT 1,2,0.3 sets the Preshoot tap coefficient of row 1 and column 2 to 0.3 (Linear Scale). HSSERIAL:PRESHOOT:COPSHOOT? 1,2 might return 4.0000000000, indicating the Preshoot tap for row 1 and column 2 is 4 dB (dB Scale).

HSSerial:PREShoot:ENABLE

This command sets or returns the Transmitter Preshoot state (enabled or disabled).

Group	Transmitter Pre/De-emphasis
Syntax	HSSerial:PREShoot:ENABLE {0 1 OFF ON} HSSerial:PREShoot:ENABLE?
Arguments	OFF or 0 disables Preshoot. OFF or 0 is the default value. ON or 1 enables Preshoot.
Returns	A single <Boolean> value.
Examples	HSSERIAL:PRESHOOT:ENABLE 1 enables Preshoot. HSSERIAL:PRESHOOT:ENABLE? might return 0, indicating that Preshoot is disabled.

HSSerial:PREShoot:TYPE

This command sets or returns the Preshoot type.

Group	Transmitter Pre/De-emphasis
Syntax	<code>HSSerial:PREShoot:TYPE {UIConstant UILinear FRActional}</code> <code>HSSerial:PREShoot:TYPE?</code>
Arguments	UIConstant indicates UI-Constant. UILinear indicates UI-Linear. FRActional indicates Fractional.
Returns	UIC UIL FRAC
Examples	<code>HSSERIAL:PRESHOOT:TYPE UIConstant</code> sets the Preshoot type to UIConstant. <code>HSSERIAL:PRESHOOT:TYPE?</code> might return UIC, indicating the Preshoot type is set to Constant UI.

HSSerial:RESet (No Query Form)

This command resets the High Speed Serial application plug-in to its default values.

Conditions	The active plug-in must be High Speed Serial.
Group	Control
Syntax	<code>HSSerial:RESet</code>
Related Commands	WPLugin:ACTive
Examples	<code>HSSERIAL:RESET</code> returns the High Speed Serial plug-in to its default values.

HSSerial:RJ1Cfactor:ENABLE

This command sets or returns the Random Jitter RJ1 Crest Factor enable state (enabled or disabled).

Group	Transmitter Random Jitter
Syntax	<code>HSSerial:RJ1Cfactor:ENABLe {0 1 OFF ON}</code> <code>HSSerial:RJ1Cfactor:ENABLe?</code>
Arguments	OFF or 0 disables RJ1 Crest Factor. OFF or 0 is the default value. ON or 1 enables RJ1 Crest Factor.
Returns	A single <Boolean> value.
Examples	<code>HSSERIAL:RJ1CFACTOR:ENABLE 1</code> enables the RJ1 Crest Factor. <code>HSSERIAL:RJ1CFACTOR:ENABLE?</code> might return 0, indicating that the RJ1 Crest Factor is disabled.

HSSerial:RJ1Cfactor:TISigma

This command sets or returns the RJ1 Crest Factor value.

Group	Transmitter Random Jitter
Syntax	<code>HSSerial:RJ1Cfactor:TISigma <crest_factor></code> <code>HSSerial:RJ1Cfactor:TISigma?</code>
Arguments	<crest_factor>::=<NR2> value.
Returns	A single <NR2> value.
Examples	<code>HSSERIAL:RJ1CFACTOR:TISIGMA 12.3</code> sets the RJ1 Crest Factor to 12.3. <code>HSSERIAL:RJ1CFACTOR:TISIGMA?</code> might return 12.3, indicating that the RJ1 Crest Factor is set to 12.3.

HSSerial:RJ[n]:ENABLe

This command sets or returns the Transmitter Random Jitter state (enabled or disabled) for the specified Random Jitter .

Group	Transmitter Random Jitter
--------------	---------------------------

Syntax	<code>HSSerial:RJ[n]:ENABle {0 1 OFF ON}</code> <code>HSSerial:RJ[n]:ENABle?</code>
Arguments	<code>[n] ::= {1 2 3}</code> ("n" determines RJ1, RJ2, and RJ3. If omitted, interpreted as 1.) OFF or 0 disables Random Jitter. OFF or 0 is the default value. ON or 1 enables Random Jitter.
Returns	A single <Boolean> value.
Examples	<code>HSSERIAL:RJ1:ENABLE 1</code> enables Random Jitter RJ1. <code>HSSERIAL:RJ2:ENABLE?</code> might return 0, indicating that Random Jitter RJ2 is disabled.

HSSerial:RJ[n]:FREQuency:END

This command sets or returns the High Frequency value of the specified Transmitter Random Jitter.

Group	Transmitter Random Jitter
Syntax	<code>HSSerial:RJ[n]:FREQuency:END <high-frequency></code> <code>HSSerial:RJ[n]:FREQuency:END?</code>
Arguments	<code>[n] ::= {1 2 3}</code> ("n" determines RJ1, RJ2, and RJ3. If omitted, interpreted as 1.) <high-frequency>::=<NR3> value.
Returns	A single <NR3> value.
Examples	<code>HSSerial:RJ1:FREQuency:END 10E5</code> sets the RJ1 High Frequency value to 1 MHz. <code>HSSERIAL:RJ2:FREQUENCY:END?</code> might return <code>250.0000000000E+6</code> , indicating that the RJ2 High Frequency value is set to 250 MHz.

HSSerial:RJ[n]:FREQuency:STARt

This command sets or returns the Low Frequency value of the specified Transmitter Random Jitter.

Group	Transmitter Random Jitter
Syntax	<pre> HSSerial:RJ[n]:FREQUENCY:START <low-frequency> HSSerial:RJ[n]:FREQUENCY:START ? </pre>
Arguments	[n] ::= {1 2 3} ("n" determines RJ1, RJ2, and RJ3. If omitted, interpreted as 1.) <low-frequency> ::= <NR3> value.
Returns	A single <NR3> value.
Examples	HSSERIAL:RJ1:FREQUENCY:START 10E5 sets the RJ1 Low Frequency value to 1 MHz. HSSERIAL:RJ2:FREQUENCY:START? might return 25.000000000E+6, indicating that the RJ2 Low Frequency value is set to 25 MHz.

HSSerial:RJ[n]:MAGNitude

This command sets or returns the Magnitude (RMS) of the specified Transmitter Random Jitter.

Conditions	The default units is UI (Unit Intervals) and is typically how this control is defined. Optionally, the units can be set as seconds. When setting the value, you must send values appropriate for the user interface setting. Units is set directly from the user interface for this value. See the examples for using the set and query forms as either UI or seconds.
Group	Transmitter Random Jitter
Syntax	<pre> HSSerial:RJ[n]:MAGNitude <magnitude> HSSerial:RJ[n]:MAGNitude </pre>
Arguments	[n] ::= {1 2 3} ("n" determines RJ1, RJ2, and RJ3. If omitted, interpreted as 1.) <magnitude> ::= <NR3> value. Range in UI: 0.001 to 0.5 Range in seconds: 1 ps to 500 ps
Returns	A single <NR3> value.

Examples Unit Intervals

HSSERIAL:RJ1:MAGNITUDE 0.35 sets the RJ1 Magnitude to 0.35 UI.

HSSERIAL:RJ2:MAGNITUDE? might return 100.0000000000E-3, indicating that the RJ2 Magnitude is set to 0.1 UI.

Seconds

HSSERIAL:RJ1:MAGNITUDE 50E-12 sets the RJ1 Magnitude to 50 ps.

HSSERIAL:RJ2:MAGNITUDE? might return 10.0000000000E-12, indicating that the RJ2 Magnitude is set to 10 ps.

HSSerial:RSEed:ENABLE

This command sets or returns the Random Jitter RJ1 Seed state (enabled or disabled).

Group Transmitter Random Jitter

Syntax HSSerial:RSEed:ENABLE {0|1|OFF|ON}
HSSerial:RSEed:ENABLE?

Arguments OFF or 0 disables RJ1 Seed. OFF or 0 is the default value.
ON or 1 enables RJ1 Seed.

Returns A single <Boolean> value.

Examples HSSERIAL:RSEED:ENABLE 1 enables the RJ1 Seed.
HSSERIAL:RSEED:ENABLE? might return 0, indicating that the RJ1 Seed is disabled.

HSSerial:RSEed:VALue

This command sets or returns the RJ1 Seed value.

Group Transmitter Random Jitter

Syntax HSSerial:RSEed:VALue <seed>
HSSerial:RSEed:VALue?

Arguments	<seed>::=<NR1> value.
Returns	A single <NR3> value.
Examples	HSSERIAL:RSEED:VALUE 1234 sets the RJ1 Seed value to 1234. HSSERIAL:RSEED:VALUE? might return 12.3450000000E+3, indicating that the RJ1 Seed value is set to 1234.

HSSerial:SCRamble:ENABLE

This command sets or returns the Scrambling enable state (enabled or disabled).

Group	Pattern Scrambling
Syntax	HSSerial:SCRamble:ENABLE {0 1 OFF ON} HSSerial:SCRamble:ENABLE?
Arguments	OFF or 0 disables Scrambling. OFF or 0 is the default value. ON or 1 enables Scrambling.
Returns	A single <Boolean> value.
Examples	HSSERIAL:SCRAMBLE:ENABLE 1 enables the Scrambling. HSSERIAL:SCRAMBLE:ENABLE? might return 0, indicating that Scrambling is disabled.

HSSerial:SCRamble:POLYnomial

This commands sets or returns the Scrambling Polynomial value.

Group	Pattern Scrambling
Syntax	HSSerial:SCRamble:POLYnomial <polynomial> HSSerial:SCRamble:POLYnomial?
Arguments	<polynomial>::=<string> defines the scrambling polynomial.

Returns A single <polynomial> string.

Examples `HSSerial:SCRAMBLE:POLYNOMIAL "X7+X5+X4+1"` sets the Scrambling Polynomial to $X^7+X^5+X^4+1$.
`HSSerial:SCRAMBLE:POLYNOMIAL?` might return `"X16+X5+X4+X3+1"`.

HSSerial:SCRAMble:RINit

This commands sets or returns the Register Initial Value.

Group Pattern Scrambling

Syntax `HSSerial:SCRAMble:RINit <register_value>`
`HSSerial:SCRAMble:RINit?`

Arguments <register_value>::=<string> defines the initial value of the register.

Returns A single <register_value> string.

Examples `HSSerial:SCRAMBLE:RINIT "11101011"` sets the Register Initial Value to 11101011.
`HSSerial:SCRAMBLE:RINIT?` might return `"11101011"`

HSSerial:SCRAMble:RTYPE

This commands sets or returns the Register Initial Value Type.

Group Pattern Scrambling

Syntax `HSSerial:SCRAMble:RTYPE {BINary|HEX}`
`HSSerial:SCRAMble:RTYPE?`

Arguments BINary sets the Register Initial Value type to binary.
 HEX sets the Register Initial Value type to hexadecimal.

Returns BIN (Binary)
 HEX (Hexadecimal)

Examples `HSSerial:SCRAMBLE:RTYPE BIN` sets the Register Initial Value type to Binary.
`HSSerial:SCRAMBLE:RTYPE?` might return HEX, indicating the Register Initial Value type is set to Hexadecimal.

HSSerial:SCRamble:TYPE

This command sets or returns the Scrambling Type.

Group Pattern Scrambling

Syntax `HSSerial:SCRamble:TYPE {ADDitive|MULTiplicative}`
 `HSSerial:SCRamble:TYPE?`

Arguments ADDitive scrambles the input data pattern using a additive polynomial.
 MULTiplicative scrambles the input data pattern using a multiplying polynomial.

Returns ADD
 MULT

Examples `HSSerial:SCRAMBLE:TYPE ADDITIVE` sets the Scrambling Type to Additive.
`HSSerial:SCRAMBLE:TYPE?` might return MULT, indicating the Scrambling Type is set to Multiplicative.

HSSerial:SRESponse:ENABLE

This command sets or returns the Step Response state (enabled or disabled).

Group Pattern Step Response

Syntax `HSSerial:SRESponse:ENABLE {0|1|OFF|ON}`
 `HSSerial:SRESponse:ENABLE`

Arguments OFF or 0 disables Step Response. OFF or 0 is the default value.
 ON or 1 enables Step Response.

Returns A single <Boolean> value.

Examples `HSSerial:SRESPONSE:ENABLE 0` disables Step Response.
`HSSerial:SRESPONSE:ENABLE?` might return 1, indicating that Step Response is enabled.

HSSerial:SRESPONSE:FTIME

This command sets or returns the Step Response Fall Time value.

Group Pattern Step Response

Syntax `HSSerial:SRESPONSE:FTIME <fall_time>`
`HSSerial:SRESPONSE:FTIME?`

Related Commands [HSSerial:SRESPONSE:FTYPE](#)

Arguments <fall_time> ::= <NR3> value.

Returns A single <NR3> value.

Examples `HSSerial:SRESPONSE:FTIME 80E12` sets the Step Response Fall Time to 80 ps.
`HSSerial:SRESPONSE:FTIME?` might return 50.0000000000E-12, indicating that the Step Response Fall Time is 50 ps.

HSSerial:SRESPONSE:FTYPE

This command sets or returns the Step Response Fall Time Type.

Group Pattern Step Response

Syntax `HSSerial:SRESPONSE:FTYPE {FORDER|LINEAR}`
`HSSerial:SRESPONSE:FTYPE?`

Arguments FORDER is the First Order type.
 LINEAR is the linear type.

Returns FORD
LIN

Examples HSSERIAL:SRESPONSE:FTYPE LINEar sets the Step Response Fall Time Type to Linear.

HSSERIAL:SRESPONSE:FTYPE? might return FORD, indicating the Step Response Fall Time Type is set to First Order.

HSSerial:SRESPonse:RFTYPE

This command sets or returns the Rise/Fall Time type (10/90 or 20/80).

Group Pattern Step Response

Syntax HSSerial:SRESPonse:RFTYPE {TWENTyeighty|TENNinety}
HSSerial:SRESPonse:RFTYPE?

Arguments TWENTyeighty is 20/80.
TENNinety is 10/90.

Returns TWEN
TENN

Examples HSSERIAL:SRESPONSE:RFTYPE TENNinety sets the Rise/Fall Time type to 10/90.

HSSERIAL:SRESPONSE:RFTYPE? might return TENNinety, indicating the Rise/Fall Time type is set to 10/90.

HSSerial:SRESPonse:RTIME

This command sets or returns the Step Response Rise Time value.

Group Pattern Step Response

Syntax HSSerial:SRESPonse:RTIME <rise_time>
HSSerial:SRESPonse:RTIME?

Related Commands [HSSerial:SResponse:FType](#)

Arguments <rise_time>::= <NR3> value.

Returns A single <NR3> value.

Examples `HSSERIAL:SRESPONSE:RTIME 80E12` sets the Step Response Rise Time to 80 ps.

`HSSERIAL:SRESPONSE:RTIME?` might return `50.0000000000E-12`, indicating that the Step Response Rise Time is 50 ps.

HSSerial:SResponse:RTYPE

This command sets or returns the Step Response Rise Time Type.

Group Pattern Step Response

Syntax `HSSerial:SResponse:RTYPE {FORDER|LINEar}`
`HSSerial:SResponse:RTYPE?`

Arguments FORDER is the First Order type.
LINEar is the linear type.

Returns FORD
LIN

Examples `HSSERIAL:SRESPONSE:RTYPE LINEar` sets the Step Response Rise Time Type to Linear.

`HSSERIAL:SRESPONSE:RTYPE?` might return `FORD`, indicating the Step Response Rise Time Type is set to First Order.

HSSerial:SSC:DFDT

This command sets or returns the Error Option df/dt value of Spread Spectrum Clocking.

Group Transmitter Spread Spectrum Clocking

Syntax	<code>HSSerial:SSC:DFDT <df/dt></code> <code>HSSerial:SSC:DFDT?</code>
Arguments	<code><df/dt>::=<NR3></code> value.
Returns	A single <code><NR3></code> value.
Examples	<code>HSSerial:SSC:DFDT 2E2</code> sets the df/dt value to 200 kppm/μs. <code>HSSerial:SSC:DFDT?</code> might return <code>1.0000000000E+3</code> , indicating that the Error Option df/dt is set to 1 kppm/μs.

HSSerial:SSC:DFDT:DURation

This command sets or returns the Error Option Minimum Duration value of Spread Spectrum Clocking.

Group	Transmitter Spread Spectrum Clocking
Syntax	<code>HSSerial:SSC:DFDT:DURation <duration></code> <code>HSSerial:SSC:DFDT:DURation?</code>
Arguments	<code><duration>::=<NR3></code> value.
Returns	A single <code><NR3></code> value.
Examples	<code>HSSerial:SSC:DFDT:DURATION 2E-6</code> sets the Error Option Duration to 2 μs. <code>HSSerial:SSC:DFDT:DURATION?</code> might return <code>1.0000000000E-6</code> , indicating the Error Option Duration is 1 μs.

HSSerial:SSC:DFDT:LOCation

This command sets or returns the Error Option Location value of Spread Spectrum Clocking.

Group	Transmitter Spread Spectrum Clocking
--------------	--------------------------------------

Syntax	<code>HSSerial:SSC:DFDT:LOCation <location></code> <code>HSSerial:SSC:DFDT:LOCation?</code>
Arguments	<code><location>::=<NR3></code> value.
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:SSC:DFDT:LOCATION 60</code> sets the Error Option Location value to 60%. <code>HSSERIAL:SSC:DFDT:LOCATION?</code> might return <code>20.0000000000</code>, indicating the Error Option Location is set to 20%.

HSSerial:SSC:ENABLE

This command sets or returns the Spread Spectrum Clocking state (enabled or disabled).

Group	Transmitter Spread Spectrum Clocking
Syntax	<code>HSSerial:SSC:ENABle {0 1 OFF ON}</code> <code>HSSerial:SSC:ENABle?</code>
Arguments	OFF or 0 disables Spread Spectrum Clocking. OFF or 0 is the default value. ON or 1 enables Spread Spectrum Clocking.
Returns	A single <code><Boolean></code> value.
Examples	<code>HSSERIAL:SSC:ENABLE 1</code> enables Spread Spectrum Clocking. <code>HSSERIAL:SSC:ENABLE?</code> might return <code>0</code>, indicating that Spread Spectrum Clocking is disabled.

HSSerial:SSC:EOPtion:ENABLE

This command sets or returns the Error Option state (enabled or disabled) in Spread Spectrum Clocking.

Group	Transmitter Spread Spectrum Clocking
--------------	--------------------------------------

Syntax	<code>HSSerial:SSC:EOPtion:ENABle {0 1 OFF ON}</code> <code>HSSerial:SSC:EOPtion:ENABle?</code>
Arguments	OFF or 0 disables Error Option. OFF or 0 is the default value. ON or 1 enables Error Option.
Returns	A single <Boolean> value.
Examples	<code>HSSERIAL:SSC:EOPtion:ENABle 1</code> enables the Error Option for SSC. <code>HSSERIAL:SSC:EOPtion:ENABle?</code> might return 0, indicating that Error Option is disabled for SSC.

HSSerial:SSC:FREQuency:DEViation

This command sets or returns the Frequency Deviation value of Spread Spectrum Clocking.

Group	Transmitter Spread Spectrum Clocking
Syntax	<code>HSSerial:SSC:FREQuency:DEViation <deviation></code> <code>HSSerial:SSC:FREQuency:DEViation?</code>
Arguments	<deviation>::=<NR3> value.
Returns	A single <NR3> value.
Examples	<code>HSSERIAL:SSC:FREQUENCY:DEVIATION 4E3</code> sets the SSC Frequency Deviation to 4 kppm. <code>HSSERIAL:SSC:FREQUENCY:DEVIATION?</code> might return <code>5.000000000E+3</code> , indicating the SSC Frequency Deviation is set to 5 kppm.

HSSerial:SSC:FREQuency:MODulation

This command sets or returns the Frequency Modulation value of Spread Spectrum Clocking.

Group	Transmitter Spread Spectrum Clocking
--------------	--------------------------------------

Syntax	<code>HSSerial:SSC:FREQUENCY:MODulation <modulation></code> <code>HSSerial:SSC:FREQUENCY:MODulation?</code>
Arguments	<code><modulation>::=<NR3></code> value.
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:SSC:FREQUENCY:MODULATION 45E3</code> sets the SSC Frequency Modulation to 45 kHz. <code>HSSERIAL:SSC:FREQUENCY:MODULATION?</code> might return <code>33.0000000000E+3</code>, indicating that the SSC Frequency Modulation is set to 33 kHz.

HSSerial:SSC:SHAPE

This command sets or returns the Shape profile of Spread Spectrum Clocking.

Group	Transmitter Spread Spectrum Clocking
Syntax	<code>HSSerial:SSC:SHAPE {SINE TRIangle}</code> <code>HSSerial:SSC:SHAPE?</code>
Arguments	<code>SINE</code> sets the shape to sinusoidal. <code>TRIangle</code> sets the shape to triangular.
Returns	<code>SINE</code> <code>TRI</code>
Examples	<code>HSSERIAL:SSC:SHAPE SINE</code> sets the shape of the SSC profile to sinusoidal. <code>HSSERIAL:SSC:SHAPE?</code> might return <code>TRI</code>, indicating the shape of the SSC profile is set to triangular.

HSSerial:SSC:SPRead

This command sets or returns the Spread of the Spread Spectrum Clocking profile.

Group	Transmitter Spread Spectrum Clocking
--------------	--------------------------------------

Syntax	<code>HSSerial:SSC:SPRead {UP DOWN CENTer UNEQual}</code> <code>HSSerial:SSC:SPRead?</code>
Arguments	UP sets the spread to UP. DOWN sets the spread to DOWN. CENTer sets the spread to CENTER. UNEQual sets the spread to UNEQUAL.
Returns	UP DOWN CENT UNEQ
Examples	<code>HSSERIAL:SSC:SPREAD UNEQUAL</code> sets the SSC profile Spread to unequal. <code>HSSERIAL:SSC:SPREAD?</code> might return <code>CENT</code> , indicating that the SSC profile Spread is set to center.

HSSerial:SSC:USPRead:PERCentage

This command sets or returns the Downward or Upward Percentage of the Unequal Spread of Spread Spectrum Clocking.

Group	Transmitter Spread Spectrum Clocking
Syntax	<code>HSSerial:SSC:USPRead:PERCentage <percentage></code> <code>HSSerial:SSC:USPRead:PERCentage?</code>
Arguments	<code><percentage>::=<NR3></code> value.
Returns	A single <code><NR3></code> value.
Examples	<code>HSSERIAL:SSC:USPREAD:PERCENTAGE 50</code> sets the SSC Unequal Spread Percentage to 50%. <code>HSSERIAL:SSC:USPREAD:PERCENTAGE?</code> might return <code>25.0000000000</code> , indicating the SSC Unequal Spread Percentage is set to 50%.

WPLugin:ACTive

This command sets or returns the active waveform plug-in. To use the High Speed Serial commands in this document, the active waveform plug-in must be set to High Speed Serial.

Group Control

Syntax WPLugin:ACTive <plug-in_name>
WPLugin:ACTive?

Arguments <plug-in_name>::=<string> defines the name of the active waveform plug-in.
“High Speed Serial” is the proper string to activate the High Speed Serial plug-in.

Returns A single <plug-in_name> string.

Examples WPLUGIN:ACTIVE "High Speed Serial" sets the High Speed Serial plug-in as the active plug-in.

WPLUGIN:ACTIVE? might return "High Speed Serial", indicating High Speed Serial is currently the active waveform plug-in.

Index

C

Command Groups, 2-3

H

HSSerial:AMPLitude:MAXimum, 2-15

HSSerial:AMPLitude:MINimum, 2-15

HSSerial:BCOMpile:PJ[n]:ENABLE, 2-17

HSSerial:BCOMpile:PJ[n]:FREQuency:HIGH, 2-18

HSSerial:BCOMpile:PJ[n]:FREQuency:
INCRement, 2-18

HSSerial:BCOMpile:PJ[n]:FREQuency:LOW, 2-19

HSSerial:BCOMpile:PJ[n]:MAGNitude:END, 2-20

HSSerial:BCOMpile:PJ[n]:MAGNitude:
INCRement, 2-20

HSSerial:BCOMpile:PJ[n]:MAGNitude:START, 2-21

HSSerial:BCOMpile:PJTter:DEViation:
MDEViation, 2-16

HSSerial:BCOMpile:PJTter:DEViation:TYPE, 2-16

HSSerial:BCOMpile:PJTter:ICENable, 2-17

HSSerial:BCOMpile:RJ1:ENABLE, 2-21

HSSerial:BCOMpile:RJ1:FREQuency:HIGH, 2-22

HSSerial:BCOMpile:RJ1:FREQuency:LOW, 2-22

HSSerial:BCOMpile:RJ1:MAGNitude:END, 2-23

HSSerial:BCOMpile:RJ1:MAGNitude:
INCRement, 2-23

HSSerial:BCOMpile:RJ1:MAGNitude:START, 2-24

HSSerial:BDATa, 2-24

HSSerial:BDATa:FILE, 2-26

HSSerial:BDATa:INVert, 2-25

HSSerial:BDATa:PATtern, 2-26

HSSerial:BDATa:PATtern:TYPE, 2-27

HSSerial:BDATa:PRBS, 2-28

HSSerial:BDATa:PRBS:UDEfined:
POLYnomial, 2-28

HSSerial:BDATa:PRBS:UDEfined:SREGister, 2-29

HSSerial:CHANnel:ENABLE, 2-29

HSSerial:CHANnel:ISI:BW, 2-30

HSSerial:CHANnel:ISI:TYPE, 2-30

HSSerial:CHANnel:ISI:VALue, 2-31

HSSerial:CHANnel:SPARameter:BANDwidth, 2-32

HSSerial:CHANnel:SPARameter:BANDwidth:
AUTO, 2-33

HSSerial:CHANnel:SPARameter:CASCading:
AGGRessor[n]:AMPLitude, 2-34

HSSerial:CHANnel:SPARameter:CASCading:
AGGRessor[n]:CTALk, 2-35

HSSerial:CHANnel:SPARameter:CASCading:
AGGRessor[n]:DRATe, 2-36

HSSerial:CHANnel:SPARameter:CASCading:
AGGRessor[n]:SIGNal, 2-37

HSSerial:CHANnel:SPARameter:CASCading:
AGGRessor[n]:SIGNal:FILE, 2-38

HSSerial:CHANnel:SPARameter:CASCading:
AGGRessor[n]:SIGNal:PRBS, 2-39

HSSerial:CHANnel:SPARameter:CASCading:
AGGRessor2[:ENABLE], 2-33

HSSerial:CHANnel:SPARameter:CASCading:
DEEMbed, 2-40

HSSerial:CHANnel:SPARameter:CASCading:
STAGe[m]:DRX[n], 2-40

HSSerial:CHANnel:SPARameter:CASCading:
STAGe[m]:DTX[n], 2-42

HSSerial:CHANnel:SPARameter:CASCading:
STAGe[m]:ENABLE, 2-45

HSSerial:CHANnel:SPARameter:CASCading:
STAGe[m]:FILE, 2-46

HSSerial:CHANnel:SPARameter:CASCading:
STAGe[m]:RX[n], 2-43

HSSerial:CHANnel:SPARameter:CASCading:
STAGe[m]:SSCHeme, 2-47

HSSerial:CHANnel:SPARameter:CASCading:
STAGe[m]:TX[n], 2-44

HSSerial:CHANnel:SPARameter:CASCading:
STYPE, 2-48

HSSerial:CHANnel:SPARameter:CASCading:
TYPE, 2-49

HSSerial:CHANnel:SPARameter:MODE, 2-49

HSSerial:CHANnel:SPARameter:NCAScading:
AGGRessor[n]:AMPLitude, 2-51

HSSerial:CHANnel:SPARameter:NCAScading:
AGGRessor[n]:CTALk, 2-51

HSSerial:CHANnel:SPARameter:NCAScading:
AGGRessor[n]:DRATe, 2-52

HSSerial:CHANnel:SPARameter:NCAScading:
AGGRessor[n]:SIGNal, 2-53

HSSerial:CHANnel:SPARameter:NCAScading:
AGGRessor[n]:SIGNal:FILE, 2-54
HSSerial:CHANnel:SPARameter:NCAScading:
AGGRessor[n]:SIGNal:PRBS, 2-55
HSSerial:CHANnel:SPARameter:NCAScading:
AGGRessor2[:ENABLE], 2-50
HSSerial:CHANnel:SPARameter:NCAScading:
DEEMbed, 2-56
HSSerial:CHANnel:SPARameter:NCAScading:
DRX[n], 2-57
HSSerial:CHANnel:SPARameter:NCAScading:
DTX[n], 2-58
HSSerial:CHANnel:SPARameter:NCAScading:
FILE, 2-59
HSSerial:CHANnel:SPARameter:NCAScading:
LAYout, 2-60
HSSerial:CHANnel:SPARameter:NCAScading:
RX[n], 2-61
HSSerial:CHANnel:SPARameter:NCAScading:
SSCHeme, 2-62
HSSerial:CHANnel:SPARameter:NCAScading:
STYPe, 2-63
HSSerial:CHANnel:SPARameter:NCAScading:
TX[n], 2-63
HSSerial:CHANnel:SPARameter:NCAScading:
TYPE, 2-65
HSSerial:CHANnel:TYPE, 2-65
HSSerial:COMPile, 2-66
HSSerial:COMPile:ATSequence:ENABLE, 2-66
HSSerial:COMPile:ATSequence:SEquence, 2-67
HSSerial:COMPile:ATSequence:TRACk, 2-67
HSSerial:COMPile:CANCel, 2-68
HSSerial:COMPile:CORRection:APPLy, 2-68
HSSerial:COMPile:CORRection:PATH, 2-69
HSSerial:COMPile:NAME, 2-69
HSSerial:COMPile:OPTions, 2-70
HSSerial:COMPile:OVERwrite, 2-71
HSSerial:COMPile:PLAY, 2-71
HSSerial:COMPile:REPeat, 2-72
HSSerial:COMPile:REPeat:AUTO, 2-72
HSSerial:COMPile:REPeat:MTYPe, 2-73
HSSerial:COMPile:SPUI, 2-74
HSSerial:COMPile:SRATe, 2-74
HSSerial:COMPile:STYPe, 2-75
HSSerial:DCD:ENABLE, 2-76
HSSerial:DCD:VALue, 2-76
HSSerial:DRATe, 2-77
HSSerial:ENCode:ENCo8b10b:DISParity, 2-77
HSSerial:ENCode:ENCo8b10b:ENABLe, 2-78
HSSerial:ENCode:PAM:LEVelS, 2-79
HSSerial:ENCode:PAM:NVALue, 2-79
HSSerial:ENCode:PWM:ENABLe, 2-80
HSSerial:ENCode:PWM:TMINor, 2-80
HSSerial:ENCode:SCHeme, 2-81
HSSerial:ISTate:OFFSet, 2-81
HSSerial:ISTate:VALue, 2-82
HSSerial:MARKer[n]:CLOCK:FREQuency, 2-82
HSSerial:MARKer[n]:CLOCK:TYPE, 2-83
HSSerial:MARKer[n]:ENABLe, 2-84
HSSerial:MARKer[n]:HLOW:TYPE, 2-84
HSSerial:MARKer[n]:TRIGger:LENGth, 2-85
HSSerial:MARKer[n]:TRIGger:TYPE, 2-85
HSSerial:MARKer[n]:TYPE, 2-86
HSSerial:NOISe:BW, 2-87
HSSerial:NOISe:ENABLe, 2-87
HSSerial:NOISe:LOCation, 2-88
HSSerial:NOISe:TYPE, 2-88
HSSerial:NOISe:VALue, 2-89
HSSerial:PJ[n]:ENABLe, 2-91
HSSerial:PJ[n]:FREQuency, 2-91
HSSerial:PJ[n]:MAGNitude, 2-92
HSSerial:PJ[n]:PHASe, 2-93
HSSerial:PJITter:DEViation:MDEViation, 2-89
HSSerial:PJITter:DEViation:TYPE, 2-90
HSSerial:PJITter:ICENable, 2-90
HSSerial:PREemphasis:COPReemp, 2-93
HSSerial:PREemphasis:DPreemph, 2-94
HSSerial:PREemphasis:ENABLe, 2-94
HSSerial:PREemphasis:TPreemph, 2-95
HSSerial:PREemphasis:TYPE, 2-95
HSSerial:PREemphasis:UNIT, 2-96
HSSerial:PREShoot:COPShoot, 2-96
HSSerial:PREShoot:ENABLe, 2-97
HSSerial:PREShoot:TYPE, 2-97
HSSerial:RESet, 2-98
HSSerial:RJ[n]:ENABLe, 2-99
HSSerial:RJ[n]:FREQuency:END, 2-100
HSSerial:RJ[n]:FREQuency:STARt, 2-100
HSSerial:RJ[n]:MAGNitude, 2-101
HSSerial:RJ1Cfactor:ENABLe, 2-98
HSSerial:RJ1Cfactor:TISigma, 2-99
HSSerial:RSEed:ENABLe, 2-102
HSSerial:RSEed:VALue, 2-102
HSSerial:SCRamble:ENABLe, 2-103

HSSerial:SCRamble:POLYnomial, 2-103
HSSerial:SCRamble:RINit, 2-104
HSSerial:SCRamble:RTYPE, 2-104
HSSerial:SCRamble:TYPE, 2-105
HSSerial:SRESponse:ENABLE, 2-105
HSSerial:SRESponse:FTIME, 2-106
HSSerial:SResponse:FTYPE, 2-106
HSSerial:SRESponse:RFTYPE, 2-107
HSSerial:SRESponse:RTIME, 2-107
HSSerial:SResponse:RTYPE, 2-108
HSSerial:SSC:DFDT, 2-108
HSSerial:SSC:DFDT:DURation, 2-109

HSSerial:SSC:DFDT:LOCation, 2-109
HSSerial:SSC:ENABLE, 2-110
HSSerial:SSC:EOPTION:ENABLE, 2-110
HSSerial:SSC:FREQUENCY:DEViation, 2-111
HSSerial:SSC:FREQUENCY:MODulation, 2-111
HSSerial:SSC:SHAPE, 2-112
HSSerial:SSC:SPRead, 2-112
HSSerial:SSC:USPRead:PERCentage, 2-113

W

WPLugin:ACTIVE, 2-114